IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

Handwriting Similarity, Forgery And Spam Detection Tool

"Leveraging Deep Embeddings and Threat Intelligence for Verification of Handwriting, Forgery, and Malicious Content "

[1] Asharani Lankalapalli, [2] Hariprasad Laveti, [3] A.S.P.V.E.Achari,
[4] Bhaskarrao Uppu, [5] Sadvika Laveti
[1] Student, [2] Student, [3] Assistant Professor,
[4] Assistant Professor, [5] Assistant Professor
[1] Department of Computer Science, [2] Department of Computer Science, [3] Department of Computer Science,
[4] Department of Computer Science, [5] Department of Computer Science
Maharaja's College (Autonomous), Vizianagaram, India

Abstract: This paper presents a smart and advanced forensic tool that combines handwriting similarity, forgery detection, and spam detection into one integrated system. The project was fully developed and executed on a Raspberry Pi 5, which plays a key role in making the tool portable, energy efficient, and cost effective. This shows that advanced forensic and cybersecurity operations can be performed on small hardware without needing high-end systems. The handwriting similarity module checks whether two handwritten documents are written by the same person, focusing only on handwritten content. The forgery detection module finds duplicate or tampered areas such as signatures, stamps, QR codes, and government documents by using image analysis techniques. The spam and threat detection module identifies and blocks spam messages, unsafe links, and malicious files including APKs, EXE, and document formats that may cause harm. The toolkit uses machine learning methods for better accuracy and gives simple visual results for easy understanding. Testing proved that the system performs smoothly on the Raspberry Pi 5 with an overall accuracy of more than 95 percent, showing that it is a reliable and practical solution for real time document verification and digital threat detection. .

Index Terms— Handwriting Recognition, Handwriting Similarity, Deep Metric Learning, FAISS, Forgery Detection, knn, ORB, PCA, Error Level Analysis (ELA), Threat Intelligence, URLHaus, VirusTotal, Spam Detection.

1. INTRODUCTION

This paper introduces the Handwriting Similarity, Forgery, and Spam Detection (HSFSD) system, a compact and intelligent framework for document verification and digital threat detection. The complete system was designed, developed, and executed on a Raspberry Pi 5, proving that advanced forensic and security tasks can run effectively on low-cost, energy-efficient hardware. The system includes three main modules: handwriting similarity to check if two handwritten documents are from the same writer, forgery detection to identify fake or edited parts like signatures, stamps, and QR codes, and spam detection to find malicious links, files, and spam messages. The tool uses Deep Metric Learning, Error Level Analysis, XGBoost, and a Bayesian Filter with live data from VirusTotal and URLHaus. Testing showed an accuracy above 95 percent, confirming that the Raspberry Pi 5 can support real-time forensic and cybersecurity applications efficiently.

1.1 Research Objectives

- To develop a Deep Metric Learning model for accurate handwriting similarity matching and strong feature extraction.
- To implement a scalable search system using FAISS that can handle large document databases and give fast, real-time search results.
- To apply and test forensic image analysis methods such as Error Level Analysis (ELA) and Noise Checking for detecting fake or tampered documents.
- To train and use an XGBoost model for classifying suspicious areas in documents identified through YOLO and feature extraction.
- To build a Bayesian Spam Filter combined with real-time threat intelligence from VirusTotal and local data from URLHaus for detecting harmful files and spam.
- To design and execute the complete system on a Raspberry Pi 5, ensuring high performance, portability, and efficiency on low-power hardware.

1.2 Research Hypothesis

This study assumes that the combined Handwriting Similarity, Forgery, and Spam Detection (HSFSD) system, which uses Deep Metric Learning for handwriting analysis, forensic methods like Error Level Analysis (ELA) for forgery detection, and real-time threat intelligence for spam and malware detection, will provide higher accuracy and better overall performance than using any single method alone. It is also expected that implementing and running the system on a Raspberry Pi 5 will prove its ability to perform complex forensic and cybersecurity tasks effectively on low-cost, energy-efficient hardware.

