

From Pixels to Provenance: Harnessing Source Camera Fingerprints to Detect AI-Created Images

Patnala Naga Satya Sesha Venkata Charishma^{1,*} and N S C Mohana Rao²

¹ Department of Computer Science and Engineering, VSM College Of Engineering (A), Ramachandrapuram , Dr.B.R.Ambhedkar Konaseema Dt , Andhra Pradesh ,India – 533255.

² Department of Computer Science and Engineering, VSM College of Engineering (A) , Ramachandrapuram , Dr.B.R.Ambhedkar Konaseema Dt , Andhra Pradesh ,India - 533255

Corresponding Author: Patnala Naga Satya Sesha Venkata Charishma. Email: charishma5034@gmail.com

DOI: <https://doi.org/10.69996/jsihs.2026006>

Abstract: The rapid advancement of generative artificial intelligence has led to the widespread creation of highly realistic synthetic images, posing significant challenges to image authenticity, digital trust, and misinformation detection. Traditional visual inspection and content-based forensic techniques are increasingly inadequate in distinguishing AI-generated images from those captured by real-world cameras. This study presents a robust forensic framework that leverages source camera fingerprints to detect AI-created images by analyzing intrinsic device-level artifacts embedded during the image formation process. By extracting and modeling sensor-specific patterns such as Photo-Response Non-Uniformity (PRNU) and noise residuals, the proposed approach differentiates between images originating from physical imaging sensors and those synthesized by generative models. The framework integrates signal processing techniques with machine learning classifiers to evaluate provenance authenticity at the pixel level. Experimental results demonstrate that camera fingerprint-based analysis significantly improves detection accuracy, even for visually convincing AI-generated images and post-processed content. The findings highlight the effectiveness of provenance-aware forensics in combating deepfake imagery and reinforcing trust in digital media ecosystems. This work contributes toward the development of reliable image authentication systems essential for digital forensics, journalism, legal evidence verification, and secure multimedia communication.

Keywords: Digital Forensics, Journalism, Legal Evidence Verification, Photo-Response, Non-Uniformity

1. Introduction

In recent years, the rapid evolution of artificial intelligence, particularly deep learning-based generative models, has transformed the way digital images are created and consumed [1-2]. Advanced generative techniques such as Generative Adversarial Networks (GANs) and diffusion models are capable of producing highly realistic images that are often indistinguishable from photographs captured by real-world cameras [3-4]. While these technologies offer significant benefits in areas such as content creation, entertainment, and design, they also introduce serious challenges related to image authenticity, misinformation, copyright infringement, and digital trust [5].

Images have traditionally been regarded as reliable visual evidence in journalism, legal proceedings, scientific research, and social media communication. However, the emergence of AI-generated images has weakened this trust, as synthetic content can be easily misused to fabricate events, manipulate public opinion, or spread false information [6-7]. Conventional



image forensics techniques that rely on visual cues, metadata analysis, or manual inspection are increasingly ineffective against sophisticated AI-generated imagery, which often lacks obvious artifacts [8].

Digital image forensics has therefore shifted toward more intrinsic and device-level analysis methods to verify image provenance. One promising direction is the use of source camera fingerprints, which exploit unique sensor imperfections introduced during the image acquisition process [9-11]. These fingerprints act as a distinctive signature of a physical camera, enabling forensic analysts to determine whether an image originated from a real imaging sensor or was artificially generated. This chapter introduces the fundamental motivation, objectives, scope, and necessity of leveraging source camera fingerprints to detect AI-created images and ensure trustworthy digital media.

With the rapid proliferation of AI-generated images, ensuring the authenticity and origin of digital imagery has become a critical challenge. Existing forensic techniques that rely on visual artifacts, metadata, or model-specific traces are increasingly unreliable due to advancements in generative models and image post-processing techniques [12-13]. As a result, there is no universally reliable method to determine whether an image originates from a physical camera or is synthetically generated.

The problem addressed in this study is the lack of a robust, provenance-aware detection mechanism that can reliably identify AI-created images by analyzing intrinsic sensor-level characteristics rather than superficial visual features. Addressing this problem is essential for maintaining trust in digital media and preventing the misuse of synthetic imagery [14-15].

