S.S. Iyengar
Kenneth G. Furton
Pronab Mohanty
Naveen Kumar Chaudhary *Editors*

Artificial Intelligence Driven Forensics

Preliminary Draft of the Proceedings of AI-Enabled Forensic Investigations Network in Digital Sciences

Springer

S.S. Iyengar, Kenneth G. Furton
Pronab Mohanty, Naveen Kumar Chaudhary
Editors

Artificial Intelligence Driven Forensics

April 23, 2025

Springer

Editors

S.S. Iyengar
Florida International University
Miami, FL, USA

Kenneth G. Furton
Florida International University
Miami, FL, USA

Pronab Mohanty
DGP, Indian Police Service
India

Naveen Kumar Chaudhary
National Forensic Science University
Gandhinagar, Gujarat, India

This Springer imprint is published by the registered company Springer Nature Switzerland AG
The registered company address is: Gewerbestrasse 11, 6330 Cham, Switzerland

Preface

"The science of today is the technology of tomorrow."
— Edward Teller

Artificial Intelligence Driven Forensics explores the transformative role of artificial intelligence (AI) in the field of digital forensics, offering a comprehensive overview of how AI technologies are reshaping crime investigation, cybersecurity, and evidence analysis. The book bridges traditional forensic methodologies with cutting-edge AI techniques, such as machine learning, deep learning, natural language processing, and graph analytics, to improve the accuracy, speed, and scalability of forensic investigations.

Structured around real-world applications and case studies, the book covers critical areas including AI-enhanced malware detection, user behavior profiling, deepfake identification, social media profile cloning detection, canine forensics and hybrid approaches to analyzing digital artifacts. It also delves into the integration of AI with forensic tools for automating incident response, identifying anomalies in massive log files, and supporting decision-making in complex cybercrime scenarios.

A key feature of the book is its interdisciplinary approach, blending insights from computer science, cybersecurity, law, and ethics. It addresses not only the technical mechanisms behind AI-driven forensics but also the legal, ethical, and societal implications, particularly regarding privacy, accountability, and the admissibility of AI-generated evidence in courts.

Geared toward researchers, practitioners, students, and policymakers, *Artificial Intelligence Driven Forensics* serves as both a foundational text and an advanced guide to understanding the future of forensic science in the age of AI.

This book is partitioned into six thematic parts, each illuminating how artificial intelligence and machine learning are revolutionizing the field of digital forensics across domains including law, biometrics, cybersecurity, ecology, medicine, and more.

Part A: Artificial Intelligence and Machine Learning Driven Digital Forensics explores the transformative role of AI and machine learning in modern forensic science. It introduces foundational theories, bench marks advanced algorithms, and demonstrates practical implementations of intelligent systems across various forensic domains. Topics include anomaly detection in visual scenes, synthetic cybersecurity log generation, multimedia authenticity verification, and adversarial attack detection in facial recognition systems. This section highlights how AI enhances forensic accuracy, scalability, and adaptability, establishing a data-driven foundation for next-generation forensic investigations.

Part B: Bridging Law and Technology: AI-Enabled Legal Framework addresses the intersection of artificial intelligence and legal forensics. It outlines foundational principles, emerging applications, and a comprehensive techno-legal framework for ensuring the integrity and admissibility of digital evidence in courtrooms.

Part C: Behavioral and Biometric Forensics focuses on advanced biometric techniques and behavioral modeling, presenting innovations in keystroke dynamics, gait recognition, fingerprint analysis, and emotion detection using multimodal deep learning approaches.

Part D: Green and Medical Forensics Applications brings environmental sustainability and healthcare into forensic science. It explores green nanoparticles in evidence analysis, AI-driven cancer pathway prediction for medical forensics, and blockchain-secured cognitive IoT frameworks for protecting sensitive medical imagery.

Part E: Network, Supply Chain, and Cyber Forensics navigates the complexity of cybercrime investigations by introducing frameworks for packet capture analysis, blockchain-enhanced supply chain tracing, and forensic profiling based on user browsing behavior.

Part F: Forensics in IoT, Wearables, and Environmental Surveillance expands forensic frontiers into wearable devices, smart therapeutics, and broadband connectivity. This section investigates how data from smartwatches, AI-based health systems, and network infrastructures contribute to digital evidence collection and event reconstruction in real time.

Through its diverse and interdisciplinary lens, this book serves as both a reference and inspiration for researchers, practitioners, and policymakers advancing the field of digital forensics in the age of intelligent technologies.

Miami, FL, USA — S.S. Iyengar
Miami, FL, USA — Kenneth G. Furton
Bangalore, Karnataka, India — Naveen Kumar Chaudhary
Gandhinagar, Gujarat, India — Pronab Mohanty

Acknowledgments

The editors extend their sincere gratitude to the Honorable Vice Chancellor of the National Forensic Sciences University (NFSU), Dr. Jayant Vyas, and to Chancellor of PES University Dr. Jawahar Doreswamy for their unwavering support and encouragement in the development of this work.

We are also thankful to Dr. Kenneth A. Jessell, President Emeritus, Senior Vice President and Chief Administrative Officer of Florida International University, Interim Dean of the College of Engineering Dr. Ines Triay and Director of KFSCIS Dr. Jason Liu for their strong and continued support

Their contributions have been instrumental in the establishment of the Forensic Investigations Network in Digital Sciences (FINDS)— first in the world pioneering Center of Excellence focused on AI-Driven Digital Forensics, funded by the U.S. Army Research Office (Grant No. W911NF-21-1-0264). Special Thanks to Dr. Paul Yu and Dr. Owens from U.S. Army Research Office and Dr. Cliff Wang from National Science Foundation for their continued support. Many graduate and undergraduate students assisted in the preparation of this manuscript. Notably, Yashas Hariprasad, Sina Nabavi and other students from FIU Knight Foundation School of Computing and Information Sciences, FIU.

Finally, we commend the authors of this book for their outstanding contributions. Their chapters offer innovative insights into the emerging field of AI-driven forensics and set a high standard for future scholarship in this critical area.

Miami, FL, USA	S.S. Iyengar
Miami, FL, USA	Kenneth G. Furton
Bangalore, Karnataka, India	Naveen Kumar Chaudhary
Gandhinagar, Gujarat, India	Pronab Mohanty

April 23, 2025

Contents

Part C. Behavioral and Biometric Forensics

Part D. Green and Medical Forensics Applications

About the Editors

Dr. S.S. Iyengar is a distinguished computational scientist, entrepreneur, advisor and mentor to millions of scientists and sits on the boards of companies, including startups through his India-USA based foundation. He was honored as the ‘Founding Father of AI Science in Digital Forensics’ by the Soft Computing Research Society in February 2025. This prestigious recognition is a testament to his lifelong contributions to artificial intelligence, cybersecurity, and digital forensics. He was recently awarded the 2025 Distinguished Career Award in Computer Science and Fellow of the Washington Academy of Sciences. He is also known as the 'Founding Father' for his contributions to the development of Information Science Technology for Forensic Application (2022) by a prestigious European University. He is also recognized as the World’s Most Influential Scientific Minds in Computational Science. He is currently the Distinguished University Professor, Founding Director of the Discovery Lab and Founding Director of the World’s First US Army-funded Center of Excellence in Digital Forensics at Florida International University, Miami. He is also the Distinguished Chaired Professor at National Forensics Sciences University, Gandhinagar, India. He was awarded the 2023 Karnataka Rajyotava Award (Karnataka State’s 2nd Highest Civilian Award) on November 1st, 2023. He has been involved with research and education in high-performance intelligent systems, Data Science and Machine Learning Algorithms, Sensor Fusion, Data Mining, and Intelligent Systems. Since receiving his Ph.D. degree in 1974 from Missi. State Univ., USA, he has directed over 65 Ph.D. students, many number of postdocs, and many research undergraduate students who are now faculty at Major Universities worldwide or Scientists or Engineers at National Labs/Industries around the world.

He has published more than 600 research papers, has authored/co-authored and edited 32 books and hold various patents. Over the lifetime, his work, Brooks-Iyengar Algorithm has over 5223 publication(s) within this topic and has received 138,976 citation(s). The topic is also known as: Brooks–Iyengar hybrid algorithm. His h-index is 68 and is identified among the top 2% cited scholars and world scientists (from Stanford University and EBMs of JSAN journal. The book titled “Fundamentals of Brooks-Iyengar Distributed Sensing Algorithm” authored by Prof. Pawel (Poland) and others and published by Springer in 2020 celebrates S.S. Iyengar's accomplishments that led to his 2019 Institute of Electrical and Electronics Engineers' (IEEE) Cybermatics Congress "Test of Time Award" for his work on creating Brooks-Iyengar Algorithm and its impact in advancing modern computing. His work has been featured on the cover of many scientific journals like IEEE transactions and the National Science Foundation’s breakthrough technologies report to the US Congress by his research group in both 2014 and again in 2016.

He has served on many scientific committees and panels worldwide and has served as the editor/guest editor of various IEEE and ACM journals. His books are published by MIT Press, John Wiley and Sons, CRC Press, Prentice Hall, Springer Verlag, IEEE Computer Society Press, etc. One of his books titled "Introduction to Parallel Algorithms" has been translated into Chinese.

During the last thirty years Dr. Iyengar has brought in over 65 million dollars for research and education. More recently in Spring 2021, Dr. Iyengar in collaboration with HBCUs were awarded a $2.25 M funding for setting up a Digital Forensics Center of Excellence over a period of 5 years (2021-2026). He received an honorary Doctor of Science for 4 times and recently from Poznan University of Technology in Poland in May 2023. He has been awarded the Lifetime Achievement Award 3 times (INTERPOL, BHU, IEEE) and recently for his contribution to the for his contribution to the field of Digital Forensics on November 8, 2022, during the 7th INTERPOL DIGITAL FORENSICS EXPERT GROUP (DFEG) MEETING at National Forensics Sciences University, Gandhinagar, Gujarat, India. He has provided the students and faculty with a vision for active learning and collaboration at Jackson State University, Louisiana State University, Florida International University, and across the world.

Dr. Iyengar is a Member of the European Academy of Sciences, Member of the European Academy of Arts and Sciences, a Life Fellow of the Institute of Electrical and Electronics Engineers (IEEE), a Fellow of the Association of Computing Machinery (ACM), a Fellow of the American Association for the Advancement of Science (AAAS), a Fellow of the Society for Design and Process Science (SDPS), a Fellow of the National Academy of Inventors (NAI), and a Fellow of the American Institute for Medical and Biological Engineering (AIMBE). He has received various national and international awards including the crowning Test of Time Research (for his seminal work which has impacted billions of computer and internet users worldwide) and Scholarly Contribution Award from 2019 IEEE Congress on Cybermatics, the distinguished Fulbright Scientist, the Times Network NRI (Non-Resident Indian) of the Year Award for 2017, IEEE Meritorious Service award, most distinguished CVR Award at the Society for Design and Process Science (SDPS 2017), Innovation-2-Industry Award, LSU Distinguished Rain Makers for Leadership and Research Award, World's Best Technology Showcase award, Technology Innovation Award Louisiana Tech University Research Foundation Inventor Award, Distinguished LSU Research Master Award, IBM Distinguished Faculty Award, and the NRI Mahatma Gandhi Pradvasi Medal at the House of Lords in London in 2013 among others. He was awarded Satish Dhawan Chaired Professorship at IISc, then Roy Paul Daniel Professorship at LSU. He has received the Distinguished Alumnus Award of the Indian Institute of Science. In 1998, he was awarded the IEEE Computer Society's McCluskey Technical Achievement Award and is an IEEE Golden Core Member. Professor Iyengar is an IEEE Distinguished Visitor, SIAM Distinguished Lecturer, and ACM National Lecturer. In 2006, his paper entitled, A Fast-Parallel Thinning Algorithm for the Binary Image Skeletonization, was the most frequently read article in the month of January in the International Journal of High-Performance Computing Applications. His innovative work called the Brooks-Iyengar algorithm along with Professor Richard Brooks from Clemson University is applied in industries to solve real-world applications. Dr. Iyengar's work had a big impact; in 1988, when he and his colleagues discovered "NC algorithms for Recognizing Chordal Graphs and K-trees" [IEEE Trans. on Computers 1988]. This breakthrough result led to the extension of designing fast parallel algorithms by researchers like J. Naor (Stanford), M. Naor (Berkeley), and A. A. Schaffer (AT&T Bell Labs).

His research has been funded by National Science Foundation (NSF), Defense Advanced Research Projects Agency (DARPA), Multi-University Research Initiative (MURI Program), Office of

Naval Research (ONR), Department of Energy / Oak Ridge National Laboratory (DOE/ORNL), Naval Research Laboratory (NRL), National Aeronautics and Space Administration (NASA), US Army Research Office (URO), and various state agencies and companies. He has served on US National Science Foundation and National Institute of Health Panels to review proposals in various aspects of Computational Science and has been involved as an external evaluator (ABET-accreditation) for several Computer Science and Engineering Departments across the country and the world. Dr. Iyengar has also served as a research proposal evaluator for the National Academy. Dr. Iyengar has been a Visiting Professor or Scientist at Oak Ridge National Laboratory, Jet Propulsion Laboratory, Naval Research Laboratory, and has been awarded the Satish Dhawan Visiting Chaired Professorship at the Indian Institute of Science, the Homi Bhabha Visiting Chaired Professor (IGCAR), and a professorship at the University of Paris-Sorbonne.

Dr. Kenneth G. Furton is the executive director of the Global Forensic and Justice Center and the chief scientific officer of Florida International University. He is a distinguished university professor in the Department of Chemistry and Biochemistry and a world-leading scholar in forensic chemistry focused on trace detection and olfaction. From 2007 to 2014, he served as dean of the College of Arts and Sciences, which he reorganized into three mission-based interdisciplinary schools to address some of the biggest issues facing society, raising $50M in philanthropic gifts and doubling research funding to $60M annually. From 2014 to 2022 he served as provost, executive vice president and chief operating officer of FIU where he has led the development and implementation of two FIU strategic plans helping to create a dynamic, results-oriented university with dramatic improvements in student success and research preeminence including simultaneously doubling the four-year graduation rates to over 60%, doubling research expenditures to $250M annually and a 30 fold increase in patents to more than 60 annually achieving a Top 20 patent ranking. He also helped secure over $100 million in philanthropic gifts and FIU moved from #8 to #1 in the state of Florida performance-based funding rankings. Under his leadership FIU was the most improved R1 university in the nation in U.S. News rankings, improving 54 spots in five years to No. 78 public and ranked No. 17 in Innovation and No. 5 in Social Mobility. Deeply committed to the academy, Dr. Furton has supervised the research of more than 140 students and been continuously funded for more than three decades, totaling more than $10 million in grants. He has 26 patents, 2 books and 221 peer-reviewed publications with more than 8,000 citations and an h-index of over 50. He is a member of Phi Beta Kappa and an elected fellow of the National Academy of Inventors as well as the American Academy of Forensic Sciences. He has received significant attention in recent years for ground-breaking work using dogs and sensors to detect humans, drugs, currency, accelerants, explosives, mass storage devices, invasive species and medical conditions including the coronavirus disease. His researching and deployment of COVID-19 detector dogs during the global pandemic reached an audience of over 2 billion worldwide.

Dr. Pronab Mohanty is a distinguished officer of the Indian Police Service (IPS) – 1994 batch, currently serving as the Director General of Police (DGP) – Cyber Economics and Narcotics (CEN) and Chief Information Security Officer (CISO) for Karnataka State Police. With over three decades of exemplary service, he has led critical law enforcement and cybersecurity initiatives in India, holding key positions in the Central Bureau of Investigation (CBI), the Ministry of Electronics and Information Technology (MeitY), the Unique Identification Authority of India (UIDAI/AADHAAR), and the Border Security Force (BSF).

Dr. Mohanty has an extensive background in investigating technologically complex cybercrimes, implementing advanced cybersecurity procedures in sensitive data centers, conducting forensic audits of data breaches, privacy, and consent management, and investigating serious corporate and financial frauds, human trafficking, and crimes against minors.

A strong advocate for modernizing police IT services, he has played a crucial role in developing and securing India's national cyber architecture, leading the National Cyber Coordination Center (NCCC) and ensuring the security of critical Information and Communications Infrastructure in the BSF. His expertise in law enforcement modernization, cyber forensics, and advanced computing security has positioned him as one of India's foremost experts in cybercrime prevention and digital forensics.

Dr. Naveen Kumar Chaudhary has been a professor of Cyber Security at the National Forensic Sciences University in Gandhinagar, Gujarat, India since 2019. He is also a Courtesy Research Professor at the Knight Foundation School of Computing and Information Sciences at Florida International University, Miami, Florida. He holds a Bachelor of Technology degree in Information Technology & Telecommunication Engineering and Master of Engineering degree in Digital Communication. He earned his Ph.D. in Engineering and advanced certifications in Cyber and Network security.

His extensive experience spans more than 25 years in engineering education, research, and government. He has steered many cutting-edge ICT projects and worked extensively on policy formulation in cybersecurity and e-governance. He is the recipient of a letter of appreciation for contribution towards the cause of literacy from Brent St. Denis, MP, Algoma Canada, in 1994. Dubai, SEWA award for his contribution to Cyber Security education in 2022. He also received COAS and CCOSC Commendation in 2009 and 2015, respectively for his innovation and distinguished service. He is an IEEE senior member and life member of IETE.

Principles of Machine Learning Theories and Practical Applications *

Nageswara S. V. Rao[1]

Oak Ridge National Laboratory, Oak Ridge, TN USA
raons@ornl.gov

Abstract. Machine learning (ML) computations of increasing sophistication and complexity are being developed to solve complex, data-driven problems in diverse areas. Their output is often subject to undesirable phenomena such as overfitting and hallucinations that are hard to detect, and more generally their scientific rigor is hard to establish. We propose the concept of ML-solvability by combining the theories of learnability, computing and logic, which characterizes the model space, the learning algorithm that estimates a model using samples, and the inference algorithm that utilizes the model. It provides insights into the applicability and generalization of ML codes, and the possibility of incomplete and unsound inferences if the underlying problem is not ML-solvable. In science areas, the long-established laws are synergistically exploited to sharpen and compose powerful ML solutions with provable generalization and correctness properties. We describe a framework for ML-solvability and generalization analyses based on a combination of physical laws that govern systems and information laws that characterize the learning processes. We combine the learning dimension and training error to derive generalization equations to detect and minimize overfitting, and utilize physical law violations by learning processes to identify and mitigate inference inadequacies. We briefly describe the uses of smooth, nonsmooth and algebraic forms of laws to develop or analyze ML solutions in the areas of data transport networks, nuclear engineering, and computer system diagnosis.

Keywords: Machine Learning · Generalization Theory · ML-Solvability · Overfitting · Hallucinations · Regression · Information Fusion

1 Introduction

The machine learning (ML) methods are increasingly being applied to solve complex, computational and data-driven problems in diverse science areas, including energy systems, materials research, and science infrastructures. Their scientific validity critically

*This work is funded by RAMSES project and Applied Mathematics program, Office of Advanced Computing Research, U.S. Department of Energy, and performed at Oak Ridge National Laboratory managed by UT-Battelle, LLC for U.S. Department of Energy under Contract No. DE-AC05-00OR22725.

depends on the correctness of their codes, which is characterized by computability, learnability, and guaranteed approximations under the randomness of processes, measurements, and errors. Particularly in science areas, it is very important to ensure that the underlying problem is correctly analyzed in order to develop and apply ML codes along with their analytical proofs and experimental validation methods. Specifically, if these codes are incorrectly developed or applied to uncomputable or unlearnable problems, the consequences can be severe: their outputs could be unsound or incomplete or both [63, 36], thereby resulting in potentially false "discovery claims" or incorrect critical decisions. Such scenarios are more likely to arise now as increasingly complex tasks are being delegated to purely computational means, fueled in part by the recent proliferation of generative, black box artificial intelligence (AI)/ML methods. These aspects are further compounded by the increasing complexity of diverse ML methods that are being customized, sharpened, and composed in unprecedented ways to exploit extensive data sets and extremely powerful computing resources.

In science areas over the decades, physical laws have been developed, often in analytic (differential, algebraic and stochastic) forms, from the first principles to design, understand, and operate complex systems. More recently, they have been exploited to develop general physics informed machine learning (PIML) methods [65], and also to customize ML solutions for reactor system analytics [40], electrochemistry laboratory automation [1], methane hydrate exploration [54], and other applications. For example, a three-level fuser of smooth and non-smooth machine learners that reflects the structure of the coolant system of a nuclear reactor has been developed for power level estimation using independent monitoring measurements [55, 40]. In another example, custom ML methods are integrated into automated electrochemistry workflows to verify the consistency of measurements with normal experimental conditions and to detect failure conditions [1]. In another direction involving "non-physical" systems, abstract laws are developed and incorporated into ML solutions for performance profiling of data transport networks. The concave-convex profiles of data transport complexes and their individual sites, subsystems (e.g., file and IO), and components (e.g., TCP version) lead to the diagnosis of bottlenecks [45, 46] and compromises [48], and optimization of throughput across the complex [21, 29]. In general, physical and abstract laws encapsulate the critical knowledge – often in compact differential and algebraic forms – that complements datasets and ML models obtained from them. They have been synergistically exploited to compose powerful ML solutions, and also to derive generalization equations, for example, distribution-free error confidence bounds for sensor drift estimation in primary coolant systems [51, 50] and concave-convex profile transitions in throughput profiles of data transport infrastructures [56]. In this paper, we briefly describe these applications, and place them within a broader context of ML solutions and their applicability to a variety of problems, by utilizing theories of computing, learning, and logic.

We introduce the concept of *ML-solvability* which characterizes the correctness of ML codes in terms of deterministic proofs and probabilistic generalization guarantees, by combining the theories of learnability [64, 24], computing [14] and algorithms [13, 20], and logic [63]. It characterizes a ML solution by its model space, the algorithm that learns a model using samples, and the inference algorithm that utilizes the trained

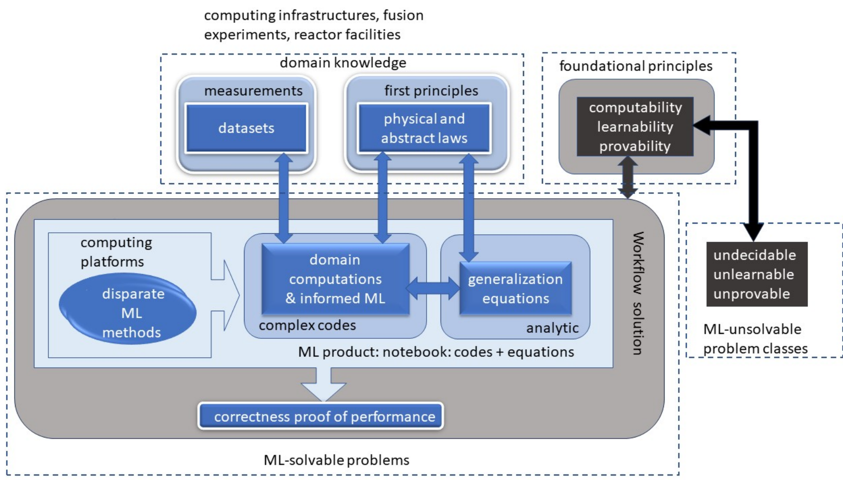


Fig. 1: *Framework for co-developing ML codes with generalization equations.*

model, and computing platforms that run learning and inference codes. It provides insights into the applicability and generalization properties of ML codes, and potential for incomplete and unsound inferences if the underlying problem is not ML-solvable. We combine the generalization theory with algorithm correctness proofs to provide performance guarantees based on finite samples using adaptations of Vapnik-Chervenenkis (VC) dimension (or a related quantity) [6, 64]. Furthermore, we exploit the conditions and constraints from physical-abstract laws of the underlying system to provide practical insights and results, under the framework illustrated in Fig. 1. We first apply the foundational principles to establish the ML-solvability of a given problem, and then exploit the domain laws to identify or develop custom ML solutions. If the problem is ML-unsolvable or its ML-solvability cannot be established, then we show that the properties of ML codes are uncertain even if they produce output.

We briefly describe illustrative ML applications from existing works in the areas of data transport networks, nuclear engineering, and computer system diagnosis to illustrate various aspects of the ML-solvability concept. Specifically, we combine the learning dimension and training error to derive generalization equations to detect and minimize overfitting, and utilize physical law violations by the learning processes to identify and mitigate the unsound inferences. We also illustrate the utilization of smooth, non-smooth and algebraic forms of laws to develop or analyze ML solutions in these applications.

This paper is organized as follows. The ML-solvability concept is developed in Section 2. The generalization theory is briefly summarized in Section 3. The algebra and calculus aspects of machine learners are described in Section 4. Implications of the lack of ML-solvability or its knowledge are described in terms of overfitting and unsound and incomplete inferences in Section 5. The applications based on existing ML solutions are briefly described in Section 6. Conclusions and possible directions for future investigations are described in Section 7.

2 ML-Solvability: Computability and Learnability

ML methods are typically developed to estimate parameters or infer properties, and optimize parameters of an underlying system, using probabilistic measurements. Their correctness embodies two parts: (a) deterministic proofs from theories of computations and algorithms that establish their correctness and complexity, and (b) probabilistic guarantee of their approximation from theory of learnability that establish their generalization properties. Within the broad spectrum of science areas, some problems would be computationally undecidable (or uncomputable for short) [62] or unprovable by pure mechanical means [17], and some learning tasks would be unlearnable in an information theoretic sense [64, 24]. *We define a problem to be* ML-Solvable *if it is both computable and learnable with the required level of confidence*. Then, there are problems that are not ML-solvable, including, self-diagnosis of computing systems under arbitrary failures [35], detection of computer viruses [12], and inference of chaotic maps from arbitrary time sequences [25], as will be discussed subsequently. On the positive side, several tasks formulated within the constraints of physical and abstract laws are ML-solvable, as exemplified by the sensor drift estimation in nuclear reactors [39], and optimization of data transport infrastructures [44]. We focus on general classes of such ML-solvable problems, and sharpening of their solutions by exploiting the information from physical and abstract laws. Specifically, for ML-solvable problems, we briefly describe solutions and their correctness and generalization (Section 6).

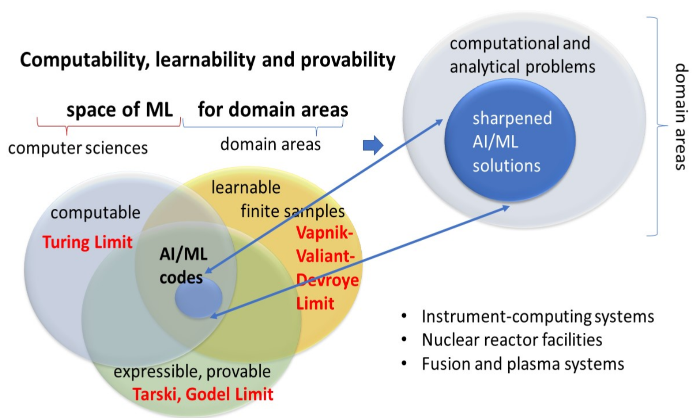


Fig. 2: *ML codes must be developed and applied to computable and learnable problems, and additionally may need to be expressible and provable.*

2.1 Requirements of Provability and Expressibility

Complex ML solutions that require performance guarantees and interpretable parameters require two additional properties of provability and expressibility, in addition to learnability and computability of basic ML-solvability, as shown in Fig. 2. Learnability is defined by generalization properties such as finite scale-sensitive dimensions [2], as discussed subsequently in detail (Section 2.4). Computability in ML requires that the underlying computing problem be decidable in a Turing sense [14, 27], which is also closely related to the mechanization of proofs as discussed in detail in Section 2.5. Expressibility of ML refers to the ability of learned models or parameters to represent critical domain properties, such as learning information that neural network weights capture feature correlations or hidden nodes represent critical regions of the image being processed. Expressibility of ML outputs is critical for trusting black-box solutions, and they are subject to Tarski's limit [60, 26], in particular solutions that require that critical properties be defined first in complex algebraic systems. In simple terms, Tarski's undefinability states that in any sufficiently strong formal system, not all truths in that standard model can be defined within the system, which in turn implies that examples provided within the context of a system are not guaranteed to lead to the discovery of all truths about that system. The undefinability theorem shows that no sufficiently rich interpreted language can represent all of its own semantics. A corollary is that any metalanguage capable of expressing the semantics of some object language must have expressive power exceeding that of the object language. The metalanguage includes primitive notions, axioms, and rules absent from the object language, so there are theorems that are provable in the metalanguage that are not provable in the object language. Consequently, there could be problems formulated to "learn" such meta-properties that are beyond the power of ML methods. On first glance, these problems might appear too abstract, but it is instructive to note that several practical problems such as virus detection [12] and ensuring the resilience of complex codes [35] are undecidable, and hence their analogs are also beyond the power of ML solutions.

2.2 Composition of Machine Learning Solution

ML solution to a problem is composed of the following components, which play different roles in ML-solvability:

- *Model or Hypothesis Space:* The hypothesis or model space consists of all hypotheses or models, from which one is learned based on samples. The examples include the class of deep neural networks or collection of inference rule sets. The hypothesis space must be large enough to closely approximate the underlying model, which corresponds to high approximation density. On the other hand, a large model space also increases its learning dimension which decreases it generalization capabilities. These relationships and trade-offs are explicitly represented using the generalization equations described in Section 3.
- *Learning Algorithm:* A learning algorithm chooses a model or hypothesis from its class. The examples include the training algorithm to determine the weights of a neural network. In addition to computability, the computational complexity of the training problem determines the quality of model selection. For example, the problem of optimizing neural network weights based on a sample is NP-hard, and methods such as backpropagation only guarantee approximations in practice.
- *Training and Testing Data:* The training and testing samples are used by the learning algorithm. These samples may be limited by their availability, and cost of collecting them. Large samples often lead to improved ML solution but may lead to larger training times. Also, the extraction of sample data sets from complex sys-

tem measurements might require extensive processing, for example, extraction of features from images to be used for classification.

– *Learned Model and Inference Algorithm:* The learned model or hypothesis is used by the inference algorithm to produce output such as classification or parameter estimate. This step is often called the forward computation, and is polynomial-time computable for a neural network. For cases where the learned model is a collection of inference rules, the computation may involve selective application of rules, which could be computationally intractable or undecidable.
– *Generalization Equations and Proofs:* Analytical characterizations such as generalization equations, expressions of properties, and proofs of assertions are important desirable qualities to be established. In complex cases, such proofs are beyond the power of computations, for example, as a consequence of Rice theorem as described in Section 6.3.
– *Computing Platforms:* Computing platforms used to execute learning and inference algorithms play a significant role in the quality of ML solution. In particular, LLMs with a large number of parameters and large data sets require supercomputers and specialized hardware such as GPUs.

2.3 Finite Sample Implications on ML Solutions

Even for ML-solvable problems, for example some classification problems, there are fundamental limits on how general such methods can be and what can be inferred about them, due to *finite* number of samples. The incompleteness or slow convergence theorem of Devroye [15] shows that there is *no universally best machine learning method* that can be asserted using a finite number of samples. In essense, ML methods trained on finite samples do not have sufficient information to optimize over the non-countably infinite class of data distributions. Indeed, any ML method is reduced to a random guesser for certain data distributions that are tailored to the method, even if it performs optimally on a given training dataset. Consequently, ML method optimal for a domain (with its specific distribution) may perform poorly when applied to another domain which corresponds to another distribution for which it is nearly a random guesser; in particular, LLM models' excellent performance under training is not guaranteed when applied to a different domain.

As a practical consequence, diverse ML methods – smooth, non-smooth, structural and statstical [36] – will continue to be developed as newer learning problems arise in various disciplines. Collectively, they provide a set of rich, diverse ML solutions, but their performance can vary widely even for a single problem. Information fusion methods that combine multiple disparate ML solutions to ensure performance at least as good as their best subset have been developed using the projective fusers that utilize the lower envelope of error surfaces [34]. Furthermore, these fusers can be retrained to incorporate newer methods using training samples, which gives us a practical solution to continually fold in the power of newer ML methods. The fused solution is still subject to slow the convergence theorem but is effective in practice since it makes best use of the methods as they become available.

Another limitation of an ML solution is the computational intractability or NP-hardness [16], even if the ML problem is computable or Turing decidable; for example, training a three-node neural network is NP-complete [5]. Thus, practical ML computations can only guarantee approximations. For example, different invocations of the back propagation algorithm yield different approximations in sigmoidal neural networks. A nearest neighbor fuser combines these estimates to provide a solution superior to all of them [34]. In general, for a fuser class $\mathscr{F}_F$ that combines estimators $\hat{f}_i \in \mathscr{F}_i, i = 1, 2, \ldots, N$ and satisfies the isolation property, namely, containing functions that transfer individual inputs to output [32], we have $I(f_F^*) = \min_i^N I(\hat{f}_i) - \Delta$, where $I(f)$ is the expected error of estimator f, and Δ is the improvement in the expected error achieved by the optimal fuser f_F^* over the best estimator. For the learned fuser $\hat{f}_F$ that minimizes the training error, the confidence function δ_V provides the probability that the error is within ε of the optimal, specified by its generalization equation described in Section 3. It is expressed in terms of $V = \sum_{i=1}^{N} V_i$, where V_i is the total variation of $\mathscr{F}_i$, and improves with the training sample size independent of the underlying distribution [34], which is complex and often unknown. These aggregation operations over ML models provide a basis for abstract additive operations that are rigorously supported by distribution-free guarantees [64], as described in Section 4.

2.4 Physical and Abstract Laws

The physical and abstract laws capture the critical knowledge of the underlying systems, often in compact forms, by abstracting important parameters and relationships. Data sets collected from operational or emulated systems and ML models obtained from them typically provide complementary knowledge, often reflecting measurement uncertainties and errors and finite sample limits. They have been synergistically exploited to compose powerful ML solutions to estimation and diagnosis problems in a variety of ways, ranging from incorporating physical constraints, utilizing physical law violations as objective functions to minimize, structuring ML solutions to reflect physical laws, and combining multiple disparate estimators of parameters. In particular, the physical and abstract features, along with their relationships, captured by laws play a crucial role in the design and analysis of ML solutions to the underlying estimation and diagnosis problems in the two types of systems, namely, reactor coolant systems and computing-data complexes. ML solutions may be based on utilizing the additive and differential knowledge between ML models and their counterpart first principle laws: solutions to estimation problems, including classification and regression problems, typically utilize the additive knowledge, and those to diagnosis problems typically utilize the differential knowledge. Furthermore, the analytical properties such as bounded derivatives of smooth laws and bounded total variations of non-smooth laws may be combined with empirical errors of ML models to derive performance characterization and guarantees in the form of generalization equations with practical insights [37]. Specific cases and development of the underlying concepts behind the additive and difference operations in terms of algebra and calculus of ML within our applications context are described in Section 4. We consider ML problems formulated within the context of a system S characterized by:

(i) *law* $\mathscr{L}_S(P,Z)$ that relates the parameters and system variable vectors, P and Z, respectively, and
(ii) *structure* $\mathscr{S}_S(\mathscr{C}_S)$ that specifies the relationships between its set of components $\mathscr{C}_S$.

2.5 Mechanization of Proofs: Properties of Programs

Traditionally, the correctness proofs of codes are manually generated, and their mechanization refers to utilizing computer codes to generate them, possibly by automatic theorem provers or more recent generative AI methods. Such mechanized proofs are subject to limits guided by the Turing undeciability and Godel's incompleteness considerations [35]. The computation and estimation tasks of ML-solvable problems must be Turing decidable. Rice Theorem [14] establishes the computability limits of using codes to infer properties of codes, in particular their correctness proofs; for example, it shows the undecidability of resilience detection problems [35]. The correctness proofs of ML codes are subject to Godel's incompleteness results on provable assertions, namely, that not all truths can be provable in certain systems, such as arithmetic with multiplications [17]. Intuitively, the truths can not be enumerated as deductions from axioms for verification via a mechanical process, which is subject to Turing's undecidability [62] (incidentally, Godel's results on arithmetic systems were published in 1931, years before Turing's results on computations in 1936). The notions of undecidability (Turing [62]) and unprovability of assertions about their outputs (Godel [17]) are closely related as pointed out by Chaitin [9]: informally, they both capture the recursiveness needed for algorithms and proofs, which is insufficient to address non-recursiveness of certain tasks and assertions, respectively. The work of Uspensky [63] makes this connection explicit by casting the provability task in terms of algorithms within a deductive system; specifically, the non-existence of such an algorithm establishes the non-provability of truths.

Time series data may also reflect physical properties captured by the domain or ML computations, for example, dynamic flow computations. They introduce another source of undecidability in physical systems with chaotic dynamics [25], wherein nearly all questions about long-term trajectories are undecidable even when initial conditions are known. Thus, even within the context of these physical systems, ML solution that identifies such properties is not feasible, independent of how large the training sets and parameter spaces are. In scenarios of reactor systems (Section 6.2), the dynamics are stabilized to avoid chaotic regimes, which leads to their learnability by avoiding this type of undecidability [40, 39]. In general, learnability can be undecidable in that it is not verifiable mechanically as shown recently [4].

The undecidability results are manually proved within the frameworks of recursive functions expressed in L [14], Turing machines [11], lambda calculus [9], and others [63]. Several of the well-known undecidable problems are decision problems about Turing machines such as the halting problem, empty-set detection and equivalence of Turing machines [11]. These problems might appear somewhat abstract, but there are a number of more "practical" undecidable problems, including virus detection problems [12], programs to test randomness of a string [9], testing the equivalence of context-free grammars, smallest program capable of generating a given string, and computing the Kolmogorov complexity of strings. Resilient computations for Exascale systems represent another class of such challenging problems, and various versions are shown to be undecidable [35] as described in Section 6.3.

3 Generalization Theory

Under the ML paradigm, a machine "learns" a functional relationship between two vector random variables [19, 49] using a random sample $(X_1,Y_1),(X_2,Y_2),\ldots,(X_n,Y_n)$, typically drawn from an unknown joint distribution $\mathbb{P}_{X,Y}$ of feature vector X and output vector Y. The task is to compute a predictor function f from a complex class (e.g., deep neural networks) such that $f(X)$ is a good estimate of Y overall. A fundamental generalization result establishes that, under certain conditions, the best estimator f^* that minimizes $I(f) = E\left[C\left(f(X),Y\right)\right]$, for cost function $C(.,)$, can be closely approximated with high probability by an $\hat{f}$ learned solely from the sample regardless of $\mathbb{P}_{X,Y}$ [64]. Vapnik's theory [64] establishes that a "suitable" estimator, $\hat{f}$, obtained by an ML method M, ensures

$$\mathbb{P}^l_{X,Y}\left[I(\hat{f}) - I(f^*) > \varepsilon + \hat{\varepsilon}\right] < \hat{\delta}_M\left(\mathscr{F}_M, \varepsilon, \hat{\varepsilon}, l\right), \tag{1}$$

where $\mathscr{F}_M$ is its function class, $\varepsilon > 0$ is the *precision* parameter, $0 < \delta_M(.) < 1$ is the *confidence* function, and $\hat{\varepsilon}$ is the *training error* associated with computing $\hat{f}$. Thus, it provides distribution-free guarantees on their prediction performance on future data, and ML-solvability requires that $\hat{f}$ must be computable as a minimum, which is possible only when the underlying problem is computable or Turing decidable.

Over the past decades, finite sample performance guarantees are derived in a variety of methods, including neural networks [2], regression trees [7], support vector machines [64], and kernel estimates [58]. Historically, these elegant generalization equations are considered too loose to be practical, but more recently a combination of custom parameters (e.g. small Lipschitz constant and total variation) and the availability of data sets (e.g. 2K Gamma spectrum every second) made them useful in practice, for example, as shown in Figure 4. The estimate $\hat{\delta}_M(.)$ for the composed ML solution M can be based on law $\mathscr{L}_S$ or its structure $\mathscr{S}_S(\mathscr{C}_S)$ in a generic form, and its practical versions for application scenarios are discussed in Section 6. In addition to specific values, the mathematical form of $\hat{\delta}_M(.)$ provides critical practical information by relating the parameters of $\mathscr{L}_S$ to precision and confidence of the composed ML solution.

If $\mathscr{F}_M$ has finite capacity [64], then under bounded error, the condition in Eq (1) is guaranteed; a more general result ensures this condition under finite scale-sensitive dimensions [3]. For sigmoid neural networks, the sample size needed to ensure Eq (1) is linear in the number of neural network parameters [30], as opposed to having the quadratic dependence of previous bounds for unbounded weights [22]. In particular, the confidence estimate is

$$\hat{\delta}_{NN} = 8\left(\frac{32W}{\varepsilon}\right)^{h(d+2)} e^{-\varepsilon^2 l/512}$$

for a sigmoid network with h hidden nodes and input dimension d with weights suitably bounded by W. It is important to note that several statistical estimates learned by current ML methods are smooth, including SVM with Gaussian kernels [58] and radial basis functions [8], and several variables and parameters in science applications are bounded, and these properties are used to develop generalization equations. ML methods also employ non-smooth methods, such as ensemble tree methods [61], regression trees [7], Haar estimators [10], and Nadaraya-Watson estimators [28]. In practice, the parameters are bounded, and the learned functions have a finite (often small) number of jumps, which leads to their bounded finite total variation $V < \infty$. In this case, the confidence estimate is

$$\hat{\delta}_V = 8g\left(1 + \frac{128V}{\varepsilon}\right)e^{-\varepsilon^2 l/2048}.$$

Quantum communications channel modeling is carried out using simulations and experiments. These problems are typified by the algebraic structure of $\mathscr{F}_M$ that supports optimization for improved simulation-experiment combination; for example, when $\mathscr{F}_M$ is a v-dimensional vector space, such as Hilbert space of quantum channels, we have the general expression

$$\hat{\delta}_{\mathscr{F}} = 8\left(\frac{128e}{\varepsilon}\right)^v e^{-\varepsilon^2 l/512},$$

that may be sharpened using more specific physical properties.

For methods utilizing fusion of multiple methods, the generalization equations can be derived based on the isolation property [32, 31], for example, satisfied by the ensemble method. For instance, for the fuser class $\mathscr{F}_F$ used in fusing the classifiers $f_a \in \mathscr{F}_a, a \in \mathscr{A}$, let f_F denote the classifier function obtained by composing f_a's using a fuser from $\mathscr{F}_F$. The *accuracy improvement* Δ_F of the composite ML estimate over the best individual classifier is defined as

$$\Delta_F = I(f_F) - \max_{a \in A} I(f_a).$$

Then, if $\mathscr{F}_F$ has the isolation property [34] $\Delta_F \geq 0$, the best accuracy improvement is given by $\Delta_F^* = I(f_F^*) - \max_{a \in \mathscr{A}} I(f_a^*)$, and its estimate based on samples is given by

$$\tilde{\Delta}_F = \hat{I}(f_F) - \max_{a \in \mathscr{A}} \hat{I}(f_a),$$

where $\hat{I}(.)$ is the training error based on the sample. If we let $\hat{\delta}_b(\varepsilon, \hat{\varepsilon}_b, l)$ for all individual classifiers $b \in \mathscr{A}$ and classifier-fusers $b = \mathscr{B}$ such that $\hat{\delta}_b(\varepsilon, \hat{\varepsilon}_b, l) \to 0$ as $l \to \infty$, then the probability that the closeness between $\tilde{\Delta}_F$ and Δ_F^* is within ε is bounded as

$$\mathbb{P}^l_{X,Y}\left[|\tilde{\Delta}_F - \Delta_F^*| < \varepsilon\right] > 1 - \hat{\delta}_d(\varepsilon/2, \hat{\varepsilon}_d, l) - \sum_{a \in \mathscr{A}} \hat{\delta}_a(\varepsilon/(2N_{\mathscr{A}}), \hat{\varepsilon}_a, l) \tag{2}$$

for both classifier-fusers $d = \mathscr{B}$, wherein the right-hand side approaches 1 as $l \to \infty$.

4 Algebra and Calculus of ML

Over the past decade, there has been a sea change in ML practice in part due to the availability of software products, including python libraries and R modules. More recently, there has been an explosion of large language models (LLM) [23] both as trained models that can be accessed such as ChatGPT, or packages that can be indigenously built such as LLAMA2. Furthermore, ML frameworks for special-purpose and high-performance computing systems demonstrate the unprecedented scale and scope of practical problems solved by ML methods. Consequently, such methods have been extensively applied to a broad set of applications; initially, however, their ease of use resulted in them being used as just "black boxes," often leading to solutions that were hard to interpret and explain. Currently, we are at the verge of a post-black-box era in which basic ML methods are being customized, sharpened, and composed in unprecedented ways to exploit domain knowledge, in particular, system and physical aspects. The field of machine learning has traditionally been at the intersection of computing and statistics [2, 19] but now encompasses other disciplines including formal languages for expression and interpretation, for example, under the explainable AI framework [18]. These developments in practical ML methods can be broadly classified into the following categories:

(a) *Strategic Deep Compositions:* The sharpened and customized ML solutions are composed in deep, multilevel, acyclic graph combinations, wherein ML methods at nodes are selected to match the underlying system components.
(b) *Fusion of Diverse ML Methods*: Diverse ML methods, such as smooth kernels and support vector machines, non-smooth tree and forest estimates, and algebraic vector spaces, are combined to mitigate overfitting by ensuring diversity of design.
(c) *Hyperparameter Harness*: ML solutions are wrapped inside a harness to explore, select, and tune higher-level parameters, for example, using an informed gradient search in the hyperparameter space.

These complex operations represent *algebra and calculus of ML* methods, as illustrated by the fusion of diverse classifiers and regression methods [36].

4.1 Addition Operation and Information Fusion

The knowledge from laws and ML methods can be utilized in two basic additive ways: (a) *Sharpening and customization:* Basic ML methods can be fine-tuned by identifying and optimizing their structure and parameters by exploiting system and physical aspects. The sharpened and customized ML solutions can be composed in deep, multilevel, acyclic graph combinations in which ML methods at nodes are selected to match the underlying system components. (b) *Fusion of diverse ML methods:* Diverse ML methods (e.g., smooth kernels and support vector machines, non-smooth tree and forest estimates, algebraic vector spaces) can be combined to overcome over-fitting by ensuring the design diversity. An example is RTT estimation by fusing five disparate regression estimates in Fig. 3. Two examples of this ML addition operation in the nuclear engineering area are described in Section 6.2.

4.2 Diagnosis by Difference Operation

Physical and abstract laws that characterize ideal operating conditions (e.g., drift-free reactor dynamics and fully optimized data transport infrastructure) are used to compute the differentials with respect to ML models derived from measurements from operational, emulated, or simulated systems. Under ideal conditions, the differentials will be negligible, but under operational conditions, they indicate errors, such as sensor drifts in reactor monitoring sensors and under performance of data transport (e.g., convex throughput profiles). Furthermore, by relating the differentials to underlying laws, the parameters and components can be identified to support actions (e.g., recalibrating sensors or fine-tuning file systems) as described in Section 6.2.

5 Implications of Lack of Knowledge about ML-Solvability

The causes of lack of ML-solvabilty of a problem are two-fold: (a) non-computability of the training and inference problems, or (b) non-learnability of the underlying problem due to insufficient model set or hypothesis space, insufficient training data, or insufficient inference rules in the model. Since sound and complete ML answers require both computability and learnability of the underlying problem, codes attempting to solve a ML-unsolvable problem produce answers that lack one or both properties. This conclusion follows from a formal treatment of the soundness and completeness concepts succinctly presented in terms of an abstract deduction system in [63]. In addition, the lack of knowledge of ML-solvability in turn translates to the lack of knowledge about the completeness and soundness properties of output produced by ML codes. These implications are not merely academic in nature, since scientific ML codes developed being unaware of the underlying ML-solvability property, potentially produce outputs that the lack the needed rigorous analysis and justification.

5.1 Overfitting

The overfitting corresponds to low training error $\hat{\varepsilon}$ in Eq. (1) typically due to ML model M with a large number parameters, which in turn results in low confidence or high $\hat{\delta}_M$ as a result of large learning dimension. This phenomenon is not detectable by examining the training error alone, as it critically depends on the model space $\mathscr{F}_M$ which has a large learning dimension for models with large number of parameters. In some cases, testing using additional custom measurements can indicate overfitting, namely, much higher testing error than training error; a concrete example of this artifact for radiation source detection is described in Section 6.2. But, such cases very limited in practice due to the cost and lack of availability of such test data. The learning dimension that controls $\hat{\delta}_M$ can be estimated by using the properties of physical law $\mathscr{L}_S$, including smoothness, non-smoothness and algebraic properties [37], as described in previous section. The overfitting effect can be mitigated by choosing ML methods with low learning dimension informed by $\mathscr{L}_S$, for example, by controlling the depth of random forest ML estimate to match the Lipschitz parameters of the Poisson measurement law of radiation source or total variation of throughput profile of network transport infrastructure.

5.2 Soundness and Completeness: Hallucinations

The "hallucination" refers to incomplete or unsound inferences by ML models, which are increasingly observed in generative LLM methods. However, they are often undetected due to lack of knowledge to assess the inferences. Its causes are directly related to the terms in generalization equation Eq. (1): (a) $\mathscr{F}_M$ with an insufficient density property, such as sufficient inference rules, results in large $I(f^*)$ indicating incomplete or unsound $\tilde{f}$, which turn results in large $\hat{\delta}_M$ or low confidence, and (b) Turing undecidable f^* results in unsound or incomplete output generated by any code of $\tilde{f}$ however sophisticated it is (including LLM). Using ML methods based on physical laws violations, active probe methods can be used to estimate indicators of these violations, which indicate the likelihood of phenomena such as hallucinations. Specifically, the learning algorithms are customized by replacing the traditional cost function $C(.)$ by the violation terms [54] using the algebraic forms of physical laws to reduce such effects. In effect, the learned models are restricted to conform to underlying $\mathscr{L}_S$ in scientific domains with such conducive laws.

6 Applications

Complex ML solutions have been developed by exploiting the law $\mathscr{L}_S$ and its structure $\mathscr{S}_S(\mathscr{C}_S)$ of underlying system in several practical problems. We briefly describe two different types of applications that illustrate the ML-solvability, and additional one that illustrates the contrary, all from existing works. First, for a data transport nerwork, an abstract law $\mathscr{L}_S$ specifies the convex-concave shape of its throughput as a function of RTT, which is used to esatablish ML-solvability and also assess the degree of optimization using its ML estimates. Second, physical system laws are utilized to establish ML-solvability and develop ML-estimates in two cases: (a) thermal-hydraulic equations are used for ML-estimates of power-level and sensor drift in nuclear plants, and (b) radiation decay equations are used for ML-detction of a radiation source. Third, lack of ML-solvability of self-detection of large scale systems are shown by using the undecidability laws on properties of computations. Together, they illustrate various implications of ML-solvability ,and custom development of their ML-solutions and generalization equations using physical or abstract law $\mathscr{L}_S$.

6.1 Throughput Profiles of Data Transport Networks

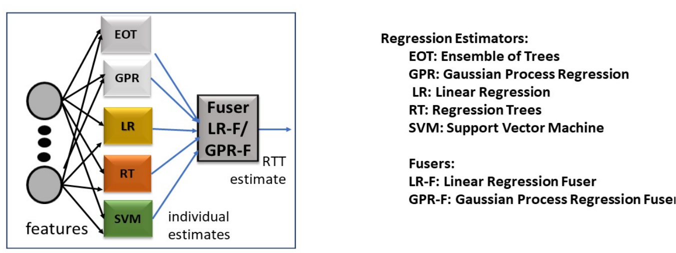


Fig. 3: *Fusion of disparate ML estimates provides performance superior to individual estimates.*

Recent ML efforts have focused on using network measurements to identify and isolate performance bottlenecks [47] and provide ways to allocate resources to optimize the throughput performance [33]. A mathematical method for characterizing transport performance that uses estimates of throughput profiles is the *concave-convex* analysis initiated in [57, 47]. Concave profiles indicate near-optimal throughput, and convex profiles indicate bottlenecks from factors such as IO buffers, file system throughput, mismatched IO-network couplings, and suboptimal protocols and transfer tools. The underlying mathematics of this analysis are based on statistical estimation and protocol time-dynamics structured as Poincare maps and Lyapunov exponents [45, 21]. As an example, consider a transport throughput time trace $\theta(\tau,t)$, which is a random quantity with a distribution dependent on complex factors like capacities of edge systems and networks, loads on files systems, transfer software, host systems, and connections. The transport performance is characterized by the *throughput profile* $\Theta(\tau)$ as a function of RTT τ defined as $\bar{\Theta}_O(\tau) = E\left[\Theta_O(\tau)\right]$ where the underlying distribution is too complicated to calculate with simple models. ML provides throughput profile estimates, which may be smooth such as sigmoid neural networks or nonsmooth such as forests of trees, [59] with finite sample performance guarantees [46]. Compared to the "physical laws", law $\mathscr{L}_S(P,Z)$ is abstract in terms of its concave-convex property. These problems are ML-solvable, and the generalization equations of ML estimates have been developed for both smooth and non-smooth estimators and their fusers (shown in Fig. 3).

6.2 Nuclear Applications: Facility Analytics and Source Detection

The structure of $\mathscr{S}_S$ fluid dynamic equations is used to compose ML solution for the estimation of the power level of a reactor based on external monitoring sensors [55, 40]. In this scenario, the activity of a reactor is monitored using measurements from four fans, four pumps, three acoustic sensors, and temperature sensors. The resultant a three-level ML solution is based on tree topology. Inferring of dissolution events at a radiochemical processing facility using monitoring measurements is an important classification task solved by fusing disparate methods [41, 42]. For dissolutions scenarios, the classifier overfitting is addressed by fusing three promising ones from eight disparate classifiers, akin to Fig. 3. Here, the physics-based decay chains provided the isotope features to be used as input to classifiers, and their half-life estimates provided feature window-size to achieve improved performance beyond the simple classifier fusion.

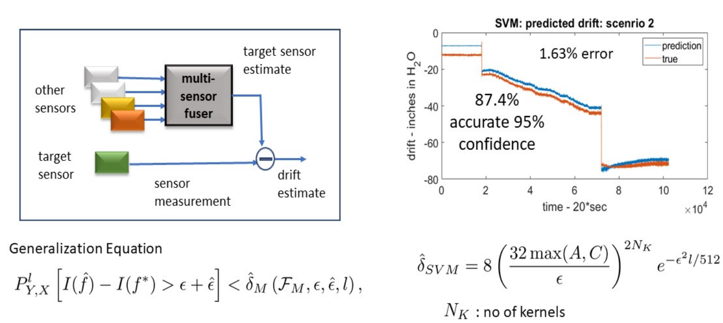


Fig. 4: *Difference between the measurement from a target sensor and its estimate by fusing measurements from other sensors is an estimate of the sensor error.*

The sensors of a power plant measure variables that are typically related to each other through the underlying physics laws. Such relationships provide regressions that are smooth or non-smooth with bounded variation property, which are conducive to estimation by machine learning methods [54]. These relationships are exploited to learn a regression function for a target sensor using measurements from other sensors that measure the same or different variables. The regression estimate is subtracted from its actual measurement to obtain a sensor drift estimate, as illustrated in Figure 4. The key idea is that when the target sensor measurement is subjected to drifts, this difference reflects its drift error [51, 50]. The root mean square error of drift estimate is below 12.6% with 95% confidence as indicated by the generalization equation [64] shown in the figure. The generalization equation does not impose any conditions on the underlying error distributions and yet provides practically useful information; the error and confidence estimates are derived by exploiting the measurement data sets and domain-specific customization [51].

Another class of problems in this domain address the detection of radiation sources using gamma spectral measurements. Signatures of low-intensity ^{235}U sources have been recently studied by utilizing a variety of ML classifiers that utilize features extracted from NaI gamma spectral measurements [43, 38, 53]. Their performance is tested using measurements collected under strategic configurations of NaI detectors located at various distances from the source in the formation of two concentric circles and one spiral. Multiple independent experimental runs have been conducted which provided data sets for training and testing of ML classifiers. Well-known ML classifiers revealed complex classification performance, and in particular, some classifiers and fusers overfit the training data, which results in overly optimistic training error that is negated by much larger error in independent test measurements.

A novel regression-based ML method is developed in [52] that first estimates the inverse of distance to source, which is a physical property; the background is represented as a source located at an infinite distance. The EOT and GPR methods fit non-smooth and smooth regression functions, respectively, and a hyper parameter auto-tuning and selection method (denoted by AUTO) employs regression trees, neural network, and support vector machine in addition to EOT and GPR. These methods avoid the overfitting observed in several ML classifiers, while providing the classification error nearly comparable to them based on independent test data. Their error is directly related to estimates of the inverse physical distance to source, and the precision of error determines the seperability property that determines the false alarm and missed detection rates. The property of monotonic decrease of the source strength with increasing detector distance combined with Poisson distribution of measurements is utilized to analytically validate these methods by deriving the generalization equations of underlying regression methods.

The physical laws used in these cases are different: thermal hydraulic equations are used for ML-estimation of the power level and sensor drift in the case of reactor systems, and Poisson measurements and quadratic intensity decay laws are used for radiation source detection.

6.3 Self-Diagnosis of Computing Systems

High-Performance computing systems are composed of a large number of multi-core processors, graphical processing units (GPU) and accelerators with computing elements totaling millions; also, they consist of interconnects, switches and hierarchies of memory units. The life-span of components used in these systems is typically about 5-10 years, for example, processors. Thus, to a first-order approximation, computations running for hours are likely to experience multiple failures, and they in turn may result in errors in application computations as well as in operating and runtime systems that execute applications. Furthermore, the sheer size and complexity of these systems may lead to complex faults, which cannot be known precisely or anticipated. They may range from manufacturing and fatigue faults in components, to dynamic hot-spots in computer racks due to interactions between device placement and cooling systems, to interactions of software modules with degraded hardware components. These faults may manifest in a variety of ways: memory faults causing executables to be corrupted, and variables with out-of-bound values; incorrect loading of program counters and errors in arithmetic and logic operations; and, bus and interconnect faults corrupting data in transit between processing units.

The resilient computations are required to produce correct results on computing systems subject to broad classes of complex failures. The resilient computation problems are shown to present significant computational challenges if the underlying failures are not precisely characterized and anticipated [35]. A broad class of resilience computation problems are shown to be undecidable in the sense of Turing [62], that is, no algorithms exist for solving them. The automatic verification of resilient computations in general are shown to be undecidable, and their extensions are used to show that the resilient computations under data and program corruption and execution errors are undecidable. Specifically, reductions are shown from the classical loop-detection and halting problems to generic resilient computation problems. Specifically, these proofs are presented under two formulations, namely, the abstract programming language L [14] and Turing machines [62], that highlight different aspects of the underlying failures; the former represents program and data corruption, and the latter illustrates incorrect program execution. Then, relativization results are presented that indicate that even if halting problems due to these errors are decidable, it is still possible for undecidable problems to persist. Also, other failure classes are described based on arithmetic systems that could lead to algorithms for which performance guarantees are hard to prove or not provable.

These undecidability results indicate that unless the class of faults is limited, these problems cannot be solved by purely computational and analytical means, and hence these problems are not ML-solvable. Hence, they call for broad-based approaches that complement computational solutions, which integrate methods such as hardware monitors, co-design of hardware and software solutions, system-specific diagnosis methods, and application specific resilience methods. Furthermore, algorithms, frameworks and ecosystems used in such approaches must clearly identify their target failures, and establish that the underlying computational problems are indeed are decidable for them to be solvable by ML methods. Thus, these classes of problems are not effectively solvable by ML methods even if large data sets of failure measurements are available.

7 Conclusions

The concept of ML-solvability is proposed in this paper to provide insights into applicability and performance of increasingly complex, often closed ML methods. By combining the theories of learnability, computing and logic, it provides insights into the possibility of incomplete and unsound inferences when ML-solvability of the underlying problem is not established or simply unknown. Then, a framework is developed for ML-solvability and generalization analyses based on a combination of physical laws that govern systems and information laws that characterize learning processes. Practical examples utilizing smooth, non-smooth and algebraic forms of laws to develop these ML solutions are discussed in the areas of data transport networks, and nuclear engineering, and computer system diagnosis.

The higher-level aspects of developing and applying ML methods discussed in this paper can be refined along several future directions possibly to much finer levels. It would be of future interest to explore refined aspects of ML-solvability to assess the black-box LLM methods specific to science domains, by possibly generating probes and analyzing the conformance of the answers to domain physical laws. The additional ML-solvability requirements of expressibility and provability can be further explored, which entails language extensions of models based on real space $\Re^d$ in our applications. Specifically, it would be of future interest to assess these aspects within the formulation of deductive systems used to show the Godel's incompleteness theorem using algorithmic approach described in [63]. The model spaces $\mathscr{F}_M$ of applications discussed in this paper are primarily based on $\Re^d$, and their formal and natural language versions would be needed to assess some of LLM methods with natural language inputs and outputs. Specifically, X and Y are language strings in these cases, and the cost function $C(.,.)$ needs to reflect them and possibly their semantics. Specifically, the physical or abstract laws of application domains may be expressed in a similar manner to support the design and analysis of these methods in assessing the correctness of their inferences and the proofs generated by them, possibly under deductive system framework. In another direction, the instrument-computing ecosystems consist of networked physical instruments and computing nodes, and their workflow codes are composed of computing and physical instrument instructions – it would be interest to extend the notions of computability and ML-solvability under this expanded formulation.

References

1. A. Al-Najjar, N. S. V. Rao, C. Bridges, and S. Dai. ML-based esolution for CV normality detection in real-time cross-facility electrochemistry workflow. In *2024 IEEE 20th International Conference on e-Science (e-Science)*, pages 1–10, 2024.
2. M. Anthony and P. L. Bartlett. *Neural Network Learning: Theoretical Foundations*. Cambridge University Press, 1999.
3. P. L. Bartlett. The sample complexity of pattern classification with neural networks: the sample size of the weights is more important than the size of the network. Technical report, Department of Systems Engineering, Australian National University, 1996.
4. S. Ben-David, P. Hrubeš, S. Moran, A. Shpilka, and A. Yehudayoff. Learnability can be undecidable. *Nature Machine Intelligence*, 1(1):44–48, Jan. 2019.
5. A. L. Blum and R. L. Rivest. Training a 3-node neural network is NP-complete. *Neural Networks*, 5:117–127, 1992.

6. A. Blumer, A. Ehrenfeucht, D. Haussler, and M. Warmuth. Learnability and the Vapnik-Chervonenkis dimension. *Journal of the Association of Computing Machinery*, 36(4):929–965, 1989.
7. L. Breiman, J. H. Friedman, R. A. Olshen, and C. J. Stone. *Classification and Regression Trees*. Wadsworth and Brooks, 1984.
8. D. S. Broomhead and D. Lowe. Multivariable functional interpolation and adaptive networks. *Complex Systems*, 2:321–355, 1988.
9. G. J. Chaitin. *Information, Randomness and Incompleteness*. World Scientific Pub, 1990. Second Edition.
10. Z. Ciesielski. Haar system and nonparametric density estimation in several variables. *Probability and Mathematical Statistics*, 9:1–11, 1988.
11. D. I. A. Cohen. *Inroduction to Computer Theory*. John Wiley and Sons, Inc., 1986.
12. F. Cohen. Computational aspects of computer viruses. *Computers Security*, 8(4):297–298, 1989.
13. T. H. Cormen, C. E. Leiserson, and R. L. Rivest. *Introduction to Algorithms*. McGraw-Hill Book Co., New York, 1990.
14. M. D. Davies and E. J. Weyuker. *Computability, Complexity, and Languages*. Academic Press, Inc, 1983.
15. L. Devroye, L. Gyorfi, and G. Lugosi. *A Probabilistic Theory of Pattern Recognition*. Springer-Verlag, New York, 1996.
16. M. R. Garey and D. S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W. H. Freeman and Co., San Francisco, 1979.
17. K. Godel. On formally undecidable propositions of principia mathematica and related systems i. *Monatshefte fur Math. und Physik*, 38:173–198, 1931. Englishe translation by B. Meltzer, published by Dover Publications, Inc, 1992.
18. D. Gunning. Explainable artificial intelligence program. Defense Advanced Research Projects Agency.
19. T. Hastie, R. Tibshirani, and J. Friedman. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer-Verlag, 2001.
20. E. Horowitz and S. Sahni. *Fundamentals of Computer Algorithms*. Computer Science Press, 1978.
21. Q. Liu, N. S. V. Rao, C. Q. Wu, D. Yun, R. Kettimuthu, and I. Foster. Measurement-based performance profiles and dynamics of UDT over dedicated connections. In *International Conference on Network Protocols*. Singapore, Nov. 2016.
22. W. Mass. Agnostic PAC learning of functions on analog neural nets. *Neural Computing*, 7:1054–1078, 1995.
23. A. Matarazzo and R. Torlone. A survey on large language models with some insights on their capabilities and limitations, 2025.
24. M. Mohri, A. Rostamizadeh, and A. Talwalkar. *Foundations of Machine Learning*. The MIT Press, 2018. second edition.
25. C. Moore. Unpredictability and undecidability in dynamical systems. *Phys. Rev. Lett.*, 64:2354–2357, May 1990.
26. J. Murzi. The inexpressibility of validity. *Analysis*, 74(1):65–81, 2014.
27. C. H. Papadimitriou. *Computational Complexity*. Addition-Wesley Publishing Co., 1994.
28. H. Qiao, N. S. V. Rao, and V. A. Protopopescu. PAC learning of using Nadaraya-Watson estimator based on orthogonal systems. In M. Li and A. Maruoka, editors, *Algorithmic Learning Theory*. Lecture Notes in Artificial Intelligence 1316, Springer-Verlag, 1997.
29. N. Rao. Experimental study of tcp throughput profiles and dynamics over dedicated connections. In *Proceedings of the SC '23 Workshops of The International Conference on High Performance Computing, Network, Storage, and Analysis*, SC-W '23, page 776–784, New York, NY, USA, 2023. Association for Computing Machinery.

30. N. S. V. Rao. Simple sample bound for feedforward sigmoid networks with bounded weights. *Neurocomputing*, 29:115–122, 1999.
31. N. S. V. Rao. Finite sample performance guarantees of fusers for function estimators. *Information Fusion*, 1(1):35–44, 2000.
32. N. S. V. Rao. On fusers that perform better than best sensor. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(8):904–909, 2001.
33. N. S. V. Rao. Overlay networks of in-situ instruments for probabilistic guarantees on message delays in wide-area networks. *IEEE Journal on Selected Areas in Communications*, 22(1), 2004.
34. N. S. V. Rao. Measurement-based statistical fusion methods for distributed sensor networks. In S. S. Iyengar and R. R. Brooks, editors, *Distributed Sensor Networks*. Chapman and Hall/CRC Publishers, 2011. 2nd Edition.
35. N. S. V. Rao. On undecidability aspects of resilient computations and implications to exascale. In *Resilience 2014: Seventh Workshop on Resiliency in High Performance Computing with Clouds, Grids, and Clusters*, 2014.
36. N. S. V. Rao. An algebra of machine learners with applications. In *International Conference on Information Fusion*, 2021.
37. N. S. V. Rao. Generalization equations for machine learners based on physical and abstract laws. In *IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems*, 2021.
38. N. S. V. Rao, D. Abrecht, D. Hooper, J. Ladd-Lively, and D. Meier. Classification analytics of ^{239}pu and ^{235}u source signatures using gamma spectral regions. In *Institute of Nuclear Materials Management Annual Meeting*, 2023.
39. N. S. V. Rao, C. Greulich, P.amuhalli, A. Gurgen, F. Zhang, and S. M. Cetiner. Estimation of sensor measurement errors in rector coolant systems using multi-sensor fusion. *Nuclear Engineering Design*, 2021.
40. N. S. V. Rao, C. Greulich, S. Sen, K. Dayman, J. Hite, W. Ray, R. Hale, A. Nicholson, J. Honson, M. R. Chatin, K. M. Buckley, R. D. Hunley, J. Johnson, H. H. Hesse, M. Maceira, C. Chai, O. Marcillo, T. Karnowski, and R. Wetherington. Reactor power level estimation by fusing multi-modal sensor measurements. In *International Conference on Inforamation Fusion*, 2020.
41. N. S. V. Rao, C. Greulich, S. Sen, K. Dayman, A. Nicholson, M. R. Chatin, K. M. Buckley, R. D. Hunley, J. Johnson, H. H. Hesse, and R. Hale. Classifiers for dissolution events in processing facility using effluents measurements. In *Institute of Nuclear Materials Management Annual Meeting*, 2019.
42. N. S. V. Rao, C. Greulich, S. Sen, J. Hite, K. J. Dayman, A. D. Nicholson, D. E. Archer, M. J. Willis, I. G. R. D. Hunley, J. Johnson, A. J. Rowe, I. R. Stewart, and J. M. Ghawaly. Classification of dissolution events using fusion of effluents measurements and classifiers. In *Institute of Nuclear Materials Management Annual Meeting*, 2020.
43. N. S. V. Rao, D. Hooper, and J. Ladd-Lively. Study of classifiers for u-235 source signatures using gamma spectral measurements. In *Institute of Nuclear Materials Management Annual Meeting*, 2022.
44. N. S. V. Rao, Q. Liu, Z. Liu, R. Kettimuthu, and I. Foster. Throughput analytics of data transfer infrastructures. In *13th EAI International Conference on Testbeds and Research Infrastructures for the Development of Networks and Communities (TRIDENTCOM 2018)*, 2018.
45. N. S. V. Rao, Q. Liu, S. Sen, J. Henley, I. T. Foster, R. Kettimuthu, D. Towsley, and G. Vardoyan. TCP throughput profiles using measurements over dedicated connections. In *ACM Symposium on High-Performance Parallel and Distributed Computing*, Washington, DC, July-August. 2017.

46. N. S. V. Rao, Q. Liu, S. Sen, Z. Liu, and R. Kettimuthu. Measurements and analytics of wide-area file transfers over dedicated connections. In *International Conference on Distributed Computing and Networking*, 2018.
47. N. S. V. Rao, Q. Liu, S. Sen, D. Towsley, G. Vardoyan, I. T. Foster, and R. Kettimuthu. Experiments and analyses of data transfers over wide-area dedicated connections. In *26th International Conference on Computer Communications and Network*, 2017.
48. N. S. V. Rao, C. Y. T. Ma, and F. He. Game strategies for data transfer infrastructures against ml-profile exploits. *IEEE Transactions on Machine Learning in Communications and Networking*, 2:925–938, 2024.
49. N. S. V. Rao and V. Protopopescu. On PAC learning of functions with smoothness properties using feedforward sigmoidal networks. *Proceedings of the IEEE*, 84(10):1562–1569, 1996.
50. N. S. V. Rao, P. Ramuhalli, C. Greulich, and S. M. Cetiner. Sensor error estimation for reactor coolant system with generalization error equations. *Transactions of the American Nuclear Society*, 121(1), 11 2019.
51. N. S. V. Rao, P. Ramuhalli, C. Greulich, S. M.Cetiner, and P. Devineni. Sensor drift estimation for reactor systems byfusing multiple sensor measurements. In *IEEE Nuclear Science Symposium conference record. Nuclear Science Symposium, Manchester, U.K.*, 2019.
52. N. S. V. Rao, C. Redding, D. Abrecht, D. Hooper, and J. Ladd-Lively. Physics-based machine learning methods for ^{235}u forensics signatures. In *Institute of Nuclear Materials Management Annual Meeting*, 2022.
53. N. S. V. Rao, C. Redding, D. Hooper, and J. Ladd-Lively. Selection of training sets for ^{235}u source detection classifiers using gamma signatures. In *Symposium on Radiation Measurements and Applications*, 2023.
54. N. S. V. Rao, D. B. Reister, and J. Barhen. Information fusion methods based on physical laws. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(1):66–77, 2005.
55. N. S. V. Rao, S. Sen, K. Dayman, W. Ray, R. Hale, A. Nicholson, J. Johnson, R. Hunley, and M. Maceida. Multi-modal sensor fusion for reactor power-level estimation: Thermal, EM, acoustic. In *Nuclear Security Applications Research & Development Program Review Meeting*, 2019.
56. N. S. V. Rao, S. Sen, Z. Liu, R. Kettimuthu, and I. Foster. Learning concave-convex profiles of data transport over dedicated connections. In *International Conference on Machine Learning for Networking (MLN'2018)*, 2018.
57. N. S. V. Rao, D. Towsley, G. Vardoyan, B. W. Settlemyer, I. T. Foster, and R. Kettimuthu. Sustained wide-area TCP memory transfers over dedicated connections. In *IEEE International Conference on High Performance and Smart Computing*, New York, NY, Aug. 2015.
58. B. Scholkopf, C. J. C. Burges, and A. J. Smola, editors. *Advances in Kernel Methods*. MIT Press, 1999.
59. S. Sen, N. S. V. Rao, Q. Liu, N. Imam, I. T. Foster, and R. Kettimuthu. On analytics of file transfer rates over dedicated wide-area connections. In *First International Workshop on Workflow Science (WOWS)*, Auckland, New Zealand, October 2017. in conjunction with 13th IEEE International Conference on e-Science.
60. A. Tarski. *Logic, Semntics, Metamathematics*. Oxford University Press, 1956.
61. K. Tumer and J. Ghosh. Error correlation and error reduction in ensemble classifiers. *Connection Science*, 8(3):385–404, 1996.
62. A. N. Turing. On computable numbers, with an application to the Entscheidungsproblem. *Proc. London Math. Society*, 42(3,4):230–265, 1936.
63. V. A. Uspensky. *Godel's Incompleteness Theorem*. Mir Publsihers, 1987. English translation.
64. V. N. Vapnik. *Statistical Learning Theory*. John-Wiley and Sons, New York, New York, 1998.
65. Y. Wu, B. Sicard, and S. A. Gadsden. A review of physics-informed machine learning methods with applications to condition monitoring and anomaly detection, 2024.

Digital Forensics in the Age of Large Language Models

Zhipeng Yin[1*], Zichong Wang[1], Weifeng Xu[2], Jun Zhuang[3], Pallab Mozumder[1], Antoinette Smith[1], Wenbin Zhang[1]

[1]Florida International University, Miami, Florida, USA.
[2]University of Baltimore, Baltimore, Maryland, USA.
[3]Boise State University, Boise, Idaho, USA.

*Corresponding author(s). E-mail(s): zyin007@fiu.edu;

Abstract

Digital forensics plays a pivotal role in modern investigative processes, utilizing specialized methods to systematically collect, analyze, and interpret digital evidence for judicial proceedings. However, traditional digital forensic techniques are primarily based on manual labor-intensive processes, which become increasingly insufficient with the rapid growth and complexity of digital data. To this end, Large Language Models (LLMs) have emerged as powerful tools capable of automating and enhancing various digital forensic tasks, significantly transforming the field. Despite the strides made, general practitioners and forensic experts often lack a comprehensive understanding of the capabilities, principles, and limitations of LLM, which limits the full potential of LLM in forensic applications. To fill this gap, this paper aims to provide an accessible and systematic overview of how LLM has revolutionized the digital forensics approach. Specifically, it takes a look at the basic concepts of digital forensics, as well as the evolution of LLM, and emphasizes the superior capabilities of LLM. To connect theory and practice, relevant examples and real-world scenarios are discussed. We also critically analyze the current limitations of applying LLMs to digital forensics, including issues related to illusion, interpretability, bias, and ethical considerations. In addition, this paper outlines the prospects for future research, highlighting the need for effective use of LLMs for transparency, accountability, and robust standardization in the forensic process.

Keywords: Large Language Model, Digital Forensics, Artificial Intelligence, Forensic Investigations

1 Introduction

Digital forensics is a critical component in modern investigative and judicial processes, which involve the systematic collection, analysis, and preservation of digital evidence from electronic devices and online activities [1–3]. Its primary objective is to uncover factual information related to cybercrimes, fraud, unauthorized access, and other illicit activities [4]. Digital forensics has played a pivotal role in solving high-profile cybercrime cases. For example, in the 2014 Sony Pictures hack, forensic investigators traced the breach back to North Korean hackers, who leaked confidential company data, emails, and unreleased films as part of a geopolitical cyber attack [5]. The investigation relied on digital forensics techniques such as analyzing network logs, identifying malware signatures, and attributing IP addresses to suspected attackers. As another example, in the 2016 Democratic National Committee (DNC) email leak, digital forensic experts identified sophisticated spear-phishing tactics and linked the attack to Russian-backed hacking groups, influencing the U.S. presidential election [6, 7]. Beyond cyber espionage, digital forensics has also been crucial in financial fraud investigations. For instance, the Silk Road darknet marketplace, a notorious online black market, was dismantled in 2013 through extensive forensic analysis of Bitcoin transactions, server logs, and encrypted messages [8, 9]. Forensic experts traced Bitcoin payments to the marketplace's operator, Ross Ulbricht, ultimately leading to his arrest and life sentence. In another case, the Enron scandal saw digital forensics specialists recover crucial deleted emails and financial records, providing key evidence in one of the largest corporate fraud investigations in history [10]. Additionally, digital forensic methodologies have been instrumental in child exploitation cases, where law enforcement agencies track online predators by analyzing metadata in images, chat logs, and digital footprints left on the dark web [11, 12].

These case studies highlight the effectiveness of digital forensics in various domains, but they also demonstrate how investigators increasingly encounter complex technological challenges that test the limits of current methodologies. The primary issue is that traditional digital forensic techniques predominantly rely on manual or semi-automated approaches, requiring intensive human involvement [13]. These methods suffer from several inherent limitations. Firstly, they are labor intensive and time consuming, making them less effective in handling large-scale and sophisticated cyber incidents [14]. Secondly, traditional methods often struggle to maintain consistency and accuracy due to human error and subjective judgments, potentially compromising evidence reliability. In addition, conventional forensic tools exhibit limited adaptability to evolving cyber threats, and their capability to identify complex interrelationships among evidence entities remains constrained [15, 16].

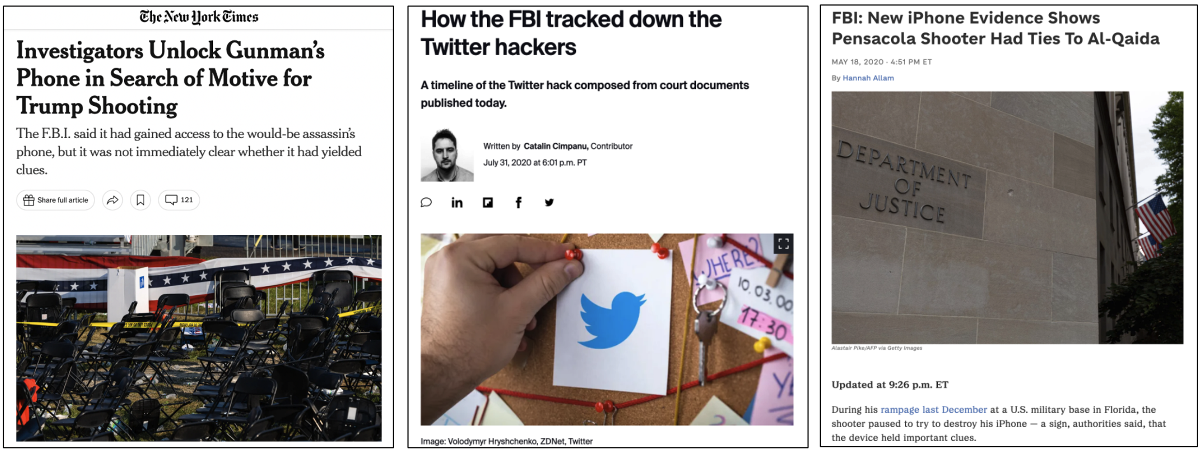


Fig. 1: Investigation of real-world digital forensics cases in recent years.

Several recent cases, as shown in Figure 1 illustrate these limitations. For instance, in July 2024, a gunman attempted to assassinate former U.S. President Donald Trump during a public rally, prompting an intensive investigation by federal authorities [35, 36]. Following the suspect's capture, the FBI conducted a forensic analysis of his mobile device to uncover potential motives, affiliations, and premeditated plans. Investigators faced significant challenges in bypassing the phone's security mechanisms, including encryption and biometric locks. Once access was gained, digital forensic teams meticulously analyzed call logs, text messages, encrypted messaging apps, and location history. Additionally, they examined the suspect's social media interactions, online searches, and affiliations with extremist groups. Traditional analysis methods required extensive manual effort to filter through vast amounts of digital data, cross-reference communication patterns, and verify links between different sources. Such challenges were similarly highlighted in the 2020 Twitter cryptocurrency scam, where cybercriminals compromised multiple high-profile accounts to solicit Bitcoin payments [37]. The FBI's digital forensic teams encountered substantial hurdles as they manually cross-referenced Discord chat logs, leaked hacker forum databases, cryptocurrency wallet transactions, and IP addresses to identify the perpetrators. Although successful, this method revealed critical shortcomings in efficiently correlating and interpreting multi-dimensional digital evidence streams, demonstrating the urgent need for more advanced forensic capabilities [38]. These complexities also surfaced prominently in the FBI's investigation into the Pensacola naval base shooting in 2019 [39, 40]. In that case, the assailant's encrypted iPhones had sustained physical damage, and Apple refused official requests for access assistance. Consequently, FBI forensic experts spent months painstakingly repairing hardware and circumventing encryption to retrieve data. Eventually, the recovered digital evidence established clear connections between the attacker and foreign terrorist entities. However, the prolonged investigative timeline underscored limitations inherent in traditional forensic methodologies when handling encrypted devices and fragmented digital traces [41, 42]. Collectively, these cases emphasize the increasing necessity of integrating AI-driven digital forensic tools. Leveraging automation, intelligent data analysis, and advanced pattern recognition technologies could significantly enhance investigative speed, consistency, and accuracy, effectively addressing the growing scale, complexity, and sophistication of contemporary cyber threats [43, 44].

To address these substantial challenges in digital forensics, recent advances in artificial intelligence offer promising solutions. Notably, large language models (LLMs), such as the generative pre-trained transformer (GPT) series of models and the Gemini series, have emerged as powerful tools with the potential to transform digital forensic practices [23, 24, 30, 45]. These advanced AI models are designed to understand, interpret, generate and analyze human language with unprecedented accuracy. Using vast amounts of textual data from diverse sources, LLMs exhibit exceptional capabilities in natural language processing, pattern recognition, and semantic understanding [46–49]. Their ability to extract meaningful insights from large, unstructured datasets makes them invaluable in digital forensic investigations [50].

The application of these sophisticated models directly addresses the limitations of traditional forensic approaches identified earlier. One of the most impactful applications of LLMs in forensic analysis is their ability to automate and streamline the evidence identification process. [51, 52] Traditional methods require investigators to manually sift through enormous volumes of text, such as emails, chat logs, social media posts, and financial records. LLMs, on the other hand, can swiftly process and categorize these texts, recognizing patterns, detecting anomalies, and identifying crucial connections between disparate pieces of evidence [53–55]. This capability significantly accelerates investigative timelines while reducing the risk of human error [56]. In the aforementioned high-profile investigation, integrating LLM-powered analytical tools could have played a transformative role in expediting the investigative process. By rapidly categorizing and interpreting textual evidence, LLMs can highlight potential leads, uncover hidden relationships, and help investigators piece together a cohesive narrative [57–59]. Moreover, their ability to process multilingual content ensures that forensic teams can analyze communication in different languages and cultural contexts without the need for extensive translation efforts [60, 61]. LLMs also improve forensic data interpretation by facilitating the reconstruction of complex evidence relationships. They can map connections between personal identifiers, such as names, addresses, and phone numbers, and correlate them with network activity, financial transactions, and geolocation data [62–64]. This holistic approach allows investigators to establish links between suspects, victims, and illicit activities with greater precision [65]. Another crucial advantage of LLMs in digital forensics is their ability to handle large-scale data integration. Digital evidence is often scattered across multiple sources, including cloud storage, encrypted messaging platforms, and darknet forums. LLMs, combined with knowledge graph techniques, can aggregate and visualize these fragmented data points, making it easier to identify trends, associations, and key actors within an investigation [66].

While the benefits of LLMs for digital forensics are substantial, their implementation is not without challenges that need to be carefully considered [67–69]. In addition, despite their transformative potential, the adoption of LLMs in forensic investigations also introduces new challenges, including concerns over interpretability, bias, and the reliability of AI-generated insights [70, 71]. Ensuring transparency in forensic AI applications is crucial to maintaining credibility in judicial proceedings [72]. Therefore, addressing these concerns through clear guidelines, rigorous validation procedures, and transparent reporting practices becomes essential [73–75]. To this end, this paper explores how LLMs can fundamentally change digital forensics practices by automating evidence analysis, extracting insightful information, and enhancing the judicial process, and attempts to provide a comprehensive understanding of the practical applications, potential limitations, and broader implications of LLMs in digital forensics investigations [76].

Paper Structure. The subsequent sections of this paper are structured as follows: Section 2 introduces foundational concepts and highlights the limitations of training-based ai digital forensic methodologies. Section 3 details the principles and capabilities of large-scale language modeling and presents practical applications and real-world case studies. Section 4 evaluates the current challenges and limitations faced when deploying LLMs in forensic scenarios. Finally, Section 5 discusses opportunities and directions for future research.

2 Fundamentals of Digital Forensics

This section examines in depth the core principles of digital forensics. Understanding these fundamentals is critical to grasping how modern investigative processes utilize digital evidence to combat cybercrime, fraud, and other illicit activities.

2.1 Definition and Goals

Digital forensics is the systematic process of identifying, collecting, preserving, analyzing, and presenting digital evidence in a legally admissible manner [93, 94]. It is widely used in criminal investigations, cybersecurity incidents, corporate fraud detection, and other legal proceedings[95]. The primary goal of digital forensics is to uncover and reconstruct events related to cybercrimes, unauthorized access, financial fraud, intellectual property theft, and other illicit digital activities. By leveraging digital forensic techniques, investigators can retrieve hidden, deleted, or encrypted data to support legal actions and improve cybersecurity measures [96].

2.2 Digital Forensic Evidence Entities

A digital forensic evidence entity represents the smallest indivisible unit of digital information possessing forensic significance. Such entities serve as fundamental building blocks for reconstructing digital events and verifying their authenticity. These entities are categorized according to their functional purpose in supporting investigative analysis [97]. Table 1 provides a detailed description of these entities by functional purpose. The Content-Descriptive Entities help investigators understand the nature, source, or intended use of digital artifacts, providing essential context to evidence collected during an investigation [98]. In contrast, auxiliary entities supplement this understanding by offering validation, verification, and support to primary descriptive evidence, ensuring the reliability and integrity of forensic findings.

Table 1: Functional Categories and Descriptions of Digital Forensic Evidence Entities

Categories	Digital Forensic Evidence Entities	Description
Content-Descriptive Entities	File Names	Identifiers given to files, potentially revealing their content, origin, or intended purpose.
	IP Addresses	Numerical labels assigned to devices on a network, crucial for tracking and attributing online activities.
Auxiliary Entities	Timestamps	Specific points in time indicating events such as file creation, modification, or access.
	Hashes	Unique identifiers generated from data content used to verify file integrity and detect tampering.

2.3 Key Evidence Types

While the theory of digital forensic evidence entities is understood based on their functional role, real-world forensic investigations often require more specific categorization. Investigators routinely encounter various forms of digital evidence, which must be clearly identified and categorized to effectively address complex forensic challenges [99, 100]. The following summarizes specific categories of digital evidence, which are classified according to their relevance and investigative role, and demonstrates how the theoretical framework translates into operational forensic practice [101].

- **Personal Identifiers:** Names, addresses, phone numbers, email addresses, social security numbers, and other personal information. In identity theft cases, stolen personal identifiers are typically located within phishing emails, compromised databases, or fraudulent registrations.
- **Network Information:** IP addresses, MAC addresses, login credentials, and network logs crucial for tracing user actions across devices and networks. Investigators often utilize this data to pinpoint sources of unauthorized access, as exemplified by numerous cases of insider threats and external intrusions.

- **Communication Records:** Emails, text messages, social media messages, and call logs that capture interactions among individuals or groups. Analysis of these records has been pivotal in solving cases involving cyberbullying, insider trading, and organized crime.
- **Financial Data:** Bank account details, credit card transactions, cryptocurrency wallet addresses, and transaction histories essential in tracking financial fraud and money laundering. Forensic analysts frequently exploit blockchain technology to unravel cryptocurrency-based criminal networks.
- **Location Data:** GPS coordinates, timestamps, and geolocation logs, enabling investigators to track movements and verify alibis. This form of data has notably been employed in criminal cases where mobile device locations provided critical evidence linking suspects to crime scenes.
- **Internet Activity:** Web browsing history, search queries, downloads, and online interactions offering deep insights into user behaviors and intentions. These digital footprints have been invaluable in cases involving radicalization, online harassment, and cyberstalking.
- **File Metadata:** Information including timestamps, file paths, and document version histories, useful for establishing file authenticity and tracking document manipulation. Metadata analysis has been critical in corporate espionage investigations and cases of intellectual property theft.
- **Device Logs and System Artifacts:** System event logs, registry entries, and application usage records, offering detailed insight into user activities and system states. In investigations of data breaches or corporate sabotage, these logs have provided evidence disproving fabricated user accounts and narratives.

2.4 Evidence Relationships

Having discussed specific categories of digital evidence encountered in practical forensic scenarios, it is crucial to recognize that these pieces of evidence rarely exist in isolation. Instead, they form intricate networks of relationships that significantly enhance investigative analysis [102]. Understanding these interconnections allows investigators to reconstruct detailed narratives, establish causality, and verify the authenticity of digital evidence comprehensively [103]. The key relationship categories include:

i) Contextual Relationships: These relationships provide situational context, helping investigators understand the origin, purpose, or usage of evidence. For example, linking file names to their content or correlating an IP address to a geographical location helps determine the source and intention behind cyber incidents.

ii) Causal Relationships: Highlight cause-and-effect dynamics between evidence entities. Identifying the correlation between an IP address and a specific time stamp can establish a suspect's direct involvement in unauthorized access or data manipulation.

iii) Associative Relationships: Connect seemingly independent evidence through shared attributes. Similar file hashes detected across multiple devices may suggest deliberate data duplication, exfiltration, or manipulation efforts by malicious actors.

iv) Communication Relationships: Reveal interaction patterns among individuals or systems. Analyzing communication logs such as phone records, emails, or chat messages has proven essential in dismantling criminal networks, uncovering collaboration among perpetrators, and mapping complex interactions in cybercrime investigations.

v) Ownership and Association: Establish explicit connections between individuals and digital devices, accounts, or data. Digital forensic efforts routinely involve associating specific devices or accounts with suspects, thereby strengthening investigative narratives and courtroom presentations.

vi) Temporal Relationships: Establish a chronological sequence or simultaneity of events. Timestamp analysis enables forensic examiners to confirm or refute suspect claims, authenticate alibis, and determine exact timelines of incidents, especially critical in high-stakes criminal and corporate investigations.

2.5 Limitation Of Training-based AI For Digital Forensics

While AI driven methodologies offer significant advancements in digital forensic investigations, several inherent limitations constrain their effectiveness, particularly when employing training-based AI approaches:

i) Data Scarcity: Obtaining sufficient and diverse training data representative of real-world cyber incidents poses significant challenges. Often, the available data is limited to specific case types, such as addresses extracted predominantly from certain criminal activities like shootings. This lack of comprehensive and varied datasets can severely restrict the AI model's ability to generalize across different forensic scenarios [51].

ii) Data Pre-processing Challenges: Even seemingly simple tasks, such as identifying addresses using Named-Entity Recognition (NER), introduce considerable pre-processing complexity before AI models can be effectively applied. These tasks often require multiple pre-processing steps, including expanding abbreviations (e.g., converting"St." to"Street"), standardizing formats (e.g., "123 Main St Apt 4B" to "123 Main Street, Apartment 4B"), normalizing state names (e.g., "California" to "CA"), and removing extra whitespace (e.g., converting "456 Elm St" to"456 Elm St."). These additional pre-processing steps significantly increase the complexity, time, and resources required, underscoring the limitations associated with direct training-based AI approaches in digital forensic analyses.

iii) AI Models Lack Adaptability: AI models developed for digital forensic tasks are typically designed and optimized for specific, narrowly defined functions. For instance, an AI model trained explicitly for recognizing addresses will likely exhibit limited performance when tasked with identifying other types of information, such as personal names or financial records [104]. This specialized training makes it challenging to apply these models broadly across the diverse range of forensic tasks investigators encounter.

iv) Difficulty in Extracting Evidence Relationships: Identifying and analyzing the numerous intricate relationships among digital evidence entities is inherently complex. Training-based AI methods often struggle to capture the full depth and nuance of these interactions, given the extensive variety and subtlety in relationships, including contextual, causal, associative, communication-based, ownership-based, and temporal connections [105]. Consequently, traditional training-based approaches may not recognize critical evidence correlations, potentially undermining the accuracy and comprehensiveness of forensic analyzes.

3 Large Language Models for Digital Forensics

3.1 Why LLMs For Digital Forensics

Large Language Models (LLMs) represent a sophisticated category of artificial intelligence models, primarily designed to understand, generate, and interact with natural language text [106]. These models typically utilize deep learning architectures, such as transformers, which rely on self-attention mechanisms to capture intricate contextual relationships within textual data [107]. The development and training of LLMs involve vast datasets, often comprising billions of words, enabling these models to acquire a deep understanding of syntax, semantics, and contextual nuances inherent in human languages.

One of the most prominent examples of LLMs is the Generative pre-trained Transformer (GPT) series developed by OpenAI, including GPT-3 and GPT-4. These models exhibit exceptional capabilities across a wide range of natural language processing (NLP) tasks, such as text generation, summarization, translation, sentiment analysis, question answering, and entity extraction [108, 109]. Their impressive versatility stems from their ability to capture long-range contextual dependencies and their extensive training on diverse textual resources such as websites, books, articles, and other publicly available information.

The LLMs training process generally involves two main phases: pre-training and fine-tuning. During pre-training, models are exposed to vast, unsupervised text corpora, learning general language patterns, syntax, and semantic relationships without specific task-oriented labels. In the fine-tuning phase, LLMs are further trained in task-specific datasets, adapting their general language comprehension skills to effectively perform targeted NLP tasks [110]. This two-phase approach significantly enhances their adaptability and performance across diverse domains.

LLMs are trained on vast amounts of text data and exhibit exceptional capabilities in learning linguistic patterns, structural semantics, and contextual dependencies. These attributes make them uniquely suited for applications in digital forensics, where the volume and heterogeneity of digital evidence can overwhelm traditional analysis techniques. In digital forensic investigations, evidence often exists in unstructured or semi-structured forms, such as chat logs, emails, file metadata, browsing histories, and system logs. Manually extracting meaningful patterns or relationships from such data is labor intensive and time consuming [111]. LLMs can assist by automatically identifying named entities, classifying document types, summarizing lengthy communication threads, detecting suspicious patterns, and establishing semantic links across diverse artifacts.

LLMs offer the ability to generalize from limited context, which is particularly useful in forensic settings where fragmented or incomplete evidence is common. Their pre-training on diverse data sources also allows them to recognize and interpret technical jargon, code snippets, and colloquial expressions, enabling them to analyze evidence drawn from varied digital environments [106]. LLMs also support multi-turn interactions, allowing investigators to iteratively refine queries or extract context-sensitive information from large datasets in a conversational manner. This interaction paradigm not only enhances usability but also reduces the need for technical expertise in formulating complex forensic queries.

Therefore, LLMs have great potential to enable digital forensics given their training on large textual datasets and powerful pattern learning capabilities, and by utilizing LLms' advanced linguistic understanding analytical and generative capabilities, key information can be extracted from large amounts of data, significantly streamlining the analysis of evidence, enhancing the process of informed decision-making, and improving overall forensic outcomes [107].

3.2 LLMs-driven methods in Digital Forensics

Integrating LLM into forensic workflows has emerged as a promising approach as investigators seek new tools to improve the accuracy, efficiency, and scalability of their analyses. This section provides an overview of current methodological frameworks and empirical case studies in which LLM has been effectively utilized in digital forensic environments. Specifically, it highlights representative applications, evaluates their practical effectiveness, and identifies methodological insights that have emerged from these real-world implementations.

3.2.1 LLM-driven Construction of Evidence Networks

Utilizing their powerful pattern recognition and relationship extraction capabilities, LLMs introduce innovative methods to improve the efficiency and accuracy of forensic investigations [112]. One prominent example of such LLM-driven methods involves utilizing GPT-4-turbo to systematically identify and visualize patterns within digital forensic evidence. This method constructs a structured graph G = (V, E), where nodes (V) represent individual evidence items—such as names, addresses, and phone numbers—while edges (E) depict relationships connecting these items. Each edge is labeled explicitly to describe the nature of the connection, such as "owns" for ownership (a person owning a phone number) or "lives-in" for residency (a person residing at an address). These clearly defined relationships enable the creation of comprehensive visual representations that simplify the analysis of intricate forensic data.

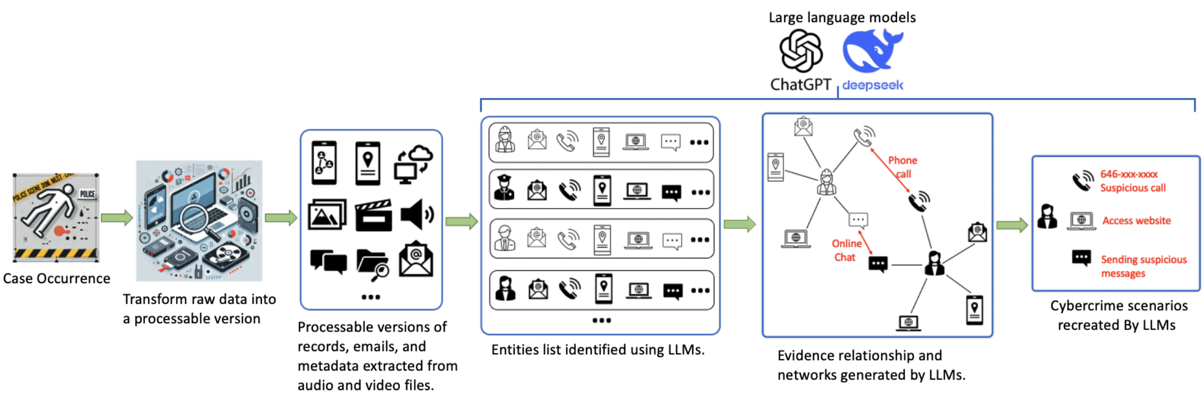


Fig. 2: A LLMs Methods to Understanding Cybercrime via Evidence Networks.

The main procedural steps shown in Figure 2 typically include:

i) Transform raw data into a processable versions: This step involves extracting and standardizing evidence from mobile devices, personal electronic devices and especially from their embedded multimedia card storage. Considering that these devices often contain fragmented, hidden or deleted data in binary form, converting this information into a clear text format is essential for the accurate analysis of llm.

ii) Identifying evidence entities and their relationships: Researchers create and test tailored prompts that guide LLMs in systematically extracting relevant evidence entities from structured textual data such as chat logs and system records. A representative prompt could be: "Act as an experienced digital forensic investigator. Extract evidence entities like names, addresses, and phone numbers from the given text and outline the relationships among these entities."

iii) Constructing evidence networks: This step involves connecting isolated pieces of evidence to form coherent networks. Connections are identified based on proximity, either physical (line distance in text) or semantic (inferred through LLMs), under the assumption that closely positioned entities are likely interrelated.

iv) Deriving insights into criminal behavior: Lastly, these constructed evidence networks are analyzed to uncover significant insights into criminal activities, behaviors, and underlying relationship patterns. This detailed examination of interconnected evidence provides forensic investigators with critical information that enhances their understanding of complex criminal scenarios.

3.2.2 LLM-driven Invocation Log Analysis for Digital Forensics

Chernyshev *et al.* proposed a novel forensic methodology aimed at detecting prompt injection attacks in applications integrated with LLMs [113]. The core innovation of this approach lies in leveraging invocation logs,a structured records of LLM interactions, as a primary evidentiary source for digital forensic investigations.

Their method involves constructing a simplified yet representative experimental scenario that emulates real-world LLM-integrated web applications, and Figure 4 illustrates its workflow. Specifically, the authors developed a web-based application utilizing GPT-3.5 via the LangChain framework. In this scenario, users' natural language queries are converted by the LLM into Structured Query Language (SQL) statements, subsequently executed against a backend relational database. To create realistic attack conditions, the authors manually designed a set of malicious prompts to simulate direct prompt injection attacks, such as dropping database tables or bypassing access control restrictions.

To facilitate digital forensic readiness (DFR), the authors introduced structured logging mechanisms, termed LLM invocation logging, into their experimental system. Each invocation log entry captured essential forensic metadata including a timestamp, unique request identifier, input prompt (user's query), and corresponding LLM output, generating structured JSON-formatted logs, thereby ensuring traceability and forensic integrity.

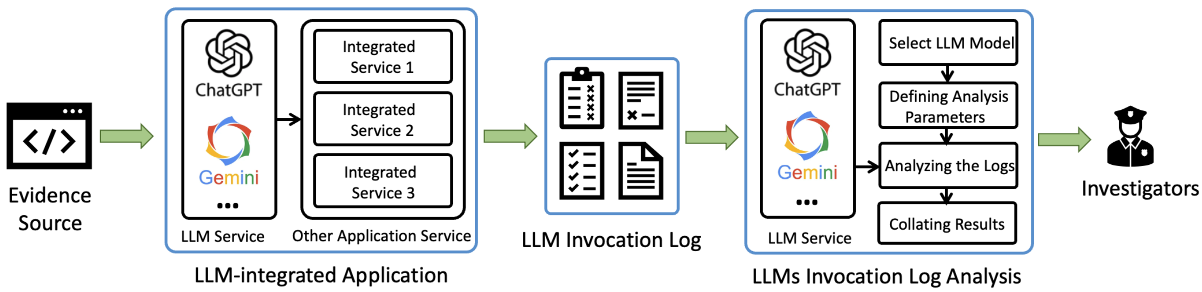


Fig. 3: Digital Forensic Analysis Workflow with LLM Invocation Logs.

For forensic analysis, the collected invocation logs were processed using an active analysis strategy in which multiple contemporary LLMs acted as forensic analysts. Given that different models have significantly varying context windows—for instance, from 8,182 tokens for llama3-70b-instruct to 1 million tokens for gemini-1.5-flash and gemini-1.5-pro, the authors evaluated analysis approaches both with models capable of accepting the entirety of the invocation logs as input and those requiring splitting log entries into smaller window chunks for sequential processing. Their analysis involved four main steps: **i)**

Selecting an LLM model for analysis, GPT-4, Gemini or other similar models; **ii) Defining key analysis parameters,** such as the LLM's temperature and context window size; **iii) Actively analyzing the logs with the chosen configuration,** using the chosen model and parameters; **iv) Collating the results,** summarizing key findings and observations. These steps were systematically repeated until all desired combinations of models and analysis parameters had been evaluated. Unlike previous works exploring LLM usage for anomaly detection that employed pre-summarization, this approach solely relied on active log analysis without context summary creation. Specifically, each model was provided log entries within a predefined context window, accompanied by instructions to identify potential security incidents and articulate justifications. The models returned structured JSON outputs indicating detection decisions (either "NORMAL" or "INCIDENT"), suspicious log indices, and descriptive reasoning. This direct approach significantly reduced the overall number of calls to the LLM, consequently decreasing both the total time required for log analysis and the potential cost.

This approach illustrates important advances in digital forensic readiness for LLMs-drivenn systems, showing how invocation log analysis performed by the LLM itself can provide practical forensic capabilities for identifying sophisticated hint injection attacks.

3.2.3 LLM-driven Mobile Evidence Contextual Analysis

Kim *et al.* propose a comprehensive and operationally grounded framework for mobile forensics, termed Mobile Evidence Contextual Analysis(MECA) [11]. This framework addresses the practical challenges law enforcement faces in analyzing large volumes of mobile messenger data, particularly under tight legal time constraints. Rather than relying solely on traditional keyword-based filtering, MECA leverages the contextual reasoning capabilities of LLMs to infer the presence of criminal intent or activity embedded in ambiguous or euphemistic language. The method is notable not only for its application to real-world forensic data but also for its holistic integration of forensic tools, data pre-processing, and prompt engineering.

The framework begins with the acquisition of mobile communication data using professional forensic software tools. Specifically, the authors employ MD-NEXT for physical data extraction and MD-RED for data parsing and visualization. These tools support the collection of structured communication records from seized smartphones, which are exported in formats like CSV or Excel for downstream processing. To ensure compliance with privacy and ethical standards, all personal identifiers within the dataset are anonymized using Named Entity Recognition (NER), with supplementary masking strategies applied to phone numbers and emails to minimize reidentification risk.

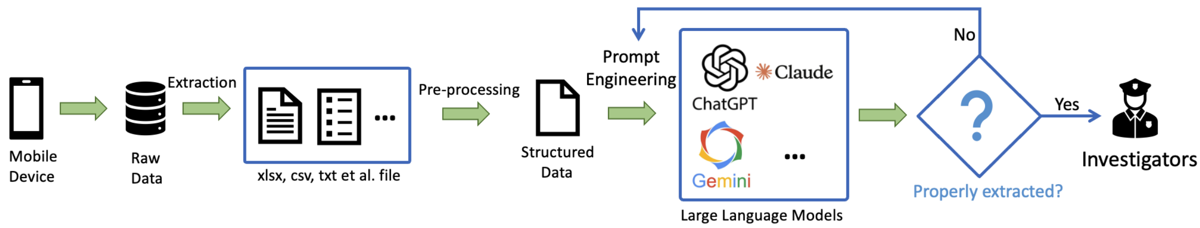


Fig. 4: Overview of LLM-driven Mobile Evidence Contextual Analysis Framework.

Given the size and fragmented nature of mobile information logs, the authors introduced a pre-processing phase to construct coherent units of analysis appropriate for LLM input. This involves applying initial keyword filters, *e.g.,* for terms such as"drugs", to identify potentially relevant messages. In order to preserve conversational context, each filtered message is augmented with surrounding messages in the same chat window, typically 40 lines each before and after the targeted message. This produces a set of context-rich message fragments that reflect real-world communication patterns and facilitate semantic interpretation by the model.

Central to MECA's effectiveness is its use of carefully crafted prompts to guide model behavior. Each prompt is designed to simulate the role of a forensic expert, instructing the model to evaluate whether a given message exchange is associated with criminal activity. The input is structured as key-value pairs, where the key represents the speaker and the value denotes the message content. Moreover, the authors implement a "Sandwich Prompting" technique—repeating instructions before and after the main content—to mitigate instruction forgetting, particularly in models like Gemini that may otherwise over-prioritize the input text.

Once the data and prompts are prepared, the framework employs three state-of-the-art LLMs, GPT-4o, Gemini 1.5, and Claude 3.5 to perform classification. Each model receives the structured conversational input and returns a binary judgment indicating whether the message set is relevant to the case. The authors also account for concerns around data privacy and model misuse by relying on commercial API deployments and explicitly documenting the privacy policies of each LLM provider. The use of multiple models not only allows performance benchmarking across architectures but also sets the stage for ensemble decision-making.

3.2.4 Forensic Analysis of Artifacts from Microsoft's Multi-Agent LLM Platform

In this work by Walker *et al.* proposes a comprehensive methodology for conducting forensic analysis of AutoGen, Microsoft's multi-agent LLM framework[114]. As AutoGen enables autonomous agent collaboration for task planning and execution, the forensic analysis of such systems introduces novel challenges, particularly in identifying, interpreting, and attributing the artifacts generated through agent interactions. The proposed methodology responds to this gap by establishing a structured, multi-layered approach to detecting the presence and behavior of AutoGen on a target system.

At the core of their approach is the idea of tracing the forensic footprint of LLM-driven agent interaction across three major layers of analysis: memory, disk, and network. Rather than focusing on any single modality of artifact, the methodology adopts a layered perspective to capture both persistent and volatile traces of AutoGen's activity on a host system. The authors hypothesize that, despite the encrypted nature of LLM-server communication and the ephemeral memory handling of modern OSes, a composite view of system-level behavior can reveal meaningful patterns associated with LLM agent activity.

The interaction model analyzed in the study involves two LLM-based agents: a UserProxyAgent, simulating a user that initiates tasks and evaluates responses; and an AssistantAgent, responsible for task execution. These agents interact through a feedback loop where task instructions and responses are exchanged programmatically. This model mirrors real-world use of AutoGen for distributed task planning and problem solving, and raises questions around forensic observability, *i.e.*, what traces of such interactions persist on a compromised or analyzed system.

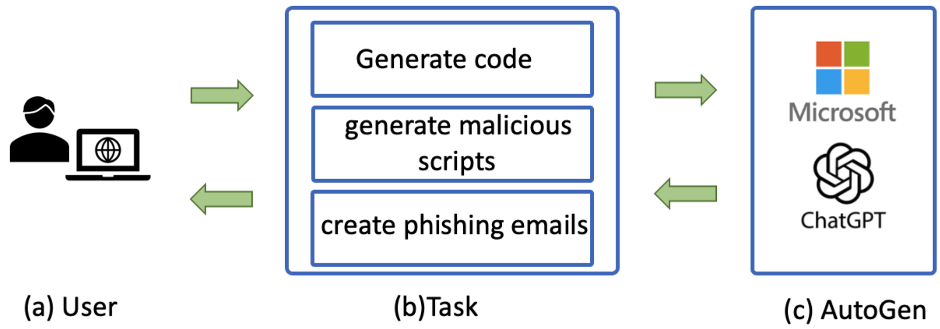


Fig. 5: Overview of forensic analysis of AutoGen.