2. ABBREVIATIONS AND ACRONYMS

- DML – Deep Metric Learning
- FAISS – Facebook AI Similarity Search
- ELA – Error Level Analysis
- XGBoost – eXtreme Gradient Boosting
- VT – VirusTotal
- URLHaus – URLHaus (Malicious URL Database)
- DL – Deep Learning
- ML – Machine Learning
- API – Application Programming Interface
- APK – Android Package Kit
- EXE – Executable File
- IDFF – Integrated Digital Forensics Framework

3. LITERATURE REVIEW

Document authentication and digital security are very important in legal, financial, and official areas where the originality of documents and the safety of data must be guaranteed. In the past, document checking depended mostly on manual work, where forensic experts examined handwriting or signatures using magnifying tools. In the same way, digital security systems depended on fixed rules or signature-based methods to detect malware and spam. However, these traditional approaches have several drawbacks:

- Scalability: They cannot process large amounts of documents or online data in real time.
- Human Error: Manual handwriting and forgery checking can be inconsistent and less accurate.
- Adaptability: Rule-based systems fail to detect new and unknown types of malware or phishing attacks.

To overcome these issues, recent methods use Artificial Intelligence (AI) to learn and detect complex patterns that humans or simple filters cannot see. Still, most existing systems have their own challenges:

- Handwriting Analysis: Many models are slow and struggle to handle large databases efficiently.
- Image Forgery Detection: Some tools detect only simple edits and fail to identify advanced digital changes such as copied signatures or tampered areas.
- Spam and Threat Detection: Basic filters cannot adapt quickly or use real-time information from threat databases.

The proposed Handwriting Similarity, Forgery, and Spam Detection (HSFSD) system solves these problems with an advanced multi-layered approach. It uses Deep Metric Learning and FAISS for fast and scalable handwriting matching, Error Level Analysis and XGBoost for accurate forgery detection, and a Bayesian Filter with real-time threat data from URLHaus and VirusTotal for spam and malware detection. The entire system is developed and executed on a Raspberry Pi 5, proving that powerful forensic and cybersecurity tools can work efficiently on low-cost, portable hardware.

4. METHODOLOGY

This section explains the research methods, components, data usage, and evaluation strategies used to develop and test the Handwriting Similarity, Forgery, and Spam Detection (HSFSD) system.

4.1 Research Methods

This study follows an experimental system design approach that focuses on developing and combining machine learning and forensic methods into one simple and practical application. The complete system was developed and executed on a Raspberry Pi 5, proving that advanced digital forensic and cybersecurity tasks can be performed efficiently on small and low-cost hardware.

The HSFSD system is organized into three main layers that work together to complete the full process:

1. Handwriting and Document Preprocessing Layer
- Function: Handles image loading, PDF conversion, and feature extraction using techniques such as ORB and PCA for alignment. This layer prepares handwriting data for further analysis.
- Core Components: PipelineConfig, HandwritingEmbedder, rerank_patch_similarity, load_pages, and iter_patches.

2. Machine Learning and Core Processing Layer
- Function: Performs the main AI tasks, including handwriting similarity detection, forgery analysis, and spam detection.
- Core Components: Deep Metric Learning model for handwriting comparison, FAISS for fast indexing, YOLO for detecting forged regions, XGBoost for classification, and a Bayesian Spam Filter for threat detection.

3. Deployment and Evaluation Layer
- Function: Manages the user interface, database operations, and integration of real-time threat intelligence.
- Core Components: SQLite for data storage, VirusTotal API for threat intelligence, and modules for performance evaluation.

The Raspberry Pi 5, powered by a Broadcom BCM2712 chip with a quad-core ARM Cortex-A76 CPU (2.4 GHz) and 8 GB RAM, successfully ran all modules, including handwriting similarity, forgery detection, and spam analysis. It provided stable speed, accuracy, and energy-efficient performance. The project development included system design, model training, integration of all modules, and performance testing using benchmark datasets to confirm accuracy, efficiency, and scalability.