From a technological perspective, this study contributes to the advancement of digital forensics by emphasizing provenance-based analysis over appearance-based detection. By leveraging source camera fingerprints, the proposed approach offers a more resilient solution against evolving generative models. Moreover, the study supports the development of intelligent forensic systems that assist human experts rather than replace them. Such systems can be integrated into content verification platforms, social media moderation tools, and legal investigation workflows, ultimately helping to preserve the integrity, authenticity, and trustworthiness of digital images.

2. Literature Review

The introduction of Generative Adversarial Networks (GANs) marked a significant turning point in synthetic image creation. Studies by Goodfellow et al. demonstrated that GANs could generate highly realistic images by learning complex data distributions. Subsequent research focused on detecting GAN-generated images using frequency-domain analysis, color inconsistencies, and deep learning-based classifiers. Although these techniques achieved promising results, their effectiveness often diminished when images were generated using unseen architectures or subjected to post-processing operations.

To address the limitations of content-based detection, researchers began exploring intrinsic image properties related to the image acquisition process. One of the most influential contributions in this area is the concept of Photo-Response Non-Uniformity (PRNU), introduced by Lukas, Fridrich, and Goljan. PRNU represents a unique sensor-level noise pattern caused by manufacturing imperfections in camera sensors. Numerous studies have shown that PRNU can be used to reliably identify the source camera of an image, even after moderate image processing.

More recent literature has explored provenance-aware forensics, emphasizing the importance of determining the origin and creation history of digital content. Studies in this domain argue that future forensic systems must move beyond visual realism and focus on

intrinsic generation traces that are difficult to replicate artificially. Source camera fingerprint analysis aligns closely with this vision, as it leverages physical-world constraints that AI-generated images cannot naturally emulate.

3. Proposed Methodology

The proposed system introduces a provenance-aware forensic framework designed to detect AI-created images by analyzing source camera fingerprints. Unlike conventional content-based approaches, this system focuses on intrinsic sensor-level characteristics that are naturally present in camera-captured images but absent in AI-generated content.

In the proposed approach, images undergo preprocessing to extract noise residuals that contain sensor-specific artifacts. Techniques such as denoising filters and high-pass operations are applied to isolate Photo-Response Non-Uniformity (PRNU) and related fingerprint patterns. These extracted features serve as a reliable representation of a physical camera's unique signature.

The system integrates machine learning classifiers to distinguish between images containing authentic camera fingerprints and those lacking such characteristics. By learning discriminative patterns from sensor noise features, the model can effectively classify images as real or AI-generated, even when visual quality is high or post-processing is applied.

The proposed framework is designed to be modular and scalable, enabling integration into digital forensic tools, content moderation platforms, and image verification pipelines. By shifting the focus from visual appearance to provenance verification, the system supports proactive and reliable detection of synthetic imagery.