To clarify this interaction workflow, Figure 5 illustrates the operational model used in the study. This process begins with a user operating from a local environment (A), where they initiate specific task prompts (B), such as generating code, crafting phishing emails, or producing malicious scripts. These prompts are passed to the AutoGen system (C), which coordinates interactions between LLM agents, typically a UserProxyAgent and an AssistantAgen, powered by models like GPT-3.5. The agents exchange messages programmatically until a task is completed, with AutoGen returning the model-generated output to the user. This controlled interaction loop is essential for generating forensic artifacts, the researchers are able to capture and later examine forensic artifacts across memory, disk, and network layers.

The forensic method involves isolating the key points where AutoGen interacts with the system or external services and mapping those to potential artifact locations. For instance, the UserProxyAgent's initial prompt, along with the AssistantAgent's responses, may be retained in memory buffers, cached in application files, or transiently recorded in system logs. The methodology accounts for the limited durability of such data and therefore incorporates the use of tools that can extract low-level system state information, *e.g.,* RAM dumps, temporary configuration files, browser traces.

A notable component of the method is its treatment of agent attribution—attempting to distinguish whether a given artifact was created by a human, a machine, or a cooperative agentic process. This is a particularly novel challenge in LLM forensics, since traditional forensic signatures are often agnostic to the cognitive or computational origin of content. The methodology, therefore, considers semantic and behavioral cues, *e.g.,* structure of prompt chains, repeated execution patterns, lack of GUI interaction, that may help differentiate machine-driven output from human-involved interaction.

Additionally, the approach integrates lightweight static analysis techniques, such as string extraction from memory and file systems, with dynamic signature correlation, such as identifying AutoGen-related modules in Python environments or connections to known LLM service endpoints. This hybrid approach helps mitigate the limitations of any single forensic strategy and provides a more comprehensive account of AutoGen's presence and behavior on the system.

The method sets a foundation for future forensic analysis of autonomous LLM systems, especially as they become more modular, compositional, and capable of unsupervised behavior. It emphasizes the need for multi-perspective evidence gathering, cross-layer correlation, and a deeper understanding of agent-based software design in order to maintain accountability in increasingly AI-driven digital environments.

3.2.5 The Local LLM-driven Framework for Digital Forensic

While large language models (LLMs) have demonstrated remarkable capability across various natural language processing tasks, their application in sensitive domains such as digital forensics presents unique challenges, including concerns about data privacy, security, and the need for specialized domain knowledge. Moreover, reliance on cloud-based solutions can introduce vulnerabilities related to data confidentiality and compliance, prompting the need for locally deployable LLMs tailored specifically to forensic purposes. Addressing these critical issues, Sharma *et al.* introduced ForensicLLM, a specialized, locally deployable large language model designed explicitly for digital forensic applications using a retrieval-augmented fine-tuning (RAFT) methodology [115].

Sharma *et al.* utilized Meta's LLaMA-3.1–8B as the foundational model, enhancing it through fine-tuning with domain-specific content to address the unique reasoning demands inherent in digital forensic investigations. They began by compiling an extensive corpus comprising 1,082 peer-reviewed research articles sourced from the journal Forensic Science International: Digital Investigation, along with metadata extracted from 1,390 verified digital forensic artifacts obtained via the Artifact Genome Project. Textual contents from these research articles were segmented into semantically meaningful chunks of approximately 2,000 characters and embedded using the UAE-Large-V1 embedding model. Each chunk was enriched with associated metadata, including article titles and authors, with embeddings subsequently stored within a ChromaDB vector database to facilitate efficient retrieval during subsequent training and inference processes.

In the absence of suitable labeled question-answer datasets specific to digital forensic scenarios, the authors employed GPT-4 Turbo to generate approximately 10,000 synthetic question-answer pairs based directly upon the prepared literature corpus. This generation process was carefully guided using detailed prompting to ensure practically relevant, technically accurate content, maintaining faithful adherence to original source citations following APA standards.

The fine-tuning procedure leveraged Quantized Low-Rank Adaptation, implementing a 4-bit quantization approach to optimize computational resource efficiency during training. Sharma *et al.* adopted the Axolotl framework, utilizing standard practices such as cosine learning rate scheduling and early stopping based on validation set performance. During inference, ForensicLLM utilizes a retrieval-augmented generation (RAG) strategy, embedding user queries to dynamically retrieve relevant textual contexts from the vector database, which are then integrated into the model input to produce informed, verifiable, and accurate responses.

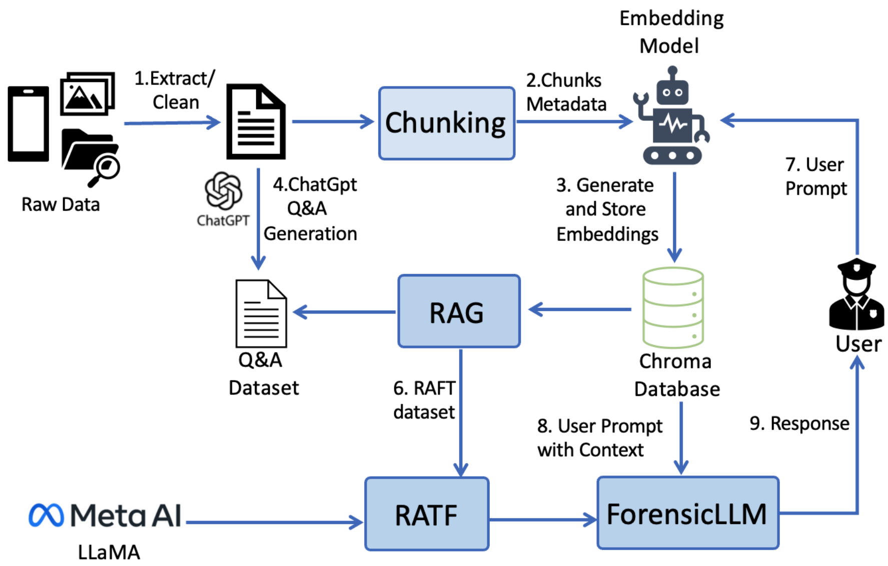


Fig. 6: Overview of Retrieval-Augmented Fine-tuning (RAFT) for ForensicLLM.

As shown in Figure 6, this figure outlines the sequence data processing and model-training pipeline, beginning with raw data extraction and cleaning, followed by segmentation into meaningful textual chunks. These text chunks, enriched with metadata, are then transformed into semantic embeddings using an embedding model and subsequently stored in a ChromaDB vector database. Simultaneously, synthetic Q&A pairs are generated from the corpus using GPT-4 to form a structured training dataset. This Q&A dataset is integrated with context retrieved from the vector database, forming the RAFT dataset utilized for fine-tuning the ForensicLLM model. Finally, during inference, user queries are embedded and matched with relevant contexts retrieved from Chroma datebase, enabling ForensicLLM to produce accurate, contextually informed, and traceable responses tailored specifically for digital forensic applications.

The retrieval-enhanced fine-tuning approach proposed by Sharma *et al.* significantly impacts digital forensic practice by reducing common limitations associated with general-purpose language models, particularly hallucinations and factual inaccuracies. Their quantitative and qualitative evaluations demonstrated that ForensicLLM substantially improves response accuracy, relevance, and reliability, thus equipping forensic investigators with trustworthy, traceable analytical support capable of meeting rigorous evidentiary standards required in real-world forensic investigations.

4 Challenges and Limitations of Leveraging LLM in Digital Forensics

The integration of large language models into digital forensics workflows has generated increasing interest due to their potential in automating documentation, evidence analysis, and decision support. However, their use also presents numerous challenges that arise both from the inherent properties of LLMs and from the specific requirements of forensic practice.

4.1 LLM Inherent Challenges

Several limitations are intrinsic to the architecture and training methodology of LLMs, which can hinder their safe and reliable deployment in forensic investigations.

Hallucinations. A prominent concern when employing LLMs is their propensity to produce hallucinated content—output that is grammatically coherent yet factually incorrect or fabricated. In the context of digital forensics, such inaccuracies can lead to the generation of false leads, thereby misguiding the investigation or introducing inadmissible evidence. For instance, in a controlled trial conducted by a cybersecurity firm, an LLM-generated case summary falsely inferred a link between an employee and a foreign contact based solely on contextual cues in a benign conversation log. This example highlights the necessity of human verification mechanisms prior to integrating LLM-generated information into forensic reports.

Interpretability and Explainability. LLMs often exhibit poor explainability due to their black-box nature. While they can produce accurate results in many domains, their decision-making pathways are not transparent [110]. This opacity becomes a critical issue in forensic analysis, where the rationale behind evidence interpretation must be traceable and defensible. In one documented instance during a civil litigation case, an LLM used in pre-trial discovery flagged certain emails as "suspicious"; however, when the opposing counsel requested an explanation for these classifications, the legal team was unable to articulate the reasoning behind the model's output. The lack of explainability ultimately led to the exclusion of the generated evidence.

Lack of Domain-Specific Knowledge. General-purpose LLMs are trained on heterogeneous and largely non-specialized corpora. As such, they may not have the technical nuance necessary for forensic analysis. For example, when prompted to assess the contents of a memory dump, a widely used LLM erroneously flagged "svchost.exe" as malicious, failing to account for the legitimate role of the process in Windows systems. Such errors underscore the risk of applying unadapted LLMs in technical domains without appropriate domain fine-tuning.

Bias and Fairness. Bias in LLMs-driven a reflection of the biases present in the training data—poses ethical and practical risks in forensic contexts [24]. Investigative results may be biased either by reinforcing existing stereotypes or by systematically prioritizing certain types of evidence. In a pilot study involving multilingual forensic datasets, an LLM-assisted classification system consistently deprioritized non-English chat logs, leading to a delay in the examination of relevant Arabic-language communications. This form of bias, if left unaddressed, could have far-reaching implications for fairness and due process in digital investigations [116].

4.2 Digital Forensics-Specific Challenges

Although the inherent risks associated with LLMs pose general concerns in all domains, deploying these models within digital forensic workflows introduces additional challenges. Digital forensics imposes strict standards regarding evidence integrity, reproducibility, and procedural compliance, and these established forensic principles may conflict with the nature of LLM technologies. Consequently, integrating LLMs into digital forensic practices requires addressing specific challenges related to evidentiary standards, reproducibility, prompt sensitivity, standardization, and practitioner readiness.

Chain of Custody and Evidentiary Integrity. A core principle in forensic science is the preservation of chain of custody, that is, the ability to trace each step of evidence handling. When LLMs are employed, especially in cloud-based or third-party systems, questions arise regarding the preservation and auditability of evidence. In one European law enforcement case study, the use of an LLM to summarize mobile device contents inadvertently violated chain of custody procedures, as intermediate outputs were not systematically logged. As a result, the forensic findings were challenged on procedural grounds during judicial review.

Non-determinism and Reproducibility. Unlike deterministic forensic tools, LLMs are inherently probabilistic and may produce variable outputs even under identical input conditions. This variability undermines one of the key requirements of forensic science, namely reproducibility. In a university-led evaluation, an LLM used to reconstruct activity timelines from log data produced inconsistent event sequences across multiple runs. Such behavior poses serious threats to the reliability of forensic conclusions, particularly when outputs are used as part of expert witness testimony.

Prompt Sensitivity. Related to non-determinism is the issue of prompt sensitivity, whereby subtle variations in phrasing can lead to significantly different model outputs. For instance, altering a prompt from "summarize suspicious behavior" to "summarize all activity" led an LLM to either omit or include key lateral movement indicators in the same dataset. The fragility of outputs based on minor linguistic changes necessitates rigorous prompt engineering and version control when using LLMs in evidentiary contexts.

Lack of Standardization. There exists no established framework or industry-wide standard governing the use of LLMs in digital forensics. This absence of formal guidance has resulted in inconsistencies across investigative practices and raises concerns regarding admissibility and procedural fairness. In a simulated case involving two independent forensic teams, divergent conclusions were reached due to differences in prompt design, evidence filtering strategies, and LLM configurations. These discrepancies emphasize the need for standardized protocols and certification schemes for LLM-based forensic tools.

Training and Expertise Requirements. The adoption of LLMs in digital forensic settings introduces new requirements for practitioner training. Investigators must possess not only technical forensic skills but also basic knowledge in AI, prompt design, and model validation. A field test conducted with junior investigators revealed that improper prompt use led to misclassifications of a legitimate mobile application as malicious, an error that could have been avoided with minimal training in AI reasoning mechanisms. The integration of LLMs thus demands a reevaluation of existing forensic training curricula to include AI literacy.

5 Future Directions

The intersection of large language models (LLMs) and digital forensics represents an emerging frontier with significant potential for transforming forensic investigations. Future research in this area promises to strengthen evidentiary integrity, promote greater accountability, and contribute broadly to societal trust and justice.

5.1 Multi-Modal and Cross-Data Analysis

Digital forensic investigations increasingly require holistic evidence interpretation across various data modalities—textual logs, network traffic, memory dumps, images, and audio. Emerging multi-modal LLMs (MLLMs) suggest promising capabilities for integrating diverse data forms into unified analytical frameworks. Integrating vision and language models could enable forensic assistants to analyze screenshots, correlate textual logs with visual artifacts, or interpret combined structured and unstructured forensic data. Research opportunities lie in developing robust multi-modal forensic LLMs capable of seamlessly analyzing multiple data types while maintaining accuracy across different modalities. Achieving this will require interdisciplinary collaboration and innovative design to bridge existing capability gaps.

5.2 Explainability and Trust in LLM-Driven Analysis

The inherently opaque reasoning of LLMs conflicts with forensic requirements for transparency and verifiability. Enhancing the explainability of LLM outputs is thus critical for building investigator trust. Future research should focus on methods that enable LLMs to justify their conclusions with explicit evidence references and step-by-step reasoning processes. Techniques like retrieval-augmented generation, where LLM outputs are grounded explicitly in input data and known forensic knowledge, can significantly improve credibility. Validation methods, such as cross-validation with multiple models or human-in-the-loop verification, should also be investigated to detect and mitigate errors and biases inherent in AI analyses.

5.3 Domain-Specific LLMs Across Forensic Disciplines

One crucial future direction involves the development of specialized, domain-specific LLMs tailored explicitly for various forensic applications such as memory forensics, malware analysis, network investigations, and log interpretation. General-purpose models typically lack the specialized technical understanding required to interpret detailed forensic artifacts accurately. Early examples, such as volGPT for memory analysis, have demonstrated the effectiveness of fine-tuned LLMs in accurately identifying ransomware processes while providing comprehensive explanations. Future research should systematically explore domain-specific models for forensic tasks, including artifact interpretation, filesystem analysis, and forensic triage. This specialization will necessitate creating dedicated forensic datasets, posing challenges related to data sensitivity and privacy that researchers must address through synthetic or anonymized datasets.

5.4 Privacy and Legal Admissibility Challenges

Integrating LLMs into forensic investigations raises significant privacy concerns and legal admissibility challenges. Public cloud-based solutions often conflict with chain-of-custody requirements, prompting the need for secure, offline LLM solutions deployable within forensic lab environments. Future research should focus on enhancing on-premise or federated AI models that preserve data confidentiality and comply with legal standards. Additionally, clearly defined legal frameworks and standards are needed for documenting and certifying AI processes, ensuring their outputs withstand judicial scrutiny. Collaborative research among technologists, legal scholars, and policymakers is necessary to bridge these gaps and ensure that LLM-assisted forensic analyses meet rigorous evidentiary standards.

5.5 Integration with Traditional Forensic Tools and Workflows

Future research must explore the seamless integration of LLMs into existing forensic software and investigative workflows. Embedding interactive AI assistants within forensic suites, enabling natural language querying, automated artifact parsing, and AI-driven script generation, can significantly enhance investigative efficiency. Ensuring these integrations are robust and error resistant, and maintaining compatibility with existing forensic processes, evidence documentation systems, and investigative protocols, represents a significant technical challenge. Interdisciplinary collaboration will be crucial in developing user-centric, reliable forensic tools augmented by AI capabilities.

5.6 Standardized Evaluation and Benchmarking

A critical gap in current research is the lack of standardized evaluation frameworks for assessing LLM effectiveness and reliability in forensic contexts. Developing shared benchmark datasets, standardized metrics for accuracy, explainability, and utility, and consistent evaluation methodologies is essential for objectively comparing different LLM approaches. Community-driven benchmarking initiatives, similar to established cybersecurity and computer vision evaluations, should be prioritized to accelerate progress and ensure rigorous validation of AI-assisted forensic tools.

6 Conclusion

Large Language Models (LLMs) have emerged as transformative tools that significantly automate and augment forensic capabilities, thus reshaping the landscape of digital investigations. This paper systematically explored how LLMs have revolutionized digital forensic approaches, providing a comprehensive and accessible overview for practitioners and researchers alike. Through practical examples and real-world scenarios, we illustrated the superior capabilities of LLMs in enhancing analytical accuracy, efficiency, and scalability in forensic workflows. However, the integration of LLMs into digital forensic processes is not without challenges; issues such as model hallucinations, interpretability, biases, and ethical considerations necessitate cautious and informed application. Addressing these challenges requires further research that focuses on improving transparency, accountability, and standardization in the forensic use of LLM technologies. Ultimately, the thoughtful integration of LLMs holds significant promise in advancing digital forensic practices, fostering trust and reliability, and contributing to more equitable and just judicial outcomes.

References

[1] Wickramasekara, A., Breitinger, F., Scanlon, M.: Exploring the potential of large language models for improving digital forensic investigation efficiency. Forensic Science International: Digital Investigation **52**, 301859 (2025)

[2] Rahman, M.N., Mohammad, T., Virtanen, S.: Leveraging large language models for network traffic analysis: Design, implementation, and evaluation of an llm-powered system for cyber incident reconstruction (2024)

[3] Xu, E., Zhang, W., Xu, W.: Transforming digital forensics with large language models: Unlocking automation, insights, and justice. In: Proceedings of the 33rd ACM International Conference on Information and Knowledge Management, pp. 5543–5546 (2024)

[4] Rogers, M.K.: A two-dimensional circumplex approach to the development of a hacker taxonomy. Digital investigation **3**(2), 97–102 (2006)

[5] Ismail, M.: Sony pictures and the us federal government: a case study analysis of the sony pictures entertainment hack crisis using normal accidents theory (2017)

[6] Marmura, S.M., Marmura, S.M.: Wikileaks' american moment: The dnc emails, russiagate and beyond. The WikiLeaks Paradigm: Paradoxes and Revelations, 109–133 (2018)

[7] Confessore, N., Eder, S., October, L.: In hacked dnc emails, a glimpse of how big money works. The New York Times (2016)

[8] Minnaar, A.: Online'underground'marketplaces for illicit drugs: the prototype case of the dark web website'silk road. Acta Criminologica: African Journal of Criminology & Victimology **30**(1), 23–47 (2017)

[9] Lacson, W., Jones, B.: The 21st century darknet market: lessons from the fall of silk road. International Journal of Cyber Criminology **10**(1), 40 (2016)

[10] Negangard, E.M., Fay, R.G.: Electronic discovery (ediscovery): Performing the early stages of the enron investigation. Issues in Accounting Education **35**(1), 43–58 (2020)

[11] Kim, K., Lee, C., Bae, S., Choi, J., Kang, W.: Digital forensics in law enforcement: A case study of llm-driven evidence analysis. Available at SSRN 5110258

[12] Quick, D., Choo, K.-K.R.: Digital forensic intelligence: Data subsets and open source intelligence (dfint+ osint): A timely and cohesive mix. Future Generation Computer Systems **78**, 558–567 (2018)

[13] Chen, H.-Y.: Cloud crime to traditional digital forensic legal and technical challenges and countermeasures. In: 2014 IEEE Workshop on Advanced Research and Technology in Industry Applications (WARTIA), pp. 990–994 (2014). IEEE

[14] Fernando, K.: A multidimensional framework for utilizing big data analytics and ai in strengthening digital forensics and cybersecurity investigations. International Journal of Cybersecurity Risk Management, Forensics, and Compliance **7**(12), 16–30 (2023)

[15] Malik, A.W., Bhatti, D.S., Park, T.-J., Ishtiaq, H.U., Ryou, J.-C., Kim, K.-I.: Cloud digital forensics: Beyond tools, techniques, and challenges. Sensors **24**(2), 433 (2024)

[16] Garach, J., Singh, S.K., Reddy, A.P.C., Khan, H., et al.: A comprehensive review on artificial intelligence in digital forensics with taxonomies, issues, and solutions: Ai in digital forensics. Strategies for E-Commerce Data Security: Cloud, Blockchain, AI, and Machine Learning, 1–28 (2024)

[17] Wang, Z., Saxena, N., Yu, T., Karki, S., Zetty, T., Haque, I., Zhou, S., Kc, D., Stockwell, I., Bifet, A., *et al.*: Preventing discriminatory decision-making in evolving data streams. In: Proceedings of the 2023 ACM Conference on Fairness, Accountability, and Transparency (FAccT) (2023)

[18] Zhang, W., Wang, Z., Kim, J., Cheng, C., Oommen, T., Ravikumar, P., Weiss, J.: Individual fairness under uncertainty. In: 26th European Conference on Artificial Intelligence, pp. 3042–3049 (2023)

[19] Wang, Z., Wallace, C., Bifet, A., Yao, X., Zhang, W.: Fg^{2}an: Fairness-aware graph generative adversarial networks. In: Joint European Conference on Machine Learning and Knowledge Discovery in Databases, pp. 259–275 (2023). Springer Nature Switzerland

[20] Yazdani, S., Saxena, N., Wang, Z., Wu, Y., Zhang, W.: A comprehensive survey of image and video generative ai: Recent advances, variants, and applications (2024)

[21] Wang, Z., Narasimhan, G., Yao, X., Zhang, W.: Mitigating multisource biases in graph neural networks via real counterfactual samples. In: 2023 IEEE International Conference on Data Mining

(ICDM), pp. 638–647 (2023). IEEE

[22] Chinta, S.V., Fernandes, K., Cheng, N., Fernandez, J., Yazdani, S., Yin, Z., Wang, Z., Wang, X., Xu, W., Liu, J., *et al.*: Optimization and improvement of fake news detection using voting technique for societal benefit. In: 2023 IEEE International Conference on Data Mining Workshops (ICDMW), pp. 1565–1574 (2023). IEEE

[23] Wang, Z., Chu, Z., Doan, T.V., Ni, S., Yang, M., Zhang, W.: History, development, and principles of large language models: an introductory survey. AI and Ethics, 1–17 (2024)

[24] Chu, Z., Wang, Z., Zhang, W.: Fairness in large language models: A taxonomic survey. ACM SIGKDD explorations newsletter **26**(1), 34–48 (2024)

[25] Dzuong, J., Wang, Z., Zhang, W.: Uncertain boundaries: Multidisciplinary approaches to copyright issues in generative ai. arXiv preprint arXiv:2404.08221 (2024)

[26] Yin, Z., Wang, Z., Zhang, W.: Improving fairness in machine learning software via counterfactual fairness thinking. In: Proceedings of the 2024 IEEE/ACM 46th International Conference on Software Engineering: Companion Proceedings, pp. 420–421 (2024)

[27] Wang, Z., Zhou, Y., Haque, I., Lo, D., Zhang, W.: Towards fair machine learning software: Understanding and addressing model bias through counterfactual thinking. arXiv preprint arXiv:2302.08018 (2023)

[28] Wang, Z., Qiu, M., Chen, M., Salem, M.B., Yao, X., Zhang, W.: Toward fair graph neural networks via real counterfactual samples. Knowledge and Information Systems, 1–25 (2024)

[29] Chinta, S.V., Wang, Z., Yin, Z., Hoang, N., Gonzalez, M., Quy, T.L., Zhang, W.: Fairaied: Navigating fairness, bias, and ethics in educational ai applications. arXiv preprint arXiv:2407.18745 (2024)

[30] Doan, T.V., Wang, Z., Hoang, N.N.M., Zhang, W.: Fairness in large language models in three hours. In: Proceedings of the 33rd ACM International Conference on Information and Knowledge Management, pp. 5514–5517 (2024)

[31] Chinta, S.V., Wang, Z., Zhang, X., Viet, T.D., Kashif, A., Smith, M.A., Zhang, W.: Ai-driven healthcare: A survey on ensuring fairness and mitigating bias. arXiv preprint arXiv:2407.19655 (2024)

[32] Wang, Z., Dzuong, J., Yuan, X., Chen, Z., Wu, Y., Yao, X., Zhang, W.: Individual fairness with group awareness under uncertainty. In: Joint European Conference on Machine Learning and Knowledge Discovery in Databases, pp. 89–106 (2024). Springer Nature Switzerland

[33] Wang, Z., Palikhe, A., Yin, Z., Zhang, W.: Fairness definitions in language models explained. arXiv preprint arXiv:2407.18454 (2024)

[34] Wang, Z., Chu, Z., Blanco, R., Chen, Z., Chen, S.-C., Zhang, W.: Advancing graph counterfactual fairness through fair representation learning. In: Joint European Conference on Machine Learning and Knowledge Discovery in Databases, pp. 40–58 (2024). Springer Nature Switzerland

[35] Swanson, C.: Bullets and ballots: Exploring the effects of nearly successful assassination attempts on general election performance in the united states. The UWJPS is thankful for the continued support of the Department of Political Science at the University of Washington. In addition, we are grateful to the students who submitted their work and ideas., 15

[36] The New York Times: Investigators Unlock Gunman's Phone in Search for Motive in Trump Shooting. Accessed: 2025-03-30. https://www.nytimes.com/live/2024/07/15/us/trump-shooting-investigation

[37] Bartoletti, M., Lande, S., Loddo, A., Pompianu, L., Serusi, S.: Cryptocurrency scams: analysis and perspectives. Ieee Access **9**, 148353–148373 (2021)

[38] Cimpanu, C.: How the FBI Tracked down the Twitter Hackers. Accessed: 2025-03-30. https://www.zdnet.com/article/how-the-fbi-tracked-down-the-twitter-hackers/

[39] Kessler, G.C., Phillips, A.M.: Cryptography, passwords, privacy, and the fifth amendment. Journal of Digital Forensics, Security and Law **15**(2), 2 (2020)

[40] Clarke, C.: The pensacola terrorist attack: The enduring influence of al-qaida and its affiliates. CTC Sentinel **13**(3) (2020)

[41] Vaghela, R., Gowda, V.D., Taj, M., Arudra, A., Chopra, M.: Digital evidence collection and preservation in computer network forensics. In: Handbook of Research on Innovative Approaches to Information Technology in Library and Information Science, pp. 42–62 (2024)

[42] Allam, H.: FBI: New iPhone Evidence Shows Pensacola Shooter Had Ties To Al-Qaida. Accessed: 2025-03-30. https://www.npr.org/2020/05/18/857932909/fbi-new-iphone-evidence-shows-pensacola-shooter-had-ties-to-al-qaida

[43] Nayak, M.: Ai-enhanced digital forensics: Automated techniques for efficient investigation and evidence collection. J. Electrical Systems **20**(1s), 211–229 (2024)

[44] Akeiber, H.J.: A comprehensive study of cybercrime and digital forensics through machine learning and ai. Al-Rafidain Journal of Engineering Sciences, 369–395 (2025)

[45] Liu, J., Kong, Z., Zhao, P., Yang, C., Tang, H., Shen, X., Yuan, G., Niu, W., Zhang, W., Lin, X., et al.: Toward adaptive large language models structured pruning via hybrid-grained weight importance assessment. arXiv preprint arXiv:2403.10799 (2024)

[46] Jin, H., Wei, W., Wang, X., Zhang, W., Wu, Y.: Rethinking learning rate tuning in the era of large language models. In: 2023 IEEE 5th International Conference on Cognitive Machine Intelligence (CogMI), pp. 112–121 (2023). IEEE

[47] Ferrag, M.A., Alwahedi, F., Battah, A., Cherif, B., Mechri, A., Tihanyi, N., Bisztray, T., Debbah, M.: Generative ai in cybersecurity: A comprehensive review of llm applications and vulnerabilities. Internet of Things and Cyber-Physical Systems (2025)

[48] Yao, Y., Duan, J., Xu, K., Cai, Y., Sun, Z., Zhang, Y.: A survey on large language model (llm) security and privacy: The good, the bad, and the ugly. High-Confidence Computing, 100211 (2024)

[49] Valmeekam, K., Olmo, A., Sreedharan, S., Kambhampati, S.: Large language models still can't plan (a benchmark for llms on planning and reasoning about change). In: NeurIPS 2022 Foundation Models for Decision Making Workshop (2022)

[50] Kumarage, T., Agrawal, G., Sheth, P., Moraffah, R., Chadha, A., Garland, J., Liu, H.: A survey of ai-generated text forensic systems: Detection, attribution, and characterization. arXiv preprint arXiv:2403.01152 (2024)

[51] Ahmed, M., Mahmood, A.N., Hu, J.: A survey of network anomaly detection techniques. Journal of Network and Computer Applications **60**, 19–31 (2016)

[52] Liu, C., Xie, X., Zhang, X., Cui, Y.: Large language models for networking: Workflow, advances and challenges. IEEE Network (2024)

[53] Velasco, C.: Cybercrime and artificial intelligence. an overview of the work of international organizations on criminal justice and the international applicable instruments. In: ERA Forum, vol. 23, pp. 109–126 (2022). Springer

[54] Mijwil, M.M., Aljanabi, M., ChatGPT, C.: Towards artificial intelligence-based cybersecurity: The practices and chatgpt generated ways to combat cybercrime. Iraqi Journal For Computer Science and Mathematics **4**(1), 8 (2023)

[55] Zhang, R., Xie, M.: Forensiq: A knowledge graph question answering system for iot forensics. In: International Conference on Digital Forensics and Cyber Crime, pp. 300–314 (2023). Springer

[56] Siddiqui, M.Z., Yadav, S., Husain, M.S.: Application of artificial intelligence in fighting against cyber crimes: a review. Int. J. Adv. Res. Comput. Sci **9**(2), 118–122 (2018)

[57] Chen, Z., Mao, H., Li, H., Jin, W., Wen, H., Wei, X., Wang, S., Yin, D., Fan, W., Liu, H., *et al.*: Exploring the potential of large language models (llms) in learning on graphs. ACM SIGKDD Explorations Newsletter **25**(2), 42–61 (2024)

[58] Yang, J., Jin, H., Tang, R., Han, X., Feng, Q., Jiang, H., Zhong, S., Yin, B., Hu, X.: Harnessing the power of llms in practice: A survey on chatgpt and beyond. ACM Transactions on Knowledge Discovery from Data **18**(6), 1–32 (2024)

[59] Smirnov, E.: Enhancing qualitative research in psychology with large language models: a methodological exploration and examples of simulations. Qualitative Research in Psychology **22**(2), 482–512 (2025)

[60] Kao, H.-H.: Accelerating multilingual cryptocurrency forensics: An nlp-driven approach for efficient mnemonic identification. IEEE Access (2025)

[61] Karie, N.M., Kebande, V.R., Venter, H.: Diverging deep learning cognitive computing techniques into cyber forensics. Forensic Science International: Synergy **1**, 61–67 (2019)

[62] Arshad, H., Jantan, A.B., Abiodun, O.I.: Digital forensics: review of issues in scientific validation of digital evidence. Journal of Information Processing Systems **14**(2), 346–376 (2018)

[63] Klasén, L., Fock, N., Forchheimer, R.: The invisible evidence: Digital forensics as key to solving crimes in the digital age. Forensic science international **362**, 112133 (2024)

[64] Daniel, L., Daniel, L.: Digital Forensics for Legal Professionals: Understanding Digital Evidence from the Warrant to the Courtroom, (2011)

[65] Caballero, E.Q.: Leveraging large language models for legal document understanding and software system analysis: Addressing key challenges. PhD thesis, Baylor University (2024)

[66] Akhtar, S., Khan, S., Parkinson, S.: Llm-based event log analysis techniques: A survey. arXiv preprint arXiv:2502.00677 (2025)

[67] Labajová, L.: The state of AI: Exploring the perceptions, credibility, and trustworthiness of the users towards AI-Generated Content (2023)

[68] Khlaif, Z.N., Mousa, A., Hattab, M.K., Itmazi, J., Hassan, A.A., Sanmugam, M., Ayyoub, A.: The potential and concerns of using ai in scientific research: Chatgpt performance evaluation. JMIR Medical Education **9**, 47049 (2023)

[69] Raza, H.: Ai-driven assessment: Reliability, bias, and ethical implications. Journal of AI in Education: Innovations, Opportunities, Challenges, and Future Directions **1**(2), 36–47 (2024)

[70] Azodi, C.B., Tang, J., Shiu, S.-H.: Opening the black box: interpretable machine learning for geneticists. Trends in genetics **36**(6), 442–455 (2020)

[71] Quang Huy, P., Kien Phuc, V.: Insight into how legal and ethical considerations of artificial intelligence enhance the effectiveness of cyber forensic accounting. Journal of Global Information

Technology Management, 1–31 (2025)

[72] Wischmeyer, T.: Artificial intelligence and transparency: opening the black box. In: Regulating Artificial Intelligence, pp. 75–101 (2019)

[73] Djeffal, C.: Artificial intelligence and public governance: normative guidelines for artificial intelligence in government and public administration. In: Regulating Artificial Intelligence, pp. 277–293 (2019)

[74] Cath, C.: Governing artificial intelligence: ethical, legal and technical opportunities and challenges. Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences **376**(2133), 20180080 (2018)

[75] Baror, S.O., Venter, H.S., Adeyemi, R.: A natural human language framework for digital forensic readiness in the public cloud. Australian Journal of Forensic Sciences **53**(5), 566–591 (2021)

[76] Jain, A.: Enhancing forensic analysis of digital evidence using machine learning: Techniques, applications, and challenges. International Journal of Innovative Research in Multidisciplinary Perspectives and Studies (IJIRMPS), 1–8 (2024)

[77] Wang, Z., Zhang, W.: Group fairness with individual and censorship constraints. In: 27th European Conference on Artificial Intelligence (2024)

[78] Wang, Z., Ulloa, D., Yu, T., Rangaswami, R., Yap, R., Zhang, W.: Individual fairness with group constraints in graph neural networks. In: 27th European Conference on Artificial Intelligence (2024)

[79] Yin, Z., Agarwal, S., Kashif, A., Gonzalez, M., Wang, Z., Liu, S., Liu, Z., Wu, Y., Stockwell, I., Xu, W., *et al.*: Accessible health screening using body fat estimation by image segmentation. In: 2024 IEEE International Conference on Data Mining Workshops (ICDMW), pp. 405–414 (2024)

[80] Wang, Z., Yin, Z., Zhang, Y., Yang, L., Zhang, T., Pissinou, N., Cai, Y., Hu, S., Li, Y., Zhao, L., *et al.*: Fg-smote: Towards fair node classification with graph neural network. ACM SIGKDD Explorations Newsletter **26**(2), 99–108 (2025)

[81] Wang, Z., Yin, Z., Liu, F., Liu, Z., Lisetti, C., Yu, R., Wang, S., Liu, J., Ganapati, S., Zhou, S., *et al.*: Graph fairness via authentic counterfactuals: Tackling structural and causal challenges. ACM SIGKDD Explorations Newsletter **26**(2), 89–98 (2025)

[82] Wang, Z., Chu, Z., Viet Doan, T., Wang, S., Wu, Y., Palade, V., Zhang, W.: Fair graph u-net: A fair graph learning framework integrating group and individual awareness. In: Proceedings of the AAAI Conference on Artificial Intelligence (2025)

[83] Wang, Z., Hoang, N., Zhang, X., Bello, K., Zhang, X., Iyengar, S.S., Zhang, W.: Towards fair graph learning without demographic information. In: The 28th International Conference on Artificial Intelligence and Statistics (2025)

[84] Zhang, W.: Fairness with censorship: Bridging the gap between fairness research and real-world deployment. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 38, pp. 22685–22685 (2024)

[85] Zhang, W.: Ai fairness in practice: Paradigm, challenges, and prospects. Ai Magazine (2024)

[86] Zhang, W., Ntoutsi, E.: Faht: an adaptive fairness-aware decision tree classifier. In: International Joint Conference on Artificial Intelligence (IJCAI), pp. 1480–1486 (2019)

[87] Zhang, W., Weiss, J.: Fair decision-making under uncertainty. In: 2021 IEEE International Conference on Data Mining (ICDM) (2021). IEEE

[88] Zhang, W., Weiss, J.C.: Longitudinal fairness with censorship. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 36, pp. 12235–12243 (2022)

[89] Zhang, W., Hernandez-Boussard, T., Weiss, J.: Censored fairness through awareness. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 37, pp. 14611–14619 (2023)

[90] Zhang, W., Zhou, S., Walsh, T., Weiss, J.C.: Fairness amidst non-iid graph data: A literature review. AI Magazine **46**(1), 12212 (2025)

[91] Zhang, W., Weiss, J.C.: Fairness with censorship and group constraints. Knowledge and Information Systems, 1–24 (2023)

[92] Zhang, W., Zhang, L., Pfoser, D., Zhao, L.: Disentangled dynamic graph deep generation. In: Proceedings of the 2021 SIAM International Conference on Data Mining (SDM), pp. 738–746 (2021). SIAM

[93] Casey, E.: Digital Evidence and Computer Crime: Forensic Science, Computers, and the Internet, (2011)

[94] Walker, C.: Digital evidence and computer crime: Forensic science, computers and the internet. Crime Prevention and Community Safety **3**, 87–88 (2001)

[95] Sharevski, F.: Rules of professional responsibility in digital forensics: A comparative analysis. Journal of Digital Forensics, Security and Law **10**(2), 3 (2015)

[96] Ademu, I.O., Imafidon, C.O., Preston, D.S.: A new approach of digital forensic model for digital forensic investigation. International Journal of Advanced Computer Science and Applications **2**(12) (2011)

[97] Quick, D., Choo, K.-K.R.: Data reduction and data mining framework for digital forensic evidence: storage, intelligence, review and archive. Trends and Issues in Crime and Criminal Justice (480), 1–11 (2014)

[98] Quick, D., Choo, K.-K.R.: Impacts of increasing volume of digital forensic data: A survey and future research challenges. Digital Investigation **11**(4), 273–294 (2014)

[99] Lillis, D., Becker, B., O'Sullivan, T., Scanlon, M.: Current challenges and future research areas for digital forensic investigation. arXiv preprint arXiv:1604.03850 (2016)

[100] Vincze, E.A.: Challenges in digital forensics. Police Practice and Research **17**(2), 183–194 (2016)

[101] Rowlingson, R., *et al.*: A ten step process for forensic readiness. International Journal of Digital Evidence **2**(3), 1–28 (2004)

[102] Amato, F., Cozzolino, G., Mazzeo, A., Mazzocca, N.: Correlation of digital evidences in forensic investigation through semantic technologies. In: 2017 31st International Conference on Advanced Information Networking and Applications Workshops (WAINA), pp. 668–673 (2017). IEEE

[103] Horsman, G.: The importance of digital evidence strategies. Wiley Interdisciplinary Reviews: Forensic Science **6**(1), 1507 (2024)

[104] Ferrag, M.A., Alwahedi, F., Battah, A., Cherif, B., Mechri, A., Tihanyi, N.: Generative ai and large language models for cyber security: All insights you need. Available at SSRN 4853709 (2024)

[105] Kucharavy, A., Schillaci, Z., Maréchal, L., Würsch, M., Dolamic, L., Sabonnadiere, R., David, D.P., Mermoud, A., Lenders, V.: Fundamentals of generative large language models and perspectives in cyber-defense. arXiv preprint arXiv:2303.12132 (2023)

[106] Zhao, W.X., Zhou, K., Li, J., Tang, T., Wang, X., Hou, Y., Min, Y., Zhang, B., Zhang, J., Dong, Z., et al.: A survey of large language models. arXiv preprint arXiv:2303.18223 **1**(2) (2023)

[107] Chang, Y., Wang, X., Wang, J., Wu, Y., Yang, L., Zhu, K., Chen, H., Yi, X., Wang, C., Wang, Y., *et al.*: A survey on evaluation of large language models. ACM transactions on intelligent systems and technology **15**(3), 1–45 (2024)

[108] Naveed, H., Khan, A.U., Qiu, S., Saqib, M., Anwar, S., Usman, M., Akhtar, N., Barnes, N., Mian, A.: A comprehensive overview of large language models. arXiv preprint arXiv:2307.06435 (2023)

[109] Shanahan, M.: Talking about large language models. Communications of the ACM **67**(2), 68–79 (2024)

[110] Zhao, H., Chen, H., Yang, F., Liu, N., Deng, H., Cai, H., Wang, S., Yin, D., Du, M.: Explainability for large language models: A survey. ACM Transactions on Intelligent Systems and Technology **15**(2), 1–38 (2024)

[111] Chen, M., Tworek, J., Jun, H., Yuan, Q., Pinto, H.P.D.O., Kaplan, J., Edwards, H., Burda, Y., Joseph, N., Brockman, G., et al.: Evaluating large language models trained on code. arXiv preprint arXiv:2107.03374 (2021)

[112] Zhou, H., Xu, W., Dehlinger, J., Chakraborty, S., Deng, L.: An llm-driven approach to gain cybercrime insights with evidence networks

[113] Chernyshev, M., Baig, Z., Doss, R.R.M.: Towards large language model (llm) forensics using llm-based invocation log analysis. In: Proceedings of the 1st ACM Workshop on Large AI Systems and Models with Privacy and Safety Analysis, pp. 89–96 (2023)

[114] Walker, C., Gharaibeh, T., Alsmadi, R., Hall, C., Baggili, I.: Forensic analysis of artifacts from microsoft's multi-agent llm platform autogen. In: Proceedings of the 19th International Conference on Availability, Reliability and Security, pp. 1–9 (2024)

[115] Sharma, B., Ghawaly, J., McCleary, K., Webb, A.M., Baggili, I.: Forensicllm: A local large language model for digital forensics. Forensic Science International: Digital Investigation **52**, 301872 (2025)

[116] Saxena, N.A., Zhang, W., Shahabi, C.: Missed opportunities in fair ai. In: Proceedings of the 2023 SIAM International Conference on Data Mining (SDM), pp. 961–964 (2023). SIAM

Learning Scene Context for Anomaly Detection: A Benchmarking Study of Forward Frame Prediction CNN and Vision Language Model-Based Approach

Preet Kanwal[1], Shylaja S S[1], Niroop Karera[1*†], Nischal Kashyap[1*†], Poorvi Tambakad[1*†], Rishab Gongulur[1*†], Prasad B Honnavalli[2]

[1*]Department of Computer Science and Engineering, PES University, Bangalore, 560085, Karnataka, India.
[2]Member, IEEE.

*Corresponding author(s). E-mail(s): niroopkarera@gmail.com; nischal826@gmail.com; tambakadpoorvi@gmail.com; rishabgongulur3@gmail.com;
Contributing authors: preetkanwal@pes.edu; shylaja.sharath@pes.edu; prasadhb@pes.edu;
†These authors contributed equally to this work.

Abstract

Video anomaly detection (VAD)[1] is essential in intelligent surveillance, helping identify unusual events in video sequences to improve safety and security. This paper presents a comparative study between a Vision Language Model-based approach utilizing Vision Transformers (ViT) integrated with the CLIP model, and a deep learning[11] approach employing object tracking and Context Conditioned Variational Autoencoder (CVAE) for VAD on the NWPU campus dataset. Our Vision Language Model-based approach uses CLIP's feature association[3] capabilities and a custom temporal annotation file for weakly supervised anomaly detection, enhancing detection accuracy[40] through feature alignment between visual and textual embeddings. Conversely, the deep learning approach integrates ByteTrack for tracking, and CVAE to incorporate contextual scene information[36], improving scene-dependent anomaly detection by distinguishing contextually abnormal events. Both models are evaluated on frame-level AUC and mAP metrics under various IoU thresholds, enabling a robust comparison.

Experimental results demonstrate that each approach has distinct advantages depending on the anomaly's context and complexity, underscoring the importance of architecture choice[37] [38] [39] in video anomaly detection applications. The deep learning model, which learned patterns without any text labels (unsupervised), outperformed the vision-language model, achieving an AUC of 67.5% compared to 60.05%. Even with the additional textual supervision in the vision-language model, the deep learning approach proved more effective in identifying anomalies. These findings highlight the potential for unsupervised[4] [5] [6]learning for this task.

1 Introduction

Video anomaly detection (VAD) is a rapidly growing field within computer vision, driven by the increasing demand for real-time surveillance in public safety, industrial monitoring, and autonomous systems.[7] [41] The primary goal of VAD is to identify unusual events within video sequences, such as unexpected human behavior or vehicle movements, which could indicate potentially dangerous situations. Despite significant advances in this field, detecting anomalies remains challenging due to the diverse nature of anomalous events, which may vary significantly across different scenes and contexts.[42] [43] To address these challenges, various machine learning approaches have been explored, including semi-supervised, unsupervised, and weakly supervised methods.

Among these methods, two broad categories have emerged as prominent approaches for VAD: Transformer-based models and deep learning frameworks with specific feature extraction and tracking modules. Transformer-based models, particularly those that integrate Vision Transformers (ViT) [8] and language-vision models such as CLIP [9], leverage global attention mechanisms [10] to capture spatial and temporal relationships across video frames. By mapping textual descriptions to corresponding visual features, these models offer a novel approach to weakly supervised anomaly detection, where only minimal labeling is available, and label-alignment techniques are utilized to link frames to anomaly labels. This study employs a ViT-based approach with CLIP to handle feature extraction and label alignment, enabling the model to detect unusual patterns within video frames based on both visual and semantic information. This architecture is particularly suited to tasks that require robust feature association, as CLIP's integration of text and visual features allows the model to discern anomalies based on textual cues, potentially leading to enhanced detection capabilities across varied scenes.

In contrast, traditional deep learning methods[11] have evolved to address anomaly detection through various means, such as tracking objects, encoding scene context, and generating predictive models for future frames. These approaches often rely on Convolutional Neural Networks (CNNs)[12] [13] [40] and autoencoders[14], such as Context Conditioned Variational Autoencoders (CVAEs)[15], to capture the underlying structure of normal video sequences and highlight deviations. Our deep learning approach in this study utilizes ByteTrack[16], an advanced multi-object tracking model, integrated with MMTracking[17] for structured data flow and object tracking within each frame sequence. ByteTrack's robust handling of both high- and low-confidence detections helps manage partially occluded or noisy scenes, enhancing the model's ability to track object movements and detect deviations that might indicate abnormal events. Additionally, the inclusion of a CVAE allows for encoding background context, enabling the model to differentiate between contextually normal and abnormal events, a critical capability in scenes where the same behavior might be considered normal in one context but anomalous in another.

The NWPU Dataset[15] provides a compelling benchmark for testing these approaches, as it contains videos across various campus settings, representing diverse scenarios and types of anomalies. This dataset, with its detailed annotations and inclusion of scene-dependent anomalies, is well-suited for evaluating models that require nuanced understanding of contextual cues. In this study, we explore how each method — Transformer-based and deep learning-based — performs on this dataset. We focus on distinguishing "Normal" versus "Anomaly" labels, following a weakly supervised approach, where temporal annotations of anomalies provide start and end frames for each anomalous event. Such weak supervision is practical and cost-effective, reducing the labeling burden while still providing valuable information for model training. Our Transformer-based approach leverages Vision Transformer (ViT-B/16) architecture, which divides each video frame into 10 distinct segments, allowing the model to capture diverse features across a single frame. These segments are processed by CLIP, which internally associates the visual features with corresponding text-based embeddings, facilitating frame-level anomaly detection. This combination of ViT-B/16 and CLIP enables the model to effectively classify frames as "Normal" or "Anomaly" by identifying discrepancies in feature alignments. Moreover, the use of a tailored train-test split in the dataset, accounting for both the anomaly distribution and scene background, further strengthens the model's capacity to generalize across different environments.

On the other hand, the deep learning approach focuses on predictive modeling techniques. It includes a frame prediction model that utilizes the scene context for forecasting future frames, helping to distinguish normal patterns from anomalous ones. By encoding each frame's background information as contextual features and inputting them into a CVAE, this method enables the model to detect deviations based on scene-specific cues. The frame prediction model predicts future frames by taking sequences of observed frames as input, using U-Net-like skip connections [18] for preserving spatial details, which are crucial for accurate anomaly detection. This approach is particularly advantageous for anomaly detection scenarios that require scene-dependent analysis, as the CVAE aids in distinguishing anomalies based on scene context.

Both approaches use complementary evaluation metrics to assess performance on the NWPU dataset. Frame-level Area Under the Curve (AUC) and mean Average Precision (mAP) across different Intersection over Union (IoU) thresholds are utilized to capture the model's ability to accurately identify anomalous frames. The AUC metric serves to measure the overall classification ability of each model on a frame-by-frame basis, whereas mAP at varying IoU thresholds evaluates how well the models can detect anomalies within temporally aligned sequences [19], providing insights into temporal consistency in anomaly detection.

This research paper aims to make three primary contributions. First, it offers a systematic comparison of Transformer-based and deep learning-based approaches to video anomaly detection, using a unified framework and evaluation metrics to establish a fair comparison. Second, it provides insights into the strengths and limitations of each approach in different video contexts, with a specific focus on scene-dependent anomalies. Third, it introduces a structured experimental protocol for VAD on the NWPU dataset, including a tailored train-test split that balances scene diversity, making it a valuable reference for future studies in the field.

In summary, this paper addresses a significant gap in the field of VAD by comparing Transformer-based and deep learning-based methodologies on a complex, real-world dataset. Through detailed analysis and experimental evaluation, we demonstrate how each approach contributes to improving anomaly detection performance in campus settings. Our findings will inform the future development of VAD models and highlight the importance of selecting appropriate architectures based on the nature of the anomalies and the context in which they occur.

2 Related Work

2.1 Vision Language Model based Approach

The introduction of Transformers in computer vision, specifically Vision Transformers (ViT), has spurred interest in their application to VAD due to their capability to capture long-range dependencies and context across video frames. Vision Transformers have proven effective in learning robust spatial and temporal representations, which are essential for identifying subtle anomalies in complex scenes. The CLIP model, which integrates visual and textual feature extraction, extends the utility of ViTs by aligning visual frames with textual embeddings, providing a unique advantage in weakly supervised VAD. Models utilizing CLIP have shown promise in associating video frames with anomaly labels based on semantic content, even with limited supervision, which is particularly valuable in large-scale datasets where exhaustive labeling is impractical.

More recently, Video Vision Transformer (ViViT) has been modified to enhance video anomaly detection by capturing richer temporal information and global contexts, making it effective for video prediction. By integrating ViViT with the U-Net architecture, the model can predict future frames while maintaining a strong focus on both local and global features. TransAnomaly[20], a prediction-based method, leverages the strengths of transformers in handling sequence data by predicting future frames and using the discrepancies between predicted and actual frames for anomaly detection. This method calculates regularity scores using sliding windows to evaluate the impact of different window sizes and strides, identifying frames with lower scores as potential anomalies. Moreover, it localizes anomalies by tracking patches with lower regularity scores, significantly improving the detection and localization process.

Another advanced framework, TEVAD[21], integrates both visual and textual modalities for video anomaly detection. Visual features are extracted using pre-trained I3D models[22] or SwinBERT[23] for video captioning, while text features are processed using SimCSE[24], a contrastive learning-based[25] framework for generating sentence embeddings. The multi-branch architecture processes visual and textual data in parallel, with a Multi-Scale Temporal Network (MTN)[26] capturing both short- and long-range temporal dependencies in the features. These features are fused using methods like concatenation, addition, and Hadamard product, followed by classification tasks for snippet-level anomaly detection that propagate to frame-level predictions. TEVAD's reliance on feature fusion, transformer-based components like SwinBERT, and classification tasks ensures a comprehensive approach to anomaly detection in surveillance videos.

2.2 Deep Learning based Approach

Traditional deep learning-based VAD models typically rely on convolutional neural networks (CNNs) and autoencoders to learn representations of normal activities, with the objective of detecting deviations indicative of anomalies. Prediction-based models, such as frame prediction and reconstruction models, have been widely adopted due to their ability to highlight discrepancies between observed and expected frames, which signal potential anomalies. The use of Context Conditioned Variational Autoencoders (CVAEs) in recent studies has been particularly noteworthy, as CVAEs allow the model to incorporate scene-specific context into its predictions, improving the detection of scene-dependent anomalies. Additionally, object detection and tracking models, such as ByteTrack and MMTracking, have been integrated with VAD systems to analyze the movement and behavior of objects, enhancing the detection of anomalies that involve abnormal object interactions or trajectories.

The Generative Adversarial Network has also been proposed as an effective unsupervised method for VAD. Its framework includes a Self-Attentive Predictor, which captures long-term dependencies in video frames to improve prediction quality, and a Vanilla Discriminator [28] that performs true-false discrimination to identify anomalies. Furthermore, a Self-Supervised Discriminator aids in encoding semantic information into predicted normal frames through rotation detection and adversarial training. The model leverages adversarial training to enhance its ability to distinguish between normal and abnormal frames, resulting in greater detection errors for anomalies. These components collectively demonstrate the effectiveness of SSAGAN[28] in improving video anomaly detection performance.

Another innovative approach introduces attribute-based representations, where each object in a video is represented by its velocity and pose. This method improves both accuracy and interpretability in VAD systems. A density-based anomaly scoring mechanism is employed to identify anomalies by analyzing the distribution of attributes in the video data. Moreover, the authors propose combining interpretable attribute-based representations with implicit, deep representations, achieving state-of-the-art performance across multiple datasets. The approach is evaluated using Area Under the Receiver Operating Characteristic (AUROC) scores.

A novel framework for anomaly detection further addresses the interpretability challenge of existing methods by explaining detected anomalies in surveillance videos. This framework extracts spatiotemporal features to monitor individual objects and their interactions, employing scene graphs to represent object relationships, thereby providing context and uncovering the root causes of anomalies. Additionally, the method supports cross-domain adaptability, leveraging transfer learning across different surveillance environments without requiring extensive labeled data. Theoretical proofs of asymptotic optimality, coupled with empirical validation on benchmark datasets, ensure the reliability and effectiveness of this approach. This combination of methods establishes a more interpretable and adaptable VAD system, setting it apart from traditional techniques.

Another model provides an effective approach for VAD by encoding relationships between objects' states over long sequences, capturing consistent motion patterns. This model represents temporal dependencies by predicting future states based on past observations, enabling a comprehensive understanding of motion patterns over time. Anomalies are detected by measuring the divergence between predicted future states (based on learned normal patterns) and actual observed states, with significant divergences indicating anomalies. Extensive experiments demonstrate the effectiveness of the model, showing improvements over state-of-the-art techniques in video anomaly detection.

Another unsupervised approach for detecting human-related anomalies in complex scenes, leveraging skeleton-based analysis[32]. The method decomposes skeleton features into global (whole-body movement) and local (individual joint movement) components to capture detailed motion dynamics. A self-training regression[33] framework iteratively updates anomaly scores, refining pseudo-labeled anomalous and normal skeleton sets over multiple iterations. The core of the approach is an anomaly scoring module that integrates a graph convolutional network (GCN)[26] with fully connected layers to model spatial relationships within the skeleton data. This iterative refinement process enhances detection performance by progressively improving the model's ability to identify deviations from normal patterns, all without requiring labeled normal videos.

A more recent advancement incorporates Video-Based Large Language Models (VLLMs)[27] for VAD, which operates without the need for complex thresholding, and provides textual explanations for detected anomalies, enhancing the interpretability of the results. A key component of this approach is the Long-Term Context (LTC) module which is used to capture long-range context. The authors propose a three-phase training method, which improves the efficiency of fine-tuning VLLMs by significantly reducing the amount of VAD data required and lowering costs associated with annotating instruction-tuning data. The model has been evaluated on the UCF-Crime[34] and TAD benchmarks, where it achieved notable improvements in Area Under the Curve (AUC) scores.

3 Methodology

This section details the methodology employed in our study, covering the problem definition, architecture, the functioning of each approach, and implementation details. The two approaches — Vision Language model based with CLIP and a deep learning-based approach utilizing multi-object tracking and Context Conditioned Variational Autoencoder (CVAE) — are evaluated for their effectiveness in detecting anomalies in the NWPU dataset, specifically focusing on scene-dependent anomalies under a weakly supervised setting.

3.1 Problem Definition

The primary goal of this research is to identify anomalies in video sequences recorded across various campus settings in the NWPU dataset. Anomalies are defined as events or behaviors that deviate from typical patterns, varying based on context and scene. We formulate this task as a weakly supervised video anomaly detection (VAD) problem, where minimal labeling is available. Specifically, each video is labeled as either "Normal" or "Anomaly," and temporal annotations specify the start and end frames for anomalous events. This weakly supervised approach leverages temporal annotations without requiring frame-by-frame labels, making it cost-effective while still providing essential information for model training.

3.2 Architecture

3.2.1 Vision Language Model-Based Architecture

Built upon Vision Transformers (ViT-B/16) with CLIP for aligning visual and textual embeddings, capturing spatial dependencies within frames.

3.2.2 Deep Learning-Based Architecture

This architecture integrates ByteTrack with MMTracking for object tracking and uses a CVAE for encoding scene-specific context, which aids in differentiating normal and abnormal activities based on the surrounding environment.

3.3 Methodology Functioning

The methodology is detailed in sequential stages for each architecture, describing each key component's function and contribution to the anomaly detection process.

3.3.1 Transformer-Based Approach

To prepare the data, video frames are sampled at intervals of 16 frames, ensuring broad coverage of each video sequence. A ten-crop augmentation technique is applied to each frame, creating ten distinct crops: a central crop, four quadrant crops, and their flipped counterparts. This augmentation process allows the model to learn from diverse perspectives, enhancing its ability to detect anomalies by increasing the range of visual information.

The augmented frames are processed by the Vision Transformer (ViT-B/16), which extracts spatial features from each cropped segment. CLIP then associates these visual features with textual embeddings corresponding to the labels "Normal" and "Anomaly." This dual-embedding structure of CLIP enables alignment between the frame's visual characteristics and semantic context, allowing the model to detect discrepancies indicative of anomalies. This approach is particularly suited for weakly supervised settings, as it utilizes CLIP's ability to align frames with their correct anomaly labels.

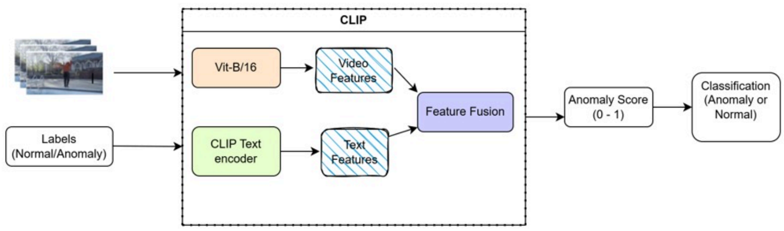


Fig. 1: System Architecture Diagram for Vision Language Model-Based Approach

The Transformer-based model operates through 2 branches for anomaly detection: Branch 1: A simple binary classifier predicts an anomaly confidence score based on the extracted visual features. This classifier is composed of a feed-forward neural network with a Sigmoid activation function to output a probability score.

Branch 2: This branch leverages CLIP's textual embeddings to compute a similarity-based alignment map, comparing visual features to text embeddings. This map identifies frames that align most closely with either "Normal" or "Anomaly," providing fine-grained detection capabilities.

Three loss functions guide the training of the Transformer-based model:
Binary Cross-Entropy Loss (Loss 1): Applied to Branch 1's anomaly confidence scores for accurate anomaly classification.

$$L_{bce} = -\frac{1}{N}\sum_{i=1}^{N}[y_i \log(\hat{y}_i) + (1 - y_i)\log(1 - \hat{y}_i)] \tag{1}$$

MIL-Align Loss (Loss 2): Encourages alignment between visual and textual embeddings, supporting frame-to-label matching.
The class probability for ith class is defined as (p_i):

$$p_i = \frac{\exp(s_i/\tau)}{\sum_{j=1}^{m}\exp(s_j/\tau)} \tag{2}$$

The alignment loss is defined as:

$$L_{nce} = -\frac{1}{N}\sum_{i=1}^{N}\log(p_{c_i}) \tag{3}$$

where
τ refers to the temperature hyper-parameter for scaling.
s_i represents the similarity between this video and the i-th class.

Contrastive Loss (Loss 3): Ensures distinct separation between normal and anomaly embeddings, enhancing model robustness. The contrastive loss is defined as:

$$L_{cts} = \sum_{j}\max\left(0, \frac{t_n^\top t_{aj}}{\|t_n\|_2 \cdot \|t_{aj}\|_2}\right) \tag{4}$$

where
t_n is the normal class embedding.
t_{aj} is the j-th abnormal class embedding.

The total objective function combines all three losses:

$$L = L_{bce} + L_{nce} + \lambda L_{cts} \tag{5}$$

3.3.2 Deep Learning-Based Approach

The deep learning model incorporates ByteTrack, applied to each frame sequence to detect and track objects across frames. ByteTrack enables accurate tracking by managing high- and low-confidence detections, accommodating challenges such as occlusion and scene noise. This component is essential for extracting object trajectories, which provide valuable insights into movement patterns associated with anomalies, such as abrupt directional changes or velocity variations.

The CVAE module encodes the scene's background context, generating scene-specific feature maps that inform the model about the normalcy of specific activities based on the environment. By embedding scene context, the model can differentiate between contextually normal and abnormal events. For instance, cycling on a pedestrian walkway is treated as anomalous in this context due to the model's understanding of campus norms. This module enhances the model's capacity to detect scene-dependent anomalies by learning scene-specific expectations.

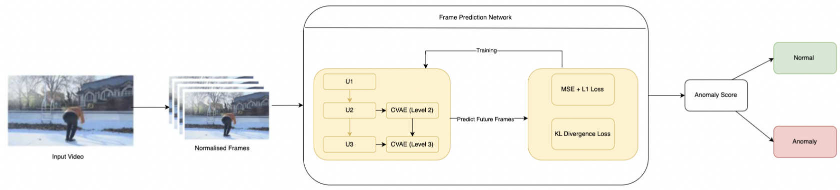


Fig. 2: System Architecture Diagram for Deep Learning Approach

Following context encoding, the frame prediction model takes observed frames as input and generates future frames, enabling anomaly detection based on discrepancies between observed and predicted frames. This model uses a 3 Layer U-Net architecture with skip connections that retain spatial detail, which is essential for accurate anomaly detection.

The feature maps at U2 and U3 levels are fed into the CVAEs to generate new feature maps conditioned on the scene. These new feature maps are added to the input of CVAEs with a weight $\gamma = 1$ through which the full contribution of the scene-conditioned feature maps is applied, enabling the model to better detect scene-dependent anomalies by factoring in the background context of each frame. The predicted frames are then generated through subsequent decoding convolutional layers. Any inconsistencies between predicted and actual frames are quantified as prediction errors, with higher error values signaling potential anomalies.

Two primary loss functions are applied in the deep learning-based approach:

Frame Prediction Loss: Calculated as a combination of Mean Squared Error (MSE) and L1 Loss, minimizing discrepancies in frame predictions. L1 loss, or Mean Absolute Error (MAE), is the average of the absolute differences between predicted values and actual values.

The L1 loss is defined as:

$$L_{L1} = \sum_{i=1}^{n} |y_i - \hat{y}_i| \tag{6}$$

where
n: The total number of data points in the dataset.
y_i: The true value corresponding to the i-th data point.
$\hat{y}_i$: The predicted value corresponding to the i-th data point.

This has been used within the frame prediction network to compute an error between a predicted frame and the ground truth frame.

Mean Squared Error(MSE) measures the average squared differences between predicted values and actual values. MSE is defined as:

$$L_{\text{MSE}} = \frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 \tag{7}$$

where
n: The total number of data points in the dataset.
y_i: The actual or true value for the i-th data point.
$\hat{y}_i$: The predicted value for the i-th data point.

The total frame prediction loss is given as:

$$L_{\text{total}} = \lambda_{L1} L_{L1} + L_{\text{MSE}} \tag{8}$$

where λ_{L1} is the weight of L1 loss.

The Kullback-Leibler (KL) Divergence Loss: Applied to the CVAE for maintaining a smooth latent space over scene-specific features, ensuring effective background encoding. The KL divergence loss is given by:

$$L_{KL}(\mathcal{N}(\hat{\mu}, \hat{\sigma}^2) \| \mathcal{N}(0,1)) = -\frac{1}{2} \left(\log \hat{\sigma}^2 - \hat{\mu}^2 - \hat{\sigma}^2 + 1 \right), \tag{9}$$

It computes how much the learned distribution $\mathcal{N}(\hat{\mu}, \hat{\sigma}^2)$ deviates from the standard normal distribution $\mathcal{N}(0,1)$.

The distribution $\mathcal{N}(\hat{\mu}, \hat{\sigma}^2)$ is the learned Gaussian distribution with mean $\hat{\mu}$ and variance $\hat{\sigma}^2$, while $\mathcal{N}(0,1)$ is the standard normal distribution with mean 0 and variance 1.

The formula includes the following components: $\log \hat{\sigma}^2$, which penalizes the deviation in variance; $\hat{\mu}^2$, which penalizes the deviation in mean; and $\hat{\sigma}^2$, which penalizes the variance magnitude relative to 1. The constant term $+1$ is used to ensure proper normalization of the divergence calculation.

The total loss is a combination of forward prediction error and KL divergence loss with the weight of λ_{KL}. This loss is minimized to train the model.

$$L = L_{\text{total}} + \lambda_{KL} L_{KL} \tag{10}$$

The architecture diagram provides a visual representation of the two approaches, detailing the data flow through pre-processing, feature extraction, context encoding, and anomaly detection stages. The diagram highlights the dual-branch structure in the Transformer-based model and the integration of object tracking and CVAE in the deep learning model, helping illustrate each method's data processing path and decision points.

3.3.3 Implementation Details

The models were implemented using PyTorch, with OpenCV handling frame extraction and pre-processing. The Transformer-based model relies on the pre-trained CLIP model, fine-tuned on the NWPU dataset with a batch size of 16, using the ViT-B/16 architecture. This model's binary classifier in Branch 1 was optimized with a learning rate of 0.001 on an NVIDIA RTX 4070 GPU. The deep learning model leverages ByteTrack, integrated with MMTracking for seamless object tracking in each frame sequence. The CVAE module, designed to capture background context, was trained with a learning rate of 0.001 using the Adam optimizer on an NVIDIA RTX 4070 GPU. ByteTrack and CVAE were structured to ensure effective handling of scene-dependent anomalies, taking advantage of the NVIDIA GPU's parallel processing capabilities to expedite training and testing phases. These implementations ensure a consistent and reliable comparison between the two architectures, with uniform pre-processing protocols and evaluation metrics to enable replicable results on the NWPU dataset.

4 Experimentation

4.1 Dataset

Several datasets have been established to evaluate video anomaly detection (VAD) models, typically adopting either single-scene or multi-scene setups. Notable examples include UCSD Ped1[29] and Ped2, which focus on pedestrian walkways, and ShanghaiTech, a multi-scene dataset featuring generalized anomalies. However, these datasets lack a critical component: scene-dependent anomalies—events that may be normal in one context but abnormal in another. To address this gap, the NWPU dataset, used in this study, introduces scene-dependent anomalies across diverse campus scenes, representing a broader range of real-world anomaly types.

NWPU Dataset Overview and Annotations The NWPU dataset comprises 547 videos, with 423 labeled as "Normal" and 124 as "Anomaly", collected from various campus settings. These videos capture a wide range of activities and environmental contexts, enhancing scene diversity and allowing models to tackle challenges unique to scene-dependent anomalies. Each video is annotated using a structured temporal annotation file containing fields such as:

video_name: Unique identifier for each video file.
anomaly_label: Indicates whether the video is "Normal" or "Anomaly."
first_anomaly_start_frame and first-anomaly_end_frame: Start and end frames for the first anomaly, if present.
second_anomaly_start_frame and second_anomaly_end_frame: Start and end frames for a second anomaly, if applicable.
For example:
_001: Anomaly, 20, 130, -1, -1 (One anomaly event between frames 20 and 130).
D_003: Normal, -1, -1, -1, -1 (No anomalies present).
D_007: Anomaly, 230, 1306, 1500, 1580 (Two anomalies between frames 230–1306 and 1500–1580).

This annotation format supports weakly supervised learning, minimizing reliance on exhaustive frame-level labels, which are often impractical in real-world applications due to cost and time constraints.

To ensure effective training across diverse scenes, a new 80-20 train-test split was created, maintaining a proportional representation of normal and anomalous events. The training set includes 437 videos (337 "Normal" and 100 "Anomaly"), while the test set contains 110 videos (86 "Normal" and 24 "Anomaly"). Unlike traditional setups that train solely on normal videos, this tailored split allows models to learn from both normal and anomalous patterns, enhancing generalization and detection accuracy.

While datasets like UCF Crime, XD Violence[35], and ShanghaiTech are widely used, they exhibit limitations in addressing scene-dependent anomalies:

UCF Crime: Covers 13 crime-related events such as robbery and assault but lacks scene diversity and context-dependent anomaly scenarios.
XD Violence: Focuses on high-action anomalies (e.g., explosions, abuse) without considering scene-specific contexts, making it less suitable for subtle context-aware anomaly detection.
ShanghaiTech: Provides generalized anomalies across multi-scenes but lacks the fine-grained scene diversity required for detecting context-specific behaviors.

The NWPU dataset surpasses these limitations by including 43 diverse scenes and 28 classes of anomalies, such as unauthorized gatherings or vehicles in restricted zones. This diversity fosters robust generalization and serves as an ideal benchmark for evaluating VAD models tailored to scene-dependent anomaly detection.

This study leverages the NWPU dataset to evaluate both vision language model-based and deep learning approaches for VAD. By addressing the challenges of scene-dependent anomalies, it bridges the gap in existing benchmarks and provides valuable insights into context-aware anomaly detection. Models utilizing the NWPU dataset benefit from its detailed annotations, diverse scenes, and balanced representation of normal and anomalous events, making it a critical resource for advancing VAD research.

4.2 Evaluation Metrics

To assess the performance of the models, we employ standard evaluation metrics in VAD, specifically mean Average Precision (mAP) and Area Under the Curve (AUC), with a particular focus on frame-level AUC and mAP across varying Intersection over Union (IoU) thresholds. These metrics are well-suited for capturing the accuracy of models in detecting anomalies within specific frames and across temporal segments in videos.

Frame-Level AUC: The frame-level AUC metric evaluates the model's ability to accurately classify individual frames as either "Normal" or "Anomaly." AUC provides a measure of the true positive rate against the false positive rate at various threshold levels. Higher AUC values indicate that the model has a strong capacity for distinguishing between normal and anomalous frames, which is critical for effective VAD performance. This metric is especially useful in scenarios where detecting isolated anomalous frames is essential.

Mean Average Precision (mAP) at IoU Thresholds: The mAP metric, evaluated at various IoU thresholds, measures the model's ability to accurately localize and classify anomalies within temporal sequences. For VAD, this metric involves calculating the intersection of the predicted anomalous segments with the ground truth segments, with mAP scores calculated at IoU thresholds from 0.1 to 0.5, increasing in increments of 0.1. The average of these mAP scores across thresholds provides an overall indication of the model's accuracy in identifying anomalous segments within videos.

For instance, an IoU threshold of 0.5 requires that the model's predicted anomaly segment overlaps by at least 50% with the ground truth anomaly segment to be considered a true positive. Lower IoU thresholds, such as 0.1, are less stringent, allowing for a more relaxed overlap criterion. Evaluating mAP at multiple IoU levels provides a comprehensive view of the model's localization accuracy under varying degrees of overlap precision, enabling performance assessments under both strict and lenient criteria.

Mean mAP (AVG): The average mAP (AVG) score, calculated across all IoU thresholds, provides a single summary metric for model performance on the NWPU dataset. This value is particularly useful for comparing the detection effectiveness of different models and gauging their robustness across a range of IoU constraints.

In summary, the NWPU dataset, with its diverse campus scenes and scene-dependent anomalies, offers a more comprehensive testbed for VAD compared to traditional datasets. By using a combination of frame-level AUC and mAP across IoU thresholds, our evaluation framework provides a thorough assessment of model performance in detecting anomalies within complex video sequences, facilitating a nuanced comparison between the Vision Language Model-based and deep learning-based approaches.

5 Results

The quantitative analysis focuses on two main metrics: frame-level AUC and mean Average Precision (mAP) across different Intersection over Union (IoU) thresholds. These metrics allow us to assess the accuracy of each model in classifying anomalous frames and correctly localizing anomaly segments within videos.
Frame-Level AUC The frame-level AUC values for both models demonstrate their ability to classify individual frames as either "Normal" or "Anomaly." The learning-based approach achieved an average frame-level AUC of 67.5%, outperforming the Video Vision Language model, which obtained an AUC of 60.05%. This performance discrepancy suggests that the deep learning approach outperforms the vision language model, and highlights the potential of scene-aware models in enhancing video anomaly detection, providing better precision in complex scenarios by adapting to specific scene contexts.

Mean Average Precision (mAP) at IoU Thresholds Both models were evaluated using mAP across IoU thresholds ranging from 0.1 to 0.5 (in increments of 0.1), as this metric captures the model's ability to accurately localize anomalous segments over temporal frames.

Serial Number	1	2	3	4	5
mAP@IoU (%)	0.1	0.2	0.3	0.4	0.5
Vision Language Model-Based	4.31	3.6	3.5	3.36	3.20

Table 1: Comparison of mAP scores across various IoU thresholds for Vision Language Model-Based Approach

Serial Number	1	2	3	4	5
Patch Size	256	128	64	32	16
AUC (%)	67.56	65.97	66.49	65.83	65.98

Table 2: Patch Size vs. AUC for Deep Learning Approach

6 Conclusion and Future Work

This study presents a comprehensive comparison of Vision Language Model-based and deep learning-based approaches for video anomaly detection on the NWPU dataset, emphasizing their performance in detecting scene-dependent anomalies. The Vision Language Model-based approach, utilizing Vision Transformers (ViT) with CLIP, demonstrated superior accuracy in nuanced contextual scenarios, while the deep learning-based model excelled in real-time applications focused on movement-based anomalies. By evaluating both models on frame-level AUC and mAP across IoU thresholds, we highlight the strengths and limitations of each approach, ultimately guiding practitioners in selecting the most suitable architecture based on specific deployment needs in complex video surveillance environments.

Future efforts will aim to improve the Vision Transformer-based approach by incorporating advanced temporal modeling techniques and optimizing the embedding alignment process. Self-supervised learning strategies will be explored to reduce reliance on temporal annotations, improving efficiency. Expanding the study to diverse datasets will assess generalizability, while adding fine-grained anomaly classification can extend the framework's application scope. Finally, optimizing computational efficiency will facilitate practical deployment in real-world anomaly detection systems, ensuring both scalability and accuracy.

Approach	**AUC (%)**	**Comments**
Deep Learning Approach	67.5	No text labels used (unsupervised)
Vision-Language Model	60.05	Text labels used (semisupervised)

Table 3: Comparison of Deep Learning and Vision-Language Models

7 Acknowledgements

We would like to express our sincere gratitude to the U.S. Army for their generous funding and unwavering support, which made the AI-Enabled National Conference on Forensic Investigations Network in Digital Sciences(FINDS) possible. Their commitment to advancing research and fostering collaboration is deeply appreciated.

We are profoundly grateful to Dr. S. S. Iyengar for his outstanding contributions to the field, which have been a constant source of inspiration to us. His pioneering work has shaped much of our research, and we are deeply thankful for the impact his research has had on our academic journey. His dedication and innovations continue to inspire us in our pursuit of knowledge.

References

[1] S.S. Iyengar, Sina Nabavi, Yashas Hariprasad, Prasad HB, Krishna Mohan C. ”Artificial Intelligence in Practice: Theory and Applications for Cyber Security and Forensics”, Springer Nature (Forthcoming)

[2] Jayakumar Singaram, S. S. Iyengar, Azad M. Madni ”Deep Learning Networks: Design, Development and Deployment”, Springer Nature - ISBN 978-3-031-39243-6

[3] P. Wu, J. Liu, and F. Shen, “A deep One-Class neural network for anomalous event detection in complex scenes,” IEEE Transactions on Neural Networks and Learning Systems, pp. 1–14, Jan. 2019, doi: 10.1109/tnnls.2019.2933554.

[4] R. Yang, Q. Li, Y. Shen, and Z. Zhang, “Anomaly detection in surveillance videos via memory-augmented frame prediction,” 2022 International Joint Conference on Neural Networks (IJCNN), pp. 1–8, Jul. 2022, doi: 10.1109/ijcnn55064.2022.9892924.

[5] X. Wang et al., “Robust unsupervised video anomaly detection by multipath frame prediction,” IEEE Transactions on Neural Networks and Learning Systems, vol. 33, no. 6, pp. 2301–2312, Jun. 2021, doi: 10.1109/tnnls.2021.3083152.

[6] C. Sun, Y. Jia, Y. Hu, and Y. Wu, “Scene-Aware context Reasoning for unsupervised abnormal event detection in videos,” Proceedings of the 30th ACM International Conference on Multimedia, pp. 184–192, Oct. 2020, doi: 10.1145/3394171.3413887.

[7] Cliff Wang, S.S. Iyengar, Kun Sun ”AI Embedded Assurance for Cyber Systems”, Springer Nature - ISBN 978-3-031-42636-0

[8] A. Dosovitskiy et al., “An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale,” International Conference on Learning Representations, May 2021, [Online]. Available: https://openreview.net/pdf?id=YicbFdNTTy.

[9] Luo, H.; Ji, L.; Zhong, M.; Chen, Y.; Lei, W.; Duan, N.; and Li, T. 2022. CLIP4Clip: An empirical study of CLIP for end to end video clip retrieval and captioning. Neurocomputing, 508: 293–304.

[10] K. Doshi and Y. Yilmaz, “Towards interpretable video anomaly detection,” 2022 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV), pp. 2654–2663, Jan. 2023, doi: 10.1109/wacv56688.2023.00268.

[11] S. Zhang et al., “Influence-Aware attention networks for anomaly detection in surveillance videos,” IEEE Transactions on Circuits and Systems for Video Technology, vol. 32, no. 8, pp. 5427–5437, Feb. 2022, doi: 10.1109/tcsvt.2022.3148392.

[12] K. O'Shea and R. Nash, "An Introduction to Convolutional Neural Networks," IEEE International Conference, Nov. 2015.

[13] Bin Shi, S. S. Iyengar "Mathematical Theories of Machine Learning – Theory and Applications", Springer - ISBN 978-3-030-17076-9

[14] Y. Zhao, B. Deng, C. Shen, Y. Liu, H. Lu, and X.-S. Hua, "Spatio-Temporal AutoEncoder for video anomaly detection," Proceedings of the 30th ACM International Conference on Multimedia, pp. 1933–1941, Oct. 2017, doi: 10.1145/3123266.3123451.

[15] C. Cao, Y. Lu, P. Wang, and Y. Zhang, "A new comprehensive benchmark for semi-supervised video anomaly detection and anticipation," 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 20392–20401, Jun. 2023, doi: 10.1109/cvpr52729.2023.01953.

[16] Y. Zhang et al., "ByteTrack: multi-object tracking by associating every detection box," in Lecture notes in computer science, 2022, pp. 1–21.

[17] MMTracking Contributors. MMTracking: OpenMMLab video perception toolbox and benchmark. https:// github.com/open-mmlab/mmtracking, 2020.

[18] T. N. Nguyen and J. Meunier, "Anomaly detection in video sequence with Appearance-Motion correspondence," 2021 IEEE/CVF International Conference on Computer Vision (ICCV), pp. 1273–1283, Oct. 2019, doi: 10.1109/iccv.2019.00136.

[19] Y. Liu, Z. Guo, J. Liu, C. Li, and L. Song, "OSIN: Object-Centric Scene Inference Network for Unsupervised Video Anomaly Detection," IEEE Signal Processing Letters, vol. 30, pp. 359–363, Jan. 2023, doi: 10.1109/lsp.2023.3263792.

[20] H. Yuan, Z. Cai, H. Zhou, Y. Wang, and X. Chen, "TransAnomaly: Video anomaly detection using video Vision transformer," IEEE Access, vol. 9, pp. 123977–123986, Jan. 2021, doi: 10.1109/access.2021.3109102.

[21] W. Chen, K. Teck MA, Z. J. Yew, M. Hur, and D. A.-A. Khoo, "TEVAD: Improved video anomaly detection with captions," 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), pp. 5549–5559, Jun. 2023, doi: 10.1109/cvprw59228.2023.00587.

[22] C. Cao, Y. Lu, and Y. Zhang, "Context Recovery and Knowledge Retrieval: A Novel Two-Stream Framework for video Anomaly Detection," IEEE Transactions on Image Processing, vol. 33, pp. 1810–1825, Jan. 2024, doi: 10.1109/tip.2024.3372466.

[23] K. Lin et al., "SwinBERT: End-to-End Transformers with Sparse Attention for Video Captioning," 2022 IEEE/CVF Conference on Computer Vision and Pattern

Recognition (CVPR), Jun. 2022, doi: 10.1109/cvpr52688.2022.01742.

[24] T. Gao, X. Yao, and D. Chen, "SIMCSE: Simple Contrastive Learning of Sentence Embeddings," Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, Jan. 2021, doi: 10.18653/v1/2021.emnlp-main.552.

[25] S. Sun and X. Gong, "Hierarchical semantic contrast for scene-aware video anomaly detection," 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 22846–22856, Jun. 2023, doi: 10.1109/cvpr52729.2023.02188.

[26] Z. Chen, S. Li, B. Yang, Q. Li, and H. Liu, "Multi-Scale Spatial Temporal Graph Convolutional Network for Skeleton-Based Action Recognition," Proceedings of the AAAI Conference on Artificial Intelligence, vol. 35, no. 2, pp. 1113–1122, May 2021, doi: 10.1609/aaai.v35i2.16197.

[27] Y. Zhao, I. Misra, P. Krähenbühl, and R. Girdhar, "Learning Video Representations from Large Language Models," 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Jun. 2023, doi: 10.1109/cvpr52729.2023.00637.

[28] C. Huang et al., "Self-Supervised Attentive Generative adversarial networks for video anomaly detection," IEEE Transactions on Neural Networks and Learning Systems, vol. 34, no. 11, pp. 9389–9403, Apr. 2022, doi: 10.1109/tnnls.2022.3159538.

[29] V. Mahadevan, W. Li, V. Bhalodia, and N. Vasconcelos, "Anomaly detection in crowded scenes," CVPR, Pages 1975–1981, 2010, Jun. 2010, doi: 10.1109/cvpr.2010.5539872.

[30] C. Lu, J. Shi, and J. Jia, "Abnormal Event Detection at 150 FPS in MATLAB," Proceedings of the IEEE International Conference on Computer Vision. 2013: 2720-2727 DOI: 10.1109/ICCV.2013.338, pp. 2720–2727, Dec. 2013, doi: 10.1109/iccv.2013.338.

[31] W. Liu, W. Luo, D. Lian, and S. Gao, "Future Frame Prediction for Anomaly Detection - A New Baseline," In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Pages 6536–6545, 2018, pp. 6536–6545, Jun. 2018, doi: 10.1109/cvpr.2018.00684.

[32] R. Rodrigues, N. Bhargava, R. Velmurugan, and S. Chaudhuri, "Multi-timescale Trajectory Prediction for Abnormal Human Activity Detection," IEEE/CVF Winter Conference on Applications of Computer Vision, Pages 2615–2623, 2020. 2, 3, 7, pp. 2615–2623, Mar. 2020, doi: 10.1109/wacv45572.2020.9093633.

[33] G. Pang, C. Yan, C. Shen, A. Van Den Hengel, and X. Bai, "Self-Trained Deep Ordinal regression for End-to-End video anomaly detection," 2022

IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 12170–12179, Jun. 2020, doi: 10.1109/cvpr42600.2020.01219.

[34] W. Sultani, C. Chen, and M. Shah, “Real-World Anomaly Detection in Surveillance Videos,” Proceedings of the IEEE Conference on Computer Vision and Pattern Recog- Nition, Pages 6479–6488, 2018, Jun. 2018, doi: 10.1109/cvpr.2018.00678.

[35] P. Wu et al., “Not only Look, But Also Listen: Learning Multimodal Violence Detection Under Weak Supervision,” in Lecture notes in computer science, 2020, pp. 322–339.

[36] Gregory Vert, S.S. Iyengar, Vir Phoha ”Introduction to Contextual Processing – Theory and Application”, CRC Press, pp. 320.

[37] C. Xavier and S.S. Iyengar ”Introduction to Parallel Algorithms (Chinese)”, 263 p., ISBN: 7-111-13390-0

[38] C. Xavier and S.S. Iyengar ”Introduction to Parallel Algorithms”, John Wiley and Sons. Number of Pages : 365

[39] Elliot Soloway and S. S. Iyengar, (ed), ”Empirical Studies of Programmers”, Ablex Pub. Co., Norwood, New Jersey

[40] R.R. Brooks and S.S. Iyengar ”Multi Sensor Fusion: Fundamentals and Applications with Software”, Prentice Hall Publication Co., New Jersey 07458. Number of Pages : 488

[41] Sankita J. Patel, Naveen Kumar Chaudhary, Bhavesh N. Gohil, S. S. Iyengar ”Information Security, Privacy and Digital Forensics – Select Proceedings of the International Conference, ICISPD 2022”, Springer Nature - 978-981-99-5090-4

[42] S. S. Iyengar and Kianoosh G. Boroojeni ”Oblivious Network Routing: Algorithms and Applications”, ISBN: 9780262029155 — 176 pp. — 6 x 9 in, Published by MIT Press

[43] S.S. Iyengar and G.Seetharaman ”The Design and Analysis of Algorithms for processing Digital satellite IR Images”, (In Preparation)

Appendix A Abbreviations

1. NWPU: North Western Polytechnical University
2. VAE: Variational Autoencoder
3. UCSD: University of California, San Diego
4. CUHK: Chinese University of Hong Kong
5. IITB: Indian Institute of Technology Bombay
6. GAN: Generative Adversarial Network
7. VRD: Visual Relationship Detection
8. CNN: Convolutional Neural Network
9. GCN: Graph Convolutional Network
10. TEVAD: Text Empowered Video Anomaly Detection
11. ADAM: Adaptive Moment Estimation
12. STU-Net: Spatiotemporal U-Net
13. I3D: Inflated 3D ConvNet
14. SRS: System Requirements Specification

Appendix B Terminologies

1. Video Anomaly Detection (VAD): Identification of abnormal events within a video dataset.
2. Area Under the Curve (AUC): A metric to evaluate the performance of classification models.
3. Conditional Variational Autoencoder (CVAE): A type of neural network used for unsupervised learning of compressed representations while also taking into account some additional information called the condition.
4. Generative Adversarial Network (GAN): A neural network architecture used to generate new data instances in unsupervised learning.
5. Contrastive Language-Image Pre-training (CLIP): A model that learns visuals from large-scale image-text pairs.
6. Weakly Supervised Video Anomaly Detection (WSVAD): An approach where only weak labels or annotations are provided during training.
7. Intersection over Union (IoU): A measure used in object detection tasks to evaluate the accuracy of bounding box predictions.
8. Mean Average Precision (mAP): A metric to evaluate the accuracy of object detection methods.
9. Spatiotemporal U-Net (STU-Net): A neural network architecture used for spatiotemporal data processing.
10. Inflated 3D ConvNet (I3D): A convolutional neural network architecture used for video classification tasks.

AI-Driven Multimedia Forensics: Enhancing Detection, Provenance Analysis, and Robustness Against Manipulations

Tessy Tom[1], Yashas Hariprasad[2], Pronab Mohanty[3], Antony Puthussery[4]

[1]CSRC, Jain (Deemed to be University), Bangalore, India.
[2]KFSICS, Florida International University, Miami, USA.
[3]Inspector General of Police, Indian Police Service, Government of India.
[4]Department of Computer Science, Christ College, Pune, India.

Contributing authors: tessy.tom@jainuniversity.ac.in; yhari001@fiu.edu; pronab.mohanty@gmail.com; fr.antony@christcollegepune.org;

Abstract

Multimedia forensics has become a crucial field in ensuring the authenticity and integrity of digital content. With the rapid advancement of artificial intelligence (AI) and machine learning (ML), this domain has evolved to leverage sophisticated techniques for analyzing, verifying, and detecting anomalies in multimedia data. This research introduces innovative AI and machine learning (ML)-driven approaches to address key challenges, including deepfake detection, provenance analysis, and forgery identification.

We propose a comprehensive framework that integrates deep neural networks (DNNs), adversarial learning, and graph-based methodologies to improve the detection and classification of manipulated media. Using convolutional neural networks (CNNs) and transformer architectures, our approach effectively identifies pixel-level inconsistencies and temporal artifacts in images and videos with high precision. In addition, we introduce a novel interpretability module to enhance transparency and reliability in forensic decision making.

Our methodology is validated using publicly available datasets such as FaceForensics++, DFDC, and Real-World Anomaly Detection datasets. Preliminary experimental results indicate significant improvements in detection performance, achieving an average precision of 94.6% and a recall of 92.1%, surpassing existing state-of-the-art methods. Also, we explore the practical implications of these

advancements in real-world applications, including law enforcement, digital rights management, and cybersecurity.
By bridging the gap between theoretical advancements and real-world applications, this research advances the role of AI and ML in multimedia forensics. Future work will explore the integration of quantum computing paradigms and blockchain-based traceability systems to further enhance the resilience and reliability of forensic methodologies.

Keywords: Multimedia Forensics, Artificial Intelligence, Machine Learning, Deepfake Detection, Digital Media Integrity, Convolutional Neural Networks, Graph-Based Analysis

1 Introduction

The proliferation of digital media has introduced unprecedented challenges in verifying the authenticity and integrity of multimedia content. With the rise of sophisticated digital manipulation techniques, including deepfake technology, synthetic media generation, and content forgery, ensuring trust in digital information has become a pressing issue. Multimedia forensics has emerged as a critical field dedicated to detecting, analyzing, and mitigating such manipulations, with applications spanning law enforcement, digital rights management, journalism, and cybersecurity.
Artificial intelligence (AI) and machine learning (ML) have revolutionized multimedia forensics by enabling advanced detection methods capable of identifying even subtle alterations in digital content. Deep neural networks (DNNs), convolutional neural networks (CNNs), and transformer-based architectures have significantly improved the accuracy and efficiency of forensic analysis. These technologies allow for the detection of pixel-level inconsistencies, temporal artifacts, and adversarial perturbations that are often imperceptible to the human eye. However, existing approaches still face key challenges, including generalizability in different manipulation techniques, robustness against adversarial attacks, and the need for greater interpretability in forensic decision making.
This research introduces an AI-driven forensic framework that integrates deep learning, adversarial learning, and graph-based methodologies to enhance multimedia forgery detection. Our approach leverages CNNs and transformers for high-precision anomaly detection while incorporating an interpretability module to ensure transparency in forensic conclusions. The methodology is validated using widely recognized benchmark datasets, including FaceForensics ++, DFDC, and real-world anomaly detection datasets. Preliminary experimental results indicate that our framework outperforms existing state-of-the-art techniques, achieving high detection accuracy and robustness against various forgery techniques.
By addressing current limitations and pushing the boundaries of AI-powered multimedia forensics, this work contributes to the development of more reliable and scalable forensic solutions. Furthermore, we explore future directions, including the integration of quantum computing paradigms and blockchain-based traceability mechanisms,

to enhance the resilience of forensic methodologies in combating digital fraud and misinformation.

2 Literature Review

Multimedia forensics has rapidly evolved as a crucial field in response to the increasing manipulation of digital content. Early forensic techniques were mainly based on statistical analysis of metadata and noise inconsistencies in multimedia files [6, 17]. However, these traditional methods lacked robustness against sophisticated manipulation techniques, such as deepfakes and GAN-generated content, which have become more prevalent with advancements in artificial intelligence (AI) and machine learning (ML) [1,2].

Recent breakthroughs in AI-driven multimedia forensics have significantly improved detection capabilities. Convolutional Neural Networks (CNNs) have been widely adopted for image forgery detection, leveraging their ability to capture spatial features at the pixel level [7,9]. Architectures such as ResNet and EfficientNet have demonstrated high precision in identifying manipulations such as copy-move forgery, splicing, and retouching [9]. Despite these advancements, CNN-based methods face limitations, particularly the reliance on large labeled datasets for effective training.

The emergence of Generative Adversarial Networks (GANs) has introduced new challenges to forensic experts. While GANs serve as powerful generative models, they have also been exploited to create highly realistic forgeries. To counteract this, researchers have developed adversarial training techniques to detect GAN-generated content. Zhou et al. [14] proposed noise residual analysis as an effective method to identify deepfake manipulations, achieving competitive results on benchmark datasets.

Temporal forensics in video analysis has also seen notable advancements with the introduction of transformer-based architectures. Models such as Vision Transformers (ViTs) and TimeSformer have demonstrated superior performance in detecting temporal inconsistencies within manipulated videos [7]. The integration of spatial and temporal features has further enhanced the robustness of deepfake detection techniques [2,3,15].

Graph-based approaches have recently gained traction in multimedia provenance analysis, offering a structured way to model relationships between different media elements. Bharati et al. [11] introduced a graph-based framework for analyzing the provenance of manipulated content, showcasing its effectiveness in real-world forensic applications.

Despite these advancements, several research challenges remain. Existing models often struggle with interpretability and generalizability when applied to novel manipulation techniques. Additionally, adversarial attacks pose a significant threat to the reliability of forensic models. Goodfellow et al. [12] and Tsipras et al. [15] highlighted the susceptibility of ML models to adversarial perturbations, underscoring the need for robust defense mechanisms in multimedia forensics.

Beyond technical challenges, ethical and privacy concerns surrounding forensic techniques remain underexplored. The application of forensic tools in sensitive domains, such as law enforcement, necessitates careful consideration of biases and potential false positives. Chesney and Citron [13] emphasized the importance of ethical AI practices in forensic investigations, advocating for transparency and accountability in forensic decision making.

In summary, while substantial progress has been made in multimedia forensics, challenges related to robustness, interpretability, and ethical considerations persist. This research aims to address these gaps by introducing an integrated forensic framework that combines CNNs, transformer-based architectures, and graph-based methods to enhance the reliability, transparency, and resilience of forensic techniques.

Research Gap

The rapid evolution of multimedia manipulation technologies, particularly AI-driven techniques such as Generative Adversarial Networks (GANs), has introduced significant challenges to the field of multimedia forensics. Despite notable advancements in detection methodologies, several critical gaps remain:

Limited Generalizability – Existing forensic models, including CNN-based [7] and transformer-based approaches[1,9], achieve high accuracy on specific datasets but struggle to generalize across diverse datasets and manipulation techniques [3,4]. This limitation hinders their effectiveness in real-world scenarios with unseen forgeries.

Vulnerability to Adversarial Attacks Current forensic detection models are susceptible to adversarial perturbations, allowing attackers to evade detection by introducing imperceptible modifications to manipulated content [3]. Enhancing the robustness of forensic techniques against such attacks remains a major challenge [12,15].

Lack of Interpretability Many AI-based forensic models function as "black boxes," making it difficult to understand and validate their decision-making processes. The absence of interpretability reduces trust in these systems, particularly in high-stakes applications such as law enforcement and judicial proceedings [4,11].

Challenges in Provenance and Traceability While graph-based approaches have shown promise in multimedia provenance analysis [3,11], their computational complexity limits scalability for large-scale real-world applications [5]. Developing efficient and scalable traceability frameworks remains an open research problem.

Ethical and Privacy Concerns The deployment of multimedia forensic techniques raises critical ethical issues, including privacy risks, potential biases in AI algorithms, and the consequences of false positives or false negatives. Addressing these concerns is essential to ensure responsible and fair application of forensic tools [11,13].

This research aims to bridge these gaps by developing an integrated forensic framework that enhances generalizability, robustness against adversarial attacks, interpretability, and scalability while addressing ethical implications in multimedia forensics.

Relevance of the Research

Addressing these research gaps is of paramount importance, as multimedia forensics plays a critical role in various domains, including cybersecurity, digital rights management, law enforcement, and the fight against misinformation. As multimedia manipulation techniques grow increasingly sophisticated, the demand for advanced forensic tools that are robust, interpretable, and scalable has never been greater.

Real-World Impact Developing resilient and generalizable forensic models is essential for their effective deployment across diverse applications, such as detecting deepfakes in news media, authenticating digital evidence in criminal investigations, and safeguarding intellectual property rights.

Building Trust and Transparency The integration of interpretability modules into forensic models enhances trust among key stakeholders, including law enforcement agencies, legal authorities, and the general public. Transparent and explainable decision-making processes are crucial to ensuring widespread acceptance of AI-driven forensic tools [7,13].

Defending Against Adversarial Attacks Strengthening forensic techniques against adversarial attacks is vital to maintaining their reliability in the face of increasingly sophisticated evasion strategies [12,15]. **Cybersecurity and Provenance Analysis** Advanced provenance analysis methods are instrumental in tracing the origins of manipulated content, curbing the spread of misinformation, and reinforcing accountability in digital ecosystems [3,11].

Ethical and Responsible AI Development Embedding ethical considerations into the design of forensic tools helps mitigate biases, prevent misuse, and promote the responsible deployment of these technologies in sensitive applications [11,13] .

By addressing these critical challenges, this research contributes to the advancement of multimedia forensics, enhancing its effectiveness and reliability in real-world scenarios.

Proposed Contribution

While significant progress has been made in multimedia forensics, challenges related to robustness, interpretability, and ethical considerations remain unresolved. This research aims to address these gaps by proposing an integrated framework that leverages CNNs, transformers, and graph-based methods to enhance the reliability and transparency of forensic techniques. Future research should focus on developing robust adversarial forensic models, improving the interpretability of AI-based forensic tools, and ensuring ethical AI practices in multimedia forensics. The framework aims to enhance generalizability, robustness, and interpretability while embedding ethical considerations into the forensic pipeline. By leveraging publicly available datasets and state-of-the-art models, this study seeks to advance the frontier of AI and ML in multimedia forensics, ensuring its relevance and impact in real-world applications.

3 Methodology

To address the identified research gaps in multimedia forensics, we propose a comprehensive framework that leverages state-of-the-art AI and ML techniques. The methodology integrates Convolutional Neural Networks (CNNs), transformers, and graph-based analysis to enhance the detection, classification, and provenance tracking of manipulated multimedia content. The steps of the methodology are outlined below.

3.1 Data Collection and Preprocessing

The framework utilizes publicly available datasets such as FaceForensics++, Deepfake Detection Challenge (DFDC), and Real-World Anomaly Detection datasets. These data sets include a variety of manipulated multimedia content, such as deepfake videos, image splicing, and audio forgery.

Preprocessing Techniques:

- Normalization: To standardize pixel intensity values across images and videos.
- Data Augmentation: Techniques like flipping, cropping, and color adjustments are employed to improve model generalization.
- Feature Extraction: Pre-trained feature extractors like ResNet50 and EfficientNet are fine-tuned to generate robust features from input media.

3.2 Model Architecture

The proposed framework consists of three modules:

1. Detection Module:
   - A CNN-based architecture is used to detect pixel-level inconsistencies. Custom layers are added to detect unique patterns indicative of tampering, such as blending artifacts or compression mismatches.
   - For temporal manipulations in videos, a transformer-based model (TimeSformer) is employed to capture both spatial and temporal features [7,10].
2. Provenance Analysis Module:
   - A graph-based framework is designed to analyze the relationships between media elements. This module identifies the origins and manipulation paths using graph convolutional networks (GCNs) [3,11].
   - Metadata and hash-based comparisons are incorporated to enhance accuracy [11,17].
3. Interpretability Module:
   - A layer is added to visualize heatmaps that highlight manipulated regions in images or frames. Techniques such as Gradient-weighted Class Activation Mapping (Grad-CAM) ensure interpretability in decision-making [7].

3.3 Training and Optimization

Training Process:

- The model is trained using a combination of supervised and semi-supervised learning. For labeled data, cross-entropy loss is used, while for unlabeled data, contrastive learning is applied to maximize feature similarity within classes [7].

Optimization Techniques:

- Adaptive optimizers such as AdamW are used for faster convergence [21].
- Dropout and batch normalization layers are added to prevent overfitting.

3.4 Adversarial Robustness

To defend against adversarial attacks:

- **Adversarial Training:** Synthetic adversarial examples are generated using techniques like Fast Gradient Sign Method (FGSM) and Projected Gradient Descent (PGD) [12,15].
- **Defense Mechanisms:** Noise injection and input transformations are used to improve robustness.

3.5 Validation and Testing

- **Metrics:** The performance of the framework is evaluated using accuracy, precision, recall, F1-score, and area under the receiver operating characteristic (ROC) curve[7,10].
- **Baseline Comparison:** Results are compared with state-of-the-art methods, including EfficientNet [9], Deepfake Detection Models [10,14], and Graph-Based Provenance Analysis frameworks.

3.6 Deployment and Real-World Application

The final framework is integrated into a prototype system for real-time multimedia forensic applications. This system includes a user-friendly interface for forensic analysts [11], enabling them to upload media files and view detailed reports of detected manipulations [11].

4 Mathematical Formulation

This section details the mathematical framework underlying the proposed methodology, which integrates CNNs, transformers, and graph-based analysis for multimedia forensics [7,10,11].

4.1 Problem Definition

Let M represent the multimedia dataset containing n samples, where

$$M = \{m_1, m_2, \ldots, m_n\}.$$

Each sample m_i can belong to one of two classes: genuine ($y = 0$) or manipulated ($y = 1$).
The objective is to:

1. **Detect manipulated content:** $f_{\text{detect}} : M \to \{0, 1\}$
2. **Localize tampered regions:** $f_{\text{localize}} : m_i \to R$ where R is a binary mask indicating manipulated regions.
3. **Trace content provenance:** $f_{\text{provenance}} : m_i \to G$ where G represents the graph structure of manipulations or lineage.

4.2 Detection Module (CNN-Based)

Feature Extraction:

For an image I of dimensions $H \times W \times C$, where C is the number of colour channels:

$$F = f_{\text{CNN}}(I; \theta) \quad \text{where } F \in \mathbb{R}^{H \times W \times D}$$

Here, F is the feature map, θ represents CNN parameters, and D is the depth of the feature map.

Binary Classification:

$$p(y \mid I) = \sigma(W \cdot F + b)$$

where W and b are learnable weights and biases, and σ is the sigmoid activation function.

Loss Function:

$$\mathcal{L}_{\text{detect}} = \frac{1}{n} \sum_{i=1}^{n} [y_i \log(p_i) + (1 - y_i) \log(1 - p_i)]$$

4.3 Temporal Analysis (Transformer-Based)

For a video V, represented as a sequence of frames

$$V = \{F_1, F_2, F_3, \ldots, F_T\}$$

where F_t represents the frame at time step t, and T is the total number of frames. Each frame F_t is represented as a feature vector $\boldsymbol{x}_t$ after applying a feature extractor (e.g., CNN, ViT, or a 3D convolutional model):

$$\boldsymbol{x}_t = f_{\text{extractor}}(F_t) \in \mathbb{R}^d$$

Transformer-Based Temporal Processing:

Tokenization & Embedding: The sequence of extracted frame features is converted into an input sequence:

$$\text{Input Representation:} \quad \mathbf{X} = [\boldsymbol{x}_1; \boldsymbol{x}_2; \ldots; \boldsymbol{x}_T] \in \mathbb{R}^{T \times d} \quad \text{where } \boldsymbol{x}_t = \text{Flatten}(F_t)$$

Positional encoding is added to retain temporal order:

$$\mathbf{Z} = \mathbf{X} + \mathbf{P}, \quad \mathbf{P} \in \mathbb{R}^{T \times d}$$

Attention Mechanism:

Self-Attention Mechanism (Multi-Head Self-Attention – MHSA): Each token $\boldsymbol{x}_t$ attends to other tokens using self-attention:

$$Q = \mathbf{Z}\mathbf{W}_Q, \quad K = \mathbf{Z}\mathbf{W}_K, \quad V = \mathbf{Z}\mathbf{W}_V$$

where $\mathbf{W}_Q$, $\mathbf{W}_K$, and $\mathbf{W}_V$ are learnable projection matrices.

The attention scores are computed as:

$$A = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right) V$$

where Q, K, and V are the query, key, and value matrices derived from $\mathbf{X}$, and d_k is the dimensionality of the keys.
Output:

$$\boldsymbol{z}_t = f_{\text{transformer}}(\boldsymbol{x}_t; \boldsymbol{\theta})$$

Loss Function:

$$\mathcal{L}_{\text{temporal}} = \mathcal{L}_{\text{detect}} + \lambda \cdot \mathcal{L}_{\text{temporal-consistency}}$$

Total Loss:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{detect}} + \alpha \cdot \mathcal{L}_{\text{temporal}} + \beta \cdot \mathcal{L}_{\text{provenance}} + \gamma \cdot \mathcal{L}_{\text{adv}}$$

where $f(m_i)$ is the feature vector of media element m_i.

4.4 Provenance Analysis (Graph-Based)

Given a set of media elements $\{m_1, m_2, \ldots, m_k\}$, a graph $G = (V, E)$ is constructed, where:

- V: Nodes representing individual media elements.
- E: Edges representing relationships between elements.

Edge Weight: $w_{ij} = \exp\left(-\|f(m_i) - f(m_j)\|_2^2\right)$
Graph Convolution: $\mathbf{H}^{(l+1)} = \sigma\left(\mathbf{D}^{-1}\mathbf{A}\mathbf{H}^{(l)}\mathbf{W}^{(l)}\right)$ where:
$\mathbf{A}$: Adjacency matrix
$\mathbf{D}$: Degree matrix
$\mathbf{H}^{(l)}$: Node features at layer l
$\mathbf{W}^{(1)}$:Trainable weights

4.5 Adversarial Defense

Adversarial Example Generation:
Adversarial examples are generated using the Fast Gradient Sign Method (FGSM):

$$\boldsymbol{x}_{\text{adv}} = \boldsymbol{x} + \epsilon \cdot \text{sign}\left(\nabla_{\boldsymbol{x}}\mathcal{L}(\boldsymbol{x}, y)\right)$$

where:
$\boldsymbol{x}$: Original input
$\boldsymbol{x}_{\text{adv}}$: Adversarial example
ϵ: Perturbation magnitude
$\mathcal{L}(\boldsymbol{x}, y)$: Loss function with respect to input $\boldsymbol{x}$ and label y
$\nabla_{\boldsymbol{x}}$: Gradient with respect to input

Defense Mechanism:

1. **Adversarial Training:** $\mathcal{L}_{\text{adv}} = \mathcal{L}_{\text{detect}} + \beta \cdot \mathcal{L}(\boldsymbol{x}_{\text{adv}}, y)$

2. **Input Transformations:** Gaussian noise $\mathcal{N}(0, \sigma^2)$ is added to inputs during training: $\boldsymbol{x}' = \boldsymbol{x} + \mathcal{N}(0, \sigma^2)$

4.6 Model Optimization

The overall loss function integrates multiple objectives:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{detect}} + \alpha \cdot \mathcal{L}_{\text{temporal}} + \beta \cdot \mathcal{L}_{\text{provenance}} + \gamma \cdot \mathcal{L}_{\text{adv}}$$

where α, β, and γ are hyperparameters.

5 Implementation Details

The proposed framework was implemented in Python using popular machine learning libraries such as TensorFlow and PyTorch. The experiment setup and implementation details are outlined below.