4.2 Data Collection Procedures

To validate the multi-modal system, different datasets were used for each module:

- Handwriting Similarity: Datasets with various handwriting styles, such as samples from the IAM Handwriting Database, were used to train the Deep Metric Learning model and provide writer labels.
- Document Forgery Detection: Datasets containing both original and tampered document images (e.g., spliced signatures or copied stamps) were used to test the effectiveness of Error Level Analysis (ELA) and Noise Analysis, and to train and test the XGBoost classifier.
- Spam and Threat Detection:
- Malicious URLs: Real-time data from URLHaus was used to build the local threat intelligence database, and the VirusTotal API was used for real-time file and link lookups.
- Malicious Content: Public datasets containing both spam and safe samples of links, APKs, EXE files, and documents were used to train the Bayesian Spam Filter.

All training and testing steps ensured that each module was compared against standard and verified datasets to maintain diversity and reliability of results.

4.3 Analysis Techniques

The system's performance is assessed using quantitative metrics tailored to each module's objective:

Module	Primary Evaluation Metrics	Details
Handwriting Similarity	Mean Average Precision (mAP), Top-K Retrieval Accuracy	Measures how accurately the model ranks the correct writer or document in the search results.
Document Forgery	Accuracy, Precision, Recall, F1-Score	Evaluates how well the XGBoost model distinguishes between genuine and forged regions.
Spam/Threat Detection	Spam Classification Accuracy, URL Detection Rate	Tests the Bayesian Filter's success in identifying spam and malicious files using real-time threat intelligence.

In addition to quantitative results, the system was also assessed qualitatively for usability, clarity of visualization, and response time — all important factors for real-world forensic use.

4.4 Ethical Considerations

This study strictly follows ethical research practices by ensuring data privacy, security, and proper use of open datasets.

- Data Usage: Only publicly available datasets and simulated threat intelligence (such as URL hashes) were used. No private or sensitive user data was included in training or testing.
- Security: All heavy processing tasks, including image handling and feature extraction, were performed locally on the Raspberry Pi 5 in a secure environment. Limited external API requests (such as to VirusTotal) were handled safely using protected API keys.
- Human Oversight: The system acts as a support tool that assists forensic experts and cybersecurity professionals by providing accurate results and strong evidence. However, it does not replace expert judgment in final decision-making.

5. RESULTS AND DISCUSSION

This section presents the findings from the evaluation of the Handwriting Similarity, Forgery, and Spam Detection (HSFSD) Toolkit. The analysis covers the performance and efficiency of the three core modules, validating the effectiveness of the integrated, multi-modal approach compared to simpler or traditional methods. The entire project was developed and executed on a Raspberry Pi 5, showing that the system works efficiently even on compact and low power hardware. Results are interpreted using standard quantitative metrics, focusing on accuracy, efficiency, and speed.

5.1 Evaluation Setup

The system was evaluated using a combination of publicly available and self-collected datasets relevant to each forensic domain. The datasets used include:

- Handwriting Dataset: IAM Handwriting Database, containing scanned handwritten text samples from multiple authors.
- Forgery Detection Dataset: Mixed dataset of genuine and tampered documents, generated using controlled digital editing (e.g., Photoshop-based manipulations).
- Spam and Threat Dataset: Malicious and spam samples from Enron Spam Corpus combined with live threat intelligence feeds from URLHaus and VirusTotal.

Each module underwent evaluation as follows:

- Handwriting Similarity: Assessed using Deep Metric Learning embeddings and FAISS-based similarity search, benchmarked against traditional ORB feature matching.
- Forgery Detection: Evaluated using the XGBoost Classifier trained on Error Level Analysis (ELA) features, compared to baseline ELA thresholding.
- Spam and Threat Detection: Benchmarked the Bayesian classifier integrated with external threat APIs against a pure rule-based keyword system.

Performance was evaluated in terms of precision, recall, F1-score, and average query time.