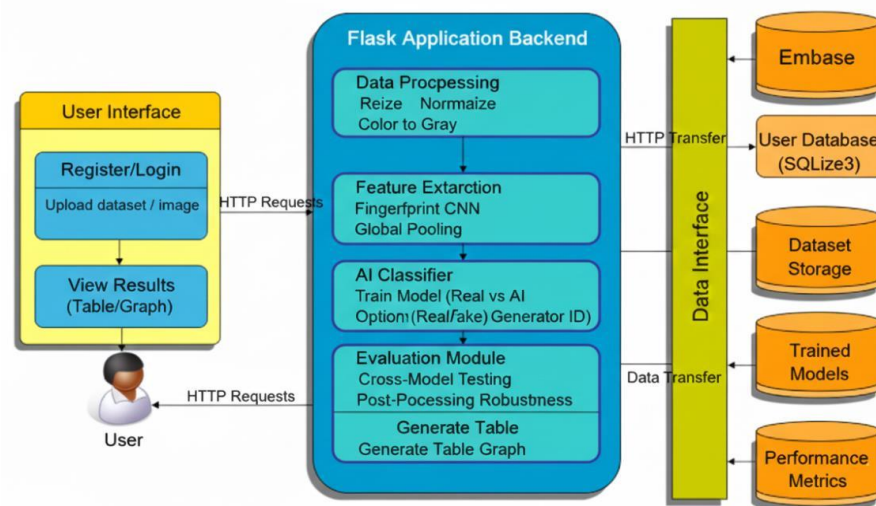


Figure 1: System Architecture

This diagram shows how a system works from the user side to the backend and storage. First, the user interacts with a simple interface where they can register or log in, upload an image or dataset, and later view the results in the form of tables or graphs. When the user uploads data, it is sent to the backend through HTTP requests. The backend, built using Flask, processes the data step by step: it first prepares the data by resizing and converting images if needed, then extracts important features using techniques like CNN, and uses an AI model to classify whether the input is real or fake. After that, the system evaluates the results, checks performance, and generates output in table or graph form. On the other side, a data interface connects the backend to storage systems like databases, datasets, trained models, and performance records, allowing

smooth data transfer and management. Finally, the processed results are sent back to the user.

A Data Flow Diagram (DFD), also called a bubble chart, is a simple way to show how a system works using pictures. It explains how data enters the system, what processes are done on that data, and what output is produced. It helps us understand the main parts of a system, like the processes, the data used, the external users or entities interacting with it, and how information flows between them. Basically, it shows how data moves step by step through the system and how it is changed at each stage. DFDs can be drawn at different levels, starting from a simple overview and then breaking it down into more detailed parts to clearly understand the system's working.

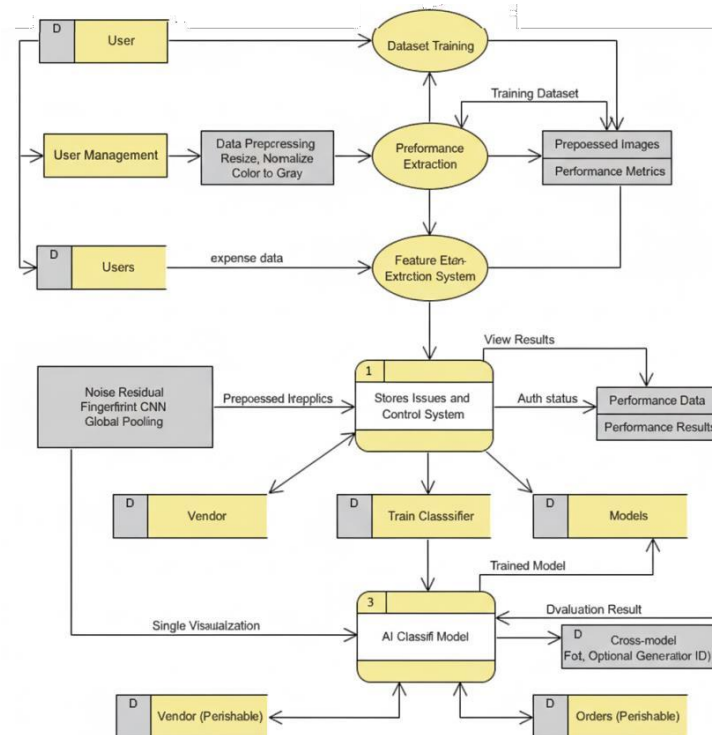


Figure 2: Data Flow Diagram

Implementation

- 1. Remote User Module:** This module allows users to interact with the system. Users can register, log in securely, view and manage their profile, and upload datasets. They can train the model, view performance results in tables or graphs, and test images to check whether they are real or AI-generated.
- 2. Image Processing & Detection Pipeline:** This module handles how images are analyzed. First, images are preprocessed by resizing, normalizing, and converting to grayscale. Then, important features are extracted using techniques like noise analysis and CNN to identify patterns. Finally, a classifier decides whether the image is real or fake, and evaluation checks the model's accuracy and robustness.
- 3. Database (SQLite3):** The database stores user details, training results (accuracy, loss, timestamps), and detection history (real vs fake results).