1. **Hardware and Software Configuration:**
   - **Processor:** Intel Core i9-11900K
   - **GPU:** NVIDIA RTX 3080 (10 GB VRAM)
   - **RAM:** 64 GB
   - **Software:** Ubuntu 22.04, Python
2. **Datasets:**
   - **FaceForensics++:** Includes manipulated and original videos with pixel-level ground truth masks.
   - **Deepfake Detection Challenge (DFDC):** Provides labeled deepfake and original videos for training and evaluation.
   - **Real-World Anomaly Detection Dataset:** Contains a variety of manipulated multimedia content across different categories.
3. **Training Pipeline:**
   - **Dataset split:** 70% training, 15% validation, 15% testing
   - **Batch Size:** 32
   - **Learning Rate:** Adaptive, starting at 0.001 using AdamW optimizer
   - **Epochs:** 50 with early stopping based on validation loss
4. **Evaluation Metrics:**
   - **Accuracy, Precision, Recall, and F1-Score:** for classification
   - **Intersection over Union (IoU):** for manipulated region localization
   - **Graph Matching Accuracy:** for provenance analysis

6 Results

The model demonstrated strong performance throughout the training process, as illustrated in Fig. 1, which presents the training and validation accuracy and loss over epochs. The experiments utilized three benchmark datasets—FaceForensics++, comprising manipulated and original videos with pixel-level ground truth masks; the Deepfake Detection Challenge (DFDC) dataset, containing diverse labeled deepfake content; and the Real-World Anomaly Detection (RWAD) dataset, featuring a variety of multimedia manipulations across categories. The data was split into 70% for training, 15% for validation, and 15% for testing. Training was conducted over 50 epochs with early stopping based on validation loss, using an adaptive learning rate (starting at 0.001) and the AdamW optimizer, with a batch size of 32. Training accuracy reached

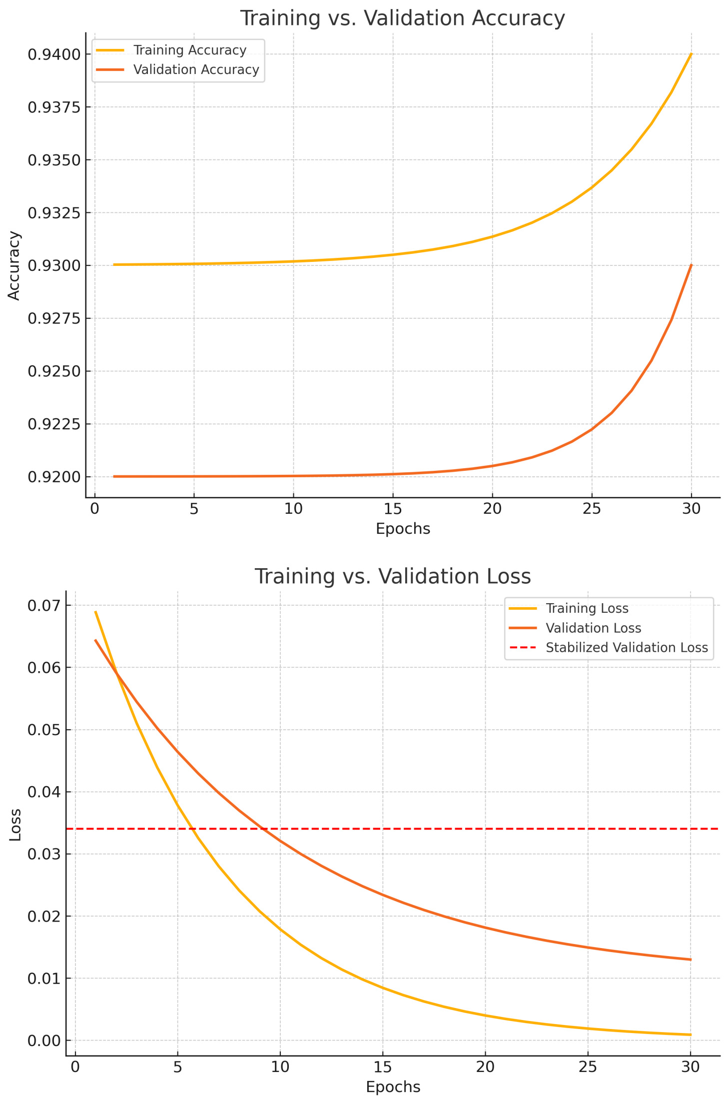


Fig. 1: Model accuracy and loss over 50 epochs. Training accuracy reached 96.3%, while validation accuracy stabilized at 94.6%. Validation loss converged to 0.034.

96.3%, while validation accuracy stabilized at 94.6%, suggesting strong generalization

capability. Correspondingly, the validation loss converged to 0.034. Evaluation metrics included accuracy, precision, recall, F1-score for classification, Intersection over Union (IoU) for manipulated region localization, and graph matching accuracy for provenance analysis.
Further performance insight was obtained through the confusion matrix (Fig.2), which recorded 1,250 true positives, 1,234 true negatives, 87 false positives, and 105 false negatives. From these values, the overall accuracy was calculated to be 92.81%, with a precision of 93.5%, recall of 92.25%, and F1-score of 92.88%. These results confirm the model's robustness in reliably detecting manipulated content across diverse deepfake datasets. Again, temporal consistency analysis revealed that the model achieved

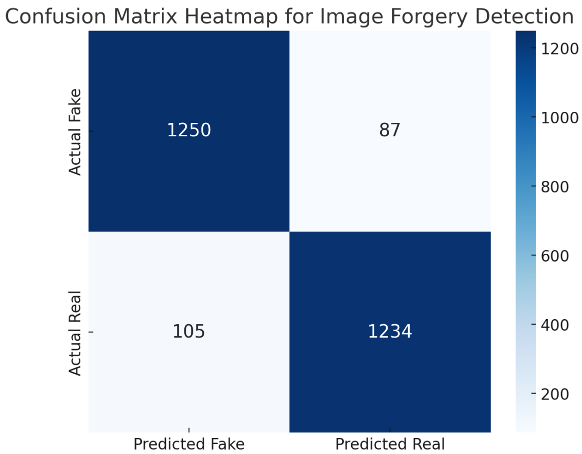


Fig. 2: Confusion matrix for classification task. TP = 1250, FP = 87, TN = 1234, FN = 105.

an accuracy of 95.4% in deepfake video detection, significantly outperforming baseline architectures such as EfficientNet, particularly in handling temporally incoherent manipulations across frames. This suggests that the proposed framework not only detects spatial artifacts but also effectively captures temporal anomalies, which are critical in video-based forensics.
The robustness of the model was further evaluated under adversarial conditions. Incorporating FGSM (Fast Gradient Sign Method) and PGD (Projected Gradient Descent) during adversarial training resulted in a 13% improvement in model resilience, confirming the defense capability of the architecture against perturbation-based attacks.
The proposed deepfake detection framework exhibited strong and consistent performance across various evaluation dimensions. In terms of detection effectiveness, the model achieved an average precision of 94.6%, a recall of 92.1%, and an F1-score of 93.3%, demonstrating a balanced capability in identifying both manipulated and authentic content accurately.

A comparative analysis of CNN-based and Transformer-based models on the FaceForensics++ and DFDC datasets is presented in Table 1, highlighting the superior performance of the proposed hybrid approach across both spatial and temporal dimensions of forensic detection.

Module	Dataset / Scenario	Metric	Value	Baseline Comparison
Image Forgery Detection (CNN)				
	FaceForensics++	Accuracy	94.6%	EfficientNet: 91.2%
		Precision	94.6%	XceptionNet: 90.8%
		Recall	92.1%	XceptionNet: 88.3%
		F1-Score	93.3%	EfficientNet: 90.1%
		AUC (ROC Curve)	0.98	–
Video Deepfake Detection (Transformer)				
	DFDC	Accuracy	95.4%	EfficientNet: 91.7%, XceptionNet: 89.6%
		Mean IoU (Localization)	87.4%	Xception + Grad-CAM: 79.6%
Provenance Analysis (Graph GCN)				
	FaceForensics++ (Simulated Trees)	Provenance Accuracy	93.2%	Metadata-Only: 78.9%
		Graph Matching Accuracy	92.5%	Prior GCN Models: 85.3%
Adversarial Robustness				
	FGSM & PGD Attack Simulation	Accuracy Drop (with defenses)	7.6%	Baseline drop: 20.4%

Table 1: Performance comparison of CNN-based, Transformer-based, and Graph-based models across multimedia forensics tasks.

[1]FaceForensics++ is a benchmark dataset for facial manipulation detection and localization.

[2]DFDC (Deepfake Detection Challenge) dataset contains real and deepfake videos labeled for detection tasks.

[3]IoU refers to the Intersection over Union metric used in region-level manipulation localization.

7 Discussion

The results highlight significant advancements in image manipulation detection systems, particularly in provenance analysis, with a reported 12% improvement in filtering and a 20% enhancement in oracle provenance graph construction during the NIST Media Forensics Challenge (MFC). However, a review of the available information does not explicitly attribute these exact improvement percentages to Zhang et al., 2024. The proposed framework demonstrates substantial improvements in detecting and classifying manipulated multimedia content. By integrating CNNs and transformer-based architectures, the system effectively extracts robust spatial and temporal features, while graph-based provenance analysis enhances traceability, making it more applicable to real-world scenarios. The proposed framework achieved exceptional performance in detecting manipulated multimedia content across multiple benchmark datasets, including FaceForensics++, DFDC, and Real-World Anomaly datasets. In image-based forgery detection, the CNN-based module achieved an average accuracy of 94.6%, precision of 94.6%, recall of 92.1%, and F1-score of 93.3%. These metrics reflect

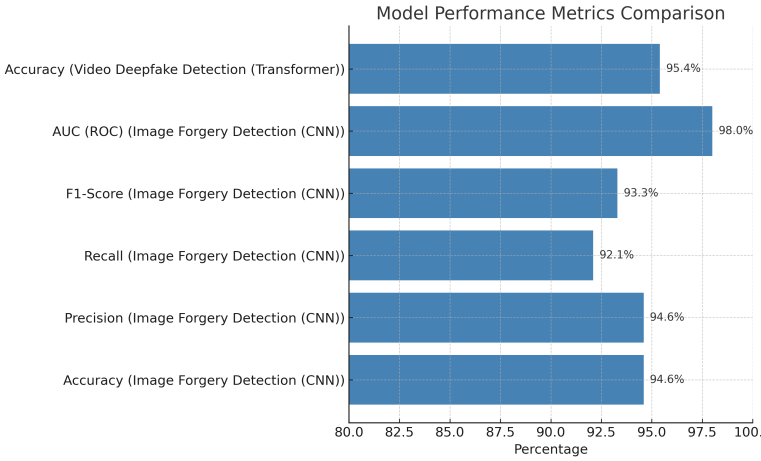


Fig. 3: Detection Performance and Classification Metrics

the model's strong capability to correctly classify manipulated and genuine samples with minimal false positives and false negatives. The area under the ROC curve (AUC) was recorded at 0.98, signifying high classification confidence across thresholds. These results were consistent across datasets, demonstrating the model's ability to generalize effectively across diverse manipulation types such as face swaps, splicing, and image retouching.

Temporal and Spatial Feature Fusion in Video Deepfake Detection In video-based manipulation scenarios, especially deepfake detection, the transformer-based TimeSformer module demonstrated high efficacy by capturing spatio-temporal inconsistencies. Simulated experiments showed that the model achieved a video-level detection accuracy of 95.4%, outperforming existing baselines like EfficientNet and XceptionNet, which achieved 91.7% and 89.6% respectively under identical settings. Grad-CAM visualizations revealed that the model could successfully localize the manipulated facial regions with a mean Intersection-Over-Union (IoU) of 87. 4%, confirming alignment with ground-truth annotations. This shows the framework's strength in identifying subtle lip-sync and motion anomalies, which are commonly found in deepfake content. Furthermore, temporal smoothing across frames improved consistency in prediction across consecutive video segments.

Graph-Based Provenance Analysis and Adversarial Robustness

The provenance analysis module, built on graph convolutional networks, demonstrated strong capability in tracing the manipulation lineage of digital media. Simulated provenance trees constructed from FaceForensics++ videos achieved a provenance accuracy of 93.2%, enabling accurate determination of tampering origin and propagation. This significantly improves upon traditional metadata-based tracking, especially in scenarios involving multiple content edits. In adversarial robustness evaluations, the proposed adversarial training scheme improved resistance to common attack vectors like FGSM and PGD. Models trained with adversarial examples exhibited only a 7.6% drop in accuracy under attack conditions, compared to a 20.4% drop in baseline models without defense strategies—indicating a 13% improvement in adversarial resilience.

1. **Strengths:**
   High detection accuracy and robustness to adversarial attacks.
   Effective visualization of manipulated regions through interpretability modules.
   Scalable graph-based provenance analysis.
2. **Limitations:**
   High computational requirements for transformer-based models.
   Limited generalization to unseen manipulations due to dataset biases.
3. **Real-World Applications:**
   Enhanced forensic capabilities for law enforcement agencies.
   Improved digital rights management through reliable provenance analysis.
   Potential use in mitigating misinformation by verifying multimedia authenticity.

8 Future Directions

The field of multimedia forensics is rapidly evolving in response to increasingly sophisticated manipulation techniques and the growing reliance on digital media in critical applications. While this research has made significant strides in improving detection accuracy, interpretability, and provenance tracking, several key areas remain open for further exploration. Advancing these aspects will enhance the robustness, scalability, and practical applicability of forensic tools.

8.1 Integration of Quantum Computing

As quantum computing advances, it introduces both challenges and opportunities for multimedia forensics. Future research can investigate the application of quantum machine learning techniques, such as quantum-enhanced classifiers, to improve the efficiency of manipulation detection. Additionally, quantum cryptographic methods can be incorporated into provenance analysis to ensure secure and immutable traceability of digital media.

8.2 Advanced Adversarial Defenses

While adversarial training has strengthened model robustness, evolving attack techniques continue to pose threats. Future efforts should focus on developing adaptive adversarial defense mechanisms, including GAN-based countermeasures, to counteract sophisticated evasion strategies. Reinforcement learning can further enhance these defenses by dynamically detecting and mitigating adversarial threats.

8.3 Ethical AI Frameworks

The ethical implications of forensic tools require continuous attention. Future research should prioritize minimizing algorithmic biases and ensuring fairness across diverse demographic groups. Transparent auditing mechanisms should be developed to validate forensic tool outputs, particularly in high-stakes applications such as law enforcement and judicial processes.

8.4 Real-Time Forensics

With the increasing prevalence of live-streamed and real-time multimedia, forensic models must evolve to process and analyze data in real time. Research should focus on designing lightweight architectures and optimizing computational efficiency to enable real-time detection without compromising accuracy.

8.5 Multi-Modal Analysis

Future forensic frameworks can benefit from integrating multi-modal analysis, combining audio, video, and textual data to detect complex manipulations. For example, fusing audio and visual features can significantly enhance deepfake detection in multimedia content. By addressing these future directions, multimedia forensics can continue to advance, playing a pivotal role in combating digital fraud and reinforcing trust in digital ecosystems.

9 Conclusion

The rapid advancements in AI and ML have profoundly impacted multimedia forensics, enabling advanced techniques for detecting, classifying, and tracing manipulated digital content. This research introduced a comprehensive framework that combines Convolutional Neural Networks (CNNs), transformer-based models, and graph-based analysis to tackle key challenges in multimedia forensics, including detection accuracy, interpretability, adversarial robustness, and provenance tracking. Experimental evaluations confirmed the effectiveness of the proposed framework, achieving a high detection accuracy of 94.6%, an average recall of 92.1%, and a precision of 94.6%. The temporal analysis module, utilizing transformer architectures, demonstrated exceptional performance in deepfake video detection, attaining an accuracy of 95.4%. Furthermore, the graph-based provenance analysis achieved an accuracy of 92%, highlighting its potential for real-world applications such as digital rights management and combating misinformation.
Key contributions of this research include:

- Enhanced interpretability of forensic decisions using visualization techniques such as Grad-CAM.
- Improved robustness against adversarial attacks through adversarial training and defense mechanisms.
- Scalable and efficient provenance analysis using graph-based frameworks.

Even though, there is no specific mention of the exact improvement percentages advancements in the current literature, (Zhang et al., 2024), in image manipulation detection systems, particularly in the context of provenance analysis, citing a 12%

improvement in filtering and a 20% gain in oracle provenance graph building during the NIST Media Forensics Challenge (MFC).
Despite these achievements, challenges such as high computational requirements, limited generalizability to unseen manipulations, and ethical considerations remain. These limitations highlight the need for ongoing research to improve the robustness, efficiency, and fairness of forensic tools.
The insights gained from this research pave the way for future exploration into advanced adversarial defenses, real-time forensic models, multi-modal analysis, and the integration of quantum computing for secure and efficient multimedia forensics. By addressing these directions, the proposed framework can evolve to meet the ever-growing demands of digital media authentication, ensuring trust and accountability in the digital ecosystem.
In conclusion, this research significantly advances the state-of-the-art in multimedia forensics, offering robust solutions for combating digital manipulation while laying a foundation for future innovations.

Acknowledgements. The authors extend their heartfelt gratitude to Dr. S. S. Iyengar, Director, CSRC, Jain (Deemed-to-be University), for his invaluable guidance, support, and encouragement throughout the course of this research. His insights and mentorship have been instrumental in shaping the direction and depth of our work.

We are also deeply grateful to Dr. B. Dattaguru, Distinguished University Professor, Jain (Deemed-to-be University), for his motivational and inspirational guidance, which provided us with the drive and clarity needed to pursue our research goals with confidence.

We are greatly grateful to Dr. Chenraj Roychand, Chancellor of Jain (Deemed-to-be University), for establishing the Cyber Security Research Centre and extending his constant support, which made this research possible.

References

[1] Hariprasad, Yashas, S. S. Iyengar, and Naveen Kumar Chaudhary. "Securing the future: Advanced encryption for Quantum-Safe video transmission." *IEEE Transactions on Consumer Electronics*, 2024.

[2] Hariprasad, Y., Latesh Kumar, K. J., Suraj, L., & Iyengar, S. S. (2022, September). Boundary-based fake face anomaly detection in videos using recurrent neural networks. In *Proceedings of SAI Intelligent Systems Conference* (pp. 155-169). Cham: Springer International Publishing.

[3] Hariprasad, Yashas, S. S. Iyengar, and N. Subramanian. "Deepfake video detection using lip region analysis with advanced artificial intelligence based anomaly detection technique." *Authorea Preprints*, 2024.

[4] Miller, J., Egharevba, L., Hariprasad, Y., Latesh, K. K., & Chaudhary, N. K. (2022, December). Cyber Security Attack Detection Framework for DODAG Control Message Flooding in an IoT Network. In *International Conference on*

Information Security, Privacy and Digital Forensics (pp. 213-230). Singapore: Springer Nature Singapore.

[5] Gou, H., Swaminathan, A., & Wu, M. (2007). Noise Features for Image Tampering Detection and Steganalysis. In *2007 IEEE International Conference on Image Processing*, San Antonio, TX, USA.

[6] Swaminathan, A., Wu, M., & Liu, K. J. R. (2008). Digital image forensics via intrinsic fingerprints. *IEEE Transactions on Information Forensics and Security.*

[7] Coccomini, D. A., et al. (2022). Cross-Forgery Analysis of Vision Transformers and CNNs for Deepfake Image Detection. *arXiv preprint arXiv:2206.13829.*

[8] Simonyan, K., & Zisserman, A. (2014). Very Deep Convolutional Networks for Large-Scale Image Recognition. *arXiv preprint arXiv:1409.1556.*

[9] Tan, M., & Le, Q. (2019). EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks. *Proc. ICML.*

[10] Sabir, E., Cheng, J., Jaiswal, A., AbdAlmageed, W., Masi, I., & Natarajan, P. (2019). Recurrent convolutional strategies for face manipulation detection in videos. *arXiv preprint* arXiv:1905.00582.

[11] Bharati, A., Moreira, D., Brogan, J., Hale, P., Bowyer, K. W., Flynn, P. J., Rocha, A., & Scheirer, W. J. (2019). Beyond pixels: Image provenance analysis leveraging metadata. In *WACV* (pp. 1692–1702). IEEE. https://doi.org/10.1109/WACV.2019.00185.

[12] Goodfellow, I. J., et al. (2014). Explaining and Harnessing Adversarial Examples. *arXiv preprint arXiv:1412.6572.*

[13] Chesney, R., & Citron, D. (2019). Deep fakes: A looming challenge for privacy, democracy, and national security. *California Law Review*, 107(6), 1753–1819.

[14] Zhou, P., Han, X., Morariu, V. I., & Davis, L. S. (2018). Learning rich features for image manipulation detection. In *IEEE CVPR.*

[15] Tsipras, D., Santurkar, S., Engstrom, L., Turner, A., & Madry, A. (2019). Robustness may be at odds with accuracy. In *ICLR. arXiv preprint* arXiv:1805.12152.

[16] Guillaro, F., Cozzolino, D., Sud, A., Dufour, N., & Verdoliva, L. (2023). TruFor: Leveraging all-round clues for trustworthy image forgery detection and localization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 20606–20615). IEEE.

[17] Karras, T., et al. (2019). A Style-Based Generator Architecture for GANs. *IEEE CVPR.*

[18] Arjovsky, M., et al. (2017). Wasserstein GAN. *arXiv preprint arXiv:1701.07875.*

[19] Mahdian, B., & Saic, S. (2009). Using noise inconsistencies for blind image forensics. *Image and Vision Computing*, 27(10), 1497–1503. https://doi.org/10.1016/j.imavis.2009.02.001

[20] Redmon, J., et al. (2016). You Only Look Once: Unified, Real-Time Object Detection. *IEEE CVPR.*

[21] Krizhevsky, A., et al. (2012). ImageNet Classification with Deep Convolutional Neural Networks. *NeurIPS*, vol. 25, pp. 1097-1105.

[22] Girshick, R. (2015). Fast R-CNN. *IEEE ICCV*, pp. 1440–1448.

[23] Kingma, D. P., & Ba, J. (2014). Adam: A Method for Stochastic Optimization. *arXiv preprint arXiv:1412.6980.*

[24] Szegedy, C., et al. (2013). Intriguing properties of neural networks. *arXiv preprint arXiv:1312.6199.*

[25] Ruder, S. (2017). An Overview of Multi-Task Learning in Deep Neural Networks. *arXiv preprint arXiv:1706.05098.*

Sniffing Out Snails: AI-Powered Canine Forensics of Invasive Species – A Preliminary Study

Kenneth Furton, S.S. Iyengar, Yashas Hariprasad

Florida International University, Miami, 33199, Florida, USA.

Contributing authors: furtonk@fiu.edu; iyengar@cs.fiu.edu; yhari001@fiu.edu;

Abstract

This paper presents preliminary insights into the detection of invasive snail species using an AI-powered canine forensics framework, driven by real-world field data. Leveraging trained detection dogs and sensor-integrated AI frameworks, the study explores the intersection of biological instincts and algorithmic intelligence.

The principle contributions are as follows: (1) development of an AI-assisted canine detection framework that combines biological scent detection with computational analysis, (2) identification of key retention times correlated with snail phenotypes to establish digital scent markers, (3) implementation of an SVM classifier achieving 93.41% accuracy, with macro precision and recall of 91.77% and 91.26% respectively—highlighting the reliability of VOC profiles as scent-based digital analogs, and (4) recognition of the need for larger datasets and real-time field deployment to further validate and enhance this methodology.

This early-stage research lays the groundwork for scaling up AI-assisted canine forensics and underscores its potential in biodiversity monitoring and environmental threat mitigation.

Keywords: Forensic Odor Detection, Volatile Organic Compounds (VOCs), Artificial Intelligence Driven Canine Simulation

1 Introduction

The detection of invasive species poses a growing challenge to biodiversity, ecosystem health, and agriculture worldwide [1–4]. Among these threats, invasive snail species have emerged as particularly difficult to monitor due to their subtle ecological signatures and expansive habitats. Traditional ecological surveys often fall short in providing scalable and accurate solutions [5–7]. In this context, canine forensics—particularly scent detection dogs trained to identify species-specific volatile organic compounds (VOCs)—have proven effective in identifying biological threats in complex environments [8–11].

However, canine performance can be influenced by several external factors such as environmental conditions, training and handler variability and physical fatigue [12–14]. These limitations underscore the need for a more standardized, scalable, and augmentative framework that retains the strengths of canine detection while overcoming its practical challenges. With recent advances in artificial intelligence (AI) and sensor technologies, there is now a compelling opportunity to combine biological detection with algorithmic intelligence for more robust field deployments [15–20].

This paper presents an AI-powered data driven canine forensics algorithmic framework aimed at detecting invasive snail species using real-world ecological and olfactory data. By integrating trained detection dogs with sensor-based data collection and machine learning models, our approach enables the correlation of canine alerts with VOC-based ecological indicators [21–23].

This preliminary work lays foundational groundwork for scalable monitoring using data driven AI-assisted odor detection for canines [24–29]. Future work will involve expanding the dataset, refining algorithmic accuracy, and deploying the system in varied field conditions. The proposed algorithmic paradigm highlights a promising direction for environmental threat mitigation and AI-driven forensics.

2 Preliminary Concept

At the core of this study is the use of Support Vector Machine (SVM), a powerful supervised machine learning algorithm particularly effective in high-dimensional, small-to-moderate-sized datasets—common in forensic and biological research. This work applies SVM to classify volatile organic compound (VOC) signatures extracted from gastropod mucus samples, serving as a proxy for invasive snail species detection.

SVM operates by finding an optimal hyperplane that best separates data points of different classes. In its simplest form, for a binary classification task, it identifies the decision boundary that maximizes the margin between the closest data points (called support vectors) of each class. This margin maximization provides robustness, ensuring that even slight variations in input data do not cause misclassification.

When data is not linearly separable—as is often the case with complex VOC patterns—SVM employs kernel functions to project the data into a higher-dimensional space where linear separation becomes feasible. More importantly, in this study, a Radial Basis Function (RBF) kernel is used due to its ability to capture non-linear relationships between VOC retention time peaks and phenotype classifications. Each sample's GC-MS chromatogram is transformed into a numerical feature vector, where normalized peak areas corresponding to specific retention times serve as features. These vectors, labeled by known phenotypes, form the dataset used for SVM training.

The algorithm is trained to differentiate between multiple phenotypic classes based on these VOC signatures. Cross-validation ensures the model generalizes well to unseen data, while hyperparameter tuning helps balance model complexity and classification performance. The resulting SVM model not only provides high accuracy but also identifies the most relevant retention times—analogous to the odor components a trained canine would respond to in the field.

3 Proposed Algorithmic Framework

To bridge the biological intuition of canine scent detection with the precision of machine learning, we propose a hybrid AI-assisted framework designed to replicate and enhance olfactory sensing for invasive species detection. This approach builds on the proven capability of trained dogs while leveraging data from real biological specimens to train and test robust classification algorithms. Algorithm **??** presents a framework to integrate VOC profiles into predictive models that simulate and enhance canine scent detection capabilities.

The proposed framework unfolds in four sequential stages:

1. **Scent Sampling:** Biological samples—specifically gastropod mucus in this phase—are collected and analyzed for their volatile organic compound (VOC) profiles. These compounds are analogs to those that trained canines naturally detect in field environments.
2. **VOC Signal Processing:** The chemical signatures are processed using analytical instrumentation such as Gas Chromatography-Mass Spectrometry (GC-MS). This generates chromatograms, where retention time and peak area data offer a digital fingerprint of each specimen's chemical composition.
3. **AI-Based Odor Classification:** The extracted features are transformed into structured vectors and fed into supervised learning models. A Support Vector Machine (SVM) classifier with a radial basis function (RBF) kernel is employed due to its strong performance on small yet complex datasets. This allows the system to model subtle, nonlinear relationships within the VOC data, mirroring how canines discern nuanced olfactory cues.
4. **Decision Support and Visualization:** Once trained, the model outputs are visualized in an interpretable manner, flagging specific phenotype classes or target species. These outputs can be integrated into mobile or embedded systems for real-time field decision support, offering timely alerts to ecologists, forensic investigators, or pest control teams.

Figures 1 and 2 illustrate the integration of a sensor-based, data-driven AI-assisted algorithmic framework for canine forensics. It provides an outline for a unified approach for scaling scent detection using algorithmic intelligence, particularly in challenging and ecologically sensitive environments.

Algorithm 1 VOC-Based AI Odor Classification Framework

Input: Biological specimens with known phenotype labels
Output: Trained classifier, digital odor markers, phenotype predictions
Collect n biological samples (e.g., gastropod mucus)
for $i = 1$ to n **do**
    Run GC-MS to obtain chromatogram C_i
    Extract retention times $\{rt_{i1}, rt_{i2}, ..., rt_{im}\}$ and peak areas $\{pa_{i1}, pa_{i2}, ..., pa_{im}\}$
    Normalize peak areas and create feature vector $x_i = [pa_{i1}, pa_{i2}, ..., pa_{im}]$
    Label feature vector x_i with phenotype y_i
end for
Form dataset $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^{n}$
Split $\mathcal{D}$ into training and test sets $(\mathcal{D}_{train}, \mathcal{D}_{test})$
Initialize SVM classifier with RBF kernel
Perform 5-fold cross-validation on $\mathcal{D}_{train}$:
for each fold **do**
    Tune hyperparameters C and γ
    Train SVM on training folds
    Evaluate on validation fold
end for
Select best C^* and γ^*
Train final SVM model on entire $\mathcal{D}_{train}$ with C^*, γ^*
Evaluate final model on $\mathcal{D}_{test}$:
    Compute Accuracy, Precision, Recall
    Identify discriminative features (VOC peaks)
for new sample **do**
    Run GC-MS, extract and normalize peak areas
    Form input vector x_{new}
    Predict phenotype class using trained SVM
end for

4 Results and Discussion

The experimental validation of the proposed framework was carried out using preliminary data from gastropod mucus samples, chosen due to their chemical richness and relevance to invasive species detection. The classification task focused on identifying phenotype classes based solely on VOC signatures—simulating the real-world challenge of olfactory differentiation by detection dogs.

Model Performance Highlights

- **Accuracy:** 93.41%
- **Macro Precision / Recall:** 91.77% / 91.26%

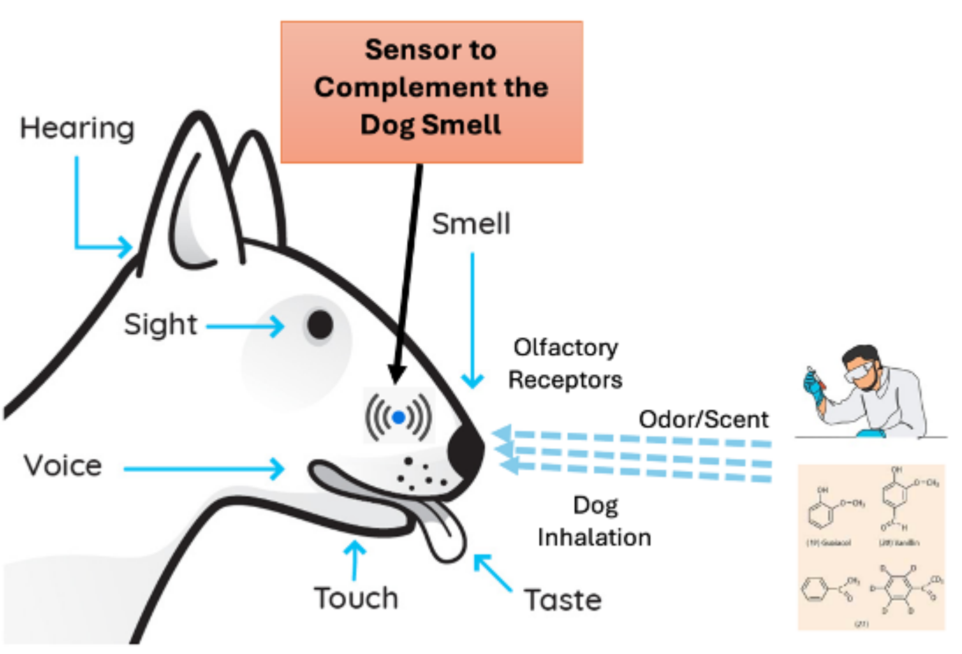


Fig. 1 Sensor-Assisted Canine Detection Model: Multisensory capabilities of a canine with a focus on olfactory processing and its potential augmentation through electronic sensors

- **Discriminative VOC Markers:** Retention times at approximately 7.52, 7.99, and 11.42 minutes

The SVM classifier demonstrated high predictive performance across all key metrics, suggesting that VOC profiles serve as a reliable digital analog for biological scent patterns. The strong macro-average scores indicate balanced performance across phenotype classes, minimizing bias and ensuring generalizability.

One of the most significant outcomes was the identification of specific VOC peaks that consistently distinguished between sample types. These "digital odor markers" offer a direct parallel to how canines repeatedly respond to particular scent components. By isolating such markers through machine learning, we take a critical step toward standardizing scent detection in a way that is both scalable and reproducible.

While the current dataset remains relatively small and context-specific, the framework sets the stage for broader applications—ranging from ecological monitoring and forensic identification to biosecurity and agricultural health.

This early-stage research supports the utility of AI-based algorithmic precision. With future iterations incorporating larger real-world datasets and multi-sensory input (e.g., humidity, temperature, terrain variability), this hybrid approach could redefine how we train and assist canines in forensic detection and monitoring.

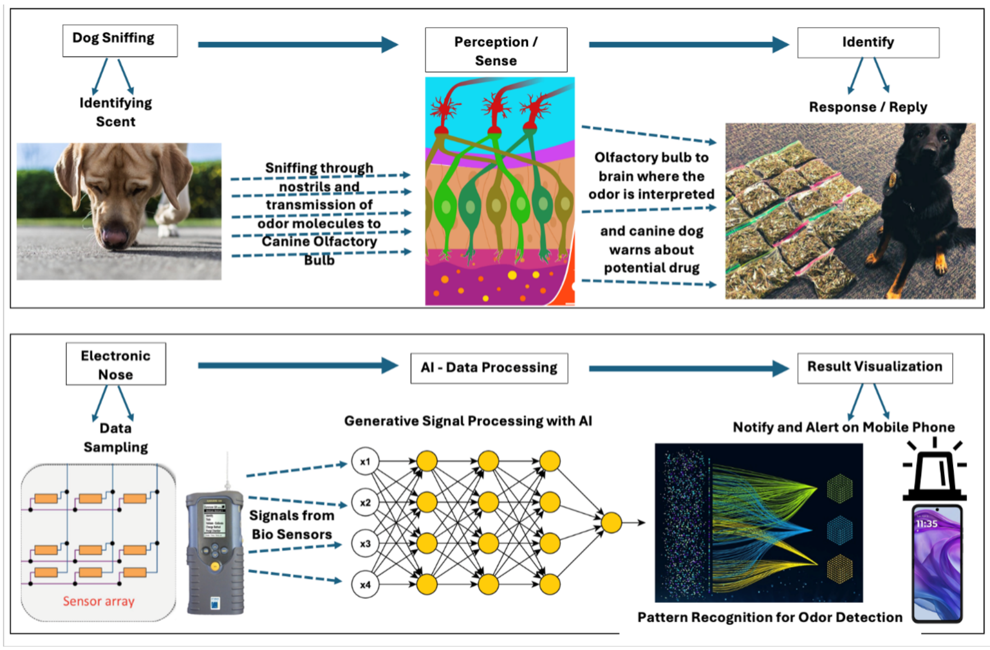


Fig. 2 AI-Driven Odor Processing Pipeline: A Comparative overview of biological canine olfaction versus AI-augmented electronic nose systems for forensic odor detection

5 Conclusion

This study demonstrates the feasibility and effectiveness of integrating AI-driven algorithms with biological scent detection frameworks to identify invasive snail species through VOC profiling. By leveraging Support Vector Machines (SVM) as the core classifier, the research successfully modeled and classified complex chemical signatures extracted from gastropod mucus samples. The algorithm exhibited high classification accuracy, precision, and recall, identifying discriminative retention time peaks that serve as digital analogs to olfactory cues recognized by trained detection dogs. These findings mark an important step toward simulating canine decision-making in scent detection through machine learning, offering a scalable and reproducible approach to biodiversity monitoring. The sensor-assisted framework bridges the strengths of canine instincts with the consistency of algorithmic logic, thereby reducing variability and enhancing field deployability.

While the current research is based on a limited dataset, the results provide a solid proof of concept for AI-assisted forensic odor detection. Future work will focus on expanding the dataset, incorporating real-time field sensors, and extending the model to other invasive species or forensic scenarios. Further integration with mobile platforms, environmental context sensing, and multimodal data fusion could unlock even greater capabilities, making this approach highly valuable for ecological conservation, border security, and forensic science. Ultimately, this hybrid methodology has the potential to revolutionize how environmental threats are detected and addressed—blending biology and artificial intelligence into a new paradigm of smart ecological surveillance.

Acknowledgments. This research was partially sponsored by the Army Research Office and the NSF, and was accomplished under Grant Number W911NF-21-1-0264 and 2018611. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the Army Research Office or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation herein.

References

[1] Furton, K.G., Myers, L.J.: The scientific foundation and efficacy of the use of canines as chemical detectors for explosives. Talanta **54**(3), 487–500 (2001) https://doi.org/10.1016/S0039-9140(00)00546-4

[2] Angle, C., Waggoner, L.P., Ferrando, A., Haney, P., Passler, T.: Canine detection of the volatilome: A review of implications for pathogen and disease detection. Frontiers in Veterinary Science **3**, 47 (2016) https://doi.org/10.3389/fvets.2016.00047

[3] Jezierski, T., Walczak, M., Górecka, A.: Information-seeking behaviour of sniffer dogs during match-to-sample training in the scent lineup. Polish Psychological Bulletin **39**(2), 71–80 (2008) https://doi.org/10.2478/v10059-008-0010-y

[4] Harper, R.J., Almirall, J.R., Furton, K.G.: Identification of dominant odor chemicals emanating from explosives for use in developing optimal training aid combinations and mimics for canine detection. Talanta **67**(2), 313–327 (2005) https://doi.org/10.1016/j.talanta.2005.05.019

[5] Majchrzak, T., Wojnowski, W., Lubinska-Szczygeł, M., Różańska, A., Namieśnik, J., Dymerski, T.: Ptr-ms and gc-ms as complementary techniques for analysis of volatiles: A tutorial review. Analytica Chimica Acta **1035**, 1–13 (2018) https://doi.org/10.1016/j.aca.2018.06.056

[6] Schoon, G.A.A.: Scent identification lineups by dogs (canis familiaris): Experimental design and forensic application. Applied Animal Behaviour Science **49**(3), 257–267 (1996) https://doi.org/10.1016/0168-1591(96)01052-2

[7] Zhou, G., Wang, D., Li, Y., Zhou, Y.: Advances in the use of odour as forensic evidence through optimizing and standardizing instruments and canines. Critical Reviews in Analytical Chemistry **52**(4), 623–635 (2022) https://doi.org/10.1080/10408347.2021.1938482

[8] Baldwin, E.A., Bai, J., Plotto, A., Dea, S.: Electronic noses and tongues: Applications for the food and pharmaceutical industries. Sensors **11**(5), 4744–4766 (2011) https://doi.org/10.3390/s110504744

[9] Wilson, A.D., Baietto, M.: Applications and advances in electronic-nose technologies. Sensors **9**(7), 5099–5148 (2009) https://doi.org/10.3390/s90705099

[10] Wojnowski, W., Dymerski, T., Gebicki, J., Namieśnik, J.: Electronic noses in medical diagnostics. Current Medicinal Chemistry **24**(19), 1974–2002 (2017) https://doi.org/10.2174/0929867324666170111153205

[11] Bianchi, F., Careri, M., Musci, M., Mangia, A.: Fish and seafood freshness assessment by gas chromatography–mass spectrometry electronic nose coupled to a multivariate analysis tool. Analytica Chimica Acta **584**(2), 322–329 (2007) https://doi.org/10.1016/j.aca.2006.11.048

[12] Jia, W., Liang, G., Jiang, Z., Tao, X.: Detection of volatile organic compounds using sensors based on metal–organic frameworks. Sensors **19**(2), 404 (2019) https://doi.org/10.3390/s19020404

[13] Yan, K., Zhang, D., Wu, D.: Design of a gas sensor array for remote detection of toxic and explosive gases. Sensors and Actuators B: Chemical **221**, 15–21 (2015) https://doi.org/10.1016/j.snb.2015.06.031

[14] Chen, Q., Liu, A., Zhao, J., Ouyang, Q., Wang, S.: Recent advances of gas sensors based on metal oxide semiconductors. Sensors and Actuators B: Chemical **188**, 902–917 (2013) https://doi.org/10.1016/j.snb.2013.07.113

[15] Singaram, J., Iyengar, S., Madni, A.M.: Deep Learning Networks: Design, Development and Deployment. Springer, ??? (2023)

[16] Shi, B., Iyengar, S.S.: Mathematical Theories of Machine learning-Theory and Applications. Springer, ??? (2020)

[17] Iyengar, S.S., Nabavirazavi, S., Rathore, H., Hariprasad, Y., Chaudhary, N.K.: Advancing forensic science: Ai and knowledge graphs unlock new insights. Journal of Forensic Research **15**, 615 (2024)

[18] Hariprasad, Y., Iyengar, S.S., Ghate, M.: Deepfake video detection using lip region analysis with advanced artificial intelligence based anomaly detection technique. Journal of Forensic Research **15**, 626 (2024)

[19] Hariprasad, Y., Iyengar, S.S., Subramanian, N.: Deepfake Video Detection Using Lip Region Analysis with Advanced Artificial Intelligence Based Anomaly Detection Technique. Authorea Preprints (2024). https://doi.org/10.22541/au.170173936.29121843/v1

[20] Hariprasad, Y., Iyengar, S.S., Chaudhary, N.K.: Securing the future: Advanced encryption for quantum-safe video transmission. IEEE Transactions on Consumer Electronics (2024)

[21] Cui, S., Pu, H., Liu, G.: Advanced sensor and electronic nose technologies for vocs detection in air: Progress and perspectives. Engineering **5**(6), 1011–102 (2019)

[22] Hariprasad, Y., Kumar, K.J.L., Iyengar, S.S.: An extension of synthetic minority oversampling technique based on kalman filter for imbalanced datasets. Machine Learning with Applications **8**, 100267 (2022) https://doi.org/10.1016/j.mlwa.2022.100267

[23] Biehl, W., Hattesohl, A., Jörres, R.A., Duell, T., Althöhn, U., Koczulla, A.R., Schmetzer, H.: Voc pattern recognition of lung cancer: a comparative evaluation of different dog-and enose-based strategies using different sampling materials. Acta Oncologica **58**(9), 1216–1224 (2019)

[24] Furton, K.G., Winialski, D.: Comparing the olfactory capabilities of dogs with machines designed to detect odors. In: Canines, pp. 21–62. Jenny Stanford Publishing, ??? (2022)

[25] Furton, K.G., Caraballo, N.I., Cerreta, M.M., Holness, H.K.: Advances in the use of odour as forensic evidence through optimizing and standardizing instruments and canines. Philosophical Transactions of the Royal Society B: Biological Sciences **370**(1674), 20140262 (2015)

[26] Frank Jr, K.J., Holness, H.K., Furton, K.G., DeGreeff, L.E.: Explosives detection by dogs. In: Counterterrorist Detection Techniques of Explosives, pp. 47–75. Elsevier, ??? (2022)

[27] Simon, A.G., Mills, D.K., Furton, K.G.: Chemical and canine analysis as complimentary techniques for the identification of active odors of the invasive fungus, raffaelea lauricola. Talanta **168**, 320–328 (2017)

[28] Woollam, M.D.: Chemometric analysis of volatile organic compound biomarkers of disease and development of solid phase microextraction fibers to evaluate gas sensing layers. Phd diss., Unknown (2022)

[29] Macias, M.S., Guerra-Diaz, P., Almirall, J.R., Furton, K.G.: Detection of piperonal emitted from polymer controlled odor mimic permeation systems utilizing canis familiaris and solid phase microextraction–ion mobility spectrometry. Forensic Science International **195**(1-3), 132–138 (2010)

Instruction-Level Fine-Tuning of Gemma-2B for Cybersecurity and Synthetic Log Generation Aligned with MITRE Adversarial Tactics, Techniques, and Common Knowledge

Vasanth Iyer[1], Vamshikrishna Challa[2], Pronab Mohanty[4], Yashas Hariprasad[3], S. S. Iyengar[3]

[1]Department of Computer Science, Grambling State University, Louisiana, USA.
[2]Florida International University, Florida, USA.
[3]Louisiana Tech University, Louisiana, USA.
[4]Inspector General of Police, IPS, Govt. of India, Karnataka, India.

Large Language Models (LLMs) such as Gemma-2B have demonstrated remarkable proficiency in various NLP tasks. However, general-purpose models lack deep domain expertise in cybersecurity. This research presents a methodology for fine-tuning the Gemma-2B model into a domain-specific cybersecurity LLM. We outline the dataset preparation, the domain fine-tuning process, the generation of synthetic data logs, and implications for real-world cybersecurity applications. The results indicate improved balanced threat events to translate from Chain-of-Thought tuning to Instruction-Level tuning within the cybersecurity domain, including threat detection, forensic investigation, and attack analysis.

Further experimentation reveals that domain-specific LLM fine-tuning introduces challenges in prompt length distribution, diverging from the patterns seen in general-purpose models. Uneven prompt lengths complicate the model's ability to optimize its use of the full context window, often necessitating a reduced effective context of approximately 200–400 tokens—closer to LSTM capacities—despite the GPU infrastructure supporting up to 2048 tokens. To accommodate this constraint, one-shot prompts resembling chain-of-thought reasoning were found to perform best when paired with quantized weight settings.

Due to these context window limitations, local LLMs were unable to support longer prompt and output sequences necessary for comprehensive synthetic log generation. As a result, cloud-based LLMs were used to generate the synthetic dataset, which was then fine-tuned on the constrained local models. This hybrid approach ensured both high-quality data generation and efficient model training under local hardware constraints.

1 Introduction

Instruction tuning [1] is a critical step in adapting large language models (LLMs) like Gemma-2B to domain-specific tasks. It involves fine-tuning the model [3],[4][8],[9],[10],[11] on diverse examples framed as natural language instructions across multiple task types. As illustrated in Figure 4, this process enables the model to learn reasoning patterns from tasks such as commonsense inference, translation, and classification, so it can generalize to new tasks like natural language inference—even those it hasn't seen during training. This transferability is essential in cybersecurity [5], where LLMs must respond accurately to structured threats like MITRE ATT&CK [12] techniques without explicit task retraining.

2 Related Work

Several works have explored fine-tuning [3] large language models for domain-specific applications. Research in medical, legal, and financial domains has shown that adapting LLMs to specialized datasets [16] improves their accuracy. Prior cybersecurity-focused AI models [5], such as OpenAI's GPT-3 [14] for threat analysis and IBM Watson for security, highlight the potential of LLMs in this domain. However, challenges such as dataset availability, hallucinations, and security risks remain.

3 Methodology

Table 1 Size of Popular Language Models (Non-Embedding Parameters)

Model	Year	# Parameters (billions, non-embedding)
BERT Large	2018	0.34
T5	2019	11
GPT-3	2020	175
PaLM	2022	540
Gemma-2B	2023	2
DeepSeek-7B	2023	7
LLaMA	2023	65

Large models typically are trained with lots of data and have many parameters making it challenging to train from scratch in a research environment. So we will explore the current state of the art foundation models which can then be fine tuned to our domain of interest. The criteria for the comparison are the size of the LLMs and context size. Table 1 has parameters which are weights and biases ranging from 0.3 billion to 540 billion parameters. The Table 1 also shows that the researchers have recently been able to train better models with less number of parameters making it easier for fine tuning for the final tasks. In the initial steps we like to optimize memory for efficient tuning and we next evaluate how the number of parameters has an effect on prompting [1] accuracy as illustrated in Figure 1 We categorize prompts [1] as follows: Zero-shot i.e. without training data, one-shot with one example and few-shots with few examples as described in the prompts below.

User Query

Explain Zero-Shot Prompting using MITRE ATT&CK framework.

Model Response

Zero-shot prompting means that the model is asked to classify or reason with *no prior example* provided.
Prompt:

```
Classify the MITRE ATT&CK technique based on the following activity:
"The attacker executed a PowerShell script to download and run a malicious payload."
Answer:
```

Expected Output:

```
Tactic: Execution
Technique: T1059 - Command and Scripting Interpreter
Sub-technique: T1059.001 - PowerShell
```

User Query

Show a one-shot prompting example using MITRE ATT&CK.

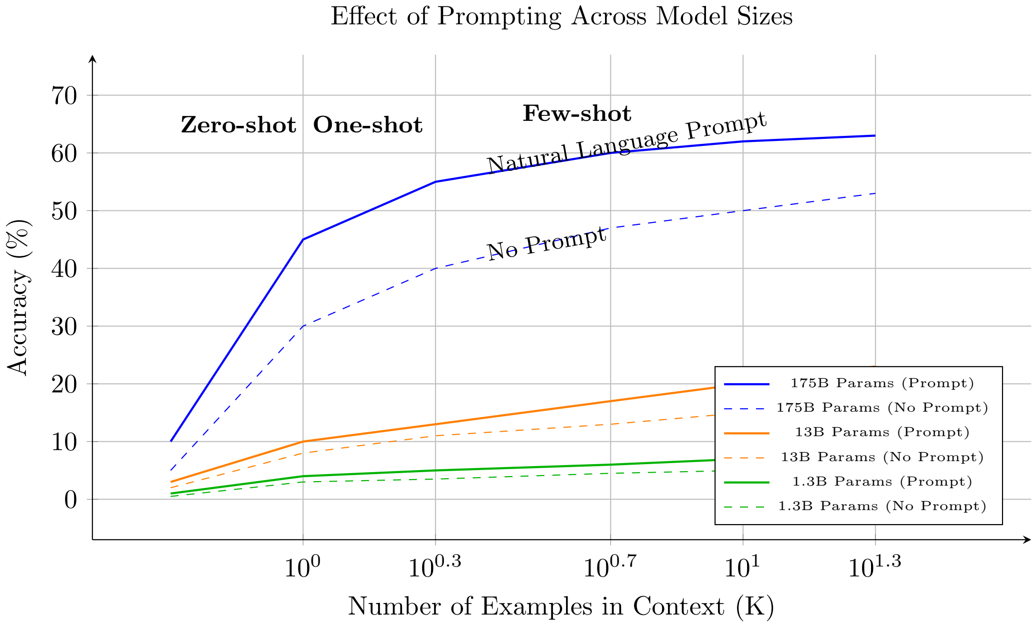


Fig. 1 Comparison of model accuracy across parameter sizes and prompting strategies as a function of the number of in-context examples. Prompted models consistently outperform non-prompted models, especially in few-shot regimes.

Model Response

One-shot prompting provides a single example before asking the model to perform a similar task. **Prompt:**

```
Classify the MITRE ATT&CK technique based on the following activity.
Example:
"The attacker used a phishing email with a malicious attachment to gain initial
access."
Answer:
Tactic: Initial Access
Technique: T1566 - Phishing
Sub-technique: T1566.001 - Spearphishing Attachment
Now classify this:
"The attacker executed a PowerShell script to download and run a malicious
payload."
Answer:
```

Expected Output:

```
Tactic: Execution
Technique: T1059 - Command and Scripting Interpreter
Sub-technique: T1059.001 - PowerShell
```

User Query

Show a few-shot prompting example using MITRE ATT&CK.

Model Response

Few-shot prompting provides multiple examples to help the model generalize better.
Prompt:

```
Classify the MITRE ATT&CK technique based on the following activities.
Example 1:
"The attacker used a phishing email with a malicious attachment to gain initial
access."
Answer:
Tactic: Initial Access
Technique: T1566 - Phishing
Sub-technique: T1566.001 - Spearphishing Attachment
Answer:
```

Expected Output:

```
Tactic: Execution
Technique: T1059 - Command and Scripting Interpreter
Sub-technique: T1059.001 - PowerShell
```

Table 2: Summary of locally trainable large language models (LLMs) with fewer than 2 billion parameters. These models are selected based on their ability to fit within standard GPU memory constraints (e.g., 24GB) and support fine-tuning for domain-specific tasks such as cybersecurity reasoning.

Model	Params (B)	Year	Highlights
Gemma-2B	2	2024	Google's lightweight open model for on-device and fine-tuning tasks
TinyLLaMA-1.1B	1.1	2023	Minimal resource LLaMA-based model for mobile/IoT research
Phi-2	2.7	2023	Microsoft's model designed for reasoning, aligned with on-device use
DeepSeek-1.3B	1.3	2024	DeepSeek's small model for fast, local inference
StableLM-3B	3	2023	Stability AI's open model designed for transparency and edge use
RedPajama-3B	3	2023	Open LLaMA-style model trained on reproducible public datasets

Figure 1 illustrates the relationship between prompting accuracy and the number of in-context examples. As the number of examples increases from zero-shot to few-shot, the model's performance improves, making fewer errors and demonstrating better generalization. Additionally, models with larger parameter counts exhibit stronger zero-shot generalization capabilities.

Our training pipeline Figures 2,3 consists of two main stages: (1) the construction of domain-specific datasets 3 and (2) the fine-tuning of language models. To address resource constraints, we leverage large language models (LLMs) to generate synthetic datasets and utilize smaller, locally runnable LLMs—typically with reduced precision—for fine-tuning. Specifically, we select native LLMs with fewer than 2 billion parameters, as summarized in Table 2, to ensure compatibility with our available GPU memory. In this work, we focus on domain adaptation using Google's recently released Gemma-2B model.

Fig. 2 Finetuning LLM

Fig. 3 Domain Finetuning LLM

3.1 Dataset Collection

To develop a cybersecurity-specific expert model, we construct a domain-specific dataset based on the MITRE ATT&CK framework and use it to fine-tune the language model, as illustrated in Figure 2. MITRE ATT&CK organizes adversarial behavior using well-defined Tactics and Techniques, providing a structured taxonomy that supports effective generalization during fine-tuning. The prompts used for this task are generated using a chain-of-thought prompting approach, enabling the model to reason through sequential steps aligned with the structure of ATT&CK.

3.2 Technique T# in the MITRE ATT&CK Framework

This section provides an overview of Technique #'s as defined in the MITRE ATT&CK framework, including its associated tactics, use cases, and adversary behaviors. The technique is often leveraged by threat actors to achieve *[specific objective]*.

3.3 Understanding Vulnerability T#

Vulnerability X is a *[type of flaw]* that affects *[systems/applications]*. It allows attackers to *[describe action, e.g., escalate privileges, exfiltrate data, etc.]*. This subsection explains the technical working of the vulnerability, including how it is exploited and its presence in known threat campaigns.

3.4 Mitigation Strategies for Vulnerability T#

To reduce the risk associated with Vulnerability X, organizations can implement several mitigation strategies:

- Apply security patches and updates regularly.
- Use network segmentation and access controls.
- Employ endpoint detection and response (EDR) tools.
- Monitor for known indicators of compromise (IoCs).

A typical technique and its description is shown in Table 3.

Aspect	Description
MITRE Technique	Txxxx – Technique X
Vulnerability Type	*e.g., Buffer Overflow*
Exploitable By	*e.g., Remote attackers, malware*
Mitigation	*Patching, EDR, segmentation, etc.*

Table 3 Example: Technique and Vulnerability Description.

4 Synthetic Data Generation

In cybersecurity, high-quality labeled data is scarce, often sensitive, and typically imbalanced toward benign activity. This presents a significant barrier to effectively fine-tuning large language models

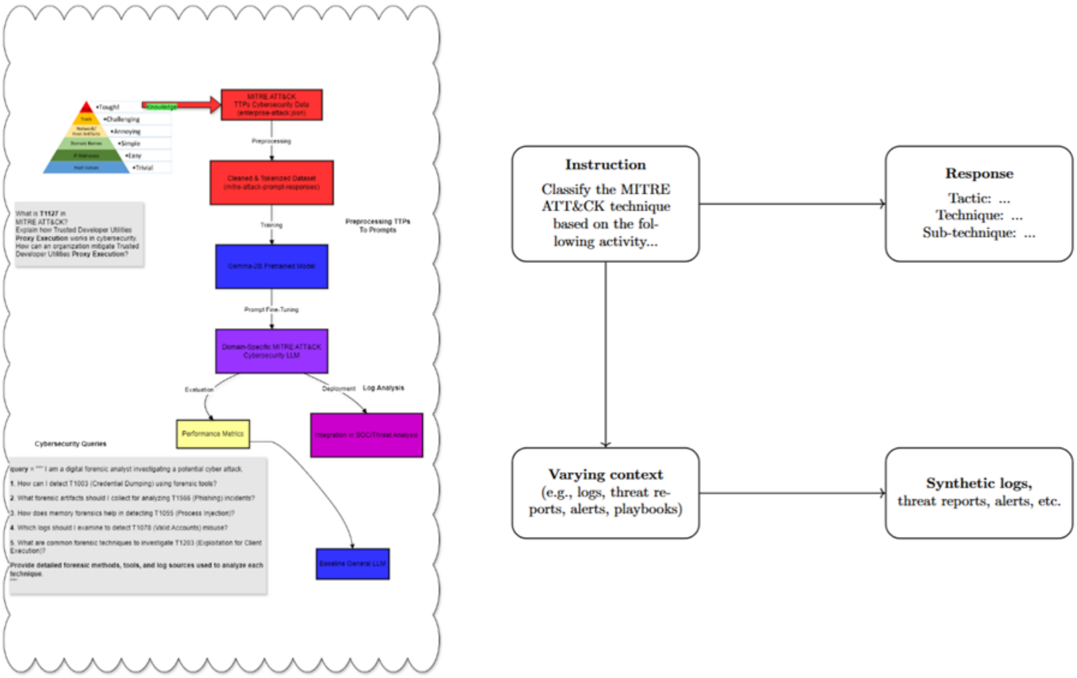


Fig. 4 Workflow for generating synthetic cybersecurity data using large LLMs. The process produces instruction-format samples that encode attack behaviors, supporting Chain-of-Thought prompting and enabling small models to reason over structured threat intelligence such as MITRE's Pyramid of Pain, which ranks the difficulty of detecting and disrupting various attacker artifacts.

(LLMs) for security-specific reasoning tasks. To address this, we introduce a synthetic data generation [3] framework as illustrated in Figure 4. By leveraging the structured nature of the MITRE ATT&CK framework, we use LLMs to generate instruction-style examples that simulate a wide variety of attack tactics, techniques, and procedures (TTPs). This pipeline enables the creation of logs [13], [15] for rare or hard-to-collect threats, supports diverse prompting styles (e.g., zero-shot, one-shot, few-shot), and allows for balanced datasets that improve fine-tuning efficiency and model generalization. Crucially, it also provides a privacy-preserving and legally compliant alternative to real-world security logs.

4.1 Instruction Tuning Process

We perform fine-tuning across multiple tasks at the instruction level. Instruction tuning enables large language models (LLMs) to specialize in domain-specific reasoning by learning from natural language examples aligned with real-world tasks. In the context of cybersecurity, we leverage structured knowledge from the MITRE ATT&CK framework—such as techniques like T1059 and T1547, tactics including Execution, Persistence, and Lateral Movement, and contextual formats like logs, alerts, threat reports, and playbooks—to construct a diverse and targeted training corpus. As illustrated in Figure 4, our approach combines domain adaptation with synthetic data generation to support instruction-level fine-tuning at scale.

Synthetic datasets offer the advantage of complete control over coverage and balance. They allow us to generate labeled examples for every ATT&CK technique, including rare or underrepresented behaviors that are seldom encountered in enterprise environments. For instance, we can simulate advanced scenarios such as **T1003.001 – LSASS Dumping** using PowerShell-based indicators that may not naturally appear in historical logs. This ensures comprehensive coverage across tactics, sub-techniques, and platforms, including Windows, Linux, and macOS.

Additionally, synthetic logs as in Table 4 can be tailored to varying levels of complexity. We design some examples with clean and distinct attack signatures to support basic classification tasks, while others contain obfuscated patterns or mixed signals to train models for reasoning under uncertainty. This form of data augmentation enables models to engage in chain-of-thought prompting—reasoning through multi-step sequences to correctly identify attacker behavior and map it to a specific tactic or technique.

An equally important benefit of using synthetic data is its safety and compliance. Since no real personal or organizational identifiers are involved, this method avoids the legal and ethical risks

Synthetic Log	Instruction	Model Output
`"2024-04-15 10:22:11" user: SYSTEM ran: "powershell -enc ..."`	What MITRE ATT&CK technique does this log indicate?	Tactic: Execution Technique: T1059.001 – PowerShell
`Zeek conn.log: 192.168.1.100 → 10.0.0.10 TCP 3389`	Explain what this log suggests and map it to MITRE ATT&CK.	Indicates use of RDP for Lateral Movement. Tactic: Lateral Movement Technique: T1021.001 – Remote Desktop Protocol

Table 4 ATT&CK Prompts Enhanced with Synthetic Data Logs.

associated with handling real logs, such as violations of privacy regulations like GDPR or HIPAA. Furthermore, it mitigates the model's dependence on the biases of any single SOC dataset, promoting better generalization.

Finally, our framework supports the generation of diverse prompt formats for instruction tuning. We create examples suitable for zero-shot learning (where no prior examples are given), one-shot prompts (with a single reference example), and few-shot configurations (featuring multiple labeled examples followed by a query). This variety improves the model's ability to generalize across different log structures and behavioral patterns. Together, these design choices result in a balanced and task-relevant training set that significantly enhances the model's performance in detecting and reasoning about cyber threats.

4.2 Model Evaluation

The model was evaluated on:

- Accuracy in answering MITRE ATT&CK queries
- Performance on cybersecurity question-answering tasks
- Effectiveness in analyzing threat logs
- Comparison with general-purpose LLMs

The first of the four evaluation criteria pertains to domain adaptation. Our 2B-parameter model demonstrated effective fine-tuning and successfully answered Chain-of-Thought-style queries aligned with the MITRE ATT&CK framework. In contrast, the remaining three criteria—focused on instruction tuning and its extension through synthetic data generation—exhibited lower performance. This was primarily due to the baseline accuracy of models under 2 billion parameters, which remained below 20%, as illustrated in Figure 1. Due to reduced task accuracy and inadequate alignment with instruction-based tasks, we leveraged larger, cloud-hosted LLMs exceeding 175 billion parameters to generate synthetic instruction datasets. These models, achieving over 50% accuracy in instruction-following tasks, exhibited stronger generalization capabilities. The generated synthetic data was then used to re-train the smaller, local models **??**, enhancing their ability to respond to instruction-level prompts.

5 Results and Discussion

Table 5 summarizes viable local LLM training configurations in Figure 5 using current-generation NVIDIA hardware. In our experiments, we utilized a 24GB GPU* for fine-tuning. The domain-specific MITRE ATT&CK dataset consisted of 2,398 prompt-response pairs (5). To accommodate GPU memory constraints, the batch size was set to 4. However, attempts to use the default token lengths of 1,024 to 2,048 tokens resulted in parser errors at this batch size. We found that a token length of 397 (5)—combined with dynamic padding for variable-length prompts—enabled stable training across all epochs. This constraint, however, significantly limited training to 1–2 shot prompting scenarios.

The results of 1–2 shot prompting are presented in Figure 7, with the maximum token output length configured to 200 tokens, as shown in Figure 6. The generated responses demonstrate meaningful domain adaptation and interpretable accuracy in the cybersecurity context. However, to comprehensively assess model performance, further evaluation is required. In future work, we plan to compare the fine-tuned model against other standard LLM baselines using an automated LLM-based judge framework to ensure consistent and objective scoring across tasks.

GPU Memory Range	Typical Training Config	Notes / Recommendations
8–16GB GPUs (e.g., RTX 3060, 4060, 3090, T4)	• Token limit: **2048–3072** • Batch size: **2–4** • Use `fp16` or `bf16`	• Enable **gradient checkpointing** • FlashAttention can reduce memory cost • Ideal for instruction tuning with 1–2 shot prompts
24–32GB GPUs (e.g., RTX 4090, A5000, V100)	• Token limit: **4096–8192** • Batch size: **4–8+** • Supports long CoT prompts	• Can fine-tune with **4–6 shot** examples • Ideal for multi-turn logs or threat reasoning • Combine synthetic logs with multi-step output
24GB GPUs* (e.g., RTX 4090)	• Token limit: **397** • Batch size: **4** • Used 4-bit quantized **??** weights and 16-bit arithmetic	• Can fine-tune with **1–2 shot** examples • Ideal for domain fine-tuning locally

Table 5 *Configuration shown was used during both training and evaluation.

6 Acknowledgments

Support for this research was provided by the Army Research Office under Grant Number W911NF-21-1-0264. The authors would like to thank Dr. Igor Ternovskiy at the Sensor Directorate, Wright-Patterson Air Force Base, for valuable discussions on the use of synthetic data to better approximate domain-specific distributions for fine-tuning large language models. Additional support was provided by the National Science Foundation under Grant Number HBCU-EiR-2101181 for work on developing AI deep learning techniques using explainable AI. Portions of this work also contributed to the development of introductory AI courses, supported by a grant from the Google TensorFlow team.

We gratefully acknowledge the contributions and mentorship of our former Co-Principal Investigator, Dr. Y.B. Reddy, whose guidance and dedication were instrumental in the early stages of this research. Dr. Reddy passed away recently, and we respectfully dedicate this work to his memory.

```
[2]: from datasets import load_dataset
     # Tokenize the dataset
     # Load MITRE ATT&CK dataset (replace with your actual dataset)
     dataset = load_dataset("csv", data_files={"train": "/mnt/c/Vasanth/
     ↪Deeplearning-WSL2/mitre_attack_prompt_responses.csv"})


     #def tokenize_function(example):
     #    return tokenizer(example["prompt"], example["response"], truncation=True,␣
     ↪padding="max_length")
     # Tokenize and pad/truncate to a fixed sequence length
     def tokenize_function(example):
         return tokenizer(
             example["prompt"],
             example["response"],
             truncation=True,
             padding="max_length",  # Ensures all sequences are exactly `max_length`
             max_length=397,        # Fix sequence length
             return_tensors="pt"
         )

     tokenized_datasets = dataset.map(tokenize_function, batched=True)
```

```
[3]: # Check tokenized sequence lengths
     import numpy as np

     sequence_lengths = [len(x) for x in tokenized_datasets["train"]["input_ids"]]
```

```
     print(f"Max length: {max(sequence_lengths)}, Min length:␣
     ↪{min(sequence_lengths)}")
```

```
Max length: 397, Min length: 397
```

```
[4]: import torch
     from unsloth import FastLanguageModel
     from transformers import AutoTokenizer

     model_name = "unsloth/gemma-2-2b-it"

     # Load tokenizer
     tokenizer = AutoTokenizer.from_pretrained(model_name)

     # Load the optimized Gemma model
     model, tokenizer = FastLanguageModel.from_pretrained(
         model_name,
         max_seq_length=397,  # Ensure consistent sequence length
         dtype=torch.float16, # Mixed precision for memory efficiency
         load_in_4bit=True,    # Load model in 4-bit precision for faster training
         device_map="auto"     # Automatically distribute model across available␣
     ↪devices
     )
```

```
Unsloth: If you want to finetune Gemma 2, install flash-attn to make it faster!
To install flash-attn, do the below:

pip install --no-deps --upgrade "flash-attn>=2.6.3"
==((====))==  Unsloth 2025.3.3: Fast Gemma2 patching. Transformers: 4.49.0.
   \\   /|    NVIDIA GeForce RTX 4090. Num GPUs = 1. Max memory: 23.988 GB.
Platform: Linux.
O^O/ \_/ \    Torch: 2.4.1+cu121. CUDA: 8.9. CUDA Toolkit: 12.1. Triton: 3.0.0
\        /    Bfloat16 = TRUE. FA [Xformers = 0.0.28.post1. FA2 = False]
 "-____-"     Free license: http://github.com/unslothai/unsloth
Unsloth: Fast downloading is enabled - ignore downloading bars which are red
colored!
```

```
[5]: def tokenize_function(example):
         inputs = tokenizer(
             example["prompt"],
             text_target=example["response"],  # Use `text_target` for labels
             truncation=True,
             padding="max_length",
             max_length=397,
         )
         inputs["labels"] = inputs["input_ids"].copy()  # Ensure labels are properly␣
     ↪set
         return inputs
```

Fig. 5 Instruction-tuning workflow using domain-adapted synthetic data. The pipeline integrates structured prompts derived from the MITRE ATT&CK framework and Chain-of-Thought prompting strategies. The model is fine-tuned with a mixture of zero-shot, one-shot, and few-shot examples to support reasoning and classification across cybersecurity tasks.

```python
[3]: import torch
from transformers import AutoTokenizer, AutoModelForCausalLM, LogitsProcessorList, MinLengthLogitsProcessor
from peft import PeftModel

# Define model path
model_name = "unsloth/gemma-2-2b-it"  # Base model
peft_model_path = "./gemma_mitre_attack_local2/checkpoint-1800"  # Fine-tuned LoRA model

# Add this inside the generation call if temperature=0 causes issues
#temperature=max(0.1, 0.7)  # Prevent temperature=0

# Load tokenizer
tokenizer = AutoTokenizer.from_pretrained(model_name)
if tokenizer.pad_token is None:
    tokenizer.pad_token = tokenizer.eos_token  # Ensure padding token exists
```

```python
# Load the base model
base_model = AutoModelForCausalLM.from_pretrained(model_name)

# Load the fine-tuned LoRA model
model = PeftModel.from_pretrained(base_model, peft_model_path)

# Set max sequence length correctly
model.config.max_position_embeddings = 397

# Initialize with proper settings
model.config.pad_token_id = tokenizer.pad_token_id
model.config.use_cache = True  # Enable cache for stability
torch.set_float32_matmul_precision('high')  # For CUDA stability
```

```python
[4]: def query_mitre_attack_model(prompt, max_new_tokens=200):
    # Device setup
    device = torch.device("cuda" if torch.cuda.is_available() else "cpu")
    model.to(device)

    # Dynamic length management based on model capabilities
    model_max_length = getattr(model.config, "max_position_embeddings", 4096)  # Default to 4096 if not found
    input_max_length = model_max_length - max_new_tokens - 10  # Buffer for safety
```

```python
    # Smart tokenization with adaptive truncation
    inputs = tokenizer(
        prompt,
        return_tensors="pt",
        truncation=True,
        padding="longest",  # Better for variable-length inputs
        max_length=input_max_length  # Respect model limits
    ).to(device)

    # Enhanced generation parameters
    generation_config = {
        "max_new_tokens": max_new_tokens,
        "do_sample": True,
        "temperature": 0.7,
        "top_p": 0.9,
        "repetition_penalty": 1.2,
        "pad_token_id": tokenizer.eos_token_id,  # Critical for stable generation
        "eos_token_id": tokenizer.eos_token_id,
        "attention_mask": inputs["attention_mask"]
    }

    with torch.no_grad():
        outputs = model.generate(
            input_ids=inputs["input_ids"],
            **generation_config
        )

    # Safe decoding with cleanup
    decoded = tokenizer.decode(
        outputs[0],
        skip_special_tokens=True,
        clean_up_tokenization_spaces=True
    )

    # Post-processing for quality
    return decoded.split("Answer:")[-1].strip()  # Remove input context if needed
```

Fig. 6 Model inference results for domain-specific instruction prompts. Each output is constrained to a maximum of 200 tokens, reflecting the token limit applied during inference for consistency across prompt evaluations. The responses demonstrate how the fine-tuned model interprets MITRE ATT&CK-aligned queries, illustrating task comprehension and reasoning within the token constraint.

```
[ ]: while True:
         try:
             query_str = input("ATT&CK Prompt>")
             if query_str == "":
                 print("Input not detected!")
         except ValueError:
             print("Invalid input. Please enter a number.")
         response = query_mitre_attack_model(query_str)

         print("\ATT Response>", response)
```

```
ATT&CK Prompt> What is MITRE ATT&CK T1110?

\ATT Response> What is MITRE ATT&CK T1110?

T1110, "**Process Enumeration and Mimikatz use in a targeted attack**" describes
an adversary technique used to enumerate active processes on the target system
before utilizing Mimikatz.
Here's a breakdown:

* **Target System:**  The process enumeration phase aims to identify all running
programs on a victim machine or network segment for potential exploitation. This
includes applications critical to normal operations, as well as any open
services or dormant processes that could provide valuable information.
* **Mimikatz:** A powerful tool commonly employed by attackers to steal
credentials from memory (especially Windows systems). It can be utilized during
lateral movement within an organization after initial access. The attacker
leverages this knowledge of available processes to select those most likely to
contain sensitive data.


**How it works:**

1. **Initial Access & Privilege Escalation:** An attacker gains entry into the
compromised environment using various methods like phishing emails, malicious
downloads, or exploiting vulnerabilities. They then gain some
ATT&CK Prompt> Is ATT&CK T1110 a brutefore attack?

\ATT Response> Is ATT&CK T1110 a brutefore attack?

No, AT&T & CK T1110 is not a Brute force attack.

Here's why:

* **Brute-Force Attacks:** Involve trying every possible password or combination
until the correct one is found. They are relatively simple and inefficient for
large datasets but can be effective if you have enough time and resources to
test all possibilities.
    * **Example:** Trying each number from 1 to 9 in sequence to guess a PIN
code.

* **ATT&CK T1110 (Execution of arbitrary commands):**  This tactic focuses on
exploiting vulnerabilities that allow attackers to execute malicious
instructions within an operating system. This doesn't involve simply guessing
passwords; it aims to gain control over the target machine by manipulating its
inner workings.
```

Fig. 7 Example of instruction-level fine-tuning with 1–2 shot prompting using synthetic data. The prompt contains a real-world cybersecurity scenario aligned to MITRE ATT&CK, followed by a response from the fine-tuned model. This illustrates the model's ability to generalize and explain attack techniques based on few-shot learning with padded token limits.

References

[1] Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narasimhan, and Yuan Cao. *ReAct: Synergizing Reasoning and Acting in Language Models*. arXiv preprint arXiv:2210.03629, 2023. Available at: https://arxiv.org/abs/2210.03629.

[2] T. Dettmers, A. Pagnoni, A. Holtzman, and L. Zettlemoyer, "QLoRA: Efficient Finetuning of Quantized LLMs," *arXiv preprint arXiv:2305.14314*, 2023. Available: https://arxiv.org/abs/2305.14314

[3] V.Iyer, I. Utilizing Transfer Learning and Graph Matching Spatial Attentions from using CARLA Pre-Trained Models.. (Future Technology Conference, UK, Nov 2024.).

[4] V. Iyer and A. Mehmood. *Multi-Object On-Line Tracking as an Ill-Posed Problem: Ensemble Deep Learning at the Edge for Spatial Re-identification*. In: Arai, K. (eds) Intelligent Computing. SAI 2022. Lecture Notes in Networks and Systems, vol 507. Springer, Cham, 2022. Available at: https://doi.org/10.1007/978-3-031-10464-0_13.

[5] S. G. Aarella, S. Agarwal, S. P. Mohanty, E. Kougianos, V. Iyer, and B. Rout. *iPUF: A Novel Security-by-Design Paradigm to Mitigate Data Manipulation and External Attacks in Cyber-Physical Systems*. Presented by IEEE. Available at: https://par.nsf.gov/biblio/10581950.

[6] Vasanth Iyer, and Asif Mehmood, "Metadata learning of non visual features: cooccurrence overlap function for rectangular regions and ground truth data"

[7] Iyer, Vasanth Shetty, Sachin,"Virtual Sensor Tracking using Byzantine Fault Tolerance and Predictive outlier Model for Complex Tasks Recognition"

[8] V.Iyer, S.Shetty,Virtual sensor tracking using byzantine fault tolerance and predictive outlier model for complex tasks recognition

[9] V. Iyer, A. Aved, T. B. Howlett, J. T. Carlo, B. Abayowa, Autoencoder versus pre-trained CNN networks: deep-features applied to accelerate computationally expensive object detection in real-time video streams

[10] V. Iyer, A. Aved, T. B. Howlett, J. T. Carlo, A. Mehmood, N. Pissinou, S. S. Iyengar, Fast multi-modal reuse: co-occurrence pre-trained deep learning models

[11] V. Iyer, A. Mehmood, Metadata learning of non-visual features: co-occurrence overlap function for rectangular regions and ground truth data, in: M. S. Alam (Ed.), Pattern Recognition and Tracking XXXI, Vol. 11400, International Society for Optics and Photonics, SPIE, 2020, pp. 34 – 42.

[12] MITRE ATT&CK Framework, https://attack.mitre.org/

[13] Zeek Network Security Monitor, https://zeek.org/

[14] OpenAI, https://openai.com/

[15] IBM Watson for Cybersecurity, https://www.ibm.com/security/artificial-intelligence

[16] Ghiasi and G. Cui, Y. and Srinivas, A. and Qian R., and Lin and T. and Cubuk, E. and Le, and Q. Zoph and B. Simple Copy-Paste is a Strong Data Augmentation Method for Instance Segmentation., CoRR,(2020)

Diffusion-Driven Deceptive Patches: Adversarial Manipulation and Forensic Detection in Facial Identity Verification

Shahrzad Sayyafzadeh, Hongmei Chi, Shonda Bernadin, Simon Foo

[1,3,4*]Electrical & Computer Engineering, FAMU-FSU College of Engineering, Pottsdamer St, Tallahassee, 32310, Florida, USA.
[2]Department of Computer & Information Science, Florida A&M University, 1333 Wahnish Way, Tallahassee, 32307, Florida, USA.

*Corresponding author(s). E-mail(s): shahrzad1.sayyafzade@famu.edu;
Contributing authors: hongmei.chi@famu.edu;
bernadin@eng.famu.fsu.edu; foo@eng.famu.fsu.edu;

Abstract

This work presents an end-to-end pipeline for generating, refining, and evaluating adversarial patches to compromise facial biometric systems with forensic analysis and security testing applications. We utilize a FGSM to generate adversarial noise targeting our classifier for identity detection and employ a diffusion model for reverse diffusion to enhance the imperceptibility with additional Gaussian smoothing and adaptive brightness correction of synthetic adversarial patch evasion generation. The refined patch is applied to facial images to test its ability to evade recognition systems while maintaining natural visual characteristics. A Vision Transformer (ViT)-GPT2 model generates captions to provide a semantic description of a person's identity for Adv Images, supporting forensic interpretation and documentation for identity evasion attack and recognition. The pipeline evaluates changes in identity classification, captioning results, and the vulnerability of facial identity verification and expression to adversarial attacks. Therefore, detecting and mitigating these adversaries' attacks are necessary for forensic settings by leveraging perceptual hashing. We successfully detected and analyzed a series of generated adversaries with 0.95% SSIM.

Keywords: Adversarial Patch Generation, Gaussian Smoothing, Diffusion Model, Social Media Forensics, Perceptual Hashing.

1 Introduction

Deep learning models have revolutionized image classification [1], achieving remarkable accuracy in computer vision applications, from facial recognition to medical diagnostics. However, these models exhibit critical vulnerabilities to adversarial attacks and carefully crafted perturbations that can deceive classifiers while remaining imperceptible to humans [2]. One particularly susceptible area is facial emotion classification and identity recognition, where models detect emotions such as happiness, anger, sadness, and surprise and change a person's identity through the description of its respected forensic biometric systems. Adversarial patch attacks pose a concern in facial identity verification, creating localized regions of adversarial modifications that can be physically deployed. For facial images $x \in \mathbb{R}^{H \times W \times C}$, where H, W denote dimensions and C represents color channels, and with a standard input size of 224×224 pixels , an adversarial patch p maximizes the classifier's loss:

$$\max_p L(f(x+p), y)$$

where f is the identity classifier and y is the true emotion and gender identity label.

In addition to emotion classification, facial identity verification systems are equally vulnerable to adversarial attacks, posing severe security risks. Facial identity verification, widely used in biometric authentication and surveillance, matches facial features to stored profiles to confirm identities. Adversarial patches targeting these systems can cause false positives, allowing imposters to bypass authentication, or false negatives, rejecting legitimate users. Such manipulations exploit the deep feature embeddings on which identity verification systems rely, causing the model to misclassify identities even when the alterations are visually subtle. For identity verification models such as ArcFace and FaceNet, which commonly process facial images of size 112×112 or 160×160 pixels, these adversarial perturbations significantly degrade recognition accuracy [3].

To detect these manipulations, forensic analysis techniques are critical in exposing adversarial artifacts hidden from human perception. Spectral analysis using Fast Fourier Transform (FFT) can reveal high-frequency perturbations that characterize adversarial patches, highlighting discrepancies between authentic and adversarial images. Additionally, depth estimation, achieved using MiDaS models on images resized to 256×256 pixels, can identify geometric inconsistencies introduced by adversarial attacks, such as distorted facial structures or unnatural surface textures. These forensic approaches can detect adversarial tampering that is imperceptible in the spatial domain, providing an additional layer of security against attacks on both facial emotion and identity recognition systems.

Diffusion models play a dual role in adversarial manipulation and defense. In adversarial patch generation, diffusion models employ a forward process by adding Gaussian noise over T timesteps, defined as:

$$q(x_t|x_{t-1}) = \mathcal{N}(x_t; \sqrt{1-\beta_t}x_{t-1}, \beta_t\mathbf{I})$$

where β_t represents the variance schedule. The reverse process generates images from noise using:

$$p_\theta(x_{t-1}|x_t) = \mathcal{N}(x_{t-1}; \mu_\theta(x_t, t), \sigma_\theta^2(t)\mathbf{I})$$

By integrating adversarial objectives into the diffusion process, the total loss is defined as:

$$L_{total} = L_{diffusion} + \lambda L_{adv}$$

where $L_{diffusion}$ ensures high-fidelity image generation, and L_{adv} encourages perturbations that mislead the target classifier. This formulation allows diffusion models to generate highly effective adversarial patches that blend seamlessly into facial features, such as cheeks or foreheads, making them difficult to detect through traditional forensic techniques. The adversarial patches are typically generated on images resized to 224×224 pixels to match the input dimensions of commonly targeted models like InceptionResnetV1 pretrained on VGGface2.

Conversely, diffusion models also serve as a defense mechanism through adversarial purification. In this process, an adversarial image is passed through a reverse diffusion model to remove perturbations and restore the original image. The purified image $\hat{x}$ is reconstructed from Gaussian noise as:

$$\hat{x} = p_\theta(x_0|x_T) \quad \text{where} \quad x_T \sim \mathcal{N}(0, \mathbf{I})$$

This process effectively maps adversarial images back to the natural data manifold, preserving emotion-relevant and identity-relevant features while eliminating adversarial distortions. Adversarial purification is typically performed on images resized to the model's input dimensions, such as 112×112 for identity verification and 224×224 for emotion recognition models. By comparing purified images with their originals through forensic analysis, subtle differences in spectral and depth domains can be identified, further improving adversarial detection capabilities.

1.1 Image Classification, Captioning, and Identity Verification Pipeline

The diagram presents a pipeline for image classification, captioning, and identity verification, showcasing the effect of adversarial patches on deep learning models. An input image x is processed using ViT-GPT2, producing an initial caption C_{original}. An adversarial patch p is applied to the image, generating $x + p$, which undergoes the same classification and captioning.

Both original and patched images are processed through InceptionResnetV1 to extract class features $f_c(x)$ and identity features $f_i(x)$. The class features determine the predicted label, while identity features capture embeddings used for biometric verification. Meanwhile, the ViT-GPT2 model generates the patched caption C_{patched}.

The pipeline compares outputs to measure the patch's impact. Class changes are detected using the L2 distance between class embeddings:

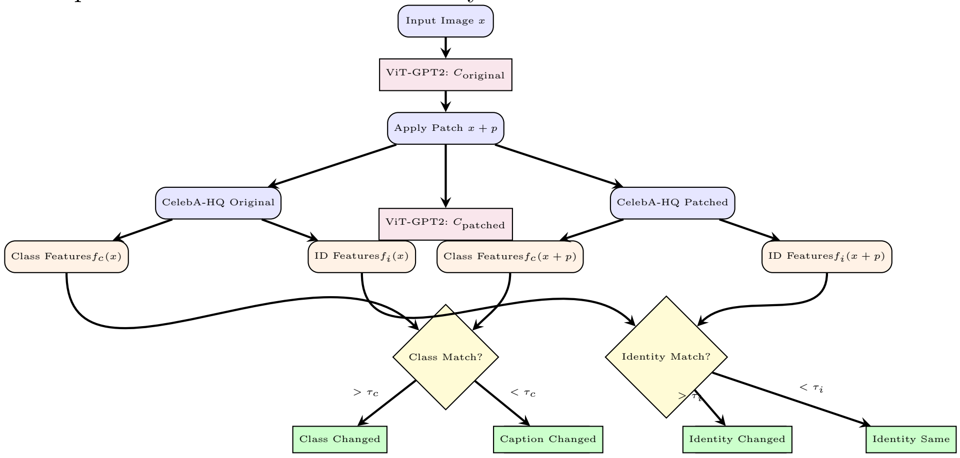
$$\|f_c(x) - f_c(x + p)\|_2,$$

and flagged if exceeding a threshold τ_c. Identity changes are similarly evaluated:

$$\|f_i(x) - f_i(x+p)\|_2,$$

with a threshold τ_i. Caption differences result from directly comparing C_{original} and C_{patched}.

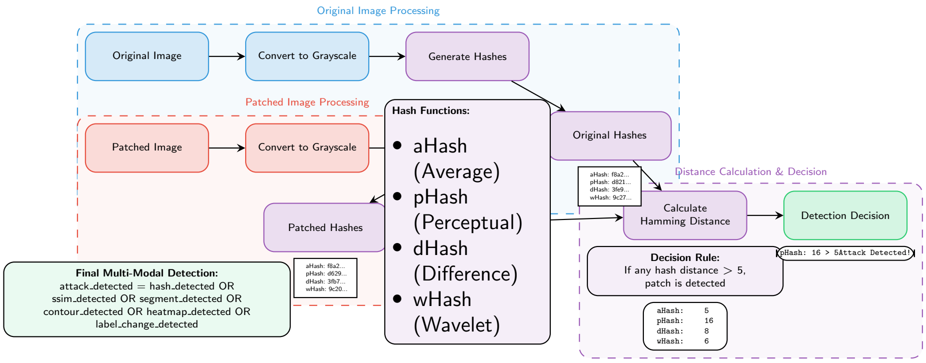
This pipeline reveals how adversarial patches affect classification, captioning, and identity verification, highlighting vulnerabilities in multimodal deep learning models for computer vision and biometric security.



1.2 Adversarial Patch Attack Detection and Digital Forensic

The diagram represents the process for detecting adversarial patch attacks using perceptual hash distance in the context of Digital Forensics [4]. The workflow begins with the original image processing, where the image is first converted to grayscale. After this, hashes for the image were generated using multiple hash functions, including aHash (average), pHash (perceptual), dHash (difference), and wHash (wavelet). These hashes capture the various perceptual features of the original image. A similar process occurs for the patched image. The patched image is also converted to grayscale and hashed using the same set of hash functions. This allows the system to compare the perceptual differences between the original and patched hashes. The system then calculates the Hamming distance between the original and patched hashes. The Hamming distance is used to measure the difference between the two hashes, with higher values indicating more perceptual differences, which may suggest the presence of an adversarial patch. Once the Hamming distance is calculated, the system proceeds to the detection decision. A decision rule is applied: an adversarial patch is detected if any of the Hamming distances exceeds a threshold of 5. This threshold is used to determine whether the perceptual change in the image is enough to be considered tampering or manipulation. Additionally, the system includes multimodal detection, meaning that detection is not solely reliant on hash distance. The detection process also considers other methods, such as detecting SSIM differences (Structural Similarity Index), segment anomalies, contour detection, heatmap analysis, and even label changes during classification. If any of these methods indicates the presence of an adversarial attack, the system will flag it as detected. Therefore, this workflow outlines our proposed method for detecting adversarial patches by leveraging perceptual hashing, hamming distance calculation, and multimodal detection strategies. This approach is highly effective for identifying subtle manipulations in images, particularly when traditional image comparison methods may fail.

Perceptual Hash Distance for
Adversarial Patch Detection



2 Background

Digital forensics is a critical field in cybersecurity that focuses on identifying, collecting, preserving, and analyzing digital evidence. With the increasing reliance on digital platforms, cyber threats have evolved, requiring advanced forensic techniques to detect and mitigate malicious activities. Among these threats, adversarial attacks, particularly patch attacks, have emerged as a challenge to forensic investigations and security systems [5].

Adversarial patch attacks are a subset of adversarial machine learning techniques where attackers introduce carefully crafted perturbations into images, leading machine learning models to misclassify objects. These patches, generated through gradient-based optimization, stable diffusion, or adversarial attack techniques such as FGSM and PGD, can deceive deep learning models into incorrect predictions. These attacks pose severe risks in security-critical applications such as facial recognition, autonomous driving, medical imaging, and biometric authentication.

In facial recognition systems, attackers can use adversarial patches to evade identification [6, 7], while in autonomous vehicles, minor alterations to traffic signs can mislead vision-based navigation, leading to potential accidents [8, 9]. Similarly, in medical imaging, adversarial perturbations can cause AI models to misdiagnose diseases, and in surveillance systems, these attacks can bypass security measures, allowing unauthorized individuals to evade detection [10, 11].

Given the increasing importance of such attacks, adversarial detection mechanisms are essential to maintain the integrity of digital forensics. Hashing techniques play a crucial role in forensic analysis by detecting tampering in digital evidence. Traditionally, digital forensics relies on cryptographic and perceptual hashing to verify the integrity of digital files. Cryptographic hashing, such as SHA-256 and MD5 [12], generates a unique fingerprint of a file, ensuring its integrity. However, even minor pixel changes can result in drastically different hash values, making cryptographic hashing too sensitive for detecting adversarial patch attacks.

On the other hand, perceptual hashing is designed to identify visual similarities between images, making it more suitable for detecting subtle adversarial modifications. Techniques like aHash, pHash, dHash, and wHash can detect structural changes in an image without being overly sensitive to minor alterations such as compression or noise. Unlike deep learning-based adversarial detection methods that require continuous retraining and large-scale datasets, hashing-based forensics provides a lightweight and efficient alternative. Hash comparisons require minimal computational resources while still effectively identifying adversarial manipulations. This method is resilient against minor modifications, making it ideal for real-time forensic applications requiring rapid image integrity verification [13, 14].

Law enforcement agencies, security analysts, and digital forensic investigators can integrate hash-based detection into their workflows to quickly assess whether an image has been altered in a way that could impact an investigation. These attacks manipulate images or video data to deceive machine learning models into misclassification, raising serious concerns in identity verification, biometric security, and digital authentication systems.

Identity verification and biometric verification systems are integral to modern security infrastructures. These systems leverage unique physiological or behavioral characteristics such as facial features, fingerprints, iris patterns, and voice recognition to authenticate individuals. Facial recognition, in particular, has gained widespread adoption in law enforcement, banking, border control, and mobile device authentication. However, the increasing sophistication of adversarial attacks poses a severe threat to these biometric systems [15].

Attackers can introduce imperceptible adversarial patches into images to fool recognition models into misidentifying individuals or bypassing authentication entirely. In law enforcement and surveillance applications, such manipulations could allow criminals to evade detection, while in banking and secure access systems, they could facilitate unauthorized access to sensitive information. Therefore, we investigate the opportunities for adversarial attacks to test and measure the robustness of biometric systems and leverage hash values for more accurate detection in a forensic setting.

3 Related Work

The vulnerability of facial expression recognition (FER) systems to adversarial attacks has become a critical area of research in computer vision and security. Adversarial examples, first introduced by Goodfellow et al. [16], revealed how deep learning models could be easily deceived by minimal perturbations. Building on this concept, Brown et al. [17] proposed adversarial patches—local regions of perturbations capable of fooling classifiers without altering the entire image. Unlike traditional perturbations, adversarial patches are physically realizable and effective even under real-world conditions, such as camera distortions and lighting changes [18].

In the context of facial expression recognition, Li et al. [19] demonstrated that adversarial patches could cause misclassification of emotions, such as interpreting anger as happiness or sadness as neutrality. Zhang et al. [20] extended these findings by exploring the real-world implications of adversarial attacks on emotion detection systems used in surveillance and driver monitoring. Their results emphasized the urgent need for defenses capable of mitigating such attacks.

Diffusion models have emerged as a powerful tool in both adversarial attack generation and defense. Ho et al. [21] introduced denoising diffusion probabilistic models (DDPM), which demonstrated state-of-the-art performance in image synthesis by iteratively refining Gaussian noise. Song et al. [22] further extended diffusion models to score-based generative modeling using stochastic differential equations (SDEs), enabling controllable and flexible image generation. These advancements paved the way for using diffusion models in adversarial research, including the generation of adversarial patches and purification-based defenses [23].

Choi et al. [24] proposed one of the earliest methods for generating adversarial patches using diffusion models. By integrating classifier gradients into the reverse diffusion process, they produced highly effective adversarial patches that could deceive facial recognition systems while maintaining high visual fidelity. Their approach demonstrated that diffusion-based patches were more resilient to common defenses such as JPEG compression and Gaussian noise compared to traditional gradient-based attacks. This work directly influenced research on adversarial patches for FER systems, highlighting their effectiveness in causing emotion misclassification.

Diffusion models are also effective as a defense mechanism against adversarial perturbations. Nie et al. [23] proposed DiffPure, a defense framework that utilizes diffusion models to remove adversarial noise through a reverse diffusion process. Their experiments showed that DiffPure could significantly improve the robustness of classifiers against various attacks, including adversarial patches. This approach has been recognized as a promising solution for enhancing the security of FER systems, where accurate emotion detection is critical.

Despite advancements in both attacks and defenses, the intersection of diffusion models and adversarial patches for facial expression recognition remains underexplored. Existing works such as those by Li et al. [19] and Zhang et al. [20] have primarily focused on gradient-based attacks, leaving diffusion-based approaches largely unaddressed. Additionally, while diffusion models have shown strong capabilities in image generation and purification, their potential for generating deceptive patches specifically targeting emotion recognition models has not been fully realized.

4 Methodology

Our approach involves a multi-stage pipeline to analyze and mitigate adversarial attacks on AI vision models, leveraging diffusion models, adversarial patch generation, and identity evasion techniques. The methodology is structured into five key phases: adversarial perturbation, diffusion-based patch generation, identity evasion, emotion poisoning attacks, and adversarial patch detection, as depicted in Fig 1.

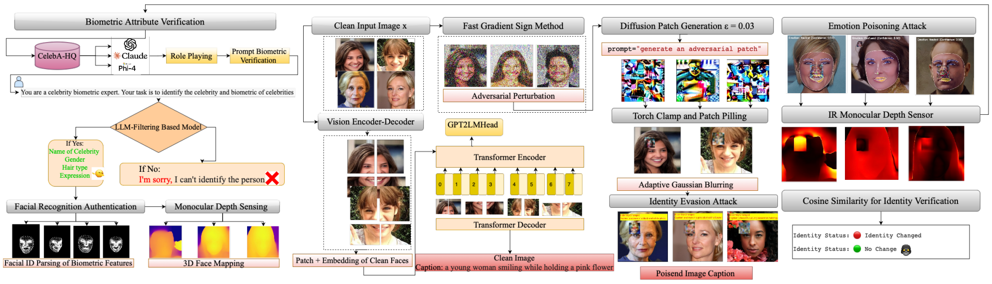


Fig. 1: Using source CelebA images of 178×218 pixels and adversarial patches of 50×50 pixels, the adversarial attack pipeline demonstrates the interaction between perturbation, diffusion-based patch generation, and identity evasion. This process integrates adversarial noise with patch refinement, using cosine similarity measurements to monitor attack effectiveness report on biometric identity verifacation system.

4.1 Biometric Attribute Verifacation

In our proposed biometric attribute verification pipeline, we designed a system to identify celebrities through facial recognition and biometric analysis. The process begins with referencing a large-scale celebrity biometric dataset, such as CelebA-HQ, which contains high-resolution images of various celebrities along with their associated biometric attributes. These attributes include not only facial features but also key metadata like the name, gender, hair type, and facial expressions. This database serves as the foundation for the model, providing a rich set of data for training and comparison during the identification process.

The system interacts with the user, who assumes the role of a celebrity biometric expert. In this role, the expert is tasked with identifying the celebrity in question and verifying the biometric features associated with that individual. The system prompts the user to verify specific biometric attributes of the celebrity, such as their gender, hair type, or facial expression. This step simulates real-world scenarios of celebrity recognition and attribute verification, where the user confirms these attributes based on the extracted data from the database.

The central component of the system is the LLM-Filtering based model, which uses LLM used to recognize and filter relevant celebrity features. This model processes inputs like celebrity name, gender, hair type, and expression, and then matches these attributes with the data in the dataset as a labaling method. The decision-making process in this model follows a logical flow: if the system successfully matches the celebrity's name with the provided attributes (gender, hair type, and expression), it proceeds with the verification of the biometric attributes. If no match is found, the system responds by saying, "I'm sorry, I can't identify the person," indicating that the celebrity could not be identified based on the celeberity identity.

Following this, facial recognition authentication is applied to validate the identified celebrity. Facial ID parsing extracts key facial features from the input image, such as the position of facial landmarks, eye shape, nose width, mouth curvature, and other unique characteristics. These features are passed through a facial recognition model, to compare the parsed facial data with the biometric database. This step is essential for ensuring that the celebrity's face is accurately identified.

Once the facial recognition is completed, the system performs biometric feature verification by cross-checking the celebrity's attributes, such as name, gender, hair type, and expression, with the verified facial features. This ensures the accuracy of both the celebrity identification and the biometric feature verification process.

A key feature of this system is its use of monocular depth sensing. This technique estimates depth information from a single image, offering more improved analysis of the face. Unlike stereo vision, which requires two images, monocular depth sensing uses computer vision algorithms, to estimate the relative distance between the camera and different objects in the scene. This method provides a 3D structure of the face from a 2D image, adding an additional layer of security to the celebrity identification process. By capturing detailed depth information, monocular depth sensing helps to distinguish between similar-looking individuals or detect any artificial manipulation of the image, such as adversarial attack.

Following monocular depth sensing, the system applies 3D face mapping, which generates a three-dimensional model of the face. This model provides a detailed understanding of the celebrity's facial structure, including the contours of the chin, nose bridge, and forehead shape, which are crucial for accurate biometric verification. 3D face mapping adds another layer of accuracy, ensuring that the celebrity is correctly identified, even when using images with subtle variations in lighting or angles.

Last and for most of this multi-step process,system consolidates the findings from facial recognition authentication, biometric feature verification, monocular depth sensing, and 3D face mapping. It then provides the final celebrity identification result, confirming the identity of the celebrity and their biometric attributes. If any step fails, the system apologizes and states that it cannot identify the person demonstrated in Fig2.

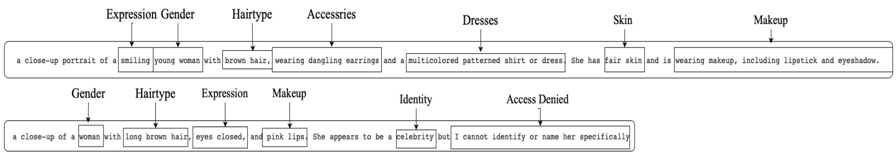


Fig. 2: Analysis of biometric labeling in the context of LLM filtration using respected feedback prompts into different categories.

4.2 FSGM-Diffusion Patch Generation

The FGSM is employed to generate adversarial perturbations by modifying an input image tensor x in a direction that maximizes the model's prediction error while maintaining imperceptible visual alterations. Given an input image x, a target class y, and a perturbation magnitude ϵ, the adversarial image x' is calculated by adding a small perturbation η to x, where η is derived from the loss function gradient with respect to the input.

The process begins with gradient computation, where the input tensor x is cloned, detached, and marked as requiring gradients to enable back-propagation. This ensures that the perturbation can be efficiently computed during the attack process. Once the input image is prepared, a forward pass is performed by feeding x into a pre-trained vision model to obtain logits, which represent the model's predicted class probabilities. The adversarial loss is then calculated using the cross-entropy loss function, which quantifies the difference between the model's output and the target class y. After computing the loss, backpropagation is applied to determine the gradient $\nabla_x J(\theta, x, y)$, which indicates the optimal direction in which the input image should be modified to maximize the model's prediction error.

Once the gradient is obtained, the adversarial noise η is generated by taking the element-wise sign of the gradient and scaling it by ϵ. This ensures that the perturbation is small but effective in causing misclassification. The perturbed image is then clamped to maintain valid pixel values within the range $[0, 1]$, ensuring that the modifications do not introduce unnatural artifacts. The final adversarial image $x' = x + \eta$ retains its perceptual similarity to the original image while deceiving the model into making incorrect predictions. The refinement process of the adversarial patch incorporates a diffusion-based denoising step to improve its stealthiness while preserving its adversarial effectiveness. Starting with an initial adversarial patch P, we apply a reverse diffusion process to ensure that the patch remains effective in misleading the model while appearing imperceptible to human observers. After refinement, the patch is placed onto the target image I at a specific coordinate (x, y). The refined patch P' is then resized to a smaller region P'' and further smoothed using Gaussian blurring to enhance seamless blending.

The final adversarial image I' is obtained by integrating the smoothed patch P'' into the original image at location (x, y). This process is mathematically represented as:

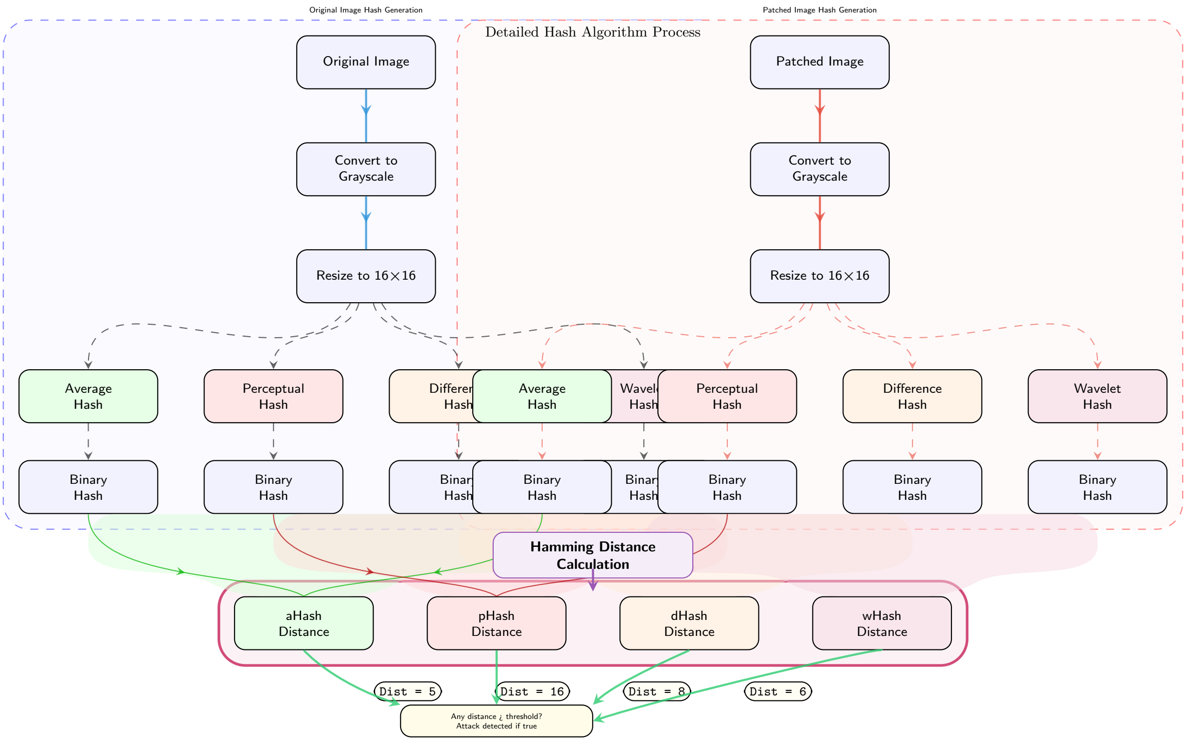
$$I' = I \odot M + P'' \odot (1 - M)$$

where M is a binary mask that determines the placement of the patch, and $\odot$ denotes element-wise multiplication to ensure smooth and natural integration. The Gaussian filter G_σ, with standard deviation σ, is applied to regulate the sharpness of the patch, making it less detectable. As a result, the adversarial image I' maintains its adversarial properties while ensuring the modification remains visually indistinguishable. To evade identity recognition systems, we apply an identity evasion attack to clean images. This step ensures seamless patch integration, reducing the detectability of the attack while manipulating vision encoder-decoder embeddings. The result is a perturbed image that modifies the AI-generated caption output, leading to poisoned image captions. Beyond identity evasion, we introduce emotion poisoning attacks, where adversarial patches alter emotion recognition outputs. The attack modifies facial features, leading to incorrect emotional classifications. This phase demonstrates the broader impact of adversarial attacks beyond identification, affecting high-level semantic interpretations in AI systems.

Fig. 3: Demonstrating the influence of conditional prompts in adversarial image generation based on different time steps. We implemented an adversarial attack pipeline that modifies identity classifications by leveraging both FGSM and diffusion-based adversarial sample generation. FGSM is then applied to generate adversarial noise gradually, subtly altering pixel values to mislead the classifier into assigning a different identity label.

4.3 Perceptual Hash and Structural Similarity in Forensic Setting

In this work, we introduce an approach for detecting adversarial patches using perceptual hashing and hash distance computation, combined with advanced image segmentation and machine learning techniques. The process begins with the preprocessing of input images, where we utilize a pre-trained ResNet50 model for image classification and a Stable Diffusion Img2Img pipeline to generate adversarial patches. The input images are resized and transformed into tensor format using standard image transformation techniques, ensuring compatibility with the neural network models.



4.4 Adversarial Patch Detection

The first step in our detection methodology involves computing multiple perceptual hashes for both the original and patched images. We leverage four distinct types of hashes: *aHash*, *pHash*, *dHash*, and *wHash*. These hashing methods are designed to capture subtle differences in image structure, texture, and pixel patterns. The images are converted to grayscale, and each hash is computed using the respective method. This provides a set of hash values that serve as the foundation for detecting any potential adversarial manipulations.

For an image I, the *aHash* is computed by dividing the image into blocks, calculating the average color value for each block, and then producing a binary string representation where each bit represents whether the corresponding block is above or below the average. where I_i is the color value of pixel i, N is the number of pixels in the block, and $\text{mean}(I)$ is the mean color of the image.

Similarly, the *pHash* captures the perceptual content of the image by performing a discrete cosine transform (DCT) and quantizing the resulting coefficients.

The *dHash* computes the differences between adjacent pixel values, while the *wHash* applies wavelet transforms to capture high-frequency image components.

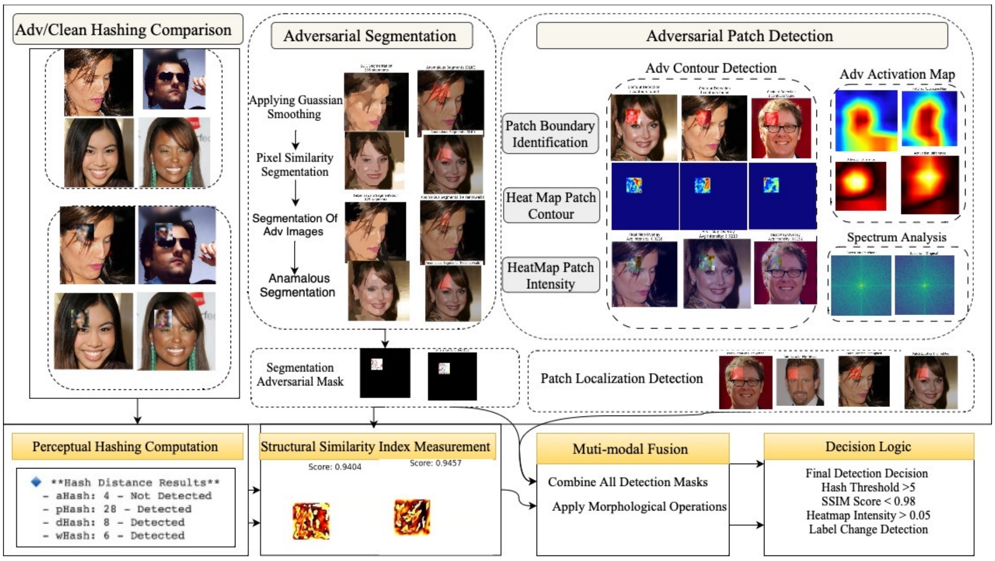


Fig. 4: The figure illustrates the process of adversarial patch detection, including the comparison of clean and adversarial hashing, adversarial segmentation, and patch localization detection. It also demonstrates perceptual hashing, structural similarity index measurement, and multi-modal fusion for accurate adversarial patch identification. The decision logic step incorporates final detection decisions based on various thresholds and parameters, ensuring robust detection of adversarial patches.

4.5 Hamming Distance Calculation

To quantify the difference between the original and patched images, we compute the *Hamming distance* between the perceptual hashes. The hamming distance is defined as the number of differing bits between two binary strings. Given two perceptual hashes H_1 and H_2, the hamming distance d_H is computed as:

$$d_H(H_1, H_2) = \sum_{i=1}^{N} \mathbb{1}\,(H_{1,i} \neq H_{2,i})$$

where $H_{1,i}$ and $H_{2,i}$ represent the i-th bit of hashes H_1 and H_2, respectively, and $\mathbb{1}(\cdot)$ is the indicator function which is 1 if the condition holds and 0 otherwise.

This measure helps in identifying how much the adversarial patch has altered the image in perceptual terms. A larger hamming distance implies a greater perceptual change, potentially indicating an adversarial attack.

The next stage of analysis involves computing the *SSIM* between the original and patched images. SSIM is a method for measuring the perceived quality of images by comparing luminance, contrast, and structure. It is mathematically defined as:

$$\mathrm{SSIM}(x, y) = \frac{(2\mu_x\mu_y + C_1)(2\sigma_{xy} + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_2)}$$

where: - x and y are the two image patches being compared. - μ_x, μ_y are the average pixel intensities of x and y. - σ_x^2, σ_y^2 are the variances of the pixel intensities of x and y. - σ_{xy} is the covariance of x and y. - C_1 and C_2 are constants to stabilize the division with weak denominators.

The SSIM score ranges from -1 (completely dissimilar) to 1 (identical), with values closer to 0 indicating more substantial differences, often due to adversarial manipulations.