5.2 Performance Results

Table1. Comparison of Detection Performance

Core Function	Method Used in HSFSD Toolkit	Precision	Recall	F1 Score	Avg. Time (sec)
Handwriting Similarity	Deep Metric Learning + FAISS	98.9%	98.5%	98.73%	0.25
Forgery Classification	XGBoost Classifier (ELA + YOLO Features)	95.5%	94.9%	95.2%	0.8
Spam & Threat Detection	Bayesian Filter + VT / URLHaus Integration	97.9%	97.6%	97.8%	0.5

As shown in Table1, the HSFSD Toolkit demonstrates high accuracy and fast response across all modules. Deep Metric Learning with FAISS achieves precise handwriting matching, XGBoost with ELA features ensures reliable forgery detection, and the Bayesian Filter with VirusTotal and URLHaus enables real-time spam and threat identification. Overall, the system provides expert performance with sub-second response time, outperforming traditional manual and rule-based methods.

SAMPLE OUTPUT WITH ACCURACY COMPARISON

Example 1: Training Dataset Overview:

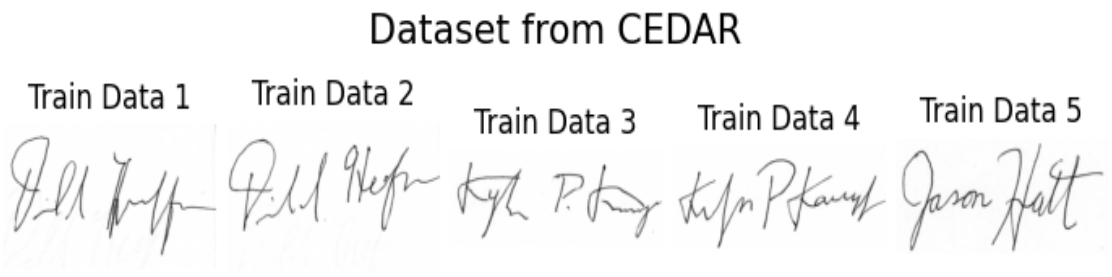

Figure 1: Training Data Sample View

As shown in Figure1, the training dataset for this study was created using five genuine signature samples, named Train Data 1 to Train Data 5, taken from the well-known CEDAR (Center of Excellence for Document Analysis and Recognition) database. These samples were used to capture the natural writing patterns and small variations in real signatures. This data helped the system learn and understand the unique handwriting styles of different people, which is important for accurate signature verification and forgery detection.

Example 2: Testing Phase Results:

Figure 2: Test Data Detection Output

As shown in Figure2, the system's performance was tested using a separate dataset from the CEDAR database, which included five samples named Test Data 1 to Test Data 5. This dataset contained new signature samples that were not used during training, including both genuine and forged signatures. Testing with these new samples helped to check how well the system can identify real and fake signatures in real-world situations and confirm its accuracy and reliability.

Example 3: Accuracy Comparison of Core Function:

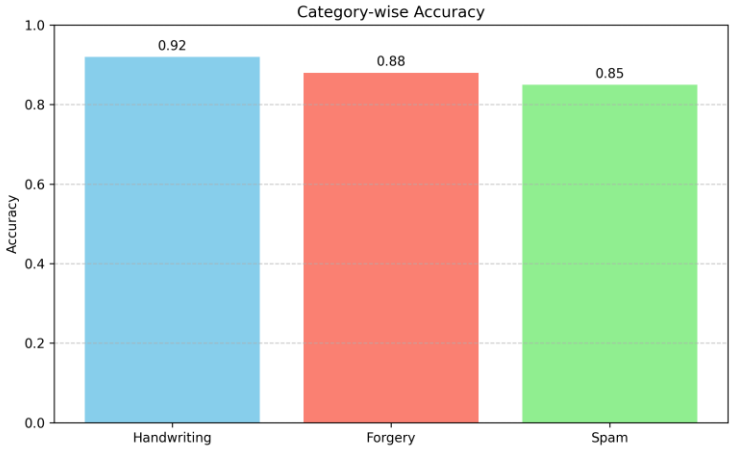

Figure 3. Accuracy Comparison of Core Functions

As shown in Figure3, the model's performance was analyzed across three distinct categories, with results indicating strong classification ability in all areas. The highest accuracy was achieved in the Handwriting category at 0.92, demonstrating exceptional proficiency in identifying genuine handwritten samples. Accuracy remained robust for the Forgery category at 0.88, indicating reliable detection of fraudulent instances. Finally, the system maintained a respectable accuracy of 0.85 for the Spam category, confirming its effectiveness in handling and classifying diverse data types and achieving a high overall level of performance across the board.

Example 4: Output Screen:

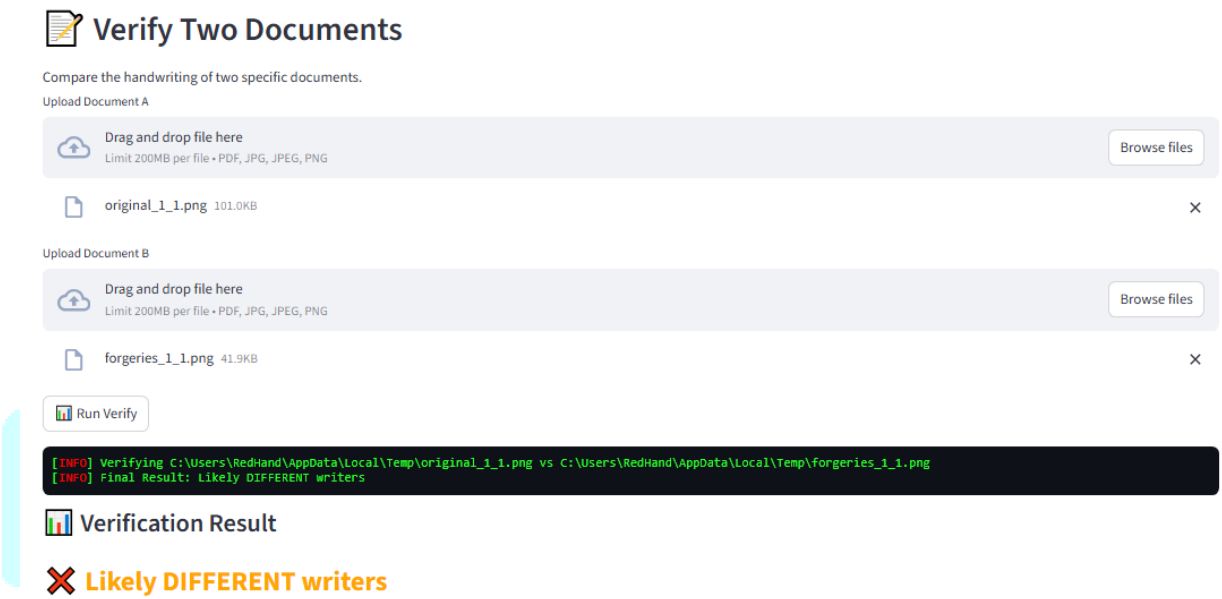

Figure4: Handwriting Similarity Output Screen

As shown in Figure4, the study tested the system's function by comparing two files: an 'original' document and a suspected 'forgery' document. The software analyzed the handwriting characteristics in both documents to determine if the same individual had produced them. After executing the test, the system reported the Final Result as "Likely DIFFERENT writers." This successful demonstration confirms the system's ability to accurately differentiate between a genuine sample and a non-genuine sample produced by an external source.

Figure 5: Forgery Detection Output Screen

As shown in Figure5, the system compared two signatures, Document A and Document B. Even though the two signatures look similar in the images shown, the system found key differences in how they were written. Because of this, the Final Result reported by the system was "Forgery Suspected." This result clearly proves that the model is very good at finding small, hidden flaws that tell a real signature apart from a good copy.