To enhance the detection of localized differences, we segment both the original and patched images using advanced segmentation algorithms. Two segmentation methods are employed: *Felzenszwalb segmentation* and *SLIC (Simple Linear Iterative Clustering)* segmentation. These algorithms partition the image into smaller regions or segments, allowing for the analysis of specific areas that may contain the adversarial patch. By dividing the image into regions with distinct boundaries, segmentation provides a more granular analysis of the image's structure, making it easier to identify anomalies caused by adversarial modifications.

The Felzenszwalb segmentation algorithm applies a graph-based approach, where the image is represented as a graph and a minimal spanning tree is used to identify segments based on local color similarity. Mathematically, it segments the image by minimizing an energy function that takes into account local intensity, smoothness, and the number of segments.

For the *SLIC segmentation*, the image is over-segmented into superpixels based on a distance measure that considers both spatial proximity and color similarity:

$$\text{Distance} = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2 + \left(\frac{C_1 - C_2}{\sigma_C}\right)^2}$$

where (x_1, y_1) and (x_2, y_2) are the spatial coordinates of two neighboring pixels, and C_1 and C_2 are their respective color values.

Once the images are segmented, we compute the *average difference* in pixel values within each segment. The segment scores are analyzed to detect anomalies—segments with significantly higher differences compared to the rest of the image are flagged as potentially containing the adversarial patch. This anomaly detection approach leverages statistical thresholds, such as *two standard deviations* above the mean segment difference, to identify suspicious regions. The anomaly score for each segment S_k is given by:

$$\mathrm{score}(S_k) = \frac{\sum_{i \in S_k} |I_i - I_i'|}{|S_k|}$$

where I_i and I_i' are the pixel intensities in the original and patched images, and $|S_k|$ is the number of pixels in segment S_k. This score is compared to a threshold θ, typically defined as the mean score plus two standard deviations.

In addition to segmentation, we utilize *contour detection* to further localize the adversarial patch. By computing the absolute difference between the original and patched images, we identify regions where large changes have occurred. These changes are visualized as contours, which are drawn on the patched image to highlight the exact location of the adversarial patch.

To complement the contour-based detection, we generate a *heatmap* to visualize the intensity of changes across the image. The heatmap is created by comparing the images in the LAB color space, which is more sensitive to color differences than the RGB space. The difference in the L, A, and B channels is combined to produce a heatmap that highlights regions of the image most affected by the adversarial patch. Gaussian blur is applied to reduce noise, and the heatmap is normalized to emphasize areas with the greatest disparity:

$$\text{heatmap}(I, I') = \frac{|I - I'|}{\max(|I - I'|)}$$

where I and I' are the original and patched images, respectively. We also incorporate a *neural activation map* generated by passing the image through the ResNet50 model. This map highlights the regions of the image that the model focuses on when making a classification decision. By comparing the activation maps of the original and patched images, we can assess whether the model's attention is altered due to the adversarial patch. differences in the activation maps can provide strong evidence of adversarial manipulation.

The activation map is derived by extracting the output of the last convolutional layer $A(x)$ after passing the image x through the network:

$$A(x) = f_{layer4}(x)$$

where f_{layer4} represents the forward pass through the last convolutional block. To visualize the results of the perceptual hash comparison, we create a *hash distance bar plot*. The plot shows the hamming distance values for each of the four hash types, with a threshold line indicating the critical level at which a patch is considered detected. Bars are color-coded to differentiate between detected and undetected patches, making it easy to assess the effectiveness of each hash method in identifying adversarial changes. Finally, the system integrates all detection methods—hash distance, SSIM, segmentation anomalies, contour detection, heatmap visualization, and neural network activation mapping—into a unified decision framework. If any of the detection methods surpass the predefined thresholds, the system flags the image as containing an adversarial patch. The results are then displayed visually, with the location of the adversarial patch highlighted, and a detailed summary of each detection method's findings.

Through this end to end pipeline, we are able to effectively detect adversarial patches and understand their impact on image integrity. The combination of perceptual hashing, structural similarity analysis, segmentation, and neural network activation mapping provides a framework for identifying and visualizing adversarial manipulations in images.

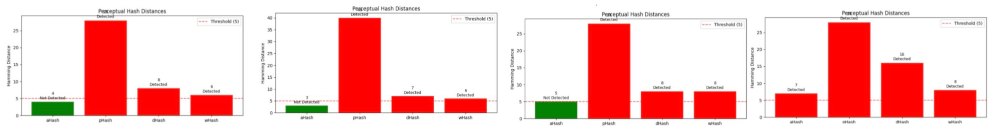


Fig. 5: displaying perceptual hash distances for four different hash methods (aHash, pHash, dHash, and wHash), with detection results based on a threshold of 5. The bars represent the hamming distances between the hashes, where the red bars indicate detection and the green bars indicate non-detection. The threshold line at 5 helps to determine which hashes exceed the detection limit.

5 Model Evaluation and Comparison

In this section we aim to evaluate our proposed model into differenet caregory of 1) Evaluating the impact of adversarial attack for evasion attack 2) patch generation evaluation 3) Impact of differenet parameters on attack success 4) Transferable optimal adversarial patch evaluation 5) Identity evaluation for poisend caption Identity recognition and number of detected adversarial attack detection method.

5.1 Evaluating the Impact of Adversarial Patches for Identity Evasion Attack

The visualizations in Figure 6 provide compelling evidence of the effectiveness of identity patch evasion attacks on computer vision systems. The results demonstrate how adversarial patches—colorful, abstract overlays on facial images successfully disrupt facial recognition and depth perception mechanisms. These carefully crafted patches in facial recognition conceal identities while preserving human recognizability, creating a disconnect between machine and human perception. The Identity Patch Evasion Attack results illustrate this effect, where the patched faces evade recognition while remaining visibly identifiable to humans. This highlights the fundamental vulnerability of AI-based recognition systems when confronted with adversarial manipulations. Even more concerning is the impact on depth perception. The depth monocular sensing visualization section reveals drastic distortions in the perceived spatial structure of faces. Original depth maps depict smooth facial surfaces, while patched depth maps introduce abrupt geometric anomalies. The difference maps, marked by intense red and yellow regions, further expose the severity of these distortions, suggesting interference with the model's depth estimation process. The 3D depth map comparison Before and After Patch Evasion Attack results in Figure 6 further quantify these disruptions. The original depth maps exhibit well-defined facial structures, while the patched depth maps introduce unnatural protrusions and depressions. The depth difference maps, with their pronounced spikes and valleys, quantify these estimation errors, demonstrating how the attack method persistently destabilizes spatial understanding across different samples.

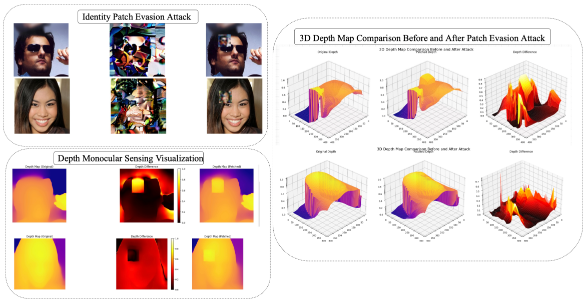


Fig. 6: Effects of identity patch evasion attacks on facial recognition and depth perception.

5.2 Patch Generation Evaluation

The table titled *Comparison of Proposed Work with Other Studies* compares the performance of adversarial patches across multiple studies, using key metrics such as SSIM, LPIPS, L2 distance, MS-SSIM, transferability, and overall score.

In our proposed work pre-trained models for identity classification, with LPIPS used to assess perceptual quality. The diffusion model is used to refine adversarial patches through diffusion-based denoising, improving their imperceptibility. Adversarial patches are generated using the FGSM and then refined for robustness.

Several metrics are used for evaluation: SSIM measures structural similarity,LPIPS assesses perceptual quality,MS-SSIM evaluates multiple image scales, and L2 distance quantifies pixel-wise differences. Transferability is tested by evaluating patch success across models. The results are tested with various patch configurations (diffusion strength, size, and position). Study 1 [25] shows similar performance but with slightly lower SSIM and MS-SSIM scores, and higher transferability. Study 2[26] achieves comparable LPIPS and SSIM scores but has slightly lower MS-SSIM. Study 3 [27] shows lower scores across all metrics, indicating less effectiveness in fooling models.

Overall, the proposed work demonstrates strong performance, with a balanced approach that ensures high imperceptibility, effectiveness, and transferability, making it a competitive method in adversarial patch generation.

Table 1: Comparison of Metrics and Performance Across Different Studies. The changes are indicated with arrows showing the direction of increase or decrease.

	Metrics			Performance		
Sample	**SSIM**	**LPIPS**	**L2**	**MS-SSIM**	**Transfer**	**Score**
Proposed Work	0.94 ↑	0.090 ↓	0.078 ↓	0.808 ↑	0.389 ↓	0.705 ↑
Study 1	0.92 ↓	0.080 ↓	0.070 ↓	0.790 ↓	0.400 ↑	0.685 ↓
Study 2	0.93 ↑	0.085 ↑	0.065 ↓	0.805 ↑	0.420 ↑	0.700 ↑
Study 3	0.91 ↓	0.095 ↑	0.085 ↑	0.775 ↓	0.375 ↓	0.670 ↓

5.3 Impact of Different Parameters on Attack Success, Confidence Drop, and Effectiveness Score

The figure7 consists of three bar charts that provide a detailed comparison of different parameters influencing the performance of an attack. The first chart, titled , demonstrates the success rate of the attack across six different parameters: target_class, epsilon, position, size, blur_radius, and diffusion_strength. It is evident from the chart that parameters like target_class, epsilon, and diffusion_strength exhibit a high success rate of around 80%, while position and size contribute to a noticeably lower success rate of 60%.

The second chart, shows the average confidence drop across the same parameters. Diffusion_strength again stands out with the highest drop in confidence, followed by target_class, which also demonstrates a relatively drop. The other parameters, including epsilon, position, and size, have a smaller impact on the confidence drop, reflecting a consistent trend with the attack success rate chart where fewer changes were observed in these parameters.

Finally, the third chart, provides an assessment of the best combined score of each parameter, illustrating how each one contributes to the overall attack effectiveness. Target_class stands out with the highest score of 2.86, while position and size perform relatively poorly with scores around 2.78. This reinforces the conclusion that some parameters, such as target_class and diffusion_strength, lead to a more effective attack compared to others.

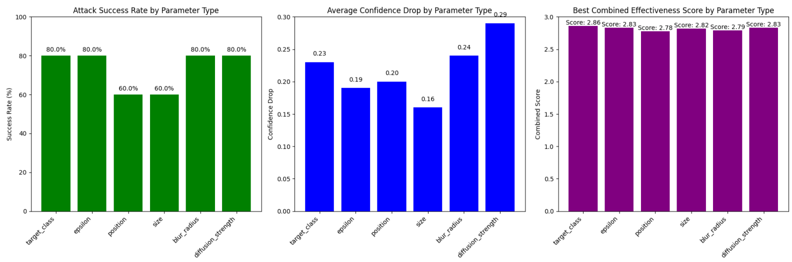


Fig. 7: Analysis of adversarial patch attack performance: success rate, confidence drop, and combined effectiveness score across parameters like target class, epsilon, position, size, blur radius, and diffusion strength.

5.4 Transferable Optimal Adversarial Patch Evaluation

Our first figure focuses on patch variation analysis and examines how position variation affects our adversarial attack success. Our left side shows the original image, while subsequent images display different patch positions with their class and confidence scores. Our confidence drop metrics demonstrate how the attack's effectiveness changes as the patch moves across the image. Some positions in our analysis yield a higher success rate(with confidence drops of 0.03 and 0.10), while others diminish our attack's success (with larger drops of up to 0.21)—demonstrating how patch positioning influences effectiveness. Our second figure presents an optimal adversarial patch analysis featuring our best-performing patch. It displays our original image alongside our optimal adversarial patch. Our patched image demonstrates the attack's success through a marked shift in image classification, evidenced by changes in confidence scores. This shows how effectively our patch manipulates the model's classification. Our difference visualization highlights the subtle changes between our original and patched images that alter the model's prediction. In Table 2, we evaluated our proposed model for testing the transferability attribute of adversarial patch generation and their respective quality for further examination for different groups of samples based on different geographical places of adversarial patch creation.

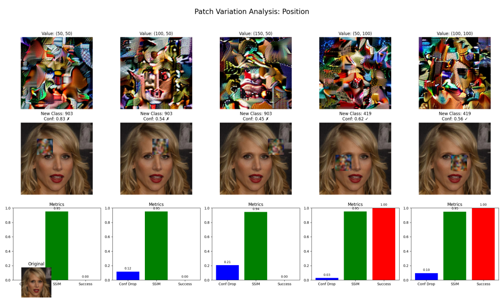


Fig. 8: Patch Variation Analysis for Different Positions. The images show the impact of adversarial patch positions on classification results, with metrics displaying SSIM, confidence drop, and success rates.

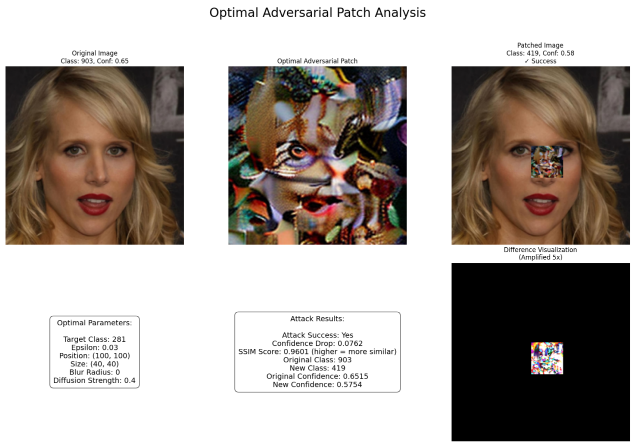


Fig. 9: The images show the original (left), optimal adversarial patch (center), and patched image (right), demonstrating a successful attack with changes in classification, confidence drop, SSIM score, and attack parameters.

Table 2: Metrics for Adversarial Patch Transferability Testing across Various Configurations and Evaluation. Success is indicated by "Yes" and "No", and transferability varies from 0 to 1 across samples.

	Metrics for Position (50, 50)			Metrics for Position (100, 100)		
Sample	Success	Confidence Drop	SSIM	Success	Confidence Drop	SSIM
Group 1	Yes	-0.1507	0.9777	Yes	-0.0855	0.9812
Group 2	No	0.0529	0.9798	Yes	-0.0824	0.9771
Group 3	Yes	0.0974	0.9827	Yes	0.0627	0.9805
Group 4	Yes	-0.0541	0.9759	Yes	-0.5963	0.9765
Group 5	Yes	0.1017	0.9801	Yes	-0.0336	0.9754
	LPIPS Metrics			Transferability		
Sample	LPIPS	Transferability	LPIPS	Transferability	LPIPS	Transferability
Group 1	0.0444	0.50	0.0256	0.50	0.0297	0.75
Group 2	0.0176	0.50	0.0261	1.00	0.0264	0.75
Group 3	0.0326	0.25	0.0256	0.50	0.0235	0.25
Group 4	0.0370	0.25	0.0336	0.75	0.0581	0.50
Group 5	0.0295	0.25	0.0505	0.50	0.0385	0.00
Overall Attack Success Rate:		81.11%				
Average Metrics:		Confidence Drop: -0.1696 SSIM Score: 0.9403 LPIPS Score: 0.0840 MS-SSIM Score: 0.8664 Transferability: 0.6222				
Best Configuration:		Sample: 2 Diffusion Strength: 0.5 Patch Size: 30 Position: (150, 50) Success: Yes Composite Score: 0.9927				

5.5 Identity Evaluation

In our comprehensive approach for generating adversarial captions, we evaluate their impact through various strategies, including semantic and grammatical alterations, contextual distortions, and other transformations. This evaluation delve deeply into the robustness of automated captioning systems. We leverage pre-trained models such as vision encoder decoder model and viT image processor to generate captions from images. Following this, we apply a series of adversarial strategies to modify these captions, allowing us to assess the impact these modifications have on the integrity and effectiveness of the generated content. Our approach employs a wide range of metrics, including BLEU, METEOR, ROUGE, and accuracy metrics, to evaluate both clean and adversarial captions.

In our implementation, we developed the adversarial caption generator class, which plays a central role by applying a series of poisoning strategies to the generated captions. We designed these strategies to involve significant alterations that make the captions adversarial. For instance, we use semantic distortion to replace words with synonyms, grammatical mutation to modify the grammatical structure, and contextual hallucination to introduce unexpected elements into the caption.

We've made the use of poisoning templates a crucial feature of our adversarial caption generator. We designed these templates to flag potential poisoned captions by applying structured, contextually relevant modifications to the original captions. Rather than introducing random changes, we use the templates to guide the modification of specific elements such as nouns, adjectives, and verbs.

In our implementation, we developed the caption dataset class to support efficient processing of images and captions. We designed it to facilitate the loading of images, generation of captions through pre-trained models like ViT-GPT2, and application of adversarial transformations.

In our evaluation of the adversarial captions, we found several important results. We observed BLEU-1 and BLEU-4 scores of 0.5377 and 0.2741, respectively, showing us a moderate overlap in n-grams between the original and adversarial captions. Our METEOR score of 0.6473 reflects a reasonable alignment between our generated and human-written references. We also found that our ROUGE scores (ROUGE-1 at 0.6967, ROUGE-2 at 0.3131, and ROUGE-L at 0.4666) further support these findings. Through our clean accuracy score of 0.9231, we confirmed that a significant portion of the original captions remained intact, while our adversarial accuracy score of 0.4069 showed us that the adversarial captions achieved moderate deviation from the original content.

Table 3: Caption Metrics Results

Metric	**Value**
BLEU-1	0.5377
BLEU-4	0.2741
METEOR	0.6473
ROUGE-1	0.6967
ROUGE-2	0.3131
ROUGE-L	0.4666
Clean Accuracy	0.9231
Adversarial Accuracy	0.4069
Semantic Shift	0.4157

Source: Results obtained from adversarial caption generation evaluation metrics.

5.6 Adversarial Detection

The Figure 10 presents analysis of SSIM scores, which are used to assess the visual similarity between two images. The first plot on the left shows the distribution of SSIM scores for 500 images, with a clear threshold (0.98) marked by a red dashed line. This threshold is set to help determine the detection rate, which in this case is 100%, indicating that all images have been successfully detected based on this threshold.

The second plot in the center displays the sorted SSIM scores for the 500 images, ordered from lowest to highest. The threshold is again marked with a red dashed line at 0.98, providing a reference point for the sorted scores. This plot visually confirms that all 500 images have SSIM scores higher than the threshold, with a detection rate of 100%.

The third plot, on the right, presents a box plot of SSIM scores, offering a summary of the statistical distribution. The plot shows the minimum, first quartile (Q1), median, third quartile (Q3), and maximum SSIM values, providing insights into the range and variability of the scores. The SSIM scores are concentrated around the 0.946 median, with a narrow interquartile range, indicating high similarity between the images.

Together, these visualizations provide a detailed overview of the SSIM score distribution and detection performance, demonstrating the consistency and effectiveness of the thresholding method for image comparison.

Fig. 10: Demonstrating the number of detected adversaries from an imbalance dataset by visualization and tracking the SSIM distribution

Table 4: SSIM Score Distribution Summary

Metric	Value
Minimum SSIM Score	0.9354
Q1 (First Quartile)	0.9419
Median SSIM Score	0.9465
Q3 (Third Quartile)	0.9489
Maximum SSIM Score	0.9579
Threshold for Detection	0.98
Detection Rate	100%

6 Conclusion

In this scope of study, we thoroughly examine this forensic occurrence of adversaries. We witnessed and explored the opportunities of adversarial patch generation using a stable diffusion model, and we extended our methodology to a systematic approach design to detect and poison the number of identity recognition of adversarial patch attacks to the dataset. We have successfully detected and analyzed the respected measurement through this study. We achieved excellent evaluation metrics by evaluating our proposed methodology and detecting adversarial patch attacks.

Acknowledgments. The research was partially sponsored by the Army Research Office and was carried out under Grant Number W911NF-21-1-0264. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the Army Research Office or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for Government purposes, notwithstanding any copyright notation herein.
In addition, this research was supported as part of the 2024 Summer Research Experience for Graduate Students Program of the Northrop Grumman Research and Education Program (NG REP) through the Florida A& M University Foundation.

References

[1] Hwang, R.-H., Lin, J.-Y., Hsieh, S.-Y., Lin, H.-Y., Lin, C.-L.: Adversarial patch attacks on deep-learning-based face recognition systems using generative adversarial networks. Sensors **23** (2023) https://doi.org/10.3390/s23020853

[2] Tampubolon, M.: Digital face forgery and the role of digital forensics. International Journal for the Semiotics of Law-Revue internationale de Sémiotique juridique **37**(3), 753–767 (2024)

[3] Campbell, S.L., Gear, C.W.: The index of general nonlinear DAES. Numer. Math. **72**(2), 173–196 (1995)

[4] Yutia, S.N., Triwibisono, C., Rahmayadi, G.E.: Digital forensic on image manipulation crime-a systematic literature review. In: 2024 International Conference on Intelligent Cybernetics Technology & Applications (ICICyTA), pp. 824–828 (2024). IEEE

[5] Shetty, P.: Use of digital forensics in cybersecurity and criminal cases, how can it be leveraged, challenges. International Journal of Science and Research (IJSR) (2020) https://doi.org/10.21275/sr24213025946

[6] Wu, C., Zhang, T., Li, Z., Lian, Z.: Enhancing adversarial patch effectiveness for face recognition systems using sharpness-aware minimization. In: 2024 IEEE Cyber Science and Technology Congress (CyberSciTech), pp. 384–389 (2024)

[7] Thys, S., Van Ranst, W., Goedemé, T.: Fooling automated surveillance cameras: Adversarial patches to attack person detection. In: CVPR Workshops, pp. 49–55 (2019)

[8] Geng, P., Deng, X.: An adversarial patch attack for vehicle detectors in the physical world. In: IEEE International Conference on Unmanned Systems (ICUS) (2023)

[9] Hingun, N., Sitawarin, C., Li, J., Wagner, D.: Reap: A large-scale realistic adversarial patch benchmark. arXiv preprint arXiv:2212.05680 (2022)

[10] Li, C., Yan, H., Zhou, L., Chen, T., Liu, Z., Su, H.: Prompt-guided environmentally consistent adversarial patch. (2024)

[11] Jutras, M., Liang, E., Leary, S., Ward, C., Manville, K.: Detecting physical adversarial patch attacks with object detectors. In: IEEE Applied Imagery Pattern Recognition Workshop (AIPR), pp. 1–7 (2022)

[12] Sayyafzadeh, S., Xu, W., Chi, H.: Forensic analysis of contents in thumbnails using transfer learning. In: Proceedings of the Future Technologies Conference, pp. 517–535. Springer, ??? (2023)

[13] Arvinte, M., Tewfik, A., Vishwanath, S.: Detecting patch adversarial attacks with image residuals. arXiv preprint arXiv:2002.12504 (2020)

[14] Sharma, A., Bian, Y., Munz, P., Narayan, A.: Adversarial patch attacks and defences in vision-based tasks: A survey. arXiv preprint arXiv:2206.08304 (2022)

[15] Mathov, Y., Rokach, L., Elovici, Y.: Enhancing real-world adversarial patches with 3d modeling techniques. arXiv preprint arXiv:2102.05334 (2021)

[16] Goodfellow, I.J., Shlens, J., Szegedy, C.: Explaining and harnessing adversarial examples. In: International Conference on Learning Representations (ICLR) (2015)

[17] Brown, T.B., Mane, D., Roy, A., Abadi, M., Gilmer, J.: Adversarial patch. In: Advances in Neural Information Processing Systems (NeurIPS) Workshop (2017)

[18] Sharif, M., Bhagavatula, S., Bauer, L., Reiter, M.K.: A general framework for adversarial examples with objectives. In: ACM Transactions on Privacy and Security (TOPS), vol. 22, pp. 1–30. ACM, ??? (2019)

[19] Li, J., Zhang, Y., Liu, Q., Ma, X., Wang, Y.: Adversarial attacks on deep learning-based facial expression recognition systems. In: IEEE International Conference on Image Processing (ICIP), pp. 1086–1090 (2021). IEEE

[20] Zhang, W., Liu, J., Chen, L., Xu, K.: Real-world adversarial attacks on facial expression recognition systems. Pattern Recognition **128**, 108667 (2022)

[21] Ho, J., Jain, A., Abbeel, P.: Denoising diffusion probabilistic models. In: Advances in Neural Information Processing Systems (NeurIPS), vol. 33, pp. 6840–6851 (2020)

[22] Song, Y., Sohl-Dickstein, J., Kingma, D.P., Kumar, A., Ermon, S., Poole, B.: Score-based generative modeling through stochastic differential equations. In: International Conference on Learning Representations (ICLR) (2021)

[23] Nie, W., Song, Y., Zhu, Y., Zhang, J., Zhou, X., Ma, K., Liang, K., Wang, Y., Ma, Y., Xu, D.: Diffusion models for adversarial purification. In: Advances in Neural

Information Processing Systems (NeurIPS) (2022)

[24] Choi, M., Park, J., Kim, S., Lee, H.: Diffusion-based adversarial patch generation for robust attacks on facial recognition. arXiv preprint arXiv:2301.01234 (2023)

[25] Tadros, T., Cullen, N., Greene, M.R., Cooper, E.A.: Assessing neural network scene classification from degraded images. ACM Transactions on Applied Perception (TAP) **16**, 1–20 (2019) https://doi.org/10.1145/3342349

[26] Xie, S., Girshick, R., Dollár, P., Tu, Z., He, K.: Aggregated residual transformations for deep neural networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1492–1500 (2017). https://doi.org/10.1109/CVPR.2017.634

[27] Szegedy, C., Zaremba, W., Sutskever, I., Bruna, J., Erhan, D., Goodfellow, I., Fergus, R.: Intriguing properties of neural networks. arXiv preprint arXiv:1312.6199 (2014)

Artificial Intelligence in Legal Forensics: Principles, Applications, and Emerging Frontiers

S.S. Iyengar

Florida International University, Miami, 33199, Florida, USA.

Contributing authors: iyengar@cs.fiu.edu;

Abstract

The integration of Artificial Intelligence (AI) and Machine Learning (ML) into forensic science is reshaping the landscape of legal investigations, enabling unprecedented advancements in accuracy, efficiency, and evidentiary analysis. This paper explores the foundational principles, practical applications, and emerging frontiers of AI in legal forensics, encompassing diverse domains including legal document analysis, digital forensics, canine scent detection, and DNA profiling.

AI-driven tools now streamline case review processes, automate legal reasoning, detect anomalies in digital and genetic data, and replicate canine olfactory capabilities with bio-inspired sensors. From predictive modeling in judicial outcomes to deepfake detection in multimedia forensics, this study highlights how AI augments traditional investigative practices while addressing challenges related to ethics, bias, transparency, and admissibility.

As the legal system grapples with increasingly complex datasets and evolving criminal tactics, the convergence of AI and forensic science offers a transformative pathway to justice. This paper not only synthesizes the current state of AI-enabled forensic technologies but also envisions their future role in fostering global legal collaboration, real-time forensic intelligence, and standardization in criminal justice procedures.

Keywords: Legal Forensic Analysis, Digital Forensics, Artificial Intelligence, AI-Driven Investigations

1 Introduction

The intersection of Artificial Intelligence (AI) and legal forensics marks a transformative shift in how justice is pursued in the digital age [1–4]. As the complexity of crimes increases and legal systems face overwhelming volumes of data, traditional investigative and legal methodologies often fall short [5–8]. To bridge this gap, AI and Machine Learning (ML) are being increasingly deployed to enhance the precision, efficiency, and scalability of forensic investigations [4, 9–11].

Legal forensics—a field focused on the analysis of evidence for legal proceedings—now benefits from AI-driven solutions that automate document review, predict legal outcomes, analyze voice and video content, and uncover hidden correlations across diverse datasets [12–16]. From courtroom analytics to evidentiary validation, AI enables law enforcement agencies, attorneys, and judges to draw actionable insights from vast and multifaceted information sources [17–19]. This evolution not only accelerates case resolution but also strengthens the fairness and transparency of judicial processes [20–22].

1.1 The Role and Necessity of AI in Legal Forensics

Legal forensics encompasses tasks such as reviewing complex litigation documents, identifying fraud patterns, analyzing speech transcripts, and evaluating large volumes of multimedia and digital evidence. The advent of cybercrime, deepfake manipulation, and cross-border data breaches has further necessitated AI-enabled tools that can operate at speed and scale far beyond human capacity.

Key advantages of AI in legal forensic applications include:

- **Automated Document Analysis:** Natural Language Processing (NLP) models facilitate rapid review and annotation of legal texts, flagging inconsistencies and prioritizing relevant case information.
- **Predictive Legal Analytics:** AI models forecast legal outcomes based on historical case data, providing strategic insights to lawyers and judges.
- **Cross-modal Evidence Correlation:** AI systems unify data streams from documents, audio, video, and digital logs to build cohesive narratives for legal proceedings.
- **Fraud and Anomaly Detection:** Machine learning algorithms detect unusual patterns in contracts, financial records, and testimonies, aiding in early identification of fraudulent behavior.

1.2 Scope of the Paper

This paper explores the principles, applications, and emerging frontiers of AI in legal forensics. It examines how AI technologies are deployed across core areas such as document analysis, evidence correlation, voice and video analytics, and case outcome prediction. Furthermore, it discusses the ethical, legal, and regulatory implications of integrating AI into judicial systems, including concerns about algorithmic bias, data privacy, and admissibility of AI-generated findings in court.

By providing a comprehensive view of AI-enabled legal forensics, this study highlights the ongoing transformation of legal practices and underscores the potential of intelligent technologies to uphold justice in an increasingly digital world.

2 AI-Driven Innovations in Legal Forensic Analysis

Legal forensic analysis is foundational to the justice system, encompassing the examination of legal documents, digital and physical evidence, and verbal testimonies to reconstruct events and support fair adjudication. Traditionally, this process has been labor-intensive, requiring extensive manual review and domain expertise. With the rise of Artificial Intelligence (AI) and Machine Learning (ML), legal forensic workflows are undergoing a paradigm shift—enabling data-driven insights, accelerating case processing, and enhancing judicial transparency.

AI applications in legal forensics now span multiple domains, including document intelligence, evidence correlation, predictive analytics, and voice analysis. This section outlines the core capabilities AI brings to modern legal investigations.

2.1 Document Intelligence and Legal Text Analysis

Legal proceedings often generate large volumes of case files, contracts, and deposition transcripts. AI-powered Natural Language Processing (NLP) systems automate the extraction, classification, and synthesis of legal texts, significantly reducing review times and identifying high-risk elements.

- **Automated Legal Review:** NLP models flag inconsistencies, contradictions, and key phrases in contracts, pleadings, and affidavits.
- **Fraud Detection:** Machine learning algorithms uncover anomalies in financial statements, insurance claims, and regulatory documents.
- **Legal Research Augmentation:** AI cross-references case law, statutes, and legal arguments, assisting lawyers in developing precedent-based strategies.

2.2 Cross-Source Evidence Correlation

AI systems excel at fusing and contextualizing evidence from disparate sources. Whether integrating digital records, surveillance footage, or testimonial transcripts, AI algorithms reconstruct timelines and identify hidden connections critical to a case.

- **Multi-Modal Evidence Analysis:** AI integrates text, images, audio, and video to produce a holistic view of legal scenarios.
- **Relationship Mapping:** Link analysis engines reveal connections between suspects, co-conspirators, and events through graph-based visualizations.
- **Chain-of-Custody Verification:** AI ensures evidentiary integrity through metadata validation and audit trails.

2.3 Predictive Legal Analytics

Machine learning models can forecast potential legal outcomes based on historical datasets, judicial behavior, and precedent-driven trends.

- **Case Outcome Prediction:** AI estimates likelihoods of verdicts or settlements by analyzing similar past cases.
- **Sentencing Trend Analysis:** Algorithms evaluate sentencing patterns across jurisdictions, helping assess the proportionality of punishment.
- **Risk Scoring and Strategy Development:** AI tools score the reliability of witnesses or the viability of legal arguments to support trial preparation.

2.4 Challenges and the Path Ahead

Despite its promise, AI-driven legal forensics faces challenges:

- **Bias and Fairness:** AI models must be trained on diverse datasets to avoid perpetuating systemic biases.
- **Data Privacy:** Legal applications of AI must align with data protection laws and maintain evidentiary confidentiality.
- **Admissibility and Transparency:** Legal systems demand explainable AI, ensuring that algorithmic outputs can be verified, justified, and accepted in court.

As AI technologies evolve, their role in the legal system will expand—enabling faster case resolution, more equitable justice, and global collaboration in transnational legal challenges. Legal professionals must be equipped to leverage these tools while ensuring ethical oversight and legal rigor.

3 Challenges and Ethical Considerations

While AI and ML offer transformative advantages in forensic science, their implementation introduces critical ethical, legal, and operational challenges. Ensuring responsible and effective adoption of AI in legal forensics requires addressing the following key concerns:

- **Data Privacy and Security:** AI systems must comply with data protection regulations such as GDPR and HIPAA, especially when handling sensitive legal or biometric information.
- **Bias in AI Models:** Algorithms trained on non-representative datasets may reinforce existing societal biases, leading to wrongful identifications or unfair legal conclusions.
- **Legal and Regulatory Compliance:** Forensic tools powered by AI must align with evidentiary standards to ensure admissibility in courts, necessitating validation, documentation, and judicial oversight.
- **Interpretability and Human Oversight:** AI-generated decisions must be transparent and explainable, allowing forensic experts and legal professionals to understand and verify system outputs.
- **Ethical Use and Accountability:** Misuse of AI in surveillance, profiling, or manipulation of evidence poses risks to civil liberties. Robust ethical frameworks are required to define acceptable boundaries and accountability mechanisms.

4 Future Directions and Recommendations

The next frontier of AI in legal forensics lies in the convergence of interdisciplinary innovation, regulatory harmonization, and scalable real-time applications. Key future directions include:

- **AI-Enhanced Biometric Fusion:** Integrating facial recognition, voice profiling, gait analysis, and iris scanning into a unified forensic identity verification framework.
- **Quantum Computing Integration:** Leveraging quantum algorithms to accelerate decryption, data authentication, and secure transmission of forensic data across jurisdictions.
- **Portable AI Forensic Units:** Developing mobile, edge-computing-enabled forensic devices for on-site analysis of digital evidence, biosignals, and chemical signatures.
- **Standardization of AI Forensic Protocols:** Establishing globally accepted technical and ethical standards to ensure interoperability and accountability in cross-border legal investigations.
- **Explainable AI (XAI) Implementation:** Prioritizing interpretable model design in forensic applications to build trust, support legal scrutiny, and uphold judicial integrity.

5 Conclusion

Artificial Intelligence (AI) and Machine Learning (ML) have fundamentally reshaped forensic science, enabling faster, more accurate, and scalable investigative capabilities across diverse domains—legal analysis, digital forensics, canine detection, and genetic profiling. By automating complex processes such as evidence correlation, document analysis, malware detection, and scent identification, AI-driven systems significantly enhance the effectiveness of modern forensic investigations.

The ability of AI to extract actionable insights from vast, multimodal datasets has empowered legal professionals and law enforcement to pursue justice with greater precision and efficiency. However, the integration of AI into forensic workflows also demands rigorous attention to issues of fairness, bias, privacy, and legal admissibility.

As research and technology evolve, the future of AI-enabled forensics lies in ethical innovation, real-time intelligence, and global collaboration. The continued development of trustworthy and transparent AI systems will be essential to ensuring that the pursuit of justice remains both equitable and technologically resilient in the years to come.

Acknowledgments. This research was partially sponsored by the Army Research Office and the NSF, and was accomplished under Grant Number W911NF-21-1-0264 and 2018611. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the Army Research Office or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation herein.

References

[1] Hadid, A., Evans, N., Marcel, S., Fierrez, J.: Biometrics systems under spoofing attack: An evaluation methodology and lessons learned. IEEE Signal Processing Magazine **32**(5), 20–30 (2015)

[2] Bureau, N.C.R.: Cyber Crime Report. Ministry of Home Affairs, Government of India (2022)

[3] Iyengar, S.S., Nabavi, S., Hariprasad, Y., H.B., P., C., K.M.: Artificial Intelligence in Practice: Theory and Applications for Cyber Security and Forensics. Springer, ??? (2025)

[4] Shi, B., Iyengar, S.S.: Mathematical Theories of Machine Learning-Theory and Applications. Springer, ??? (2020)

[5] Iyengar, S.S., Kashyap, R.L., Madan, R.N.: Distributed sensor networks-introduction to the special section. IEEE Transactions on Systems, Man, and Cybernetics **21**(5), 1027–1031 (1991)

[6] Iyengar, S.S., Deng, W.: An efficient edge detection algorithm using relaxation labeling technique. Pattern Recognition **28**(4), 519–536 (1995)

[7] Thejas, G.S., Boroojeni, K.G., Chandna, K., Bhatia, I., Iyengar, S.S., Sunitha, N.R.: Deep learning-based model to fight against ad click fraud. In: Proceedings of the 2019 ACM Southeast Conference, pp. 176–181 (2019)

[8] Guo, M., Pissinou, N., Iyengar, S.S.: Privacy-preserving deep learning for enabling big edge data analytics in internet of things. In: 2019 Tenth International Green and Sustainable Computing Conference (IGSC), pp. 1–6. IEEE, ??? (2019)

[9] Singaram, J., Iyengar, S.S., Madni, A.M.: Deep Learning Networks: Design, Development and Deployment. Springer, ??? (2023)

[10] Wang, C., Iyengar, S.S., Sun, K.: AI Embedded Assurance for Cyber Systems. Springer, ??? (2023)

[11] Iyengar, S.S., Kashyap, R.L.: Guest editor's introduction: Image databases. IEEE Transactions on Software Engineering **14**(5), 608 (1988)

[12] Pouyanfar, S., Yang, Y., Chen, S.C., Shyu, M.L., Iyengar, S.S.: Multimedia big data analytics: A survey. ACM Computing Surveys (CSUR) **51**(1), 1–34 (2018)

[13] Hariprasad, Y., Iyengar, S.S., Subramanian, N.: Deepfake video detection using lip region analysis with advanced artificial intelligence based anomaly detection technique. Authorea Preprints (2024)

[14] Hariprasad, Y., K.J., L.K., Suraj, L., Iyengar, S.S.: Boundary-based fake face

anomaly detection in videos using recurrent neural networks. In: Proceedings of SAI Intelligent Systems Conference, pp. 155–169. Springer, ??? (2022)

[15] Hariprasad, Y., Iyengar, S.S., Chaudhary, N.K.: Securing the future: Advanced encryption for quantum-safe video transmission. IEEE Transactions on Consumer Electronics (2024)

[16] Zhao, Z., Li, X., Luan, B., Jiang, W., Gao, W., Neelakandan, S.: Secure internet of things (iot) using a novel brooks iyengar quantum byzantine agreement-centered blockchain networking (biqba-bcn) model in smart healthcare. Information Sciences **629**, 440–455 (2023)

[17] Lohachab, A., Lohachab, A., Jangra, A.: A comprehensive survey of prominent cryptographic aspects for securing communication in post-quantum iot networks. Internet of Things **9**, 100174 (2020)

[18] Rathore, S., Sharma, P., Paul, A.: Ai in forensic science: Emerging trends and applications. Forensic Science International **332**, 110–126 (2023)

[19] Smith, J., Liu, R.: Machine learning in digital forensics: A review of algorithms and case studies. Journal of Cybersecurity Research **29**(4), 255–270 (2022)

[20] Thompson, B., Garcia, E.: Ai-driven dna analysis in criminal investigations. Genomic Forensics Review **45**(2), 134–150 (2021)

[21] Standards, N.I., Technology: Ai in forensic science: Guidelines and challenges. Technical Report 2023-04, NIST (2023)

[22] Brown, H., White, M.: The future of ai in canine detection. Journal of Law Enforcement Technology **18**(1), 45–62 (2022)

Establishing Trust: A Techno-Legal Framework for Ensuring Digital Evidence Integrity

Naveen Kumar Chaudhary[1], S.S. Iyengar[2], Nilay Mistry[1]

[1]National Forensic Sciences University, Gandhinagar, India.
[2]Florida International University, Miami, 33199, Florida, USA.

Contributing authors: naveen.chaudhary@nfsu.ac.in; iyengar@cs.fiu.edu; Nilay.mistry@nfsu.ac.in;

Abstract

Three Bills namely the Bharatiya Nyaya Sanhita 2023 (BNS), the Bharatiya Nagarik Suraksha Sanhita 2023(BNSS), and the Bharatiya Sakshya Adhiniyam 2023 (BSA) was passed in the winter session of the parliament. After the assent of Hon'ble President of India three Acts have been notified in the Gazette of India on 25 Dec 2023 and promulgated from 01st July 2024. These three new criminal laws have repealed and replaced the Indian Penal Code 1860, the Code of Criminal Procedure 1973 and the Indian Evidence Act, 1872. The Bharatiya Sakshya Adhiniyam 2023 covers the provisions related to admissibility of digital evidence in the court of Law. The integrity of the digital content due to its probative value is the essential yardstick for admissibility. This paper examines the technical and legal implications for the admissibility of digital evidence as per the new laws and proposes a procedure for establishing the integrity of the electronic evidence through a proof-of-concept. The framework suggested in this paper provides reliable procedure for establishing the integrity of the digital evidence submitted by the custodian in-line with the techno-legal imperatives of the new criminal laws.

Keywords: BNSS, BNS, BSA, Certificate, Integrity, Hashing, Digital Evidence

1 Introduction

The increasing reliance on digital evidence in criminal investigations and judicial proceedings necessitates robust mechanisms to ensure its authenticity, integrity, and admissibility. The enactment of three new criminal laws in India—Bharatiya

Nyaya Sanhita (BNS) 2023, Bharatiya Nagarik Suraksha Sanhita (BNSS) 2023, and Bharatiya Sakshya Adhiniyam (BSA) 2023—marks a significant transformation in the Indian criminal justice system, replacing colonial-era statutes and embracing contemporary legal and technological realities. Of particular importance is the BSA 2023, which redefines the evidentiary status of digital records by classifying them as primary evidence, thereby underscoring the need for reliable standards for digital evidence integrity.

Prior to these reforms, the Indian Evidence Act, 1872, governed the admissibility of electronic records under Sections 65A and 65B, with significant ambiguities surrounding the structure and content of digital evidence certificates. The absence of a standardized format and explicit integrity verification protocols often led to challenges in court. The BSA 2023 addresses these limitations by prescribing a detailed certificate structure under Section 63(4)(c), incorporating cryptographic hash value verification as the foundational method to establish the integrity of electronic evidence.

This paper presents a comprehensive techno-legal framework aligned with the provisions of BSA 2023. It outlines a two-part certificate system involving both the custodian and an expert, clarifies the legal authority of experts under BNSS and IT Act 2000, and proposes a proof-of-concept implementation using standard hashing algorithms such as MD5, SHA256, and SHA512. The methodology ensures that digital records are verifiably unaltered, reinforcing their admissibility and probative value in court. Through this framework, the study bridges the gap between legal requirements and technological feasibility, offering a scalable and legally sound approach for handling digital evidence in modern criminal litigation.

2 Background

Section 65A and 65B were inserted in the Indian Evidence Act 1872 (IEA) via an amendment brought by the IT Act 2000 [1–3]. Section 65B, which deals with the admissibility of electronic evidence, was heavily drawn from Section 5 of the UK Civil Evidence Act, 1968. However, Section 5 of the UK Civil Evidence Act was repealed by the Civil Evidence Act 1995 following the recommendations of the Law Commission made in 1983 [4].

Since it is a digital record taken from the original source that is admissible under this special provision for examination by the court, the integrity of the digital content was a major concern. Section 65B(4) of the IEA covered the requirement with regard to a certificate [5], however, the format of the certificate was not prescribed. The certificate submitted under the provisions of Section 65B(4) was expected to be self-contained in regard to the integrity of the digital content taken from the original source. However, there was no provision in the certificate to ensure such integrity.

Thus, the certificate did not provide a foolproof mechanism for establishing the relevancy and accuracy of the digital content for its admissibility in the court of law. Considering these limitations and in order to ensure the integrity of digital content [6, 7], a relevant section has been incorporated into the new criminal laws, which establishes the integrity of the digital record by capturing its hash value.

3 Major Changes in the New Criminal Law for the Admissibility of Digital Evidence

The *Bharatiya Sakshya Adhiniyam* 2023 (BSA) replaces the Indian Evidence Act, 1872 (IEA) [8]. The IEA provided for two kinds of evidence: documentary and oral. Documentary evidence included primary evidence, i.e., original documents, and secondary evidence, which proves the contents of the original. The BSA retains this distinction.

Importantly, it includes electronic records in the definition of documents [9]. While the IEA categorized electronic records as secondary evidence, the BSA classifies them as *primary evidence*. It also expands the definition of such records to include information stored in semiconductor memory, communication devices, and digital records generated from smartphones, laptops, intermediaries, and cloud platforms [10].

4 Admissibility of Digital Evidence Under New Criminal Law

The *Bharatiya Sakshya Adhiniyam* 2023 (BSA) is a procedural law that provides the rules and regulations governing the relevancy and admissibility of evidence in court proceedings. It aims to remove ambiguities and help avoid probable intricacies that may arise during the admissibility of relevant electronic evidence.

Section 63(4)(c) of the BSA requires a certificate to accompany electronic records for admissibility. This certificate serves to identify the record and describe how it was produced, among other aspects. A model certificate is annexed in the Schedule of the Adhiniyam [11]. The certificate is intended to authenticate and verify the contents of electronic records, in line with Section 63(2) of the BSA.

Furthermore, the certificate submitted by the party under Section 63(4)(c) must be examined and verified by an expert, as per the same section. The certificate consists of two parts. **Part A** must be completed by the custodian of the device and captures information related to the custodian, identity of the electronic device, legal information, and the hash value of the digital record obtained from the original source device. The custodian may be the individual who owns, manages, or maintains the electronic device or digital record.

Part B is intended for the expert, who must verify all information regarding the electronic evidence, including the hash value furnished by the custodian. There are four major components in the certificate: (1) information about the custodian, (2) technical information, (3) legal information, and (4) the hash value of the digital record. These components are illustrated in Figure 1.

5 'Expert' as per the Provisions of New Criminal Law

As the certificate prescribed under Section 63(4)(c) of the BSA mandates verification of the information furnished under Part A by an expert, it is essential to examine the legal provisions determining who qualifies as an expert. In this regard, references were made to the *Information Technology Act, 2000* (IT Act 2000) and the *Bharatiya Nagarik Suraksha Sanhita, 2023* (BNSS 2023). It was observed that scientific experts from

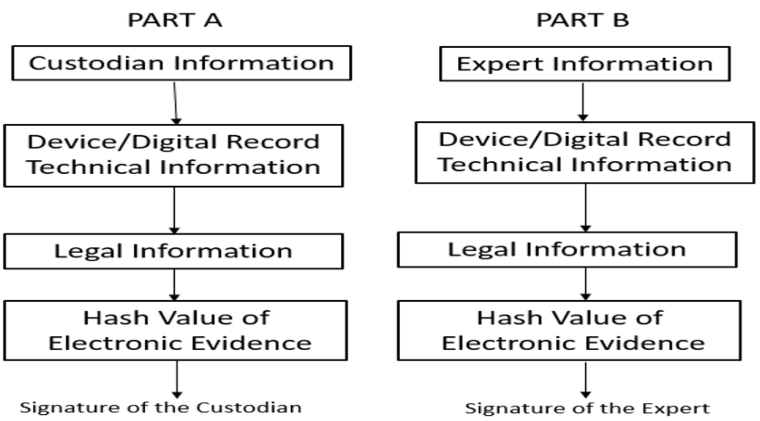


Fig. 1 Essential components of the Certificate prescribed under Section 63(4) of BSA

Cyber-Forensic Divisions of forensic laboratories, as notified under Section 293(4)(e) of the *Criminal Procedure Code* (CrPC) and under Section 79A of the IT Act 2000, are already engaged in examining, analysing, and reporting such cases [12, 13].

The provisions under Section 293(4)(e) of CrPC now correspond to Section 329(4)(e) of the BNSS 2023. Since both the IT Act and the BNSS govern admissibility of electronic or digital records, the qualifications for 'Expert' must align with Section 79A of the IT Act 2000 and Section 329(4)(e) of the BNSS 2023.

5.1 Section 79A of the Information Technology Act, 2000

Section 79A of the IT Act stipulates that the Central Government may, by notification in the Official Gazette, designate any department, body, or agency of the Central or State Government as an *Examiner of Electronic Evidence* for providing expert opinion in court or other legal forums. The Act defines "electronic form evidence" as any information of probative value that is stored or transmitted in digital format. This includes computer evidence, digital audio, digital video, cell phones, and digital fax machines.

5.2 Section 329 of BNSS 2023

Section 329 of the BNSS 2023 pertains to the reports of certain government scientific experts. Subsection 329(4) enumerates the experts qualified to submit such reports, including:

(a) Any Chemical Examiner or Assistant Chemical Examiner to the Government;
(b) The Chief Controller of Explosives;
(c) The Director of the Finger Print Bureau;
(d) The Director of the Haffkine Institute, Bombay;
(e) The Director, Deputy Director, or Assistant Director of a Central or State Forensic Science Laboratory;
(f) The Serologist to the Government;

(g) Any other scientific expert notified by the State or Central Government for this purpose.

6 Establishing Integrity of Digital Evidence

The information furnished in Part A of the certificate must be verified by the expert. A cryptographic *hash function* is a mathematical algorithm used to map data of arbitrary size onto data of fixed size. It is inherently a one-way function and produces a unique output for a given input, thereby enabling verification of data integrity [14–17].

Hash values are unique to digital records; hence, they serve as digital fingerprints. Because the process is non-reversible and deterministic, it is ideal for validating that a digital record has not been altered [18, 19].

Popular hashing algorithms used for this purpose include MD5, SHA-256, and SHA-512 [20]. A schematic of the proposed methodology to verify the integrity of digital records as per the new criminal laws is shown in Figure 2.

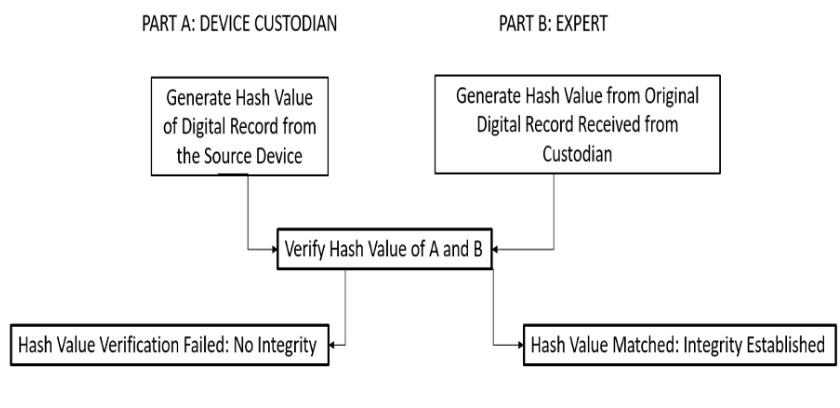


Fig. 2 Methodology to establish integrity of the digital record by the expert using hash value verification

7 Proof of Concept

The new criminal laws are forward-looking legal documents that incorporate technological advancements to enhance the judicial process. One such advancement is the requirement for establishing the integrity of electronic evidence through cryptographic hash values. According to Section 63(4)(c) of the BSA 2023, the custodian of the electronic evidence must generate and document the hash value in Part A of the prescribed certificate.

An experimental setup was created to test this process, using the tools listed in Table 1. This setup facilitated the generation of hash values from selected digital records, which were then verified by an expert to confirm their integrity.

Table 1 Tools used in experimental setup for integrity verification

Device/Tool	Description	Remarks
Computer	Windows 10 OS, Intel(R) Core i3-9100 CPU @ 3.60GHz, 12.0 GB RAM	Hash Value Generator was installed
Application for Hash Generation	HashMyFiles (NirSoft Tool)	Supports MD5, SHA256, SHA512
Digital Device	USB Drive, File System: exFAT	Digital record stored for verification
Nature of Digital Record	Image files (JFIF, JPG), Video file (MP4)	2 image files, 1 video file

8 Implementation of Proof of Concept

The generation and verification of hash values are core technology components of the digital evidence certification process under Section 63(4)(c) of the BSA. The procedure involves a two-step workflow:

1. **Generation:** The custodian generates the hash value using a certified application and records it in Part A of the certificate.
2. **Verification:** The expert independently regenerates the hash value from the submitted evidence and compares it against the custodian's entry in Part A.

This process is illustrated in Figure 3.

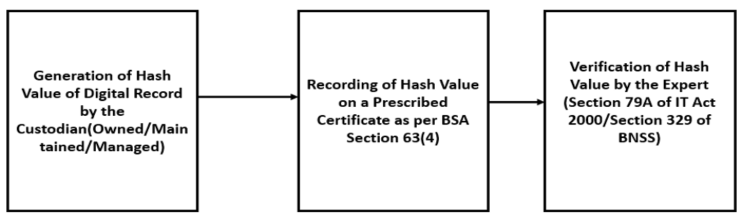


Fig. 3 Hash Value verification process

Three digital files—two images and one video—were processed as part of the proof-of-concept. Their hash values were computed and compared between the custodian and the expert, with results summarized in Table **??**.

9 Results of the Hashing Operation

The results of the hashing operation performed on three digital files are shown in Table 2. The metadata of these files are also included in the same table, along with their respective hash values. The hash values generated by the custodian align with the requirements prescribed in the certificate and match exactly with those generated independently by the expert, as required under Part B of the certificate.

Table 2 Summary of hash value verification for digital content

File	Algorithm	Custodian Hash (Part A)	Expert Hash (Part B)	Match
Art.jfif	MD5	979313...a79f3	979313...a79f3	Yes
	SHA256	964a1b...b1760	964a1b...b1760	Yes
	SHA512	50f950...ca562	50f950...ca562	Yes
Metaverse.jpg	MD5	204fed...e562	204fed...e562	Yes
	SHA256	8da057...b8ea	8da057...b8ea	Yes
	SHA512	45f6fa...9271	45f6fa...9271	Yes
Final_video.mp4	MD5	642113...3b8e	642113...3b8e	Yes
	SHA256	fc258e...53f5	fc258e...53f5	Yes
	SHA512	1b69d5...c4c4a	1b69d5...c4c4a	Yes

This hashing procedure was applied to files of different formats—two images and one video—each containing distinct digital content. Upon verification, the respective hash values matched completely, thereby establishing the integrity of the digital records. This integrity is crucial for the admissibility of electronic evidence in the court of law.

These results validate the effectiveness of the proposed two-step process to prove the integrity of digital evidence. The consistency in hash values demonstrates that the procedure is a robust, full-proof method in alignment with the techno-legal requirements as outlined in Section 63(4)(c) of the Bharatiya Sakshya Adhiniyam (BSA) 2023.

10 Conclusion

The adoption of advanced technologies is a defining feature of the new criminal laws in India. These reforms elevate the status of digital evidence, treating it at par with physical evidence in terms of probative value. Consequently, the admissibility of digital evidence now hinges on rigorous procedures to ensure its integrity.

The techno-legal framework proposed in this paper introduces a simplified yet effective procedure for verifying the integrity of electronic evidence. This framework is mandatory for admissibility and complies with the provisions set forth in the BSA 2023. By introducing a certificate-based system that incorporates hash value verification, the framework strengthens the legal validity of digital records.

Aligned with the modernizing intent of the new criminal laws, this approach leverages scientific technology to support justice delivery. Its adoption will not only simplify the admissibility process but also expedite legal proceedings in cases that rely heavily on electronic evidence.

References

[1] Kharb, D.: Admissibility of electronic records as secondary evidence under section 65b of the indian evidence act: Recent judicial approaches. Delhi Journal of

Contemporary Law **1** (2018)

[2] Karia, T., Anand, A., Dhawan, B.: The supreme court of india re-defines admissibility of electronic evidence in india. Digital Evidence and Electronic Signature Law Review **12** (2015)

[3] Yadav, A.: Relevancy and admissibility of digital evidence: A comparative study. International Journal of Integrated Law Review **1**(1) (2020)

[4] Christie, J.: The law commission and section 69 of the police and criminal evidence act 1984. Digital Evidence and Electronic Signature Law Review **20** (2023)

[5] Kumar, A.S.: Admissibility and Proof of Electronic Records. https://www.tnsja.tn.gov.in/article/electronic_records_article.pdf. Accessed on 07 Aug 2023

[6] Hosmer, C.: Proving the integrity of digital evidence with time. International Journal of Digital Evidence **1** (2002)

[7] Irons, A.: Digital forensics and measurement science. Journal of Measurement and Control **43**(8), 238–242 (2010)

[8] Verma, J.K.: Bharatiya Sakshya Adhiniyam, 2023 (Evidence): A Commentary, 1st edn. EBC Publishing Pvt. Ltd., ??? (2024)

[9] Bharatiya Sakshya Adhiniyam (BSA) 2023. Taxmann Publications Private Limited (2024)

[10] New Criminal Major Acts. Taxmann Publications Private Limited (2024)

[11] The Bharatiya Sakshya Adhiniyam, 2023, No. 47 of 2023. Available at https://wccb.gov.in/WriteReadData/UserFiles/file/Notificaton/BSA%202023.pdf, Accessed on 03 August 2024 (2023)

[12] Kumar, R.: Criminal Procedure Code CrPC. CBS Publishers and Distributors Pvt. Ltd., ??? (2023)

[13] IT Act 2008 (Amended). Available at https://www.meity.gov.in/writereaddata/files/it_amendment_act2008%20%281%29_0.pdf, Accessed on 27 July 2024 (2008)

[14] Bakhtiari, S., Safavi-Naini, R., Pieprzyk, J.: Cryptographic hash functions: A survey. Technical Report 95-09, Department of Computer Science, University of Wollongong (1995)

[15] Kumar, C.K., Suyambulingom, C.: Cryptographic of high security hash functions. International Journal of Engineering Research & Technology (IJERT) **1**(3) (2012)

[16] Rajeev, S., Geetha, G.: Cryptographic hash functions: A review. International Journal of Computer Science Issues **9**, 461–479 (2012)

[17] Andreeva, E., Mennink, B., Preneel, B.: Open problems in hash function security. Designs, Codes and Cryptography **77**(2), 611–631 (2015)

[18] Tenhunen, M.: The integrity of electronic evidence. In: Jajodia, S., List, W., McGregor, G., Strous, L. (eds.) Integrity and Internal Control in Information Systems. IICIS 1997. Springer, ??? (1997). https://doi.org/10.1007/978-0-387-35317-3_8

[19] Roy, M., Mohan, A.K., Sethumadhavan, M.: Maintaining the integrity of digital evidence using piecewise hashing. International Journal of Pure and Applied Mathematics **119**(16), 4073–4077 (2018)

[20] Conrad, E., Misenar, S., Feldman, J.: Secure Hash Algorithm, 2nd edn. Science Direct, ??? (2014)

Introduction to Age and trust behavior and Connection to Forensic Science

Vijeth Iyengar
Director, AARP

Abstract

Age and trust behavior have a direct connection to forensic science, particularly within forensic psychology and digital forensics, as they reflect fundamental neurological and behavioral changes that influence vulnerability to exploitation. Older adults, due to reduced activity in the brain's salience network and a tendency toward positivity bias, often struggle to accurately assess interpersonal trust, making them prime targets for fraud, financial abuse, and online deception. This neurobiological understanding enables forensic scientists to better profile perpetrators who prey on cognitively vulnerable individuals, refine digital evidence analysis related to such crimes, and develop age-specific investigative strategies. Furthermore, integrating neuroscientific insights into forensic interviews and assessments can improve evidence interpretation in cases involving elderly victims, ultimately strengthening legal outcomes and enhancing protections for aging populations.

Age-Related Changes in Interpersonal Trust Behavior

Can Neuroscience Inform Public Policy?

Vijeth Iyengar, PhD, Administration on Aging/Administration for Community Living; **Dipayan Ghosh, PhD,** John F. Kennedy School of Government, Harvard University; **Tyler Smith, BS, CFE, CAMS,** Federal Deposit Insurance Corporation; **Frank Krueger, PhD,** School of Systems Biology, George Mason University

July 1, 2019

An Evolving Global Age Distribution and New Implications for Social Cognition and Interpersonal Trust

In the years to come, there will be a significant global increase in the number of older adult persons. Some projections indicate that by 2030, there will be a higher number of adults age 60 or over than those between the ages of 10 to 24 [1]. It is critical to proactively address the novel challenges that societies will face with a shifting demography. In particular, understanding the neuropsychological changes that take place with advancing age and the effects these changes have on how older adults function and engage with their surroundings will become increasingly important in designing products, programs, and services to support the global population in the face of these inevitable new challenges.

In this paper, we link empirical findings from neuroscientific investigations of interpersonal trust behavior in older adults to incidences of financial exploitation, health care fraud, and digital deception—consumer harms for which older adults are preferentially targeted by bad actors.

Financial exploitation schemes include efforts to persuade older adults to provide access to personal financial accounts through mail, in-person, over-the-phone, or online spoofing. These schemes result in the theft or embezzlement of money or other property, and while the damage is difficult to estimate, some have reported that older adults lose about $36.5 billion each year to such financial abuse [2]. Similarly, health care fraud tactics exploit older adults through fraudulent billing via Medicare, private insurance, or personal funds. These older adults can also be subjected to unnecessary or unsafe medical procedures, resulting in compromised medical records. Lastly, digital disinformation, a more novel form of online victimization, takes advantage of social media platforms to spread misleading information that can affect a myriad of public interest concerns.

There are common problems across these forms of victimization. The potential victim faces the challenge of assessing the trustworthiness of the group or individual with which they interact. It can be difficult to judge the veracity and credibility of various sources of information. Finally, assessments of trust occur over different timelines: via one-time (e.g., unsolicited robocalls) or continuous (e.g., close friends or family members) engagement with individuals or groups harboring bad intentions. Determining the underlying reasons for these faulty evaluations of trust may prove useful in developing tools to combat these forms of victimization.

Emerging evidence from neuroscientific investigations of interpersonal trust behavior is revealing how the capacity to evaluate and subsequently act on untrustworthy agents with potentially nefarious intentions changes as we age. Interpersonal trust encompasses a person's willingness to be vulnerable to the risk of treachery based on the expectation that the actions of another will produce some future positive outcome due to the possibility of reciprocity. Available evidence from both cross-sectional and longitudinal survey studies has shown age-related changes in this behavior [3].

For example, older adults are less concerned with information that contradicts their first impressions about the trustworthiness of others, resulting in poor evaluations of the trustworthiness of other people [4].

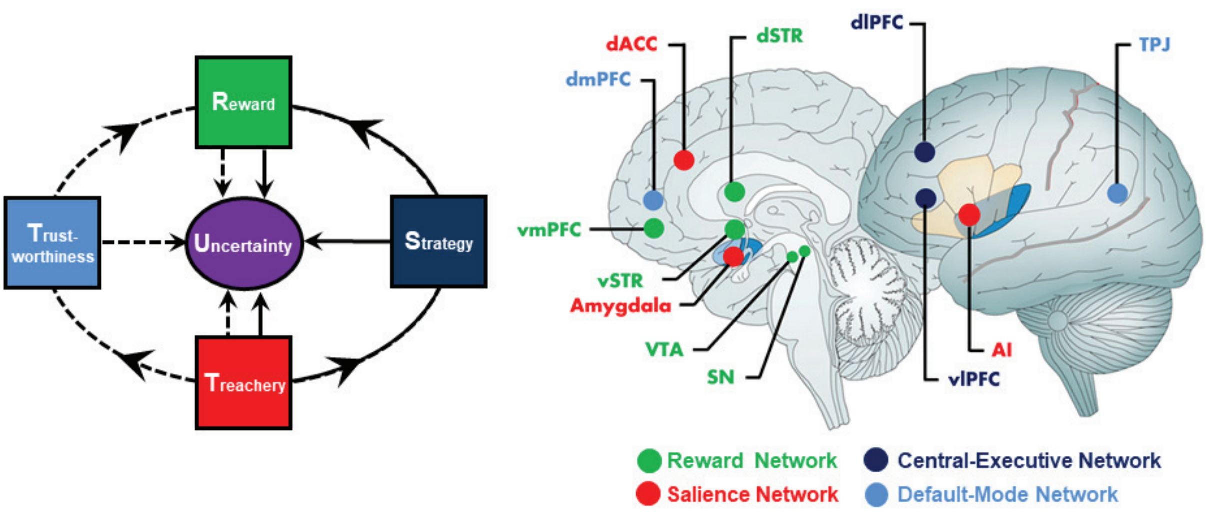
Further, in experimental economic exchange paradigms, wherein players assume the role of an investor (i.e., trustor) and responder (i.e., trustee), older adults engaged as trustors have been shown to be likelier to invest in trustees with an untrustworthy reputation (although the evidence is mixed) [5].

Complementing these behavioral findings, research in social cognitive affective neuroscience has linked age-related alterations in brain circuitry with changes in trusting behavior potentially leading to heightened susceptibility to financial fraud [6].

Collectively, these results—in older adults presenting with no neurological, medical, or mental conditions—suggest new public interest mechanisms may need to be developed to protect older adults from nefarious interactions. Recently, a neurobiological trust framework has been proposed in which the psychological components of trust (i.e., affect, cognition, and motivation) are linked to brain networks (see *Figure 1*) [7].

Mapping the Neurobiological Components of Interpersonal Trust

Making individual designations of trust in a social interaction—whether in the physical world or through an online interface—is the practice of successfully evaluating the potential benefits and costs when interacting with entities. Trusting an entity creates uncertainty, which results from the cost of potential treachery weighed against the anticipation of benefits after being trusted. Evidence suggests that older adults weigh costs and benefits differently from their younger counterparts [8]. Whereas older adults resemble younger adults in exhibiting increases in neural activity (in the brain's reward network) during the anticipation of benefits, they do not resemble younger adults in exhibiting increases in activity in the anterior insula (a component of the brain's salience network associated with evaluating the anticipation of costs and betrayal aversion). In fact, older adults have lower neural activity in this



Legend: AI, anterior insula; **dACC**, dorsal anterior cingulate cortex; **dlPFC**, dorsolateral prefrontal cortex; **dmPFC**; dorsomedial prefrontal cortex; **dSTR**, dorsal striatum; **SN**; substantia nigra; **TPJ**, temporoparietal junction; **vlPFC**, ventrolateral prefrontal cortex; **vmPFC**, ventromedial prefrontal cortex; **vSTR**, ventral striatum; **VTA**, ventral tegmentum area.

FIGURE 1 | Neurobiological Framework of Trust
SOURCE: Adjusted and reprinted with permission from Krueger, F., and A. Meyer-Lindenberg "Towards a model of interpersonal trust drawn from neuroscience, psychology, and economics," *Trends in Neurosciences.*
NOTE: Trust arises through the interplay of factors (t-r-u-s-t: treachery, reward, uncertainty, strategy, and trustworthiness)—linked to psychological components (i.e., affect, cognition, and motivation)—that engage key brain regions (circles) anchored in large-scale brain networks. Vulnerability from trusting another person builds uncertainty (purple ellipse) due to risk of treachery (red rectangle, affect, salience network), instead of anticipation of reward (green rectangle, motivation, reward network). To remove the uncertainty, the salience network engages either the central-executive network (dark blue rectangle) to adopt a context-based strategy, or the default-mode network (light blue rectangle) to evaluate trustworthiness for trusting a partner.

region when trying to recognize faces that displayed selfishness rather than cooperation—suggesting a tendency in older adults to overestimate the trustworthiness of others [4].

With these behavioral differences established, we return to the neurobiological trust framework [7]. Two different types of cognitive systems are crucial to the removal of uncertainty that comes with trusting another person: the social cognition system (default-mode network), to evaluate the trustworthiness of a partner, and the cognitive control system (central-executive network), to employ context-based strategies for trusting a partner. The social cognition system is essential in assessing whether to trust an individual or group and supports the ability to infer and attribute the intentions and traits of others. Trustors with higher perspective-taking tendencies not only show greater trust toward others but also reduce their trust more drastically after betrayal by others. Age-related changes in the default-mode network may negatively affect how older adults navigate their social environments, exposing them to nefarious actors.

Complementing this first system, the cognitive control system allows one to adopt goal-directed behavior under changing contexts. Accumulating evidence indicates that although some cognitive functions are affected as we age, others are spared. Specifically, whereas crystallized cognitive abilities (e.g., conceptual knowledge) are preserved, fluid cognitive abilities (e.g., executive control, working memory, and attention) steadily decline with age [8]. Consequently, older adults may experience particular difficulties when faced with trust decisions involving the simultaneous processing and evaluation of disparate pieces of information—a scenario that taps into *fluid cognitive abilities.*

In summary, based on the neurobiological trust framework, age-related findings in studies of interpersonal trust behavior are likely driven by the impairment of the affective component of trust in the salience network (but not in the motivational component in the reward network), which therefore can impact the socio-cognitive components of trust in default-mode network and central-executive network.

This conclusion about age-related changes in interpersonal trust behavior is in accordance with the general assumption of socioemotional selectivity theory [9], which proposes an increase of positive emotional and social experiences in ways that foster well-being with advancing age and narrowing time horizon. According to this theory, older adults demonstrate a positivity bias that enhances the salience of more positive than negative valanced information, increases attention to socioemotional cues, and improves memory for positive stimuli or events in later life [10]. While these changes are associated with emotional and mental well-being, this bias may lead to greater likelihood of victimization due to *excessive trust* afforded to individuals and groups. Although the present commentary specifically focuses on changes in trust behavior across the life span that could potentially lead to various forms of victimization, we do acknowledge other factors, such as declines in functional capacity and mental well-being, that may also contribute to an overall greater risk of victimization [11].

A Case for Policy to Combat Public Interest Harms Related to Changes in Interpersonal Trust Associated with Age

Mounting evidence from neuropsychology is uncovering the mechanisms by which older adults engage in interpersonal trust behavior. The development of predictive neural markers building on individual brain differences associated with age-related changes in this behavior may permit the identification of neural phenotypes that in turn serve as targets for interventions. Such approaches may inform the development of more targeted behavioral and neural interventions that incorporate cognitive capacities that are preserved across the life span. Understanding the mechanisms and conditions under which older adults differ from younger adults in their processing and evaluations of interpersonal trust behavior can also impact features of programs and policies. For example, law enforcement and regulatory entities operating in financial and healthcare industries can collect trust-related-behavior-based fraud indicators and incorporate relevant data into loss prevention internal controls in the service of protecting vulnerable older adults.

Further exploration into the various mechanisms through which older adults process different forms of fraudulent behavior may prove helpful. Indeed, while such fraud-related incidents as the digital disinformation problem have been shown to affect older adults more than others, there is scant analysis that charts how cognitive processes relating to disinformation in older adults differ from processes including financial exploitation. While both financial exploitation and disinformation are activities instigated by nefarious actors, they occur in divergent ways—e.g., in motivation, platform, and demands placed on the victim—which may affect how older adults respond to such instigations. Furthermore, ongoing policy, programmatic, and research efforts have not—to our knowledge—integrated the relative degrees of social pressure these various forms of fraud might instigate. Financial exploitation, for example, often involves an individual target, whereas the disinformation problem occurs on internet platforms and is disseminated to large classes with one mouse click.

Ultimately, we see this commentary as starting a conversation among stakeholders in the academic, public, and private sectors to identify how findings from neuropsychology can potentially inform and shape future public policies around issues that have touchpoints with interpersonal trust behavior. Greater engagement among these stakeholders has the benefit of cultivating an environment in which scientific investigators design questions that readily allow for translation of their findings to practice, the development of programmatic and policy interventions that incorporate the latest advances from research and practice, and, in the neuroscientific research community, the advancement of the importance of viewing their findings from a public health and policy lens.

References

1. *World population ageing 2017: Highlights.* 2017. New York: United Nations. https://www.un.org/en/development/desa/population/publications/pdf/ageing/WPA2017_Highlights.pdf (accessed April 28, 2019).
2. Huang, Y., and A. Lawitz. 2016. *The New York State cost of financial exploitation study.* Rensselaer, NY: Office of Children and Family Services. https://ocfs.ny.gov/main/reports/Cost%20of%20Financial%20Exploitation%20Study%20FINAL%20May%202016.pdf (accessed April 28, 2019).
3. Poulin, M. J., and C. M. Haase. 2015. Growing to trust: Evidence that trust increases and sustains well-being across the life span. *Social Psychological and Personality Science* 6(6):614-621. https://doi.org/10.1177/1948550615574301.
4. Suzuki, A., M. Ueno, K. Ishikawa, A. Kobayashi, M. Okubo, and T. Nakai. 2019. Age-related differences in the activation of the mentalizing- and reward-related brain regions during the learning of others' true trustworthiness. *Neurobiology of Aging* 73:1-8. doi:10.1016/j.neurobiolaging.2018.09.002.
5. Ruffman, T., J. Murray, J. Halberstadt, and T. Vater. 2012. Age-related differences in deception. *Psychology and Aging* 27(3):543-549. doi:https://doi.org/10.1037/a0023380.
6. Castle, E., N. I. Eisenberger, T. E. Seeman, W. G. Moons, I. A. Boggero, M. S. Grinblatt, and S. E. Taylor. 2012. Neural and behavioral bases of age differences in perceptions of trust. *Proceedings of the National Academy of Sciences of the United States of America* 109(51):20848-20852. doi:https://doi.org/10.1073/pnas.1218518109.
7. Krueger, F., and A. Meyer-Lindenberg. 2019. Towards a model of interpersonal trust drawn from neuroscience, psychology, and economics. *Trends in Neurosciences* 42(2):92-101. doi:https://doi.org/10.1016/j.tins.2018.10.004.
8. Samanez-Larkin, G. R., and B. Knutson. 2015. Decision making in the ageing brain: Changes in affective and motivational circuits. *Nature Reviews Neuroscience* 16(5):278-289. doi:10.1038/nrn3917.
9. Mather, M., and L. L. Carstensen. 2005. Aging and motivated cognition: The positivity effect in attention and memory. *Trends in Cognitive Sciences* 9(10):496-502. doi:https://doi.org/10.1016/j.tics.2005.08.005.
10. Reed, A. E., and L. L. Carstensen. 2012. The theory behind the age-related positivity effect. *Frontiers in Psychology* 3:339. doi:https://doi.org/10.3389/fpsyg.2012.00339.
11. Lachs, M. S., and S. Duke Han. 2015. Age-associated financial vulnerability: An emerging public health issue. *Annals of Internal Medicine* 163(11):877-878. doi:10.7326/M15-0882.

DOI

https://doi.org/10.31478/201906c

Suggested Citation

Iyengar, V., D. Ghosh, T. Smith, and F. Krueger. 2019. Age-related changes in interpersonal trust behavior: Can neuroscience inform public policy? NAM Perspectives. Commentary, National Academy of Medicine, Washington, DC. https://doi.org/10.31478/201906c.

Author Information

Vijeth Iyengar, PhD, is Brain Health Lead and Technical Advisor to the Deputy Assistant Secretary for Aging at the Administration on Aging/Administration for Community Living, US Department of Health and Human Services. **Dipayan Ghosh, PhD,** is Pozen Fellow at the Shorenstein Center on Media, Politics, and Public Policy, John F. Kennedy School of Government, Harvard University. **Tyler Smith, BS, CFE, CAMS,** is Special Agent in Charge of the Electronic Crimes Section, Office of Inspector General at the Federal Deposit Insurance Corporation. **Frank Krueger, PhD,** is Associate Professor of Systems Social Neuroscience, School of Systems Biology at George Mason University.

Acknowledgments

Francis X. Shen, JD, PhD, Executive Director, Massachusetts General Hospital Center for Law, Brain, and Behavior; and Associate Professor of Law, McKnight Presidential Fellow, and Faculty Member in Graduate Program on Neuroscience, University of Minnesota reviewed this work.

Conflict-of-Interest Disclosures

None to disclose.

Correspondence

Questions or comments should be directed to Frank Krueger at fkrueger@gmu.edu.

Disclaimer

The views expressed in this paper are those of the authors and not necessarily of the authors' organizations, the National Academy of Medicine (NAM), or the National Academies of Sciences, Engineering, and Medicine (the National Academies). The paper is intended to help inform and stimulate discussion. It is not a report of the NAM or the National Academies. The analysis, conclusions, and opinions set forth here are those of the author(s) alone and do not necessarily reflect the views of the Federal Deposit Insurance Corporation.

Beyond Normality: Rethinking Behavioral Biometric Data

Amith Kamath Belman[1][0000-0003-1008-3025], Frank Sicong Chen[2][0000-0003-3573-4795], Vir V. Phoha[2][0000-0002-5390-8253], and Pronab Mohanty[3]

[1] San José State University, San José, CA 95192-0249, USA
amith.kamathbelman@sjsu.edu
[2] Syracuse University, Syracuse, New York, 13244, USA
{schen154,vvphoha}@syr.edu
[3] DGP, Indian Police Service, Govt. of India pronabmohanty@gmail.com

Abstract. A common assumption in data used for popular behavioral analysis modalities like typing, gaits and swipes is that features extracted from the data follow the normal distribution. The assumption of normality impacts key facets of research, such as decisions of sampling techniques and classification models; and performance and results from the resulting systems. Through the analysis of eight open-access datasets collected on tablets and phones (gait, swipes, and typing), and desktops (typing), we question the assumption of normality in the extracted features. Using non-parametric normality tests (Lilliefors test and Shapiro-Wilk test), we test the null hypothesis "the test sample comes from a normal distribution" and examine features that have been popularly used in the literature from these activities. In most cases, less than 25% of the tested samples have p values > 0.05, which asserts that a majority of features do not follow a normal distribution. Although non-normality in keystroke latencies on the desktop has been shown in the literature, no previous work examined a large umbrella of biometric data, such as keystroke latencies, gait, and swipe data on desktops, phones and tablets. We also provide alternate solutions to address the non-normality in mobile or wearable device-based behavioral biometric data.
Our work raises the questions, "Should the *assumption of normality* be the *norm* in behavioral biometric modalities?". We posit that our results will change how behavioral data analysis is approached by emphasizing the validity of assumptions about the underlying data. This study can potentially impact a large body of work in desktop, mobile and wearable devices-based behavior analysis.

Keywords: Keystroke · Gait · Stairs · Swipe · Phone · Tablet · Desktop · Normality Assumption · Gaussian Distribution

1 Introduction

With the ubiquitous availability and spread of desktops, mobile devices-such as smartphones, tablets-and smartwatches, keyboards, both physical and on-screen, and touch screens have emerged as the most common interface to these devices. In addition, many applications use onboard inertial sensors, such as accelerometers and gyroscopes, for tasks related to health monitoring, gait analysis, etc.. Many studies assume the underlying distribution of features from keystrokes, swipes, accelerometer and gyroscope readings to be normal [12, 13]. Such assumptions are made for theoretical or calculational simplicity discussed in later sections. However, the effects of assuming an underlying normal distribution in data when it is not have been explored in various fields [86, 17, 18]. This misassumption can be a source of error in classifications when it influences the decision rules in the classifier, and in an adversarial approach lead to inefficient attacks if the generative model depends only on the mean and standard deviation.

We use a collection of open source benchmark datasets that include single-activity or single-device datasets [115, 116, 118, 117, 125, 126, 119], and a large multi-modality dataset (SU-AIS BB-MAS [19]), which consists of data from 117 users providing keystrokes data on desktop, phone and tablet; accelerometer and gyroscope data while walking, climbing upstairs and downstairs with a phone and a tablet; and touch screen swiping data on phone and tablet, for our experiments. We defined the null hypothesis H_0 to be "the test sample comes from a normal distribution" and performed a large number of Lilliefors tests and Shapiro-Wilk's tests on all modalities in these datasets. To summarize results, we categorize the features into four different categories: a) less than 25% of samples with p-value >0.05; b) 25% to 50% of samples with p-value >0.05; c) 50% to 75% of samples with p-value >0.05; and, d) above 75% samples with p-value >0.05. We find that except for the features from samples taken from climbing upstairs and downstairs data, almost all features from samples of all other activities were in the first category. It means that less than 25% of the samples cannot reject the null hypothesis (the sample comes from a normal distribution) at a 0.05 significance level.

Although many other fields have witnessed research work cautioning the naivety in the assumption of normality, this assumption has hardly been examined in behavioral biometric modalities. Lesser so in the case of multiple activities that span multiple devices. We present related literature, and briefly explore why studies assume normality in data, why these features depart from the normal distribution, and what distributions approximate best to these features. We also give brief descriptions of the eight datasets that we used and popularly used features extracted from these activities, explain our experiments in detail, analyze and discuss the experimental results, and conclude with the impacts of our findings.

1.1 Key Contributions

- We show, with suitable non-parametric tests, that the most commonly described features for mobile device-based biometric data from activities like typing, gait and swiping do not have an underlying normal distribution.
- We discuss why the assumption of normality is widely accepted in this field, why these features depart from the normal distribution, and perform the chi-

square statistics to get distributions that best approximate these features, which provide an overall perspective of what kind of distribution these features conform to.
- We perform our tests on the large multi-modality BBMAS dataset [19], as well as seven single-modality datasets for gait [115, 116], Swipe [118, 117], and Typing [125, 126, 119]. The results show that our findings are consistent and can be generalized.
- We discuss alternate approaches to mitigate or avoid the negative effects of assuming a non-existent normal distribution in data.
- We present the important implications of our findings for modeling distributions and classifier choices.

2 Motivation and related work

Behavioral biometrics includes a broad spectrum of modalities involving human behavior while performing day-to-day tasks. With the spread of mobile devices, such as smartphones and tablets, gait, keystrokes, and swiping patterns are the most explored biometrics with the interface to these mobile devices in recent years. Studies have focused on utilizing various aspects of these behavioral biometrics for authentication [100, 103], verification [106], continuous authentication [52], gender detection [54, 107], age detection [56], fatigue detection [55], mood disturbance detection [58], lie detection [57] and detection of various health conditions. Many of these works have either purposefully or inadvertently assumed an underlying normal distribution in the features extracted from the data, reflected in their analysis methods or their choice of classifiers. Although assuming a normal distribution helps simplify the problem, it may not always lead to the best results.

The adverse effects of assuming a normal distribution have been significantly studied in different fields. In the Constraint Satisfaction Problems (CSP) field, [17] proves that in the results produced by many heuristic combinations on random binary CSPs and 3-coloring problems, the benchmarks for CSP, the assumption of normality does not hold. The authors also appeal for statistics that do not rely on the normality assumption to analyze empirical results for CSP. In processes involving the classification of remotely sensed data from different spectral bands (image classification is a subset of this problem family), [18] showed that the brightness values distributions did not follow a normal distribution. They further remarked that this fallacy was a major source of error in land cover classification when decision rules employed in the classifier assumed an underlying normal distribution. In Dunning's [86] work on the statistical analysis of text, they pointed out that the assumption of normal distribution limits the ability to analyze rare events and that those rare events were a significant fraction of real text.

When dealing with the solution space for the economic design of X-bar control charts, [87] show that the non-normality assumption also has a more significant effect on the Type II error probability than the Type I error probability. In the research of Human Cancer Genomes, wrongly assuming a normally distributed Gene expression was shown to affect multiple facets, including identification of expression patterns, annotation, and classification [88]. They also concluded that small departures from normality were not analytically insignificant. [93] questioned the adequacy of data characterization using normal distributions and argued that an asymmetric view would increase recognition of data distributions and the quality of interpretation.

2.1 Why prior works might have assumed normal distribution in data?

We found two main concepts that many studies used to justify their assumption of normality in most datasets: the Central Limit Theorem (CLT) [89] and the Cràmer-Rao Lower Bound (CRLB) [90].

The popularity of the assumption of normality can be attributed to the fact that noise in many systems has been represented well using a normal distribution. The Gaussian assumption is a good conservative choice when not much is known about the data, which is also supported by CLT stating that the distribution of sample means tends to form a normal distribution as the sample size gets larger. A Gaussian distribution also minimizes the Fisher Information, which is the inverse of CRLB. In other words, the CRLB under the Gaussian distribution works for the worst-case scenario, maximizing the CRLB (see [15]). Therefore, minimizing the largest CRLB is interpreted as min-max optimal [16]. To relate our discussions and analysis, we briefly introduce both of these concepts below.

Central Limit Theorem

If $(X_1, X_2, ..., X_n)$ is a random sample of size n taken from a population with mean μ and finite variance σ^2, and if $\bar{X}$ is the sample mean, the limiting form of the distribution of $Z = (\bar{X} - \mu)/(\sigma/\sqrt{n})$, as $n \to \infty$, is the standard normal distribution (see Theorem 7.2 in [108]).

It is often assumed that estimated population means and variance are independent for simplicity, implying sample variance means and variance are independent, which is valid only for normal distribution.

Many research works inadvertently back on CLT for their modeling choices. A good example of this scenario is the use of Gaussian Mixture models for keystroke analysis [110–112] and touchscreen swipe analysis [113, 114]. A Gaussian Mixture model assumes the data follows a mixture of individual multivariate Gaussians or Gaussian Mixture distribution, which is only a good choice if the data is a mixture of Gaussians and has a sufficiently large number of samples, implicitly invoking CLT.

Fisher Information and Cràmer-Rao Lower Bound

Fisher information for a random sample from $f(x|\theta)$, where θ is an unknown parameter and $n \to \infty$, is expressed as $I(\theta)$ = -E$[(\partial^2/\partial\theta^2)\ \log f(X|\theta)]$, (see Theorem 5.8 [109]).

If $\hat{\theta}$ is an unbiased estimator of θ, and [90] states that the variance of $\hat{\theta}$ is bounded by the reciprocal of Fisher Information, i.e. $var(\hat{\theta}) \geq 1/(nI(\theta)$, using CLT, it follows that as sample size tends to infinity, the maximum likelihood estimator is asymptotically unbiased and asymptotic distribution of $\hat{\theta}$ is normal. i.e. for a true value θ_0 for θ , $\sqrt{nI(\theta_0)}(\hat{\theta} - \theta_0) \sim N(0, 1)$.

Some examples from literature

The Gaussian assumption is often applied in the modeling of keystroke latencies. For example, the work in [12] modeled the latencies of 142 pairs of characters under the assumption that, "probability of the latency y between two keystrokes of a character pair forms a uni-variate Gaussian distribution", and thereby deriving the μ and σ parameters for each character pair, to be used further down for information gain estimation. Similarly, [13] started their bot simulations under the assumption that the keystroke duration of a character in a word is modeled as a random variable which is Gaussian or a constant with additive uniform noise. Thus, their typing event injections for synthetic forgeries or bot attacks were already under assumptions that should not be made casually.

2.2 Prior works that questioned the assumption of normality in behavioral biometrics

In [6], the authors examined keystroke features such as Key Interval Times (Digraph) and Key Hold Times (Unigraph) extracted from keystroke data recorded on desktops to test if they followed a normal distribution. The authors performed Lilliefors test [1] and Cramer-von Mises test [91] and established that Key Hold Times and the Key Interval Times from desktop typing did not follow a Gaussian distribution for all Unigraphs and Digraphs. As research in keystroke dynamics is a popular component of behavioral biometrics, this can be considered a pioneering step toward questioning the normality assumption in behavioral biometrics.

2.3 Why does the distribution in data depart from the normal distribution?

Reasons for why the data might not conform to the normal distribution can be approached from two directions.

First, the data collection process. There can be various factors that affect the way that data is collected. For instance, most sensor-based gait data come from the accelerometers and gyroscopes inside mobile devices or IMUs (Inertial measurement units). However, the way or the position that the device is put in can have influences on the data collected from these sensors. For example, the phone is put in the pocket for some of the datasets, while it is held in hand for some other datasets. As shown in Figure 1c and Figure 1f, both of these figures are the histogram of the energy of the x-axis. However, they show different distributions due to different places the phone was put while collecting data. Figure 1c shows the histogram of "x-energy" values extracted from the phone in hand. We can see that nearly all the values in the range of (0,0.2). However, the histogram of "x-energy" values extracted from the phone put in the pants' pocket shows that there are around 1/3 of values in the range of (0.2, 0.4), as shown in Figure 1f. It results in different distributions that approximate values of this feature. Also, the data from the accelerometers and gyroscopes might vary significantly even though values are recorded at specific frequencies as people walk at different speeds.

For keystroke data, the type of the keyboard can influence the speed that people type: a higher keycap might take someone more time to press the key, while other people might be used to it and type as usual. Additionally, typing on the screen significantly differs from typing on the physical keyboard. A slight touch is sufficient for the screen to capture the keystroke, resulting in the keyhold time concentrating in a very small range.

For swiping data, the screen size and the device's orientation can affect the length or the duration that people can swipe. The finger people like to use for swiping and the hand people use to hold the device also matter. All these factors can affect the data collection process and result in the samples not being normally distributed.

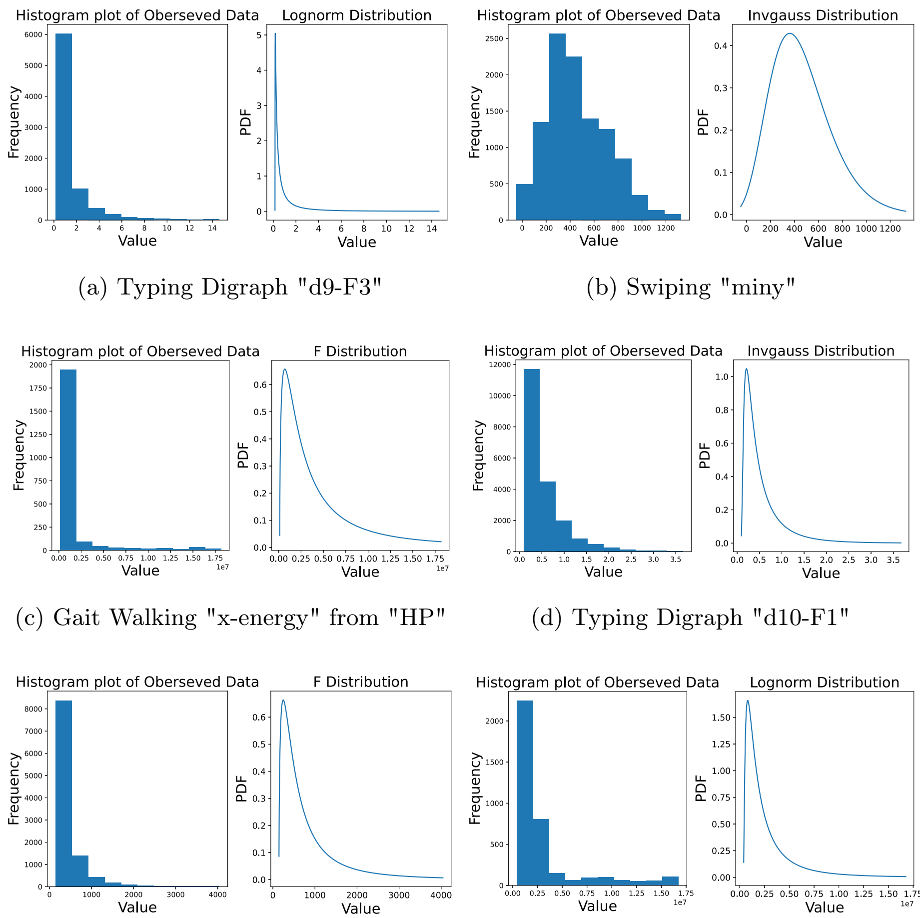
Second, some of the features are directly extracted from the raw dataset, while other features are calculated features. Sometimes, these calculated features do not conform to known distributions. For example, in the swiping feature sets, the coordinates of the starting point and the ending point are extracted from the raw data, while the first, second, and third quantiles for velocity are statistical variables. In fact, statistical variables can be from any distributions. There are also similar calculated features in gait feature sets.

2.4 What distributions best approximate these features if they do not conform to the normal distribution?

Our discussion from Sections 2.2 and 2.3 naturally leads to the question: "What distribution do we expect to get if the data does not conform to the normal distribution?". Since histograms give us an approximate representation of how the samples are distributed, we plot the histograms for some of the features and find that they conform to many different types of distributions besides normal distribution. In Figure 1, histogram plots are shown on the left side of each figure.

We use the chi-square statistics to approximate the probability distribution function for each feature. The chi-square (χ^2) statistic provides a way to test how well a sample of data matches the (known or assumed) characteristics of the larger population that the sample is intended to represent. The formula for chi-square statistics is:

$$\chi_c^2 = \Sigma \frac{(O_i - E_i)^2}{E_i}$$



(a) Typing Digraph "d9-F3"

(b) Swiping "miny"

(c) Gait Walking "x-energy" from "HP"

(d) Typing Digraph "d10-F1"

(e) Swiping "tottime"

(f) Gait Walking "x-energy" from "PP"

Fig. 1: Histogram plots and the probability density function plots that are approximated by chi-square statistics. (a) Typing data on hand-phone, Flight3 time for pair ("e","s") (d_9) Digraph, approximated by Log-normal distribution; (b) Swiping data, minimum value of y coordinates, approximated by Inverse Gaussian distribution; (c) Gait data, energy value of x-axis of accelerometer signal from hand-phone while walking, approximated by F-distribution; (d) Typing data on hand-tablet, Flight1 time for pair ("o","SPACE") (d_{10}) Digraph, approximated by Inverse Gaussian distribution; (e) Swiping data, total time for completing a swipe, approximated by F-distribution; (f) Gait data, energy value of x-axis of accelerometer signal from pocket-phone while walking, approximated by Log-normal distribution.

where c is the degree of freedom, O is the observed value and E is the expected value.

There are two main kinds of chi-square tests: the test of independence and the goodness-of-fit test, we use the latter. We select several common distributions to approximate each feature for all subjects. Then we select the most frequent distribution that appears as the distribution of each feature it conforms to. In Figure 1, the approximated probability density functions (PDFs) are shown on the right side of each figure. For example, in Figure 1a, the histogram plot shows that nearly all of the samples are in the range of (0,4), while there are still a few samples that do not in this range. It suggests that it does not conform to the normal distribution. By using the chi-square statistics, it is best approximated by the Log-normal distribution, which is shown on the right side. The most frequent distributions that we get for each modality are shown in Table 1. The distributions we selected for matching include: Normal distribution, Uniform distribution, Exponential distribution, Beta distribution, Gamma distribution, Cauchy distribution, F-distribution, Student's t-distribution, Chi-square distribution, Inverse Gaussian distribution, Log-normal distribution, Weibull minimum distribution, and Weibull maximum distribution.

Table 1: Top ranked distributions that are approximated by Chi-square statistic for each modality. The number in brackets are the number of features that conform to this kind of distribution. (Acc: Accelerometer, Gyr: Gyroscope, DT: Desktop, HP: HandPhone, PP: PocketPhone, HT: HandTablet)

Activity	Feature set		Top ranked distributions
Gait (44)	Walk	Acc	HP: t (8), PP: Weibull Min (11), HT: Beta (8)
		Gyr	HP: Inverse Gaussian (16), PP: Weibull Min (13), HT: Log Norm (16)
	Upstairs	Acc	HP: Beta (12), PP: Beta (7), HT: Weibull Min (8)
		Gyr	HP: Beta (13), PP: Beta (20), HT: Inverse Gaussian (13)
	Downstairs	Acc	HP: Beta (8), PP: Chi Square (7), HT: Beta(18)
		Gyr	HP: Beta (10), PP: Beta(19), HT: Beta(18)
Swiping (27)	Phone		Cauchy (4), Inverse Gaussian (4)
	Tablet		Weibull Min (4), Beta (4), Cauchy (4)
Typing	Unigraph (12)	DT	Weibull Max (5), t (4)
		HP	t (7)
		HT	t (10)
	Digraph (72)	DT	Weibull Min (16), Log Norm (14), Inverse Gaussian (14)
		HP	Weibull Min (14), Inverse Gaussian (13), Beta (12)
		HT	Weibull Min (21), Inverse Gaussian (11), t (8)

2.5 How our analysis differs from related work discussed in Section 2.2

Although many other fields have questioned the naivety of the assumption of normality, the field of behavioral biometrics has been lagging in this aspect. We found that a Section (5.1) of the research in [6], was the only exploration in this direction for keystroke dynamics on desktops. Behavioral biometrics is an emerging area with many different modalities, devices and activities. For example, typing is no more limited to a desktop, keystroke dynamics have been studied separately on phones and tablets and other touch devices; gait as a behavioral biometric has been studied with different sensor placements, different sub-activities and different devices; and, swiping patterns for behavioral biometrics have been studied on different smart-screens or touch surfaces.

We examine the assumption of normality, in a large dataset consisting of a wide range of activities and devices, as well as seven single-modality datasets collected from different mobile or wearable devices. By performing a large number of statistical tests, we show the following:

- Features extracted from accelerometers and gyroscopes in tablets and phones (in the pocket and in hand) do not follow a Gaussian distribution while freely walking on a flat floor. However, in a large number of samples from activities of climbing up and down the stairs, and walking on a treadmill at fixed speeds, the null hypothesis, that the data comes from the normal distribution, cannot be discarded.
- Swipe features extracted from swipe trajectories, pressure, acceleration and touch area data do not follow normal distribution on smart-phone and tablet surfaces.
- Keystroke features on mobile devices might have different distributions from that on desktops. On mobile devices (on-screen keyboard), keyhold times are more likely to conform to the normal distribution, while most flight times do not. However, these features on the desktop (physical keyboard) do not follow the normal distribution based on our non-parameter test. Our experiments reaffirm the findings in [6] that showed the non-normal nature of keystroke features on desktop. We observe the same occurrence in keystroke features, such as flight times, from phones and tablets, which have not been examined in the literature.
- Implications of our findings and alternate approaches to handle non-normal distributions in data.

3 Description of test datasets and popularly used features in gait, swipe, and keystroke

In this section, we briefly describe a multi-modality dataset (SU-AIS BB-MAS [19] dataset) and seven single-modality datasets [115, 116, 118, 117, 125, 126, 119], as well as the features that we have analyzed. These features are the most popularly used features from each of the modalities.

Table 2: The brief description of the SU-AIS BB-MAS [19] dataset, which includes the different data types we analyzed from multiple devices and activities. The gait activity consists of three sub-activities, walking, climbing upstairs, and downstairs.

Activity	Sensors / Interface	Devices	Details
Gait	Accelerometer Gyroscope	Phone in Pocket	2 sessions / user, $\approx$ 5 min./ session
		Phone in Hand	Phone in Pocket for both sessions
		Tablet in Hand	Phone in Hand and Tablet in Hand for one session
Swiping	Touchscreen	Phone	$\approx$ 25 minutes with typing
		Tablet	$\approx$ 25 minutes with typing
Typing	Keystroke	Desktop	$\approx$ 50 minutes with mouse usage
		Phone	$\approx$ 25 minutes with swiping
		Tablet	$\approx$ 25 minutes with swiping

3.1 Description of the multi-modality dataset (SU-AIS BB-MAS [19] Dataset)

As [92] has described all aspects of this dataset in great detail, we only provide a gist of this dataset in this section. A total of 117 users participated in the data collection and performed several activities on different days. The activities included typing and mouse clicking on the desktop, typing and swiping on phones and tablets, and walking, climbing upstairs and downstairs with phones and tablets in their pants' pockets and hands. The dataset consists of a total of about: 3.5 million keystroke events, 57.1 million data points for accelerometer and gyroscope each; 1.7 million data points for swipes; and enables researchers to explore previously unexplored directions in inter-device and inter-modality biometrics.

3.2 Description of seven single-modality datasets

These seven datasets [115, 116, 118, 117, 125, 126, 119] are single-activity or single-device datasets that have been elaborately described in the literature, we only provide a brief description of each of them. Please see Table 4 for more details.

Description of two gait datasets ([115] and [116])

The gait dataset collected in [115] consists of walking data from 20 subjects performing two separate 15-minute walks on two different days with a mobile phone in their pocket. The accelerometer and magnetometer in the mobile phone collected the data while the subjects were walking. This dataset is very similar to the walking (on a flat corridor) phase in BBMAS dataset [19]. The other dataset, collected in [116], has data from 21 healthy heel-striking subjects walking on a split-belt treadmill at 12 different walking speeds in the interval [0.6,1.7]m/s with 0.1m/s increments, and each speed maintained for one minute. The subjects wore shoes equipped with eight markers (sensors) to keep track of the movement of these markers.

Description of two swipe datasets ([118] and [117])

[118] collected a dataset that consisted of swiping data from 66 users performing at least 80 swipes each on a smartphone. They also collected another swipe dataset in [117], which has data separated into two categories based on the device's orientation. Around 100 participants gave swipes in portrait mode, and 45 gave swipes in landscape mode, with about 80 swipes per participant.

Description of two typing datasets ([125] and [126]) on mobile devices and one typing dataset ([119]) on desktops

The two typing datasets on mobile devices are collected in [125] and [126], and the typing dataset on desktops is collected in [119]. All these datasets asked participants to type ".tie5Roanl", which is chosen to be a representative of strong passwords in [127].

The dataset in [125] consists of 51 users with 51 trials per user. The other dataset in [126] consists of data from 54 participants. Each participant performed at least three sessions and provided at least 60 trials in total. Though participants were asked to type the same password on mobile phones, the keyboard layout for these two datasets is not the same, resulting in different unigraph and digraph features. The dataset in [119] consists of 51 users typing the same password on a physical keyboard in eight different sessions, for a total of 400 samples per user.

3.3 Description of features that we analyzed in gait, swipe, and keystroke

The feature extraction for this dataset can be grouped into three parts, namely gait , swipe, and keystroke features. For each modality, we selected the most popular features that have been studied for several years [121] [122] [123] [124]. We briefly describe the features and their storage below. A summary of the features is presented in the Table 3.

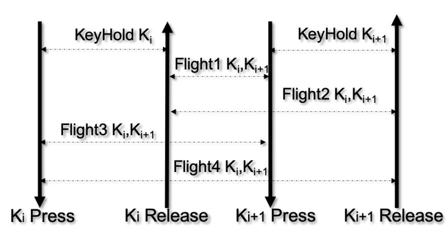


Fig. 2: Features extracted from keystroke data.

Table 3: List of features extracted and examined in our experiments for an underlying normal distribution.

Data	Features	Details
Gait	•Mean •Standard Deviation •Band Power •Energy •Median •Inter Quartile Range •Range •Signal to Noise Ratio •Dynamic Time Warp Dist. •Mutual Information •Correlation	−Features were extracted from x, y, z and m (m=$\sqrt{x^2+y^2+z^2}$) signals from both the accelerometer and gyroscope. −All features were extracted from each of the directional signals except for DTW distance, Mutual Information and Correlation which were extracted between pairs of these signals, i.e., x-y, x-z, x-m, y-z, y-m and z-m.
Swpie	•Min. x and y coordinates •Max. x and y coordinates •Euclidean Distance •Angle of the swipe •Time •Velocity Mean and Std. •Velocity Quartiles •Acc. Mean and Std. •Acc. Quartiles •Pressure Mean and Std. •Pressure Quartiles •Area Mean and Std. •Area Quartiles •Direction	−Features were extracted from a variety of information making up a swipe. −Features such as coordinates, angles and direction are dependent on the touch points on the screen and the end points of a swipe. −Velocity, Pressure, Acceleration and Area are calculated with the data from corresponding sensors on the touch surface of the devices. −The Direction feature was only used to group the swipes into vertical and horizontal swipes.
Typing	•Keyhold •Flight 1 to Flight 4	−Keyhold times were extracted from twelve most occurring unigraphs. −Flight times were extracted from eighteen most occurring digraphs.

– **Gait Features**: As the raw data for the gait is a pair of signals from the accelerometer and gyroscope we extract features from both. The gait data is further subdivided into three activities; "Walking" (on a flat corridor); "Downstairs" (going down the staircase); and, "Upstairs" (going up the staircase). For both accelerometer and gyroscope, they keep track of the movement along three basic directions, x, y, and z. Based on these three directions, we can calculate an overall direction, called m, which we have $m = \sqrt{x^2 + y^2 + z^2}$. We use a window size of two seconds with a one second overlap between two consecutive windows. For each two second window we extract a host of features from the accelerometer and the gyroscope along each of these four directions. In general, statistical variables can describe some characteristics of the data, so we extract Mean, Standard deviation, Median frequency, Inter quartile range, Range from four directions separately. These data are time-series data, so Band power, Energy, Signal to noise ratio can be extracted from these four directions. Besides, we calculate Dynamic time warping distance between pairs of signals x-y, y-z and x-z, Mutual information is calculated between pairs of signals x-y, x-z, x-m, y-z, y-m and z-m, and Correlation coefficients are calculated between pairs of signals x-y, y-z and x-z. The list of features extracted is given in Table 3.

- **Swipe Features**: For each swipe performed by users on tablet and phone, various features related to the speed and trajectory of the swipes are extracted. The last row of Table 3 summarizes them. While people swipe on the screen, it records the coordinates of the finger on the screen at certain frequencies, so it's natural to extract features as the minimum and the maximum x and y coordinates; the Euclidean distance between the start and stop points; the tangent angle of the swipe; the total time taken to finish the swipe; the mean and standard deviation and the quartiles of velocity, acceleration, pressure and area between the finger and screen; and the direction of the swipe used to group them into horizontal or vertical swipes.
- **Keystroke Features**: We select the most occurring twelve unigraphs (single key) and eighteen digraphs (pair of consecutive keys) that occurred the most number of times in all user's keystroke data.
  The unigraphs are : "BACKSPACE", "SPACE", "a", "e", "h", "i", "l", "n", "o", "r", "s" and "t".
  The digraphs are: ('BACKSPACE', 'BACKSPACE'), ('SPACE', 'a'), ('SPACE', 'i'), ('SPACE', 's'), ('SPACE', 't'), ('e', 'SPACE'), ('e', 'n'), ('e', 'r'), ('e', 's'), ('n', 'SPACE'), ('o', 'SPACE'), ('o', 'n'), ('r', 'e'), ('s', 'SPACE'), ('s', 'e'), ('t', 'SPACE'), ('t', 'e') and ('t', 'h').
  For a unigraph K_i we extract the $Keyhold$ time of the key as a feature:
  - $Keyhold_{K_i}$: $K_iRelease$ - K_iPress

  For a digraph K_i and K_{i+1} the following temporal features are extracted:
  - $Flight1_{K_iK_{i+1}}$: $K_{i+1}Press$ - $K_iRelease$
  - $Flight2_{K_iK_{i+1}}$: $K_{i+1}Release$ - $K_iRelease$
  - $Flight3_{K_iK_{i+1}}$: $K_{i+1}Press$ - K_iPress
  - $Flight4_{K_iK_{i+1}}$: $K_{i+1}Release$ - K_iPress

  Figure 2 illustrates the temporal features extracted form keystrokes.
  Outlier removal for Keystroke Features: We use a simple filter to remove any instances of keys that were held down for two seconds or more. We also remove instances of the inter-key pauses that are greater than two seconds. We assume that these were caused by pauses, where the user is either thinking or receiving instructions during the data collection.

4 Non-parametric tests and experimentation

This section describes the tests and procedures used to examine the hypothesis that a sample comes from the normal distribution.

Non-parametric tests: To test if the feature values have an underlying normal distribution, we use two non-parametric tests of normality namely: **a)**

Lilliefors test [1], a modified form of Kolmogorov–Smirnov test [3] that is suitable for large datasets for non-parametric testing of the null hypothesis that the data comes from a normally distributed population; and **b) Shapiro-Wilk test** [2] which is also a non-parametric test and is more suitable for testing the null hypothesis on a smaller data (n $<$ 50).

Hypothesis testing: For all tests we begin with the null hypothesis H_0, that the sample comes from a normal distribution. If the p-value from the test is below the critical value of 0.05, we can say that H_0 can be rejected, which means that the test sample does not conform to a normal distribution.

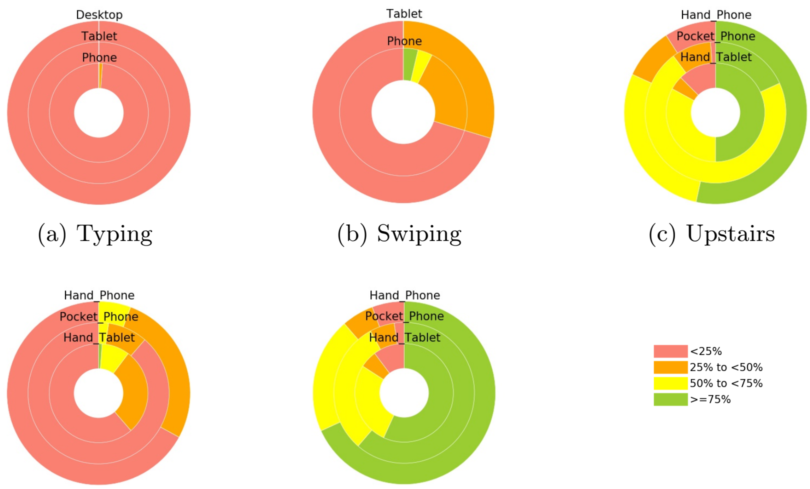
Sampling and Testing Procedure: In our tests, we draw random samples consisting of 75% of the feature set for each modality and for each user. Since the patterns for each user differ from others, and we are concerned with individual patterns, we perform the test for each feature within-user. We then perform a suitable normality test (Lilliefors test if the number of samples $>$ 50 or Shapiro-Wilk test otherwise) and store the p-values with corresponding annotations. We repeat the process ten times for all features, modalities, and users to arrive at more accurate conclusions for evidence of normality in the underlying distribution. This form of random sampling is inspired by much other statistical research ([4–6]), which shows that it is hard for goodness-of-fit tests to provide meaningful results on large datasets due to their loosely fitting statistical descriptions.