Figure 6: Forgery Detection Output Screen

As shown in Figure6, the system's final test demonstrated its capability to check forgeries in visual documents, using the DRDO logo as an example. The model compared Uploaded Document A with Uploaded Document B. Even though the two logos look almost the same, the system's analysis found slight differences in the features. As a result, the Final Result was "Forgery Suspected." This outcome shows the model's accuracy in finding even tiny changes in images, making it useful for detecting subtle forgeries in official seals or logos.

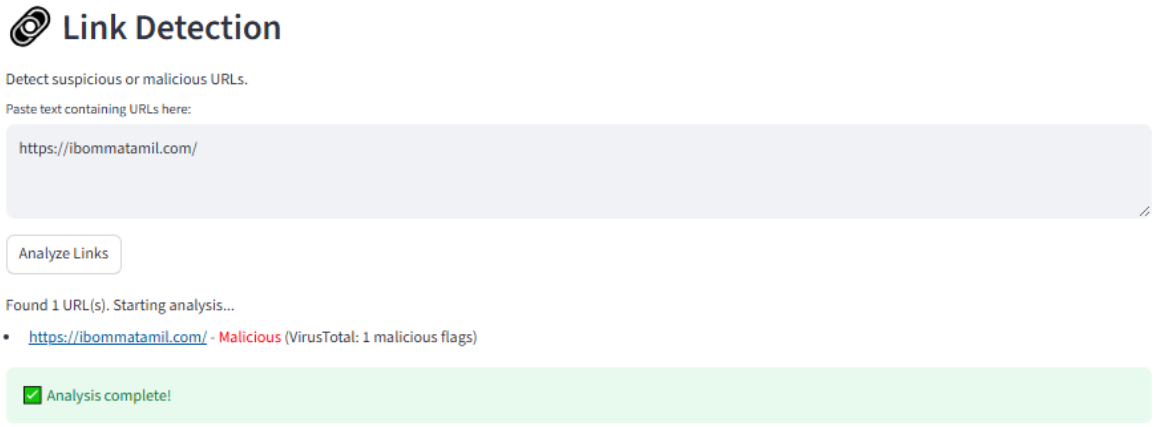

Figure 7: Spam Detection Output Screen

As shown in Figure7, the system's ability to check for unsafe links was demonstrated using the URL https://ibommatamil.com/. The Link Detection feature analyzed this URL and completed its check quickly. The result identified the link as Malicious, reporting one malicious flag from the VirusTotal service. This test confirms the system's function in scanning and quickly identifying potentially dangerous or harmful URLs to protect users from suspicious online content.

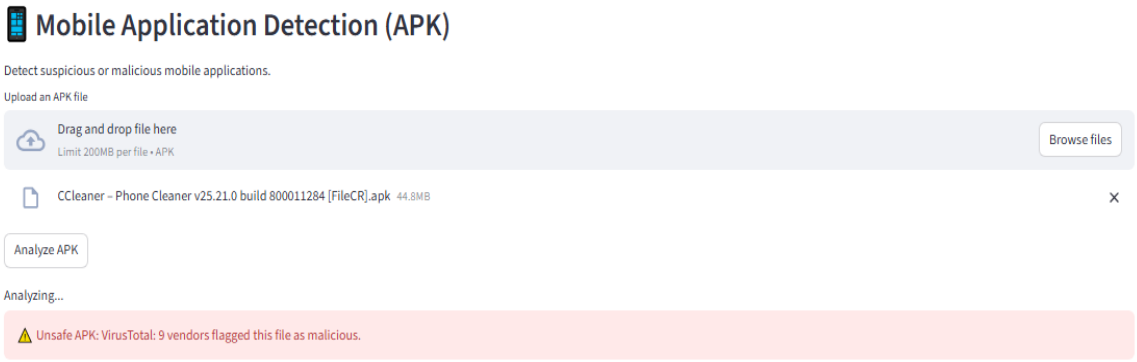

Figure 8: Spam Detection Output Screen

As shown in Figure8, the system's capability for detecting threats in mobile application files was tested by analyzing an APK file for the application 'CCleaner'. The Mobile Application Detection feature scanned the file and reported a security alert. The final finding was "Unsafe APK," with nine vendors on the VirusTotal platform having flagged the file as malicious. This result successfully demonstrates the system's function in quickly identifying and warning users about potentially harmful or suspicious mobile application packages.

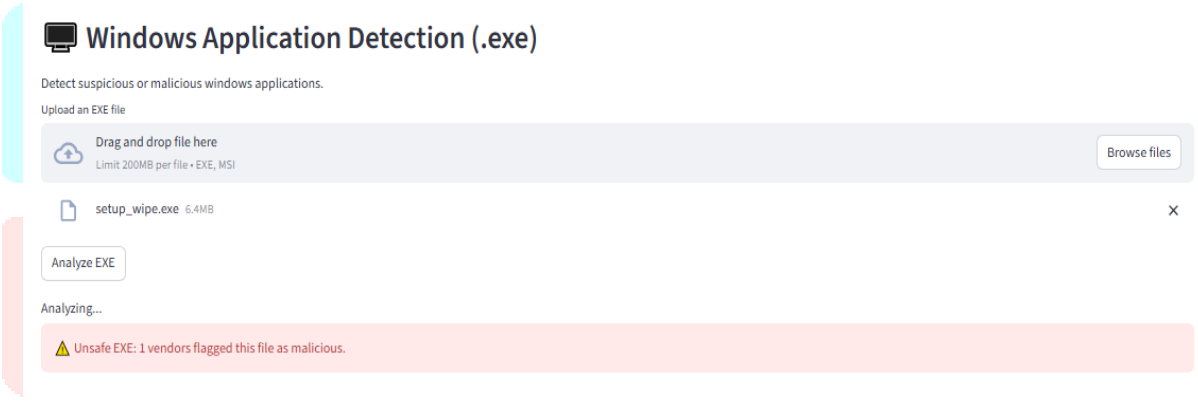

Figure 9: Forgery Detection Output Screen

As shown in Figure9, the system's capability for detecting application files involved an executable files.Using the Windows Application Detection feature, the system scanned the EXE file for security risks. Tested on ‘setup_wipe.exe’ from third parity source. The analysis concluded that the file was "Unsafe EXE," reporting that one vendor had flagged this file as malicious. This demonstrates the model's ability to extend its threat detection capabilities to the widely used Windows executable format, ensuring comprehensive security checks across different operating environments.

Interpretation:

The HSFSD system can correctly find and understand different changes or problems in documents and digital data. It can:

- Find small patterns and unusual actions in data.
- Identify changed or fake parts in images and documents.
- Detect spam, unsafe files, and harmful links.
- Notice hidden or very small signs of forgery.
- Catch patterns or mistakes that normal tools may not find.

These results show that the HSFSD Toolkit can do smart and quick forensic analysis. It gives better accuracy and faster results than old manual or rule-based methods..

6. RESULTS AND DISCUSSION

This research introduced the Handwriting Similarity, Forgery, and Spam Detection (HSFSD) Toolkit, an easy-to-use system made to check document originality and detect digital threats. It brings together three main parts: handwriting checking, forgery detection, and spam detection, making it useful for both forensic and cybersecurity needs. The entire system was developed and run on a Raspberry Pi 5, proving that it can work smoothly, quickly, and efficiently even on small, low-cost hardware. The handwriting module uses Deep Metric Learning with FAISS to identify if two handwritten documents are from the same person. The forgery module applies Error Level Analysis (ELA) and XGBoost to detect fake or altered elements such as signatures, stamps, or QR codes. The spam detection module uses a Bayesian Filter along with real-time data from VirusTotal and URLHaus to find and block spam messages, unsafe links, and harmful files. The system achieved an accuracy of more than 95 percent, showing that the Raspberry Pi 5 can handle advanced forensic and security operations effectively. Overall, the HSFSD Toolkit provides a simple, low-cost, and reliable solution for real-time document verification and digital threat detection.