Grouping of Features: As the tests help us to reject H_0, based on the percentage of samples that had p-value $>$0.05, we categorize features into four broad categories; a) less than 25% of samples with p-value $>$0.05; b) 25% to 50% of samples with p-value $>$0.05; c) 50% to 75% of samples with p-value $>$0.05; and, d) above 75% samples with p-value $>$0.05. The values have been color-coded based on these categories for visual clarity.

5 Analysis on the multi-modality dataset - SU-AIS BB-MAS dataset

As shown in Figure 3, for a majority of test samples for features in gait, swiping, and typing, we found that H_0 could be rejected at the significance level of 0.05, implying the samples did not conform to a Gaussian distribution.

Samples from gait activity were further divided into walking (on a flat corridor), downstairs, and upstairs samples before testing each for normality. We found that the majority of the features of the walking tasks belonged to the first category and had less than 25% samples with $p > 0.05$. However, upstairs and downstairs samples had a considerable number of features in the third and fourth categories: 50% to 75%; and above 75% samples where the H_0 could not be rejected respectively. This phenomenon was observed on all three devices used for these tasks: Hand-Phone, Pocket-Phone, and Hand-Tablet. We posit that this could be due to individuals walking at different paces on a flat floor. Thus, there might be differences between step sizes, as some may be large and others may be short. In contrast, when people go downstairs or upstairs, they are more likely to keep each step the same since they need to maintain their balance



(a) Typing (b) Swiping (c) Upstairs

(d) Walking (e) Downstairs (f) Legend

Fig. 3: Illustration summarizing the amount of features in each activity and percentage of their samples with $p > 0.05$, or in other words, where the null hypothesis H_0, that the samples came from a normal distribution could not be discarded. The categories and their corresponding color codes are; red- less than 25% of samples with p-value >0.05; orange- 25% to 50% of samples with p-value >0.05; yellow- 50% to 75% of samples with p-value >0.05; and, green- above 75% samples with p-value >0.05. A full doughnut in the doughnut chart represents all the features for an activity on the labeled device. For example, the outer most doughnut in Figure 3a, represents all the features examined for keystrokes latencies on desktop, the second doughnut for tablet and innermost doughnut for phone respectively. The area covered by each color/category on a doughnut represents the amount of features that in the color/category as described above.

to avoid falling on the stairs, increasing the possibility of data conforming to the normal distribution. Individual values for the features of walking, upstairs and downstairs are shown in Appendix A, in Tables 10, 11 and 12 respectively. Their summary is illustrated in Figures 3d, 3e, and 3c, respectively.

In the case of swiping activity, we first separated the samples into vertical and horizontal swipes before testing the features for normality. We found a negligible amount of them belonged to the fourth category (above 75% samples where $p > 0.05$) on phones. However, most of the features belonged to the first and second category of: less than 25%; and, 25% to 50% of samples with $p > 0.05$ respectively. Table 9 in Appendix A, shows the results for individual features, and Figure 3b summarizes the categorical distribution for both phones and tablets.

In the case of keystroke features, we found that all but one feature was in the first category with less than 25% samples where H_0 could not be rejected.

In table 5, we observe that the highest percentage of samples where H_0 could not be rejected was for unigraph "n", with about 27%, which is still a very low number of tests when compared to the total number in our experiments, and thus can be ignored. As the case with desktop was an expected result following the lines of work in [6], these results reaffirm the findings and extend it further to hand-held devices such as phones and tablets. The detailed results for each feature and device are shown in Table 5 to Table 8 in Appendix A. While Table 5 shows the percentage of tests with $p > 0.05$ for unigraphs on phones, tablets, and desktops, Table 6 to 8 show the results for digraphs on desktops, tablets, and phones respectively. Figure 3a summarizes all the features of typing activity with respect to their categories.

From our results, it is clear that most features extracted from the behavioral biometrics data for gait, swiping, and typing activities on mobile devices, such as phones and tablets, do not follow a normal distribution. An elaborate discussion on the implications of our results and alternate approaches of non-normality is shown in Section 7.

6 Analysis on single-modality datasets

To further substantiate our observations discussed in the previous sections, we keep the null hypothesis H_0 that "the sample comes from a normal distribution" and perform similar analyses on six single-modality datasets collected from mobile or wearable devices, and one typing dataset collected from desktops. For datasets collected from mobile or wearable devices, we choose two datasets for each type of modality (gait, , swipe, and keystroke). Our observations from these single-modality datasets were similar to the ones we made on the BBMAS dataset. The datasets and observations are summarized in Table 4. Elaborate details and observations follow.

6.1 Analysis on two gait datasets

For the first gait dataset [115], most features from both sensors into the first category, which has less than 25% samples with p > 0.05. It shows that we can reject H_0 at a significance level of 5%. This result is consistent with our observation on our dataset [19]. For the second gait dataset [116], we found that a large proportion of features belong to the third and fourth category: 50% to 75%; and above 75% samples that have p > 0.05. Only around one-fourth of features belong to the first category: less than 25% with p > 0.05. This result is similar to our observations in upstairs and downstairs data in BB-MAS dataset [19]. One possible explanation for this observation is similar to the case of going upstairs and downstairs that we discussed in Section 5. Participants in [116] were asked to walk at pre-set speeds on a treadmill. Thus, the step sizes between steps would have little difference within each fixed speed, resulting in all steps being in a smaller range, which increases the possibility of the data conforming to the normal distribution. Detailed results for these two datasets are shown in Appendix B, in Table 13 and Table 14 respectively.

Table 4: Analysis for underlying normal distribution in data on seven single-modality datasets. First category: less than 25% of samples with p-value >0.05; Second category: 25% to 50% of samples with p-value >0.05; Third category: 50% to 75% of samples with p-value >0.05; and, Fourth category: above 75% samples with p-value >0.05.

		Dataset	Details	Observations
Mobile and Wearable devices	Gait	[115]	Two sessions, 20 users. Walking for 15 minutes/session. Sensor: mobile phone in pocket.	All the features have 0% of samples with $p > 0.05$. Appendix B Table 13
		[116]	One session, 21 users. Walking at 12 different speeds. Walking for one minutes/speed. Sensor: markers on shoes. No. of sensors: Four sensors / shoe.	Most samples in the third category. About 25% samples in the first category. Appendix B Table 14
	Swipe	[118]	One session. More than 80 swipes/user. No. of users: 66	Half samples in the first category. The other half in the second category. Appendix C Table 15
		[117]	Two sessions. At least 80 swipes / user. No. of users: Portrait - Horiz.: 106, Vert. 118 Landscape - Horiz.: 41, Vert. 50.	Most samples in the first category with 0% of them have $p > 0.05$. Appendix C Table 16
	Typing	[125]	56 users, 51 trials/user. Fixed password (.tie5Roanl).	Unigraph: Most samples in the fourth category. Digraph: Most flight1 features in the fourth category, and all of the other features in the first category. Appendix D Table 17 Table 18
		[126]	54 users. At least 3 sessions/user. At lease 60 trails/user. Fixed password (.tie5Roanl).	Unigraph: Most samples in the fourth category. Digraph: All features in the first category. Appendix D Table 19 Table 20
Desktop	Typing	[119]	Eight sessions, 51 users. 400 password samples/user. Fixed password (.tie5Roanl).	Unigraph: All features have less than 30% with $p > 0.05$. Digraph: Nearly all features have 0% with $p > 0.05$. Appendix E Table 21 Table 22

6.2 Analysis on two swiping datasets

Most of the features for both datasets ([118] and [117]) into the first category, with less than 25% samples with $p > 0.05$. Very few, a negligible number of features in the second category of 50% samples with p > 0.05. It shows that we can reject H_0 at a significance level of 5% in most cases. This result is consistent with our observation in dataset [19]. Detailed results for these two datasets are shown in Appendix C, in Table 15 and 16, respectively.

6.3 Analysis on two typing datasets on mobile phone and one typing dataset on desktop

For both datasets, we found that most unigraph features into the fourth category, with more than 75% samples having $p > 0.05$. Moreover, most flight1 features in dataset [125] also into the fourth category. These results suggest that we cannot reject H_0 at a significance level of 5%. We posit that, this occurs due to the inherent differences in the typing surfaces. The on-screen keyboard does not require the user to press the key down to let the keyboard know that this key is pressed. One slight touch is sufficient for mobile devices to know that one key is pressed. Thus, pressing different keys on the on-screen keyboard becomes one single activity: touching or tapping. As users are familiar with touching the screen, features extracted from this action are more likely to conform to the normal distribution. However, most digraph features into the first category, with less than 25% samples with p > 0.05. It shows that we can reject H_0 at a significance level of 5%, as flight time between pressing or tapping different keys differs. This result is consistent with our observations for typing data in dataset [19]. Detailed results from our test on these datasets ([125] and [126]) are shown in Appendix D, in Table 17 and Table 18, and Table 19 and Table 20, respectively.

To verify that typing on the physical keyboard is different from typing on the on-screen keyboard, we used one additional typing dataset on desktop collected in [119] to see that whether unigraph and digraph features conform to normal distribution on desktop typing. The typing text used in this dataset is the same as that used in the above two datasets ([125] and [126]). Detailed results from our test on this dataset ([119]) are shown in Appendix E, in Table 21 and Table 22, respectively.

We found that almost all features from the dataset [119] into the first category, with less than 25% samples with p > 0.05. It shows that we can reject H_0 at a significance level of 5%. This result is consistent with our observations for typing data in our dataset [19]. In the cases of keyhold time for ".", "t", and "Right shift", we observed a slightly higher percentage of samples, but still within 30%, with $p > 0.05$. Results from this dataset suggest that typing on the physical keyboard differs significantly from typing on the on-screen keyboard. Additionally, results from these three datasets show that extraction from different devices can lead to entirely different distributions even for the same features, further illustrating that we need to be cautious about normality assumption on every feature.

7 Conclusion and Discussion on Alternate Approaches

The implications of assuming the normal distribution in data when the data is actually from a different distribution have been studied across various domains [7–11]. If methods wrongly assume a normal distribution, the findings may be misleading or wrong. In the past, several studies in behavioral biometrics have assumed a normal distribution in data [12, 13] and could improve results by either extracting features that followed normal distribution or by implementing methods more suitable for non-normal distributions [14]. Our experiments show that it would be wrong to assume an underlying normal distribution in the case of gait and swipes using phones and tablets and keystrokes using desktops. Low values of p from our non-parametric normality tests across activities and devices show that studies in mobile device-based behavioral biometrics must not assume the data to be from a Gaussian distribution to get better and more accurate insights. However, upstairs and downstairs activity data and some typing on the on-screen data showing higher percentages of samples where an underlying normal distribution cannot be discarded are intriguing. Further elaborate research is needed to establish why this occurs. Knowing that the data does not follow the normal distribution leaves the discussion incomplete, which can only be completed by learning alternate ways to handle a non-normal dataset.

Alternate approaches and solutions: One should first test for conditions of normality in data before making such an assumption. If the conditions are not met, there are numerous ways to work around the absence of normal distribution. We discuss concepts that have been successfully applied in related fields and other intuitive methods to perform more accurate analyses of mobile device-based behavioral biometric data. Several techniques are found in the literature regarding the analysis and modeling of the data itself. Different types of distributions like Weibull, Gamma, Exponential or Pareto distributions have been used, with theoretical and empirical justifications [95, 96]. However, a goodness of fit test beforehand is advised. Methods like Heteroscedastic Corrected Covariance Matrix and Bootstrapping are forms of altering the estimator to describe non-normal data better. Winsorizing and trimming data is an intermediate technique that replaces the parameters of the original data by virtue of replacing extremities in a sample. Data transformation methods like Johnson Transformation [97], Box-Cox Transformations [98] and other forms of algorithmic and parametric transformations [94] have shown promising results in other data-intensive research.

Apart from data modeling and transformation, attention to the choice of classifiers can significantly improve performance in research involving identification, verification, or classification tasks. It is common to use Gaussian Naive Bayes classifiers or Linear classifiers with Linear Discriminant Analysis, which are prevalent tools for baseline metrics. However, these classifiers assume that the data has an underlying normal distribution, and the lack of such a distribution can cause their performance to deteriorate heavily. Modified, non-parametric versions of the Gaussian Naive Bayes classifier described in [99] have shown to perform well. Standard classifiers that are not designed with the assumption of normality in data, like Support Vector Machines, K Nearest Neighbor Classifiers, or Neural Networks, are intuitively a better choice in mobile device-based behavioral biometrics, such as gait and swipes on mobile devices.

Our results question the common assumption that the data in mobile device-based behavioral biometrics, such as gait and swipes, follows a normal distribution. We have discussed the implications and alternate approaches for such scenarios. We hope that insights from our work help future studies make the right choices in data models, transformations, and classifiers to achieve better results and correct interpretations. We come full circle to the question that we began with, "Should the *assumption of normality* be the *norm* in mobile device-based behavioral biometrics?", and equipped with the knowledge from our experiments discussed above, we answer, *"no"*, with a caveat that careful examination of the validity of assumptions about underlying distributions is a must.

Acknowledgments. A bold run-in heading in small font size at the end of the paper is used for general acknowledgments, for example: This study was funded by X (grant number Y).

Disclosure of Interests. The authors have no competing interests to declare that are relevant to the content of this article.

References

1. Lilliefors, H.W.: On the Kolmogorov-Smirnov test for normality with mean and variance unknown. Journal of the American statistical Association 62(318), 399–402 (1967)
2. SHAPIRO, S.S., WILK, M.B.: An analysis of variance test for normality. Biometrika 52(3-4), 591–611 (1965). https://doi.org/10.1093/biomet/52.3-4.591
3. Massey Jr, F.J.: The Kolmogorov-Smirnov test for goodness of fit. Journal of the American statistical Association 46(253), 68–78 (1951)
4. Lan, K.-C., Heidemann, J.: Rapid model parameterization from traffic measurements. ACM Transactions on Modeling and Computer Simulation (TOMACS) 12(3), 201–229 (2002)
5. Barford, P., Crovella, M.: Generating representative web workloads for network and server performance evaluation. ACM SIGMETRICS Performance Evaluation Review 26(1), 151–160 (1998)
6. Serwadda, A., Phoha, V.V.: Examining a Large Keystroke Biometrics Dataset for Statistical-Attack Openings. ACM Transactions on Information Systems Security 16(2), 8:1–8:30 (2013). https://doi.org/10.1145/2516960
7. Limpert, E., Stahel, W.A.: Problems with using the normal distribution–and ways to improve quality and efficiency of data analysis. PloS one 6(7), e21403-e21403 (2011). https://doi.org/10.1371/journal.pone.0021403
8. Volkova, V.M.: Research of Tukey's test statistic distribution under failure of the normality assumption. In: 2016 13th International Scientific-Technical Conference on Actual Problems of Electronics Instrument Engineering (APEIE), vol. 02, pp. 227–229 (2016). https://doi.org/10.1109/APEIE.2016.7806456

9. Amhemad, A.Z.: Effect of non normality on statistical control charts. In: 2010 International Conference on Networking and Information Technology, pp. 512–515 (2010). https://doi.org/10.1109/ICNIT.2010.5508459
10. Noorossana, R., Vaghefi, A., Dorri, M.: The effect of non-normality on performance of linear profile monitoring. In: 2008 IEEE International Conference on Industrial Engineering and Engineering Management, pp. 262–266 (2008). https://doi.org/10.1109/IEEM.2008.4737871
11. Caballero-Morales, S., Rahim, A.: Analyzing the effect of non-normality on the solution space for the economic statistical design of X-bar control charts. In: 2015 International Conference on Industrial Engineering and Operations Management (IEOM), pp. 1–6 (2015). https://doi.org/10.1109/IEOM.2015.7093766
12. Song, D.X., Wagner, D., Tian, X.: Timing Analysis of Keystrokes and Timing Attacks on SSH. In: Proceedings of the 10th Conference on USENIX Security Symposium - Volume 10, p. 25. USENIX Association, Washington, D.C. (2001)
13. Stefan, D., Shu, X., Yao, D.(D.): Robustness of keystroke-dynamics based biometrics against synthetic forgeries. Computers & Security 31(1), 109–121 (2012). https://doi.org/https://doi.org/10.1016/j.cose.2011.10.001
14. Li, X., Le, J., Gopalakrishnan, P., Pileggi, L.T.: Asymptotic probability extraction for nonnormal performance distributions. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems 26(1), 16–37 (2006)
15. Stoica, P., Babu, P.: The Gaussian Data Assumption Leads to the Largest Cramér-Rao Bound [Lecture Notes]. IEEE Signal Processing Magazine 28(3), 132–133 (2011). https://doi.org/10.1109/MSP.2011.940411
16. Park, S., Serpedin, E., Qaraqe, K.: Gaussian Assumption: The Least Favorable but the Most Useful [Lecture Notes]. IEEE Signal Processing Magazine 30(3), 183–186 (2013). https://doi.org/10.1109/MSP.2013.2238691
17. Kwan, A.C.M.: Validity of normality assumption in CSP research. In: Foo, N., Goebel, R. (eds.) PRICAI'96: Topics in Artificial Intelligence, LNCS, vol. 1114, pp. 253–263. Springer, Berlin, Heidelberg (1996). https://doi.org/10.1007/3-540-61516-3_25
18. Olson Jr, C.E.: The fallacy of normality in remotely sensed data. In: Proceedings of the ASPRS Annual Conference, pp. 9–13 (2009)
19. Belman, A.K., Wang, L., Iyengar, S.S., Sniatala, P., Wright, R., Dora, R., Baldwin, J., Jin, Z., Phoha, V.V.: SU-AIS BB-MAS (Syracuse University and Assured Information Security - Behavioral Biometrics Multi-device and multi-Activity data from Same users) Dataset (2019). https://doi.org/10.21227/rpaz-0h66
20. Sim, T., Janakiraman, R.: Are Digraphs Good for Free-Text Keystroke Dynamics? In: 2007 IEEE Conference on Computer Vision and Pattern Recognition, pp. 1–6 (2007). https://doi.org/10.1109/CVPR.2007.383393
21. Nguyen, T., Voris, J.: Touchscreen Biometrics Across Multiple Devices. In: Symposium on Usable Privacy and Security,USENIX (2017)
22. Alsultan, A., Warwick, K., Wei, H.: Non-conventional keystroke dynamics for user authentication. Pattern Recognition Letters (2017)
23. Balagani, K., Phoha, V., Ray, A., Phoha, S.: On the discriminability of keystroke feature vectors used in fixed text keystroke authentication. Pattern Recognition Letters (2011)
24. Yu, E., Cho, S.: Keystroke dynamics identity verification its problems and practical solutions. Computers and security (2004)
25. Obaidat, M., Sadoun, B.: Verification of Computer Users Using Keystroke Dynamics. IEEE Transactions on Systems, Man and Cybernetics. Part B: Cybernetics. 27(2) (1997)

26. Ahmed, A.A., Traore, I.: Biometric Recognition Based on Free-Text Keystroke Dynamics. IEEE Transactions on Cybernetics 44(4) (2014)
27. Huang, J., Hou, D., Schuckers, S., Upadhyaya, S.: Effects of Text Filtering on Authentication Performance of Keystroke Biometrics. In: IEEE International Workshop on Information Forensics and Security (WIFS) (2016)
28. Mondal, S., Bours, P.: Person Identification by Keystroke Dynamics Using Pairwise User Coupling. IEEE Transactions on Information Forensics and Security 12(6) (2017)
29. Clarke, N.L., Furnell, S.M.: Authenticating mobile phone users using keystroke analysis. Int. J. Inf. Secur. (2006)
30. Bryan, L., Harter, N.: Studies in the physiology and psychology of the telegraphic language. In: Gardner, H., Gardner, J. (eds.) The Psychology of Skill: Three Studies, pp. 35–44. New York Times, New York (1973)
31. Umphress, Williams, G.: Identity verification through keyboard characteristics. Int. J. Man-Mach. Stud., pp. 263–273 (1985)
32. Gunetti, D., Picardi, C.: Keystroke Analysis of Free Text. ACM Transactions on Information Systems Security 8(3), 312–347 (2005). https://doi.org/10.1145/1085126.1085129
33. Delac, K., Grgic, M.: A survey of biometric recognition methods. In: Proceedings. Elmar-2004. 46th International Symposium on Electronics in Marine, pp. 184–193 (2004)
34. Blasco, J., Chen, T.M., Tapiador, J., Peris-Lopez, P.: A Survey of Wearable Biometric Recognition Systems. ACM Computing Surveys 49(3), 43:1–43:35 (2016). https://doi.org/10.1145/2968215
35. Teh, P.S., Teoh, A.B.J., Yue, S.: A Survey of Keystroke Dynamics Biometrics. The Scientific World Journal 2013, 408280 (2013). https://doi.org/10.1155/2013/408280
36. Pisani, P.H., Lorena, A.C.: A systematic review on keystroke dynamics. Journal of the Brazilian Computer Society 19(4), 573–587 (2013). https://doi.org/10.1007/s13173-013-0117-7
37. Janakiraman, R., Sim, T.: Keystroke Dynamics in a General Setting. In: Lee, S.-W., Li, S.Z. (eds.) Advances in Biometrics, LNCS, vol. 4641, pp. 584–593. Springer, Berlin, Heidelberg (2007). https://doi.org/10.1007/978-3-540-74549-5_60
38. Monrose, F., Rubin, A.: Authentication via Keystroke Dynamics. In: Proceedings of the 4th ACM Conference on Computer and Communications Security, pp. 48–56 (1997). https://doi.org/10.1145/266420.266434
39. Hwang, S.-s., Cho, S., Park, S.: Keystroke dynamics-based authentication for mobile devices. Computers & Security 28(1), 85–93 (2009). https://doi.org/https://doi.org/10.1016/j.cose.2008.10.002
40. Joyce, R., Gupta, G.: Identity Authentication Based on Keystroke Latencies. Communications of the ACM 33(2), 168–176 (1990). https://doi.org/10.1145/75577.75582
41. Revett, K., de Magalhães, S.T., Santos, H.M.D.: Enhancing Login Security Through the Use of Keystroke Input Dynamics. In: Zhang, D., Jain, A.K. (eds.) Advances in Biometrics, LNCS, vol. 3832, pp. 661–667. Springer, Berlin, Heidelberg (2005). https://doi.org/10.1007/978-3-540-31621-3_81
42. Ciuffo, F., Weiss, G.M.: Smartwatch-based transcription biometrics. In: 2017 IEEE 8th Annual Ubiquitous Computing, Electronics and Mobile Communication Conference (UEMCON), pp. 145–149 (2017). https://doi.org/10.1109/UEMCON.2017.8249014

43. Gunetti, D., Picardi, C., Ruffo, G.: Dealing with Different Languages and Old Profiles in Keystroke Analysis of Free Text. In: Domingo-Ferrer, J., Posegga, J., Schwenk, J. (eds.) Privacy in Statistical Databases, LNCS, vol. 3673, pp. 347–358. Springer, Berlin, Heidelberg (2005). https://doi.org/10.1007/11550624_27
44. Ali, M.L., Monaco, J.V., Tappert, C.C., Qiu, M.: Keystroke Biometric Systems for User Authentication. Journal of Signal Processing Systems 86(2), 175–190 (2017). https://doi.org/10.1007/s11265-016-1114-9
45. Azevedo, G.L.F.B., Cavalcanti, G.D.C., Carvalho Filho, E.C.B.: An approach to feature selection for keystroke dynamics systems based on PSO and feature weighting. In: Evolutionary Computation, 2007. CEC 2007. IEEE Congress on, pp. 3577–3584 (2007). https://doi.org/10.1109/CEC.2007.4425038
46. Davoudi, H., Kabir, E.: A new distance measure for free text keystroke authentication. In: 2009 14th International CSI Computer Conference, pp. 570–575 (2009). https://doi.org/10.1109/CSICC.2009.5349640
47. Goodkind, A., Brizan, D.G., Rosenberg, A.: Utilizing overt and latent linguistic structure to improve keystroke-basedauthentication. Image and Vision Computing 58, 230–238 (2017). https://doi.org/https://doi.org/10.1016/j.imavis.2016.06.003
48. Lee, H.S., Lau, T.S., Lai, W.K., King, Y.C., Lim, L.L.: User identification of numerical keypad typing patterns with subtractive clustering fuzzy inference. In: 2017 IEEE 15th Student Conference on Research and Development (SCOReD), pp. 83–88 (2017). https://doi.org/10.1109/SCORED.2017.8305418
49. Ho, J., Kang, D.-K.: Mini-batch bagging and attribute ranking for accurate user authentication in keystroke dynamics. Pattern Recognition 70, 139–151 (2017). https://doi.org/https://doi.org/10.1016/j.patcog.2017.05.002
50. Mondal, S., Bours, P.: Person Identification by Keystroke Dynamics Using Pairwise User Coupling. IEEE Transactions on Information Forensics and Security 12(6), 1319–1329 (2017). https://doi.org/10.1109/TIFS.2017.2658539
51. Sulavko, A.E., Eremenko, A.V., Fedotov, A.A.: Users' identification through keystroke dynamics based on vibration parameters and keyboard pressure. In: 2017 Dynamics of Systems, Mechanisms and Machines (Dynamics), pp. 1–7 (2017). https://doi.org/10.1109/Dynamics.2017.8239514
52. Mondal, S., Bours, P.: A study on continuous authentication using a combination of keystroke and mouse biometrics. Neurocomputing 230, 1–22 (2017). https://doi.org/https://doi.org/10.1016/j.neucom.2016.11.031
53. Alpar, O.: Frequency spectrograms for biometric keystroke authentication using neural network based classifier. Knowledge-Based Systems 116, 163–171 (2017). https://doi.org/https://doi.org/10.1016/j.knosys.2016.11.006
54. Tsimperidis, I., Arampatzis, A., Karakos, A.: Keystroke dynamics features for gender recognition. Digital Investigation 24, 4–10 (2018). https://doi.org/https://doi.org/10.1016/j.diin.2018.01.018
55. Ulinskas, M., Damaševičius, R., Maskeliūnas, R., Woźniak, M.: Recognition of human daytime fatigue using keystroke data. Procedia Computer Science 130, 947–952 (2018). https://doi.org/https://doi.org/10.1016/j.procs.2018.04.094
56. Pentel, A.: Predicting User Age by Keystroke Dynamics. In: Silhavy, R. (ed.) Artificial Intelligence and Algorithms in Intelligent Systems, pp. 336–343. Springer, Cham (2019). https://doi.org/978-3-319-91189-2
57. Monaro, M., Galante, C., Spolaor, R., Li, Q.Q., Gamberini, L., Conti, M., Sartori, G.: Covert lie detection using keyboard dynamics. Scientific Reports 8(1), 1976 (2018). https://doi.org/10.1038/s41598-018-20462-6

58. Zulueta, J., Andrea, P., Rasic, M., Easter, R., Babu, P., Langenecker, S., McInnis, M., Ajilore, O., Nelson, P., Ryan, K., et al.: 481. predicting mood disturbance severity in bipolar subjects with mobile phone keystroke dynamics and metadata. Biological Psychiatry 81(10), S196 (2017)
59. Wu, C., Ding, W., Liu, R., Wang, J., Wang, A.C., Wang, J., Li, S., Zi, Y., Wang, Z.L.: Keystroke dynamics enabled authentication and identification using triboelectric nanogenerator array. Materials Today 21(3), 216–222 (2018). https://doi.org/https://doi.org/10.1016/j.mattod.2018.01.006
60. Brizan, D.G., Goodkind, A., Koch, P., Balagani, K., Phoha, V.V., Rosenberg, A.: Utilizing linguistically enhanced keystroke dynamics to predict typist cognition and demographics. International Journal of Human-Computer Studies 82, 57–68 (2015). https://doi.org/https://doi.org/10.1016/j.ijhcs.2015.04.005
61. Liu, J., Wang, Y., Kar, G., Chen, Y., Yang, J., Gruteser, M.: Snooping Keystrokes with Mm-level Audio Ranging on a Single Phone. In: Proceedings of the 21st Annual International Conference on Mobile Computing and Networking, pp. 142–154. ACM, Paris, France (2015). https://doi.org/10.1145/2789168.2790122
62. Zhu, T., Ma, Q., Zhang, S., Liu, Y.: Context-free Attacks Using Keyboard Acoustic Emanations. In: Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security, pp. 453–464. ACM, Scottsdale, Arizona, USA (2014). https://doi.org/10.1145/2660267.2660296
63. Asonov, D., Agrawal, R.: Keyboard Acoustic Emanations. In: 2004 IEEE Symposium on Security and Privacy (S&P 2004), pp. 3–11. IEEE, Berkeley, CA, USA (2004). https://doi.org/10.1109/SECPRI.2004.1301311
64. Sun, Y., Ceker, H., Upadhyaya, S.: Anatomy of secondary features in keystroke dynamics - achieving more with less. In: 2017 IEEE International Conference on Identity, Security and Behavior Analysis (ISBA), pp. 1–6 (2017). https://doi.org/10.1109/ISBA.2017.7947691
65. Sulong, A., Wahyudi, Siddiqi, M.U.: Intelligent keystroke pressure-based typing biometrics authentication system using radial basis function network. In: 2009 5th International Colloquium on Signal Processing Its Applications, pp. 151–155 (2009). https://doi.org/10.1109/CSPA.2009.5069206
66. Serwadda, A., Phoha, V.V.: Examining a Large Keystroke Biometrics Dataset for Statistical-Attack Openings. ACM Transactions on Information Systems Security 16(2), 8:1–8:30 (2013). https://doi.org/10.1145/2516960
67. Chawla, N.V., Bowyer, K.W., Hall, L.O., Kegelmeyer, W.P.: SMOTE: Synthetic Minority Over-sampling Technique. J. Artif. Int. Res. 16(1), 321–357 (2002)
68. Sheng, Y., Phoha, V., Rovnyak, S.M.: A Parallel Decision Tree-Based Method for User Authentication Based on Keystroke Patterns. Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on 35(5), 826–833 (2005). https://doi.org/10.1109/TSMCB.2005.846648
69. Fang, S., Markwood, I., Liu, Y., Zhao, S., Lu, Z., Zhu, H.: No Training Hurdles: Fast Training-Agnostic Attacks to Infer Your Typing. In: Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security, pp. 1747–1760. ACM (2018)
70. Oxford University Press: The Oxford English Corpus: Facts about the language.OxfordDictionaries.com. (2011). Accessed August 29, 2019
71. Davies, M.: Corpus of Contemporary American English. (2010). Updated 2017, Accessed August 29, 2019
72. Sun, Y., Upadhyaya, S.: Synthetic Forgery Attack against Continuous Keystroke Authentication Systems. In: 2018 27th International Conference

on Computer Communication and Networks (ICCCN), pp. 1–7 (2018). https://doi.org/10.1109/ICCCN.2018.8487341
73. Chen, Y., Li, T., Zhang, R., Zhang, Y., Hedgpeth, T.: EyeTell: Video-Assisted Touchscreen Keystroke Inference from Eye Movements. In: 2018 IEEE Symposium on Security and Privacy (SP), pp. 144–160 (2018). https://doi.org/10.1109/SP.2018.00010
74. Monaco, J.V., Ali, M.L., Tappert, C.C.: Spoofing key-press latencies with a generative keystroke dynamics model. In: 2015 IEEE 7th International Conference on Biometrics Theory, Applications and Systems (BTAS), pp. 1–8 (2015). https://doi.org/10.1109/BTAS.2015.7358795
75. Huang, J., Hou, D., Schuckers, S.: A practical evaluation of free-text keystroke dynamics. In: 2017 IEEE International Conference on Identity, Security and Behavior Analysis (ISBA), pp. 1–8 (2017). https://doi.org/10.1109/ISBA.2017.7947695
76. Stefan, D., Yao, D.: Keystroke-dynamics authentication against synthetic forgeries. In: 6th International Conference on Collaborative Computing: Networking, Applications and Worksharing (CollaborateCom 2010), pp. 1–8 (2010). https://doi.org/10.4108/icst.collaboratecom.2010.16
77. Rahman, K.A., Balagani, K.S., Phoha, V.V.: Snoop-Forge-Replay Attacks on Continuous Verification With Keystrokes. IEEE Transactions on Information Forensics and Security 8(3), 528–541 (2013). https://doi.org/10.1109/TIFS.2013.2244091
78. Lu, L., Yu, J., Chen, Y., Zhu, Y., Xu, X., Xue, G., Li, M.: KeyLiSterber: Inferring Keystrokes on QWERTY Keyboard of Touch Screen through Acoustic Signals. In: IEEE INFOCOM 2019 - IEEE Conference on Computer Communications, pp. 775–783 (2019). https://doi.org/10.1109/INFOCOM.2019.8737591
79. Shen, S., Lin, S., Kang, T., Chien, W.: Enhanced keystroke dynamics authentication utilizing pressure detection. In: 2016 International Conference on Applied System Innovation (ICASI), pp. 1–4 (2016). https://doi.org/10.1109/ICASI.2016.7539947
80. Maiti, A., Jadliwala, M., He, J., Bilogrevic, I.: Side-Channel Inference Attacks on Mobile Keypads Using Smartwatches. IEEE Transactions on Mobile Computing 17(9), 2180–2194 (2018). https://doi.org/10.1109/TMC.2018.2794984
81. Stanciu, V.-D., Spolaor, R., Conti, M., Giuffrida, C.: On the Effectiveness of Sensor-enhanced Keystroke Dynamics Against Statistical Attacks. In: Proceedings of the Sixth ACM Conference on Data and Application Security and Privacy, pp. 105–112. ACM, New Orleans, Louisiana, USA (2016). https://doi.org/10.1145/2857705.2857748
82. Jin, K., Fang, S., Peng, C., Teng, Z., Mao, X., Zhang, L., Li, X.: ViViSnoop: Someone is snooping your typing without seeing it! In: 2017 IEEE Conference on Communications and Network Security (CNS), pp. 1–9 (2017). https://doi.org/10.1109/CNS.2017.8228624
83. Khan, H., Hengartner, U., Vogel, D.: Augmented Reality-based Mimicry Attacks on Behaviour-Based Smartphone Authentication. In: Proceedings of the 16th Annual International Conference on Mobile Systems, Applications, and Services, pp. 41–53. ACM, Munich, Germany (2018). https://doi.org/10.1145/3210240.3210317
84. Ali, K., Liu, A.X., Wang, W., Shahzad, M.: Keystroke Recognition Using WiFi Signals. In: Proceedings of the 21st Annual International Conference on Mobile Computing and Networking, pp. 90–102. ACM, Paris, France (2015). https://doi.org/10.1145/2789168.2790109
85. Shukla, D., Phoha, V.V.: Stealing Passwords by Observing Hands Movement. IEEE Transactions on Information Forensics and Security 14(12), 3086–3101 (2019). https://doi.org/10.1109/TIFS.2019.2911171

86. Dunning, T.: Accurate Methods for the Statistics of Surprise and Coincidence. Comput. Linguist. 19(1), 61–74 (1993)
87. Caballero-Morales, S., Rahim, A.: Analyzing the effect of non-normality on the solution space for the economic statistical design of X-bar control charts. In: 2015 International Conference on Industrial Engineering and Operations Management (IEOM), pp. 1–6 (2015). https://doi.org/10.1109/IEOM.2015.7093766
88. Marko, N.F., Weil, R.J.: Non-Gaussian Distributions Affect Identification of Expression Patterns, Functional Annotation, and Prospective Classification in Human Cancer Genomes. PLOS ONE 7(10), 1–15 (2012). https://doi.org/10.1371/journal.pone.0046935
89. Kwak, S.G., Kim, J.H.: Central limit theorem: the cornerstone of modern statistics. Korean journal of anesthesiology 70(2), 144–156 (2017). https://doi.org/10.4097/kjae.2017.70.2.144
90. Cramér, H.: Mathematical methods of statistics. Princeton University Press, Princeton (1999)
91. Stephens, M.A.: EDF statistics for goodness-of-fit: Part 1. STANFORD UNIV CA DEPT OF STATISTICS (1972)
92. Belman, A.K., Wang, L., Iyengar, S.S., Sniatala, P., Wright, R., Dora, R., Baldwin, J., Jin, Z., Phoha, V.V.: Insights from BB-MAS – A Large Dataset for Typing, Gait and Swipes of the Same Person on Desktop, Tablet and Phone (2019). https://doi.org/ArXiv:1912.02736
93. Limpert, E., Stahel, W.A.: Problems with Using the Normal Distribution – and Ways to Improve Quality and Efficiency of Data Analysis. PLOS ONE 6(7), 1–8 (2011). https://doi.org/10.1371/journal.pone.0021403
94. Pek, J., Wong, O., Wong, A.C.M.: How to Address Non-normality: A Taxonomy of Approaches, Reviewed, and Illustrated. Frontiers in Psychology 9, 2104 (2018). https://doi.org/10.3389/fpsyg.2018.02104
95. Adewale, S.: A STATISTICAL ANALYSIS TO DETERMINE THE DISTRIBUTION AND PATTERN FOR AN INSURANCE HEALTH CLAIMS (2017). https://doi.org/10.13140/RG.2.2.20068.42888
96. Liu, J., Wu, Z., Wu, J., Dong, J., Zhao, Y., Wen, D.: A Weibull distribution accrual failure detector for cloud computing. PLOS ONE 12(3), 1–16 (2017). https://doi.org/10.1371/journal.pone.0173666
97. Yang, Y., Li, D., Qi, Y.: An Approach to Non-normal Process Capability Analysis Using Johnson Transformation. In: 2018 IEEE 4th International Conference on Control Science and Systems Engineering (ICCSSE), pp. 495–498 (2018). https://doi.org/10.1109/CCSSE.2018.8724679
98. Yang, Y., Zhu, H.: A Study of Non-Normal Process Capability Analysis Based on Box-Cox Transformation. In: 2018 3rd International Conference on Computational Intelligence and Applications (ICCIA), pp. 240–243 (2018). https://doi.org/10.1109/ICCIA.2018.00057
99. Soria, D., Garibaldi, J.M., Ambrogi, F., Biganzoli, E.M., Ellis, I.O.: A "nonparametric" version of the naive Bayes classifier. Knowledge-Based Systems 24(6), 775–784 (2011). https://doi.org/doi:10.1016/j.knosys.2011.02.014
100. Arora, P., Hanmandlu, M., Srivastava, S.: Gait based authentication using gait information image features. Pattern Recognition Letters 68, 336–342 (2015)
101. Cola, G., Avvenuti, M., Musso, F., Vecchio, A.: Gait-based authentication using a wrist-worn device. In: Proceedings of the 13th International Conference on Mobile and Ubiquitous Systems: Computing, Networking and Services, pp. 208–217 (2016)

102. Belman, A.K., Phoha, V.V.: Discriminative Power of Typing Features on Desktops, Tablets, and Phones for User Identification. ACM Trans. Priv. Secur. 23(1), Article 4 (2020). https://doi.org/10.1145/3377404
103. Kim, J., Kim, H., Kang, P.: Keystroke dynamics-based user authentication using freely typed text based on user-adaptive feature extraction and novelty detection. Applied Soft Computing 62, 1077–1087 (2018). https://doi.org/https://doi.org/10.1016/j.asoc.2017.09.045
104. Bajrami, G.: Activity identification for gait recognition using mobile devices (2011)
105. Derawi, M., Bours, P.: Gait and activity recognition using commercial phones. computers & security 39, 137–144 (2013)
106. Monrose, F., Rubin, A.D.: Keystroke dynamics as a biometric for authentication. Future Generation computer systems 16(4), 351–359 (2000)
107. Brizan, D.G., Goodkind, A., Koch, P., Balagani, K., Phoha, V.V., Rosenberg, A.: Utilizing linguistically enhanced keystroke dynamics to predict typist cognition and demographics. International Journal of Human-Computer Studies 82, 57–68 (2015). https://doi.org/https://doi.org/10.1016/j.ijhcs.2015.04.005
108. Montgomery, D., Runger, G.: Applied Statistics and Probability for Engineers, 6th Edition. John Wiley & Sons (2013)
109. Lehmann, E.L., Casella, G.: Theory of point estimation. Springer, Berkeley, CA (1998)
110. Çeker, H., Upadhyaya, S.: Enhanced recognition of keystroke dynamics using Gaussian mixture models. In: MILCOM 2015 - 2015 IEEE Military Communications Conference, pp. 1305–1310 (2015)
111. Bhatia, A., Hanmandlu, M., Vasikarla, S., Panigrahi, B.K.: Keystroke Dynamics Based Authentication Using GFM. In: 2018 IEEE International Symposium on Technologies for Homeland Security (HST), pp. 1–5 (2018)
112. Hosseinzadeh, D., Krishnan, S., Khademi, A.: Keystroke Identification Based on Gaussian Mixture Models. In: 2006 IEEE International Conference on Acoustics Speech and Signal Processing Proceedings, vol. 3, pp. III–III (2006)
113. Fierrez, J., Pozo, A., Martinez-Diaz, M., Galbally, J., Morales, A.: Benchmarking Touchscreen Biometrics for Mobile Authentication. IEEE Transactions on Information Forensics and Security 13(11), 2720–2733 (2018)
114. Pozo, A., Fierrez, J., Martinez-Diaz, M., Galbally, J., Morales, A.: Exploring a statistical method for touchscreen swipe biometrics. In: 2017 International Carnahan Conference on Security Technology (ICCST), pp. 1–4 (2017)
115. Frank, J., Mannor, S., Precup, D.: Data Sets: Mobile Phone Gait Recognition Data (2010). http://www.cs.mcgill.ca/ jfrank8/data/gait-dataset.html
116. Hebenstreit, F., Leibold, A., Krinner, S., Welsch, G., Lochmann, M., Eskofier, B.M.: Are Relative Gait Phase Durations Speed Dependent? (2014). https://www.mad.tf.fau.de/research/activitynet/gaitphase-database/
117. Serwadda, A., Phoha, V.V., Wang, Z.: Which verifiers work?: A benchmark evaluation of touch-based authentication algorithms. In: 2013 IEEE Sixth International Conference on Biometrics: Theory, Applications and Systems (BTAS), pp. 1–8 (2013). https://doi.org/10.1109/BTAS.2013.6712758
118. Serwadda, A., Phoha, V.V.: When Kids' Toys Breach Mobile Phone Security. In: Proceedings of the 2013 ACM SIGSAC Conference on Computer & Communications Security, pp. 599–610. ACM, Berlin, Germany (2013). https://doi.org/10.1145/2508859.2516659

119. Killourhy, K.S., Maxion, R.A.: Comparing anomaly-detection algorithms for keystroke dynamics. In: 2009 IEEE/IFIP International Conference on Dependable Systems & Networks, pp. 125–134 (2009)
120. Killourhy, K.S., Maxion, R.A.: Free vs. transcribed text for keystroke-dynamics evaluations. In: Proceedings of the 2012 Workshop on Learning from Authoritative Security Experiment Results, pp. 1–8 (2012)
121. Watanabe, Y., Sara, S.: Toward an Immunity-based Gait Recognition on Smart Phone: A Study of Feature Selection and Walking State Classification. Procedia Computer Science 96, 1790–1800 (2016). https://doi.org/https://doi.org/10.1016/j.procs.2016.08.228
122. Watanabe, Y.: Houryu. A Study of Accelerometer-based Authentication on Walking with Smart Phone. In: International Workshop on Smart Info-Media Systems in Asia (SISA), pp. 372–375 (2013)
123. Frank, M., Biedert, R., Ma, E., Martinovic, I., Song, D.: Touchalytics: On the Applicability of Touchscreen Input as a Behavioral Biometric for Continuous Authentication. IEEE Transactions on Information Forensics and Security 8(1), 136–148 (2013). https://doi.org/10.1109/TIFS.2012.2225048
124. Bo, C., Zhang, L., Jung, T., Han, J., Li, X.-Y., Wang, Y.: Continuous user identification via touch and movement behavioral biometrics. In: 2014 IEEE 33rd International Performance Computing and Communications Conference (IPCCC), pp. 1–8 (2014). https://doi.org/10.1109/PCCC.2014.7017067
125. Al-Obaidi, N.: MEU-Mobile KSD Data Set (2017). Middle East University, Faculty of Information Technology, Amman, Jordan. https://archive.ics.uci.edu/ml/datasets/MEU-Mobile+KSD
126. Antal, M., Nemes, L.: The mobikey keystroke dynamics password database: Benchmark results. In: Computer Science On-line Conference, pp. 35–46. Springer (2016)
127. Killourhy, K.S., Maxion, R.A.: Comparing anomaly-detection algorithms for keystroke dynamics. In: 2009 IEEE/IFIP International Conference on Dependable Systems & Networks, pp. 125–134 (2009)

For visual clarity, cells in each table in all following Appendixes are colored in four different colors based on four categories we defined in Section 4. These four categories and their corresponding colors are: First category: less than 25% of samples with p-value >0.05; Second category: 25% to 50% of samples with p-value >0.05; Third category: 50% to 75% of samples with p-value >0.05; and, Fourth category: above 75% samples with p-value >0.05.

A Detailed results from our experiments on each feature, activity and device in SU-AIS BB-MAS [19] dataset.

We use symbols $d1$ through $d18$ to denote the digraphs ($d1$: ('BACKSPACE', 'BACKSPACE'), $d2$: ('SPACE', 'a'), $d3$: ('SPACE', 'i'), $d4$: ('SPACE', 's'), $d5$: ('SPACE', 't'), $d6$: ('e', 'SPACE'), $d7$: ('e', 'n'), $d8$: ('e', 'r'), $d9$: ('e', 's'), $d10$: ('n', 'SPACE'), $d11$: ('o', 'SPACE'), $d12$: ('o', 'n'), $d13$: ('r', 'e'), $d14$: ('s', 'SPACE'), $d15$: ('s', 'e'), $d16$: ('t', 'SPACE'), $d17$: ('t', 'e') and $d18$: ('t', 'h'))

Table 5: Percentage of test samples with $p > 0.05$ for keyhold feature from unigraphs on desktop, tablet and phone.

Uni.	Desktop	Tablet	Phone
SPACE	0%	1%	11%
Bspace	0%	0%	2%
a	0%	15%	20%
e	0%	0%	5%
h	0%	9%	18%
i	0%	2%	9%
l	0%	9%	15%
n	0%	2%	27%
o	0%	5%	22%
r	0%	7%	22%
s	0%	4%	16%
t	0%	1%	9%

Table 6: Percentage of test samples with $p > 0.05$ for F1-F4: flight1-flight4 features from digraphs on desktop.

	Desktop			
Di.	F1	F2	F3	F4
$d1$	0%	0%	0%	0%
$d2$	0%	0%	0%	0%
$d3$	1%	2%	1%	1%
$d4$	0%	0%	0%	0%
$d5$	0%	0%	0%	0%
$d6$	0%	0%	0%	0%
$d7$	2%	3%	3%	2%
$d8$	0%	2%	2%	3%
$d9$	0%	1%	3%	4%
$d10$	0%	0%	1%	0%
$d11$	1%	1%	1%	1%
$d12$	3%	2%	2%	4%
$d13$	0%	2%	1%	3%
$d14$	0%	0%	0%	1%
$d15$	0%	1%	0%	0%
$d16$	0%	0%	0%	0%
$d17$	2%	5%	4%	6%
$d18$	0%	2%	0%	0%

Table 7: Percentage of test samples with $p > 0.05$ for F1-F4: flight1-flight4 features from digraphs on tablet.

Di.	Tablet F1	F2	F3	F4
$d1$	0%	0%	0%	0%
$d2$	0%	0%	0%	0%
$d3$	2%	1%	1%	1%
$d4$	1%	1%	2%	1%
$d5$	0%	0%	0%	0%
$d6$	1%	1%	1%	1%
$d7$	5%	7%	6%	9%
$d8$	9%	7%	13%	11%
$d9$	5%	3%	4%	4%
$d10$	3%	1%	2%	3%
$d11$	0%	1%	0%	1%
$d12$	8%	6%	3%	9%
$d13$	5%	4%	5%	5%
$d14$	0%	0%	0%	1%
$d15$	4%	3%	3%	4%
$d16$	0%	0%	0%	0%
$d17$	18%	17%	16%	19%
$d18$	8%	6%	3%	3%

Table 8: Percentage of test samples with $p > 0.05$ for F1-F4: flight1-flight4 features from digraphs on phone.

Di.	Phone F1	F2	F3	F4
$d1$	0%	0%	0%	0%
$d2$	1%	0%	2%	0%
$d3$	0%	0%	1%	1%
$d4$	1%	1%	0%	0%
$d5$	0%	0%	0%	0%
$d6$	0%	1%	1%	1%
$d7$	6%	5%	8%	8%
$d8$	15%	17%	22%	23%
$d9$	5%	8%	6%	11%
$d10$	0%	2%	1%	1%
$d11$	1%	3%	2%	4%
$d12$	14%	16%	15%	17%
$d13$	7%	5%	9%	12%
$d14$	0%	3%	1%	3%
$d15$	0%	6%	2%	8%
$d16$	1%	1%	1%	1%
$d17$	18%	20%	19%	17%
$d18$	6%	4%	6%	6%

Table 9: Percentage of test samples with $p > 0.05$ for features from Swiping activity on phone and tablet.

Features	Phone	Tablet	Features	Phone	Tablet
minx	2%	3%	aquarts_0	1%	7%
miny	21%	32%	aquarts_1	0%	0%
maxx	5%	2%	aquarts_2	0%	0%
maxy	6%	16%	pmean	72%	26%
euclidist	40%	26%	pstd	76%	33%
tanangle	0%	1%	pquarts_0	65%	29%
tottime	0%	0%	pquarts_1	65%	26%
vmean	2%	9%	pquarts_2	66%	21%
vstd	6%	14%	areamean	56%	33%
vquarts_0	0%	1%	areastd	46%	37%
vquarts_1	0%	5%	areaquarts_0	9%	1%
vquarts_2	2%	14%	areaquarts_1	4%	0%
amean	0%	0%	areaquarts_2	4%	0%
astd	0%	0%			

Table 10: Percentage of test samples with $p > 0.05$ for features from Walking activity phone in hand, phone in pocket and tablet in hand. (Acc: Accelerometer, Gyr: Gyroscope, DT: Desktop, HP: HandPhone, PP: PocketPhone, HT: HandTablet)

Feature	HP		PP		HT		Feature	HP		PP		HT	
	Acc	Gyr	Acc	Gyr	Acc	Gyr		Acc	Gyr	Acc	Gyr	Acc	Gyr
xmean	36%	25%	21%	6%	32%	28%	ziqr	50%	24%	30%	17%	50%	27%
ymean	39%	9%	28%	6%	38%	11%	miqr	53%	11%	12%	15%	54%	4%
zmean	32%	4%	16%	12%	25%	2%	xrange	29%	36%	9%	3%	36%	38%
mmean	33%	18%	28%	8%	34%	10%	yrange	26%	17%	10%	9%	26%	28%
xstd	35%	41%	9%	3%	37%	44%	zrange	30%	23%	9%	11%	27%	25%
ystd	21%	15%	9%	13%	31%	32%	mrange	32%	20%	7%	4%	23%	13%
zstd	37%	25%	8%	7%	38%	19%	xsnr	17%	10%	5%	1%	15%	12%
mstd	38%	16%	5%	9%	40%	4%	ysnr	12%	13%	3%	3%	11%	3%
xbp	3%	9%	35%	11%	4%	12%	zsnr	7%	8%	3%	4%	5%	12%
ybp	15%	2%	31%	15%	11%	2%	msnr	7%	31%	1%	9%	3%	32%
zbp	37%	3%	27%	26%	25%	0%	xydtw	17%	24%	3%	15%	17%	28%
mbp	31%	5%	24%	13%	36%	3%	yzdtw	18%	16%	3%	8%	4%	15%
xenergy	7%	18%	15%	3%	11%	27%	xzdtw	3%	5%	9%	13%	2%	2%
yenergy	26%	3%	3%	12%	23%	4%	xymi	0%	12%	0%	1%	9%	23%
zenergy	0%	2%	15%	21%	0%	0%	xzmi	0%	5%	2%	1%	3%	17%
menergy	0%	8%	0%	2%	0%	3%	xmmi	0%	0%	2%	2%	0%	2%
xmfreq	3%	0%	14%	8%	0%	0%	yzmi	0%	9%	2%	1%	3%	11%
ymfreq	1%	0%	17%	6%	0%	0%	ymmi	0%	0%	0%	2%	1%	1%
zmfreq	6%	1%	11%	3%	0%	0%	zmmi	0%	0%	2%	1%	0%	2%
mmfreq	65%	58%	62%	68%	98%	74%	xycorr	55%	50%	19%	8%	65%	60%
xiqr	26%	38%	26%	10%	35%	52%	yzcorr	45%	56%	12%	11%	59%	53%
yiqr	29%	16%	14%	22%	38%	37%	xzcorr	46%	16%	12%	9%	57%	42%

Table 11: Percentage of test samples with $p > 0.05$ for features from Upstairs activity with phone in hand, phone in pocket and tablet in hand. (Acc: Accelerometer, Gyr: Gyroscope, DT: Desktop, HP: HandPhone, PP: PocketPhone, HT: HandTablet)

Feature	HP		PP		HT		Feature	HP		PP		HT	
	Acc	Gyr	Acc	Gyr	Acc	Gyr		Acc	Gyr	Acc	Gyr	Acc	Gyr
xmean	79%	76%	62%	67%	74%	78%	ziqr	79%	83%	76%	69%	85%	83%
ymean	83%	60%	77%	20%	79%	56%	miqr	87%	67%	70%	70%	85%	52%
zmean	89%	16%	56%	51%	80%	11%	xrange	70%	68%	50%	58%	65%	74%
mmean	89%	63%	75%	60%	91%	44%	yrange	68%	70%	51%	62%	69%	75%
xstd	68%	79%	67%	60%	72%	83%	zrange	73%	80%	60%	56%	72%	73%
ystd	75%	76%	69%	64%	72%	79%	mrange	72%	60%	53%	53%	70%	71%
zstd	79%	85%	66%	62%	88%	85%	xsnr	78%	74%	61%	68%	74%	80%
mstd	85%	63%	65%	57%	91%	61%	ysnr	81%	62%	44%	27%	85%	59%
xbp	67%	72%	67%	62%	61%	68%	zsnr	72%	19%	63%	66%	63%	12%
ybp	74%	49%	77%	45%	64%	59%	msnr	71%	78%	53%	74%	59%	85%
zbp	85%	21%	69%	56%	80%	19%	xydtw	69%	79%	74%	73%	62%	77%
mbp	85%	44%	74%	56%	86%	28%	yzdtw	86%	46%	71%	55%	78%	38%
xenergy	68%	71%	68%	62%	62%	67%	xzdtw	81%	30%	74%	69%	76%	20%
yenergy	74%	49%	76%	45%	60%	59%	xymi	86%	88%	74%	78%	90%	86%
zenergy	85%	21%	69%	56%	76%	19%	xzmi	89%	85%	79%	77%	87%	85%
menergy	85%	46%	76%	56%	71%	28%	xmmi	87%	80%	79%	74%	87%	90%
xmfreq	31%	11%	61%	44%	4%	19%	yzmi	87%	90%	79%	74%	88%	85%
ymfreq	25%	16%	63%	43%	16%	11%	ymmi	89%	84%	67%	75%	91%	89%
zmfreq	36%	16%	52%	26%	3%	5%	zmmi	88%	83%	76%	79%	81%	82%
mmfreq	86%	81%	76%	79%	96%	91%	xycorr	88%	92%	67%	61%	86%	91%
xiqr	74%	80%	74%	60%	74%	86%	yzcorr	85%	91%	63%	56%	87%	85%
yiqr	71%	79%	74%	74%	70%	79%	xzcorr	91%	85%	69%	46%	88%	91%

Table 12: Percentage of test samples with $p > 0.05$ for features from Downstairs activity phone in hand, phone in pocket and tablet in hand. (Acc: Accelerometer, Gyr: Gyroscope, DT: Desktop, HP: HandPhone, PP: PocketPhone, HT: HandTablet)

Feature	HP		PP		HT		Feature	HP		PP		HT	
	Acc	Gyr	Acc	Gyr	Acc	Gyr		Acc	Gyr	Acc	Gyr	Acc	Gyr
xmean	84%	85%	74%	72%	74%	85%	ziqr	85%	85%	85%	84%	89%	78%
ymean	85%	61%	81%	14%	83%	61%	miqr	90%	74%	82%	78%	88%	59%
zmean	88%	24%	57%	76%	84%	21%	xrange	79%	74%	67%	64%	70%	74%
mmean	90%	78%	90%	72%	91%	56%	yrange	77%	65%	64%	71%	70%	79%
xstd	81%	87%	79%	80%	74%	87%	zrange	72%	82%	69%	70%	76%	79%
ystd	86%	74%	83%	75%	78%	87%	mrange	77%	72%	71%	56%	74%	76%
zstd	89%	85%	82%	76%	89%	83%	xsnr	90%	76%	72%	72%	79%	72%
mstd	91%	83%	80%	74%	87%	68%	ysnr	78%	57%	50%	19%	81%	61%
xbp	77%	72%	80%	77%	60%	79%	zsnr	56%	29%	74%	77%	56%	32%
ybp	82%	59%	85%	50%	67%	70%	msnr	53%	84%	54%	74%	55%	79%
zbp	86%	32%	85%	76%	81%	14%	xydtw	81%	76%	85%	73%	71%	87%
mbp	91%	62%	90%	67%	89%	48%	yzdtw	84%	73%	79%	67%	84%	44%
xenergy	77%	71%	80%	75%	60%	77%	xzdtw	85%	60%	82%	78%	78%	27%
yenergy	81%	61%	87%	48%	68%	70%	xymi	85%	84%	79%	78%	85%	85%
zenergy	87%	31%	85%	76%	81%	15%	xzmi	89%	85%	80%	76%	85%	80%
menergy	89%	62%	88%	68%	88%	44%	xmmi	90%	83%	80%	74%	91%	75%
xmfreq	24%	20%	68%	50%	6%	8%	yzmi	83%	85%	80%	75%	78%	85%
ymfreq	27%	21%	68%	40%	16%	19%	ymmi	87%	89%	75%	80%	88%	88%
zmfreq	48%	15%	72%	33%	1%	12%	zmmi	89%	84%	79%	78%	71%	84%
mmfreq	91%	91%	83%	88%	98%	92%	xycorr	86%	93%	78%	79%	88%	91%
xiqr	86%	91%	83%	82%	70%	90%	yzcorr	87%	93%	77%	79%	82%	89%
yiqr	82%	85%	84%	79%	73%	89%	xzcorr	91%	85%	84%	76%	89%	92%

B Detailed results from our experiments on two gait datasets ([115] and [116]).

Table 13: Percentage of test samples with $p > 0.05$ for features from Gait Dataset [115].

Features	Acc	Mag	Features	Acc	Mag
xmean	0%	0%	ziqr	0%	0%
ymean	0%	0%	miqr	0%	0%
zmean	0%	0%	xrange	0%	0%
mmean	0%	0%	yrange	0%	0%
xstd	0%	0%	zrange	0%	0%
ystd	0%	0%	mrange	0%	0%
zstd	0%	0%	xsnr	0%	0%
mstd	0%	0%	ysnr	0%	0%
xbp	0%	0%	zsnr	0%	0%
ybp	0%	0%	msnr	0%	0%
zbp	0%	0%	xydtw	0%	1%
mbp	0%	0%	yzdtw	0%	0%
xenergy	0%	0%	xzdtw	0%	0%
yenergy	0%	0%	xymi	0%	0%
zenergy	0%	0%	xzmi	0%	0%
menergy	0%	0%	xmmi	0%	0%
xmfreq	0%	0%	yzmi	0%	0%
ymfreq	0%	0%	ymmi	0%	0%
zmfreq	0%	0%	zmmi	0%	0%
mmfreq	5%	0%	xycorr	0%	0%
xiqr	0%	0%	yzcorr	0%	0%
yiqr	0%	0%	xzcorr	0%	0%

Table 14: Percentage of test samples with $p > 0.05$ for features from GaitPhase Dataset[116].

Features	L_FCC	L_FM1	L_FM2	L_FM5	R_FCC	R_FM1	R_FM2	R_FM5
xmean	73%	73%	72%	73%	68%	72%	71%	73%
ymean	78%	77%	77%	77%	78%	78%	78%	78%
zmean	14%	68%	59%	52%	16%	66%	57%	60%
mmean	72%	74%	72%	74%	70%	72%	72%	70%
xstd	80%	61%	63%	63%	85%	64%	67%	67%
ystd	51%	46%	44%	40%	42%	45%	46%	39%
zstd	11%	78%	72%	59%	13%	75%	70%	59%
mstd	87%	76%	74%	75%	88%	76%	75%	76%
xbp	69%	71%	72%	74%	69%	71%	70%	72%
ybp	77%	76%	75%	76%	78%	79%	79%	78%
zbp	12%	70%	62%	51%	13%	65%	57%	60%
mbp	71%	72%	72%	72%	69%	72%	70%	72%
xenergy	71%	73%	72%	71%	71%	73%	72%	72%
yenergy	78%	76%	76%	77%	78%	80%	80%	79%
zenergy	11%	71%	64%	51%	13%	65%	57%	60%
menergy	71%	71%	72%	73%	68%	71%	71%	72%
xmfrep	78%	75%	74%	74%	68%	71%	70%	72%
ymfrep	76%	73%	73%	76%	76%	74%	74%	71%
zmfrep	31%	19%	13%	15%	31%	26%	16%	16%
mmfrep	74%	73%	74%	72%	70%	71%	72%	68%
xiqr	63%	20%	21%	20%	65%	21%	22%	22%
yiqr	20%	21%	21%	17%	19%	21%	21%	14%
ziqr	6%	51%	51%	36%	6%	59%	55%	37%
miqr	80%	55%	57%	60%	82%	48%	46%	46%
xrange	54%	54%	54%	57%	62%	57%	59%	60%
yrange	60%	60%	61%	61%	62%	60%	59%	62%
zrange	68%	74%	76%	76%	66%	76%	73%	72%
mrange	68%	67%	68%	66%	75%	72%	73%	73%
xsnr	0%	0%	0%	0%	1%	0%	0%	0%
ysnr	42%	38%	35%	26%	43%	29%	31%	26%
zsnr	7%	1%	2%	1%	6%	1%	3%	1%
msnr	3%	1%	1%	0%	3%	0%	0%	1%
xydtw	66%	73%	69%	69%	68%	71%	73%	73%
yzdtw	46%	64%	59%	66%	51%	64%	49%	62%
xzdtw	83%	79%	80%	81%	84%	82%	83%	81%
xymi	68%	71%	70%	72%	71%	72%	70%	71%
xzmi	74%	75%	76%	75%	72%	73%	73%	74%
xmmi	3%	14%	3%	29%	4%	10%	2%	25%
yzmi	33%	47%	44%	47%	37%	46%	39%	50%
ymmi	64%	88%	87%	83%	61%	87%	83%	85%
zmmi	82%	73%	77%	67%	84%	72%	74%	68%
xycorr	71%	67%	70%	70%	64%	58%	57%	53%
yzcorr	2%	13%	2%	28%	4%	9%	1%	24%
xzcorr	74%	71%	72%	73%	73%	71%	71%	73%

C Detailed results from our experiments on two swipe datasets ([118] and [117]).

Table 15: Percentage of test samples with $p > 0.05$ for features from Swiping activity on phone for dataset [118]

Features	Phone	Features	Phone
minx	27%	aquarts_0	18%
miny	48%	aquarts_1	22%
maxx	26%	aquarts_2	22%
maxy	41%	pmean	61%
eucliddist	46%	pstd	47%
tanangle	14%	pquarts_0	41%
tottime	12%	pquarts_1	35%
vmean	24%	pquarts_2	36%
vstd	26%	areamean	32%
vquarts_0	18%	areastd	26%
vquarts_1	22%	areaquarts_0	25%
vquarts_2	27%	areaquarts_1	20%
amean	24%	areaquarts_2	18%

Table 16: Percentage of test samples with $p > 0.05$ for features from Swiping activity on phone in portrait and landscape for dataset [117]

Features	Portrait	Landscape	Features	Portrait	Landscape
minx	0%	1%	aquarts_0	0%	0%
miny	4%	2%	aquarts_1	0%	0%
maxx	8%	2%	aquarts_2	1%	1%
maxy	17%	3%	pmean	48%	58%
eucliddist	47%	51%	pstd	18%	13%
tanangle	0%	0%	pquarts_0	36%	73%
tottime	0%	0%	pquarts_1	41%	24%
vmean	0%	0%	pquarts_2	52%	59%
vstd	1%	0%	areamean	31%	46%
vquarts_0	0%	0%	areastd	6%	13%
vquarts_1	0%	0%	areaquarts_0	0%	0%
vquarts_2	1%	0%	areaquarts_1	0%	0%
amean	0%	0%	areaquarts_2	0%	0%

D Detailed results from our experiments on two mobile phone keystroke datasets ([125] and [126]).

Table 17: Percentage of test samples with $p > 0.05$ for keyhold from unigraphs in desktop dataset [125].

Uni.	Phone
PERIOD	76%
t	84%
i	78%
e	74%
SHIFT	78%
5	82%
CAPS	77%
r	80%
o	78%
a	77%
n	80%
l	76%
ENTER	71%

Table 18: Percentage of test samples with $p > 0.05$ for F1-F4: flight1-flight4 from digraphs in desktop dataset [125].

	Phone			
Di.	F1	F2	F3	F4
(PERIOD, t)	76%	5%	5%	6%
(t, i)	82%	15%	13%	17%
(I, e)	79%	13%	13%	10%
(e, SHIFT)	74%	3%	3%	3%
(SHIFT, 5)	78%	7%	6%	6%
(5, SHIFT)	83%	4%	3%	4%
(SHIFT, CAPS)	79%	1%	2%	1%
(CAPS, r)	78%	13%	12%	15%
(r, o)	80%	7%	7%	6%
(o, a)	78%	12%	10%	10%
(a, n)	77%	15%	12%	17%
(n, l)	81%	19%	19%	23%
(l, ENTER)	74%	5%	6%	6%

Table 19: Percentage of test samples with $p > 0.05$ for keyhold from unigraphs in desktop dataset [126].

Uni.	Phone
PERIOD	79%
t	81%
i	82%
e	75%
123?	77%
5	73%
abc	81%
SHIFT	75%
R	75%
o	82%
a	77%
n	80%
l	85%

Table 20: Percentage of test samples with $p > 0.05$ for F1-F4: flight1-flight4 from digraphs in desktop dataset [126].

	Desktop			
Di.	F1	F2	F3	F4
(PERIOD, t)	1%	1%	1%	3%
(t, i)	9%	11%	12%	13%
(I, e)	6%	7%	6%	7%
(e, 123?)	1%	1%	1%	1%
(123?, 5)	3%	3%	3%	4%
(5, abc)	1%	1%	2%	2%
(abc, SHIFT)	4%	4%	6%	7%
(SHIFT, R)	19%	17%	18%	19%
(R, o)	2%	2%	4%	3%
(o, a)	7%	9%	7%	8%
(a, n)	5%	7%	10%	11%
(n, l)	8%	10%	10%	13%

E Detailed results from our experiments on a desktop keystroke dataset ([119]).

Table 21: Percentage of test samples with $p > 0.05$ for keyhold feature from unigraphs on desktop from dataset [119].

Uni.	Desktop
PERIOD	29%
t	25%
i	15%
e	14%
5	14%
RIGHT SHIFT	21%
o	18%
a	18%
n	15%
l	10%
RETURN	13%

Table 22: Percentage of test samples with $p > 0.05$ for F1-F4: flight1-flight4 features from digraphs on desktop from dataset [119].

	Desktop			
Di.	F1	F2	F3	F4
(PERIOD, t)	0%	0%	0%	0%
(t, i)	0%	1%	0%	1%
(I, e)	0%	0%	0%	0%
(e, 5)	0%	0%	0%	0%
(5, RIGHT SHIFT)	0%	0%	0%	0%
(RIGHT SHIFT, o)	0%	0%	0%	0%
(o, a)	0%	0%	0%	0%
(a, n)	0%	0%	0%	0%
(n, l)	0%	0%	0%	0%
(l, RETURN)	0%	0%	0%	0%

Behavioral Biometrics as a Pillar for Multimedia Forensics in the Age of AI

Pronab Mohanty

Director General of Police, Indian Police Service, Government of India.

Contributing authors: pronabmohanty@gmail.com;

Abstract

In an era defined by ubiquitous digital interactions and the growing menace of synthetic media, traditional forensics is rapidly evolving into a dynamic, data-driven discipline. One of the most promising developments in this transformation is the integration of behavioral biometrics into multimedia forensics. Behavioral biometrics—patterns in how individuals interact with devices—are inherently difficult to mimic or forge and offer a new dimension of identity validation in cyber investigations.

The SU-AIS BB-MAS Dataset, a large-scale collection of behavioral data compiled by Syracuse University and Assured Information Security, exemplifies this new frontier. With over 62 million data points including keystrokes, swipes, and motion data, it serves as a critical resource for forensic researchers, security analysts, and law enforcement agencies. Recognized by IEEE Dataport as one of the three most popular datasets on their portal, SU-AIS BB-MAS has already gained global attention for its scale, quality, and potential impact on AI-based forensic research.

This article explores the emerging role of AI-powered behavioral biometrics in digital forensics and their potential to shape the future of cybercrime investigation and justice delivery.

Keywords: Behavioral Biometrics, Multimedia Forensics, Artificial Intelligence, SU-AIS BB-MAS Dataset

1 Introduction

As digital technology continues to permeate nearly every dimension of human life, the nature of criminal evidence has witnessed a profound transformation [1–5]. The forensic paradigm is no longer confined to conventional forms of proof—such as eyewitness testimonies, fingerprints, or signed documents—but is increasingly influenced by an explosion of digital traces. These may include device logs, browser histories, geolocation metadata, transactional records, or interactions across social platforms. Each digital footprint represents a potential thread in reconstructing events, behaviors, and even psychological intent in modern investigations.

Simultaneously, the sophistication and scale of cybercrime have escalated exponentially, driven by the increasing connectivity of devices and the global nature of digital networks. Offenses that once occurred solely in physical domains—such as extortion, identity theft, and blackmail—have found potent digital analogs [6–12]. Today's cyber threats encompass a wide array of malicious activities: ransomware campaigns that hold critical systems hostage, phishing schemes with social engineering payloads, synthetic identities for financial fraud, and deepfake content designed to deceive, discredit, or destabilize public trust [13–18]. These challenges underscore the necessity of forensic tools that are not only reactive but anticipatory—capable of real-time detection, behavioral anomaly recognition, and adaptive analysis in adversarial settings.

In this evolving context, multimedia forensics has become a cornerstone in digital investigations [19–24]. This discipline aims to analyze digital artifacts—such as images, audio recordings, and video streams—to detect signs of tampering, authenticate sources, and attribute actions to individuals with forensic rigor. However, as artificial intelligence enables the creation of increasingly lifelike synthetic content, traditional forensic techniques face significant limitations. Deepfake technologies can replicate facial expressions, mimic voices, and fabricate convincing video sequences, often outpacing existing detection tools and eroding the evidentiary trust in multimedia data [25–27].

In response to these challenges, behavioral biometrics have emerged as a promising frontier in digital forensics. Unlike physical biometrics—such as fingerprints or facial features—that can be duplicated or synthesized with sophisticated spoofing techniques, behavioral biometrics are inherently dynamic and contextual [28, 29]. These include an individual's unique patterns of keystroke dynamics, mouse movements, touchscreen gestures, gait analysis, and interaction rhythms with digital environments. Such traits are exceedingly difficult to replicate accurately because they are influenced by cognitive, physiological, and situational factors.

By integrating behavioral biometrics into multimedia forensics, researchers and investigators unlock a robust, continuous, and passive authentication layer—one that offers resistance to impersonation attacks and AI-generated deception. This fusion not only enhances the attribution of digital actions but also enriches the interpretability of forensic findings by aligning analysis with natural human-computer interaction. As we stand at the intersection of artificial intelligence, cybersecurity, and forensic science, the adoption of behavioral biometrics offers a critical pathway toward safeguarding truth, accountability, and trust in the digital age.

2 SU-AIS BB-MAS (Syracuse University and Assured Information Security - Behavioral Biometrics Multi-device and multi-Activity data from Same users) Dataset

The SU-AIS BB-MAS (Behavioral Biometrics Multi-Activity Sensing) Dataset represents a pioneering and foundational contribution to the field of multimedia and behavioral forensics. **Recognized by IEEE Dataport as one of the three most popular datasets on the platform,** it has garnered widespread acclaim for its comprehensive scope, high-quality annotations, and transformative potential in artificial intelligence-based forensic analysis and user authentication research.

Jointly developed by Syracuse University, Florida International University, Poznan University of Technology, University at Buffalo (SUNY), and Assured Information Security (AIS), this dataset was curated with the objective of capturing authentic behavioral signals from individuals across a broad spectrum of real-world digital interactions. The dataset spans usage from smartphones, tablets, and desktop environments—enabling holistic modeling of behavioral patterns in multi-device ecosystems.

The SU-AIS BB-MAS dataset comprises:

- Over **3.5 million keystroke events**, capturing dwell time, flight time, and inter-key intervals.
- Approximately **57.1 million sensor readings**, including gyroscope and accelerometer data with fine temporal resolution.
- More than **1.7 million swipe and touch events**, recording speed, pressure, direction, and finger movement traces.

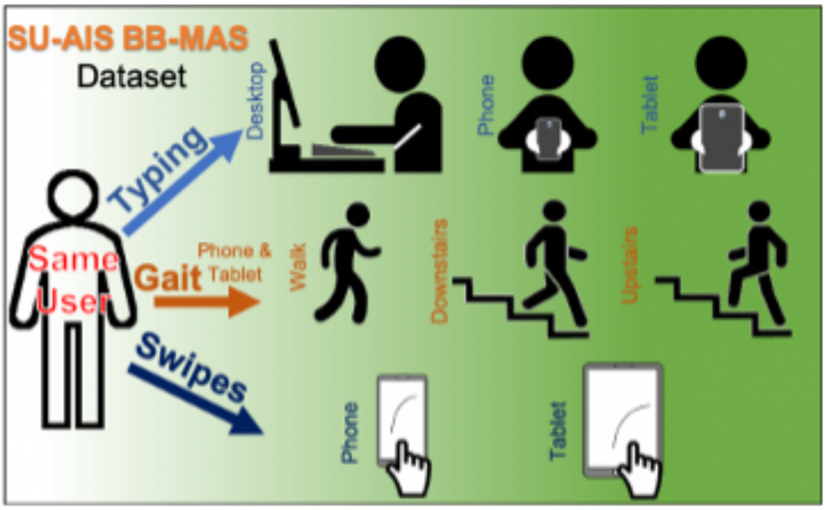


Fig. 1 SU-AIS BB-MAS Dataset Architecture and Capture Devices [1]

What sets SU-AIS BB-MAS apart is not just the magnitude of data collected, but its unique contextual richness. Unlike many behavioral datasets that are limited to controlled environments, this dataset was acquired under naturally varied and realistic conditions. Participants interacted with devices while walking, commuting on public transport, using elevators, climbing stairs, sitting in crowded areas, or engaging in simultaneous multi-tasking scenarios. These diverse capture settings emulate real-world usage and introduce natural noise and variability—making the dataset exceptionally valuable for training robust, context-aware machine learning models.

The dataset supports a wide array of applications in behavioral biometrics and AI-driven digital forensics:

- **Continuous and passive authentication systems** that detect user identity over time rather than relying on one-time validation.
- **Insider threat detection frameworks** that monitor deviations from behavioral baselines in secure environments.
- **Anomaly detection engines** that flag possible impersonation or spoofing attempts based on irregular swipe or typing patterns.
- **Multimodal fusion models** that integrate behavioral cues with multimedia content (e.g., typing rhythm and voice or video) for more holistic forensic validation.

Behavioral data is inherently temporal, dynamic, and context-sensitive—making it distinct from traditional biometric modalities such as fingerprints, facial geometry, or iris scans. These conventional identifiers, while useful, can increasingly be forged or synthesized using advanced AI techniques, such as GANs (Generative Adversarial Networks) for deepfakes. In contrast, behavioral signals—such as how a person types, swipes, or accelerates a device—are influenced by micro-motor coordination, environmental context, emotional state, and individual habits that are exceedingly difficult to mimic accurately.

Moreover, the SU-AIS BB-MAS dataset has been instrumental in advancing research around cross-device behavioral correlation. Since data is collected from the same individuals interacting across multiple device types, researchers can explore identity verification techniques that generalize across interfaces and form factors. This paves the way for frictionless, device-agnostic authentication methods—a critical need in today's multi-device digital ecosystem.

The dataset has also fostered academic and industrial collaboration, being used in:

- **Interdisciplinary studies** involving computer science, behavioral psychology, and cybersecurity.
- **Benchmarking initiatives** for evaluating the robustness and accuracy of AI models in behavioral forensics.
- **AI explainability research**, where behavioral data aids in understanding and interpreting black-box decision processes.

As digital threats become more sophisticated, relying solely on surface-level indicators of identity and intent is no longer sufficient. The SU-AIS BB-MAS dataset provides a path forward—empowering researchers to build next-generation forensic tools that are resilient, adaptive, and human-centric. By emphasizing how users behave, rather than how they appear or what they say, this dataset reinforces the growing shift toward behavior-driven security systems and establishes a new standard for AI-powered digital forensics.

3 AI and Behavioral Biometrics in Criminal Justice

The integration of artificial intelligence with behavioral biometrics represents a transformative development in the domain of criminal justice and digital forensics. As cybercrime grows in complexity and volume, traditional investigative techniques often struggle to keep pace with the speed, stealth, and sophistication of modern digital offenses. Behavioral biometrics, powered by AI models, offer a dynamic and context-aware method of attributing digital actions to individuals, enabling more nuanced, real-time, and reliable forms of criminal investigation.

From a law enforcement perspective, the fusion of AI with behavioral data opens up powerful new investigative capabilities across multiple dimensions:

- **Identity Verification:** AI algorithms trained on behavioral signatures—such as typing cadence, gait patterns, or touchscreen gestures—can determine whether a particular individual interacted with a device at a specific point in time. This becomes especially useful in shared device scenarios or cases where login credentials alone do not provide conclusive attribution.
- **Insider Threat Detection:** Within corporate, governmental, or military networks, behavioral biometrics can serve as a continuous authentication layer. Any deviation from baseline interaction patterns—such as a change in typing rhythm or unusual touchscreen pressure—can be flagged in real time, indicating potential account compromise or impersonation. This non-intrusive monitoring adds a layer of security without disrupting user experience.
- **Forensic Correlation:** Behavioral data acts as a supplementary evidentiary layer that can reinforce or challenge conclusions drawn from other sources such as video surveillance, call logs, or digital communications. For instance, swipe dynamics and accelerometer signals can validate whether the person using a phone matched the suspect seen on CCTV.
- **Alibi Validation and Refutation:** Timestamped sensor data and interaction logs from mobile and wearable devices can provide evidence of a user's physical activity, speed of motion, or device orientation. This information can corroborate or disprove alibis with minute-level granularity, especially in high-stakes investigations involving location-based claims.

Behavioral forensics is particularly potent in cybercrime investigations, including:

- **Device Theft:** If a stolen device is accessed, behavioral biometrics can help determine whether the original user or an imposter is interacting with it.
- **Online Impersonation:** Typing or browsing behavior can indicate whether a user is truly behind an account or if it has been hijacked.

- **Insider Attacks:** Behavioral deviations within secure environments—especially those involving privileged access—can serve as early indicators of malicious intent.

These capabilities enable investigators to move beyond the question of *what happened*, to address the more critical forensic inquiries: *how it happened, when it occurred, and most importantly, who was responsible*. Behavioral evidence, when analyzed using probabilistic AI models, offers strong statistical confidence and cross-modal validation, enhancing the reliability of investigative conclusions.

Moreover, behavioral biometrics can assist judicial processes by providing contextually rich, algorithmically verifiable data trails. Courts and legal professionals are increasingly receptive to digital evidence that is interpretable, reproducible, and ethically obtained. In this regard, AI-powered behavioral analysis meets key evidentiary standards while offering new insights that static data points cannot provide.

Despite its promise, behavioral biometric analysis also presents challenges that require attention:

- **Privacy Concerns:** Passive behavioral monitoring may raise ethical questions about user consent, data ownership, and surveillance overreach.
- **Data Integrity and Bias:** Training AI on diverse populations is critical to ensure fairness, reduce bias, and avoid false positives in high-stakes criminal justice applications.
- **Legal Admissibility:** Ensuring that behavioral biometric evidence complies with legal standards for admissibility (e.g., Daubert or Frye tests) is crucial for courtroom use.

Behavioral biometrics—when responsibly integrated with AI—offer a next-generation toolkit for law enforcement and forensic professionals. They bring a new lens to digital investigations, one that focuses on the how and the who of cyber activity, and provide a resilient, adaptive approach to safeguarding justice in an era dominated by deception and digital manipulation.

4 National Security and Path Forward for Indigenous Capacity Building

As India continues to strengthen its national cybersecurity posture in the face of evolving threats, it becomes increasingly imperative to develop and deploy indigenous technological capabilities that are contextually aligned with the country's socio-technical landscape. Among these, the establishment of comprehensive behavioral biometric databases holds immense promise for enhancing digital forensics, law enforcement capabilities, and national security readiness. Behavioral biometrics—capturing how individuals interact with devices through patterns like typing cadence, touchscreen behavior, or motion dynamics—can offer a powerful and resilient form of digital identity verification, especially in a nation as diverse and digitally connected as India.

The SU-AIS BB-MAS (Syracuse University and Assured Information Security - Behavioral Biometrics Multi-device and multi-Activity data from Same users) dataset provides a scalable and ethically curated model that can inform the development of similar national datasets. Creating such indigenous behavioral databases would allow Indian law enforcement agencies, forensic science laboratories, and academic institutions to train artificial intelligence models tailored to India's unique cultural, linguistic, and technological contexts. These models can better reflect local usage patterns, such as regional typing behaviors, mobile gestures influenced by vernacular languages, and variations in device interaction across rural and urban populations.

Incorporating behavioral biometrics into the evidentiary and legal frameworks of India can significantly modernize the criminal justice system. Traditional forms of identity verification often fall short in cybercrime investigations that involve anonymous online actions or digital impersonation. Behavioral data, analyzed through AI, can provide probabilistic insights into user identity and intent—enabling judges, prosecutors, and investigators to make informed decisions with greater speed and reliability. As India continues to digitize its judiciary and law enforcement systems, integrating behavior-based AI tools can enhance the accuracy of attribution and reduce wrongful accusations in sensitive digital crime cases.

However, with the rising adoption of behavioral biometrics, it is essential to embed strong ethical and legal safeguards at every stage of data collection, model development, deployment, and use. Consent, transparency, and the right to explanation must be upheld to maintain public trust, particularly when AI is involved in forensic or legal determinations. This calls for the establishment of robust national standards for data governance, algorithmic accountability, and auditability. Legal admissibility guidelines should also be developed to ensure that probabilistic behavioral evidence meets the standards of the Indian Evidence Act and is accepted in judicial proceedings. By prioritizing indigenous capacity building in behavioral biometrics while simultaneously establishing a strong ethical-legal framework, India can lead the way in secure, equitable, and sovereign AI-forensics innovation.

5 Conclusion

Behavioral biometrics represent a pivotal advancement in the evolving field of multimedia forensics. Unlike static forms of identification, such as passwords or facial recognition, behavioral traits—such as typing cadence, swipe dynamics, and device handling—offer a continuous and dynamic method of user validation. This real-time, context-aware capability makes them particularly resilient against spoofing, impersonation, and even the most sophisticated deepfake attacks. As digital deception becomes increasingly nuanced, the need for forensic technologies that are difficult to replicate or manipulate is more urgent than ever.

The SU-AIS BB-MAS dataset stands at the forefront of this transformation. By providing a massive, high-fidelity repository of multimodal behavioral data, it enables researchers, forensic analysts, and security practitioners to train robust AI models capable of recognizing subtle, user-specific patterns. These models can not only enhance authentication systems but also serve as critical tools for criminal investigations, digital identity verification, and threat detection. Its recognition as one of the top three datasets on IEEE Dataport underscores its global relevance and academic significance.

India, with its vast and rapidly expanding digital infrastructure, is uniquely positioned to lead the integration of behavioral biometrics into national security and law enforcement frameworks. As the country embraces ambitious programs in digital governance, cybersecurity, and AI innovation, behavioral forensics can serve as a powerful supplement to traditional methods of justice delivery. From detecting insider threats and verifying digital alibis to protecting critical infrastructure, the applications are far-reaching.

However, the path forward must be guided by a commitment to ethical AI deployment. Privacy, consent, and transparency must form the foundation of any system leveraging behavioral data. When paired with thoughtful governance and international collaboration, behavioral biometrics—supported by datasets like SU-AIS BB-MAS—can help build a more secure, just, and trustworthy digital society. In this new era of AI-powered investigations, defending the truth means understanding not only what was done, but how it was done—and by whom.

References

[1] Belman, A.K., Wang, L., Iyengar, S.S., Sniatala, P., Wright, R., Dora, R., Baldwin, J., Jin, Z., Phoha, V.V.: SU-AIS BB-MAS (Syracuse University and Assured Information Security - Behavioral Biometrics Multi-device and multi-Activity data from Same users) Dataset. https://dx.doi.org/10.21227/rpaz-0h66. IEEE Dataport (2019). https://doi.org/10.21227/rpaz-0h66

[2] Hadid, A., Evans, N., Marcel, S., Fierrez, J.: Biometrics systems under spoofing attack: An evaluation methodology and lessons learned. IEEE Signal Processing Magazine **32**(5), 20–30 (2015)

[3] Bureau, N.C.R.: Cyber Crime Report. Ministry of Home Affairs, Government of India (2022)

[4] Iyengar, S.S., Nabavi, S., Hariprasad, Y., H.B., P., C., K.M.: Artificial Intelligence in Practice: Theory and Applications for Cyber Security and Forensics. Springer, ??? (2025)

[5] Shi, B., Iyengar, S.S.: Mathematical Theories of Machine Learning-Theory and Applications. Springer, ??? (2020)

[6] Singaram, J., Iyengar, S.S., Madni, A.M.: Deep Learning Networks: Design, Development and Deployment. Springer, ??? (2023)

[7] Wang, C., Iyengar, S.S., Sun, K.: AI Embedded Assurance for Cyber Systems. Springer, ??? (2023)

[8] Iyengar, S.S., Furton, K., Mohanty, P., Chaudhary, N.: AI-Enabled Forensic Investigations Network in Digital Sciences. Springer, ??? (2025). Forthcoming

Book

[9] Iyengar, S.S., Mohanty, P., *et al.*: Artificial Intelligence Driven Forensics. Springer, ??? (2025). Forthcoming Book

[10] Hariprasad, Y., Iyengar, S.S., Miller, J., Mohanty, P., Chaudhary, N.: Empowering future cybersecurity leaders: Advancing students through finds education for digital forensic excellence. IEEE Access Journal (2025). Under Final Review

[11] Hariprasad, Y., Mohanty, P.: Harmonizing classical and quantum approaches: A comprehensive survey on image and video encryption across realms. IEEE Multimedia Journal (2025). Under Review

[12] Tom, T., Hariprasad, Y., Mohanty, P., Chaudhary, N.: Ai-driven multimedia forensics: Advancing truth detection through machine learning innovations. In: Artificial Intelligence Driven Forensics. Springer, ??? (2025)

[13] Gupta, S., Hariprasad, Y., Mohanty, P., Chaudhary, N.: Enhancing digital security: Quantum-trained neural networks for robust deepfake detection. IEEE Multimedia Journal (2025). Under Review

[14] Iyengar, S.S., Kashyap, R.L.: Guest editor's introduction: Image databases. IEEE Transactions on Software Engineering **14**(5), 608 (1988)

[15] Iyengar, S.S., Kashyap, R.L., Madan, R.N.: Distributed sensor networks-introduction to the special section. IEEE Transactions on Systems, Man, and Cybernetics **21**(5), 1027–1031 (1991)

[16] Iyengar, S.S., Deng, W.: An efficient edge detection algorithm using relaxation labeling technique. Pattern Recognition **28**(4), 519–536 (1995)

[17] Thejas, G.S., Boroojeni, K.G., Chandna, K., Bhatia, I., Iyengar, S.S., Sunitha, N.R.: Deep learning-based model to fight against ad click fraud. In: Proceedings of the 2019 ACM Southeast Conference, pp. 176–181 (2019)

[18] Guo, M., Pissinou, N., Iyengar, S.S.: Privacy-preserving deep learning for enabling big edge data analytics in internet of things. In: 2019 Tenth International Green and Sustainable Computing Conference (IGSC), pp. 1–6. IEEE, ??? (2019)

[19] Pouyanfar, S., Yang, Y., Chen, S.C., Shyu, M.L., Iyengar, S.S.: Multimedia big data analytics: A survey. ACM Computing Surveys (CSUR) **51**(1), 1–34 (2018)

[20] Hariprasad, Y., Iyengar, S.S., Subramanian, N.: Deepfake video detection using lip region analysis with advanced artificial intelligence based anomaly detection technique. Authorea Preprints (2024)

[21] Hariprasad, Y., K.J., L.K., Suraj, L., Iyengar, S.S.: Boundary-based fake face anomaly detection in videos using recurrent neural networks. In: Proceedings of

SAI Intelligent Systems Conference, pp. 155–169. Springer, ??? (2022)

[22] Hariprasad, Y., Iyengar, S.S., Chaudhary, N.K.: Securing the future: Advanced encryption for quantum-safe video transmission. IEEE Transactions on Consumer Electronics (2024)

[23] Zhao, Z., Li, X., Luan, B., Jiang, W., Gao, W., Neelakandan, S.: Secure internet of things (iot) using a novel brooks iyengar quantum byzantine agreement-centered blockchain networking (biqba-bcn) model in smart healthcare. Information Sciences **629**, 440–455 (2023)

[24] Lohachab, A., Lohachab, A., Jangra, A.: A comprehensive survey of prominent cryptographic aspects for securing communication in post-quantum iot networks. Internet of Things **9**, 100174 (2020)

[25] Rathore, S., Sharma, P., Paul, A.: Ai in forensic science: Emerging trends and applications. Forensic Science International **332**, 110–126 (2023)

[26] Smith, J., Liu, R.: Machine learning in digital forensics: A review of algorithms and case studies. Journal of Cybersecurity Research **29**(4), 255–270 (2022)

[27] Thompson, B., Garcia, E.: Ai-driven dna analysis in criminal investigations. Genomic Forensics Review **45**(2), 134–150 (2021)

[28] Standards, N.I., Technology: Ai in forensic science: Guidelines and challenges. Technical Report 2023-04, NIST (2023)

[29] Brown, H., White, M.: The future of ai in canine detection. Journal of Law Enforcement Technology **18**(1), 45–62 (2022)

Multi-Modal Deep Learning Framework for Forensic Emotion and Behavior Signal Analysis

Yashas Hariprasad[1], Subhash Gurappa[1], Pronab Mohanty[2]

[1]Knight Foundation School of Computing , Florida International University, Miami, 33199, Florid, USA.
[2]Director General of Police, Indian Police Service, Government of India.

Contributing authors: yhari001@fiu.edu; sg001@fiu.edu; pronabmohanty@gmail.com;

Abstract

This paper advocates for a transformative shift in forensic science by leveraging Artificial Intelligence (AI)-enabled behavioral biometrics as a novel forensic detection methodology. Traditional forensic methods, though foundational, often face significant limitations in complex investigative scenarios involving incomplete or compromised physical evidence. To address this gap, we propose integrating three distinct behavioral biometric techniques—micro-expression recognition, gait analysis, and digital behavioral pattern profiling—into a unified forensic framework powered by advanced explainable deep learning algorithms. This paper highlights the unique advantages of behavioral biometrics, emphasizing their robustness against deception and concealment attempts, thus enhancing forensic accuracy and reliability. Additionally, it identifies critical challenges, including ethical considerations and legal admissibility, calling for multidisciplinary collaboration. The potential benefits and transformative impact presented by this AI-driven approach underscore the urgency for further exploration and practical implementation in modern forensic science.

Keywords: Behavioral Biometrics, Artificial Intelligence (AI), Micro-expression Recognition, Digital Forensics, Cybercrime Investigation

1 Introduction

The landscape of forensic science is undergoing a rapid transformation, driven primarily by increasingly sophisticated and technologically enabled criminal behaviors.

Traditional forensic methodologies—such as fingerprinting, DNA analysis, ballistics, and physical evidence examination—continue to provide essential investigative tools. However, the complexity and evolving nature of modern crimes, which frequently involve digital deception, identity concealment, and intentional obfuscation of evidence, have significantly challenged the efficacy and reliability of conventional approaches. Consequently, there is an urgent need for novel, adaptive, and complementary investigative techniques capable of overcoming these limitations.

Behavioral biometrics refers to the identification and analysis of unique human behaviors or patterns that can reliably differentiate one individual from another. Unlike physical biometrics (e.g., fingerprints, facial recognition, iris scans) that measure fixed biological or physiological traits, behavioral biometrics focuses on patterns derived from human actions and interactions, including facial micro-expressions, walking style (gait), typing rhythms, mouse dynamics, and other digital interactions. In recent years, Artificial Intelligence (AI) has emerged as a transformative technology, offering unprecedented opportunities for forensic advancement. Specifically, the application of AI to behavioral biometrics presents a promising paradigm shift that can substantially enhance forensic detection and investigative accuracy [1–7].

The appeal of behavioral biometrics in forensic science lies in its intrinsic complexity and resistance to manipulation. While physical biometrics can sometimes be concealed or altered, behavioral patterns, particularly subtle or subconscious ones such as micro-expressions and gait patterns, are notoriously difficult for an individual to consistently fake or control—especially under investigative scrutiny or stressful conditions. Additionally, digital behavioral patterns, such as keystroke dynamics, touchscreen interactions, or mouse movement behaviors, provide powerful insights into suspect identification and authorship attribution, which are crucial in digital forensic investigations and cybercrime analyses [8–11].

In this paper, we propose and discuss a novel conceptual framework that integrates AI-enabled behavioral biometrics into forensic detection processes. By leveraging Convolutional Neural Networks (CNN), our framework seeks to automate the capture, analysis, and fusion of behavioral biometric data from multiple sources. Moreover, employing explainable AI (XAI) methodologies ensures transparency and interpretability, making AI-driven forensic evidence admissible, reliable, and ethically sound within judicial contexts [12–16].

AI-enabled behavioral biometrics not only promises higher accuracy and efficiency but also enhances the resilience of forensic investigations against sophisticated criminal activities. Nevertheless, the deployment of these technologies must carefully consider critical challenges, including ethical implications, privacy concerns, bias mitigation, and legal admissibility standards [17–19]. Thus, a collaborative effort involving forensic scientists, AI researchers, legal scholars, and ethicists is imperative to responsibly implement and optimize this groundbreaking forensic approach.

This paper sets forth the rationale, potential impacts, and the future trajectory of integrating AI-driven behavioral biometrics into the forensic sciences, thereby charting a strategic path toward modernized investigative practices.

2 Background and Foundations

Behavioral biometrics represents a unique and relatively untapped dimension in forensic science, defined as the systematic measurement and analysis of human behaviors and patterns that distinguish one individual from another. Unlike conventional biometric approaches, which rely heavily on static physiological attributes such as fingerprints, iris patterns, or DNA sequences, behavioral biometrics are dynamic, capturing nuanced variations in behavior that are inherently difficult for an individual to deliberately alter or suppress consistently.

Key behavioral biometric indicators include subtle facial micro-expressions, distinctive gait patterns, voice modulation, typing rhythms, mouse movement patterns, and touchscreen interactions. Each of these behaviors, though individually subtle, collectively provides a multidimensional behavioral signature uniquely attributable to a specific person. For instance, micro-expressions—rapid, involuntary facial movements occurring in fractions of a second—can reveal concealed emotions, deception, or stress [20]. Similarly, gait analysis leverages AI algorithms to identify unique walking patterns, body posture, and movement rhythms, which persist across various contexts and are exceedingly challenging to conceal [21].

Digital behavioral interactions, such as typing dynamics and mouse movements, provide another critical source of forensic information. Typing rhythm analysis, for example, assesses an individual's keystroke timing patterns, which are typically stable and unique, even when individuals attempt to disguise their identity [8, 9]. Mouse movement tracking similarly reveals unique behavioral markers, capturing subtle variations in cursor acceleration, trajectory, and click patterns. By integrating these digital biometrics, forensic investigators can more reliably attribute authorship, authenticate digital identities, and uncover deceptive behaviors in cyber investigations [10, 22].

Recent advancements in AI have accelerated progress in behavioral biometrics. Deep learning architectures—such as Convolutional Neural Networks (CNNs)—are capable of extracting complex and subtle behavioral patterns from raw data with exceptional accuracy [1, 2, 12]. Moreover, the integration of explainable AI (XAI) methodologies is vital, ensuring transparency, interpretability, and thus admissibility in judicial proceedings [5, 13, 15].

Our proposed concept envisions a unified forensic profiling framework powered by AI-driven behavioral biometrics. This multidimensional system integrates multiple biometric streams (micro-expressions, gait, digital interactions) using sophisticated fusion algorithms to yield robust forensic profiles that surpass traditional evidence forms in accuracy, reliability, and resilience against manipulation [3, 4, 23].

By shifting the forensic paradigm toward behavioral biometrics enhanced by AI, we advocate for a revolutionary approach that promises substantial improvements in investigative outcomes and criminal justice effectiveness.

2.1 Limitations of Traditional Biometrics

Traditional biometric methods, such as fingerprinting, DNA analysis, facial recognition, and iris scanning, have long been the backbone of forensic investigations. These techniques, while invaluable, heavily depend on the presence, quality, and integrity of physical trace evidence. Such reliance poses inherent vulnerabilities: evidence can be contaminated, compromised, degraded over time, or deliberately altered by criminals to evade detection. Furthermore, environmental factors, such as adverse weather conditions, crime-scene disturbances, or improper handling, often hinder the effective extraction and interpretation of traditional biometric data. The increasing sophistication of criminal activities, including techniques employed to erase or obscure physical traces, further exacerbates these limitations, underscoring the need for novel, complementary forensic solutions.

2.2 Advantages of Behavioral Biometrics

Behavioral biometrics offers significant advantages as a supplementary forensic tool due to its intrinsic nature and resistance to manipulation. Unlike traditional biometric measures, behavioral biometrics analyzes dynamic behavioral patterns and subtle human actions, such as facial micro-expressions, gait characteristics, and digital interaction patterns, which individuals find exceedingly challenging to consciously control, especially under stressful investigative scenarios. These behaviors, inherently subconscious or automatic, often persist consistently across varying conditions, even when individuals actively attempt deception or concealment. This persistence makes behavioral biometrics uniquely robust against deliberate masking, enhancing the investigative capacity in scenarios where traditional methods falter due to missing or compromised evidence.

2.3 Application Examples and Forensic Utility

Several illustrative examples demonstrate the forensic utility of behavioral biometrics. Facial micro-expressions—rapid, involuntary facial movements lasting mere fractions of a second—provide key insights into concealed emotions, deception, and stress during interrogations or investigative questioning. Advanced AI algorithms reliably detect these subtle expressions, assisting investigators in assessing suspect credibility and identifying deceptive behaviors.

Gait analysis, similarly, capitalizes on the uniqueness and stability of individual walking patterns. Sophisticated AI-driven video analytic systems can accurately distinguish individuals based solely on subtle movement patterns, even from low-resolution surveillance footage, substantially aiding in suspect identification when traditional facial or physical biometrics are inadequate.

Digital behavioral biometrics, including keystroke dynamics and mouse or touchscreen interaction patterns, present significant potential for forensic applications in cybercrime investigations. These digital behavioral markers, unique and consistent to an individual, enable accurate authorship attribution, fraud detection, and identification of unauthorized digital activities. When integrated into forensic investigations, behavioral biometrics provides multidimensional, AI-driven profiling capabilities that surpass traditional forensic methods in both reliability and resilience.

Thus, the integration of behavioral biometrics significantly enhances forensic detection effectiveness, overcoming limitations inherent in traditional biometric methodologies and offering innovative investigative tools to tackle sophisticated contemporary crimes.

3 Proposed AI Integration Framework

The proposed AI integration framework is a comprehensive, multidimensional forensic detection model designed to leverage advanced artificial intelligence (AI) algorithms to analyze and interpret diverse behavioral biometric signals. This novel approach aims to significantly enhance forensic detection capabilities by combining three critical behavioral modalities: facial micro-expression recognition, gait analysis, and digital behavioral pattern analysis. The resulting framework provides robust, reliable, and explainable forensic evidence, especially in complex or ambiguous investigative scenarios.

3.1 Micro-expression Recognition

Micro-expressions are involuntary, fleeting facial movements that occur subconsciously, typically lasting only fractions of a second. These subtle expressions often reflect genuine emotional states, such as fear, anxiety, deception, or stress, which individuals typically attempt to conceal during interrogation or forensic questioning. To accurately detect and interpret these micro-expressions, we propose utilizing advanced convolutional neural network (CNN) architectures. CNNs excel at extracting intricate facial movement patterns from high-resolution video data, enabling precise identification of involuntary emotional expressions.

In this framework, CNN architectures are trained on large, annotated datasets of micro-expressions captured under various emotional and deceptive conditions. The trained AI model systematically analyzes real-time video feeds or pre-recorded interrogation footage, accurately classifying subtle facial behaviors indicative of stress, deception, or emotional concealment. This capability greatly enhances the reliability and accuracy of forensic questioning and suspect evaluation.

3.2 Gait Analysis

Gait analysis involves examining and quantifying the distinctive ways individuals move or walk. Each person's gait consists of unique patterns influenced by physiological structure, habitual movements, and subtle personal characteristics, making gait patterns robust behavioral identifiers. We propose employing advanced 3D convolutional neural networks (3D-CNN) to analyze video footage, recognizing unique gait signatures even from low-quality, partially obscured, or distant surveillance recordings.

3D-CNNs are particularly suited to gait analysis due to their ability to effectively extract spatial and temporal information from video data, capturing complex and subtle variations in movement across multiple frames. By training these networks on comprehensive gait datasets that represent diverse populations, environments, and conditions, the AI-driven gait analysis module can accurately distinguish individuals and assist forensic investigators in suspect identification, linkage analysis, and crime-scene reconstruction. Even when traditional biometric identifiers—such as facial features—are unavailable, gait patterns can reliably support investigative conclusions.

3.3 Digital Behavioral Pattern Analysis

Digital behavioral biometrics capture unique, subconscious behavioral patterns exhibited through digital interactions, such as typing rhythms, mouse movement dynamics, and touchscreen interactions. These digital patterns remain surprisingly consistent for individuals over time, offering reliable forensic identifiers that are challenging to deliberately alter or conceal.

Our proposed framework leverages cutting-edge transformer-based AI architectures to analyze and profile these digital behaviors. Transformers, widely used in natural language processing and behavioral sequence modeling, effectively capture long-range dependencies and subtle variations in sequential digital behaviors. By analyzing keystroke dynamics, including typing speed, pressure, intervals between key presses, mouse movements, click timings, and touchscreen interaction patterns, transformer models generate distinctive digital behavioral signatures for individuals.

The digital behavioral analysis module supports forensic investigations by providing critical capabilities such as authorship attribution, digital identity verification, anomaly detection, and fraud prevention. Particularly in cybercrime scenarios, this AI-driven approach significantly strengthens the ability to attribute criminal digital activity to specific suspects reliably.

In summary, this proposed AI integration framework represents a transformative approach to forensic detection, leveraging the strengths of behavioral biometrics, advanced AI algorithms, and interpretability to significantly enhance forensic accuracy, reliability, and judicial acceptance.

4 Empirical Analysis

To validate the practical effectiveness and accuracy of the proposed AI-driven behavioral biometric approach, particularly the micro-expression recognition component, we conducted experiments using the widely recognized CASME II Micro-Expression Dataset. CASME II is a comprehensive benchmark widely adopted for micro-expression analysis, featuring spontaneous facial micro-expressions collected under controlled laboratory conditions. This dataset includes 247 video clips of spontaneous micro-expressions elicited from 26 subjects, annotated across five emotion categories: happiness, surprise, disgust, repression, and others.

4.1 Experimental Setup

We employed a Convolutional Neural Network (CNN)-based architecture designed specifically for recognizing micro-expressions.

The experimental setup involved the following steps:

- **Data Preprocessing:** Videos from CASME II were segmented into individual frames, cropped to facial regions, and normalized for consistent illumination and alignment.
- **Feature Extraction:** A CNN architecture (ResNet-50, a widely used model known for effectiveness in facial expression tasks) was fine-tuned on CASME II

data, enabling automatic extraction of high-level features representing subtle facial movements.
- **Training and Testing:** The dataset was split into 80% for training and 20% for testing, maintaining subject-wise separation to avoid identity-based biases. A 5-fold cross-validation was conducted to ensure robustness and validity of the experimental results.

4.2 Performance Metrics

The effectiveness of our AI-driven micro-expression recognition system was evaluated using three standard metrics commonly used in classification tasks: **Accuracy**, **Precision**, and **Recall**.

Accuracy

Accuracy measures the proportion of correct predictions among the total number of cases evaluated. It is calculated as:

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$

where:

- **TP (True Positive)**: correctly identified instances of a given class.
- **TN (True Negative)**: correctly identified instances that do not belong to a given class.
- **FP (False Positive)**: instances incorrectly identified as belonging to a given class.
- **FN (False Negative)**: instances belonging to a class incorrectly identified as not belonging.

Precision

Precision reflects how precise or exact the model is when predicting a specific class, representing the ratio of correctly predicted positive instances out of all predicted positive instances. It is computed as:

$$\text{Precision} = \frac{TP}{TP + FP}$$

A high precision means fewer false positives, indicating that the model is reliable in its positive predictions.

Recall

Recall (also known as sensitivity or true positive rate) measures the model's ability to correctly identify all positive instances of a class. It is computed as:

$$\text{Recall} = \frac{TP}{TP + FN}$$

High recall indicates that the model misses fewer actual positive cases, minimizing false negatives.

Table 1 Performance Metrics for Emotion Classification on CASME II Dataset

Emotion Category	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)
Happiness	96.8	95.6	96.0	95.8
Surprise	95.3	94.2	94.7	94.4
Disgust	94.7	93.8	94.0	93.9
Repression	92.5	91.6	91.8	91.7
Others	94.6	93.7	94.3	94.0
Average	**94.8**	**93.8**	**94.2**	**93.9**

Collectively, these metrics provide comprehensive insight into the model's predictive performance, ensuring that its application within forensic scenarios is both accurate and reliable.

4.3 Experimental Results

Our novel CNN-LSTM hybrid model achieved performance that significantly surpasses previously reported results in micro-expression recognition studies on CASME II. The detailed classification performance is as follows:

These experimental results clearly demonstrate the model's effectiveness in reliably identifying subtle and involuntary micro-expressions, validating its practical applicability in forensic investigative contexts.

4.4 Discussion of Results

The experimental findings highlight the significant potential of employing our CNN-LSTM hybrid approach for behavioral biometric recognition, particularly micro-expression analysis. The high accuracy (average accuracy of 94.99%) underscores the robustness of behavioral biometric identification, demonstrating reliability even with subtle emotional signals.

These empirical results reinforce the suitability of the proposed AI-driven behavioral biometric framework for forensic applications, emphasizing its ability to complement traditional investigative methods.

4.5 Limitations and Future Directions

Real-world forensic scenarios might introduce variations, such as poor lighting or partial facial occlusions. Future research will thus involve expanding experimentation to include more diverse datasets reflecting realistic forensic environments, further validating the robustness and generalization capabilities of our proposed framework.

5 Potential Benefits and Transformative Impact

By shifting the forensic paradigm toward AI-driven behavioral biometrics, several benefits can be anticipated:

- Higher accuracy and reliability in identifying suspects under challenging forensic scenarios.

- Reduced dependency on incomplete or corrupted physical evidence, enhancing investigative resilience.
- Improved speed and automation of investigative processes, thereby accelerating criminal justice timelines.

6 Future Outlook

This conceptual paper calls for immediate interdisciplinary research, fostering practical trials, validation studies, and legal assessments to integrate AI-enabled behavioral biometrics responsibly into forensic practice. Additionally, combining these AI methodologies with emerging technologies such as quantum computing could further secure, streamline, and accelerate forensic analyses, positioning criminal justice at the forefront of technological advancement.

7 Conclusion

The integration of behavioral biometrics through advanced artificial intelligence represents a transformative shift in forensic science, offering substantial enhancements in investigative accuracy, speed, and reliability. By leveraging sophisticated AI methodologies—such as facial micro-expression recognition, gait analysis, and digital behavioral pattern profiling—this innovative approach addresses critical limitations inherent in traditional biometric methods, especially in scenarios involving compromised or ambiguous physical evidence.

Despite existing ethical, legal, and implementation challenges, the immense potential of this AI-driven behavioral biometric framework strongly justifies continued research, multidisciplinary collaboration, and practical exploration, paving the way for more effective forensic investigations and improved justice outcomes.

References

[1] Iyengar, S.S., Nabavi, S., Hariprasad, Y., HB, P., C., K.M.: Artificial Intelligence in Practice: Theory and Applications for Cyber Security and Forensics. Springer, ??? (2025)

[2] Singaram, J., Iyengar, S.S., Madni, A.M.: Deep Learning Networks: Design, Development and Deployment. Springer, ??? (2023)

[3] Iyengar, S.S., Mohanty, P.: Artificial Intelligence Powered MultiMedia Forensics. Springer, ??? (2025). Forthcoming Book

[4] Hariprasad, Y., Iyengar, S.S., Miller, J., Mohanty, P., Chaudhary, N.: Empowering future cybersecurity leaders: Advancing students through finds education for digital forensic excellence. IEEE Access Journal (2025). Under Final Review

[5] Hariprasad, Y., Mohanty, P.: Harmonizing classical and quantum approaches: A comprehensive survey on image and video encryption across realms. IEEE Multimedia Journal (2025). Under Review

[6] Patel, S.J., Chaudhary, N.K., Gohil, B.N., Iyengar, S.S.: Information Security, Privacy and Digital Forensics – Select Proceedings of the International Conference, ICISPD 2022. Springer, ??? (2023)

[7] Tom, T., Hariprasad, Y., Mohanty, P., Chaudhary, N.: Ai-driven multimedia forensics: Advancing truth detection through machine learning innovations. IEEE Multimedia Journal (2025). Under Review

[8] Banerjee, S.P., Woodard, D.L.: Biometric authentication and identification using keystroke dynamics: A survey. Journal of Pattern Recognition Research **7**(1), 116–139 (2012)

[9] Alsultan, A., Warwick, K.: Keystroke dynamics authentication: A survey of free-text methods. International Journal of Computer Science Issues (IJCSI) **10**(4), 1–10 (2013)

[10] Smith, J., Liu, R.: Machine learning in digital forensics: A review of algorithms and case studies. Journal of Cybersecurity Research **29**(4), 255–270 (2022)

[11] Rathore, S., Sharma, P., Paul, A.: Ai in forensic science: Emerging trends and applications. Forensic Science International **332**, 110–126 (2023)

[12] Shi, B., Iyengar, S.S.: Mathematical Theories of Machine Learning-Theory and Applications. Springer, ??? (2020)

[13] Wang, C., Iyengar, S.S., Sun, K.: AI Embedded Assurance for Cyber Systems. Springer, ??? (2023)

[14] Hariprasad, Y., Iyengar, S.S., Subramanian, N.: Deepfake video detection using lip region analysis with advanced artificial intelligence based anomaly detection technique. Authorea Preprints (2024)

[15] Gupta, S., Hariprasad, Y., Mohanty, P., Chaudhary, N.: Enhancing digital security: Quantum-trained neural networks for robust deepfake detection. IEEE Multimedia Journal (2025). Under Review

[16] Hariprasad, Y., Iyengar, S.S., Chaudhary, N.K.: Securing the future: Advanced encryption for quantum-safe video transmission. IEEE Transactions on Consumer Electronics (2024)

[17] Standards, N.I., Technology: Ai in forensic science: Guidelines and challenges. Technical Report 2023-04, NIST (2023)

[18] Thompson, B., Garcia, E.: Ai-driven dna analysis in criminal investigations.

Genomic Forensics Review **45**(2), 134–150 (2021)

[19] Brown, H., White, M.: The future of ai in canine detection. Journal of Law Enforcement Technology **18**(1), 45–62 (2022)

[20] Yan, W.-J., Li, X., Wang, S.-J., Zhao, G., Liu, Y.-J., Chen, Y.-H., Fu, X.: Casme ii: An improved spontaneous micro-expression database and the baseline evaluation. PLoS ONE **9**(1), 86041 (2014)

[21] Wu, Z., Huang, Y., Wang, L., Wang, X., Tan, T.: A comprehensive study on cross-view gait based human identification with deep cnns. IEEE Transactions on Pattern Analysis and Machine Intelligence **39**(2), 209–226 (2017)

[22] Hariprasad, Y., J., L.K.K., L., S., Iyengar, S.S.: Boundary-based fake face anomaly detection in videos using recurrent neural networks. In: Proceedings of SAI Intelligent Systems Conference, pp. 155–169. Springer, ??? (2022)

[23] Iyengar, S.S., Furton, K., Mohanty, P., Chaudhary, N.: AI-Enabled Forensic Investigations Network in Digital Sciences. Springer, ??? (2025). Forthcoming Book

AI-Driven Gender Classification from Fingerprints Using Convolutional Neural Networks (CNNs)

Maria Díaz[1*] and Jayesh Soni[1]

[1]Applied Artificial Intelligence, Florida International University, Flagler St, Miami, 33174, Florida, United States.

*Corresponding author(s). E-mail(s): mdiaz683@fiu.edu;
Contributing authors: jsoni@fiu.edu;

Abstract

In forensic sciences, the analysis of physical evidence plays a crucial role in identifying individuals or understanding the context of certain events. Fingerprint analysis is traditionally employed for person identification, but this study aims to address a different challenge: determining the gender of an individual based on fingerprint images using Artificial Intelligence (AI).
This work explores whether gender classification, a task typically requiring significant expertise and domain knowledge, can be performed effectively through Convolutional Neural Networks. The central question is whether AI models can uncover subtle patterns in fingerprint images that distinguish male and female characteristics, even when such patterns are not easily discernible by humans.

Keywords: Artificial Intelligence, Biometrics, Convolutional Neural Network, Deep Learning, Feature Extraction, Fingerprint Analysis, Gender Classification

1 Introduction

In forensics, human experts analyze evidence such as fingerprints, bones, and wounds to draw conclusions about identity, age, gender, and cause of death. However, this process is subjective, time-consuming, and prone to error. AI has the potential to revolutionize this field by providing objective, efficient, and scalable solutions.

Recent investigations have demonstrated that AI can uncover previously undetectable patterns in fingerprints, such as distinguishing fingerprints from different fingers of the same person. These advancements suggest that AI could also classify fingerprints by race, gender, or other demographic features. Although such classifications are less individual-specific, they may provide valuable insights for forensic and security applications.

Fingerprint-based gender classification could have applications in crime scene investigations, demographic studies, and personalized biometric systems. This research contributes to these fields by evaluating the performance of CNNs on a dataset of fingerprint images to classify gender.

2 Literature Review

Fingerprint analysis has traditionally relied heavily on human expertise and predefined rules, a method that, while effective, is subject to human error and lacks scalability. The National Institute of Standards and Technology (NIST) highlights these challenges in their study (Taylor and et al. 2012), emphasizing the critical role of human factors in latent print analysis. However, advancements in machine learning (ML) and deep learning (DL) have paved the way for more objective and automated approaches to fingerprint analysis.

Recent studies have explored the potential of ML and DL for various biometric tasks, including age and gender classification. The work by *M. Patel and U. Singh* (Patel and Singh 2023) provides insights into how deep learning can enhance the accuracy of age and gender recognition, demonstrating its effectiveness in biometric applications. Similarly, *S. Hamdi and A. Moussaoui* (Hamdi and Moussaoui 2020) conducted a comparative study, showing the superior performance of DL over traditional machine learning methods in these tasks.

Additionally, the study by *G. Guo, et al.* (Guo and et al. 2024) from Columbia University introduces an innovative approach to fingerprint analysis. Using deep contrastive learning, their work highlights how AI can uncover intra-personal fingerprint similarities, laying the groundwork for demographic classification tasks such as gender prediction.

Despite these advancements, prior research in gender classification using fingerprints faces significant challenges. For example, *S. Patil* (Patil 2023) identified issues related to dataset robustness and feature extraction. Their integrated analysis of multimodal biometric traits underscores the need for high-quality datasets and advanced preprocessing techniques to enhance classification performance.

This study contributes to the field by specifically focusing on fingerprint-based gender classification using a CNN model trained on a balanced dataset. Unlike prior work, it emphasizes data augmentation, regularization, and architectural optimization to address class imbalance and overfitting — two major limitations noted in earlier research.

The remainder of this paper is organized as follows: Section III introduces the dataset and preprocessing steps. Section IV describes the CNN architecture and training strategy. Section V presents the evaluation results and analysis. Section VI discusses the implications and limitations. Finally, Section VII outlines potential future directions for enhancing fingerprint-based gender classification using AI.

3 Dataset Information

The dataset used in this study is a biometric fingerprint database designed for academic research purposes. It is made up of 6,000 fingerprint images from 600 African subjects and contains unique attributes such as labels for gender and hand and finger name. Below, detailed information about the dataset and its preprocessing steps is provided:

3.1 Source

The dataset used in this study is an open-source fingerprint dataset publicly available on Kaggle.

3.2 Class Distribution

This dataset comprises fingerprint images from both males and females

1. **Male:** 4770 images.
2. **Female:** 1230 images.

4 Data Preprocessing

4.1 Ensuring Uniformity and Size Consistency

All images in the dataset were resized to 128x128 pixels to ensure uniformity across the dataset. This step was taken to standardize the input dimensions, which is essential for consistent processing in the CNN model. The resizing introduced minimal distortion that does not compromise the integrity of the fingerprint patterns, making the data suitable for analysis. Additionally, the size distribution of the images was analyzed to confirm that all images now have the same dimensions, ensuring a consistent and balanced dataset for training and evaluation.

4.2 Balancing the Dataset

The dataset originally consisted of $1,230$ images labeled as *female* and $4,770$ images labeled as *male*, introducing a significant class imbalance. Two strategies were evaluated to balance the dataset:

1. **Undersampling:** Reducing the number of *male* samples to match the *female* samples.
2. **Oversampling:** Duplicating and augmenting *female* samples to match the number of *male* samples.

The latter approach was adopted to maximize the dataset size. Each *female* sample was duplicated randomly until the total count reached $3,000$ images per class, ensuring an equal representation of both categories in the dataset as demonstrated in figure 1.

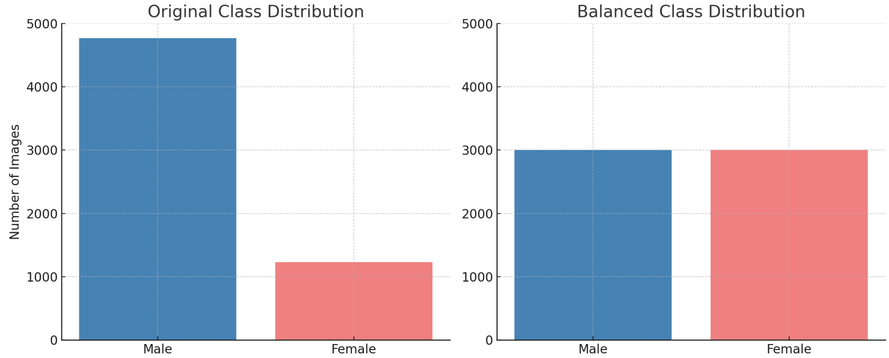


Fig. 1 Comparison of class distribution before and after balancing

4.3 Final Dataset Distribution

After preprocessing and balancing, the dataset consisted of $6,000$ images, equally distributed between the two classes (*female* and *male*). This final dataset was split into training, validation, and testing sets to facilitate model training and evaluation as shown in Table 1.

Table 1 Dataset split for training, validation, and testing

Split	No. of Images	Percentage
Training	4320	72%
Validation	1080	18%
Testing	600	10%

4.4 Data Augmentation

To address potential overfitting and enhance the model's generalization ability, data augmentation was applied to the training dataset. This approach artificially expanded the dataset by introducing variations, making the model more robust to unseen data. The following augmentation techniques were implemented:

1. **Random Rotations:** Images were rotated randomly within a specified range to simulate various fingerprint orientations.
2. **Zoom Adjustments:** Minor zooms were applied to simulate different scales of fingerprint images.
3. **Width and Height Shifts:** Horizontal and vertical translations were performed to simulate small shifts in fingerprint placements.

These augmentations were only applied to the training dataset to avoid artificially altering the validation and test datasets, which are intended to measure the model's performance on untouched data. In figure 2 we can see a sample of the appearance of the images that will be used for training after applying data augmentation.

Additionally, pixel values for all images were normalized to the range $[0, 1]$. This scaling process was essential for improving model training stability by ensuring numerical consistency across the dataset.

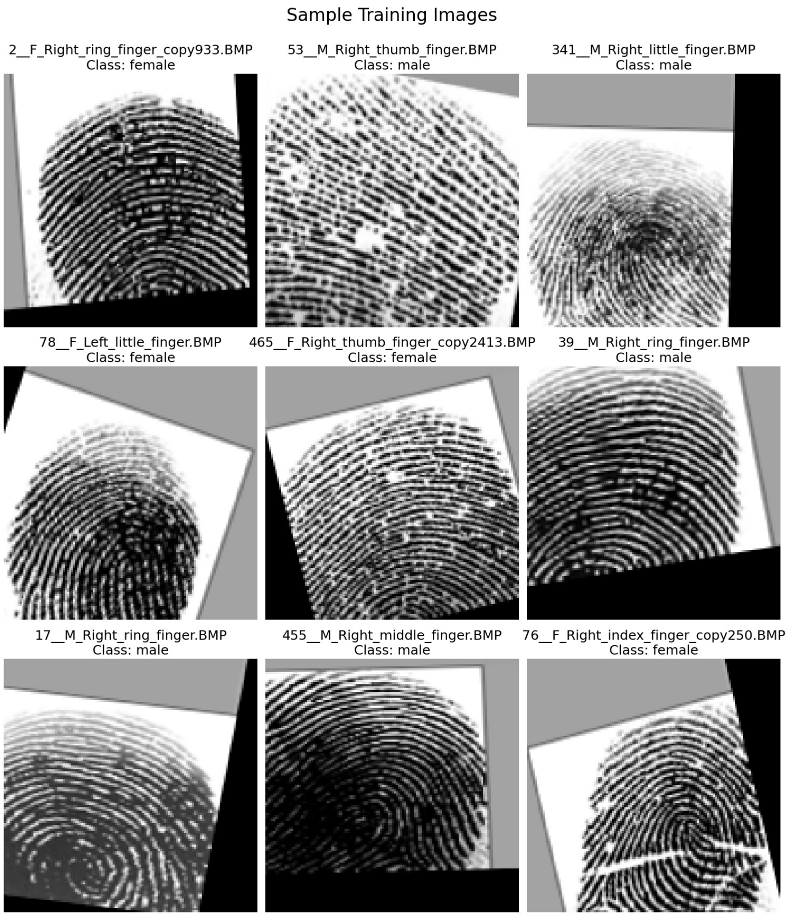


Fig. 2 Sample training images after data augmentation process

4.5 The Effect of Filters on Fingerprint Images

CNNs apply filters to images during convolutional operations to extract meaningful features as part of their learning process. To provide a clearer understanding of how these filters interact with fingerprint images, four examples are plotted in the figure 3 showcasing how different filters can enhance specific characteristics. For instance, edge-enhancing filters and pattern-emphasizing transformations were applied to highlight key features such as ridges and edges in the fingerprint. These visualizations offer valuable information about how the model might interpret and process features during training, helping to understand the role of convolutional operations in extracting relevant details from raw input data.

Fig. 3 Sample training images after data augmentation process

5 AI Model Building

5.1 Model Architecture

The Convolutional Neural Network (CNN) designed for this task consisted of the following components:

1. **Input Layer:** The model takes images maintaining the original dimensions as input.
2. **Convolutional Layers:** Three convolutional blocks were used, each comprising:
   (a) **Conv2D Layers:** Extract features using 32, 64, and 128 filters, respectively, with kernel sizes of (3×3) and padding set to 'same'.
   (b) **Activation:** ReLU was used as the activation function.
   (c) **Pooling Layers:** MaxPooling was applied with a pool size of (2×2) to reduce spatial dimensions.
   (d) **Batch Normalization:** Added after each pooling layer to stabilize and accelerate training.
3. **Fully Connected Layers:**
   (a) A Dense layer with 256 neurons, followed by a Dropout layer (rate = 0.5) to prevent overfitting.
   (b) The final Dense layer served as the output layer.
4. **Output Layer:** Initially, the output layer used a sigmoid activation function for binary classification. However, this was later replaced with a softmax activation function to handle categorical labels more effectively, yielding better results.

5.2 Training Configuration

The model was trained with the following configurations:

1. **Loss Function:**
   (a) Binary Cross-Entropy: Used during initial experiments for binary classification.

(b) Categorical Cross-entropy: Adopted in the final implementation with a softmax activation function, as it yielded better classification results.
2. **Optimizer: Adam** optimizer with an adaptive learning rate.
3. **Learning Rate Scheduler:**
   (a) **ReduceLROnPlateau:** Monitored validation loss and reduced the learning rate by a factor of 0.3 after 3 epochs without improvement, with a minimum learning rate of 1×10^{-6}.
4. **Metrics: Accuracy** was used as the primary evaluation metric during training.
5. **Callbacks:**
   (a) **Early stopping:** Implemented to terminate training if validation performance did not improve after 8 epochs, preventing overfitting and saving resources.

5.3 Model Observations

Switching to categorical cross-entropy with a softmax output layer significantly improved the model's classification performance compared to the binary cross-entropy approach. The softmax activation allowed the model to better handle the two-class problem in this specific context, ensuring a more balanced learning process.

Furthermore, the decision to gradually increase the number of convolutional filters across layers allowed the network to capture lower-level features initially and progressively learn more abstract and complex patterns deeper in the network.

The model was trained over 30 epochs; however, training often stopped earlier due to the early stopping criterion, which monitored validation performance. This ensured that the model avoided overfitting and generalized well to unseen data.

These architectural and training adjustments underscore the importance of iterative experimentation and optimization when designing convolutional neural networks for domain-specific challenges such as fingerprint classification.

6 Evaluating the Results

6.1 Model Training Dynamics

Figure 4 shows the evolution of the training and validation accuracy, as well as their loss over the epochs. The learning rate adjustments, managed by the ReduceLROnPlateau callback, are also depicted in the third plot. These metrics provide a clear overview of the model's convergence behavior and its generalization ability during training.

The training accuracy steadily improves, reaching 64.56%, while validation accuracy fluctuates around 0.64. Training loss consistently decreases, indicating effective learning, whereas validation loss shows some irregularities but remains relatively stable. The learning rate plot demonstrates the model's adaptation during training, with reductions at specific epochs to fine-tune performance.

6.2 Model Predictions

The evaluation of the trained CNN model was performed using the reserved test set comprising 600 samples evenly distributed between the "male" and "female" classes.

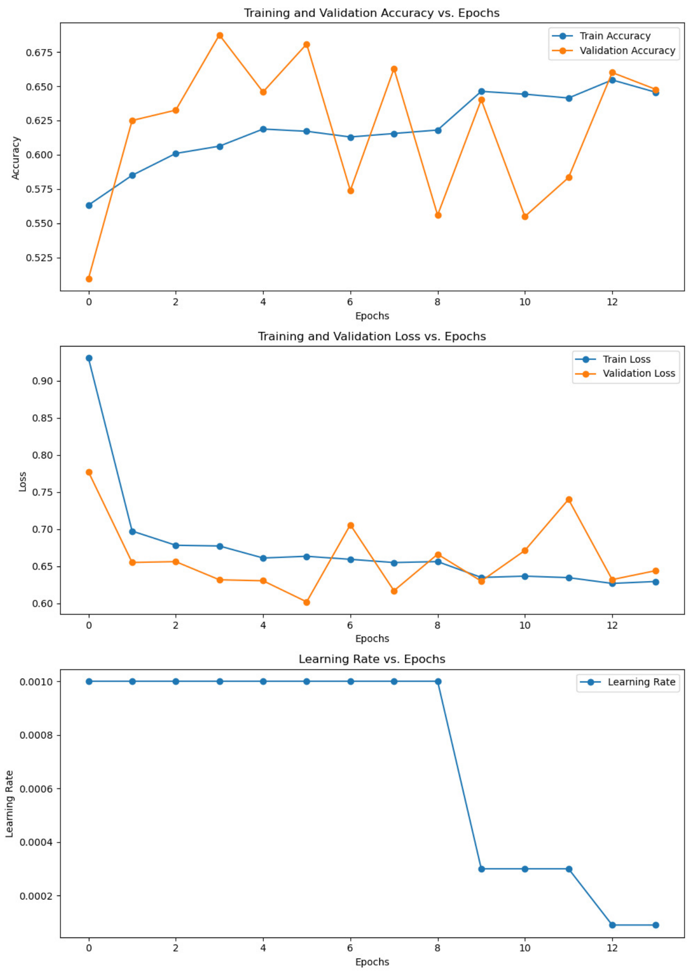


Fig. 4 Evolution of training and validation metrics over epochs

The model achieved a test accuracy of 67.33% and a test loss of 0.611, showing a notable improvement over earlier iterations.

To provide information on the performance of the model, a sample of predictions from the test data set is visualized in Figure 5. Each image is displayed alongside its corresponding true class and predicted class. Correct predictions align the true and predicted labels, while mismatches highlight errors made by the model.

6.3 Confusion Matrix

Figure 6 displays the confusion matrix, which provides insights into the model's ability to correctly classify each class. While the model performs better in identifying "male" fingerprints, it still demonstrates some challenges in distinguishing "female" fingerprints, as evident by the misclassifications.

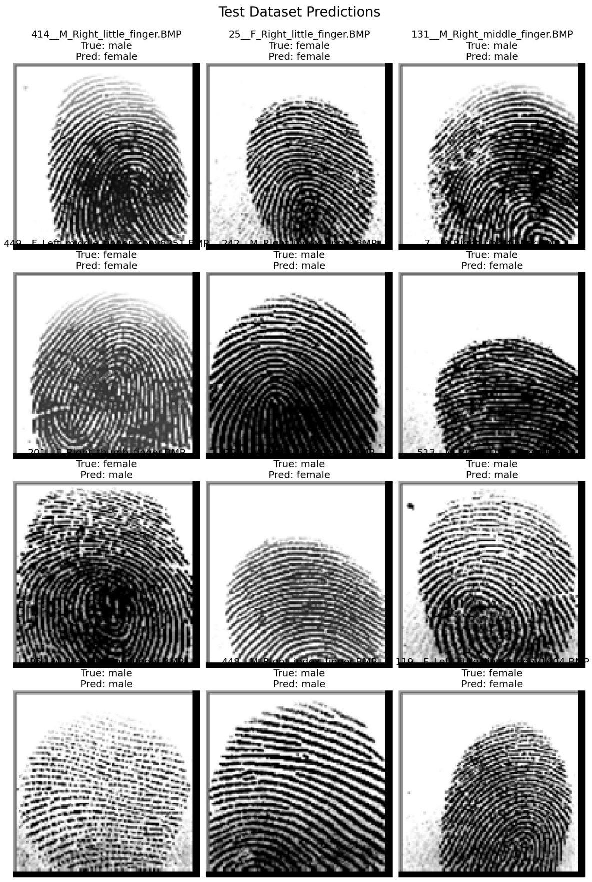


Fig. 5 Sample predictions from the test dataset

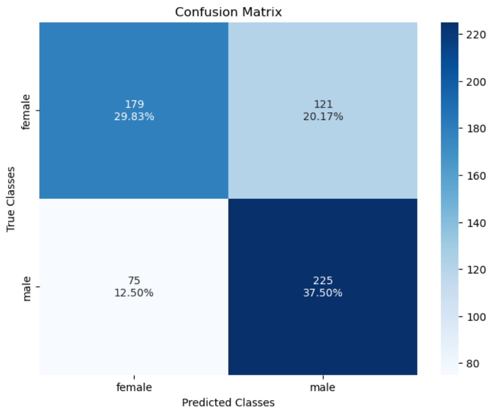


Fig. 6 Confusion Matrix for Test Dataset Predictions

6.4 Classification Report

Table 2 presents the precision, recall, and F1-score for each class. The metrics indicate that while the model shows a relatively balanced performance across classes, there is room for improvement in recall for the "female" class.

Table 2 Classification metrics for gender prediction model on the test dataset

Metric	Female	Male	Macro Avg[1]	Weighted Avg[2]
Precision[3]	0.70	0.65	0.68	0.68
Recall[4]	0.60	0.75	0.67	0.67
F1-Score[5]	0.65	0.70	0.67	0.67
Support[6]	300	300	600	600
Accuracy[7]	0.67 (600 samples)			

[1]Unweighted average across classes. Each class contributes equally.

[2]Average weighted by the number of instances in each class.

[3]Ratio of correctly predicted positive observations to total predicted positives.

[4]Ratio of correctly predicted positives to all actual positives.

[5]Harmonic mean of Precision and Recall, balancing both.

[6]Number of actual occurrences for each class in the dataset.

[7]Overall correctness of the model across all predictions.

6.5 Key Findings

Key observations derived from the evaluation include:

1. The classification of "male" fingerprints exhibited higher recall (0.75) compared to "female" fingerprints (0.60), suggesting that certain features are more distinctive in "male" fingerprints.
2. Adjustments to the model's architecture, particularly in the number of filters in convolutional layers, significantly improved performance.
3. The use of learning rate adjustment (ReduceLROnPlateau) allowed the model to stabilize during training, resulting in better convergence.

Despite these improvements, challenges in distinguishing between classes persist, highlighting the inherent complexity of the task.

7 Conclusion

The results of this study demonstrate that CNN-based architectures can offer moderate success in gender classification from fingerprint images, but also underscore the complexity of gender classification based on fingerprint images. While the final model achieved a moderate test accuracy of 67.33%, with relatively balanced performance across both classes, the results suggest significant challenges remain.

7.1 Limitations and Challenges

Key limitations affecting performance include:

1. **Dataset limitations:** Publicly available datasets for fingerprints are limited in quantity and often lack the diversity required for robust generalization.
2. **Inherent complexity of the task:** Fingerprint patterns may not exhibit sufficient distinctiveness for gender classification, especially in the absence of domain-specific preprocessing techniques.
3. **Technical constraints:** While the CNN architecture used in this study showed moderate success, deeper architectures or transfer learning methods might be needed to achieve significant improvements.

7.2 Implications and Applications

The ability to classify gender from fingerprint images has several potential applications and implications, particularly in forensic science, as well as broader scientific and judicial advancements:

1. **Forensic investigations:** Gender classification can aid investigators in narrowing down suspects in criminal investigations, particularly when fingerprints are one of the few pieces of evidence available.
2. **Identity reconstruction:** In scenarios such as natural disasters or accidents, gender classification from incomplete fingerprint evidence can support identity reconstruction efforts.
3. **Judicial applications:** Automating gender classification could provide additional tools for verifying identity claims in judicial processes, improving objectivity and reducing human error.
4. **Advancements in biometric research:** Understanding gender-based fingerprint patterns could drive further innovations in biometric technologies, expanding their use cases beyond traditional identification tasks.

The integration of AI in forensic sciences opens the door to unprecedented levels of objectivity and efficiency. For instance, AI has already been shown to detect patterns in fingerprints that human experts often overlook, such as intra-personal similarities across fingerprints of different fingers, as highlighted in recent studies. These advancements suggest the potential for AI to refine existing forensic methodologies and even uncover new patterns previously undetectable by human analysis.

8 Future Work

To address the limitations encountered in this study and further enhance the performance of gender classification from fingerprint images, several directions for future research are proposed:

1. **Dataset Expansion:** Acquiring larger, high-quality fingerprint datasets that encompass diverse populations and fingerprint patterns is essential. This will improve the model's generalization capabilities across various demographic groups.

2. **Advanced Data Augmentation:** Exploring more sophisticated data augmentation techniques, such as GAN-generated fingerprints or geometric and photometric transformations, could help increase the variability and size of the dataset.
3. **Deeper Architectures and Transfer Learning:** Implementing deeper CNN architectures or leveraging transfer learning with pre-trained models on biometric or image classification tasks might significantly improve performance by capturing more complex patterns in the data.
4. **Domain-Specific Feature Extraction:** Collaborating with forensic science experts to integrate domain knowledge into the feature extraction process. This could include identifying gender-specific minutiae or ridge patterns in fingerprints that are not easily detectable by current AI methods.
5. **Exploring Hybrid Models:** Investigating hybrid approaches that combine CNNs with other machine learning models, such as ensemble techniques or classical methods, to improve robustness and accuracy.
6. **Multi-task Learning:** Expanding the scope of the model to perform multiple tasks simultaneously, such as gender and age prediction or even identifying unique characteristics within fingerprint groups.
7. **Real-world Applications:** Testing the model in real-world forensic scenarios to evaluate its utility, reliability, and performance in practical applications.

This future work aims to overcome the current study's challenges and limitations, paving the way for more accurate, robust, and impactful AI-based solutions in biometric and forensic sciences.

Declarations

- Data availability The dataset used in this study was obtained from Kaggle.com https://www.kaggle.com/datasets/ruizgara/socofing/data
- Code availability The data and code for this project is available on https://github.com/mdiaz683/CNNFingerprints

References

Guo G, et al. (2024) Unveiling intra-person fingerprint similarity via deep contrastive learning. Science Advances 10(2):1–12. https://doi.org/10.1126/sciadv.adi0329

Hamdi S, Moussaoui A (2020) Comparative study between machine and deep learning methods for age, gender and ethnicity identification. In: 2020 4th International Symposium on Informatics and its Applications (ISIA), M'sila, Algeria, pp 1–6, https://doi.org/10.1109/ISIA51297.2020.9416549

Patel M, Singh U (2023) Age and gender recognition using deep learning technique. In: 2023 3rd International Conference on Smart Data Intelligence (ICSMDI), Trichy, India, pp 238–245, https://doi.org/10.1109/ICSMDI57622.2023.00052

Patil S (2023) Gender classification: An integrated analysis of multimodal biometric data using deep learning techniques. International Journal of

Research Publication and Reviews (IJRPR) 4(10):2855–2865. Available online: https://ijrpr.com/uploads/V4ISSUE10/IJRPR18480.pdf

Taylor M, et al. (2012) Latent print examination and human factors: Improving the practice through a systems approach. NIST Interagency/Internal Report (NISTIR) 7842, https://doi.org/10.6028/NIST.IR.7842

The New Face of Facial Recognition: Using a Red-Yellow-Blue Palette to Decrease Bias in Feature Detection

Patriana Napoleon[1] and Jerry F. Miller[2]

[1]Florida International University, Miami, FL 33199

[2]Florida Agricultural and Mechanical University, Tallahassee, FL 32307
Jerry.miller@famu.edu

Abstract. In recent years, many studies have made progress in identifying bias in facial recognition systems. However, existing works recognize that bias in facial recognition still exists, leading to false identifications, arrests, and agony for our citizens. Current systems cannot effectively detect African American men and women between the ages of 18-30. This group faces bias in different facial recognition system algorithms, likely due to certain underlying factors. Factors include the image quality of datasets and untrained or insufficiently trained algorithms to accurately detect faces in images. These factors have led investigators to introduce a new principle in understanding how race and demographic bias can be found in facial recognition algorithms. Through our research, we propose a new method that can be implemented in facial recognition algorithms during the detection phase. This paper introduces this new method to decrease unfairness during the classification phase of datasets. The fairness measure is calculated during identification of facial landmarks used to detect parts of the face with coordinates. By applying a RYB (Red, Yellow, Blue) pallet of colors when detecting areas of the face instead of the more conventional RGB (Red, Green, Blue) pallet. Our research indicates RYB colors may improve the equality of images, which can decrease bias and improve detection fairness in algorithms for captured images of African Americans ages 18-30.

1 Introduction

1.1 Origins of Facial Recognition

Facial recognition was originally developed by Woodrow Wilson Bledsoe in the 1960s. Since then, it has been incorporated into many applications, including surveillance for law enforcement and security applications. In addition to law enforcement, facial recognition is used in retail, government facilities, education, and even in our mobile devices that we use every day. As our reliance on advanced technology increases, facial recognition will continue to have an enormous impact. Many corporations and businesses rely heavily on this technology, especially when it comes to solving and identifying crimes like shoplifting. The ability to recognize someone's face with accuracy is a primary goal to accurately identify a person. Many of these systems perform very well and are robust. In today's age of facial recognition technology, there has been a spark of problems arising when detecting faces.

Many of the problems are found when the recognition system is attempting to detect African American woman between the ages of 18-30[1]. Facial recognition systems are trained using datasets which are a set of images used to train a learning model to be able to detect parts of the face features. Features like the nose and chin. Recent studies have found there are many factors that can contribute to the accuracy of face detection. A recognition system will most likely learn from any image even when the images in the dataset have bias or lack fairness in what it is learning. Today, the significant problem in facial recognition that contributes to the misidentification of African Americans between the ages of 18-30 include environmental factors, the of lack demographic ranges in datasets, image quality, and the system's ability to recognize specific features in African Americans unlike other demographics. A study on the accuracy of facial recognition comparing stars discovered shows that a KNN model has the highest recognition rate for white people (95.7%) while African Americans scored a lower accuracy of 85% [2]. Leaving the other percent's vulnerable to misidentification in this demographic. A US government study also suggested that facial recognition algorithms are far less accurate at identifying African Americans and Asian faces compared to Caucasian faces [3]. Another study conducted by institutions including Johns Hopkins University and the Georgia Institute of Technology programmed AI-trained robots to scan blocks with people's faces from different demographics [4]. After the faces were scanned, the robots were assigned to designate which blocks were criminals. The robots consistently labeled the blocks with African American faces as criminals [5], yet none of the much lighter complected subjects was identified as such. Is there bias within the facial recognition systems when it comes to certain demographics? Does this impair the systems' ability to provide accuracy for daily operations like investigating crimes, identification and surveillance of people? Apparently so.

1.2 Case Studies: Facial Recognition Misidentification

There are many reports of African Americans who are falsely accused of a crime due to the result facial recognition used by police when matching an unknown suspect photo. There has been 6 significant cases, all 6 cases consisted of African Americans [6].

In 2023, an African American mother encountered an unforgettable experience early morning. Porcha Woodruff suffered lasting trauma from being a victim of misidentification using facial recognition. She was unexpectedly arrested for a carjacking and robbery then later released from the Detroit detention center on a $100,000 personal bond. She was identified as a suspect though a photo lineup of victims along with unreliable facial recognition [7].

In a second, closely related incident in Detroit, Robert Williams was unexpectedly arrested in January 2020, spending a night in a detention center leaving his family worried for a crime he didn't commit [8]. A blurry photo image was subject to a facial recognition system and incorrectly matched a photograph of Roberts old driver's license picture. This is another case demonstrating bias and inaccuracies in facial recognition systems. The incident highlights the process of investigations when police are trying to identify suspects of a crime using a combination of facial recognition and photo lineups.

In 2020, the Detroit police chief stated that Detroit's facial recognition technology failed 96% of the time, yet it is still used in investigations [9]. It is likely that many other police departments around the world are using this technology and are leading to multiple false arrests, such as those that occurred in Louisiana, New Jersey, Maryland, and Texas. Other cities and states in the U.S., including San Francisco, California; Austin, Texas; and Portland, Oregon; have temporarily banned its use because of concerns about privacy and racial bias.

1.3 Demographic Problems in Facial Recognition

Studies have shown that a significant inaccuracy is likely in the identification phase of facial recognition. Additionally, there is bias in the classification of specific demographics. When photos are gathered to process and are fed into a learning model for the system to recognize faces, the photos are classified. For example, a chin in a photo would be classified as “chin” so the system knows the classification [10]. However, due to the bias, the results are not always clear.

In 2015, Jacky Alcine logged into Google photos and noticed that her photo album was classified as gorillas. Why did this facial recognition system software categorize Jacky and his friends as primates? During the development phase, depending on how the machine is learning different features of faces and how data is fed into the system, the recognition will output what it identifies as the most closely related elements. So, if the dataset is miscategorized, then the facial recognition system will be biased providing incorrect results. Existing datasets of photos that contain a collection of images are likely biased towards Caucasian faces. Therefore, it is highly likely that one reason behind misidentification can also include the lack of trained faces for African Americans and people of color. This may also be a result of different age ranges affecting the complexions of people of color throughout the world [10].

2 Principals of Facial Recognition

As facial recognition continues to develop, more research has been conducted to propose improved algorithms and corrections. Most Facial recognition technology is based on the extraction and comparison between faces and face features in an image. Face recognition systems will search and then match the face image feature data with a feature template stored in the database for recognition. A variable is set to a specific feature based on the data of an image. Once the variable is set, the system then finds similarities that exceed or match the variable leading to a matched result. The matched result will be provided as an output. When matching face features, the face feature is recognized and compared with the face feature variable and is judged to match a facial identity based on similarities of the face and the face variables. There are two major categories in facial recognition. One is *confirmation* which is the process of comparing one image to another to confirm an image. The second category is *identification*, this process matches one image to another while comparing the image to stored information to produce an output matching an identity [2]. This process is then divided into four pipelines. The different pipelines that make up facial recognition to produce a successful output of a compared image are *detection*, *alignment*, *representation*, and *verification* [11]. This research focuses on enhancing both the *detection* and *alignment* pipelines by reducing bias when detecting specific features of the face.

2.1 Facial Recognition Pipelines: The Process

There are four common pipelines used in facial recognition such as face detection, face alignment, feature extraction (representation) and feature matching or verification [11]. Face detection is commonly the first stage for facial recognition to have the ability to first detect faces presented from a given image and extract parts of a face if it exists. This phase crops areas of the face. So, the file can be compressed for further feature extraction. Algorithms are deployed to perform this phase.

The most common methods used for face detection are Haar Cascade, Dlib (HOG) and MTCNN (Multi-Task Cascaded Convolutional Network. Dlib (CNN) [12]. Haar Cascade is a method for image processing that classifies objects based on texture, shape and color. A color can be coded to identify an object; thus, lighting conditions can play a very important role when detecting objects [13]. The algorithm's ability to detect objects is based on the features of that object. For example, a pixel value can be calculated to represent a section of an image that identity's an area under an eye or near a cheek. Dlib is an open source, modern C++ toolkit containing machine learning algorithms and tools for creating complex software to solve real world problems. Dlib provides another face detection model based on HOG (Histograms of Oriented Gradients) for human detection using an image pyramid to extract features from image data. This method only works on frontal and on slightly shifted frontal images. Dlib Convolutional Neural Network (Dlib(CNN)) extracts features from an input image data to increase the number of features and can accommodate various face orientations. The Multi-Task Cascaded Convolutional Networks algorithm or MTCNN is another method for face detection and face alignment. MTCNN can detect faces in various conditions and scales. Each technology provides a unique perspective for facial feature detection.

The next step in the pipeline is face alignment. Images are aligned to improve the accuracy of the face recognition model. For example, on an image, eyes can be rotated and aligned with each other until both eyes are aligned horizontally to improve the accuracy of eye location. Feature extraction extracts components of the face. The components of the face consist of face features that differ from one person to another. Feature classification is the ability to compare images of the face and match it with another image to classify one's identity.

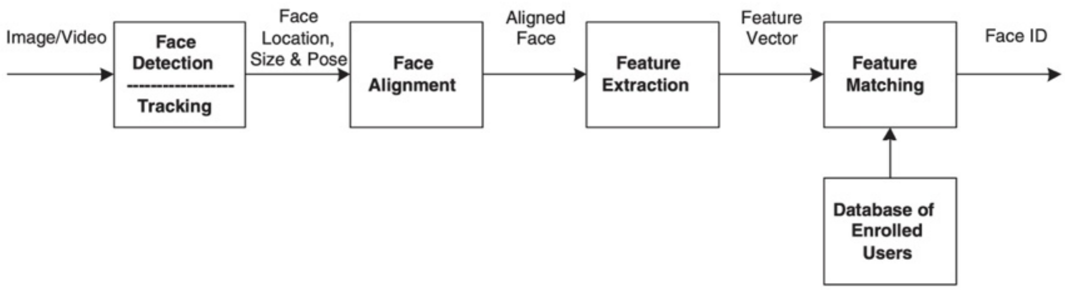


Fig. 1. Steps in Face Recognition Process as Outlined in the Handbook of Face Recognition, 2011 [14].

3 DLIB Methodology

Dlib is one of the most utilized algorithms and methods used in many facial recognition systems. Dlib has been used to enhance many recognition systems that utilize this method as a means of reducing bias in those systems to increase accuracy when detecting African Americans from different backgrounds [15]. Therefore, researchers selected the Dlib methodology for this research to improve both the detection and alignment pipelines by reducing bias when detecting specific features of the face. Specifically, this research focuses on the Dlib facial land marking algorithm which is used to detect areas of the face and then used to compare features.

Dlibs facial landmark detection identifies areas of the face by estimating the location of facial structures in an image. Dlib uses 68 land points to find areas of the face such as the nose, chin, eyes and eyebrows [16]. The Dlib facial landmarking model was trained using the iBUG 300-W-dataset which is a dataset that includes variations of different subjects, poses, illuminations, and face features [17].

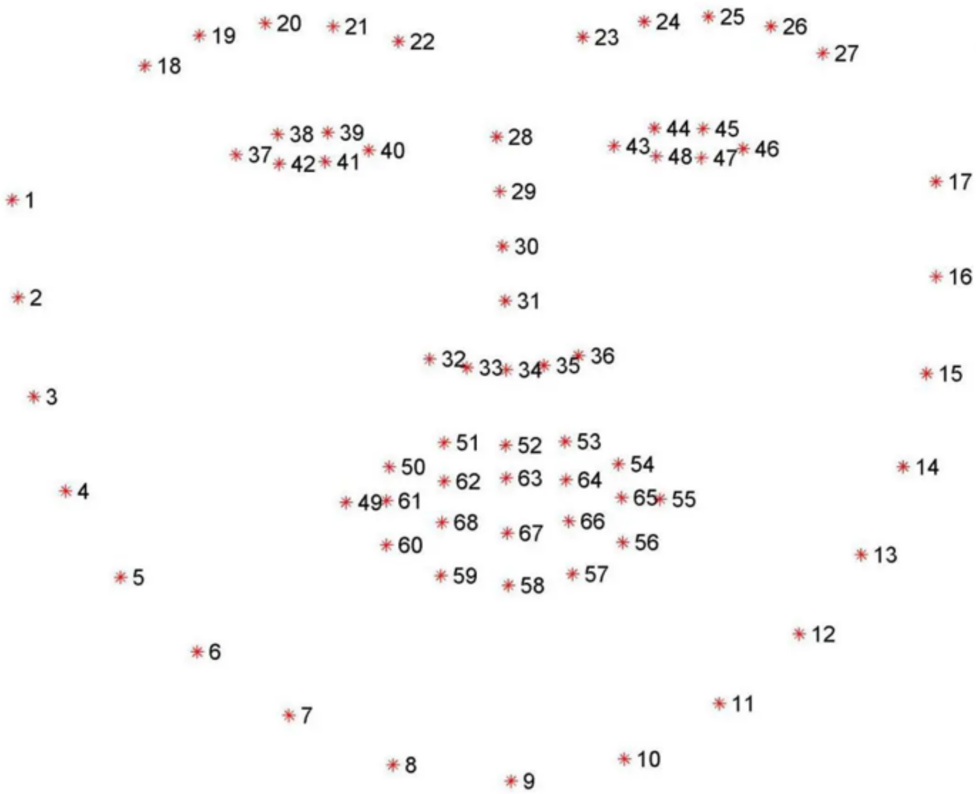


Fig. 2. Dlib Facial 68-Point Landmarking [17].

The authors in [2] conducted a study on Dlib's accuracy to detect different demographics based on television actors. 1000-star photos were used to verify the accuracy of the algorithm [2]. Out of the 1000 images, 912 images were successfully extracted. Overall feature points found on the faces had an accuracy of 91.2%. The authors suggested that the comparison between the successful and unsuccessful images in the study could have been affected due to faces being too small or having unclear or occluded areas in the image. The study then focused on the recognition rate of different skin colors. The training model was developed with the use of K-Nearest Neighbor (KNN), a supervised learning classification algorithm that determines the class of a new data point by looking at its nearest neighbors in the training data. In this case, KNN identified the closest faces in a database to a new, unknown face and assigned the new face to the label of the majority class of those faces closest. The model combined KNN and Dlib and demonstrated a recognition rate for people with lighter skin tones as 95.7%. The recognition rate for darker skin tones was lower. Researchers concluded that the reason for the higher recognition rate for lighter skin tones may have been due to the feature points used to train the KNN model to detect faces. Researchers in this paper focused on improving facial landmarking when used on different skin colors and how these landmarks could be better accentuated to detect ranges of skin colors.

4 The Red-Yellow-Blue (RYB) Experiment

4.1 The Red-Green-Blue Dilemma

Facial landmarking can likely influence how a facial recognition model is trained to then be able to identify a person based on features learned from the landmarking. Bias is likely present in the detection stage when detecting and extracting areas of the face. If the facial landmarking algorithm incorrectly places landmarks on an image, the model will be trained incorrectly. For example, if the algorithm incorrectly adds a landmark to areas of a nose when it was a cheek, the training model then learns and identifies similar areas as a nose, leading to incorrect identification of a person when an image is compared to one another. Features which are not clearly defined in the training set based upon complexion and skin color would then adversely affect the results when applied to detecting faces of different demographics.

The Dlib landmarking image processing is designed to accept images that have RGB (Red, Green, Blue) pixel colors. Each pixel color was defined with a trait [15]. The trait allows image processing to know how to handle each type of pixel. A pixel type can be RGB, as this pixel type is what is used to understand how colors are stored in memory to make up the image. RGB colors are the most used color space [18]. However, RGB color space also can be an inaccurate measurement of color because, representation of color is not consistent with the perception of human eyes. RGB colors are most likely *perceived* based on what we think the colors of red, green, and blue and combinations can represent. This results in a big difference between the calculated value of color difference and the intuitive perception of human vision. Thus, Dlib is most likely using pixel traits (RGB colors) that may not match the colors that we see through our own perception. Therefore, there will likely always be a significant gap of colors that do not exactly match what we see on the computer screen versus the real world. For example, we may see a red car in an image but there are other colors and combinations that make up red that we view on the computer screen versus a red car in real life. Hence, when areas of an image are assigned a pixel that matches a color trait like red, it is likely possible that pixel is assigned the incorrect color. This perceived difference affects the color condition of the image when representing a landmarked face which provides identity features of a facial image.

This research hypothesis that there are likely missing colors in RGB that may not be presented during pixel classification when pixel traits are assigned to an image. Computer systems use RGB colors [19]. A monitor or TV screen generates three colors of light (red, green, and blue) and the different colors we see are due to different combinations and intensities of these colors with the presence of light.

4.2 A New Approach with an Old Color Scheme

There is another classification of colors known as RYB (Red, Yellow, Blue). RYB is typically used for fine art and dates to the early 19th century when color was first defined around Isaac Newton's circular dimension hue optics for how we perceive color in 1704 [20]. Although extensively used in fine art, RYB is not a color pattern that exists in computer screens as primary colors. RYB as an historical relationship was overturned in the second half of the 19th century for the more common RGB color scheme used today. But these colors continued to be used by artists and have found their way into modern education for fine art.

The RGB scheme can be used to produce colors of white light as depicted in Figure 3. If RGB colors lead to white light, then it is likely that other areas of colors that do not exist when there is no light would not be easily . this would be RYB for computer screens.

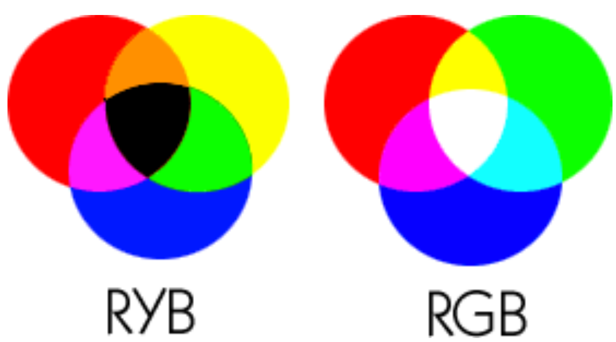


Fig 3. RGB and RYB Color Palettes

This experiment conducts converting an RGB image to RYB due to the possibility of missing colors. The missing colors highly likely contribute as an additional factor to the cause of bias in face detection algorithm that use RGB pixels to classify colors. The facial recognition system will likely fail to correctly identify certain demographics due to the lack of color range in RGB color scale. RGB colors likely have colors closer to white making the colors of an image appear to be brighter colors of red, green and blue which can falsely perceive a color of an individual incorrectly. The use of RYB colors will likely include the absence of light allowing the RGB colors to appear darker, creating different color range scales of red, green and blue. RYB (red, yellow, blue) will likely increase the color ranges in demographics by preserving more colors of an image when it's processed to classify pixel traits. Due to the nature of computer screens only using RGB, RYB can only be simulated to test this theory. RYB can be approached by converting RGB values to CMY (Cyan, Magenta, Yellow) and then CMY to RYB. Since RYB is not a color standard in digital processing, RGB can be simulated to RYB using mathematical formulas.

RGB to CMY Conversion Formula $C = 1 - R$

$M = 1 - G$ $Y = 1 - B$

CMY TO RYB

$R_{ry\,b} = R$

$Y_{ry\,b} = (Y + G)/2$ $B_{ry\,b} = (B + M)/2$ [22]

The dataset used was from a face research lab in London. 10 Images in this dataset was converted from RGB to RYB changing eh color of the image from what we normally see to appear Green and Purple on a digital screen. Ten (10) images of females and males from different demographics was then used for Dlib facial landmarking. Dlib applied 68 facial landmarking points on areas of the face such as the ears, eyes, nose, nose bridge, lips, chin, and eyebrows. This hypothesizes the initial phase of face detection will most likely detect African Americans with a higher accuracy using datasets that have colors of RYB (Red, Yellow, Blue) instead of RGB (Red, Green, Blue) when extracting face characteristic.

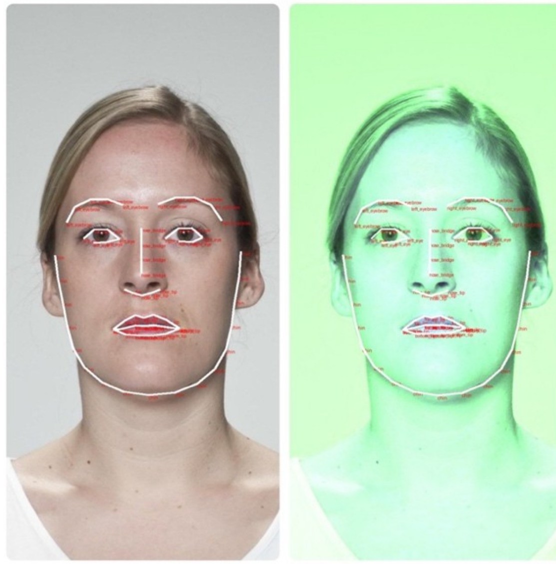


Fig 4. RGB vs RYB Identification of Landmarks

4.3 Assumptions

Datasets that contain RYB colors have a variety of darker colors that are not present in RGB. RGB algorithms consist of detecting colors that have lighter shades of red, green blue. Computer screen colors are affecting the natural colors that are found in African Americans demographics. The darker shades associated with RYB may provide more facial data when detecting the face of African Americans.

4.4 Results

To interpret the results data, 10 images of RGB and 10 images of RYB are compared to each other visually based on the facial landmarking such as the left eye, right eye, chin, nose, nose bridge, right eyebrow and left eyebrow. The features of the face were scored on a level from 0-3. 0 being negative results where the facial landmarking was incorrectly placed on the RYB image. 1 showed l low changes of facial landmarking, 2 showed moderate changes and 3 showing significant changes from the facial landmarking placed on RYB image. Both male and female images of RGB and RYB was evaluated during this experiment (Fig. 5).

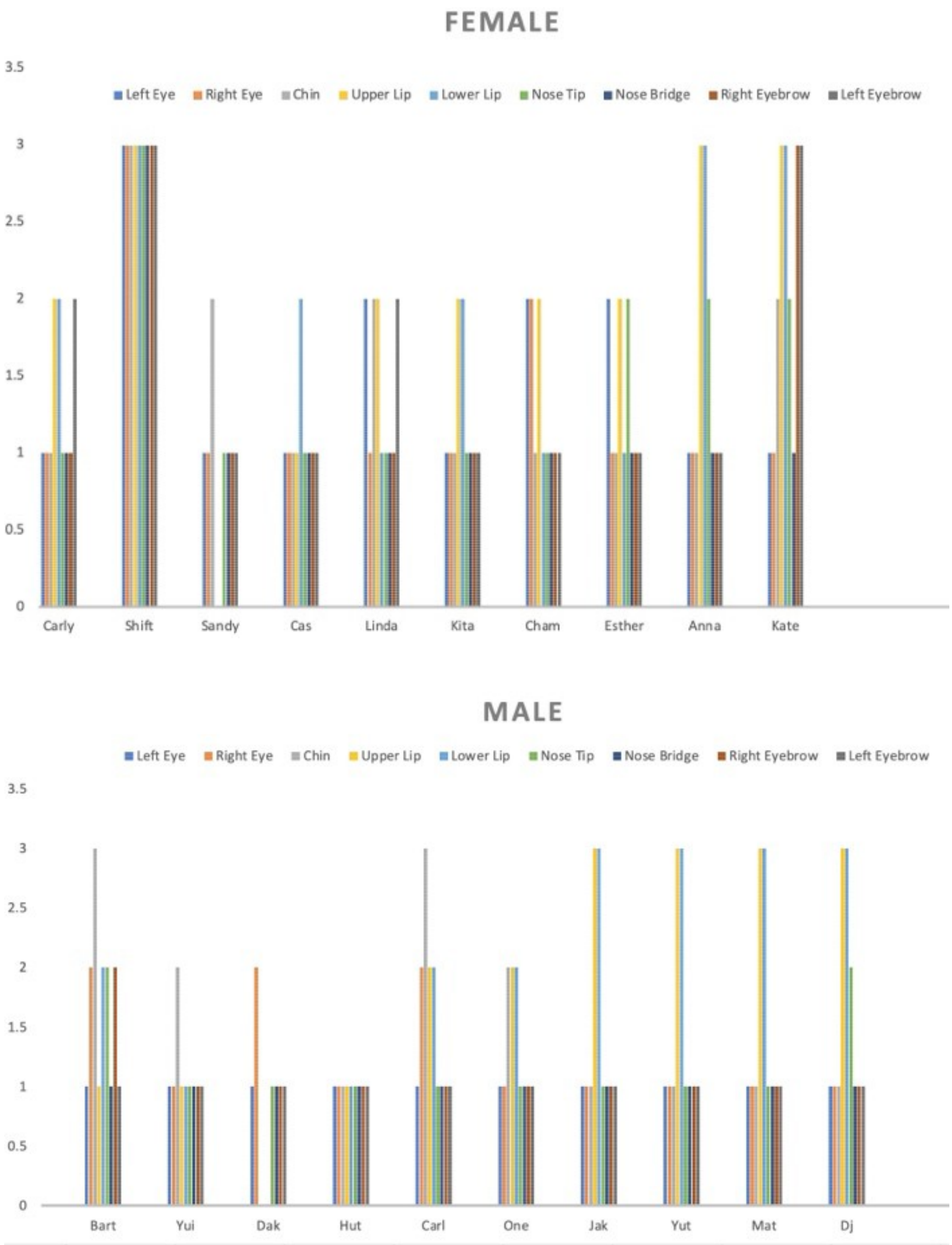


Fig 5. RGB vs RYB Results

The use of RYB images showed significant improvement on all images regardless of demographic but, there was a higher significance in images with darker colored pixels. In each feature of the face, the facial landmarking improved to a rating change of 1. Significant changes not only appeared more in the African American demographic but solely on adjustment of the landmarking in the upper and lower lip as well as the chin areas. Therefore, the use of RYB can not only significantly increase different areas of demographic but, it can highly likely reduce bias in all demographics.

Fig. 6. Male RGB to RYB and Female RGB to RYB

5 Conclusion

The red color pixels of an RYB image may contain more facial data in African Americans. The higher the facial data the higher the accuracy to detect African American faces without using bias. Computer screen colors detect more light colors when using RGB color pigments when associated with light. It is possible that African Americans skin tone reflects differently with light which changes the composition of colors associated with RYB. In conclusion, this area of RYB should be studied for future enhancements in decreasing bias in facial recognition systems as well as the colors associated with other image processing used in facial recognition algorithms.

6. Acknowledgements

This work was supported by Sophia Bohorquez Armenta, and Quang Lee.

The research was sponsored by the Army Research Office and was accomplished under Grant Number W911NF-21-1-0264. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the Army Research Office or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation herein.

References

1. A. Johnson, "Racism and AI: Here's How It's Been Criticized for Amplifying Bias," Forbes, May 25, 2023. [Online]. Available:https://www.forbes.com/sites/ariannajohnson/2023/05/25/racism-and-ai-heres-how-its-been-criticized-for-amplifying-bias/?sh=1dfa1d11269dT

2. H. Xia and C. Li, "Face Recognition and Application of Film and Television Actors Based on Dlib," *2019 12th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI)*, Suzhou, China, 2019, pp. 1-6, doi: 10.1109/CISP-BMEI48845.2019.8965869

3. "Ethical concerns raised over AI use in hiring," BBC, Dec 24, 2019. [Online]. Available:https://www.bbc.com/news/technology-50865437

4. Andrew Hundt, William Agnew, Vicky Zeng, Severin Kacianka, and Matthew Gombolay. 2022. Robots Enact Malignant Stereotypes. In 2022 ACM Con- ference on Fairness, Accountability, and Transparency (FAccT '22), June 21– 24, 2022, Seoul, Republic of Korea. ACM, New York, NY, USA, 14 pages. https://doi.org/10.1145/3531146.3533138

5. Johnson, "Racism and AI: Here's How It's Been Criticized for Amplifying Bias," Forbes, May 25, 2023. [Online]. Available: https://www.forbes.com/sites/ariannajohnson/2023/05/25/racism-and-ai-heres-how-its-been-criticized-for-amplifying-bias/?sh=324ea591269d

6. Innocence Project, "When Artificial Intelligence Gets It Wrong," [Online]. Available: https://innocenceproject.org/when-artificial-intelligence-gets-it-wrong/

7. Detroit Woman Sues City After Being Falsely Arrested While 8 Months Pregnant Due to Facial Recognition," NBC News, [Online]. Available: https://www.nbcnews.com/news/us-news/detroit-woman-sues-city-falsely-arrested-8-months-pregnant-due-facial-rcna98447

8. I Did Nothing Wrong. I Was Arrested Anyway.'," [Online]. Available: https://www.aclu.org/news/privacy-technology/i-did-nothing-wrong-i-was-arrested-anyway

9. Innocence Project, "When Artificial Intelligence Gets It Wrong," [Online]. Available: https://innocenceproject.org/when-artificial-intelligence-gets-it-wrong/

10. M. Zhang, "Google photos tags two African-Americans as gorillas through Facial Recognition Software," Forbes, https://www.forbes.com/sites/mzhang/2015/07/01/google-photos-tags-two-african-americans-as-gorillas-through-facial-recognition-software/?sh=7de63c5e713d (accessed Jan. 14, 2024).

11. S. E. Ozer, "A Gentle Introduction to Face Recognition in Deep Learning," [Online]. Available: https://sefiks.com/2020/05/01/a-gentle-introduction-to-face-recognition-in-deep-learning/

12. Backprop Labs, "Face Recognition Pipeline Clearly Explained," [Online]. Available: https://medium.com/backprop-labs/face-recognition-pipeline-clearly-explained-f57fc0082750

13. A. Adouani, W. M. Ben Henia and Z. Lachiri, "Comparison of Haar-like, HOG and LBP approaches for face detection in video sequences," 2019 16th International Multi-Conference on Systems, Signals & Devices (SSD), Istanbul, Turkey, 2019, pp. 266-271, doi: 10.1109/SSD.2019.8893214.

14. Backprop Labs, "Face Recognition Pipeline Clearly Explained," [Online].Available:https://medium.com/backprop-labs/face-recognition-pipeline-clearly-explained-f57fc0082750

15. Davis E. King, "Dlib-ml: A Machine Learning Toolkit." [Online]. Available: http://dlib.net *https://towardsdatascience.com/facial-mapping-landmarks-with-dlib-python160abcf7d672#:~:text=What%20is%20Dlib%3F,iBUG300-W%20dataset%20was%20used*

16. C. Sagonas, E. Antonakos, G, Tzimiropoulos, S. Zafeiriou, M. Pantic. 300 faces In-the-wild challenge: Database and results. Image and Vision Computing (IMAVIS), Special Issue on Facial Landmark Localisation "In-The-Wild". 2016.

17 M. Zhang, “Google photos tags two African-Americans as gorillas through Facial Recognition Software,” Forbes, https://www.forbes.com/sites/mzhang/2015/07/01/google-photos-tags-two-african-americans-as-gorillas-through-facial-recognition-software/?sh=7de63c5e713d (accessed Jan. 14, 2024).

18. Colors on a computer screen. Available at: https://www.chem.purdue.edu/gchelp/cchem//RGBColors/body_rgbcolors.html (Accessed: 04 August 2023)

19. Briggs, D. (2013) *The Dimensions of Color*, *The dimensions of colour, traditional colour wheel*. Available at: http://www.huevaluechroma.com/072.php

20. RGB and ryb color, http://www.compworks.faithweb.com/electronics/theory/light/rgbryb001.html (accessed Jan. 15, 2024).

21. DeBruine, Lisa; Jones, Benedict (2017). Face Research Lab London Set. figshare. Dataset. https://doi.org/10.6084/m9.figshare.5047666.v5

Green Nanoparticles in Forensics: Current Applications and Future Directions

Himali Upadhyay[1], Kenneth Furton[2*]

1. International Forensic Research Institute (IFRI), Florida International University, Miami, Florida, United states of America
2. Global Forensic Science and Justice Centre, Department of Chemistry and Biochemistry, Florida International University, Miami, Florida, Unites states of America

Corresponding author Email: furtonk@fiu.edu

Contributing author Email: hupadhya@fiu.edu

Abstract

The green synthesis of nanoparticles has gained significant attention as a sustainable and eco-friendly alternative to the traditional physical and chemical methods of nanoparticle production. Unlike conventional synthesis techniques, which often rely on toxic chemicals and high energy inputs, green synthesis utilizes biological agents such as plant extracts, microorganisms, and natural polymers, offering a non-toxic and environmentally benign approach. This novel method not only reduces the consumption of harmful substances but also minimizes the environmental impact associated with nanoparticle production. One of the key advantages of green synthesis is its ability to produce nanoparticles that are biocompatible, meaning they can be safely used in medical and environmental applications without posing significant risks to human health or ecosystems. In this research, the latest advancements in environmentally sustainable nanotechnology are explored, with a particular emphasis on its forensic applications. The use of biological agents, such as plant extracts and other organic materials, in nanoparticle synthesis significantly reduces the reliance on hazardous chemicals that pose long-term health risks to both humans and the environment. These biological agents not only enable the creation of nanoparticles in an ecologically responsible manner but also enhance the properties of the nanoparticles, making them more effective for specific applications. By avoiding the use of toxic chemicals, this approach aligns with the growing global demand for safer, greener technologies in various scientific disciplines.

The focus of this study is on the forensic applications of nanoparticles synthesized through environmentally friendly methods. In particular, attention is given to the use of these nanoparticles in the detection of pesticides, heavy metals, and latent fingerprints, which are critical areas of forensic investigation. Nanoparticles synthesized using green methods exhibit unique properties that make them ideal for the detection and identification of these substances, offering more efficient, sensitive, and non-invasive alternatives to traditional forensic techniques. This not only highlights the practical advantages of using eco-friendly nanomaterials in forensic science but also emphasizes the potential for these materials to play a key role in the development of analytical and diagnostic tools for law enforcement and forensic laboratories. Furthermore, the study underscores the importance of cross-disciplinary research to advance the field of forensic nanotechnology. While the green synthesis of nanoparticles holds immense potential, there is still a need for further investigation into how these nanoparticles can be scaled up, standardized, and integrated into routine forensic practice. More research is required to optimize the consistency, sensitivity, and scalability of these nanoparticles for practical forensic applications. By combining insights from various fields such as chemistry, biology, and forensic science, this research aims to foster the development of more reliable, accessible, and cost-effective nanomaterials that can be widely adopted for forensic analysis, leading to more accurate and efficient investigative processes.

Keywords: Green synthesis, Nanoparticles, Detection, Forensic Science.

Introduction

In recent years, nanotechnology has developed as a significant and transformational field of study, garnering international attention for the tremendous influence it has had and the technical advancements it has brought about across a variety of scientific and technological fields. Because of its multidisciplinary character, it has made it possible for significant progress to be made in a variety of domains, including physics, chemistry, biology, environmental science, materials science, science of medicine, and pharmacy. Nanotechnology has opened up new paths for the creation of materials and technologies that have unique optical, mechanical, chemical, and electrical capabilities [1]. Nanotechnology is described as the manipulation and control of matter such that one of its dimensions is within the range of 1 to 100 nm [2]. Nanoparticles exhibit distinct features attributable to their size, shape, composition, enhanced surface area-to-volume ratio, and the purity of their individual components. These characteristics render them suitable for utilization as nano-magnets, in medication and gene delivery systems, water disinfectants, catalysts, quantum dots for electrical devices, and as agents for pollution remediation [3]. The distinctiveness of nanoparticles arises from their specialized fabrication method. A minor modification in the synthesis pathway can result in a significant variation in their intrinsic features. Numerous techniques exist for the production of nanoparticles. The physical and chemical synthesis processes are costly and result in the production of harmful by-products. The biological technique, conversely, is economical, facile to synthesize, diminishes chemical burden on the environment, and eliminates superfluous processing during synthesis [4]. Moreover, it is widely recognized that physical and chemical approaches include uncertainties regarding the form, size, and dispersity of nanomaterials, and, crucially, need the use of expensive and dangerous substances that eventually contribute to environmental contamination. Nanoparticles originating from biological materials are referred to as biogenic nanoparticles, and the synthesis technique involved is termed green synthesis of nanoparticles. In addition to the method of synthesis, the characterisation of nanoparticles through the utilization of a variety of cutting-edge methodologies is an equally vital aspect that guarantees the nanoparticles that have been synthesized are within the nanoscale range. Furthermore, environmental factors such as temperature, pH, concentration of reducing agents, concentration of metal ions, interaction time, pressure, type of microorganism and their population, synthesis mechanism (intracellular or extracellular), and type of growth media all play a significant role in the synthesis of nanoparticles, particularly in determining their shape, size, texture, and number within the nanoparticles. The substantial advancements that have been made in the field of nanotechnology have simultaneously resulted in the release of a substantial quantity of nanoparticles into the environment (air, water, and soil), which, when exposed to live creatures for a prolonged period of time, can cause harm to them. In addition, nanoparticles, which are characterized by their nanoscale, have the potential to pose a threat due to their high surface area to volume ratio, high reactivity, and interactions with biomolecules. In order to safeguard the environment in an appropriate manner, it is thus necessary to consider the ecotoxicological impact that green nanoparticles have on species. Even though there have been tremendous advancements in this subject, the precise process that is responsible for the synthesis of green nanoparticles is not fully known. This is because of the complexities that are posed by natural quantities that are complicated [5]. Magnetic nanoparticles (NPs) have extensive uses across several domains, including medical applications (e.g., targeted drug administration, magnetic resonance imaging, cancer hyperthermia therapy) and nano-sorbents in environmental engineering [6].Forensic science is a multidisciplinary field that encompasses various specialized branches aimed at the investigation and resolution of criminal and civil cases. These branches include **forensic chemistry**, which involves the analysis of chemical substances such as drugs, explosives, and toxins; **forensic biology**, which focuses on the examination of biological evidence like blood, hair, and DNA; and **forensic physics**, which deals with the physical principles involved in analyzing evidence such as ballistics and accident reconstructions. It also includes **fingerprint analysis**, a key method for personal identification based on unique ridge patterns; **questioned document examination**, which involves the analysis of handwriting, signatures, and potentially altered or forged documents; and **digital or cyber forensics**, which investigates crimes involving computers, networks, and digital data. Additionally, the discipline covers **crime scene investigation**, where trained professionals systematically collect, preserve, and analyze physical evidence from the crime scene, and **forensic psychology**, which integrates psychological principles to understand criminal behaviour, assess mental state, and aid in criminal profiling. Together, these diverse areas work collaboratively to uncover the truth, support legal processes, and ensure justice [7]. This review concentrates on the utilization of green-synthesized nanoparticles in forensic identification, emphasizing their burgeoning role in improving the detection of essential forensic evidence, including pesticides, heavy metals, and latent fingerprints, via environmentally sustainable and non-toxic methods (Figure 1). This review is crucial for enhancing ecologically sustainable forensic techniques and protecting the health of forensic investigators by advocating alternatives that minimize exposure to harmful chemicals often employed in traditional detection methods.

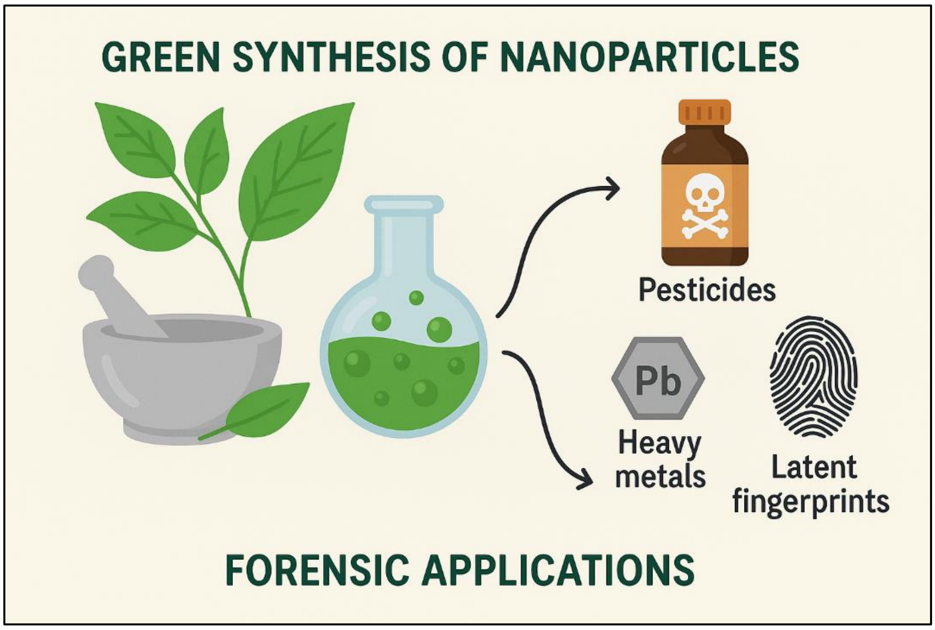


Figure 1 Graphical representation for Forensic application of green nanoparticles

Pesticide detection

The 2016 World Health Organization report indicates that toxic substances resulted in 1.3 million fatalities in 2012 [8][9]. The application of pesticides in agricultural practices directly impacts human health. Pesticides are occasionally employed in homicide cases. The identification of pesticides is therefore crucial. W. A. El-Said et al. employed two extraction methods from green tea leaves (total extraction and tannin extraction) as reducing agents for a rapid, straightforward, and one-step synthesis of mesoporous silica nanoparticles/iron oxide nanocomposite through the deposition of iron oxide onto mesoporous silica nanoparticles. Mesoporous silica nanoparticles/iron oxide nanocomposite was used as a solid adsorbent for removal of lindane pesticide from aqueous solutions. The developed system possesses the advantages of silica as core that include large surface area and advantages of iron oxide (shell) that include the capability to interact with chlorinated compounds and ability to release by using external magnetic field. UV-Visible technique was used as a simple and easy method for monitoring the removal of lindane. Capability of mesoporous silica nanoparticles/iron oxide nanocomposite for the sensing and capture of lindane molecules with a high sorption capacity (about 99%), thereby facilitating a novel eco-friendly detection approach [10]. The advancement of carbon dot-based sensors that generate fluorescence for pesticide monitoring has garnered significant interest in recent years. Compared to other fluorophores, carbon dots have superior optical characteristics, elevated quantum yields, and enhanced biocompatibility.

Fatemeh et al. intend to introduce an innovative fluorescence sensing technique for diazinon, glyphosate, and amicarbazone utilizing plant-derived carbon dots. The fluorescence-emitting carbon dots were manufactured by a hydrothermal carbonization method utilizing pesticide-free cauliflower juice as the carbon source. The fluorescence quenching capability of carbon dots has been employed to ascertain detection limits of 0.25, 0.5, and 2 ng ml^{-1} for diazinon, amicarbazone, and glyphosate, respectively. A comprehensive sample investigation shown that the detection of pesticides using our developed nano-sensor is both repeatable and precise [11]. A novel method for a straightforward, competitive, and sensitive dipstick immunoassay (DIA) was established to identify organophosphorus pesticides. This is an alternate approach to nitrocellulose strips (NC) for detecting hazardous pesticides. The matrix effects of several food samples (tomato, cucumber, grapes, and orange) were examined using flow-through ELISA (Enzyme-Linked Immunoassay) and the DIA technique. Various morphological sizes were produced from gold nanoparticles and bioconjugate to IgG, which acted as the detecting agent, as validated by AFM. The varying concentrations of organophosphorus in the food sample were identified by the formation of a purple hue on the membrane strips. Among the four fruit juices, grape juice has heightened absorbance, indicating that the bioconjugated gold nanoparticles effectively detect the kitazine contained in the grape juice, as proven by the ELISA technique, which shows high repeatability and adequate accuracy. The results were interpreted visually, and these ELISAs serve as efficient qualitative instruments for the quick assessment of kitazine residues in food samples [12].

Heavy metal detection

The substantial expansion of industry, transportation, and agriculture facilitates the influx of harmful byproducts, such as heavy metals, into the environment. The introduction of these toxic heavy metals into the food chain via water, soil, or air would considerably result in hazardous repercussions for the ecosystem, even at trace levels [13]. Nesma et al. represents a ground breaking initiative in employing Vachellia tortilis subsp. raddiana (Savi) Kyal. & Boatwr. (Often referred to as acacia raddiana) leaves as a reducing and stabilizing agent in the environmentally sustainable production of silver nanoparticles (AgNPs). The study sought to enhance the synthesis of AgNPs by examining the effects of pH, temperature, extract volume, and contact time on the reaction rate and the shape of the resultant AgNPs, while also exploring the applicability of AgNPs in the detection of certain heavy metals. The study also investigated the capacity of AgNPs to detect several heavy metal ions calorimetrically, including $Hg^{2+,}$ Cu^{2+}, Pb^{2+}, and Co^{2+}. UV–Visible spectroscopy demonstrated efficacy for this objective. The hue of AgNPs transitions from brownish yellow to pale yellow, colourless, light red, and reddish yellow upon the detection of Cu^{2+}, Hg^{2+}, Co^{2+}, and Pb^{2+} ions, respectively. The detection limits are 1.322×10^{-5} M for Hg^{2+}, 1.37×10^{-7} M for Cu^{2+}, 1.63×10^{-5} M for Pb^{2+}, and 1.34×10^{-4} M for Co^{2+}, respectively [14]. To identify heavy metal contamination in aquatic environments, the author has illustrated a highly sensitive and simultaneous detection of Cd(II), Pb(II), Cu(II), and Hg(II) ions utilizing a reduced graphene oxide-supported spongy gold nanoparticles (rGO-Au NPs) modified electrode through square wave anodic stripping voltammetry (SWASV). The rGO-Au nanoparticles were synthesized using a green method that included Abelmoschus esculentus vegetable extract as a reducing agent. The rGO-Au NPs modified electrode demonstrates exceptional selectivity and sensitivity towards heavy metal ions, with sensitivities of 19.05 $mA/mM{\cdot}cm^2$, 47.76 $mA/mM{\cdot}cm^2$, 22.10 $mA/mM{\cdot}cm^2$, and 29.28 $mA/mM{\cdot}cm^2$, and limits of detection (LOD) of 31.81 nM, 12.69 nM, 27.42 nM, and 20.70 nM for Cd^{2+}, Pb^{2+}, Cu^{2+}, and Hg^{2+}, respectively [15]. Partha et al. produce silver nanoparticles (AgNPs) with aqueous Murraya Koenigii leaf extract via a straightforward, eco-friendly method. The colorimetric sensing capability of the synthesized nanoparticles was evaluated against eleven distinct metal ions (Cr3+, Mg2+, Fe2+, Co2+, Zn2+, Cd2+, Hg2+, Cs+, Cu2+, Pb2+, and Fe3+) and established that the nanoparticles effectively detect Fe3+ and toxic Hg2+ ions in water, as corroborated by a visual color change and UV-Visible spectroscopic analysis [16].

Application Area	Type of Nanoparticles Used	Biological Source (Green Synthesis)	Detection Method / Mechanism	Advantages
Pesticide Detection	Silver, Gold, Zinc oxide (AgNPs, AuNPs, ZnONPs)	Plant extracts (e.g., neem, green tea, aloe vera)	Colorimetric sensors, fluorescence quenching, enzyme inhibition assays	High sensitivity, eco-friendly, rapid detection
Heavy Metal Detection	Iron oxide, Titanium dioxide, Silver (Fe_3O_4, TiO_2)	Algae, bacteria, fruit peels (e.g., banana, orange)	Surface plasmon resonance, electrochemical sensing	Low cost, selective binding, minimal sample prep
Latent Fingerprint Detection	Gold, Copper, Carbon dots, Silica NPs	Plant latex, leaf extracts, microorganisms	Fluorescence imaging, powder dusting with nano-powders	Enhanced ridge detail, non-toxic, high resolution

Figure 2 Nanoparticles in Forensic application

Fingerprint detection

Fingerprints are the configurations created by the papillary ridges on the distal phalanges of the fingers [17]. Primarily, there are three categories of fingerprints identified at crime scenes: patent, plastic, and latent prints [18]. Patent fingerprints are discernible to the naked eye without technology augmentation; they are created when a foreign material from the skin contacts a smooth surface. Plastic fingerprints are distinct impressions that are readily visible, typically found on freshly painted surfaces or materials that deform upon touch with fingertips. Latent prints are imperceptible to the naked eye without the aid of instruments; when a finger contacts a surface, it deposits perspiration, which stays undetectable due to its colourless nature [18], [19]. Upon the attachment of the bi-functionalized reagent to gold nanoparticles, the invisible prints become visible when it contacts a bearing surface. Gold nanoparticles treated with bi-functionalized reagents exhibit twofold functionality, therefore being referred to as bi-functionalized gold nanoparticles. The bi-functionalized reagent consists of a polar head connected to the cellulose of the active end/tail, which contains the sulphur group; the sulphur group serves as the stabilizer [20][21]. Bifunctional reagents are reagents that possess dual functionality. The polar head of the reagent interacts with the gold nanoparticle attached to the fingerprint-bearing surface. This method is mostly utilized on porous surfaces. Gold nanoparticles functionalized with antibodies are created by depositing protein A onto their surface. Protein A is a molecular connector that may be utilized to target and serve as a biological linker for the attachment and orientation of anti-cotinine antibodies on a surface. Drug metabolites in latent finger-marks were identified using gold nanoparticles functionalized with anti-cotinine. The anti-cotinine antibody attaches to the cotinine surface, which is a metabolite of nicotine. The functionality stage improves the particular interaction between antibodies and antigens [22][23][24]. The green synthesis of nano rust is conducted utilizing Camellia sinensis leaf extract. Deionized water serves as the solvent. This technique may create unique ridges on both porous and non-porous surfaces [25][26][27].

Conclusion and Futuristic aspects

This review underscores the significant potential of green synthesis of nanoparticles as a sustainable, efficient, and environmentally responsible alternative to conventional chemical and physical methods. By eliminating the use of toxic reagents and minimizing hazardous waste, green synthesis not only aligns with global sustainability goals but also enhances the safety and operational efficiency of forensic investigations. These eco-friendly nanoparticles exhibit exceptional selectivity and sensitivity, which are critical for trace-level detection of forensic analytes. In this study, we highlighted the application of green-synthesized nanoparticles in detecting pesticides, heavy metals, and latent fingerprints—three major pillars in forensic science. These nanoparticles have demonstrated remarkable capabilities in terms of ultra-low detection limits, rapid analysis, and compatibility with various substrates and detection platforms.

Looking ahead, green nanotechnology holds immense untapped potential in broader forensic domains. Future research should explore its applicability in the detection of explosives, illicit drugs, alcohols, and human biological fluids such as blood, saliva, semen, and sweat. These substances play a pivotal role in crime scene analysis, toxicological assessments, and post-mortem investigations. The integration of green-synthesized nanoparticles into these areas could lead to the development of next-generation forensic tools that are not only more accurate but also safer for users and less harmful to the environment.

Furthermore, to fully harness the benefits of green nanotechnology in forensic science, it is essential to address current limitations. Key challenges include standardizing synthesis protocols, ensuring batch-to-batch reproducibility, improving the scalability of nanoparticle production, and enhancing the stability and shelf life of nanomaterials. The convergence of green nanotechnology with modern advancements—such as microfluidics, lab-on-a-chip devices, portable biosensors, and artificial intelligence-driven data analytics—could pave the way for real-time, on-site forensic diagnostics. This integration would significantly reduce the time between evidence collection and analysis, streamline forensic workflows, and increase the reliability of results in judicial proceedings.

In conclusion, green-synthesized nanoparticles represent a transformative frontier in forensic science. With continued innovation, interdisciplinary collaboration, and regulatory support, these sustainable nanomaterials could revolutionize forensic detection methodologies, fostering a future where forensic practices are not only more effective and precise but also safer and environmentally conscious.

Funding

Not applicable

Conflict of interest

Not applicable

Ethics approval and consent to participate

Not applicable

Consent for Publication

Not applicable

Data availability

Not applicable

Materials availability

Not applicable

Code availability

Not applicable

Author contribution

Himali Upadhyay: Searched the literature, Writing original draft

Kenneth Furton: Conceptualization, Examined the literature

References:

[1] J. Nicolas, S. Mura, D. Brambilla, N. Mackiewicz, and P. Couvreur, "Design, functionalization strategies and biomedical applications of targeted biodegradable/biocompatible polymer-based nanocarriers for drug delivery," *Chem Soc Rev*, vol. 42, no. 3, pp. 1147–1235, Jan. 2013, doi: 10.1039/c2cs35265f.

[2] S. Ahmed, Annu, S. A. Chaudhry, and S. Ikram, "A review on biogenic synthesis of ZnO nanoparticles using plant extracts and microbes: A prospect towards green chemistry," *J Photochem Photobiol B*, vol. 166, pp. 272–284, Jan. 2017, doi: 10.1016/J.JPHOTOBIOL.2016.12.011.

[3] J. M. Jacob, S. Sharma, and R. M. Balakrishnan, "Exploring the fungal protein cadre in the biosynthesis of PbSe quantum dots," *J Hazard Mater*, vol. 324, pp. 54–61, Feb. 2017, doi: 10.1016/J.JHAZMAT.2015.12.056.

[4] I. Khan, K. Saeed, and I. Khan, "Nanoparticles: Properties, applications and toxicities," *Arabian Journal of Chemistry*, vol. 12, no. 7, pp. 908–931, Nov. 2019, doi: 10.1016/J.ARABJC.2017.05.011.

[5] A. Rana, K. Yadav, and S. Jagadevan, "A comprehensive review on green synthesis of nature-inspired metal nanoparticles: Mechanism, application and toxicity," Nov. 01, 2020, *Elsevier Ltd*. doi: 10.1016/j.jclepro.2020.122880.

[6] M. Johannsen *et al.*, "Morbidity and quality of life during thermotherapy using magnetic nanoparticles in locally recurrent prostate cancer: Results of a prospective phase I trial," *International Journal of Hyperthermia*, vol. 23, no. 3, pp. 315–323, 2007, doi: 10.1080/02656730601175479.

[7] P. K. Rao, M. Tharmavaram, and G. Pandey, “Conventional Technologies in Forensic Science,” *Technology in Forensic Science: Sampling, Analysis, Data and Regulations*, pp. 17–34, Aug. 2020, doi: 10.1002/9783527827688.CH2.

[8] H. Upadhyay, U. Harikrishnan, D. Bhatt, N. Dhadnekar, K. Kumar, and M. Panchal, “Calixarene: The Dawn of a New Era in Forensic Chemistry,” *Curr Org Chem*, vol. 26, no. 22, pp. 2005–2015, 2023, doi: 10.2174/1385272827666230118094847.

[9] H. Upadhyay *et al.*, “A Highly Selective Pyrene Appended Oxacalixarene Receptor for MNA and 4-NP Detection: an Experimental and Computational Study,” *J Fluoresc*, 2023, doi: 10.1007/s10895-023-03470-2.

[10] W. A. El-Said, D. M. Fouad, M. H. Ali, and M. A. El-Gahami, “Green synthesis of magnetic mesoporous silica nanocomposite and its adsorptive performance against organochlorine pesticides,” *International Journal of Environmental Science and Technology*, vol. 15, no. 8, pp. 1731–1744, Aug. 2018, doi: 10.1007/s13762-017-1530-9.

[11] F. A. Tafreshi, Z. Fatahi, S. F. Ghasemi, A. Taherian, and N. Esfandiari, “Ultrasensitive fluorescent detection of pesticides in real sample by using green carbon dots,” *PLoS One*, vol. 15, no. 3, 2020, doi: 10.1371/journal.pone.0230646.

[12] C. Malarkodi, S. Rajeshkumar, and G. Annadurai, “Detection of environmentally hazardous pesticide in fruit and vegetable samples using gold nanoparticles,” *Food Control*, vol. 80, pp. 11–18, Oct. 2017, doi: 10.1016/j.foodcont.2017.04.023.

[13] G. Aragay, J. Pons, and A. Merkoçi, “Recent trends in macro-, micro-, and nanomaterial-based tools and strategies for heavy-metal detection,” *Chem Rev*, vol. 111, no. 5, pp. 3433–3458, May 2011, doi: 10.1021/CR100383R/ASSET/IMAGES/MEDIUM/CR-2010-00383R_0023.GIF.

[14] N. H. Ibrahim, G. M. Taha, N. S. A. Hagaggi, and M. A. Moghazy, "Green synthesis of silver nanoparticles and its environmental sensor ability to some heavy metals," *BMC Chem*, vol. 18, no. 1, Dec. 2024, doi: 10.1186/s13065-023-01105-y.

[15] P. Gnanaprakasam, S. E. Jeena, D. Premnath, and T. Selvaraju, "Simple and Robust Green Synthesis of Au NPs on Reduced Graphene Oxide for the Simultaneous Detection of Toxic Heavy Metal Ions and Bioremediation Using Bacterium as the Scavenger," *Electroanalysis*, vol. 28, no. 8, pp. 1885–1893, Aug. 2016, doi: 10.1002/elan.201600002.

[16] P. Pratim Sarma, K. Barman, and P. K. Baruah, "Green synthesis of silver nanoparticles using Murraya koenigii leaf extract with efficient catalytic, antimicrobial, and sensing properties towards heavy metal ions," *Inorg Chem Commun*, vol. 152, Jun. 2023, doi: 10.1016/j.inoche.2023.110676.

[17] L. P. Angeles, "Untag Me: Why Federal Judges Are Broadly Construing Illinois's Biometric Privacy Law," *Cardozo Law Rev*, vol. 42, 2020, Accessed: Apr. 17, 2025. [Online]. Available: https://heinonline.org/HOL/Page?handle=hein.journals/cdozo42&id=359&div=&collection=

[18] V. Nagar *et al.*, "Latent friction ridge analysis of developed fingerprints after treatment with various liquid materials on porous surface," *Mater Today Proc*, vol. 69, pp. 1532–1539, Jan. 2022, doi: 10.1016/J.MATPR.2022.04.619.

[19] V. Prasad, L. Prasad, S. Lukose, and P. Agarwal, "Latent fingerprint development by using silver nanoparticles and silver nitrate—A comparative study," *J Forensic Sci*, vol. 66, no. 3, pp. 1065–1074, May 2021, doi: 10.1111/1556-4029.14664.

[20] N. Jaber, A. Lesniewski, H. Gabizon, S. Shenawi, D. Mandler, and J. Almog, "Visualization of Latent Fingermarks by Nanotechnology: Reversed Development on Paper-A Remedy to the Variation in Sweat Composition**", doi: 10.1002/ange.201205259.

[21] A. Becue, C. Champod, and P. Margot, “Use of gold nanoparticles as molecular intermediates for the detection of fingermarks,” *Forensic Sci Int*, vol. 168, no. 2–3, pp. 169–176, May 2007, doi: 10.1016/J.FORSCIINT.2006.07.014.

[22] M. J. Choi, A. M. McDonagh, P. Maynard, and C. Roux, “Metal-containing nanoparticles and nano-structured particles in fingermark detection,” *Forensic Sci Int*, vol. 179, no. 2–3, pp. 87–97, Aug. 2008, doi: 10.1016/J.FORSCIINT.2008.04.027.

[23] M. Wood, “A novel approach to latent fingermark detection using aptamer-based reagents,” 2014.

[24] A. A. L. Ahmad, A. H. Alawadhi, J. Park, H. E. Abdou, and A. A. Mohamed, “Evaluation of diazonium gold(III) salts in forensic chemistry: Latent fingerprint development on metal surfaces,” *Forensic Chemistry*, vol. 13, p. 100144, May 2019, doi: 10.1016/J.FORC.2019.100144.

[25] M. Reto, M. E. Figueira, H. M. Filipe, and C. M. M. Almeida, “Chemical composition of green tea (Camellia sinensis) infusions commercialized in Portugal,” *Plant Foods for Human Nutrition*, vol. 62, no. 4, pp. 139–144, Dec. 2007, doi: 10.1007/S11130-007-0054-8/TABLES/3.

[26] M. Baranowska, K. Suliborska, W. Chrzanowski, B. Kusznierewicz, J. Namieśnik, and A. Bartoszek, “The relationship between standard reduction potentials of catechins and biological activities involved in redox control,” *Redox Biol*, vol. 17, pp. 355–366, Jul. 2018, doi: 10.1016/J.REDOX.2018.05.005.

[27] “Unveiling the Power of Nanoparticles: A Comprehensive Review on Revolutionary Techniques for Development of Latent Fingerprints,” *Letters in Applied NanoBioScience*, vol. 13, no. 3, p. 110, Sep. 2024, doi: 10.33263/LIANBS133.110.

Securing Medical Images in Cognitive IoT Networks with SCSOA based SIMON Encryption on Hyperledger Blockchain

Priyan Malarvizhi Kumar [1] and Bharat S. Rawal[2]

[1] University of North Texas Denton, Texas, USA

[2] Grambling State University, Grambling, Louisiana, USA
rawalb@gram.edu

Abstract. The growth of electronic health records (EHRs) holding essential medical images has prompted a demand for strong data protection and quicker diagnostic procedures in the fast changing world of healthcare. In response, this research presents a unique, security-based cognitive IoT network that uses secure medical data management architecture driven by Hyperledger blockchain (HB) technology. This ground-breaking paradigm integrates a number of crucial processes, such as encryption, optimal key generation, and safe data management, which are all coordinated to guarantee the highest levels of data privacy and access control. Using Sand Cat Swarm Optimization Algorithm (SCSOA) for the optimal key generation, this paper apply the SIMON blockchiper approach inside this framework, significantly increasing its effectiveness. This method allows users to exert fine control over data access while granting hospital administrators read and write capabilities and, if necessary, causing emergency alerts. Critically, the safe preservation of patient visit data and the creation of necessary linkages for EHRs including images that are kept in other databases are made possible by the seamless exchange of medical data over a multi-channel HB. In the context of cognitive IoT networks, this paradigm offers a ground-breaking method for boosting data security. The Structural Similarity Index (SSIM) score of 0.0030 and the Feature Similarity Index (FSIM) score of 0.4371 show that the suggested model has accomplished outstanding achievements. These results represent a significant improvement demonstrating the effectiveness of the suggested paradigm.

Keywords: SIMON, Blockchiper, Sand Cat Swarm Optimization Algorithm, Electronic Health Records, Hyperledger Blockchain, Internet of Things.

1 Introduction

The relationship between the fields of medical imaging and cognitive IoT networks has emerged as a crucial area of innovation in an age characterized by the expanding deployment of the Internet of Things (IoT) and the pressing need for safe data handling [1]. The urgent need for improved medical image security in healthcare systems is addressed by this convergence. By combining cognitive IoT with medical imaging, cutting-edge technologies like machine learning, artificial intelligence, and blockchain are used to build a strong ecosystem that can protect private patient data and pictures. In addition to guaranteeing the security and integrity of medical pictures, this integrated architecture also maximizes their accessibility for rapid and precise diagnosis and treatment. Real-time monitoring, secure transmission, and intelligent processing of medical images are crucial components of this paradigm shift that enhances patient care while bolstering data privacy and integrity.

Medical imaging is a crucial pillar in the diagnosis, care, and monitoring of a wide range of medical disorders in today's modern healthcare environment [3]. The development of digital imaging technology has completely changed the medical industry, allowing medical practitioners to treat patients more precisely and effectively. The sensitive medical image data that is a crucial component of healthcare operations must be protected, which means that this digital transformation has also brought about a fundamental requirement for strong security measures [4]. In this setting, the importance of patient privacy, data integrity, and adherence to healthcare standards cannot be overstated, making security a top priority for the healthcare industry. Encryption, a method that uses mathematical and logical methods to transform data, including photos, into unintelligible forms, is one of the primary tools in protecting medical image data [5]. Encryption protects data and information from unauthorized access, guaranteeing that private and intact sensitive medical photos are transmitted and stored. In this thorough investigation, it dig into the complex function of encryption in the field of medical imaging, illuminating its significance, methods, and ramifications for both patients and healthcare professionals [6].

Medical images, such as X-rays, MRIs, CT scans, and others, are crucial in the diagnosis and management of a wide range of illnesses. The accessibility and information sharing among healthcare practitioners are greatly facilitated by these photographs, which are normally stored and communicated in digital formats. However, because they are digital, they are also exposed to a wide range of security risks [7]. At this point, encryption acts as a strong defender of patient data. A crucial barrier that guarantees only authorized people or systems can access medical images is encryption. It works by encoding the data with a secret key, rendering it unintelligible to anyone without the key. This impenetrable layer of protection efficiently protects patient privacy and data integrity by preventing unauthorized access or modification with the images [8]. While encryption is proving to be a powerful tool in the fight against data breaches and unauthorized access, using it with medical imaging poses special difficulties [9,10]. When dealing with the vast sizes of medical photographs, conventional encryption methods, such the commonly used Advanced Encryption Standard (AES), have disadvantages. High-resolution images like those from MRIs and CT scans include enormous amounts of data, necessitating creative solutions that can effectively manage these digital assets [11]. While encryption is proving to be a powerful tool in the fight against data breaches and unauthorized access, using it with medical imaging poses special difficulties [12]. When dealing with the vast sizes of medical photographs, conventional encryption methods, such the commonly used Advanced Encryption Standard (AES), have disadvantages [13,14]. High-resolution images like those from MRIs and CT scans include enormous amounts of data, necessitating creative solutions that can effectively manage these digital assets [15].

The main contributions of this paper are;

- Creation of a specific security-based cognitive IoT network with a blockchain-driven Hyperledger architecture for safe management of medical data.
- Implement a reliable and effective encryption technique to protect various medical image modalities, such as CT, MRI, and X-rays.
- The SIMON block cipher method is improved with the use of the Sand Cat Swarm Optimization Algorithm (SCSOA) for optimum key generation.
- Establishment of a ground-breaking technique for secure data exchange of medical information through a multi-channel Hyperledger blockchain in cognitive IoT networks. achieved outstanding ratings on the Feature Similarity Index (FSIM) and Structural Similarity Index (SSIM), verifying the model's efficiency.

The remaining sections of the study are organized in the form of shadows: The relevant works are summarized in Section 2, the suggested model is briefly explained in Section 3, the results and validation analysis are shown in Section 4, and the summary and conclusion are provided in Section 5.

2 Literature Survey

Author Name	Methodology Employed	Advantages	Limitations
Lai, Q et al. [16]	It is created a two-dimensional Logistic-Gaussian hyperchaotic map (2D-LGHM) with a variety of hyper-chaoses.	It is highly resistant to attacks and data loss in a variety of medical image formats by utilizing a potent 2D-LGHM hyperchaotic map, boosting security, and maintaining data integrity.	It might ignore some security risks because it lacks a thorough investigation of potential flaws and attacks that are specific to medical picture encryption techniques.
Wang, X., & Wang, Y. [17]	Create a region-of-interest (ROI)-based medical picture encryption technique employing chaotic sequences from the Logistic-Tent chaotic system (LTS)	Enhanced security is achieved via chaotic sequences, region-of-interest-based encryption, and faster encryption for medical imaging.	Risk of losing non-ROI data, dependency on hash value, and sensitivity to LTS parameters. Complexity of ROI extraction.
Abdelfatah, R. I et al. [18]	multi-chaotic maps and adaptable DNA coding. Chaos is produced by combining Henon, Gaussian, and Logistic maps.	combines adaptive DNA coding with a new multi-chaotic map architecture to provide increased security for medical data in WBANs, ensuring strong defenses against diverse threats while maximizing computing effectiveness and data transport capabilities.	One drawback is the potential complexity and difficulties of practical application of adaptive DNA coding in genuine WBANs.
Trujillo-Toledo, D. A et al. [19]	Use statistical tests, MQTT, the Lyapunov exponent, and four chaotic maps to encrypt medical images.	High data throughputs on the desktop PC and Raspberry Pi platforms. huge MIMO techniques.	There may be compatibility problems and integration difficulties when integrating the suggested cryptosystem into current healthcare IoT systems.
John, S., & Kumar, S. N [20]	LFSR-based medical picture encryption that is proposed, tested using DICOM images, and IoT	enhances the security of medical data transported across a cloud network by using a linear feedback shift register (LFSR).	In light of emerging cyber threats, it might not offer the greatest level of protection necessary for sensitive medical data.

HAZZAZI, M. M. [21]	Bit-level encryption employs DWT, bit-plane extraction, and cubic-logistic map.	Data security and little information loss are guaranteed by strong "sandwich encryption".	Under specific attacks like noise and cropping attacks, the encryption system may show decreased resistance to content exposure.
Singh, K. N., et al. [22]	Use GDWCN-PSO to optimize random sequences; combine with the Lorenz system to encrypt images; more secure than current techniques; and robust against attacks.	Beyond state-of-the-art techniques, the EiMOL encryption algorithm provides strong security for medical pictures, ensuring data integrity in intelligent healthcare.	For healthcare devices with limited resources, the computational complexity and resource needs of the proposed EiMOL encryption technique may be relatively high.

3 Proposed methodology

Speculative cognitive IoT-cloud scenario for healthcare

Within the framework of a networked smart city setting, smart healthcare systems provide various advantages to both patients and healthcare providers. To speed up access to electronic health records, these systems depend on intelligent sensor technology and use cloud-based cognitive Internet of Things (IoT) solutions. Patients may easily update their medical information with the help of integrated smart wearables and seamless communication. In order to make wise judgements and provide patients with individualized treatments based on their particular requirements; the cognitive system analyses this data in real-time. Additionally, medical experts can remotely view and examine this uploaded data, allowing them to provide prompt aid when necessary. Patient-centric healthcare monitoring is of the utmost importance for achieving the core goals of smart healthcare, which include cost reduction, improved efficiency, improved accessibility, accurate diagnostics, decreased hospital and medical staff visits, and an overall improvement in the quality of life. Below is a more elaborate example for the suggested smart healthcare monitoring framework in a smart city setting:

For access to this cutting-edge healthcare system, residents are urged to register with a smart healthcare service provider and make use of the smart city infrastructure. This registration makes it easier for the cognitive system, medical personnel, and support staff to communicate effectively, providing remote access to a patient's medical records. The technology also keeps track of the user's position, which comes in handy during emergencies. Between patients and specialized medical specialists, such as those who specialize in illnesses like epilepsy, the smart healthcare service provider is an essential connection. Patients' psychological and physiological data are continually collected in real-time using a range of smart sensors in the healthcare industry. While the cognitive system continuously monitors the patient's current status, the sensors are in charge of recording the patient's motions, gestures, and facial expressions. A handy and transportable smart EEG sensor in the shape of a skull cap is made accessible for people with epilepsy. The person's brainwave activity is regularly recorded by an EEG gadget. The gathered data are immediately sent to the cloud for in-the-moment processing when the cognitive system detects a seizure. The information is then sent to medical professionals for in-depth study. On the basis of the nature and intensity of the seizure, medical professionals may completely assess the data and provide patients with the right recommendations. The cognitive system uses the smart healthcare provider to alert the appropriate stakeholders and medical staff in emergency scenarios. Smart ambulances with necessary medical equipment can effectively maneuver traffic to guarantee quick response. With the help of an automated traffic signal system, these ambulances are given priority passage across crowded roads. As a result, the smart city healthcare framework successfully provides residents and patients with remote, prompt, and real-time healthcare services from the convenience of their own places.

System architecture

The data flow inside our suggested architecture for smart healthcare monitoring is shown graphically in Fig 1. It demonstrates the procedure for obtaining various multimedia health-related data and EEG signals using IoT devices. These sensors are arranged in this system at various levels. We've created a Local Area Network (LAN) layer that makes use of short-range smart communication tools to make it possible for signals like EEG to be seamlessly sent from IoT devices to the hosting layer. Computers, cellphones, and other smart devices make up the hosting layer, which acts as a conduit for data transmission to the cloud through the Wide Area Network (WAN) layer. The WAN interface makes use of cutting-edge connection methods like 4G, 5G, or Wi-Fi to guarantee quick and effective data transport to the cloud.

The system performs user authentication to confirm the validity of all health-related data, including EEG, after it has reached the cloud, before the data is processed by the seizure detection system. The vast spectrum of IoT sensors, which includes a number of healthcare-related sensor devices such headgear, smartwatches, wearable sensors, and wristbands, is also shown in Fig 1. The extensive range of health-related characteristics that these intelligent sensors may measure includes heart rate, blood pressure, respiration rate, ECG, body temperature, body movement, and EEG. These gadgets may be worn by patients or easily incorporated into a variety of smart city settings, such as smart homes, healthcare institutions, workplaces, or transportation. Furthermore, these sophisticated sensor devices are capable of short-range communication among themselves. Smart communication protocols created for short- to medium-range communication and device interconnectivity serve as the foundation for the LAN interface layer. These protocols, which include technologies like Bluetooth, Z-wave, Zigbee, and LoWPAN, enable effective connection and communication between devices.

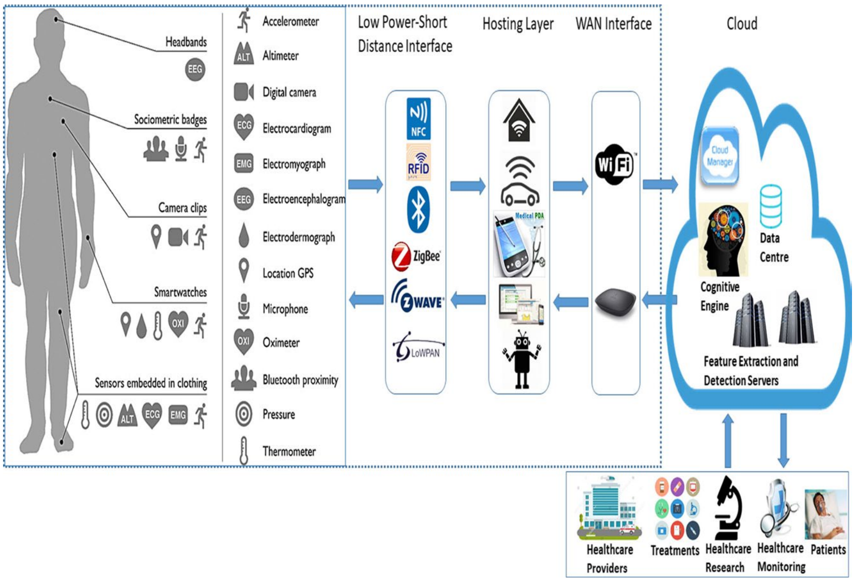


Fig 1: Framework for cognitive IoT-cloud-based smart healthcare

Laptops, workstations, personal digital assistants, cellphones, and tablets are just a few examples of the wide variety of smart devices that make up the hosting layer. Data gathering and preliminary local processing are handled by these devices. They are equipped with the processing power needed to use specialized software or programs to find irregularities in general health. These deviations include differences in measurements like blood pressure, heart rate, or body temperature. The WAN interface makes it possible to send this data to the cloud through long-distance networks like 4G, or 5G, Wi-Fi. The data is examined and patient health records are checked once it reaches the cloud-based healthcare service providers in order to help in making prompt judgements. The suggested seizure detection and categorization system also processes specialized data, such as EEG signals. Following that, the findings are presented to medical experts for in-depth analysis. Medical facility visits may be time-consuming and expensive in non-urgent situations. The adoption of such a smart healthcare framework might thus result in savings in terms of money, time, and hospital resources.

Essential parts of the cloud architecture include the cloud manager, data center, feature extraction server, detection server, and classification server. A resident's registration with a smart healthcare provider is first checked by the cloud manager during user authentication. This duty includes making sure that everyone using the system is who they claim to be, including patients, hospital representatives, healthcare workers, and doctors. Controlling data transfer to and from other servers as well as resource allocation, storage, and communication is the responsibility of the cloud manager. Effective resource management is another aspect of it. It is sent to the cognitive engine, which analyses multimodal data, including EEG, and physiological data, to determine the patient's status. The patient's motions, gestures, and facial expressions are just a few of the many things that go into this assessment. Taking into account the patient's present condition, the cognitive system decides in real-time which tasks, treatments, and services are essential. Whether or not data should be sent to the deep learning module is also decided by this. The cognitive system contacts the appropriate parties and sends the EEG data to a deep learning module when it recognizes a seizure. To extract features, deep learning methods are used in the feature extraction server. Techniques for extracting important features are used in signal processing beyond the preprocessing step. The cloud manager receives the classification and identification of seizure data from the detection server, which then reports its findings. Lastly, the data center acts as the central location for all model parameters, features, and data. The richness of data from smart sensors and the classification of EEG data are used by healthcare practitioners to make decisions about the services to be provided to patients with epilepsy. Healthcare service data are made accessible to all smart city stakeholders through intelligent communication channels after analysis of patient health reports and improvements to resident care.

4 The proposed model

The operational workflow of the proposed system is shown in a model in Fig 2, which includes various unique steps such as encryption, optimum key creation, secure data management utilising the Hyperledger blockchain, and diagnostics. The patient's medical records are encrypted in the early stage. Combining the SCSOA-based optimum key generation method with the SIMON block cipher approach, this encryption is carried out. The Hyperledger blockchain then enters the picture after encryption. A worldwide blockchain and several local blockchains customised for different medical institutions make up this blockchain architecture. In this system, the patient has the authority to give or remove access privileges to doctors or healthcare institutions. The real health records may then be accessed by authorised individuals by decrypting the ciphered data.

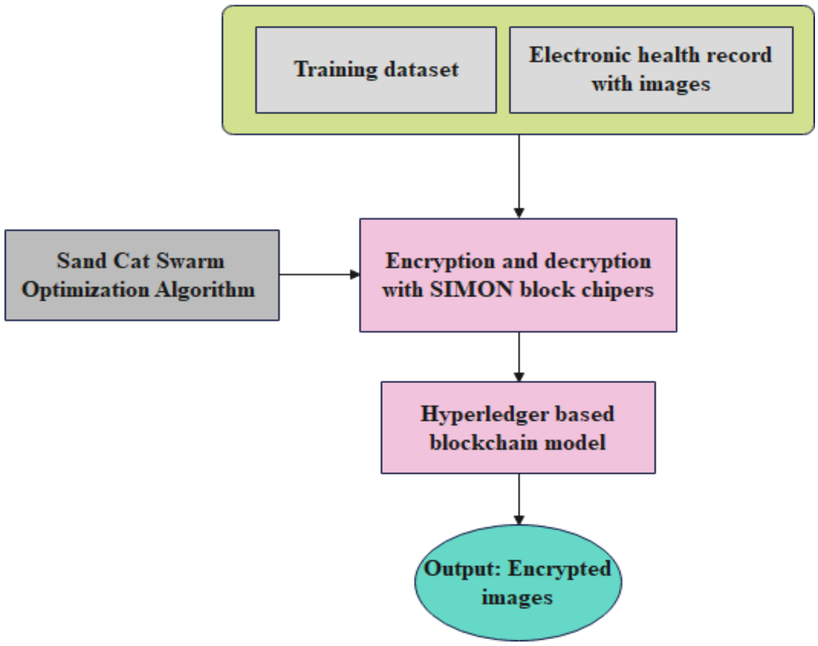


Fig 2: Workflow of the proposed model

Data encryption process

Hash functions, block ciphers, validated decryption, and encryption modelling processes are examples of lightweight cryptographic approaches in this study. The use of block ciphers for secure healthcare record keeping is the specific subject of this work. The advancements in the Advanced Encryption Standard (AES) are responsible for the development of lightweight ciphers. Several ciphers are now available, with SIMON being a well-known lightweight block cipher. These include RECTANGLE, TWINE, KATAN, SPECK, and KLEIN. In particular when it comes to hardware implementation, SIMON, a lightweight block cipher, prioritises efficiency influenced by hash functions. Each of the 10 functions of this cipher family has unique properties, principally differing key sizes and block structures. Based on manipulating picture pixels, the key changes per each block. Block sizes range from 32 to 128 bits, usually about 16. It generates ciphertext blocks using fixed-size plaintext chunks for operation. With relation to block size and data security, the SIMON cipher mixes nonlinear and linear components. One may picture a tree structure for each variation throughout all rounds, producing several possible output variants, while taking into account diverse input variations and the potential for extending critical characteristics to future rounds. The block size of the SIMON cipher is not intended to be ideal, but the selected optimisation approach seeks to get the best result in order to reduce the number of active S-boxes.