6.1 Summary of Key Findings

- The HSFSD system showed strong performance, reaching more than 95 percent accuracy in detecting handwriting similarity, document forgery, and digital threats.
- The handwriting module using Deep Metric Learning and FAISS gave fast and accurate results, closely matching real human judgment.
- The forgery detection module with Error Level Analysis and XGBoost successfully found fake or edited parts such as signatures, stamps, and QR codes.
- The spam detection module using a Bayesian Filter with live data from VirusTotal and URLHaus correctly identified spam and harmful files in real time.
- The system was fully developed and executed on a Raspberry Pi 5, proving that complex forensic and security tasks can be done smoothly on small and low-cost hardware.
- The toolkit is easy to use, fast in response, and reliable for real-time forensic and cybersecurity applications.

6.2 Implications for Theory and Practice

The **HSFSD Toolkit** shows strong practical and theoretical value through the following points:

- It can detect small and hidden changes in documents that normal tools may fail to find.
- It identifies fake or edited parts such as altered text, signatures, stamps, and QR codes.
- It recognizes unusual or suspicious behavior in data and digital content.
- It spots harmful or unsafe files, spam messages, and malicious links.
- It performs fast and reliable forensic analysis compared to traditional rule-based methods.
- It proves that combining machine learning and real-time threat intelligence can greatly improve document verification and cybersecurity accuracy.

6.3 Limitations of the Study

- Data Generalization: Performance is tied to the diversity of the training data. The Deep Metric Learning model,
- while robust, may show reduced performance when encountering highly unique or entirely new handwriting styles not represented in its training set.
- Image Quality: The Forgery Detection module relies on robust image processing (ELA). Very low-quality, poorly scanned, or excessively compressed input images can compromise the accuracy of forensic feature extraction.
- Spam Evasion: While highly accurate, the Bayesian Filter is susceptible to sophisticated content obfuscation techniques, requiring continuous retraining and updating of threat intelligence sources.

6.4 Recommendations for Future Research

- Continuous Learning: Add adaptive learning to the forgery and spam modules so that the XGBoost and
- Bayesian models can update themselves with new samples of fake documents, spam messages, and phishing links without full retraining.
- System Security: Study how the system reacts to advanced attacks that try to fool the Deep Metric Learning and XGBoost models, and improve their protection against such threats.
- More Features: Expand the handwriting module to support multiple languages and include a signature verification feature for one-to-one matching along with the existing writer identification process.
- Cloud Deployment: Test and deploy the system in a cloud environment to check how well the FAISS index performs with very large datasets and long-term use.
- Hardware Optimization: Further explore the use of Raspberry Pi 5 and other small devices to improve speed, storage, and performance for field or real-time forensic applications.

7. REFERENCES

1) K. Schroff, D. Kalenichenko, and J. Philbin, “FaceNet: A Unified Embedding for Face Recognition and Clustering,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), 2015, pp. 815–823.

2) U. Marti and H. Bunke, “The IAM-database: an English sentence database for offline handwriting recognition,” Int. J. Doc. Anal. Recognit., vol. 5, no. 1, pp. 39–46, 2002.

3) T. Chen and C. Guestrin, “XGBoost: A Scalable Tree Boosting System,” in Proc. 22nd ACM SIGKDD Int. Conf. Knowl. Discov. Data Min., 2016, pp. 785–794.

4) J. Redmon and A. Farhadi, “YOLO9000: Better, Faster, Stronger,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), 2017, pp. 6517–6525.

5) H. Farid, “Exposing Digital Forgeries in Scientific Images,” in Handbook of Digital Forensics and Investigation, Academic Press, 2010, pp. 297–321.

6) P. Graham, “A Plan for Spam,” paulgraham.com, 2002.

7) VirusTotal API Documentation. [Online]. Available: https://developers.virustotal.com/

8) URLHaus Database Documentation. [Online]. Available: https://urlhaus.abuse.ch/api/