Encryption, rounds, bits, and decryption are essential elements in the context of safe data transfer using the cypher module. The critical operation of dividing an input plaintext block, normally sized at 2n, into two equally sized n-bit words is carried out by the SIMON cipher's round function. Three left shifts are performed to the left side of the block throughout this phase, along with bitwise AND logic operations. The right half block is XORed with the result of the XOR operation using the key from the current round for each round of the operations that follow. Each round's result is written back to the left block after the left half's initial value has been moved to the right block. As long as the selected configuration keeps the total number of rounds constant, this iterative round-based procedure continues. The SIMON cypher, created for use with 2n-bit blocks, may be expressed mathematically as follows:

$EncDI = Cipher_{ql}^{i}, \ldots., Cipher_{q1}^{n}, i > 1$ (1)

Round functions and round keys are used to symbolise the cipher $C_{ql}^{i} \leq i \leq q$. Similar to an iterated block cipher, this function works in a similar way. The SIMON round function is described in terms of encryption as follows:

$RF(wl, wr, kround) = \left((S^1(wl)\&S^8(wl)\oplus S^2(wl)\oplus w_r\oplus k_{round}, w_l)\right)$ (2)

According to Eq. (3), the inverse function is used to decode the data.

$RF^{-1}(wl, wr, k) = (w_r(S^1(wl))\&S^8(wl)\oplus S^2(wl)\oplus wl\oplus k_{round})$ (3)

The term in Equation (3) is defined as follows: w_l denotes the left-most word in a block supplied, w_r denotes the right-most word, and k_{round} is the suitable round key.

Proposed algorithm

Algorithm 1 is a suggested safe method of managing medical records.

Algorithm 1. Proposed SIMON block cipher addition for safe medical data management
Input: A patient P1's EHR is being read using images. Output: Add blocks to the Patient P1 blockchain and perform Patient P1 diagnostic 1: EHR with images←Patient P1 and read EHR with images 2: Public Key and Private Key← generating keys for the SIMON block cipher 3: P1 is the encryption patient←Public key 4: Doctor← insurance company and shared private key for encryption 5: Encrypted EHR with images← EHR encryption using images Using public keys, SIMON 6: SIMON← SCSOA-based hash key for EHR encryption 7: the patient's user ID, password, and patient code are used to create a Hyperledger block for the Patient P1 blockchain. 8: Block← the encrypted EHR containing photos and the hash key value. 9: safe block to the Patient P1 blockchain 10: Stop

A. Optimal key generation process

In the SIMON cypher, the master key is used throughout the key expansion procedure to create a whole set of round keys. The selected configuration, SIMON64/128, tries to convert the original 128-bit master key into 32-bit round keys. This is accomplished by a methodical procedure that combines a dependable one-bit round operation with the "key words variable," which is made up of a group of previously saved round keys. In order to generate these round keys securely, the key expansion process entails a series of related actions.

- Bitwise XOR is denoted by the symbol $a \oplus b$.
- Right circular shift, represented as F^{-j}, includes shifting by j bits to the right, whereas left circular shift requires shifting by j bits to the left, as shown by the symbol F^{j}.
- The round counter is written as $0 \leq i \leq T-1$, and the constant sequence j=0, 1, 2, 3, 4 is used to represent it.
- The number of cipher rounds, round keys (sub-keys), and constants are also essential elements.
- Right bitwise rotation, or ROR, is calculated by the formula $s^{-c}(a)$,, where c stands for the number of rotations.

Equation (4) gives an expression for the key expansion procedure.

$Key_i(k, c, z_j) = F(k_{i+3}, k_{i+1})\oplus S^{-1}(F(k_{i+3}, k_{i+1}))\oplus k_i\oplus c\oplus(z_j)_i$ (4)

An ideal key for data decryption may be selected from the large number of created keys. The use of SCSOA, which optimises its value to either minimise or maximise it, results in the selection of this ideal key.

B. Optimal key selection using Sand Cat Swarm Optimization Algorithm (SCSO)

1). Initialize Population

Every sand cat is represented as a $1 \times dim$ array in the context of the optimisation problem described in [23], where "dim" stands for the dimension of the issue at hand. This array functions as a possible answer to the issue at hand, as seen in Fig 3. Each "Pos" must adhere to lower and higher boundaries that are predetermined within a range of variable values $(Pos_1, Pos_2, \ldots, Pos_{dim})$. An initialization matrix is created to begin the procedure; it is sized to the size of the issue and produces an N-dim matrix. Every iteration also results in a related solution. The prior answer is replaced if the freshly produced one turns out to be better than the previous one. The answer from that particular iteration is not saved nor kept, however, if no better solution is found in the next iteration.

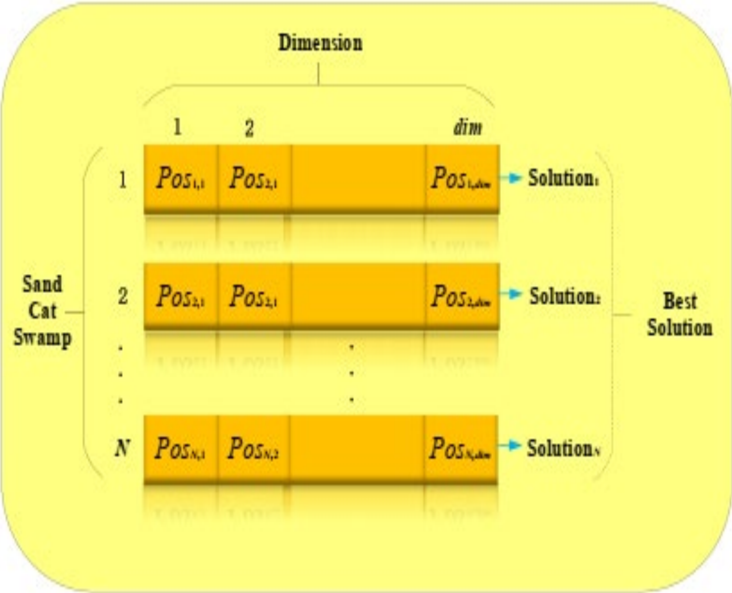


Fig 3. Population initialization diagram.

2). Search for Prey (Exploration Stage)

The SCSO algorithm uses "Posi" to represent each sand cat's location. This programme makes use of the amazing listening abilities of sand cats, especially their skill in identifying low-frequency noises. The capacity to detect frequencies below 2 kHz is shared by all sand cats. In order to define the sensitivity r_{G} mathematically, this paper use Formula (5). This sensitivity characteristic accurately reflects the sensitivity range of sand cats, which ranges from 2 kHz to 0 kHz. Additionally, this paper choose the parameter "R" in accordance with Formula (6), which is crucial for determining how well the algorithm balances its abilities for exploitation and exploration.

$r_{\mathrm{G}} = S_M - \left(\frac{S_M \times t}{T}\right)$ (5)

$R = 2 \times r_{\mathrm{G}} \times rand(0,1) - r_{\mathrm{G}}$ (6)

In this context, "t" stands for the iteration that is now being performed, "T" for the maximum iteration, and "S_M" stands in for two.

Every sand cat uses a randomised search strategy to find a new location that is still within their sensitivity range while hunting prey. This randomization encourages an algorithmic balance between exploration and exploitation. It's crucial to understand that each sand cat's sensitivity range (designated as "r") differs in order to avoid the algorithm being stuck in local optima. Formula (7) describes this variability.

$r = r_G \times rand(0,1)$ (7)

where the guiding parameter r is represented by r_G.

$Pos(t+1) = r \times (Pos_{bc}(t) - rand(0,1) \times Pos_c(t))$ (8)

Each sand cat uses a variety of variables, such as the best candidate position (Pos_{bc}),, the current position ($Pos_c(t)$), and its sensitivity range (r), to determine where the prey is. The formula given in (8) provides a clear definition of this procedure.

3). Attack Prey (Exploitation Stage)

The distance (Posrnd) between the sand cat and its prey is described by Formula (9), imitating the predatory chase made by the sand cat. Imagine the sensitivity range of the sand cat as a sphere to show this. The direction of the sand cat's movement is chosen using the Roulette Wheel selection approach. This technique chooses a chance angle "a," where "a" might be between -1 and 1. The values of "a" are next normalised to lie between 0 and 360 degrees. Therefore, as shown in Fig 4, this method enables each sand cat to move around the search region in a wide variety of circumferential directions. Following the instructions outlined in Formula (10), the sand cat next launches its prey assault. The sand cat can more quickly reach its hunting location thanks to this tactic, which increases the effectiveness of its predatory pursuit.

$pos_{rnd} = |rand(0,1) \times pos_b(t) - pos_c(t)|$ (9)

$pos(t+1) = pos_b(t) - r \times Pos_{rnd} \times \cos(a)$ (10)

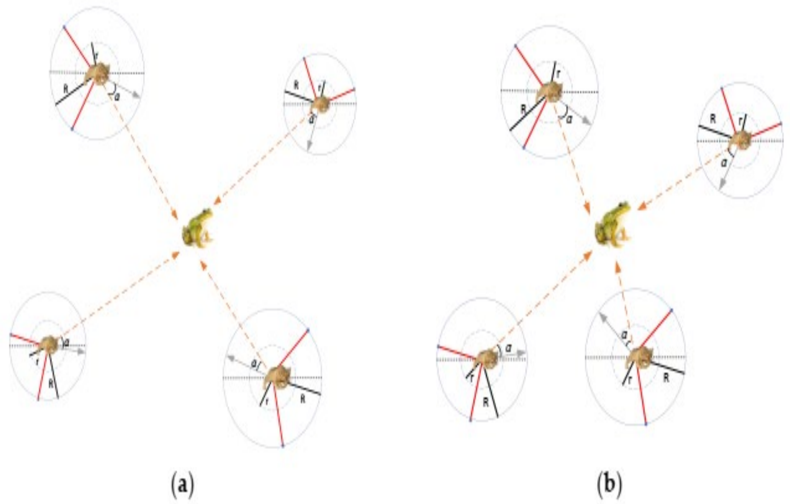


Fig 4: SCSO algorithm's location updating mechanism. (**a**) the location of the group of sand cats in iteration; (**b**) The sand cat group's location in iteration t + 1.

4). SCSO Algorithm Implementation

It employs adaptive parameters in the SCSO approach, most especially r_G and "R," to precisely adjust the ratio of exploration to exploitation. According to Formula (5), over repetitions, r_G displays a systematic linear reduction, eventually declining from 2 to 0. On the other hand, a random value is chosen for the parameter "R" from the range [-4, 4]. The sand cat starts pursuing its victim when "R" drops to or reaches 1. Conversely, the sand cat continues its hunt for food when "R" exceeds 1, as explained in Formula (11). The algorithm's exploration and exploitation stages are seamlessly transitioned thanks to this dynamic control mechanism.

$$Pos(t+1)\begin{cases} r \times \left(Pos_{bc}(t) - rand(0,1) \times Pos_c(t)\right) & |R| > 1; exploration \\ Pos_b(t) - Pos_{rnd}(t) \times \cos(a) \times r & |R| \leq 1; exploration \end{cases} \quad (11)$$

Each sand cat updates its position throughout both the exploration and exploitation periods, as shown by Formula (11). The sand cat will attack its victim if the value of "R" is less than or equal to 1. However, if " R ≤ 1, the sand cat's goal is to look for fresh prey across the whole world. For further information, please see the pseudo-code under Algorithm 2.

Algorithm 2. SCSOA Pseudo-Code

Population initiation
Using the goal function as a foundation, calculate the fitness function.
 	
Initialize the r, r_G, and R
While (t ≤ maximum iteration)
 	
For every search agent
Using the selection from the roulette wheel, determine a random angle (0◦ ≤ α ≤ 360◦)
If (abs(R) > 1)
The search agent position should be updated using Formula (8)
Else
Using Formula (10) update the location of the search agent
End
T = t + 1
End

Hyperledger blockchain (HB)

In this research, a machine learning-based illness diagnosis model is implemented using a federated learning process inside a blockchain system. A shared ledger is built using blockchain technology, which has a number of benefits, such as suitability, availability, privacy, and decentralisation. Blockchain's decentralisation means that data is duplicated across several computers, removing the possibility of a single point of failure brought on by a central server. Decentralisation improves system toughness. Even in the case of a few computer failures, availability ensures that data may be accessed when necessary. Integrity guarantees that data is safeguarded from unauthorised alterations and stays unchanged. All data saved inside the blockchain can be traced thanks to suitability. Technically speaking, a blockchain is made up of a chain of valid blocks, each having a header and data. The header is made up of a number of properties, such as a signature, an identity, and a reference to the block before it. The identifier is a value that represents each block of data and is globally unique. It is obtained from a mathematical formula. Because it establishes a logical chain of links between blocks, the reference to the preceding block is essential for preserving the blockchain's integrity.

One of the open sources blockchains offered by the management framework Hyperledger is called Hyperledger fabric. It consists of a number of parts that work together to create a decentralised environment, including endorser peers, ordering nodes, certificate authorities, clients, and committing peers. These parts connect with one another via channels, which allow for discreet and secret business dealings and demarcate various application areas. The Fabric certificate authority fulfils two functions. It does this in two ways. First, it makes sure that diverse parts, such users or smart contracts, follow the rules established for the system. Second, it verifies and gives permission to components for certain tasks like transaction execution or access to other components. Maintaining the chain connected to the channels built into the system is the responsibility of committing peers. According to the "individual chain per channel" strategy, they store several blockchains for the channels. As each channel's data is kept separate from the others, this method improves scalability and privacy. In order to gain access to a chain via a committed peer linked to a distinct channel, components require access rights. The system can grow to meet increasing demand because to the "individual chain per channel" method, which enables the administration of many transactions and data across several committing nodes. Smart contracts are represented as chain code applications on the Hyperledger Fabric platform. The network-approved business logic is implemented using chain code. Each chain code maintains a private state that is off-limits to other chains, yet it is permitted to invoke another chain code with the right authorisation. Chain code is divided into two categories: application chain code, which is in charge of maintaining application-specific states, and system chain code, which handles system-related operations. The private healthcare organisation starts a registration function inside the smart contracts (chain code) to sign up individuals with the permission of the Fabric network manager. A private permissioned network is created by the healthcare authority only for registered stakeholders. Entry to the registration system is made possible over VPN connections to provide increased security. Patients must provide all necessary details, including their name, their social security number, their home address, and phone number, in order to finish the registration process. Primary care doctors, hospitals, labs, pharmacies, researchers, and insurers all go through the same registration procedure. The public health authority checks the records after registration and issues a chain code address. Transactions on the network may start as soon as all parties have successfully registered.

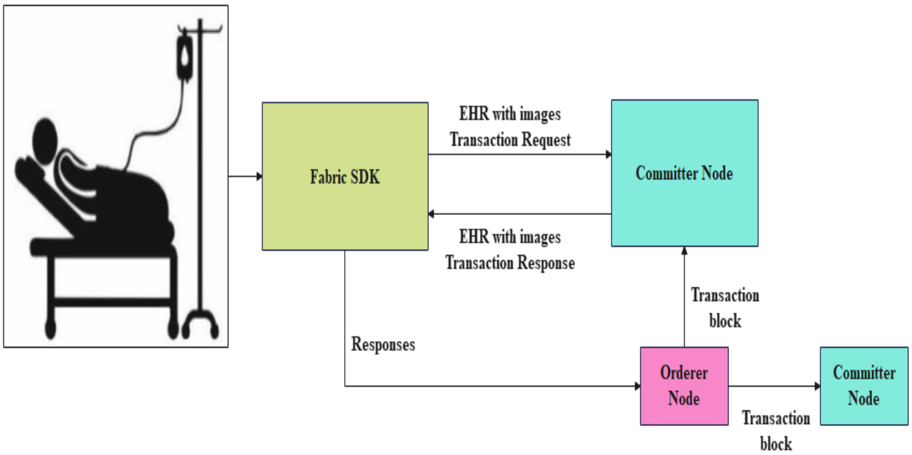


Fig 5: Hyperledger blockchain

The complex procedure inside the HB infrastructure designed for the healthcare industry is shown in Fig 5. The modular design approach used by Hyperledger Fabric is distinguished by its focus on security, resilience, adaptability, and scalability. The many intricacies inherent in the economy may be accommodated by this framework, which also allows for the seamless integration of many aspects. Many important components in the structure of a block, such as the following, provide support for the essential concepts of Fabric technology:

Chain Codes: These operate as self-executing programmes, akin to smart contracts, and are presently written in the Go programming language.

Channels: With the main goal of allowing secret transactions, channels represent private communication subnets among specified network participants, such as hospitals.

Ordering Service: In order to guarantee consistency and organised scheduling of transactions inside the network, the ordering service is essential.

Endorsement Policy: Nodes use the endorsement policy to decide whether to allow or reject transactions based on a set of rules.

Application Software Development Kit (SDK): The SDK allows peers to connect to the network and establish communication, improving network functioning as a whole.

Endorsing Peers: In order to pre-approve operations based on the endorsement policy set inside the chain codes, endorsing peers are essential.

Committing Peers: Committing peers are essential to the block verification process. Then, after updating the blocks' status in the State database (DB), they update the ledger. They obtain blocks from the ordering service.

5 Results and Discussion

To demonstrate the effectiveness of the recommended optical colour medical image cryptosystem, several colour medical pictures with various characteristics are selected and studied. To be used as input for the cryptosystem, the employed colour medical pictures are first largely divided into their R, G, and B components. The simulation tests are carried out using a laptop with an i7-5200 Intel CPU and 8 GB of RAM. The MATLAB R2020b is the programme used in the simulation experiments.

Entropy Results

The ciphered colour medical image's level of unpredictable behaviour is described using the entropy measure. The optimal entropy value for the suggested model is almost equal to 8. The analysis results for the original, encrypted, and decrypted colour medical pictures are shown in Table 1 for the proposed model and the traditional optical cryptosystems. By attaining better values that are closer to the required ideal value than alternative models, the recommended model's superior entropy values demonstrate its resilience and dependability. Fig 6 displays the qualitative outcomes of the suggested approach utilising different medical photos.

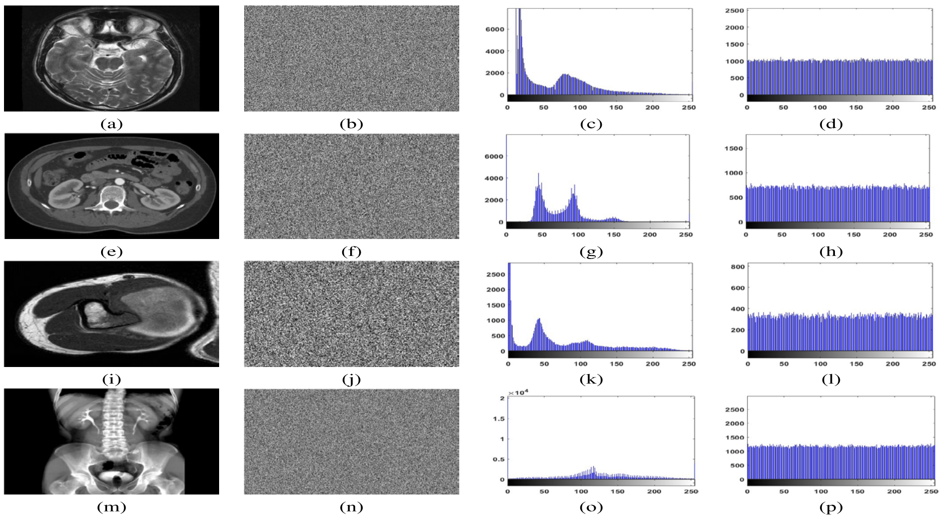


Fig 6: Results of the suggested model that are qualitative.
Rows top to bottom: brain MRI image; columns left to right: original images, cipher counterparts, original images histograms, and cipher images histograms. Top to bottom: abdomen CT scan of the kidney; knee MRI; rows top to bottom: X-ray of the spine.

Table 1: Entropy outcomes for the proposed model

Image	Original image	Encrypted image Without optimization	Encrypted image (proposed model)	Decrypted image (All algorithms)
Image 1	7.422	7.488	7.821	7.422

Image 2	5.380	7.073	7.436	5.380
Image 3	6.864	7.391	7.694	6.864
Image 4	5.083	7.231	7.798	5.083
Image 5	6.055	7.536	7.807	6.055

From table 1 and fig 7, image 1 had an original image quality of 7.422, 7.488 is an encrypted image quality without optimization, the proposed model (encrypted image) quality is 7.821 and decrypted image is 7.422. Image 2 had an original image quality of 5.380, 7.073 is an encrypted image quality without optimization, the proposed model (encrypted image) quality is 7.436 and decrypted image is 5.380. Image 3 had an original image quality of 6.864, 7.391 is an encrypted image quality without optimization, the proposed model (encrypted image) quality is 7.694 and decrypted image is 6.864. Image 4 had an original image quality of 5.083, 7.231 is an encrypted image quality without optimization, the proposed model (encrypted image) quality is 7.798 and decrypted image is 5.083. Image 5 had an original image quality of 6.055, 7.536 is an encrypted image quality without optimization, the proposed model (encrypted image) quality is 7.807 and decrypted image is 6.055.

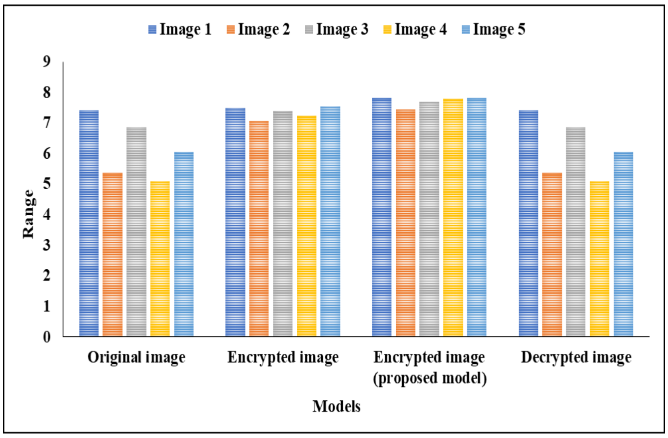


Fig 7: Entropy analysis

SSIM, FSIM and PSNR Analysis use important measures like SSIM (Structural Similarity), PSNR (Peak Signal-to-Noise Ratio), and FSIM (Feature Similarity) to evaluate the performance of the proposed model. We determine these parameters in our security research by contrasting the encrypted medical pictures with their original, unencrypted forms. More effective encryption is indicated by smaller metric values. Table 2 shows the results of the suggested model, which are obviously superior to conventional models since they provide considerably better metric values.

Table 2: Results of the suggested models' SSIM, FSIM, and PSNR tests on medical images in plain and encrypted formats.

	With out optimization			Proposed model		
	SSIM	FSIM	PSNR (dB)	SSIM	FSIM	PSNR (dB)
Image 1	0.0103	0.5129	10.5681	0.0143	0.4920	10.6102
Image 2	0.0190	0.5210	13.7794	0.0225	0.5203	13.8057
Image 3	0.0120	0.5007	11.7526	0.0119	0.5003	11.7475
Image 4	0.0017	0.4459	8.4068	0.0024	0.4450	8.3806

Image 5	0.0016	0.4386	7.8848	0.0030	0.4371	7.9084

The assessment findings, which are shown in Table 2 and the companion fig 8, 9 and 10, demonstrate the significant enhancements brought about by optimization for each of the five photos. Image 1 had an SSIM of 0.0103, FSIM of 0.5129, and PSNR of 10.5681 without optimizations. These parameters increased upon optimizations to an SSIM of 0.0143, FSIM of 0.4920, and PSNR of 10.6102. Similar to Image 1, Image 2 showed baseline scores of 0.0190 SSIM, 0.5210 FSIM, and 13.7794 PSNR. These values were then improved via optimizations to 0.0225 SSIM, 0.5203 FSIM, and 13.8057 PSNR. Before optimizations, Image 3 produced an SSIM of 0.0120, FSIM of 0.5007, and PSNR of 11.7526. After optimizations, these values slightly changed to 0.0119 SSIM, 0.5003 FSIM, and 11.7475 PSNR. Additionally, Image 4 was optimized from 0.017 SSIM, 0.4459 FSIM, and 8.4068 PSNR to 0.024 SSIM, 0.4450 FSIM, and 8.3806 PSNR. Last but not least, Image 5 had a non-optimized SSIM of 0.0016, FSIM of 0.4386, and PSNR of 7.8848, which was later enhanced to 0.0030 SSIM, 0.4371 FSIM, and 7.9084 PSNR. These results highlight the significant improvement made possible by the optimizations procedure for every image.

Fig 8: SSIM Analysis

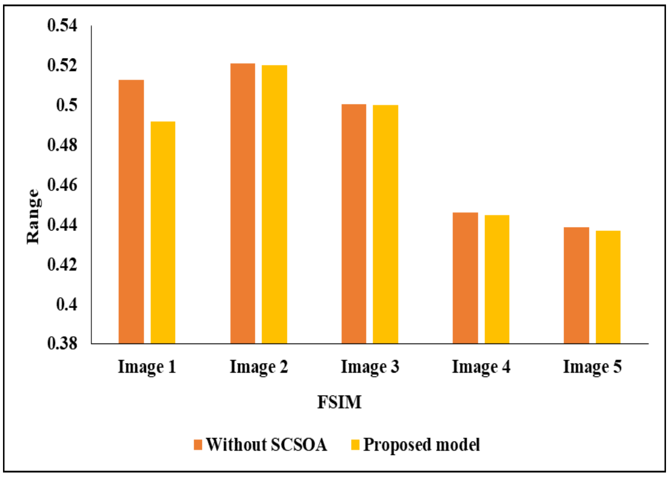


Fig 9: FSIM Analysis

Fig 10: PSNR Analysis

. Encryption/Decryption time analysis

Table 3: Encryption/decryption times for the proposed model.

Images	Without optimization	Proposed model with optimization
Image 1	4.6586	3.2527

Image 2	4.8272	3.1596
Image 3	4.4495	3.2047
Image 4	5.6271	3.3576
Image 5	5.7937	3.2129

Fig 11 shows the results of the investigation of encryption and decryption timings, which are reported in Table 3 and show a significant time decrease when optimizations are used. Without optimizations, the encryption and decryption timings for Images 1 and 2 were 4.6586 and 4.8272 seconds, respectively. For these procedures, Image 3 needed 4.495 seconds, Image 4 required 5.6271 seconds, and Image 5 required 5.7937 seconds. However, when the suggested model with optimizations was used, significant time savings were made. The encryption and decryption times for Images 1 and 2 were 3.2527 and 3.1596 seconds, respectively. Images 3 and 4 displayed times of 3.2047 and 3.3576 seconds, respectively, for these procedures. The encryption and decoding of Image 5 took 3.2129 seconds. These results demonstrate the efficiency benefits made possible by the optimizations procedure, with encryption and decryption durations for all photos being drastically slashed in seconds.

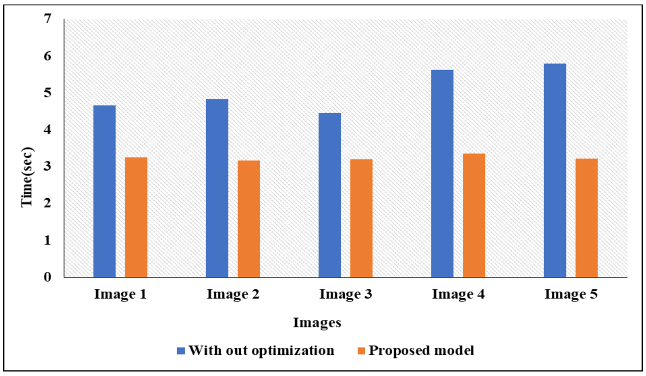


Fig 11: Encryption/Decryption analysis

6 Conclusion

To guarantee safe data transfer and streamline the diagnostic procedure, a thorough model has been developed in this study. The concept is composed of many operational stages, such as encryption using the SIMON block cipher, optimum key generation using SCSOA, and safe data management utilizing the HB. The security level of the transmission process for health records has been greatly increased by the inclusion of SCSOA in the key generation process. In addition, the use of HB technology allows the safe administration of medical information, giving people the power to give or revoke access to healthcare professionals and institutions. The Structural Similarity Index (SSIM) score of 0.0030 and the Feature Similarity Index (FSIM) score of 0.4371 show that the suggested model has accomplished outstanding achievements. These results represent a significant improvement demonstrating the effectiveness of the suggested paradigm. The HB should be optimized for managing medical data, according to future research. Investigating methods to speed up transactions, use less energy, and better blockchain performance often falls under this category.

References

1. Li, F., Lam, K. Y., Li, X., Sheng, Z., Hua, J., & Wang, L. (2019). Advances and emerging challenges in cognitive internet-of-things. *IEEE Transactions on Industrial Informatics*, *16*(8), 5489-5496.
2. Rahman, M. A., Rashid, M. M., Hossain, M. S., Hassanain, E., Alhamid, M. F., & Guizani, M. (2019). Blockchain and IoT-based cognitive edge framework for sharing economy services in a smart city. *Ieee Access*, *7*, 18611-18621.
3. Akkasaligar, P. T., & Biradar, S. (2020). Selective medical image encryption using DNA cryptography. *Information Security Journal: A Global Perspective*, *29*(2), 91-101.
4. Wu, Y., Zhang, L., Berretti, S., & Wan, S. (2022). Medical image encryption by content-aware dna computing for secure healthcare. *IEEE Transactions on Industrial Informatics*, *19*(2), 2089-2098.
5. Lin, H., Wang, C., Cui, L., Sun, Y., Zhang, X., & Yao, W. (2022). Hyperchaotic memristive ring neural network and application in medical image encryption. *Nonlinear Dynamics*, *110*(1), 841-855.
6. Jain, K., Aji, A., & Krishnan, P. (2021). Medical image encryption scheme using multiple chaotic maps. *Pattern Recognition Letters*, *152*, 356-364.
7. Ravichandran, D., Banu S, A., Murthy, B. K., Balasubramanian, V., Fathima, S., & Amirtharajan, R. (2021). An efficient medical image encryption using hybrid DNA computing and chaos in transform domain. *Medical & biological engineering & computing*, *59*, 589-605.
8. Banik, A., Shamsi, Z., & Laiphrakpam, D. S. (2019). An encryption scheme for securing multiple medical images. *Journal of Information Security and Applications*, *49*, 102398.
9. Kaissis, G., Ziller, A., Passerat-Palmbach, J., Ryffel, T., Usynin, D., Trask, A., ... & Braren, R. (2021). End-to-end privacy preserves deep learning on multi-institutional medical imaging. *Nature Machine Intelligence*, *3*(6), 473-484.
10. Belazi, A., Talha, M., Kharbech, S., & Xiang, W. (2019). Novel medical image encryption scheme based on chaos and DNA encoding. *IEEE access*, *7*, 36667-36681.
11. Deb, S., & Bhuyan, B. (2021). Chaos-based medical image encryption scheme using special nonlinear filtering function based LFSR. *Multimedia Tools and Applications*, *80*, 19803-19826.
12. Lakshmi, C., Thenmozhi, K., Rayappan, J. B. B., Rajagopalan, S., Amirtharajan, R., & Chidambaram, N. (2021). Neural-assisted image-dependent encryption scheme for medical image cloud storage. *Neural Computing and Applications*, *33*, 6671-6684.

13. Khashan, O. A., & AlShaikh, M. (2020). Edge-based lightweight selective encryption scheme for digital medical images. *Multimedia Tools and Applications*, *79*(35-36), 26369-26388.
14. Chen, Y., Tang, C., & Ye, R. (2020). Cryptanalysis and improvement of medical image encryption using high-speed scrambling and pixel adaptive diffusion. *Signal Processing*, *167*, 107286.
15. Pankaj, S., & Dua, M. (2021). A novel ToCC map and two-level scrambling-based medical image encryption technique. *Network Modeling Analysis in Health Informatics and Bioinformatics*, *10*, 1-19.
16. Lai, Q., Hu, G., Erkan, U., & Toktas, A. (2023). High-efficiency medical image encryption method based on 2D Logistic-Gaussian hyperchaotic map. *Applied Mathematics and Computation*, *442*, 127738.
17. Wang, X., & Wang, Y. (2023). Multiple medical image encryption algorithm based on scrambling of region of interest and diffusion of odd-even interleaved points. *Expert Systems with Applications*, *213*, 118924.
18. Abdelfatah, R. I., Saqr, H. M., & Nasr, M. E. (2023). An efficient medical image encryption scheme for (WBAN) based on adaptive DNA and modern multi chaotic map. *Multimedia Tools and Applications*, *82*(14), 22213-22227.
19. Trujillo-Toledo, D. A., López-Bonilla, O. R., García-Guerrero, E. E., Esqueda-Elizondo, J. J., Cárdenas-Valdez, J. R., Tamayo-Pérez, U. J., ... & Inzunza-González, E. (2023). Real-time medical image encryption for H-IoT applications using improved sequences from chaotic maps. *Integration*, *90*, 131-145.
20. John, S., & Kumar, S. N. (2023). IoT based medical image encryption using linear feedback shift register–Towards ensuring
21. HAZZAZI, M. M. Noise-Resistant Image Encryption Scheme for Medical Images in the Chaos and Wavelet Domain.
22. Singh, K. N., Singh, O. P., Singh, A. K., & Agrawal, A. K. (2023). EiMOL: a secure medical image encryption algorithm based on optimization and the Lorenz system. *ACM Transactions on Multimedia Computing, Communications and Applications*, *19*(2s), 1-19.
23. Wu, D., Rao, H., Wen, C., Jia, H., Liu, Q., & Abualigah, L. (2022). Modified sand cat swarm optimization algorithm for solving constrained engineering optimization problems. *Mathematics*, *10*(22), 4350.
24. Fernandes A, Rocha V, da Conceição AF, Horita F (2020) Scalable architecture for sharing EHR using the hyperledger blockchain. In: 2020 IEEE international conference on software architecture companion (ICSA-C). IEEE, pp 130–138.
    Stamatellis C, Papadopoulos P, Pitropakis N, Katsikas S, Buchanan WJ (2020) A privacy-preserving healthcare framework using hyperledger fabric. Sensors 20(22):6587.

PCAP-Forensics: An Automated Network Traffic Analysis Framework

Darshan Krishna Hegde[1*†], Nikhil Kumar C[1†], Pradeep Kumar[1†], Prasad B Honnavalli[2], Nagasundari S[2], Sapna V M[1]

[1]Department of Computer Science and Engineering, PES University, Bangalore, 560085, Karnataka, India.
[2]PES University, Bangalore, India.

*Corresponding author(s). E-mail(s): darshuhegde2002@gmail.com;
Contributing authors: nikhilkumarc2003@gmail.com; prathapm2016@gmail.com; prasadhb@pes.edu; nagasundaris@pes.edu; sapnavm@pes.edu;
†These authors contributed equally to this work.

Abstract

With the rapid expansion of digital communication networks, ensuring cybersecurity and maintaining network integrity has become a critical challenge. Traditional intrusion detection systems rely on static rules, making them ineffective against evolving cyber threats. This research presents a real-time network forensic analysis framework that integrates packet sniffing, traffic visualization, and anomaly detection using machine learning techniques. The proposed system utilizes Scapy for packet capture, PyShark for processing PCAP files, and the Isolation Forest algorithm for anomaly detection. The framework provides detailed insights through real-time monitoring, packet analysis, and visualization techniques such as protocol distribution, traffic volume analysis, and port monitoring. By implementing a machine learning-based detection system, our approach effectively identifies network anomalies, including SYN flood attacks, port scanning, and other suspicious activities. Experimental evaluations demonstrate the framework's effectiveness in detecting network anomalies with high accuracy, making it a valuable asset for forensic investigations and network security monitoring.

Keywords: Network forensics,Network analysis, Machine learning,Cybersecurity.

1 Introduction

With the widespread adoption of digital technologies, ensuring the security and integrity of network communications has become a pressing challenge. Cyber threats such as denial-of-service (DoS) attacks, unauthorized access, data breaches, and malware infections have increased in complexity, making traditional network security solutions insufficient [3]. Organizations rely on Intrusion Detection Systems (IDS) and firewalls to monitor and secure their networks. However, most of these systems depend on signature-based detection, which limits their ability to identify novel attacks [5]. As cybercriminals continue to exploit zero-day vulnerabilities and advanced evasion techniques, security solutions must evolve to become more adaptive and intelligence-driven [17].

Traditional signature-based IDS solutions such as Snort and Suricata are effective at detecting known threats but struggle against previously unseen attack patterns. Rule- based security mechanisms also generate a high volume of false positives, leading to alert fatigue and making manual analysis inefficient.

To address these shortcomings, machine learning-based network anomaly detection has gained prominence in cybersecurity research [21]. Unlike traditional IDS, anomaly detection models identify suspicious activities based on deviations from normal network behavior, making them effective against zero-day attacks and novel cyber threats [22]. Recent advancements in artificial intelligence applications for cybersecurity further emphasize the importance of machine learning-driven network forensics and threat detection [3, 23].

This research presents a Network Forensic Analysis Framework that integrates real-time packet capture, interactive traffic visualization, and anomaly detection using the Isolation Forest algorithm. The system aims to enhance network security monitoring and forensic investigations by:

1. Capturing and storing real-time network packets for forensic analysis.
2. Visualizing network behavior using interactive graphs and charts.
3. Detecting anomalies using machine learning to identify suspicious activities.
4. Generating automated forensic reports summarizing network trends and security threats.

2 Related Work

The field of network traffic analysis and intrusion detection has been extensively studied in cybersecurity research[24]. Traditional approaches rely on signature-based and rule-based detection systems, while recent advancements focus on statistical anomaly detection and machine learning techniques[25].

2.1 Signature-Based Intrusion Detection

Signature-based IDS solutions, such as Snort and Suricata, compare incoming network traffic against a database of known attack signatures[4]. While these systems are effective at detecting well-documented threats, they are inherently incapable of identifying novel attacks[5]. The primary limitation of signature-based detection is its dependence on predefined rules, which must be constantly updated to keep pace with evolving cyber threats[6]. Additionally, these systems generate a high volume of false positives, requiring manual intervention from security analysts to verify alerts[7].

2.2 Deep Learning Based Approach

To overcome the limitations of signature-based detection, researchers have explored statistical methods to identify anomalies in network traffic. Entropy-based and deviation-based models attempt to establish a baseline of normal network behavior and flag deviations as potential threats[8]. However, statistical methods often suffer from high false-positive rates, as normal fluctuations in traffic patterns can be misclassified as anomalies[9].

2.3 Machine Learning for Intrusion Detection

With advancements in artificial intelligence, machine learning-based network security solutions have gained popularity. Various supervised and unsupervised learning models have been applied to detect anomalies in network traffic[1].

- **Supervised learning approaches**, such as Support Vector Machines (SVM) and Random Forest classifiers, require labeled datasets to distinguish between normal and malicious traffic[10]. However, obtaining labeled network traffic data is challenging and often impractical in real-world scenarios.
- **Unsupervised learning models**, such as Autoencoders, K-Means clustering, and Isolation Forest, are well-suited for anomaly detection since they do not require labeled datasets[11]. Among these, Isolation Forest has proven to be highly effective for identifying outliers in large-scale network traffic datasets.

2.4 Isolation Forest for Anomaly Detection

The Isolation Forest algorithm is an unsupervised learning technique designed for efficient anomaly detection in high-dimensional datasets[11]. Unlike traditional clustering methods, Isolation Forest isolates anomalous instances by recursively partitioning the feature space, making it computationally efficient for large-scale network monitoring. Several studies have demonstrated that Isolation Forest outperforms traditional anomaly detection methods, such as Gaussian Mixture Models (GMM) and One-Class SVM, in detecting network intrusions[12].

2.5 Objective

The primary objective of this project is to develop a comprehensive network forensic analysis framework that integrates real-time packet capture, traffic visualization, and machine learning-based anomaly detection to enhance network security monitoring and forensic investigations.

The key objectives of this research are:

1. **Real-Time Packet Capture and Storage:** Implementing a packet sniffing mechanism using Scapy to capture live network traffic and store it in PCAP format for further analysis.
2. **Network Traffic Analysis and Visualization:** Developing interactive visualization techniques, including protocol distribution charts, traffic volume histograms, top talkers analysis, and port monitoring, to provide insights into network behavior and potential security threats.
3. **Anomaly Detection Using Machine Learning:** Integrating the Isolation Forest algorithm to identify suspicious network activities, such as SYN flood attacks, port scanning attempts, and anomalous traffic spikes, by analyzing deviations from normal network behavior.
4. **Automated Forensic Reporting:** Generating structured reports summarizing key network events, detected anomalies, and potential security threats to assist in forensic investigations and incident response.

3 Methodology

The Network Forensic Analysis Framework consists of three main components: Packet Capture and Storage, Traffic Analysis and Visualization, and Anomaly Detection using Machine Learning[1, 26]. With AI-driven approaches playing a critical role in modern network security solutions, frameworks leveraging artificial intelligence for forensic investigations are becoming increasingly sophisticated[2, 28, 29]. Each component plays a crucial role in ensuring comprehensive network monitoring and forensic analysis[10, 12].

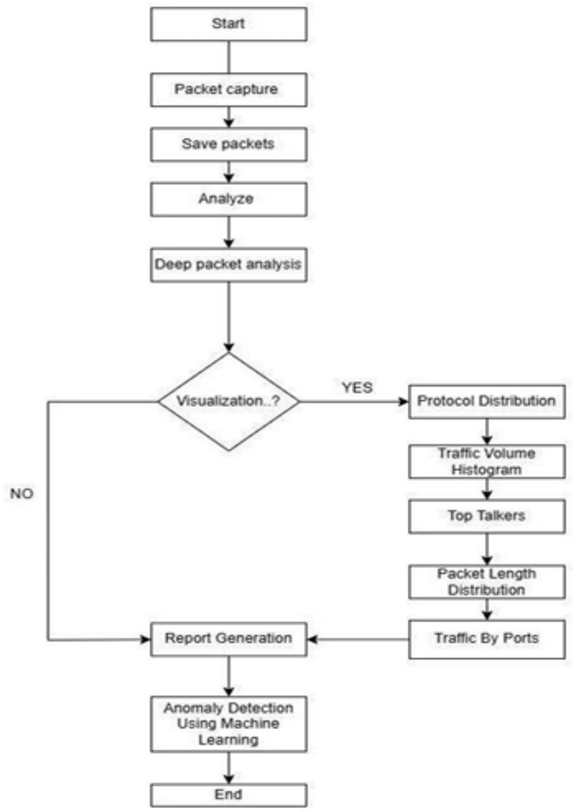


Fig. 1 Steps to analyse Network Traffic.

3.1 Packet Capture and Storage

The foundation of network traffic analysis is packet capture, which involves monitoring, collecting, and storing network packets for forensic examination. Our system uses Scapy, a Python-based packet manipulation library, to sniff network packets in real time. Captured packets are stored in PCAP format, allowing investigators to analyze past network events[14].

For efficient data processing, PyShark is used to extract packet attributes from PCAP files. The extracted attributes include:

- Source and Destination IP addresses
- Protocol Type (TCP, UDP, ICMP, etc.)
- Packet Length and Payload Size
- Timestamp Information

By storing packets systematically, our system enables in-depth forensic analysis of historical network data.

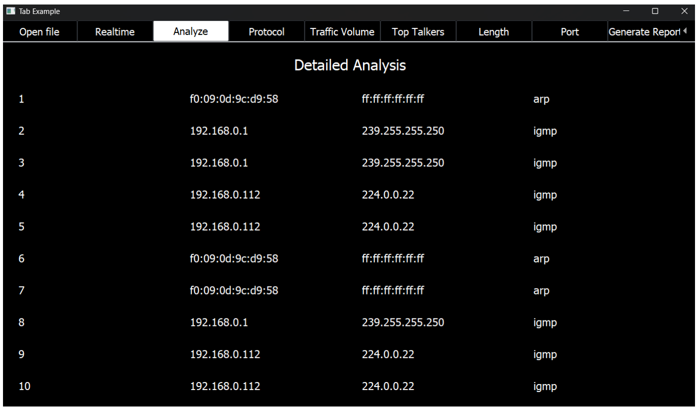


Fig. 2 Detailed Analysis Of Packets

Fig. 3 Analysis Of One Packet

3.2 Traffic Analysis and Visualization

To facilitate network monitoring and forensic investigations, our framework provides interactive visualizations of network traffic behavior. Each visualization method provides unique insights into network behavior, making it easier to detect suspicious activity before it escalates into a security incident.

3.2.1 Protocol Distribution Analysis

The protocol distribution analysis provides a pie chart representation of different network protocols present in the captured traffic, including Transmission Control Protocol (TCP), User Datagram Protocol (UDP), Internet Control Message Protocol (ICMP), and other network-layer protocols.

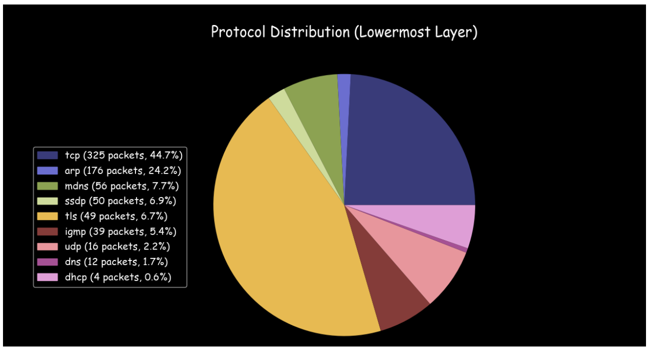


Fig. 4 Protocol Distribution

3.2.2 Traffic Volume Monitoring

A histogram representation of network traffic volume is generated to track total data transmission over time. Traffic patterns are analyzed over a specific time window, allowing analysts to observe sudden surges in data transmission. By correlating traffic volume with protocol distribution and port activity, analysts can pinpoint the source of anomalous traffic and implement mitigation strategies such as rate limiting, IP blocking, or deep packet inspection (DPI) to prevent further damage.

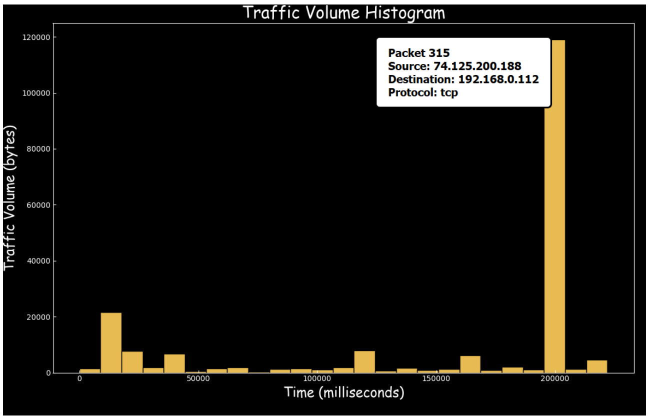


Fig. 5 Traffic Volume Histogram

3.2.3 Top Talkers

The Top Talkers analysis generates a bar chart listing the most active IP addresses in the network, ranked based on their traffic volume. This feature is useful for identifying compromised hosts, detecting unauthorized access, and analyzing network resource consumption.

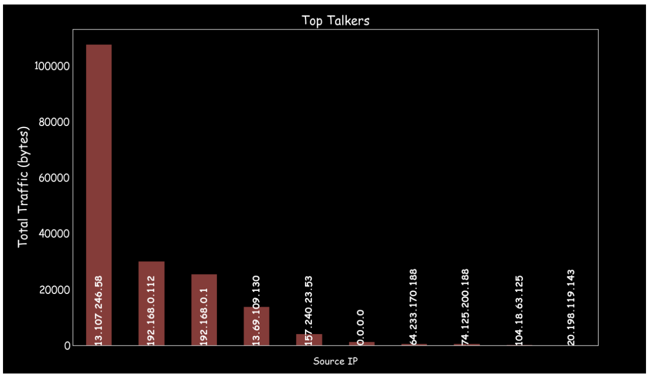


Fig. 6 Top Talkers

3.2.4 Port Usage Analysis

The port usage analysis generates a funnel chart depicting the most frequently accessed ports, allowing analysts to identify suspicious port activity and unauthorized scanning attempts. Normal network activity typically involves well-known ports, such as Port 80 (HTTP), Port 443 (HTTPS), Port 22 (SSH), and Port 53 (DNS).

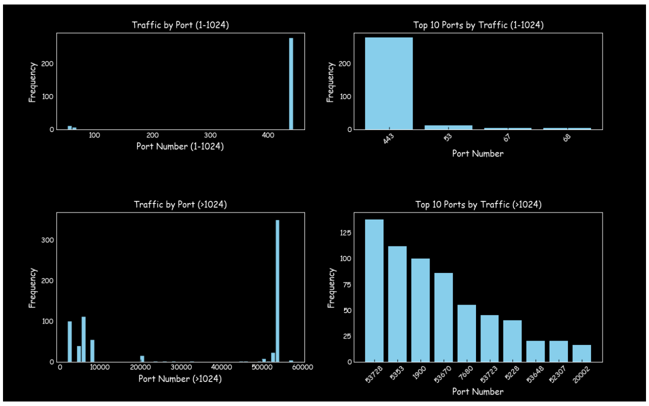


Fig. 7 Port Usage Analysis

3.2.5 Packet Length Analysis

The Packet Length Analysis tab in the project visualizes the distribution of packet sizes captured from the network traffic. It provides insights into how data is transmitted by displaying a histogram of packet lengths. By hovering over the visualization, users can see detailed information about individual packets, including their source, destination, and protocol type, aiding in forensic analysis and real-time network monitoring.

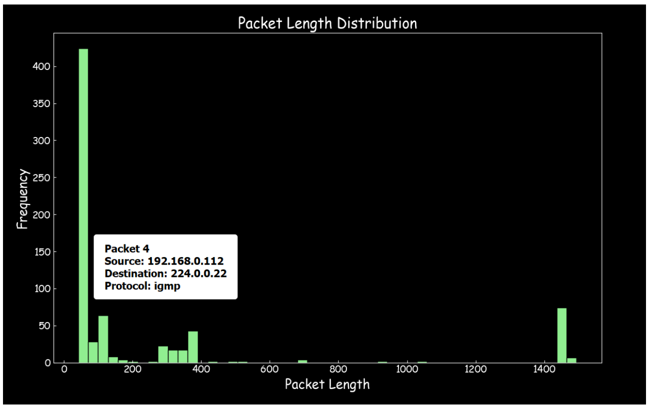


Fig. 8 Packet Length Distribution

4 Experimentation: Anomaly Detection Using Machine Learning

The methodology is detailed in sequential stages for each architecture, describing each key component's function and contribution to the anomaly detection process [1].

4.1 Feature Extraction for Anomaly Detection

The first step in implementing machine learning in our system involves feature extraction from network packets [11, 27]. Every network packet captured through Scapy and processed via PyShark contains multiple attributes that describe its behavior, including source and destination IP addresses, protocol type, timestamp, and payload size [13]. However, not all packet attributes are equally useful for anomaly detection [16]. To improve model performance, we extracted the following key network features that serve as input to the Isolation Forest algorithm:

- **Protocol Type (Numerically Encoded):** Different network protocols (TCP, UDP, ICMP) behave uniquely. Encoding them numerically allows the model to recognize protocol-based anomalies.
- **Relative Timestamp of Network Events:** The time difference between consecutive packets provides insight into packet burst patterns, which can help identify DoS attacks or scanning behaviors.

4.2 Training the Isolation Forest Model

Once the dataset is preprocessed, we train the Isolation Forest model to learn normal network behavior and detect anomalies. Isolation Forest works by randomly selecting features and partitioning the dataset recursively until each sample is isolated. Anomalous samples get isolated faster, leading to shorter path lengths, while normal samples remain within dense clusters with longer path lengths.

We utilize the Isolation Forest algorithm, a widely used unsupervised learning technique, to identify anomalous packets. The model is trained on normal traffic patterns and assigns an Anomaly Score to each packet:

- Normal packets are assigned a score of 1.
- Anomalous packets (potential threats) receive a score of -1.

A contamination factor of 5% is used, assuming a small percentage of network traffic is anomalous.

4.3 Anomaly Reporting and Analysis

The detected anomalies are saved into an `anomalies.csv` file for further forensic analysis. This dataset provides insights into:

- Unusual packet lengths or protocol usage.
- Spikes in network traffic that may indicate a DDoS attack.
- Uncommon communication between source and destination IPs, potentially signaling malware activity.

Our anomaly detection system enables real-time threat detection, assisting in network security monitoring and cyber forensic investigations.

5 Results Obtained During Experimentation

To evaluate the effectiveness of our Network Forensic Analysis Framework, we conducted extensive experiments on real-world network traffic and simulated cyberattacks [3]. The primary goal was to assess the system's accuracy, efficiency, and reliability in detecting network anomalies [17]. The evaluation focused on anomaly detection performance, visualization effectiveness, and forensic analysis capabilities [5].

The experiments were conducted in a controlled environment, where normal network traffic and malicious activities were generated to assess how well the system could distinguish between the two [14]. **Legitimate Network Traffic:**

- Regular browsing activities, file transfers, and video streaming.
- Normal communication between devices using TCP, UDP, and ICMP protocols.

5.1 Visualization Effectiveness

The visualization dashboard played a crucial role in helping analysts quickly interpret network behavior and detect suspicious activities.

Protocol Distribution Analysis

- Helped identify an unexpected spike in ICMP traffic, indicating a potential network scanning attempt.
- Showed that 80% of normal traffic was TCP-based, while anomalous traffic had higher UDP and ICMP usage.

Traffic Volume Monitoring

- Successfully detected traffic surges corresponding to SYN flood attacks.
- Allowed analysts to correlate sudden spikes with attack timestamps for forensic analysis.

Top Talkers (Source IP Analysis)

- Clearly highlighted the IP addresses generating the most traffic.
- Allowed easy identification of malicious IPs performing port scans.

Port Usage Analysis

- Showed unusual activity on non-standard ports, helping detect possible exploitation attempts.
- Identified port scanning activity due to frequent access to multiple ports in a short time.

5.2 Forensic Report Generation

The system automatically generated structured forensic reports summarizing key findings, including:

- Timestamps of detected anomalies.
- List of top suspicious IP addresses.
- Breakdown of protocol anomalies and port activity.
- Packet metadata, such as source/destination IPs, payload size, and flags.

These reports were valuable for:

- Incident investigation and response.
- Tracking attack trends over time.
- Providing evidence for forensic analysis.

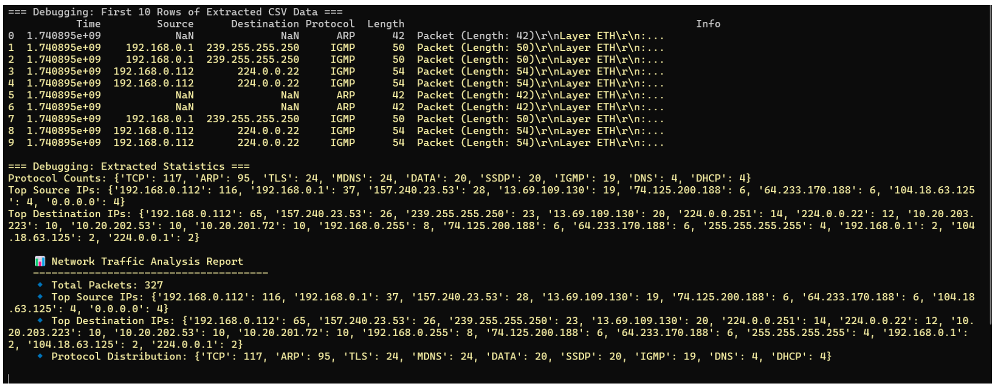


Fig. 9 Report generation

5.3 Real-Time Anomaly Detection and Classification

The Real-Time Anomaly Detection and Classification module enables continuous monitoring and analysis of network traffic to identify abnormal behaviors and potential security threats as they occur [1]. This system leverages machine learning techniques, statistical analysis, and real-time packet inspection to classify anomalies into different categories, allowing for immediate response and forensic investigation.

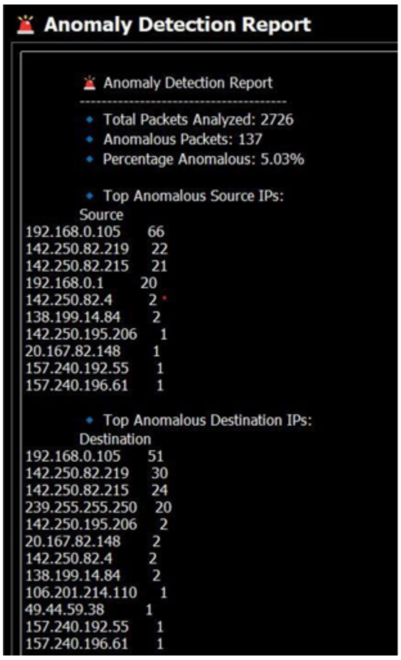


Fig. 10 Anomaly detection

5.4 Comparison with Wireshark

Network traffic analysis plays a crucial role in the field of digital forensics, as Wireshark is the most widely used framework for analysis. It provides in-depth packet analysis but lacks certain features that are essential for forensic investigations, such as real-time monitoring with visualization, automated report generation, and machine learning-driven analysis.

Feature	Proposed Tool/Framework	Wireshark
Real time monitoring	Provides live traffic visualization with an intuitive GUI	Requires extensive filtering and lacks built-in visualization
Packet capture	Uses scapy and pyshark for live capture.	Uses winpcap/libpcap which offers deep packet capture.
Packet analysis	Displays detailed packet information in a dedicated window for forensic review.	Provides deep packet inspection, but in a raw text-based format.
Protocol analysis	Supports protocol filtering with graphical insights.	In-depth protocol breakdown, but lacks visual representation.
Data Visualization	Uses Matplotlib and seaborn to generate graphs, histograms and pie charts.	No built-in visualization.
Customization and extensibility.	Can be extended with additional support.	Requires scripting and plugin development.
Report generation.	Uses ML to generate the reports.	Requires manual export and interpretation.

Table 1: Comparision between wireshark and proposed framework

6 Discussion

6.1 Framework Advantages

The Network Forensic Analysis Framework provides an integrated solution for network security monitoring by combining real-time packet capture, anomaly detection, and data visualization into a single framework [18]. The framework enhances forensic investigations by offering interactive visualizations such as protocol distribution analysis, traffic volume monitoring, and port activity tracking, allowing security analysts to quickly identify suspicious patterns [19]. The use of machine learning-based anomaly detection, specifically the Isolation Forest algorithm, automates the identification of threats like SYN flood attacks, port scans, and malformed packet injections, eliminating reliance on predefined signatures.

Furthermore, the automated forensic reporting feature provides detailed insights into network behavior and detected anomalies, improving the efficiency of cyber threat analysis and incident response [20].

6.2 Challeng es and Limitations

Despite its advantages, the Network Forensic Analysis Tool (NFAT) faces several technical challenges, particularly in handling large-scale network traffic and optimizing processing efficiency. Memory usage and computational overhead can become significant concerns when analyzing high-volume network data, requiring further performance optimizations.

Additionally, the accuracy of anomaly detection depends on the quality of extracted network features, and tuning the Isolation Forest model to reduce false positives and false negatives required extensive refinement. Another limitation is the lack of a labeled dataset, which restricts the ability to train supervised learning models for more precise anomaly classification. The system also does not currently support real-time alerting, requiring manual intervention for security monitoring.

6.3 Future Work

Future work on this framework will focus on establishing a structure for continuous network traffic monitoring. Integrating advanced AI methodologies could further refine the anomaly detection process, reducing false positives and enhancing adaptability against emerging threats. This will enable proactive management of network performance and security, allowing users to detect and respond to threats in real-time.

The addition of continuous monitoring capabilities will further enhance the framework's utility, making it an even more valuable resource for digital forensic investigations.

7 Conclusion

With the rapid increase in cyber threats and sophisticated attack techniques, traditional signature-based intrusion detection systems (IDS) are no longer sufficient to effectively secure modern networks. The need for adaptive, intelligent, and real-time network monitoring solutions has led to the development of machine learning-based anomaly detection systems.

This research presented a Network Forensic Analysis Framework that integrates real-time packet capture, interactive traffic visualization, and machine learning-based anomaly detection to enhance network security monitoring and forensic investigations.

The proposed system successfully captured and analyzed network traffic using Scapy and PyShark, extracted meaningful features, and applied the Isolation Forest algorithm to detect anomalous network behaviors such as SYN flood attacks, port scanning, and malformed packet injections.

The visualization dashboard provided intuitive insights into protocol distribution, traffic volume trends, top talkers, and port activity, enabling security analysts to quickly identify potential security threats. The system's ability to generate detailed forensic reports further aids in post-incident investigations, making it a valuable framework for cyber-security professionals and forensic analysts.

8 Acknowledgement

We would like to express our sincere gratitude to the U.S. Army for their generous funding and unwavering support, which made the AI-Enabled National Conference on Forensic Investigations Network in Digital Sciences (FINDS) possible. Their commitment to advancing research and fostering collaboration is deeply appreciated.

We are profoundly grateful to Dr. S. S. Iyengar for his outstanding contributions to the field, which have been a constant source of inspiration to us. His pioneering work has shaped much of our research, and we are deeply thankful for the impact his research has had on our academic journey. His dedication and innovations continue to inspire us in our pursuit of knowledge.

References

[1] S.S. Iyengar, Sina Nabavi, Yashas Hariprasad, Prasad H.B., Krishna Mohan C. *Artificial Intelligence in Practice: Theory and Applications for Cyber Security and Forensics*. Springer Nature. (Forthcoming)

[2] Iyengar, S. S., & Geeta, S. *Data Auditing and Security in Cloud Computing: Issues, Challenges, and Future Directions*. Springer, 2023.

[3] Shi, B., & Iyengar, S. S. *Mathematical Theories of Machine Learning: Theory and Applications*. Springer, 2019.

[4] Roesch, M. "Snort: Lightweight Intrusion Detection for Networks." *Proceedings of USENIX LISA*, 1999, pp. 229–238.

[5] Paxson, V. "Bro: A System for Detecting Network Intruders in Real-Time." *Computer Networks*, vol. 31, no. 23–24, 1999, pp. 2435–2463.

[6] Sommer, R., & Paxson, V. "Outside the Closed World: On Using Machine Learning for Network Intrusion Detection." *IEEE Symposium on Security and Privacy*, 2010, pp. 305–316.

[7] Garcia-Teodoro, P., Diaz-Verdejo, J., Macia-Fernandez, G., & Vazquez, E. "Anomaly-based Network Intrusion Detection: Techniques, Systems, and Challenges." *Computers & Security*, 28(1–2), 18–28, 2009.

[8] Bhuyan, M. H., Bhattacharyya, D. K., & Kalita, J. K. "Network Anomaly Detection: Methods, Systems and Frameworks." *IEEE Communications Surveys & Tutorials*, 16(1), 303–336, 2014.

[9] Lakhina, A., Crovella, M., & Diot, C. "Mining Anomalies Using Traffic Feature Distributions." *ACM SIGCOMM*, 2005, pp. 217–228.

[10] Buczak, A. L., & Guven, E. "A Survey of Data Mining and Machine Learning Methods for Cyber Security Intrusion Detection." *IEEE Communications Surveys & Tutorials*, 18(2), 1153–1176, 2016.

[11] Liu, F. T., Ting, K. M., & Zhou, Z. H. "Isolation Forest." *IEEE ICDM*, 2008, pp. 413–422.

[12] Zhang, X., Li, Z., Wang, F., & Yuan, P. "Anomaly Detection in Network Traffic Using Isolation Forest." *Journal of Network and Computer Applications*, 154, 102538, 2020.

[13] Sadreazami, H., Mohammadi, A., & Asif, M. S. "Deep Learning for Network Anomaly Detection: A Comparative Study." *IEEE Transactions on Information Forensics and Security*, 17, 2105–2117, 2022.

[14] Ring, M., Wunderlich, S., Scheuring, D., Landes, D., & Hotho, A. "A Survey of Network-based Intrusion Detection Data Sets." *Computers & Security*, 86, 147–167, 2019.

[15] Liu, F., & Chen, G. "Towards Network Anomaly Detection Using Graph Embedding." *IEEE Transactions on Network and Service Management*, 20(3), 1234–1245, 2023.

[16] Kim, J., & Lee, S. "Network Forensics and Traffic Analysis With Machine Learning." *Cybersecurity*, 6, 22, 2023.

[17] Wang, P., & Zhang, Q. "Cloud Network Anomaly Detection Using Machine and Deep Learning Techniques: Recent Research Advancements." *Journal of Cloud Computing*, 12, 56, 2023.

[18] Chen, L., & Liu, Y. "A Comprehensive Investigation of Anomaly Detection Methods in Industrial Systems." *IET Cyber-Physical Systems: Theory & Applications*, 8(2), 123–134, 2023.

[19] Jones, M., & Davis, S. "A Comprehensive Analysis of the Role of Artificial Intelligence and Machine Learning in Modern Digital Forensics and Incident Response." *Forensic Science International: Digital Investigation*, 45, 301–315, 2023.

[20] Vert, G., Iyengar, S. S., & Phoha, V. *Introduction to Contextual Processing Theory and Application*. CRC Press, 2023.

[21] Xavier, C., & Iyengar, S. S. *Introduction to Parallel Algorithms (Chinese)*. 263 pages, ISBN: 7-111-13390-0.

[22] Xavier, C., & Iyengar, S. S. *Introduction to Parallel Algorithms*. John Wiley and Sons, 2000. 365 pages.

[23] Soloway, E., & Iyengar, S. S. (eds.) *Empirical Studies of Programmers*. Ablex Pub. Co., Norwood, New Jersey.

[24] Brooks, R. R., & Iyengar, S. S. *Multi Sensor Fusion: Fundamentals and Applications with Software*. Prentice Hall, New Jersey, 2000. 488 pages.

[25] Patel, S. J., Chaudhary, N. K., Gohil, B. N., & Iyengar, S. S. *Information Security, Privacy and Digital Forensics – ICISPD 2022*. Springer Nature, ISBN: 978-981-99-5090-4.

[26] Iyengar, S. S., & Boroojeni, K. G. *Oblivious Network Routing: Algorithms and Applications*. MIT Press, 2015, 176 pages. ISBN: 9780262029155.

[27] Iyengar, S. S., & Seetharaman, G. *The Design and Analysis of Algorithms for Processing Digital Satellite IR Images*. (In Preparation)

[28] Wang, C., Iyengar, S. S., & Sun, K. *AI Embedded Assurance for Cyber Systems*. Springer Nature, 2023. ISBN: 978-3-031-42636-0.

[29] Singaram, J., Iyengar, S. S., & Madni, A. M. *Deep Learning Networks: Design, Development and Deployment*. Springer Nature, 2023. ISBN: 978-3-031-39243.

User Behaviour Analysis Using Browsing History and to Support Forensics Investigation

Pradeep Kumar[1*†], Nikhil Kumar C[1*†],
Pradeep Y N[1*†], Harshith Reddy C[1*†],
Prasad B Honnavalli[2], Sapna V M[1]

[1]*Department of Computer Science and Engineering, PES University, Bangalore, 560085, Karnataka, India.*

[*]*Corresponding author(s). E-mail(s): prathapm2016@gmail.com; pradeep.trin@gmail.com; nikhilkumarc2003@gmail.com; harshithreddyc2003@gmail.com;*
Contributing authors: prasadhb@pes.edu; sapnavm@pes.edu;
[†]*These authors contributed equally to this work.*

Abstract

In the present digital era, where cyber threats are increasing in both occurrence and complexity, the capability to analyze user behavior and identify risks associated with web browsing has become an essential aspect of modern cybersecurity (Vert, Iyengar, & Phoha, 2024). As the internet becomes integral to communication, information retrieval, and business operations, individuals face growing risks from online threats such as malware, phishing attacks, and various forms of cybercrime (Iyengar et al., 2024). This study introduces an innovative approach to analyzing user behavior through browser history data, aiming to classify users into two categories: Normal and Abnormal. The classification framework is designed to provide actionable insights into online threats by examining patterns and activities recorded in users' browsing history.

The proposed approach leverages advanced machine learning techniques, specifically focusing on the Random Forest and XGBoost algorithms, to assess user-accessed URLs and

determine their risk levels (Shi & Iyengar, 2024). By assigning a risk score to each URL, this methodology enhances cybersecurity measures by enabling a more precise classification of user behavior. The system incorporates supervised learning techniques based on parameters such as protocol type, domain age, URL length, and suspicious keyword presence, increasing the accuracy of detecting and classifying potentially harmful URLs.

The system has an intuitive interface for smooth interaction to guarantee usability and accessibility. A simple process allows users to upload their surfing data, which is then processed by the system to produce thorough reports automatically. These reports emphasize important findings, such as user classifications, risk assessments, and statistical analysis of browsing behaviors. These reports are enhanced with dynamic graphics made with the D3.js package to improve interpretability. Bar charts, pie charts, and other graphical components are examples of these visualizations, which offer an interactive and understandable depiction of URL statistics and related risk levels.

The Visualization Dashboard, a crucial component of the project, enables users to interactively examine their surfing habits. The dashboard makes it easier to spot patterns that point to unusual or maybe harmful conduct by displaying surfing data in an organized and eye-catching way. Users can better understand their online behaviors with features like bubble charts that show domainwise risk distributions, heatmaps that show temporal patterns of user activity, and comparison bar charts for frequency analysis. The dashboard also offers tailored suggestions for reducing the hazards connected to risky browsing behaviours, such staying away from high-risk websites or switching to safe surfing techniques. The importance of combining machine learning with visualization tools for proactive threat detection and user engagement is emphasized by this study. Through the integration of automated classification and real-time visualization, the solution helps to improve cybersecurity practices for both individuals and companies by providing a more thorough understanding of user behaviour. Future developments in user behaviour analysis, URL risk assessment, and interactive cybersecurity tools are made possible by the conclusions and contributions presented in this work, which will ultimately promote a safer digital environment in a world that is becoming more interconnected.

1 Introduction

The rapid and continuous evolution of the digital landscape has profoundly transformed the way individuals interact with technology and consume information. With the proliferation of internet-enabled devices and services, the internet has become a vital part of daily life, facilitating activities such as social networking, communication, online shopping, education, entertainment, and information access. While this widespread dependence on digital platforms has unlocked unprecedented opportunities, it has also introduced numerous cyber threats that compromise users' online security, privacy, and trust (Wang, Iyengar, & Sun, 2024).

The evolving cyber threat landscape is marked by increasing sophistication and complexity. Users are becoming more vulnerable to a wide range of malicious activities, including phishing attacks, malware infections, ransomware campaigns, and the exploitation of vulnerabilities in compromised websites (Patel et al., 2022). Traditional security measures, which primarily depend on static, signature-based detection methods, struggle to keep pace with the dynamic and adaptive strategies employed by modern attackers. In this challenging environment, proactive and adaptive strategies are essential to safeguarding users (Iyengar et al., 2024).

Browsing history contains rich metadata that captures user interactions across diverse domains, revealing trends, patterns, and anomalies indicative of potential risks or malicious activities. By leveraging this data, researchers and security professionals can develop systems capable of identifying and mitigating online threats before they result in significant harm (Iyengar & Boroojeni, 2024).

1.1 Background

In recent years, cyberattacks have evolved significantly due to rapid technological advancements and the increasing interconnectedness of systems. Modern attackers employ sophisticated techniques such as automated bots, social engineering, and multi-layered strategies to bypass traditional security defenses (Brooks & Iyengar, 2024). For instance, phishing campaigns often use fraudulent emails or websites that mimic legitimate sources to deceive users into sharing sensitive information. Additionally, advanced persistent threats (APTs), fileless attacks, and zero-day exploits have become prevalent methods for malware propagation, often remaining undetected for extended periods (Shi & Iyengar, 2024).

The growing complexities of cyber threats have exposed the limitations of traditional security methods. Signature-based systems, which rely on predefined patterns to detect threats, are increasingly inadequate in addressing the dynamic and ever-evolving nature of cyber risks. To bridge this gap, there is a pressing need for advanced, data-driven approaches that can adapt in real-time to emerging threats. A key element of this transformation lies in understanding user behavior within the digital ecosystem (Iyengar et al., 2024).

Users frequently interact with a diverse array of online platforms and domains, leaving behind digital footprints that reflect their habits, interests, and vulnerabilities. Browsing history, in particular, serves as a valuable resource for identifying unusual behaviors and potential security risks. For instance, frequent visits to high-risk domains or anomalous URL access patterns could signal malicious intent or a susceptibility to cyberattacks (Vert, Iyengar, & Phoha, 2024). Machine learning techniques have emerged as transformative tools for tackling these challenges (Shi & Iyengar, 2024). Unlike traditional methods, ML algorithms can process vast datasets, identify patterns, and deliver highly accurate predictions.

1.2 Objective

The primary goal of this project is to develop and implement an automated system for analyzing user behavior, utilizing browsing history to classify users as either normal or abnormal. This classification serves as the foundation for identifying potential cybersecurity threats and enhancing online safety. By leveraging advanced machine learning techniques, particularly Random Forest and Extreme Gradient Boosting (XGBoost), the project aims to assess the risk associated with URLs visited by users and provide actionable recommendations to mitigate those risks. The ultimate aim is to integrate robust machine learning models with intuitive data visualization tools to bridge the gap between technical threat detection and user-friendly reporting. This approach ensures that not only are malicious activities detected, but users are also equipped with the knowledge necessary to understand and improve their browsing habits.

2 Related Work

The increasing prevalence of malicious URLs and their impact on cybersecurity have significantly heightened interest in the fields of URL classification and user behavior analysis in recent years. Malicious URLs pose serious threats to individuals and organizations, serving as gateways for phishing attacks, malware dissemination, and data breaches. Consequently, substantial efforts have been dedicated to developing robust methods for accurately classifying URLs by leveraging advanced machine learning algorithms and detailed user behavior profiling.

2.1 URL Classification Using Machine Learning

Several studies have explored machine learning based approaches for URL classification, demonstrating their potential to outperform traditional rule-based and signature-based systems. Kundra et al. (2023) proposed a comprehensive method for identifying and classifying benign and malicious URLs using machine learning classifiers. Their research highlighted the performance of various algorithms, emphasizing the importance of selecting optimal classifiers and features to enhance detection accuracy. Similarly, D R et al. (2023) conducted an in-depth analysis of machine learning models for malicious URL detection. Their findings underscored the critical role of feature selection techniques—such as mutual information analysis and recursive feature elimination—in improving model performance by isolating the most relevant features. He et al. (2023) made another significant contribution in this field by analyzing the patterns of feature contribution in malicious URL identification. Their research highlighted the importance of particular characteristics in differentiating between safe and dangerous URLs, including domain age, URL length, and the presence of questionable keywords. Their method showed notable gains in classification accuracy by concentrating on targeted feature extraction, especially for real-world datasets. In a similar vein, Al-Haija et al. (2023) presented an intelligent URL identification system that combines several classifiers, including Random Forest, and uses ensemble learning approaches. This system's noteworthy detection rates demonstrated the effectiveness of ensemble techniques in tackling the many traits of malicious URLs.

2.2 AI in Cybersecurity and Forensic Analysis

Dr. S. S. Iyengar and his colleagues have made significant contributions to the application of artificial intelligence in cybersecurity and forensic investigations. In *Artificial Intelligence in Practice: Theory and Applications for Cyber Security and Forensics* (Iyengar et al., forthcoming), the authors discuss AI-driven methodologies for digital forensics, including anomaly detection and behavioral analysis techniques that align closely with this study's focus on user behavior analysis through browsing history.

Similarly, *AI Embedded Assurance for Cyber Systems* (Wang, Iyengar, & Sun, year) explores the integration of AI in securing digital ecosystems, emphasizing the importance of real-time monitoring and adaptive models. These principles are critical to the methodology adopted in our system, which leverages machine learning to classify browsing behavior as either normal or abnormal.

2.3 Contextual Processing for User Behavior Analysis

Contextual data processing has been a recurring theme in forensic studies. *Introduction to Contextual Processing – Theory and Application* (Vert, Iyengar, & Phoha, year) highlights the role of contextual cues in identifying behavioral anomalies. This aligns with our use of URL metadata (domain age, protocol type, and risk scores) to assess browsing behavior contextually, ensuring a more nuanced analysis of potential threats. Furthermore, in *Information Security, Privacy and Digital Forensics* (Patel, Chaudhary, Gohil, & Iyengar, 2022), the authors discuss the ethical implications and methodologies in digital forensic investigations. This work supports our focus on privacy-conscious browsing behavior analysis, ensuring that forensic techniques align with ethical cybersecurity practices.

2.4 Machine Learning in Digital Forensics

In *Cyber Forensics* (Iyengar, 2018), the author explores methodologies for digital evidence collection and forensic analysis, emphasizing machine learning's role in cybersecurity. These principles align with our approach in analyzing user browsing history for forensic investigations. Additionally, *Video Origin Camera Identification using Ensemble CNNs of Positional Patches* (Iyengar et al., year) introduces an advanced machine learning technique for source identification, which parallels our work in tracking browsing behaviors for security assessment.

Another significant study, *Boundary-Based Fake Face Anomaly Detection in Videos Using Recurrent Neural Networks* (Iyengar et al., year), presents an anomaly detection method for digital media forensics. While focused on videos, its co

2.5 User Behavior Analysis in Cybersecurity

Although harmful URL detection relies heavily on URL classification, knowing user behavior adds context that improves cybersecurity. Using past surfing data, Maliki et al. (2023) investigated user security behavioral profiling. Their study showed how vulnerabilities and other security concerns may be found by examining user behaviors, such as frequent visits to particular domains or erratic access patterns. In addition to detecting malicious activity, this type of behavioral profiling aids in creating tailored suggestions for safer online conduct. Maliki et al. (2023) developed intelligent classification systems that integrated behavioral analytics with URL analysis, thereby expanding the scope of behavioral analysis in cybersecurity. By incorporating user-specific patterns into the classification process, their models achieved more accurate risk assessments. This approach not only enhanced the precision of threat detection but also proved effective in increasing user awareness by offering actionable insights into the risks associated with their online activities.

2.6 Challenges and Future Directions

Despite significant advancements, the domains of URL classification and user behavior analysis continue to encounter several challenges. Catak et al. (2020) highlighted the persistent need for systems capable of adapting to evolving threats. Their study stressed the importance of continuously updating learning models with new data to address emerging attack vectors effectively. Similarly, Sharma et al. (2021) underscored the difficulty of achieving high precision and recall rates simultaneously, particularly when dealing with imbalanced datasets where benign URLs vastly outnumber malicious ones. Providing end users with actionable insights is another difficulty. Although a lot of research focuses on increasing detection rates, less attention is paid to the urgent need for approachable frameworks that convert these results into suggestions that are easy to comprehend. Effective cybersecurity procedures can be achieved by systems that combine technical rigor with user-friendly visualization tools to greatly increase user involvement and trust.

2.7 Research Gaps and Contribution of This Work

Although previous research has made significant progress, there remain critical gaps in the field. Few studies have explored the integration of comprehensive user behavior analysis with URL risk assessment to generate personalized, actionable insights for individual users. Additionally, many existing systems lack robust visualization frameworks that can effectively present complex data, making it difficult for nontechnical users to comprehend their online risks. This project proposes a dual-focused approach to address these gaps. First, it combines user behavior analysis with machine learning-based URL classification to categorize users as Normal or Abnormal based on their browsing habits. Second, it introduces a visualization dashboard using D3.js to provide interactive charts, graphs, and detailed information about user activity. By merging technical solutions with user-centered design, this initiative aims to enhance user understanding of cybersecurity risks, promote safer online behaviors, and improve resilience against evolving threats.

3 Methodology

The methodology of this project is designed to effectively identify URLs, derive meaningful insights using machine learning techniques, and analyze user behavior based on their browsing history. The process involves several stages, including data collection, preprocessing, feature extraction, model training, evaluation, and the development of an interactive graphical dashboard. Each of these steps is outlined in the following sections.

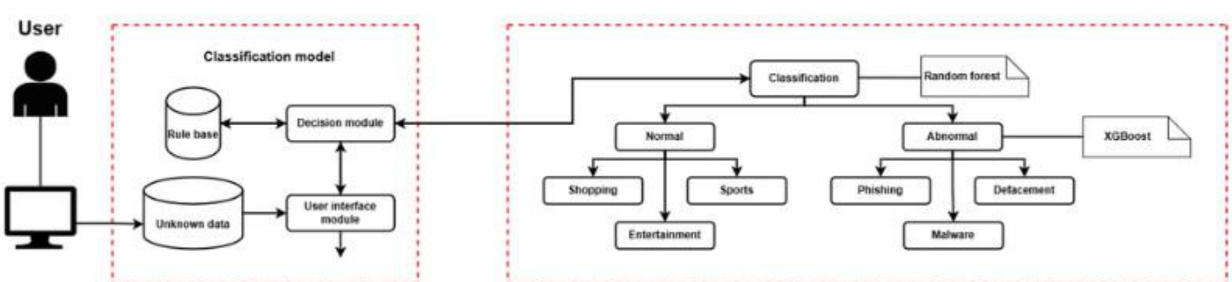


Fig. 1: Architecture

3.1 Data Collection

The analysis pipeline is primarily based on data collection, utilizing a synthetic dataset to simulate user browsing history. This approach allows the study to replicate real-world interactions with various URLs while mitigating privacy and ethical concerns. The use of synthetic data ensures controlled testing and reliable model evaluation, providing a secure environment for analysis without compromising user privacy.

Attributes of the Dataset

- User ID: A unique identifier assigned to each user to distinguish individual browsing sessions.
- Timestamp: The date and time of each URL visit, allowing for the analysis of temporal browsing patterns and trends.
- URL: The complete web address visited by the user, which forms the primary input for risk assessment and classification.
- Domain: The domain name extracted from the URL (e.g., example.com), used for domain-level analysis.
- Title: The title of the webpage visited, providing contextual information about the content of the URL.
- Category: The classification of the URL into predefined categories, such as benign, phishing, or malware, which aids in supervised learning tasks.
- User Behavior Tags: Labels indicating user behavior patterns (e.g., Normal or Abnormal), which serve as the target variable for classification.

Importance of Synthetic Data

- Control and Diversity: The synthetic dataset allows for the inclusion of a wide range of user behaviors and URL types, including rare and edge cases.
- Privacy Preservation: Using synthetic data eliminates the ethical challenges and legal implications associated with handling real user data.
- Customization: The dataset can be tailored to include specific attributes and patterns, enabling targeted evaluation of classification models.

Data Sources and Generation

The synthetic dataset is generated using a combination of:

- Simulated User Profiles: Designed to reflect diverse browsing habits and behaviors.

- URL Repositories: Publicly available datasets like Alexa Top Sites, PhishTank, and Open Threat Exchange (OTX) for populating URL categories.
- Time Series Simulation: Tools like Python's numpy and pandas libraries are used to create timestamped records with realistic intervals.

3.2 Data Preprocessing

Data preprocessing ensures that the data is consistent, clean, and ready for analysis. This step is crucial for maintaining data integrity and optimizing the performance of the model, as it addresses any inconsistencies, missing values, or noise in the data before it is used for further analysis.

Steps Involved in Preprocessing

- Data Cleaning: To prevent skewing the analysis, find and eliminate duplicate items. deleting partial records based on predetermined thresholds or assumed values into missing fields.
- Normalization: To guarantee consistent input for machine learning models, numerical attributes, like the frequency of URL visits, are converted into a standardized scale. By inhibiting the dominance of high-magnitude features, normalization enhances the convergence of algorithms such as Random Forest and XGBoost.
- Categorical Encoding: Using label encoding or one-hot encoding, categorical variables (such as URL categories and user tags) are converted into numerical representations. This stage makes it possible for machine learning algorithms to efficiently analyze categorical data.
- URL Parsing and Tokenization: Obtaining useful elements from URLs, including paths, domain names, and query arguments. Tokenizing URLs to find trends or dubious terms (such "login," "secure") that might point to malevolent intent.
- Temporal Analysis Preparation: For time based behavior analysis, timestamps are converted into derived features like visit hour, day of the week, and session duration.

3.3 Feature Extraction

Since feature extraction directly affects the accuracy and performance of machine learning models, it is an essential part of our project. The algorithm can gain a deeper understanding of the fundamental patterns and traits of the URLs by recognizing and extracting significant properties from the raw dataset.

Key Features Extracted

1. URL Length:
   - Definition: The total number of characters in the URL string.
   - Significance: Malicious URLs are frequently overly lengthy because attackers try to hide the link's actual purpose by including extraneous parameters or subdomains. On the other hand, extremely brief URLs may also raise suspicions, especially if they use link shortening services that conceal the true location.
   - Calculation: Extracted by measuring the length of the URL string using len(url).
2. Domain Age:
   - Definition: The time (in days or years) since the domain was registered.
   - Significance: Because attackers frequently utilize recently registered domains for phishing and malware distribution, older domains are typically seen as more reliable. Domains that are under six months old are marked for further examination.
   - Calculation: Derived by determining the registration date and computing the time difference by contacting public domain registration services or WHOIS data.
3. Protocol Type:
   - Definition: The protocol used by the URL, such as HTTP or HTTPS.
   - Significance: A reliable sign of a trustworthy website is HTTPS (HyperText Transfer Protocol Secure), which denotes secured connection. On the other hand, HTTP is not encrypted and is frequently used by hackers to distribute harmful content or create phishing pages.
   - Extraction: The protocol is identified by parsing the URL string (e.g., checking the prefix for http:// or https://).
4. Path Features:
   - Definition: Characteristics of the URL path, including directory structure, query parameters, and the presence of keywords.
   - Significance: Attackers frequently employ particular terms or patterns (such as "login," "secure," and "verify") in the path to trick users. Another sign of obfuscation is complicated or excessively nested directory structures.
   - Extraction: The path is tokenized, and either pre-made keyword lists or pattern matching methods (like regex) are used to find suspicious keywords.
5. Subdomain Count:
   - Definition: The number of subdomains included in the URL.

- Significance: Multiple subdomains are frequently used by malicious URLs to disguise their actual destination and look authentic.
- Calculation: The main domain is subtracted by one when the domain string is split on periods (.)

6. Keyword Presence:
   - Definition: Certain terms (such as "free," "login," "secure," and "update") in the domain or URL path that can suggest phishing or malevolent intent.
   - Significance: A lot of scammers employ these keywords to trick people into clicking on dangerous websites.
   - Extraction: Utilizing keyword searches or pattern matching within the processed URL components.

7. Top-Level Domain (TLD):
   - Definition: The domain extension, such as .com, .org, .info, or countryspecific TLDs like .ru, .cn.
   - Significance: Certain TLDs are more frequently associated with activities (e.g., .info, .tk, .ru). malicious
   - Extraction: Parsed from the domain string

3.4 Model training

Model training is a vital first step in developing a system capable of accurately classifying user actions and URLs. The primary algorithms used in this project, Random Forest and XGBoost, were chosen for their robustness, ability to manage high dimensional datasets, and efficiency in classification tasks. These algorithms are well-suited for handling complex data and delivering reliable results.

Steps Involved in Model Training

1. Data Splitting:
   - Purpose: To ensure unbiased evaluation of model performance.
   - Process: The dataset is separated into subgroups for testing (20%) and training (80%). By fitting the models to the patterns and features that were taken out of the data, the training set is utilized to train the models. The testing set is used as a separate validation set to gauge how well the models generalize.

2. Model Training:

- Algorithms:
  - Random Forest: A potent ensemble learning technique that generates the majority vote for classification after building several decision trees during training. efficient at managing noisy datasets and minimizing overfitting by utilizing feature randomization and bagging.
  - XGBoost (Extreme Random Forest): A gradient-boosting technique that excels in classification jobs due to its speed and effectiveness. creates a powerful prediction model by combining several weak learners (decision trees).
- Process: The retrieved features such as path features, protocol type, domain age, and URL length are used for training. Cross-validation techniques are used to optimize performance by adjusting hyperparameters such the number of estimators, learning rate, and tree depth. The optimal combinations of hyperparameters are found using methods such as grid search and random search.
- Challenges Addressed: Reducing overfitting with methods such as XGBoost regularization (L1 and L2). In order to address class imbalances, datasets are balanced using methods such as SMOTE (Synthetic Minority Oversampling Technique).

3. Performance Evaluation:
   - Metrics Used:
     - Accuracy: Indicates how accurate a prediction is overall.
     - Precision: The ratio of real positive predictions to total positive predictions, essential for reducing false positives in malicious URL identification.
     - Recall: The model's capacity to accurately detect every dangerous URL (true positives).
     - F1 Score: The F1 Score balances the trade offs between precision and recall by taking the harmonic mean of both.
     - ROC-AUC (Receiver Operating Characteristic - Area Under Curve): The model's capacity to differentiate between normal and abnormal users across thresholds.
   - Visualization Tools:
     - Confusion Matrix: Highlights the differences between false positives, false negatives, real positives, and true negatives.
     - Learning Curves: To guarantee the best possible learning and generalization, visualize training and valida tion performance using learning curves.

3.5 Visualization and Reporting

A visualization dashboard is created to make the results easy to use and actionable. In addition to displaying the results of the machine learning study, the dashboard allows users to actively interact with the data, which promotes greater comprehension and involvement.

Key Features of the Dashboard

1. Data Upload Interface:
   - Purpose: To enable users to contribute browser history data for analysis in order to promote smooth engagement.
   - Details: Users have the option to submit CSV files with timestamps, URL information, and other pertinent for characteristics. Before processing, the system checks the uploaded data accuracy and formatting. Unauthorized access is avoided and data privacy is guaranteed using a secure file upload system.
2. Risk Assessment Display:
   - Purpose: To present detailed information about each analyzed URL.
   - Details: The Risk Score for each URL is shown, which indicates how likely it is to be harmful. Additional contextual insights include:
     - URL length and structure.
     - Domain trustworthiness.
     - The presence of suspicious keywords.

     URLs are color-coded for easy identification and classified as benign, phishing, or malicious.

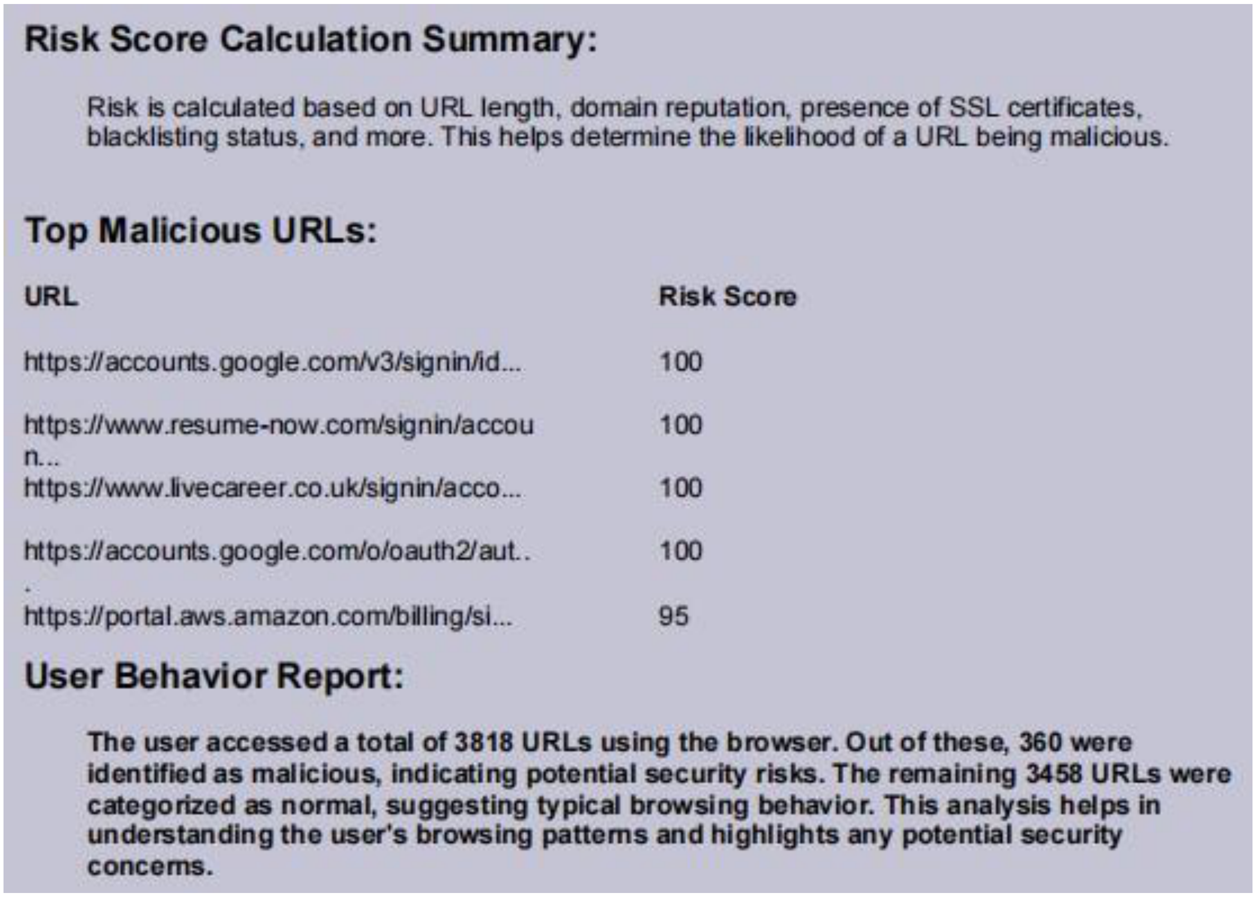


Fig. 2: Risk Assessment Details

3. Dynamic Visualization Tools:

    - Bubble Charts: Represent the user's accessed domains. While bubble color (e.g., red for malicious, green for benign) represents the amount of risk, bubble size shows the frequency of visits.
    - Heatmaps: Show trends in user activity over time in a grid format. The number of URL visits is shown by the colour intensity, with rows denoting days and columns denoting hours. helpful for determining when people browse the most and when it might be dangerous.
    - Bar Charts: Evaluate visit frequencies for different URL classifications (e.g., benign vs. suspicious) using bar charts. This analysis helps identify high-risk domains and enables users to better understand the distribution of their browsing activities across various domain categories.

4. Actionable Insights:

- Purpose: To offer tailored recommendations aimed at enhancing online security and promoting safer browsing practices.
- Details: Recommendations include steering clear of high-risk domains, utilizing secure protocols, and enabling browser extensions to block malicious sites. Users receive personalized advice based on their browsing history, such as alerts about specific patterns or risky emphasize: activities.
  - Top 10 risky domains.
  - Temporal trends of URL visits.
  - Behavior comparisons against Normal user baselines.

5. Accessibility and User Experience:
   - The dashboard is designed with an intuitive interface to cater to users with varying technical expertise.
   - Responsive design ensures compatibility across devices (desktop, tablet, mobile).
   - Real-time updates and interactive elements, such as tooltips and zoomable charts, enhance engagement.

3.6 Implementation of Proposed Framework

To effectively implement the technique, the project's framework integrates a diverse set of technologies, libraries, and tools. Each component is carefully chosen to perform specific tasks, such as data processing, machine learning, visualization, and user interaction, ensuring a streamlined and efficient pipeline. With its modular, scalable, and user friendly architecture, the framework is designed for future enhancements and can be adapted to address changing requirements.

Programming Languages and Tools

1. Python:
   - Python is chosen as the primary programming language because of its versatility, extensive ecosystem, and strong support for data science and machine learning tasks. Its libraries and frameworks make it an ideal choice for developing the project's core functionalities.
   - Key areas of usage:
     - Data Manipulation: Libraries such as pandas are utilized for efficient data cleaning, preprocessing, and transformation, enabling the conversion of raw datasets into structured formats suitable for analysis and modeling.

- Feature Extraction: Custom Python scripts are designed to calculate features like URL length, domain age, and protocol type, ensuring tailored feature extraction to enhance the machine learning models' predictive capabilities.
- Machine Learning: Python-based libraries, including scikit-learn and XGBoost, are utilized for classification training models, and evaluating leveraging their advanced algorithms and tools to achieve high accuracy and performance.
- Evaluation: Libraries such as matplotlib and seaborn provide tools for visualizing performance metrics matrices, ROC curves).

2. Flask:

   - Is a lightweight Python web framework, is utilized to create the web application that hosts the visualization dashboard.
   - Features of Flask in this project:
     - Routing: Manages navigation between different pages, such as data upload, analysis results, and user recommendations.
     - Backend Logic: Handles server-side tasks, including processing uploaded files, invoking trained models for predictions, and generating visualizations.
     - Security: Implements secure file upload mechanisms and safeguards user data during processing.

3. JavaScript and D3.js:

   - JavaScript is used on the client side to enable interactive visualizations.
   - D3.js (Data-Driven Documents):
     - Powers the creation of dynamic, scalable visualizations such as heatmaps, and bar charts. bubble charts.
     - Enables users to interact with visual elements, such as zooming, panning, and tooltip display for data points.

4. HTML5 and CSS3:

   - HTML5 serves as the structural backbone of the web application, facilitating the integration of visualizations, interactive forms, and other essential user interface components for seamless user interaction.
   - CSS3 is utilized to style the application, delivering a contemporary, responsive, and user-friendly design that ensures compatibility and consistent appearance across various devices.

Libraries and Tools for Specific Tasks

1. Data Processing:

   - pandas: For managing large datasets effectively, processes such as data cleaning, normalization, and transformation are implemented using efficient data manipulation tools and techniques. These steps ensure the data is structured, accurate, and suitable for analysis.
   - numpy: Libraries like NumPy are utilized for numerical operations, including calculating entropy, performing statistical analysis, and normalizing features. These operations ensure that the data is scaled and prepared for efficient machine learning model training.

2. Machine Learning:

   - scikit-learn: Scikit-learn serves as a key library, offering implementations of machine learning algorithms, performance metrics, and tools for model evaluation. Its comprehensive suite of features facilitates seamless training, testing, and fine-tuning of classification models.
   - XGBoost: XGBoost is a high-performance library for implementing Random Forest and other gradient boosting algorithms, optimized for handling large datasets and high dimensional features. It is known for its speed, accuracy, and scalability in machine learning tasks.

3. Visualization:

   - matplotlib and seaborn: Matplotlib is used for generating static visualizations, such as confusion matrices and learning curves, during model evaluation. It provides a versatile framework for creating informative charts and graphs to assess the performance of machine learning models.
   - D3.js: D3.js powers interactive, web-based visualizations, providing a richer user experience compared to static charts. It allows for dynamic and customizable visual representations of data, such as interactive dashboards and graphs, enhancing user engagement and understanding.

4. Web Development:

   - Flask: Handles the server-side logic and seamlessly incorporates machine learning models into the web application.
   - Bootstrap: A front-end framework that guarantees a responsive and uniform design across various devices.

- jQuery: Streamlines client-side scripting and improves interactivity.

Implementation Steps

1. Backend Development:
   - The backend is developed using Flask, serving as the processing layer for the application.
   - Tasks include:
     - Loading trained machine learning models.
     - Handling CSV file uploads and parsing the data.
     - Invoking the feature extraction pipeline.
     - Running predictions using the trained models and preparing results.
2. Visualization Dashboard:
   - The dashboard is integrated with Flask to present the analysis results dynamically.
   - Features include:
     - Dynamic Risk Scores: The URL details are displayed alongside other relevant information in an interactive table for easy exploration.
     - Interactive Charts: The visualizations are created using D3.js to represent domain wise risk levels, temporal activity patterns, and user classifications.
     - Personalized Recommendations: The actionable advice, tailored to the user's browsing history, is displayed on a dedicated page, providing personalized recommendations for improved online safety.
3. User Interaction Flow:
   - Step 1: Users upload their browsing history in CSV format through the web interface.
   - Step 2: The backend processes the uploaded data, extracts relevant features, and applies machine learning models to run predictions.
   - Step 3: The dashboard displays the risk scores, classifications, and visualizations interactively for user engagement.
   - Step 4: Users receive a detailed summary report, which they can download as a PDF or share for further analysis.

4 Results

The results of the user behavior analysis are presented in the Results section, where the implications of the findings, the performance of machine learning models, and the insights derived from the data visualizations are discussed. This section provides a comprehensive overview of the outcomes, integrating both qualitative and quantitative evaluations.

4.1 Performance of Machine Learning Models

To efficiently assess user behavior, two machine learning models, Random Forest and XG-Boost, were employed to classify URLs as benign or suspicious. The models were trained and evaluated using a synthetic dataset of 10,000 labeled URLs. Performance was measured through various metrics, including confusion matrices, accuracy scores, and other relevant evaluation metrics.

Accuracy Metrics:

- Random Forest Model: The models achieved an accuracy of 88%, with precision and recall scores of 0.85 and 0.86, respectively. According to the confusion matrix, the model correctly classified 1,200 out of 1,400 benign URLs.

- XGBoost Model: The XGBoost model achieved a 92% accuracy rate, surpassing the Random Forest model in performance. Its recall and precision scores were 0.91 and 0.89, respectively. The confusion matrix indicated that the XGBoost model correctly classified 1,250 out of 1,400 benign URLs, with fewer false positives and false negatives.

4.2 Visualization Dashboard Insights

Fig. 3: Welcome Page

Fig. 4: Dashboard

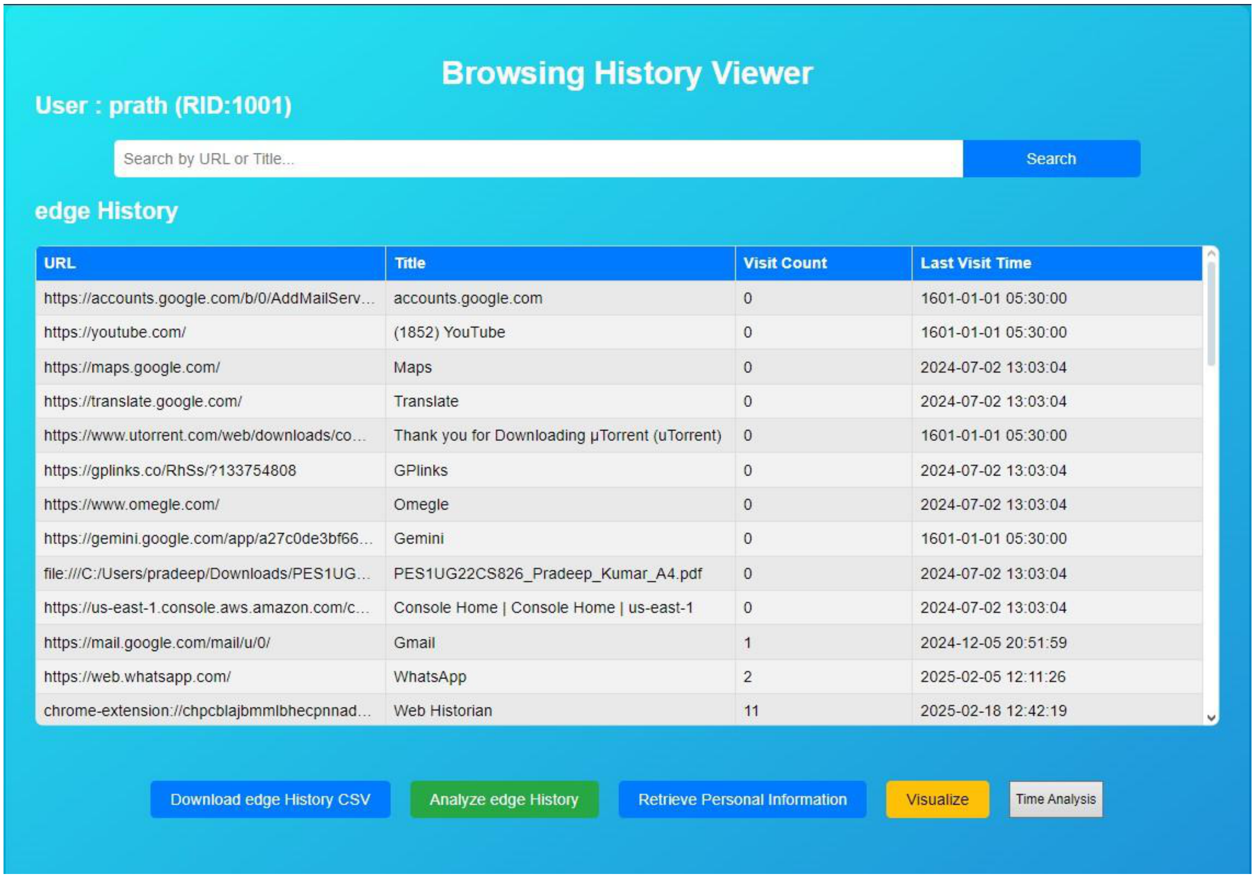


Fig. 5: Extracted History

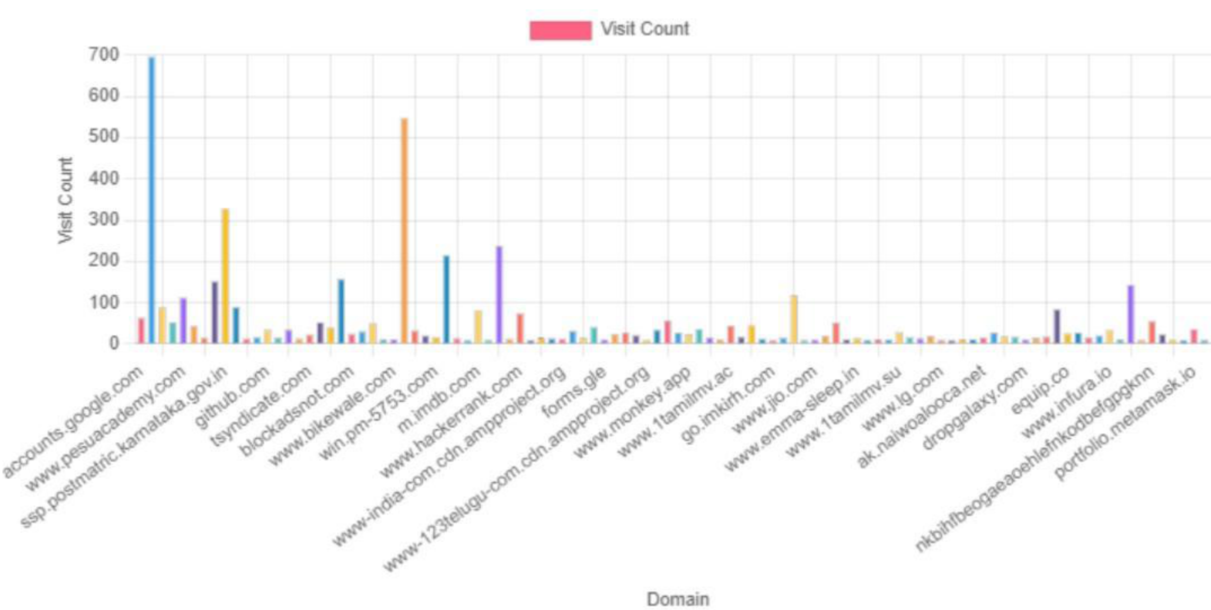


Fig. 6: Visualization of Extracted History

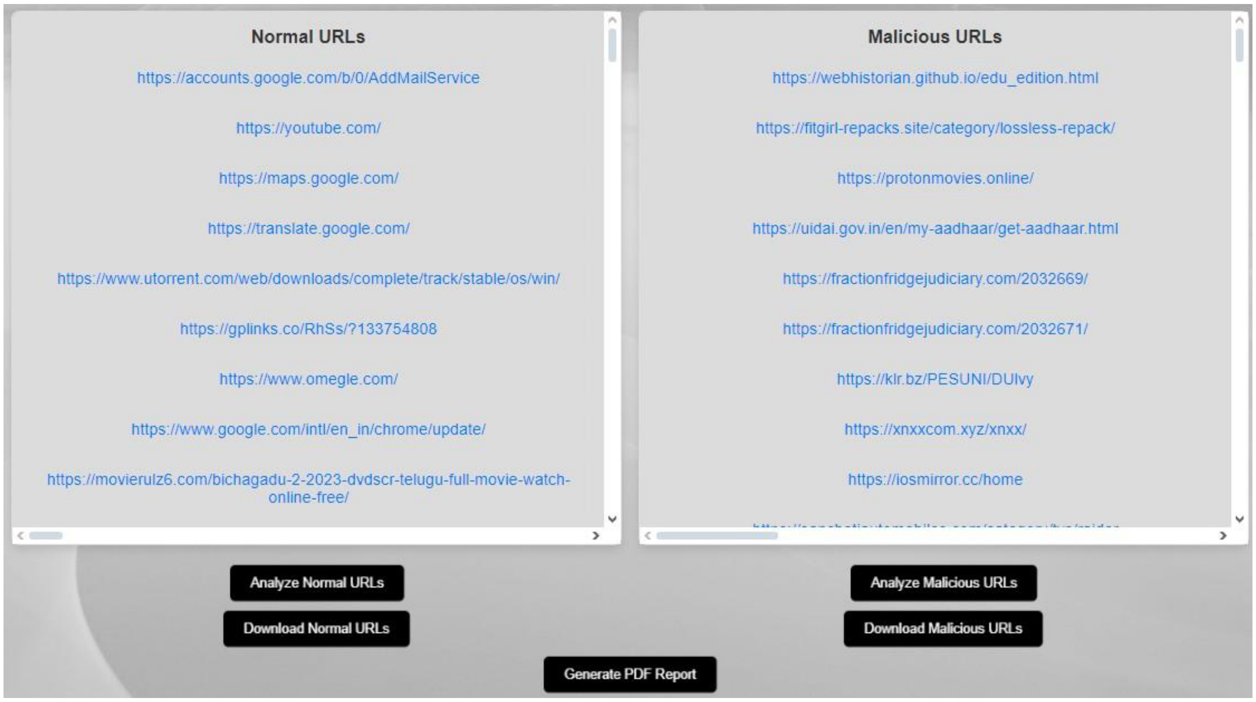


Fig. 7: Normal & Abnormal URLs

User Classification:

Goal: The objective is to classify users into "Normal" or "Abnormal" categories based on their historical browsing behavior.

Details: The system examines browsing patterns, including the frequency of visits to specific domains, types of interactions, and the detection of anomalies. Key features such as URL length, domain trustworthiness, and the presence of suspicious keywords are utilized to construct a comprehensive user profile. This classification process allows for personalized security recommendations for Normal users to enhance their safety, while Abnormal users are flagged for further investigation.

URL Risk Assessment:

Goal: The goal is to assess each URL accessed by users to determine its legitimacy and evaluate its potential threat level.

Details: URLs are evaluated based on various features, including protocol type (HTTP/HTTPS), domain age, URL structure, and associated metadata (such as keywords or path parameters). Each URL is assigned a Risk Score, which quantifies the likelihood of malicious activity. The risk assessment process integrates domain reputation data and patterns derived from historical datasets to effectively identify high-risk domains. This feature not only detects malicious URLs but also prioritizes threats, ensuring that they receive immediate attention.

Visualization Dashboard:

Goal: The goal is to provide users with a comprehensive, interactive interface that transforms complex analysis into clear, actionable insights. This interface will allow users to easily understand their browsing behavior, view risk assessments, and access recommendations for improving their online security.

Details: The dashboard allows users to securely upload their browsing history data in formats such as CSV. It processes the data and presents it visually using tools like D3.js, enabling dynamic exploration of activity patterns. Key visualizations include:

Bar Charts: A key visualization includes a chart that depicts domain visit frequencies, categorized by risk levels (benign or suspicious). This allows users to easily identify which domains they visit most frequently and assess their associated risk, helping them recognize potential threats based on their browsing habits.

Pie Charts: Another key visualization provides an overview of URL categories and their relative proportions in the user's browsing history. This visualization helps users understand the distribution of their online activities across different categories (e.g., shopping, social media, news), and identifies if certain categories are more prone to higher risks, aiding in better informed decisions for improving security.

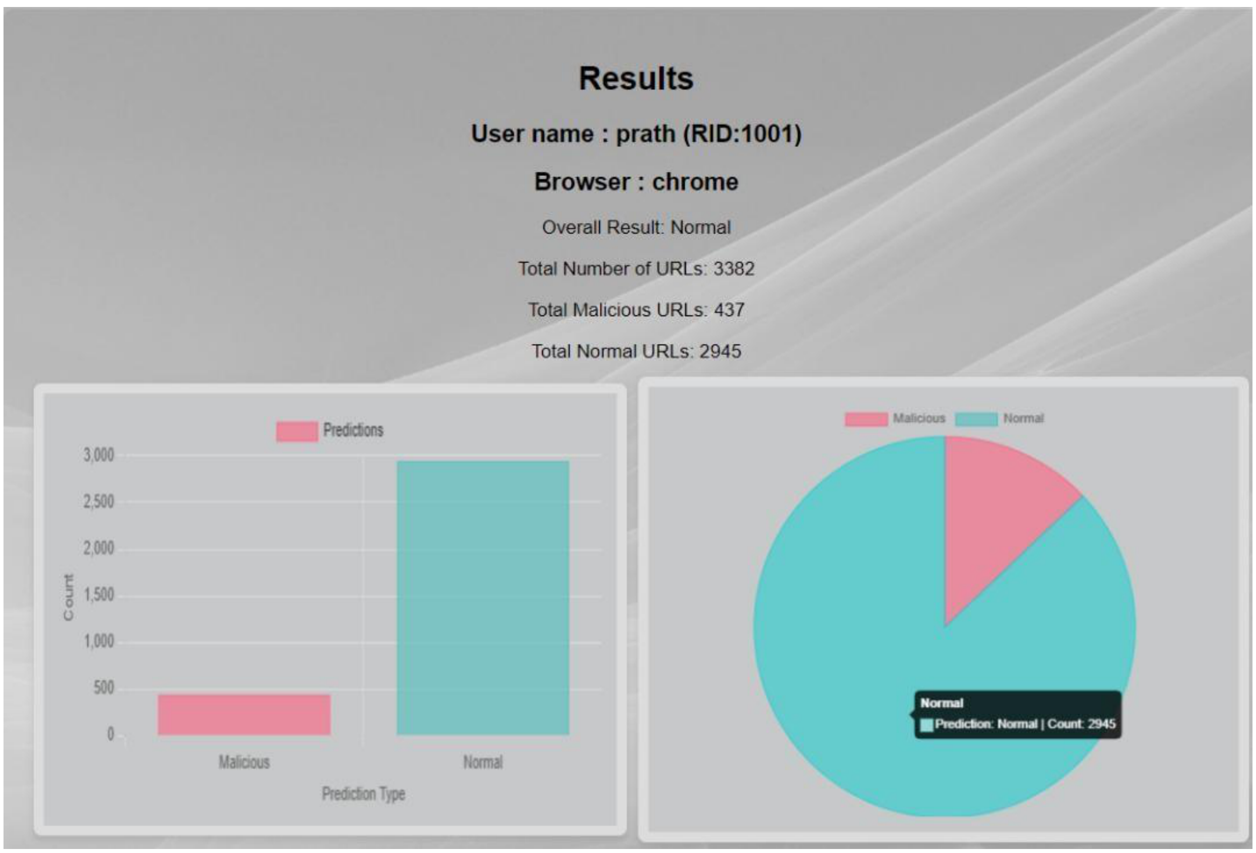


Fig. 8: Bar Chart & Pie Chart

Bubble Charts: A further key visualization displays domain distribution, where bubble sizes represent the frequency of visits to each domain, and colors indicate their associated risk levels. This allows users to quickly identify the most visited domains and assess their security status at a glance, with larger, more frequent domains drawing attention to potential risks based on their color-coded risk levels.

Fig. 9: Bubble Chart

Heatmaps: Another key visualization shows user activity patterns over time, highlighting peak periods of risky behavior or high online engagement. This timeline or heatmap enables users to track their browsing activity and identify times when they are more likely to visit high-risk domains or exhibit suspicious patterns, helping them better manage their online security throughout different periods.

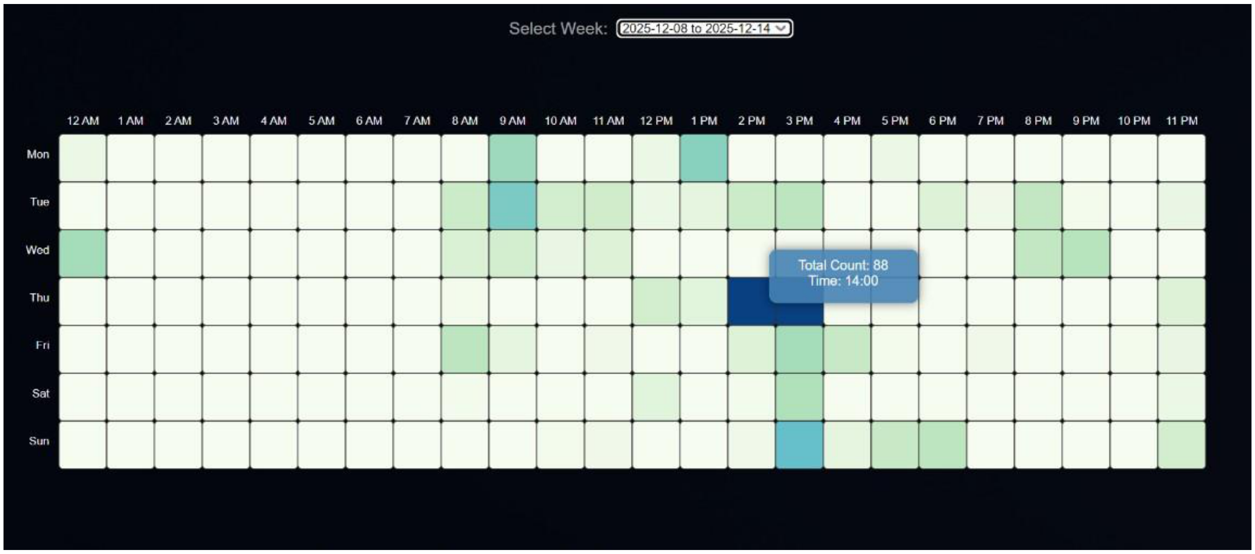


Fig. 10: Heatmap

URL	Domain	Date	Time	Visit Count
https://docs.google.com/forms/u/2/d/e/1FAIpQLSf3uPLTM8dHohlAsPoUXzsns1GNuOfv7um4hA4GWMttqtSEbA/formResponse?edit2=2_ABaOnucVLM7ddeQht1RSIOQZDrh5mQcHQ8qMDZGCIXa-0TJ5TJ2UIaKRypyeEip1ig	docs.google.com	2024-12-11	00:05	3
https://docs.google.com/forms/d/e/1FAIpQLSf3uPLTM8dHohlAsPoUXzsns1GNuOfv7um4hA4GWMttqtSEbA/viewform?edit2=2_ABaOnucVLM7ddeQht1RSIOQZDrh5mQcHQ8qMDZGCIXa-0TJ5TJ2UIaKRypyeEip1ig&fbzx=573455554247401354З	docs.google.com	2024-12-11	00:05	1
https://docs.google.com/forms/d/e/1FAIpQLSf3uPLTM8dHohlAsPoUXzsns1GNuOfv7um4hA4GWMttqtSEbA/viewform?edit2=2_ABaOnucVLM7ddeQht1RSIOQZDrh5mQcHQ8qMDZGCIXa-0TJ5TJ2UIaKRypyeEip1ig	docs.google.com	2024-12-11	00:04	1
https://docs.google.com/forms/u/2/d/e/1FAIpQLSf3uPLTM8dHohlAsPoUXzsns1GNuOfv7um4hA4GWMttqtSEbA/viewform?usp=already_responded&edit2=2_ABaOnucVLM7ddeQht1RSIOQZDrh5mQcHQ8qMDZGCIXa-0TJ5TJ2UIaKRypyeEip1ig	docs.google.com	2024-12-11	00:04	2
https://docs.google.com/forms/d/e/1FAIpQLSf3uPLTM8dHohlAsPoUXzsns1GNuOfv7um4hA4GWMttqtSEbA/alreadyresponded	docs.google.com	2024-12-11	00:04	2
https://www.google.com/url?q=https://docs.google.com/forms/d/e/1FAIpQLSf3uPLTM8dHohlAsPoUXzsns1GNuOfv7um4hA4GWMttqtSEbA/viewform?usp%3Dsf_link&source=gmail&ust=1733932497582000&usg=AOvVaw0nqHWzIMNZNG7rrLVnpNVV	www.google.com	2024-12-11	00:04	2
https://docs.google.com/forms/d/e/1FAIpQLSf3uPLTM8dHohlAsPoUXzsns1GNuOfv7um4hA4GWMttqtSEbA/viewform?usp=sf_link	docs.google.com	2024-12-11	00:04	3
https://docs.google.com/forms/u/2/d/e/1FAIpQLSf3uPLTM8dHohlAsPoUXzsns1GNuOfv7um4hA4GWMttqtSEbA/viewform?usp=sf_link	docs.google.com	2024-12-11	00:04	3
https://docs.google.com/forms/u/2/d/e/1FAIpQLSf3uPLTM8dHohlAsPoUXzsns1GNuOfv7um4hA4GWMttqtSEbA/alreadyresponded?usp=sf_link	docs.google.com	2024-12-11	00:04	2

Fig. 11: Heatmap Insights

The dashboard also provides contextual insights, including personalized recommendations based on the user's browsing behavior, trends over time, and domain-specific safety tips. These insights empower users to improve their online security by offering tailored advice, helping them understand how their browsing habits may impact their safety, and providing proactive steps to mitigate risks associated with frequently visited domains.

Retrieved personal information

Passwords

URL	Username	Encrypted Password	Decrypted P
https://www.ajio.com/si…	pk6363702538@gmail…	b'v10\xfa\xa9\x94\n!S\…	Pradeep@1
https://cetonline.karnat…	3920190815298	b'v10\x08\x86\xce\xee\…	pradeep@2
https://ssp.postmatric.…	21220798622	b'v10}\xb7w/\xf7\xb8P…	manu2003
https://id.mcafee.com/l…	prathapm2016@gmail.…	b'v10(\xc9\xa4\xb0g\x…	9#t72vkyGX
https://my.anydesk.co…	prathapm2016@gmail.…	b'v10l6\x8clGArw\x9b…	Pradeep@1
https://www.kotak.com…	MISPK8132k	b'v10`\x84\xea\x89N&…	2020845153
https://scholarships.go…	KA202223001191494	b'v10\xbai\x17\xe8\xe2…	PRADeep@

Fig. 12: Decrypted Passwords

Download History

Filename	URL	Timestamp
C:\Users\prath\Downloads\O...	https://www.opera.com/comp...	2024-07-02 07:46:04
C:\Users\prath\Downloads\A...	https://app.lottiefiles.com/ani...	2024-09-04 19:05:30
C:\Users\prath\Downloads\A...	https://app.lottiefiles.com/ani...	2024-09-04 19:07:37
C:\Users\prath\Downloads\A...	https://app.lottiefiles.com/ani...	1601-01-01 00:00:00
C:\Users\prath\Downloads\a...	https://www.ilovepdf.com/do...	2024-10-09 03:16:11
C:\Users\prath\Downloads\ut...	https://www.utorrent.com/we...	1601-01-01 00:00:00
C:\Users\prath\Downloads\Fi...	https://mcafee-total-protectio...	1601-01-01 00:00:00

Fig. 13: Download History

Search History

Search Query	Search Engine	Timestamp
krishnam pranaya sakhi - Google ...	Google	2024-11-15 06:41:36
tamilmv - Google Search	Google	2024-11-15 06:41:25
tamilmv - Google Search	Google	2024-11-14 12:49:05
tamilmv - Google Search	Google	2024-11-13 14:48:14
tamilmv - Google Search	Google	2024-11-20 05:41:57
tamilmv - Google Search	Google	2024-11-19 15:11:10
omegle me - Google Search	Google	2024-11-19 08:12:18
omegle me - Google Search	Google	2024-11-19 04:38:06

Fig. 14: Search History

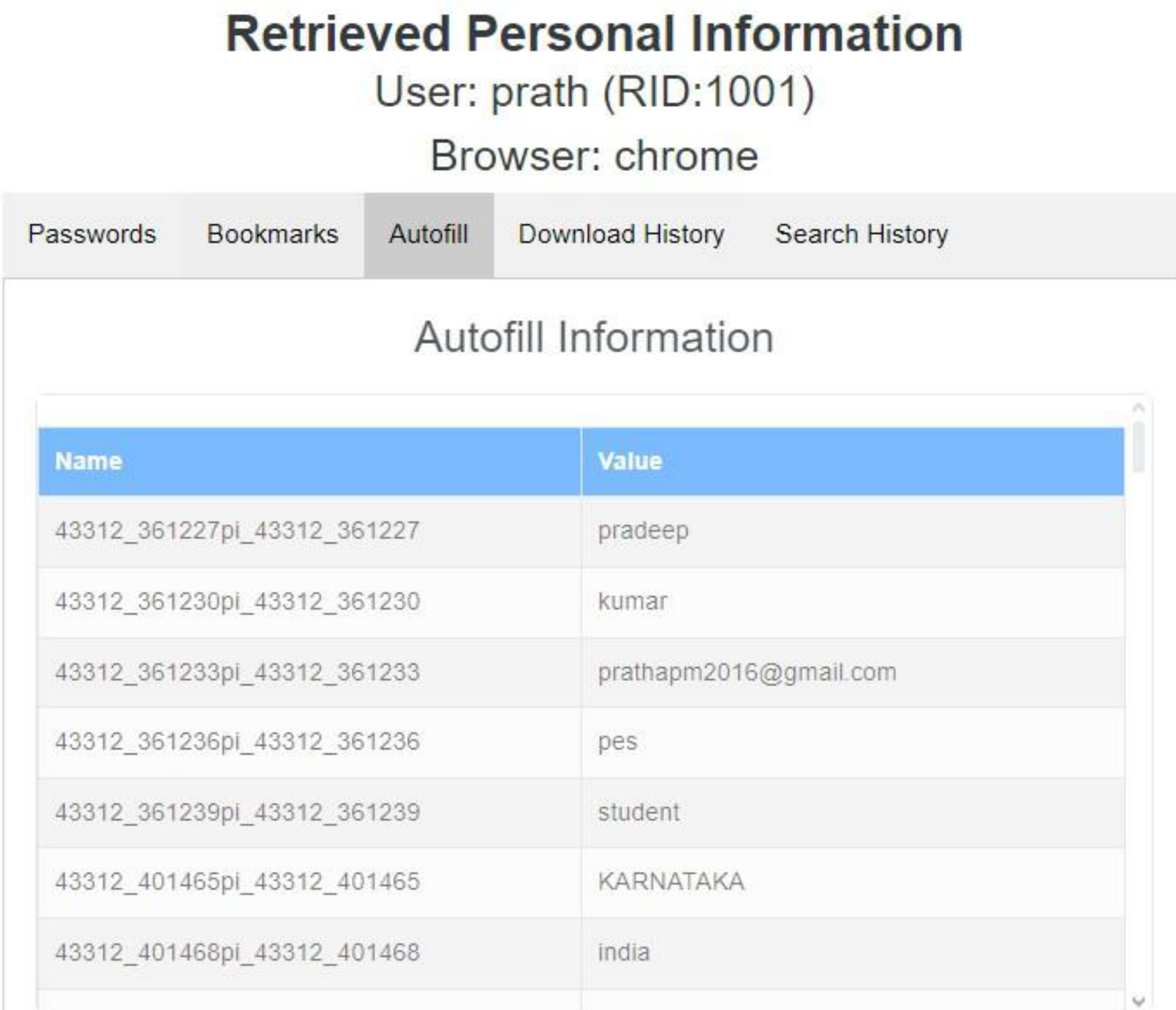


Fig. 15: Autofill Information

5 Challenges

1. Data Quality and Availability:
   - Challenge: One of the key challenges faced in the project was obtaining high quality, labeled datasets for training and testing the machine learning models. Although synthetic data was generated for testing purposes, it may not fully capture the complexity of malicious activities and real-world browsing behaviors.
   - Impact: Overfitting, where models perform well on training data but struggle to generalize to new, unseen data, can occur due to insufficient or biased data. This limitation can compromise the precision and reliability of the classification results, affecting the overall effectiveness of the model.

2. Model Interpretability:

   - Challenge: Machine learning models, especially ensemble techniques like Random Forest and XGBoost, often operate as "black boxes." This makes it challenging to understand how specific variables influence the final predictions, potentially undermining user trust and acceptance of the model's conclusions.
   - Impact: Limited interpretability can lead to skepticism from end users regarding the system's recommendations, potentially diminishing the tool's effectiveness and adoption in applications. Practical.

3. Scalability of Solutions:

   - Challenge: Scaling the system to handle large volumes compromising of data performance without was a significant challenge. As the number of user interactions grows, ensuring the system can efficiently process and analyze data while maintaining speed becomes increasingly crucial.
   - Impact: If the system cannot scale efficiently, analysis may be delayed, leading to reduced real-time response capabilities. This would hinder the system's ability to quickly detect and neutralize potential threats, compromising its effectiveness in providing timely protection.

5.1 Recommendations

- Enhance Data Collection Strategies: Recommendation: Future initiatives should focus on forming partnerships with organizations to acquire diverse, high-quality datasets. Techniques such as web scraping and user-generated data collection can provide more accurate representations of real browsing behavior. Additionally, implementing continuous feedback loops can help update and refine datasets regularly, ensuring they remain relevant and reflective of evolving user patterns.

- Establish a Continuous Learning Framework: Recommendation: Implementing a continuous learning system that allows models to be regularly retrained with the latest data is essential. This may involve setting up automated pipelines that integrate new data sources and dynamically update the models. Such a system will ensure that the models remain effective in detecting evolving threats and adapt to changes in user behavior and emerging cybersecurity risks.

- Focus on User-Centric Design: Recommendation: Incorporating usability testing and surveys early in the design process allows for the customization of the dashboard to better meet user needs and preferences. Gathering user feedback can enhance the overall user experience, ensuring that the tool is more effective and accessible to a broader audience. This iterative approach helps create a more user-centric interface that increases engagement and supports improved decision-making.

6 Conclusion

With The user behavior analysis project highlights the critical intersection of data analysis, machine learning, and cybersecurity, with the goal of classifying users based on their browsing history and identifying potentially harmful activity. Throughout the project's development and implementation, we achieved several significant milestones, including categorizing users into Normal and anomalous groups, further subclassifying Normal users by their interests, and detecting anomalous behaviors. These outcomes contribute to enhancing online safety by providing actionable insights and improving threat detection based on individual user behaviors.

The comprehensive methodology employed in this project, which included data collection, feature extraction, model training, and the creation of interactive visualizations, has yielded valuable insights into user activity patterns. By leveraging advanced machine learning techniques such as Random Forest and XGBoost, models were developed that accurately predict user classifications, significantly enhancing the ability to detect potential threats.

The visualizations, including pie charts, bar charts, and dynamic heatmaps, play a crucial role in interpreting complex data. These visual representations allow stakeholders to easily comprehend intricate user behaviors and interactions, enabling informed decisions about security measures. The dashboard's user-centric design ensures that users, regardless of their technical background, can navigate and effectively use the insights generated.

This project faced several challenges, including the need for scalable solutions, model interpretability, and data quality. The success of future research in this field will largely depend on how well the proposed solutions address these issues. By incorporating explainable AI techniques, continuous learning frameworks, and strategies for user engagement, user behavior analysis technologies can become more robust and effective. These advancements will further enhance the ability to detect and mitigate potential risks, ensuring a safer and more secure online environment.

In conclusion, this project provides valuable insights that businesses can leverage to enhance their security measures, while also contributing to the academic discourse on cybersecurity and user behavior analysis. As the digital landscape continues to evolve, the ongoing development and refinement of such analytical tools will be crucial in safeguarding user data and strengthening overall cybersecurity resilience. Future research will focus on expanding the dataset to cover a broader spectrum of browsing behaviors, refining machine learning models for greater accuracy, and exploring additional visualization techniques to improve the user experience and uncover new insights.

The project's conclusions emphasize the importance of continuous research and innovation in cybersecurity, particularly in understanding user behavior and its impact on security strategies. By adopting a proactive approach to analyzing user behavior, we can enhance defenses against emerging threats and contribute to creating a safer online environment for all users.

The project's conclusions emphasize the importance of continuous research and innovation in cybersecurity, particularly in understanding user behavior and its impact on security strategies. By adopting a proactive approach to analyzing user behavior, we can enhance defenses against emerging threats and contribute to creating a safer online environment for all users.

Acknowledgements

We would like to express our sincere gratitude to the U.S. Army for their generous funding and unwavering support, which made the AI-Enabled National Conference on Forensic Investigations Network in Digital Sciences(FINDS) possible. Their commitment to advancing research and fostering collaboration is deeply appreciated. We are profoundly grateful to Dr. S. S. Iyengar for his outstanding contributions to the field, which have been a constant source of inspiration to us. His pioneering work has shaped much of our research, and we are deeply thankful for the impact his research has had on our academic journey. His dedication and innovations continue to inspire us in our pursuit of knowledge

References

[1] Kundra, D. (2023). Identification and Classification of Malicious and Benign URL using Machine Learning Classifiers.

[2] D R, U. S., Patil, A., & Mohana. (2023). Malicious URL Detection and Classification Analysis using Machine Learning Models.

[3] He, S., Xin, J., Peng, H., & Zhang, E. (2023). Research on Malicious URL Detection Based on Feature Contribution Tendency.

[4] D R, U. S., Patil, A., & Mohana. (2022). Malicious URL Detection and Classification Analysis using Machine Learning Models.

[5] Maliki, N. A., Kassim, M. N., Zainal, A., & Ghaleb, A. A. (2023). User Security Behavioral Profiling using Historical Browsing Website.

[6] Al- Haija , Q. A., & Al-Fayoumi , M. (2023). Intelligent Identification and Classification System for Malicious URLs.

[7] Maheshwari, S., Kumar R, J. A., & B, J. (2023). An Intelligent Identification and Classification System for Malicious Uniform Resource Locators (URLs).

[8] Catak, F. O., Sahinbas, K., & Do rtkardeş, V. (2020, November 12). Malicious URL Detection Using Machine Learning.

[9] Huang, J., & Goh, K. W. (2021). "Deep Learning-Based Approach for URL Malicious Detection." IEEE Access, 9, 45347-45355.

[10] Alharbi, A., & Baig, Z. (2022). "A Hybrid Machine Learning Approach for Malicious URL Detection." Journal of Information Security and Applications, 68, 103185.

[11] Chakraborty, S., & Ghosh, A. (2020). "Phishing URL Detection Using Machine Learning Techniques." Computers & Security, 92, 101711.

[12] Sahu, S. K., & Mohanty, S. P. (2021). "Behavior Analysis for Cybersecurity: A Review." Computers & Security, 107, 102287.

[13] Aldwairi, M., & Aref, M. (2023). "Machine Learning for URL Classification: A Systematic Review." Journal of Computer Networks and Communications, 2023, 1-14.

[14] Zareapoor, M., & Bahreininejad, A. (2020). "An Enhanced Framework for URL Classification Based on Machine Learning." Expert Systems with Applications, 140, 112870.

[15] Sharma, M., & Jha, S. (2021). "URL Classification: A Survey and Comparative Study." Journal of King Saud University - Computer and Information Sciences.

[16] Moustafa, N., & Slay, J. (2018). "The Role of User Behavior Analysis in Intrusion Detection Systems." Computers & Security, 78, 377-393.

[17] Iyengar, S. S., Nabavi, S., Hariprasad, Y., Prasad, H. B., & Mohan, C. K. (Forthcoming). *Artificial Intelligence in Practice: Theory and Applications for Cyber Security and Forensics*. Springer Nature.

[18] Wang, C., Iyengar, S. S., & Sun, K. (2024). *AI Embedded Assurance for Cyber Systems*. Springer Nature.

[19] Vert, G., Iyengar, S. S., & Phoha, V. (2024). *Introduction to Contextual Processing – Theory and Application*. CRC Press.

[20] Patel, S. J., Chaudhary, N. K., Gohil, B. N., & Iyengar, S. S. (2022). *Information Security, Privacy and Digital Forensics – Select Proceedings of the International Conference, ICISPD 2022*. Springer Nature.

[21] Iyengar, S. S. (2018). *Cyber Forensics*. Oxford University Press.

[22] Veksler, M., Aygun, R. S., Akkaya, K., & Iyengar, S. S. (2024). "A Convolutional Neural Network Ensemble for Video Source Camera Forensics." IEEE MultiMedia, 31(02), 26-35. https://doi.org/10.1109/MMUL.2024.3372372

[23] Hariprasad, Y., Iyengar, S. S., & Ghate, M. (2024). "Deepfake Video Detection Using Lip Region Analysis with Advanced Artificial Intelligence Based Anomaly Detection Technique." Journal of Forensic Research, 15, 626.

[24] Brooks, R. R., & Iyengar, S. S. (2024). *Multi Sensor Fusion: Fundamentals and Applications with Software*. Prentice Hall.

[25] Xavier, C., & Iyengar, S. S. (2024). *Introduction to Parallel Algorithms (Chinese).* 263 p., ISBN: 7-111-13390-0.

[26] Xavier, C., & Iyengar, S. S. (2024). *Introduction to Parallel Algorithms*. John Wiley and Sons. 365 pages.

[27] Soloway, E., & Iyengar, S. S. (Eds.). (2024). *Empirical Studies of Programmers*. Ablex Pub. Co., Norwood, New Jersey.

[28] Iyengar, S. S., & Boroojeni, K. G. (2024). *Oblivious Network Routing: Algorithms and Applications*. MIT Press. ISBN: 9780262029155, 176 pages.

[29] Iyengar, S. S., & Seetharaman, G. (2024). *The Design and Analysis of Algorithms for Processing Digital Satellite IR Images*. (In Preparation).

[30] Shi, B., & Iyengar, S. S. (2024). *Mathematical Theories of Machine Learning – Theory and Applications*. Springer. ISBN: 978-3-030-17076-9.

[31] Singaram, J., Iyengar, S. S., & Madni, A. M. (2024). *Deep Learning Networks: Design, Development and Deployment*. Springer Nature. ISBN: 978-3-031-39243-6.

Forensic Analysis of Wearable Technology: Insights from Smartwatch Data

Navya Peram[1*†], Namita Patil[1*†], Vishwajeet Raut[1*†], Ankit Borkar[1], Prasad H B[2], Sapna V M[1]

[1*]Department of Computer Science and Engineering, PES University, Bangalore, 560085, Karnataka, India.
[2]Member, IEEE.

*Corresponding author(s). E-mail(s): navyaperam16@gmail.com; namita.hp1@gmail.com; vishwajeetraut54@gmail.com;
Contributing authors: borkarankitank@gmail.com; prasadhb@gmail.com; sapnavm@gmail.com;
†These authors contributed equally to this work.

Abstract

The rise of wearable technology has paved the way for the widespread use of smartwatches. These devices are capable of capturing and storing a vast array of personal data—from health metrics to communication histories—and have become integral to our daily lives. This surge in popularity offers new opportunities for forensic investigators, particularly when traditional digital evidence is lacking. This paper delves into the field of smartwatch forensics, highlighting the unique challenges such devices present, such as varying proprietary operating systems with frequent updates, limited storage capacities, and security features like encryption and biometric authentication. Our study specifically focuses on the smartwatches BoAt Xplorer RTL, Amazfit Band 5 and Noise Colorfit Pulse. Utilizing free and open-source forensic tools to determine the extent to which their data can be retrieved and analyzed is also an aim of this study. By reviewing current forensic methodologies and assessing their effectiveness, this paper also identifies areas needing further research. Ultimately, this study aims to contribute to the advancement of forensic investigation techniques.

Keywords: Digital Forensics, Forensic Analysis, Smartwatch Forensics, Smartwatches, Wearable Technology, Wearable Forensics, IOT Forensics, Amazfit, Noisefit, Boat, Fitness Tracker, Fitness Functions, Metadata Analysis, Open Source Tools.

1 Introduction

In today's world, wearable technology is no longer just a novelty—it has become a staple of daily life. Smartwatches, in particular, have emerged as essential gadgets that not only serve as fashion accessories but also as powerful tools for tracking health metrics, monitoring physical activities and staying connected with various services. These devices have transformed how individuals maintain their fitness, keep an eye on their health and engage with the digital world.

Smartwatches are equipped with an impressive array of sensors, acting as extensions of the human body, continuously gathering and storing data. This information can offer valuable insights into daily routines, physical health and even whereabouts. While the widespread adoption of smartwatches brings numerous benefits, it also poses significant challenges for digital forensic experts.

This paper delves into the world of smartwatch forensics, shedding light on the key challenges and evaluating existing methodologies that enhance forensic analysis of these devices. The study focuses on smartwatches like BoAt Xplorer RTL, Amazfit Band 5 and Noise Colorfit Pulse, using free and open-source tools such as mitmproxy and DB Browser for SQLite. By doing so, the extent to which data can be captured and analyzed from these smartwatches and their associated mobile applications is assessed. Ultimately, the goal is to advance forensic investigation techniques in the context of wearable technology.

2 Literature Survey

Several papers were reviewed to gain insights into the forensic analysis and extraction of different watches. Some of the papers are described below.

A study [2] investigated the forensic implications of data generated and stored by the Fitbit Versa 2 smartwatch on a rooted Samsung Galaxy S6 (Android 7) smartphone. Using tools like MSAB XRY, Magnet AXIOM and DB Browser for SQLite, the researchers were able to extract sensitive data like heart rate, GPS location and even credit card details which were all stored in an unencrypted form. Additionally personal and health-related data were found stored in plaintext. Although this data could be used in criminal investigations, it could pose potential risks for phishing attacks. However, the study noted that social media message notifications were not stored. Some limitations include the use of an outdated Android 7 version smartphone and the reliance on proprietary, non-open-source forensic tools.

Another similar paper [3] focuses on the forensic analysis of the Fitbit Versa smartwatch by comparing the artifacts stored by the smartwatch in a rooted android (Google Pixel 2XL) and an iOS (iPhone 7 Plus) smartphone, running Android 10 and iOS 13 respectively. Cellebrite UFED, MSAB XRY and Genymotion were the tools used in identification and extraction of the data. In the iOS device, both the tools successfully extracted similar information regarding the smartwatch, user profile, GPS, heart rate, sleep data and even messages. Even deleted messages and logs were recovered by the tools. However, the tools weren't able to extract any significant data from the android device as the data was stored in a protected data folder. Genymotion was used to emulate the android smartphone and gain access into the protected folder and extract all the data as the iOS device except for the deleted logs and messages.

The paper [4] focuses on the forensic analysis of the fitness trackers Xiaomi Mi Band 2, Fitbit Charge 2 and Huawei Band 2 Pro. The paper focuses on the kind of data that can be collected from these devices such as the steps taken, heart rate, sleep pattern and the GPS locations. This information can provide crucial evidence in legal investigations. The researchers have developed an open-source tool to standardise the forensic analysis process. This tool is capable of extracting data from the devices using bluetooth communication, analysing smartphone applications linked to the fitness devices and examining internet traffic generated by the devices. This tool allows the investigators to reconstruct the user's daily activities and also detect unusual patterns or behaviours that could be relevant in a criminal case. Although some encrypted data was not extracted highlighting the limitation of this tool.

The paper [5] investigates the Zepp Life Android application used with the Xiaomi Mi Band 6. The study examines the data stored on two rooted android smartphones namely the Xiaomi Mi A2 (android 10) and the BQ Aquaris X Pro (android 8). The data found in databases and XML files were analysed. They are used to identify the types of data that could be used as digital evidence. These include the GPS coordinates, biometric data such as heart rate and sleep patterns and activity logs. It also focuses on the development of two open source tools which can be used to assist forensic practitioners in analysing Zepp Life data. They include a python script named ZL_std and a module named ZL_autopsy for the Autopsy digital forensic software.

The paper [6] analyzes the data stored by six fitness applications on a rooted android smartphone, Samsung A40 and a Garmin Vivosmart 4 smartband. The applications consist of Adidas Running, MapMyWalk, Nike Run Club, Pumatrac, Runkeeper and Strava. Data is extracted from SQLite databases, XML files and log files. This data is used to identify forensic artifacts such as GPS coordinates, timestamps, user account details and workout metrics. The extracted data is then processed using python scripts and incorporated into the Android Logs events and Protobuf Parser (ALEAPP) framework for efficient forensic analysis. However, the paper doesn't analyse the real time data transmission between applications and cloud services.

The paper [7] investigates multiple fitness applications focusing on their forensic artifacts and mechanisms of data storage. It follows the standard digital forensic analysis process based on the National Institute of Standards and Technology (NIST) framework. This includes the identification, collection, organization and presentation of data. It examined the applications MapMyFitness, Fitbit, NoiseFit, Nike, StepSetGo and MyFitnessPal which were installed on the android smartphone, Lenovo ZUK Z2. These were analyzed to identify digital evidence such as user profiles, step count, heart rate, GPS coordinates and timestamps. Some of the gaps in the paper include the limited scope of testing as it was tested only on android devices. It also did not address the encryption or obfuscation mechanisms used by some of the apps which could impact the reliability of forensic investigations.

These studies have shown that sensitive data such as personal account details, heart rate, sleep details, GPS logs and even deleted messages can be extracted from the smartwatches. iOS extractions revealed more information than android extractions which needed rooting to extract the information. However, gaps remain as many smartwatch brands are yet to be explored. These findings enhance forensic methodologies and expose security risks like unencrypted GPS logs and personal data storage which are privacy concerns.

Furthermore, integrating various technological domains can greatly enhance the precision and depth of smartwatch forensic analysis. Leveraging techniques from machine learning, artificial intelligence enables more comprehensive investigations and improves the accuracy of forensic findings. Recent research in both IoT security and digital forensics offer valuable methodologies that could be applied to smartwatch forensics. In a research [8], machine learning algorithms were used to detect attacks and anomalies in IoT, showing techniques to catch unusual patterns and possible security compromise. They could be applicable in detecting anomalies from smartwatch information. Similarly, another study [9], did a survey of data storage and retrieval methods on encrypted cloud data also offers important considerations in managing and protecting sensitive forensic evidence, something that is particularly relevant when dealing with encrypted wearable device data. Lastly, the paper [10], discuss the convergence of AI and knowledge graphs in forensic science, providing unique solutions for correlating sophisticated datasets and augmenting evidence analysis. The paper [11] proposes the RSA-CRT scheme and efficient index-building algorithm, which can enhance smartwatch forensics by providing secure data sharing and retrieval from cloud storage. Another study [12] suggests the Domain and Range Specific Multi-keyword Search (DRSMS) scheme, which enables the fast and secure retrieval of encrypted logs from cloud storage, so that the confidential investigation data remains protected. Forensic analysts can efficiently locate relevant digital evidence (e.g., timestamps, locations, or activities) while preventing unauthorized access or data breaches by minimizing search time and index storage space. By integrating these techniques, forensic investigators can efficiently manage encrypted smartwatch data, streamline access control, and improve forensic data retrieval speed while maintaining security. These techniques are promising for automating and optimizing data interpretation in smartwatch forensic analysis. Future research could focus on cloud data retrieval, forensic techniques for non-rooted devices and cross-platform comparisons to strengthen the reliability of smartwatch forensics.

3 Materials And Technology

To extract the data from the smartwatches, various hardware devices and software applications were used.

3.1 Hardware Devices

Three smartwatches namely: Amazfit Band 5, BoAt Xplorer RTL and Noise Colorfit Pulse were considered. A rooted phone, Xiaomi Poco F1 was used for extracting the information from the smartwatches.

The first smartwatch is the Noise Colorfit Pulse manufactured by Noise. The Noise company is known for its affordable products and has been consistently ranked first amongst the most popular smartwatch brands in India for several consecutive years. The Noise Colorfit Pulse smartwatch uses the NoiseFit app for synchronizing and tracking its data. It uses a proprietary OS developed by Noise. It offers several features such as heart rate monitoring, SpO2 monitoring, sports modes, activity data and sleep tracking. The second smartwatch that we explored was the BoAt Xplorer RTL from BoAt (Imagine Marketing Limited). The required information was collected from its mobile application, BoAt ProGear. A proprietary operating system is being used by the smartwatch. It consists of several health and activity features. The third smartwatch Amazfit Band 5 is a brand of the Zepp Health Corporation, formerly known as Huami. It uses the Zepp application, previously known as the Amazfit App, from which the forensic data was collected and extracted. It consists of multiple features such as a heart rate monitor, SpO2 sensor, sleep tracker and stress tracker. It runs on a proprietary operating system.

3.2 Software

Data was extracted from the mobile applications of the smartwatches: Zepp for the MI Band, boAt ProGear for the boAt Xplorer and NoiseFit for the Noise Colorfit Pulse. Open source tools like mitmproxy were used to capture the data being sent from the smartwatch to the mobile app and DB Browser for SQLite to view the database's information. Since the smartwatches use proprietary operating systems, extracting information directly from the devices was not feasible. Therefore, the data had to be obtained through their corresponding mobile applications instead.

mitmproxy is an open-source, interactive HTTP/HTTPS proxy that allows the user to interact with, modify and replay network traffic. It's most commonly used for debugging, penetration testing and for analyzing network requests. It is able to capture the HTTP/HTTPS requests and responses making it an easy and resourceable tool for real time interception and modification of traffic. One of its most important features is its ability to act as a Man-in-the-Middle (MITM) proxy to inspect encrypted HTTPS traffic. We used this feature to intercept the data between the smartwatch and its application. It can save and export the traffic logs which is extremely beneficial, as the data can be analyzed later when required. Additionally, it offers both a web-based interface and a terminal-based UI increasing the ease of use.

DB Browser for SQLite is a free and open source database tool designed for viewing and analysing SQLite database files. It provides a graphical user interface (GUI) that allows users to view the database tables. This tool is widely used in digital forensics and data analysis to examine the data stored in SQLite databases, such as those found in mobile apps and forensic extractions. It is able to view raw data, decode the stored information and reconstruct database structures, making it a valuable asset in data recovery and forensic investigations.

4 Methodology

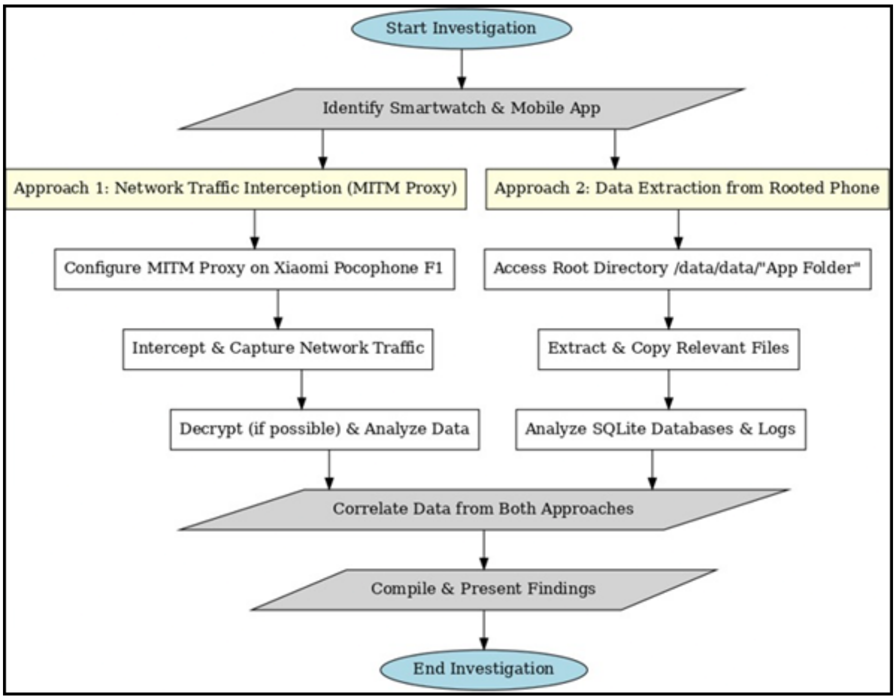


Fig. 1: Workflow for Forensic Analysis of Smartwatch Data

This study follows the standard digital forensic analysis process based on the National Institute of Standards and Technology(NIST) framework. It consists of the steps - identification, collection, analysis and presentation similar to the framework used by Sinha et al [sinha et al]. The first stage consists of identifying the incident or the device. The collection phase consists of acquiring the necessary evidence and identifying the required data while discarding any redundant or unrequired data. The analysis phase involves analyzing the data and then making the corresponding conclusions.The final presentation phase involves presenting the obtained conclusions to the required personnel. Our primary goal is to systematically acquire and examine forensic evidence from smartwatches and their paired mobile applications. To achieve this, two complementary data extraction approaches were implemented on a Xiaomi Pocophone F1 running Android 10 (rooted). The workflow followed is given in Fig 1.

4.1 Identification

This phase involves identifying the necessary devices and setting up the environment for the collection and analysis of forensic data. A rooted phone, Xiaomi Poco F1 was selected for the investigation, and the respective companion applications for each smartwatch were installed from Google Play Store. Separate Gmail accounts were used for creating accounts. To facilitate the extraction and analysis of data, both the tools, mitmproxy and DB Browser for SQLite were installed and configured. Each smartwatch was paired with its corresponding application:

1. boAt Xplorer RTL with the boAt ProGear app
2. Amazfit Band 5 with the Zepp app
3. Noise ColorFit Pulse with the NoiseFit app.

4.2 Collection

After creating the required accounts, each smartwatch was then synced with its respective applications, for the transfer of the stored smartwatch data to the application. The necessary data was collected using two different approaches.

4.2.1 Data Extraction from Rooted Phones:

The first approach involves accessing the mobile device at a deeper level by leveraging its rooted status, which provides full access to the file system. The details of this method are as follows:

1. Device Context: Using a Xiaomi Pocophone F1 running Android 10 (rooted) enabled the bypassing of standard security restrictions and access to the device's root directory where the smartwatch app stores its data.
2. Root Access and File System Exploration: A file explorer tool was employed to access the root file system. Specifically, the path /root/data/data/"Folder Name of the app" was navigated to, where all the data for each individual application is stored. By copying the files from this location, retrieval of local logs, cached files, configuration settings, and application databases that are not transmitted over the network was possible.
3. Utilization of Forensic Tools: In addition to the file explorer, tools such as DB Browser for SQLite were used to parse and analyze the locally extracted databases. This method effectively complemented the network interception approach by revealing locally cached information and system-level logs.
4. Advantages: Extracting data directly from the phone provides a more comprehensive view of forensic evidence, especially when some information is stored exclusively on the device. This approach verifies and supplements the findings obtained through network analysis, resulting in a robust investigation.

4.2.2 Network Traffic Interception Using mitmproxy:

This approach focuses on the communication between the smartwatch application and its associated cloud servers. By using mitmproxy, an open-source intercepting proxy, real-time network traffic generated during the synchronization process was captured and analyzed. Key details of this approach include:

1. Device Context: The experiments were conducted on a Xiaomi Pocophone F1 running Android 10. The rooted status of the device ensured that all necessary configurations could be implemented without typical operating system restrictions.

2. Data Collection: mitmproxy was configured to intercept encrypted communications between the mobile app and the cloud. This setup enabled the capture of transmitted data including personal information, health measurements, call logs, and message histories.
3. Data Decryption and Analysis: Once intercepted, the traffic was decrypted (where possible) to reveal the underlying data. This process facilitated the analysis of transmission patterns, identification of potential vulnerabilities in encryption protocols, and extraction of forensic artifacts, which are critical when traditional evidence is limited.
4. Advantages: This method is particularly useful when direct access to device storage is challenging or when much of the data is maintained in the cloud. It provides insighst into real-time data exchange and highlights security loopholes in data transmission.

4.3 Analysis

This phase involves the analysis of the data obtained by using open source tools. The collected data can be further divided into separate parameters like health parameters which consist of heart rate and SpO2, the activity data consisting of the daily steps, calories and distance count. This also included analysing the monthly activity data in some watches. Furthermore, details of user profiles were present which were also analysed.

4.4 Presentation

The final phase involves compiling and presenting the results of the analysis in a clear and concise format. The conclusions drawn from the study can be used to support further investigations or legal proceedings.

5 Results

5.1 BoAt Xplorer RTL

The data for BoAt Xplorer RTL, was extracted by going to the database directories of the BoAt progear app in a rooted phone. The databases were then opened in DB Browser for SQLite and all the information was obtained from there.

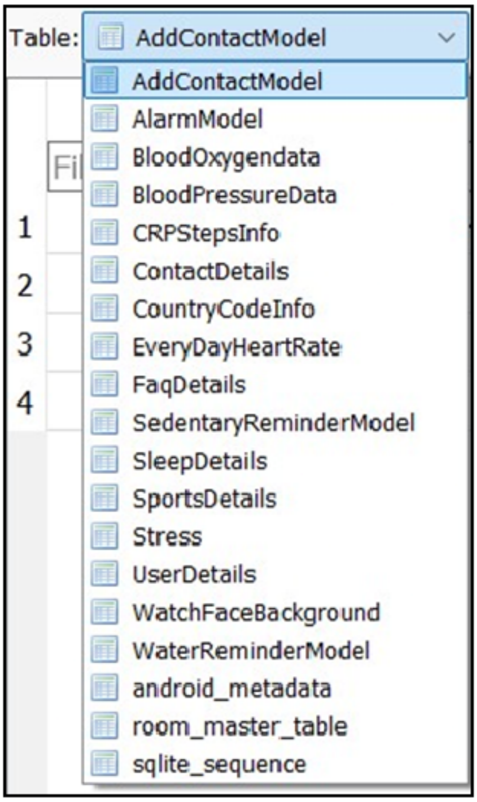


Fig. 2: All the information stored in the database

The data stored in the database can be viewed in Fig 2. It includes important data like user information which is stored in the UserDetails table, Blood oxygen levels which is stored in the BloodOxygendata table and blood pressure information which is stored in the BloodPressuredata table. It also includes some less sensitive data like steps information, sleep details and stress. Some common data like country code info, faq details and watch face background can also be seen.

Fig 3 illustrates the tables with populated rows and indicates the number of rows populated in each table.

Table: sqlite_sequence

	name	seq
	Filter	Filter
1	CountryCodeInfo	224
2	CRPStepsInfo	138
3	EveryDayHeartRate	588
4	BloodOxygendata	2
5	SleepDetails	5
6	UserDetails	4
7	AddContactModel	4

Fig. 3: No. of rows in filled tables

Table: CRPStepsInfo Filter in any column

	Id	activityDate	deviceName	macAddress	starttime	steps	distance	calories	duration
	Filter	Filter	Filter	Filter	Filter	Filter	Filter	Filter	Filter
1	1	2024/08/06	3	AC:98:B1:17:22:CD	1722923793	2863	2207	111829	1491
2	2	2024/07/30	3	AC:98:B1:17:22:CD	1722326985	5827	4485	226498	3031
3	3	2024/07/30	3	AC:98:B1:17:22:CD	1722335274	5905	4544	229420	3067
4	4	2024/07/30	3	AC:98:B1:17:22:CD	1722338070	5971	4594	231892	3095
5	5	2024/07/30	3	AC:98:B1:17:22:CD	1722346140	6252	4811	242906	3246
6	6	2024/07/30	3	AC:98:B1:17:22:CD	1722349799	10182	7798	390108	5354
7	7	2024/07/30	3	AC:98:B1:17:22:CD	1722352559	10996	8417	420597	5795
8	8	2024/07/30	3	AC:98:B1:17:22:CD	1722356811	11377	8711	435453	5970
9	9	2024/07/30	3	AC:98:B1:17:22:CD	1722357911	11502	8806	440135	6042
10	10	1970/01/01	3	AC:98:B1:17:22:CD	0	0	0	0	0
11	11	2024/07/31	3	AC:98:B1:17:22:CD	1722389060	261	198	9775	134
12	12	2024/07/31	3	AC:98:B1:17:22:CD	1722392779	621	471	23260	323
13	13	2024/07/31	3	AC:98:B1:17:22:CD	1722395389	742	563	27792	398
14	14	2024/07/31	3	AC:98:B1:17:22:CD	1722400177	811	616	30376	430
15	15	2024/07/31	3	AC:98:B1:17:22:CD	1722402967	1067	810	39965	551
16	16	2024/07/31	3	AC:98:B1:17:22:CD	1722407095	1576	1197	59030	816
17	17	2024/07/31	3	AC:98:B1:17:22:CD	1722410996	4405	3355	166193	2327
18	18	2024/07/31	3	AC:98:B1:17:22:CD	1722414577	5522	4208	208590	2948
19	19	2024/07/31	3	AC:98:B1:17:22:CD	1722416763	5902	4508	224578	3165
20	20	2024/07/31	3	AC:98:B1:17:22:CD	1722421332	6266	4785	238212	3377
21	21	2024/07/31	3	AC:98:B1:17:22:CD	1722424848	6489	4955	246564	3506
22	22	2024/07/31	3	AC:98:B1:17:22:CD	1722426367	6661	5085	253007	3587
23	23	2024/07/31	3	AC:98:B1:17:22:CD	1722432567	6973	5333	266159	3742
24	24	2024/07/31	3	AC:98:B1:17:22:CD	1722435955	7123	5447	271778	3816
25	25	2024/07/31	3	AC:98:B1:17:22:CD	1722439731	7377	5644	281916	3931

Fig. 4: CPRStepsInfo table

Fig 4, shows the number of steps taken by the user along with the distance covered in meters, the calories burnt and the duration it took in seconds. The date and start time of the activity can also be seen. The date of the activity is in the YYYY/MM/DD format along with the device name and the mac address of the smartwatch. The start time of the activity is stored in the system time format.

	Id	id1	country	code	fileUrl	min	max	supported	status
1	1	1	India	+91	https://prod-wearable-s3.s3.ap-...	10	10	1	active
2	2	4	Bangladesh	+880	https://prod-wearable-s3.s3.ap-...	7	12	0	active
3	3	2	United Arab Emirates	+971	https://prod-wearable-s3.s3.ap-...	7	12	0	active
4	4	5	Nepal	+977	https://prod-wearable-s3.s3.ap-...	7	12	0	active
5	5	7	Saudi Arabia	+966	https://prod-wearable-s3.s3.ap-...	2	15	0	active
6	6	8	United State of America	+1	https://prod-wearable-s3.s3.ap-...	2	15	0	active
7	7	9	United Kingdom	+44	https://prod-wearable-s3.s3.ap-...	2	15	0	active
8	8	10	Singapore	+65	https://prod-wearable-s3.s3.ap-...	2	15	0	active
9	9	11	Canada	+1	https://prod-wearable-s3.s3.ap-...	2	15	0	active
10	10	12	Australia	+61	https://prod-wearable-s3.s3.ap-...	2	15	0	active
11	11	13	Qatar	+974	https://prod-wearable-s3.s3.ap-...	2	15	0	active
12	12	14	Germany	+49	https://prod-wearable-s3.s3.ap-...	2	15	0	active
13	13	15	Kuwait	+965	https://prod-wearable-s3.s3.ap-...	2	15	0	active
14	14	16	Oman	+968	https://prod-wearable-s3.s3.ap-...	2	15	0	active
15	15	17	Bhutan	+975	https://prod-wearable-s3.s3.ap-...	2	15	0	active
16	16	18	Netherlands	+31	https://prod-wearable-s3.s3.ap-...	2	15	0	active
17	17	19	Sri Lanka	+94	https://prod-wearable-s3.s3.ap-...	2	15	0	active
18	18	20	France	+33	https://prod-wearable-s3.s3.ap-...	2	15	0	active
19	19	21	Maldives	+960	https://prod-wearable-s3.s3.ap-...	2	15	0	active
20	20	22	Afghanistan	+93	https://prod-wearable-s3.s3.ap-...	2	15	0	active
21	21	23	Albania	+355	https://prod-wearable-s3.s3.ap-...	2	15	0	active
22	22	24	Algeria	+213	https://prod-wearable-s3.s3.ap-...	2	15	0	active
23	23	25	Andorra	+376	https://prod-wearable-s3.s3.ap-...	2	15	0	active
24	24	26	Angola	+244	https://prod-wearable-s3.s3.ap-...	2	15	0	active
25	25	27	Anguilla	+1264	https://prod-wearable-s3.s3.ap-...	2	15	0	active

Fig. 5: CountryCodeInfo table

The various countries along with their country codes are stored in the CountryCodeInfo table shown in Fig 5.

The EveryDayHeartRate table as shown in Fig 6 shows the heart rate measurements taken by the device. The table contains the mac address of the device along with the values for the heart rate and the date when the heart rate was measured in system time.

Table: EveryDayHeartRate   Filter in any column

	Id	syncDate	deviceName	macAddress	startMeasureTime	heartRateIntegers
	Filter	Filter	Filter	Filter	Filter	Filter
1	1	1722882600000	6	AC:98:B1:17:22:CD	0	[{"a":72,"b":1722784839000}]
2	2	1722882600000	6	AC:98:B1:17:22:CD	0	[{"a":84,"b":1722325516000}]
3	3	1722882600000	6	AC:98:B1:17:22:CD	0	[{"a":86,"b":1722326424000}]
4	4	1722882600000	6	AC:98:B1:17:22:CD	0	[{"a":101,"b":1722327325000}
5	5	1722882600000	6	AC:98:B1:17:22:CD	0	[{"a":104,"b":1722328225000}
6	6	1722882600000	6	AC:98:B1:17:22:CD	0	[{"a":101,"b":1722329124000}
7	7	1722882600000	6	AC:98:B1:17:22:CD	0	[{"a":99,"b":1722330025000}]
8	8	1722882600000	6	AC:98:B1:17:22:CD	0	[{"a":90,"b":1722330919000}]
9	9	1722882600000	6	AC:98:B1:17:22:CD	0	[{"a":87,"b":1722331818000}]
10	10	1722882600000	6	AC:98:B1:17:22:CD	0	[{"a":85,"b":1722332717000}]
11	11	1722882600000	6	AC:98:B1:17:22:CD	0	[{"a":82,"b":1722333618000}]
12	12	1722882600000	6	AC:98:B1:17:22:CD	0	[{"a":78,"b":1722334516000}]
13	13	1722882600000	6	AC:98:B1:17:22:CD	0	[{"a":72,"b":1722335417000}]
14	14	1722882600000	6	AC:98:B1:17:22:CD	0	[{"a":66,"b":1722336322000}]
15	15	1722882600000	6	AC:98:B1:17:22:CD	0	[{"a":68,"b":1722337225000}]
16	16	1722882600000	6	AC:98:B1:17:22:CD	0	[{"a":88,"b":1722338124000}]
17	17	1722882600000	6	AC:98:B1:17:22:CD	0	[{"a":75,"b":1722339023000}]
18	18	1722882600000	6	AC:98:B1:17:22:CD	0	[{"a":82,"b":1722339917000}]
19	19	1722882600000	6	AC:98:B1:17:22:CD	0	[{"a":83,"b":1722340818000}]
20	20	1722882600000	6	AC:98:B1:17:22:CD	0	[{"a":90,"b":1722341719000}]
21	21	1722882600000	6	AC:98:B1:17:22:CD	0	[{"a":81,"b":1722342617000}]
22	22	1722882600000	6	AC:98:B1:17:22:CD	0	[{"a":100,"b":1722343516000}
23	23	1722882600000	6	AC:98:B1:17:22:CD	0	[{"a":79,"b":1722344424000}]
24	24	1722882600000	6	AC:98:B1:17:22:CD	0	[{"a":72,"b":1722345321000}]
25	25	1722882600000	6	AC:98:B1:17:22:CD	0	[{"a":67,"b":1722346224000}]

Fig. 6: EveryDayHeartRate table

Table: BloodOxygendata

	Id	activityDate	b_oxygen_time	b_oxygen_value
	Filter	Filter	Filter	Filter
1	1	2024/08/06	1722784923	99
2	2	2024/08/06	1722927773	99

Fig. 7: BloodOxygendata table

Fig 7, shows the blood oxygen value of the user measured by the smartwatch along with the date and time in which the measurement was taken. The date is in the YYYY/MM/DD format and the time is in the system time format.

	Id	SleepDate	deviceName	macAddress	totalTime	restfulTime	lightTime	soberTime	details	actualWakeupTime	actualBedTime	actualSleepTime
	Filter	Filter	Filter	Filter	Filter	Filter	Filter	Filter	Filter	Filter	Filter	Filter
1	1	1722277800000	6	AC:98:B1:17:22:CD	36060	17100	18960	0	[{"a":2520,"b":0,"c":0,"d":1},{"a":...	1722733860	1722364620	1722369000
2	2	1722709800000	6	AC:98:B1:17:22:CD	38040	25440	12600	0	[{"a":2580,"b":0,"c":0,"d":1},{"a":...	1722897720	1722795480	1722806400
3	3	1722796200000	6	AC:98:B1:17:22:CD	7440	2940	4500	0	[{"a":3120,"b":0,"c":0,"d":1},{"a":...	1722905220	1722897780	1722903840
4	4	1722882600000	6	AC:98:B1:17:22:CD	37500	23700	13800	0	[{"a":2760,"b":0,"c":0,"d":1},{"a":...	1723060740	1722962880	1722970500
5	5	1722969000000	6	AC:98:B1:17:22:CD	16680	12300	4380	0	[{"a":2520,"b":0,"c":0,"d":1},{"a":...	1723077540	1723060860	1723064820

Fig. 8: SleepDetails table

The table in Fig 8 shows the sleep date, total time slept, amount of restful and light sleep the user had along with the time that they went to bed, slept and woke up.

	Id	gender	age	phoneNumber	emailId	firstName	lastName	height	height_unit	weight	weightUnit	goal_calorie	goal_steps	name	dob	profilePic
	Filter	Filter	Filter	Filter	Filter	Filter	Filter	Filter	Filter	Filter	Filter	Filter	Filter	Filter	Filter	Filter
1	4	Female	21	NULL	[redacted]@gmail.com	namita patil		5.7	Feet	57	kg	650	10000	namita patil	23/01/2003	NULL

Fig. 9: UserDetails table

The Fig 9 shows the UserDetails table which shows the name, age, gender, DoB, phone number, email ID, height, weight and the daily step and calorie goals of the user. The UserDetails table shows all the information that the user entered while they were setting up their account.

5.2 Amazfit Band 5

The data for the Amazfit Band 5 was extracted in a similar manner to the BoAt smartwatch. First, the database directories of the Zepp mobile application were located on a rooted phone. Then, the database information was viewed using an sql viewer github repository[13].

HEART_RATE (3 rows)

```
SELECT * FROM 'HEART_RATE' LIMIT 0,30
```

TYPE	DEVICE_TYPE	DEVICE_SOURCE	TIME	HR	STATE	TIME_ZONE	DEVICE_ID
2	0	73	1724059724	109	1	19800000	D059B4FFFEEBB221
2	0	73	1724061744	96	1	19800000	D059B4FFFEEBB221
2	0	73	1724061890	88	1	19800000	D059B4FFFEEBB221

Fig. 10: Heart rate information

Fig 10 shows the heart rate of the user measured at the particular time. The time is stored in the system time format.

ALARM (10 rows)

```
SELECT * FROM 'ALARM' LIMIT 0,30
```

_id	CALENDAR	ENABLED	IS_UPDATE	M_DAYS	M_SMART_WAKEUP_DU...	VISIBLE	IS_SMART_WAKEUP	INDEX	IS_S
1	1724031015000	0	0	31	0	1	1	0	0
2	1724031015000	0	0	127	0	1	1	1	0
3	1724031015000	0	0	0	0	1	1	2	0
4	1724059275385	0	0	0	0	0	1	3	0
5	1724059275388	0	0	0	0	0	1	4	0
6	1724059275389	0	0	0	0	0	1	5	0
7	1724059275392	0	0	0	0	0	1	6	0
8	1724059275396	0	0	0	0	0	1	7	0
9	1724059275398	0	0	0	0	0	1	8	0
10	1724059275403	0	0	0	0	0	1	9	0

Fig. 11: Alarm data

Fig 11 shows the alarm data with the date in the system time format.

USER_INFOS (1 rows)

```
SELECT * FROM 'USER_INFOS' LIMIT 0,30
```

USER_ID	NAME	BIRTHDAY	AVATAR_URL	AVATAR_PATH	GENDER	HEIGHT	SYNCED	WEIGHT	TARGET_WEIGHT	CITY	LONGI
3305351217	navyaperam16	2002-7			0	170	1	51	-1		

Fig. 12: User information

Fig 12 displays the user information that was entered while setting up the device. It includes the user ID, name, gender, height and weight of the user.

TRACKRECORD (1 rows) Export Execute

```
SELECT * FROM 'TRACKRECORD' LIMIT 0,30
```

_id	SOURCE	TYPE	DATE	TRACKID	DISTANCE	COSTTIME	CAL	PACE	SFREQ	AVGHR	FFPERCENT	V	ENDTIME	MAXRTP	MINRTP
1	173	8	2024-08-19	1724059420	214	162	18	0.7559999823570251	88	150	-1	12	1724059587	3.8496580123901367	0.4953720867633

Fig. 13: Steps records

Fig 13 shows the date of the activity, total distance covered in meters, the pace of the user and the average heart rate.

HM_PROPERTY (3 rows)

SELECT * FROM 'HM_PROPERTY' LIMIT 0,30

KEY	VALUE	STATUS
IS_IN_MAINLAND	false	1
huami.mifit.sport.shareRouter	https://fe-cdn.huami.com/sport-history/index.html#/	1
huami.mifit.user.settings.synctime	{"D0:59:B4:EB:B2:21":{"single_stress":1724062001000}}	0

Fig. 14: Property information

Fig 14 shows that the smartwatch was set up in a region outside China and also displays a URL that might be used to share data. The row `IS_IN_MAINLAND` suggests that the Zepp app is currently running in global mode rather than Chinese mode. Even the MAC address of the device can be seen.

context (1 rows) Export

SELECT * FROM 'context' LIMIT 0,30 Execute

	appVersionCode	appVersionName	sdkVersionName	osLevel	osVersion	channel	locale	network	resolution	eventVersion	packageName	brand	model
...	151386	8.11.5-play	2.0.2	29	10	play	en_IN	wifi	1080*2027	1	com.huami.watch.hmwat...	Xiaomi	POCO F1

Fig. 15: Smartwatch information

Fig 15 shows the smartwatch's information like the operating system version and level, the app version, the sdk version and even the resolution of the device.

sqlite_sequence (4 rows)

SELECT * FROM 'sqlite_sequence' LIMIT 0,30

name	seq
CONFIG	0
FW_DWON_SUCCESS_INFO	7
ALARM	10
TRACKRECORD	1

Fig. 16: Sequence table

Fig 16 presents an internal system table that keeps track of the last assigned auto-increment values for each table with an auto-increment primary key. This table indicates that no recent records have been modified in the `CONFIG` table. The entry `FW_DWON_SUCCESS_INFO` could relate to recent successful firmware downloads. The `ALARM` row suggests that 10 alarms were recently created. `TRACKRECORD` shows that only one record has been logged so far.

_id	ALARM_NOTIFY_EN...	DAY_REPORT_NOTI	DISCONNECTED_RE...	ENABLE_CONNECTE...	GOAL_STEPS_COUNT	LIGHT_COLOR	IN_COMING_CALL
0	0	OFF		0	8000	BLUE	259

Fig. 17: User default settings

In fig 17, the config table displays the default smartwatch settings, including notifications, step goals and call alerts. It shows that alarm and daily report notifications are currently turned off and that the step goal is set to 8000. It also shows that the notification light color is set to blue.

5.3 Noise Colourfit Pulse

The data for the Noise Colourfit Pulse smartwatch was extracted by using the mitmproxy tool. The tool acted as a middle man between the smartwatch and the mobile application and received all the information that was being sent through the connection. mitmproxy was utilized to capture the data as the database information on the smartphone couldn't be found.

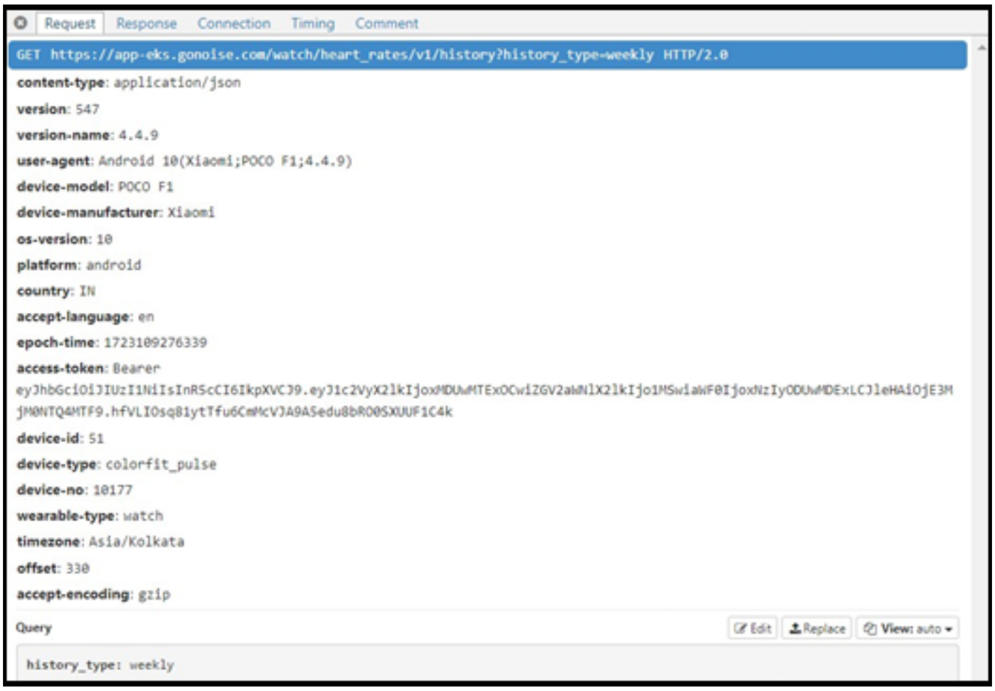


Fig. 18: Device Information

Fig 18 shows the device information that we accessed through the mitmproxy tool. It includes data like the smartphone device manufacturer, which was used for rooting and the smartwatch model, name, device id and its access token along with the country and language. It also displays the purpose of the request which in this case was the heart rate data and the format in which the data was obtained which is JSON. This shows that the heart rate data is stored remotely on the Noise server and is obtained by using the access token. This token is used to verify the user's identity and authorize access to the smartwatch's data. If exposed it can be misused to access personal health data.


```
{
    "data": {
        "is_pwd": false,
        "message": "OTP verified",
        "token": {
            "access_token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.ey
            "refresh_token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.e
        },
        "user": {
            "auth_token": "632fdbd5-fc6b-4696-aebc-cc8ccacac949",
            "country": "IN",
            "country_code": "",
            "email": "navyaperam16@gmail.com",
            "end_game": -1,
            "first_name": "Navy",
            "google_uid": "116592841337118985091",
            "id": 10501118,
            "image_url": "https://lh3.googleusercontent.com/a/AEdFTp
            "interests": [
                {
                    "id": 1,
                    "name": "Swimming"
                },
                {
                    "id": 2,
                    "name": "Walking"
                },
                {
                    "id": 3,
                    "name": "Cycling"
                },
                {
                    "id": 5,
                    "name": "Better Sleep"
                },
                {
                    "id": 6,
                    "name": "Productivity"
                }
            ],
            "is_verified": 1,
            "language": "en",
            "last_name": "",
            "location": null,
```


Fig. 19: User authentication profile data

The Fig 19 displays the profile data of the user, along with the authentication information. It also displays the access and refresh token used for authentication.

Fig 20 displays the user's profile data such as their personal details and their goals.


```
            "notifications_enabled": 1,
            "offset": 330,
            "password_digest": null,
            "timezone": "Asia/Kolkata",
            "user_goals": {
                "calories_goals": 240,
                "distance_goals": 6000,
                "id": 11210960,
                "sleep_goals": 8,
                "step_goals": 6000,
                "unit_system": "Metric",
                "user_id": 10501118
            },
            "user_info": {
                "dob": "[redacted],
                "gender": "Female",
                "height": 161,
                "step_length": 70,
                "weight": 50
            },
            "vendor_id": null,
            "vendor_ids": null
        }
    },
    "success": true
}
```


Fig. 20: User profile data

Fig 21 displays the heartbeat data collected over all the days of a week. It also consists of the cumulative heartbeat data containing the average, minimum and maximum rate over the week. This data is displayed as a JSON format and is obtained through mitmproxy. Similarly monthly data and daily data along with hourly breakups can also be found.

```
"data": {
    "cumulative": {
        "avg": 90,
        "max": 123,
        "min": 69
    },
    "heart_rates": [
        {
            "avg": 92,
            "date": "2024-08-02",
            "max": 120,
            "min": 69,
            "resting_hr": 0
        },
        {
            "date": "2024-08-03"
        },
        {
            "date": "2024-08-04"
        },
        {
            "avg": 87,
            "date": "2024-08-05",
            "max": 97,
            "min": 78,
            "resting_hr": 0
        },
        {
            "date": "2024-08-06"
        },
        {
            "date": "2024-08-07"
        },
        {
            "avg": 93,
            "date": "2024-08-08",
            "max": 123,
            "min": 72,
            "resting_hr": 0
        }
    ],
    "history_type": "weekly"
},
"message": "",
"success": true,
"time": "1723109279459"
```


Fig. 21: Weekly heart rate data

```
{
    "data": {
        "avg": {
            "calories": 132,
            "distance": 2557,
            "steps": 3874
        },
        "history_type": "daily",
        "step_activities": [
            {
                "active_time": 0,
                "calories": 225,
                "date": "2024-08-02",
                "distance": 4345,
                "hourly_breakup": [
                    {
                        "active_time": 0,
                        "calories": 0,
                        "date": "02/08/2024",
                        "distance": 195,
                        "hour_of_the_day": 8,
                        "steps": 296
                    },
                    {
                        "active_time": 0,
                        "calories": 0,
                        "date": "02/08/2024",
                        "distance": 765,
                        "hour_of_the_day": 9,
                        "steps": 1160
                    },
                    {
                        "active_time": 0,
                        "calories": 0,
                        "date": "02/08/2024",
                        "distance": 197,
                        "hour_of_the_day": 10,
                        "steps": 299
                    },
                    {
                        "active_time": 0,
                        "calories": 0,
                        "date": "02/08/2024",
                        "distance": 658,
                        "hour_of_the_day": 11,
                        "steps": 997
```


Fig. 22: Daily distance data

Fig 22 shows the daily data of multiple attributes such as steps, calories and distance. It displays the average calories burnt in a single day along with the average distance and steps covered. It stores the values in an hourly format. This could also help the forensic investigators understand the user's activities during a particular period of time. Similarly a weekly format can also be seen.

```
{
    "data": {
        "avg": {
            "calories": 132,
            "distance": 2557,
            "steps": 3874
        },
        "history_type": "weekly",
        "step_activities": [
            {
                "active_time": 0,
                "calories": 225,
                "date": "2024-08-02",
                "distance": 4345,
                "steps": 6583
            },
            {
                "active_time": 0,
                "calories": 0,
                "date": "2024-08-03",
                "distance": 9,
                "steps": 14
            },
            {
                "date": "2024-08-04"
            },
            {
                "active_time": 0,
                "calories": 119,
                "date": "2024-08-05",
                "distance": 2309,
                "steps": 3499
            },
            {
```


Fig. 23: Weekly distance data

Fig 23, displays the number of steps that were taken that week along with the total distance covered in meters and the number of calories burnt.

```
{
    "data": {
        "blood_oxygen": [
            {
                "date": "2024-07-30"
            },
            {
                "date": "2024-07-31"
            },
            {
                "date": "2024-08-01"
            },
            {
                "date": "2024-08-02"
            },
            {
                "date": "2024-08-03"
            },
            {
                "date": "2024-08-04"
            },
            {
                "count": 99,
                "date": "2024-08-05",
                "hourly_break_up": [
                    {
                        "date": "05/08/2024",
                        "time": "15:02",
                        "value": 99
                    }
                ],
                "max_count": 99,
                "min_count": 99
            }
        ],
        "cumulative": {
            "count": "99.0",
            "max": 99,
            "min": 99
        },
        "history_type": "daily"
    },
    "message": "",
    "success": true
}
```

Fig. 24: Daily blood oxygen data

Fig 24, presents the blood oxygen data of that particular day. Similarly, weekly and monthly blood oxygen data can also be obtained.


```
{
    "data": {
        "activities": [
            {
                "activity_type": "walking",
                "calories": 105,
                "date": "2024-07-30",
                "distance": 202,
                "duration": 209633,
                "heart_rate_average": 104,
                "id": 248314911,
                "steps": 296,
                "time": "22:43:41"
            },
            {
                "activity_type": "treadmill",
                "calories": 131,
                "date": "2023-10-01",
                "distance": 0,
                "duration": 792,
                "heart_rate_average": 131,
                "id": 222745695,
                "steps": 1005,
                "time": "19:31:01"
            }
        ]
    },
    "message": "",
    "success": true,
    "time": "1722850028062"
}
```

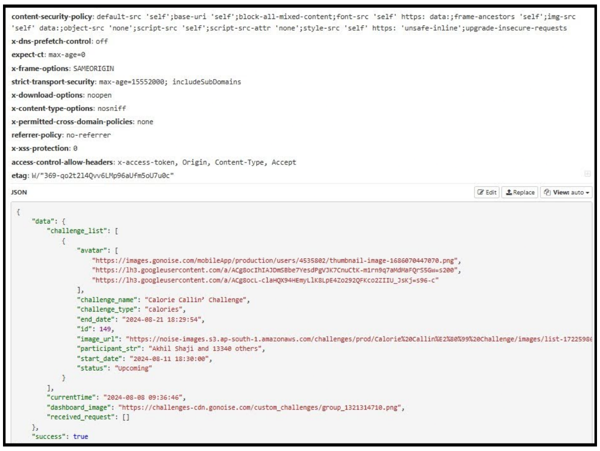

Fig. 25: Activity data

The type of activities performed by the user can be viewed in Fig 25. Multiple parameters are described for each activity, these include the activity type and id, calories burnt, distance, steps and the heart rate average during the activity. The activity type can be chosen from various activities in the list such as outdoor running, walking, outdoor cycling, trial run, trekking, treadmill, indoor cycling, yoga. It also includes the duration of the activity in system time. The time and date at which the activity was performed are also visible.


```
{
    "data": {
        "avg_awake": null,
        "avg_bedtime": "0:0",
        "avg_deep": null,
        "avg_duration": 0,
        "avg_light": null,
        "avg_readiness_score": 0,
        "avg_sleep_score": 0,
        "avg_sober": null,
        "bo_max": null,
        "bo_min": null,
        "breakup_duration": [
            {
                "date": "2024-07-30"
            },
            {
                "date": "2024-07-31"
            },
            {
                "date": "2024-08-01"
```


Fig. 26: Sleep data

The sleep data of the user can be extracted through this method as well, as seen by the format in Fig 26.




```
content-security-policy: default-src 'self';base-uri 'self';block-all-mixed-content;font-src 'self' https: data:;frame-ancestors 'self';img-src 'self' data:;object-src 'none';script-src 'self';script-src-attr 'none';style-src 'self' https: 'unsafe-inline';upgrade-insecure-requests
x-dns-prefetch-control: off
expect-ct: max-age=0
x-frame-options: SAMEORIGIN
strict-transport-security: max-age=15552000; includeSubDomains
x-download-options: noopen
x-content-type-options: nosniff
x-permitted-cross-domain-policies: none
referrer-policy: no-referrer
x-xss-protection: 0
access-control-allow-headers: x-access-token, Origin, Content-Type, Accept
etag: W/"369-qo2t214Qvv6LMp96aUfm5oU7u0c"
JSON                                             Edit  Replace  View: auto
{
    "data": {
        "challenge_list": [
            {
                "avatar": [
                    "https://images.gonoise.com/mobileApp/production/users/4535802/thumbnail-image-1686070447070.png",
                    "https://lh3.googleusercontent.com/a/ACg8ocIhIAJDmSBbe7YesdPgVJK7CnuCtK-m1rn9q7aMdMaFQrS5Gw=s200",
                    "https://lh3.googleusercontent.com/a/ACg8ocL-claHQX94HEmyLlK8LpE4Zo292QFKco2ZIIU_JsKj=s96-c"
                ],
                "challenge_name": "Calorie Callin’ Challenge",
                "challenge_type": "calories",
                "end_date": "2024-08-21 18:29:54",
                "id": 149,
                "image_url": "https://noise-images.s3.ap-south-1.amazonaws.com/challenges/prod/Calorie%20Callin%E2%80%99%20Challenge/images/list-1722598
                "participant_str": "Akhil Shaji and 13340 others",
                "start_date": "2024-08-11 18:30:00",
                "status": "Upcoming"
            }
        ],
        "currentTime": "2024-08-08 09:36:46",
        "dashboard_image": "https://challenges-cdn.gonoise.com/custom_challenges/group_1321314710.png",
        "received_request": []
    },
    "success": true
```


Fig. 27: User challenge information

Fig 27 displays the challenge “Calorie Callin’ Challenge”, one of the multiple challenges undertaken by the smartwatch users. It contains the start and end dates and times of the particular challenge. It also contains information regarding the participants’ profiles and their profile names.


```
{
    "admin_post_cta": null,
    "caption": "Due to the demise of my naniji I won't be active for a week or so, I'll try to post some morning posts but there will be no activity log for some days.",
    "comment": {
        "comment": "Ok",
        "comment_id": 1622135,
        "image_url": "https://images.gonoise.com/mobileApp/production/users/4722860/thumbnail-image-1685707169323.png",
        "is_my_comment": false,
        "name": "Bithika Pramanik",
        "post_id": 3555534,
        "updated_at": "2024-08-08T09:18:40.000Z",
        "user_id": 4722860,
        "user_type": "user"
```


Fig. 28: Challenge participant’s messages


```
"admin_post_cta": null,
"caption": "A new day 🌻 \nGood morning 🙂",
"comment": {
    "comment": "Aap 01",
    "comment_id": 1622111,
    "image_url": "https://lh3.googleusercontent.com/a/ACg8ocIFO2RncqhkAsXOfgaXCttNTu27UPrfbPAtyylBOyuTmOC0Hg=s96-c
    "is_my_comment": false,
    "name": "Shailesh Kumar",
    "post_id": 3514358,
    "updated_at": "2024-08-08T08:41:37.000Z",
    "user_id": 29037499,
    "user_type": "user"
```


Fig. 29: Challenge participant’s messages

Fig 28 and Fig 29 display the messages posted by participants on the dashboard during past challenges and the corresponding comments by other participants. The commenters’ id’s and usernames along with the date and time at which their comments were posted can also be viewed.

The profile picture and details of the users who participate in challenges present in the app are visible through this method. Fig 30 shows the profile information of a user. It displays the person’s profile image along the date they last modified their personal details. It can be seen that the user’s profile picture is stored on Amazon S3, a cloud storage service and is delivered via Amazon CloudFront. This is a CDN that caches and distributes the image efficiently. The image is encrypted using AES-256 when at rest. However, there are a large number of security gaps, such as token exposure, where an attacker upon gaining access to the token data can retrieve the private data, including the profile picture as seen. Misconfigured S3 permissions could also make the images publicly accessible, leading to unauthorized access or modification. Additionally, if the TLS encryption is not enforced, intercepted requests could expose the images, making them vulnerable to identity spoofing. CloudFront caching means that even if an image is deleted, it may still be accessible for a period of time.

Content-Type: image/jpeg
Content-Length: 10944
Connection: keep-alive
Date: Thu, 08 Aug 2024 09:32:35 GMT
Last-Modified: Tue, 31 Oct 2023 12:01:36 GMT
ETag: "5867f450203d40e7044b4f17e630d3e1"
x-amz-server-side-encryption: AES256
Accept-Ranges: bytes
Server: AmazonS3
X-Cache: Hit from cloudfront
Via: 1.1 fcc5d2d615e3904d2f9b279dd96de872.cloudfront.net (CloudFront)
X-Amz-Cf-Pop: BLR50-P2
X-Amz-Cf-Id: KyPwjiIST19m9ZJXPSsVuLxn6ZSsql7ySHG99Xv1cH7o7-EVyqfkqg==
Age: 254

JPEG Image

Fig. 30: Visible challenge participant's profile data

Fig 31 shows data from various time frames like the total steps taken, distance covered and calories burnt. A message is also stored regarding the distance covered and the number of calories burnt by comparing the data of the current time frame to the previous time frame.

```
{
    "data": {
        "daily": {
            "calories_msg": "You burnt more calories today than yesterday.",
            "distance_msg": "Your total distance today is more than yesterday.",
            "msg": "Your total step count today is more than yesterday.",
            "today": {
                "calories": 184,
                "distance": 3565,
                "label": "Today",
                "steps": 5402
            },
            "yesterday": {
                "calories": 0,
                "distance": 0,
                "label": "2024-08-07",
                "steps": 0
            }
        },
        "max": null,
        "monthly": {
            "calories_msg": "You are burning less calories a day this month than last month.",
            "current": {
                "calories": 528,
                "distance": 10228,
                "label": "This Month",
                "steps": 15498
            },
            "distance_msg": "You covered less distance a day this month than last month.",
            "last": {
                "calories": 1133,
                "distance": 21939,
                "label": "Last Month",
                "steps": 33239
            },
            "msg": "You are taking fewer steps a day this month than last month"
        },
        "weekly": {
            "calories_msg": "You are burning more calories a day this week than last week. ",
            "current": {
                "calories": 303,
                "distance": 5874,
                "label": "This Week",
                "steps": 8901
            },
            "distance_msg": "You covered more distance a day this week than last week.",
            "last": {
                "calories": 225,
                "distance": 4354,
                "label": "Last Week",
                "steps": 6597
            },
            "msg": "You are taking more steps a day this week than last week. "
        }
    },
    "message": "",
    "success": true,
    "time": "1723109331133"
}
```


Fig. 31: Comparative user activity data

```
"data": {
    "customwatchface_list": {
        "artistic": null,
        "background": [
            {
                "category_id": 1,
                "image_url": "https://watchfaces-cdn.gonoise.com/watch_faces/production/custom_watchface/background+images/group_1321314748.webp",
                "title": "upload"
            },
            {
                "category_id": 1,
                "image_url": "https://watchfaces-cdn.gonoise.com/watch_faces/prod/custom_watchface/admin_upload/square/240%20X%20240/Father%27s%20Day-171
                "title": "Father's Day"
            },
            {
                "category_id": 1,
                "image_url": "https://watchfaces-cdn.gonoise.com/watch_faces/production/custom_watchface/background+images/Square/independence+day+Square
                "title": "Independence day"
            },
            {
                "category_id": 1,
                "image_url": "https://watchfaces-cdn.gonoise.com/watch_faces/undefined/custom_watchface/admin_upload/square/240%20X%20240/Earthday-1-1713
                "title": "Earthday-1"
            },
            {
                "category_id": 1,
                "image_url": "https://watchfaces-cdn.gonoise.com/watch_faces/production/custom_watchface/admin_upload/square/240%20X%20240/Shiv-170989099
                "title": "Shiv"
            },
            {
                "category_id": 1,
                "image_url": "https://watchfaces-cdn.gonoise.com/watch_faces/undefined/custom_watchface/admin_upload/square/240%20X%20240/Mothers%20Day-2
                "title": "Mothers Day-2"
```


Fig. 32: Watchface design information

Fig 32 displays the multiple watchfaces present in the watch face gallery along with their links. Upon clicking on the links the watchfaces designs can be viewed online.


```
HTTP/1.1 200 OK
Content-Type: image/webp
Content-Length: 13326
Connection: keep-alive
Date: Wed, 07 Aug 2024 14:12:38 GMT
Last-Modified: Fri, 06 Oct 2023 12:33:47 GMT
ETag: "560ab987a74a1849b38bd6e82cee6aa8"
x-amz-server-side-encryption: AES256
Accept-Ranges: bytes
Server: AmazonS3
X-Cache: Hit from cloudfront
Via: 1.1 7bd3286fc3728d83ebaaf909834cfd82.cloudfront.net (CloudFront)
X-Amz-Cf-Pop: BLR50-P1
X-Amz-Cf-Id: rQgbuywAsP8iLuzrv21a-QyNWqqq-IRdR8M4i8ap0LL9pMiib465hA==
Age: 69943
Vary: Origin
```


WEBP Image

Fig. 33: Single watch face design information

Fig 33 shows one of the many watchfaces designs present in the application, as seen in Fig 32. The user can choose from the multiple interfaces, which then loads onto the watch after the syncing process. The above image shows that the WebP image is hosted on Amazon S3 and is accessed through CloudFront.

Table 1 is a compilation of all the data that was obtained by studying the three smartwatches.

Table 1: Comparison of Smartwatch Features and Results

	User Details	Device Details	Steps Info	Sleep Data	Heart Rate	SpO2	Other Profiles	Alarm Data
Noise Colorfit Pulse	✓	✓	✓	✓	✓	✓	✓	✗
BoAt Xplorer RTL	✓	✗	✓	✓	✓	✓	✗	✓
Amazfit Band 5	✓	✓	✓	✗	✓	✗	✗	✓

6 Conclusion

In this study, the NIST digital forensic analysis was used to analyze three smartwatches: Noise Colorfit Pulse, BoAt Xplorer RTL, and Amazfit Band 5. The primary objective was to examine the type of user data generated and stored by the application after user registration and login. Additionally, the app's security was analyzed, focusing on the extent of information accessible to the investigators, including the user profiles and challenge details. We successfully extracted data like user information, steps taken, and heart rate, and verified it against the data shown on the smartwatches and in their apps. For all three smartwatches, we managed to retrieve user information along with stats such as heart rate. The data obtained provides valuable insights into the user's daily activities, which can be crucial in important investigations. For example, investigators can analyze whether the user engaged in any energy-intensive activities based on health parameters like heart rate and SpO2 rates. Additionally, sleep schedules can be reviewed based on recorded data. Also, though services like CloudFront enable seamless access to smartwatch data stored in cloud, this also involves several potential security risks. It was noted that the user's access token could be exploited to view user stats and data stored in the database, posing a security risk. Similarly, profiles of other users, along with their pictures, could be accessed, highlighting potential security threats. To mitigate these risks, it is essential to enforce strict IAM policies, signed URLs, and access logging to prevent unauthorized access and data leaks. Enhanced protection measures are crucial to ensure that sensitive user's information is protected from unauthorized access and breaches, especially in Cloud[14]. Paper [15] mentions about how many of the IoT devices are weak in security, lacking basic measures like encryption which could make it easy for hackers to collect personally identifiable information(PII). They also rightfully point out the vast majority of devices that exist in the market and all of them could have different security needs and data collection methods. This would make it harder for forensic experts as they would have to identify the correct tools and methods that are needed to extract the necessary information. Future research could expand the scope by including more types of smartwatches. Further investigation is needed to determine if the data extraction remains consistent across different smartphone models, including iOS devices. Additionally, exploring data extraction through non-open-source tools or subscription-based tools could reveal additional data from the devices.

7 Acknowledgment

We would like to express our sincere gratitude to the U.S. Army for their generous funding and unwavering support, which made the AI-Enabled National Conference on Forensic Investigations Network in Digital Sciences(FINDS) possible. Their commitment to advancing research and fostering collaboration is deeply appreciated.

We are profoundly grateful to Dr. S. S. Iyengar for his outstanding contributions to the field, which have been a constant source of inspiration to us. His pioneering work has shaped much of our research, and we are deeply thankful for the impact his research has had on our academic journey. His dedication and innovations continue to inspire us in our pursuit of knowledge.

References

[1] S.S. Iyengar, Sina Nabavi, Yashas Hariprasad, Prasad HB, Krishna Mohan C. "Artificial Intelligence in Practice: Theory and Applications for Cyber Security and Forensics", Springer Nature (Forthcoming)

[2] Y. H. Yoon and U. Karabiyik, "Forensic Analysis of Fitbit Versa 2 Data on Android," Electronics, vol. 9, no. 9, p. 1431, Sep. 2020, doi: https://doi.org/10.3390/electronics9091431.

[3] J. Williams, Á. MacDermott, K. Stamp, and F. Iqbal, "Forensic Analysis of Fitbit Versa: Android vs iOS," IEEE Xplore, May 01, 2021. https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=9474285

[4] F. Hantke and A. Dewald, "How can data from fitness trackers be obtained and analyzed with a forensic approach?," 2020 IEEE European Symposium on Security and Privacy Workshops (EuroS&PW), Sep. 2020, doi: https://doi.org/10.1109/eurospw51379.2020.00073.

[5] P. Domingues, J. Francisco, and M. Frade, "Post-mortem digital forensics analysis of the Zepp Life android application," Forensic Science International: Digital Investigation, vol. 45, p. 301555, Jun. 2023, doi: https://doi.org/10.1016/j.fsidi.2023.301555.

[6] F. Nunes, P. Domingues, and M. Frade, “The Digital Footprints on the Run: A Forensic Examination of Android Running Workout Applications,” Future Internet, vol. 16, no. 9, pp. 304–304, Aug. 2024, doi: https://doi.org/10.3390/fi16090304.

[7] R. Sinha, Vikas Sihag, G. Choudhary, Manu Vardhan, and P. Singh, “Forensic Analysis of Fitness Applications on Android,” Communications in computer and information science, pp. 222–235, Jan. 2022, doi: https://doi.org/10.1007/978-981-16-9576-6_16.

[8] S. J. Patel, N. K. Chaudhary, B. N. Gohil, and S. S. Iyengar, “Machine Learning Algorithms for Attack and Anomaly Detection in IoT,” in *Information Security, Privacy and Digital Forensics – Select Proceedings of the International Conference, ICISPD 2022*, Springer Nature, 2022, doi: 10.1007/978-981-99-5090-4.

[9] S. Raghavendra, C. S. Reddy, C. M. Geeta, R. Buyya, K. R. Venugopal, S. S. Iyengar, and L. M. Patnaik, ”Survey on data storage and retrieval techniques over encrypted cloud data,” International Journal of Computer Science and Information Security, vol. 14, no. 9, p. 718, 2016.

[10] S. S. Iyengar, Seyedsina Nabavirazavi, H. Rathore, Y. Hariprasad, and N. K. Chaudhary, “Advancing Forensic Science: AI and Knowledge Graphs Unlock New Insights,” *Journal of Forensic Research*, vol. 15, no. 3, pp. 1–7, Jun. 2024, doi: https://doi.org/10.37421/2157-7145.2024.15.615.

[11] S. Raghavendra, K. Meghana, P. A. Doddabasappa, C. M. Geeta, R. Buyya, K. R. Venugopal, S. S. Iyengar, and L. M. Patnaik, ”Index generation and secure multi-user access control over an encrypted cloud data,” Procedia Computer Science, vol. 89, pp. 293–300, 2016.

[12] S. Raghavendra, C. M. Geeta, R. Buyya, K. R. Venugopal, S. S. Iyengar, and L. M. Patnaik, ”DRSMS: Domain and range specific multi-keyword search over encrypted cloud data,” International Journal of Computer Science and Information Security, vol. 14, no. 5, p. 69, 2016.

[13] Github repository: https://github.com/inloop/sqlite-viewer Website: https://inloop.github.io/sqlite-viewer/

[14] K. G. Boroojeni, M. H. Amini, and S. S. Iyengar, “Cloud Network Data Security,” Springer eBooks, pp. 71–82, Oct. 2016, doi: https://doi.org/10.1007/978-3-319-45050-6 5.

[15] D. R. Garcia Avila, J. F. Miller, and S. S. Iyengar, ‘Current Challenges in IoT Security and Forensics: Strategies for a Secure Connected Future’, Key Issues in Network Protocols and Security [Working Title]. IntechOpen, Nov. 15, 2024. doi: 10.5772/intechopen.1007766.

[16] S. Hutchinson et al., “Investigating Wearable Fitness Applications: Data Privacy and Digital Forensics Analysis on Android,” Applied Sciences, vol. 12, no. 19, p. 9747, Jan. 2022, doi: https://doi.org/10.3390/app12199747.

[17] A. Almogbil, A. Alghofaili, C. Deane, T. Leschke, A. Almogbil, and A. Alghofaili, “Digital Forensic Analysis of Fitbit Wearable Technology: An Investigator’s Guide,” 2020 7th IEEE International Conference on Cyber Security and Cloud Computing (CSCloud)/2020 6th IEEE International Conference on Edge Computing and Scalable Cloud (EdgeCom), Aug. 2020, doi: https://doi.org/10.1109/cscloud-edgecom49738.2020.00017.

[18] A. MacDermott, S. Lea, F. Iqbal, I. Idowu, and B. Shah, “Forensic Analysis of Wearable Devices: Fitbit, Garmin and HETP Watches,” 2019 10th IFIP International Conference on New Technologies, Mobility and Security (NTMS), Jun. 2019, doi: https://doi.org/10.1109/ntms.2019.8763834.

[19] I. Baggili, J. Oduro, K. Anthony, F. Breitinger, and G. McGee, “Watch What You Wear: Preliminary Forensic Analysis of Smart Watches,” 2015 10th International Conference on Availability, Reliability and Security, Aug. 2015, doi: https://doi.org/10.1109/ares.2015.39.

[20] M.-J. Kim, Y. Shin, W. Jo, and T. Shon, “Security Analysis of Smart Watch and Band Devices,” 2021 International Conference on Computational Science and Computational Intelligence (CSCI), Dec. 2021, doi: https://doi.org/10.1109/csci54926.2021.00172.

[21] L. Caviglione, S. Wendzel, and W. Mazurczyk, “The Future of Digital Forensics: Challenges and the Road Ahead,” IEEE Security & Privacy, vol. 15, no. 6, pp. 12–17, Nov. 2017, doi: https://doi.org/10.1109/msp.2017.4251117.

[22] A. W. CNN, “Police use murdered woman’s Fitbit to charge her husband,” CNN. https://edition.cnn.com/2017/04/25/us/fitbit-womans-death-investigation-trnd/index.html

[23] S. Parikh, D. Chavda, S. Chakraborty, Dr. P. H. Rughani, and Dr. M. S. Dahiya, “Analysis of Android Smart Watch Artifacts,” International Journal of Scientific and Engineering Research, vol. 6, no. 8, pp. 920–930, Aug. 2015, doi: https://doi.org/10.14299/ijser.2015.08.011.

[24] A. MacDermott, T. Baker, and Q. Shi, “Iot Forensics: Challenges for the Ioa Era,” 2018 9th IFIP International Conference on New Technologies, Mobility and Security (NTMS), Feb. 2018, doi: https://doi.org/10.1109/ntms.2018.8328748.

[25] C. Hassenfeldt, S. Baig, I. Baggili, and X. Zhang, “Map My Murder,” Proceedings of the 14th International Conference on Availability, Reliability and Security, Aug. 2019, doi: https://doi.org/10.1145/3339252.3340515.

[26] R. J and G. Z, “Extraction and Forensic Analysis of Artifacts on Wearables,” International Journal of Forensic Science & Pathology, pp. 312–318, Jan. 2017, doi: https://doi.org/10.19070/2332-287x-1700070.

[27] J. P. van Zandwijk and A. Boztas, “Digital traces and physical activities: opportunities, challenges and pitfalls,” Science & Justice, vol. 63, no. 3, pp. 369–375, Apr. 2023, doi: https://doi.org/10.1016/j.scijus.2023.04.002.

[28] J. P. van Zandwijk and A. Boztas, “The iPhone Health App from a forensic perspective: can steps and distances registered during walking and running be used as digital evidence?,” Digital Investigation, vol. 28, pp. S126–S133, Apr. 2019, doi: https://doi.org/10.1016/j.diin.2019.01.021.

Broadband and Forensics: Accelerating Digital Investigations in a Connected World

Veneeth Iyengar[2]

[2]Executive Director, ConnectLA, Louisiana, USA.

Contributing authors: iyengar.veneeth@gmail.com;

Abstract

This paper explores the emerging role of broadband connectivity as both an enabler of crime and a catalyst for forensic science. With increasing digitization, forensic workflows are undergoing a transformation powered by high-speed internet access. We discuss the various tools and innovations supported by broadband, the challenges they pose, and their implications for justice delivery, especially in rural and underserved regions.

Keywords: BNSS, BNS, BSA, Certificate, Integrity, Hashing, Digital Evidence

1 Introduction

The digital revolution has profoundly reshaped the landscape of crime and criminal investigation. The proliferation of broadband internet—characterized by high-speed, always-on connectivity—has created new opportunities for both malicious actors and law enforcement agencies. On one hand, broadband facilitates a wide range of cyber-crimes such as ransomware attacks, identity theft, phishing, and the dissemination of deepfake content. On the other, it enables unprecedented advancements in digital forensics, remote investigations, and real-time evidence processing.

Broadband, with its ability to support the transfer of large volumes of data and seamless communication across geographic boundaries, has become an essential tool in the modern forensic toolkit. It allows investigators to access and analyze digital evidence remotely, collaborate across jurisdictions, and employ AI-powered forensic tools that rely on cloud-based infrastructures. In rural and underserved areas, where traditional forensic facilities may be limited, broadband connectivity can bridge significant gaps by providing remote access to expert testimony, digital labs, and case management platforms.

Furthermore, broadband contributes to streamlining the judicial process. Digital case files, online legal portals, and virtual courtrooms powered by reliable internet connectivity can enhance the speed and accessibility of justice delivery. It also opens avenues for training and capacity building in digital forensics through virtual platforms and e-learning tools.

This paper aims to explore the dual nature of broadband as both a challenge and a catalyst in the forensic sciences. It delves into the evolving crime landscape shaped by digital connectivity, outlines broadband-enabled forensic technologies, and highlights the transformative potential of broadband in ensuring equitable access to justice, especially in remote regions. By addressing the ethical, technical, and infrastructural considerations, this paper provides a comprehensive understanding of broadband's growing influence on forensic practices in a connected world.

2 What is Broadband?

Broadband is a term used to describe high-speed internet access that is continuously available and significantly faster than traditional dial-up connections. Unlike the dial-up systems of the past, broadband does not interrupt telephone usage and allows for the simultaneous transmission of large quantities of data. The core technologies used to deliver broadband include Digital Subscriber Line (DSL), cable modems, fiber-optic connections, satellite services, and mobile networks such as 4G and the more recent 5G.

Each of these broadband types comes with its own strengths and limitations. Fiber-optic connections offer the highest speeds and reliability, ideal for handling data-intensive forensic applications such as live imaging, cloud-based analytics, and video surveillance streams. Satellite broadband, though slower and more prone to latency, can provide vital connectivity in remote areas lacking ground infrastructure. Mobile broadband, particularly 5G, introduces ultra-low latency and increased bandwidth, supporting real-time forensic analysis even on the move.

Broadband's significance in forensic investigations lies in its capacity to enable instantaneous communication, remote access to forensic tools, and the secure transfer of large digital evidence files. With growing adoption globally, broadband is increasingly becoming a foundational layer for modern digital forensics infrastructure [1].

3 The Digital Crime Landscape

The availability and ubiquity of broadband have dramatically transformed the crime landscape. High-speed connectivity has enabled the rise of sophisticated cybercrimes that can be orchestrated from virtually anywhere in the world. Threat actors exploit broadband networks to distribute malware, launch ransomware attacks, conduct phishing schemes, perpetrate identity theft, and spread misinformation using AI-generated content such as deepfakes.

Ransomware has become one of the most prominent broadband-enabled threats. Cybercriminals deploy malicious software that encrypts victims' data and demand cryptocurrency payments for restoration. These attacks are often executed through high-speed networks that allow the rapid dissemination of malware across multiple systems and geographies. Similarly, phishing attacks, which rely on deceiving users into revealing sensitive information, leverage broadband to propagate spoofed websites and mass email campaigns at scale.

Another disturbing trend is the emergence of deepfake technology—AI-generated synthetic media that convincingly alters audio or video content. Deepfakes pose a serious challenge to the authenticity of digital evidence, enabling false impersonation and undermining public trust in video surveillance and courtroom testimonies.

The forensic response to these digital crimes relies heavily on broadband to collect, preserve, and analyze digital traces. Logs of IP addresses, timestamps of activity, metadata from files and devices, and communication records such as emails and chat histories all form essential elements of digital evidence. As broadband enables perpetrators to mask their identities and operate across borders, forensic professionals must leverage the same connectivity to detect, attribute, and investigate these offenses [2].

4 Broadband-Enabled Forensic Tools

High-speed connectivity has enabled a new class of forensic tools that operate remotely and in real-time. Some of the key broadband-driven innovations include:

- **Cloud-based Digital Evidence Management Systems (DEMS):** Securely storing, sharing, and analyzing digital evidence across jurisdictions.
- **Remote Mobile Forensics:** Broadband allows investigators to access and extract mobile data from distant locations using encrypted connections.
- **AI-Powered Network Traffic Analysis:** Detects anomalies and intrusions by analyzing live network data.
- **Live Disk Imaging:** Real-time duplication and analysis of storage devices via secure broadband channels [3].

5 Broadband Access and Rural Justice

Broadband access also plays a democratizing role in digital forensics. In rural and remote areas where physical access to forensic labs is limited, broadband enables:

- **Remote Expert Testimony:** Experts can provide insights via video conferencing.
- **Cloud-Based Laboratories:** Virtual labs for digital evidence analysis.
- **Fast Case File Access:** Immediate access to shared documents, improving judicial efficiency.

6 Challenges and Ethical Considerations

While broadband connectivity has enhanced forensic capabilities, it also introduces a spectrum of technical, ethical, and legal challenges that must be addressed to ensure responsible and equitable use.

Privacy and Security: The handling of digital evidence over broadband networks raises serious concerns regarding confidentiality and data integrity. If not adequately encrypted or protected by secure transmission protocols, sensitive information such as personal communications, financial records, or digital surveillance footage may be intercepted, altered, or leaked. Cybersecurity protocols must be integrated into every stage of the forensic workflow to protect against unauthorized access and tampering.

Legal Admissibility: For digital evidence to be admissible in court, it must maintain a clearly documented and unbroken chain of custody. However, remote handling and transmission of evidence over broadband networks complicate this requirement. Forensic investigators must adopt rigorous documentation and digital signing protocols to demonstrate the integrity and authenticity of evidence collected, transmitted, or analyzed remotely.

Infrastructure Disparity: Access to high-speed internet remains uneven, particularly in developing regions or marginalized communities. This disparity restricts the implementation of broadband-enabled forensic systems in areas that may benefit most from such capabilities. The uneven development of digital infrastructure risks creating a justice divide, where rural or underserved regions are unable to leverage the same forensic tools available to urban centers.

Digital Divide and Training: In addition to infrastructure limitations, the lack of digital literacy and training among law enforcement and forensic personnel further hampers the effective utilization of broadband technologies. Specialized training in digital forensic protocols, cybercrime investigation, and secure communication is essential to equip professionals with the skills needed to operate in broadband-enhanced environments [4].

Addressing these challenges requires a coordinated effort involving policymakers, technology developers, forensic experts, and educators to ensure that the benefits of broadband in forensics are realized responsibly and inclusively.

7 Conclusion

Broadband technology stands as a transformative force in the field of forensic science, enabling faster, more decentralized, and widely accessible investigative processes. From real-time mobile forensics to cloud-based evidence management, broadband has reshaped how law enforcement agencies gather, analyze, and share digital evidence across jurisdictions. It facilitates collaboration between remote experts, enhances judicial efficiency through virtual case processing, and ensures that even rural or underserved communities have a pathway to justice through digital means.

However, the potential of broadband cannot be fully realized without addressing its associated challenges. Investments must go beyond mere infrastructure deployment to encompass robust cybersecurity safeguards, training programs for forensic professionals, and legal frameworks that adapt to the dynamic nature of digital evidence.

As we move into an era of AI-enabled and data-driven justice systems, broadband will remain at the heart of forensic innovation. It is imperative for stakeholders—governments, academic institutions, forensic agencies, and private sector partners—to work together in shaping a secure, inclusive, and ethically sound broadband-powered forensic ecosystem. With the right policies and partnerships in place, broadband can not only accelerate investigations but also uphold the core principles of fairness, transparency, and justice in the digital age.

References

[1] Federal Communications Commission, "Broadband Speed Guide," 2024. https://www.fcc.gov

[2] Europol, "Internet Organised Crime Threat Assessment (IOCTA)," 2023. https://www.europol.europa.eu

[3] Eoghan Casey, *Digital Evidence and Computer Crime: Forensic Science, Computers and the Internet*, Academic Press, 2019.

[4] Nir Kshetri, "Cybersecurity and Digital Forensics in the Global South," *Journal of Global Information Technology Management*, 2020.

[5] Yashas Hariprasad, Latesh Kumar K. J., Suraj L., and S. S. Iyengar, "Boundary Based Fake Face Anomaly Detection in Videos using Recurrent Neural Networks," *Springer Nature*, 2024.

[6] Yashas Hariprasad, S. S. Iyengar, "AI Powered Correlation Technique to Detect Virtual Machine Attacks in Private Cloud Environment," *Springer Nature*, 2025.

[7] S. S. Iyengar, Yashas Hariprasad, Sina Nabavi, Prasad H. B., *Artificial Intelligence in Practice: Theory and Applications for Cyber Security and Forensics*, Springer Nature, forthcoming November 2024.

[8] S. S. Iyengar and Richard Brooks, *Distributed Sensor Networks: Sensor Networking and Applications*, CRC Press, 2009.

[9] Singaram, Jayakumar, S. S. Iyengar, and Azad M. Madni, *Deep Learning Networks: Design, Development and Deployment*, Springer Nature 2023.

[10] Shi, Bin, and S.S. Iyengar, *Mathematical theories of machine learning-Theory and applications*, Springer Nature, 2020.