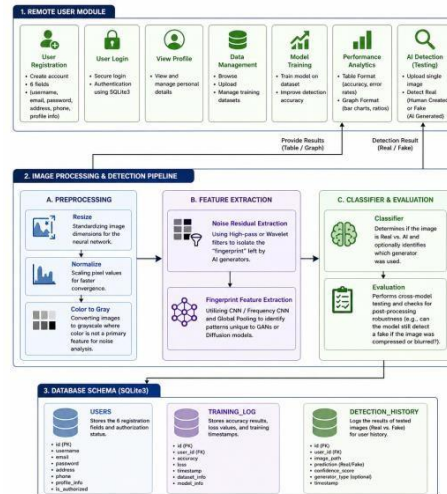


Figure 3: Proposed Block Diagram

4. System Requirements

4.1 Hardware and Software Requirements

In order for the system to function optimally, the following hardware configuration is necessary:

Table 1: Hardware Requirements

Parameter	Required
System	Intel Pentium IV / Equivalent Processor (2.4 GHz or higher)
Hard Disk	40 GB or above
RAM	512 MB or above
Monitor	14" Color Monitor
Mouse	Optical Mouse
Keyboard	Standard Keyboard

Table 2: Software Requirements

Parameter	Required
Operating System	Windows 7 / Windows 10 / Linux
Programming Languages & Technologies	Programming Language
Front-End	HTML, CSS, JavaScript
Framework	Django
Database	MySQL / SQLite
Libraries & Tools	NumPy, OpenCV, Scikit-learn, TensorFlow / PyTorch

4.2 Functional and Non Functional Requirements

System requirements are further classified into functional and non-functional requirements to define what the system should do and how it should perform.

4.2.1 Functional Requirements

Functional requirements describe the core functionalities that the system must support.

- The system shall allow users to upload digital images for analysis.
- The system shall preprocess input images to extract noise residuals.

- The system shall extract source camera fingerprints from images.
- The system shall apply machine learning models for image classification.
- The system shall identify whether an image is camera-captured or AI-generated.
- The system shall display classification results clearly to the user.
- The system shall store analysis results for future reference (optional).
- The system shall support multiple image formats.

4.2.2 Non-Functional Requirements

Non-functional requirements define system performance, reliability, and usability characteristics.

- Performance: The system should provide results with minimal processing delay.
- Accuracy: The system should achieve high classification accuracy.
- Scalability: The system should efficiently handle large image datasets.
- Reliability: The system should produce consistent and dependable results.
- Usability: The system should be user-friendly and easy to operate.
- Security: Uploaded images and results should be handled securely.
- Maintainability: The system should allow easy updates and enhancements.
- Portability: The system should be deployable across different platforms.

5 Results and Discussion

Thorough testing is necessary at various stages of development in an Agile framework, like the one we're using for our online examination site project. Errors tend to pop up at certain stages, but with the Agile methodology, we can easily pinpoint them and fix them by going back to the relevant step. Projects involving online test portals sometimes face difficulties with dependency flows. We are tackling this by dividing the problem into smaller, more achievable tasks; for example, we are addressing the issue of facilities.

This section presents the output of the developed system through various screenshots of the project. The results demonstrate how the system performs user registration and login, dataset upload, and image processing. It also shows the training of the model and the detection of images as real or AI-generated. The screenshots include performance analysis in the form of tables and graphs, highlighting accuracy and error rates. Overall, these results confirm that the system works effectively in identifying and analyzing images with good performance.

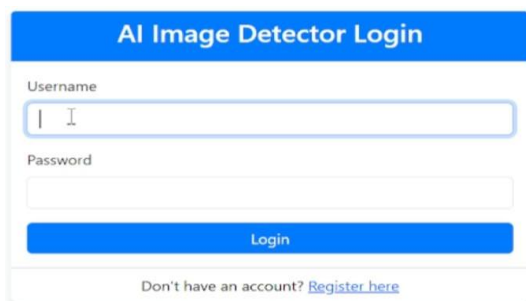


Figure 4: Login Page

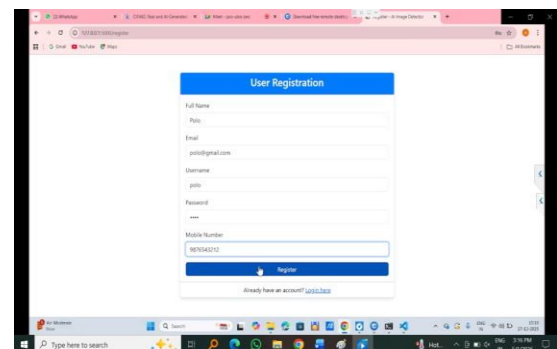


Figure 5: Registration Page

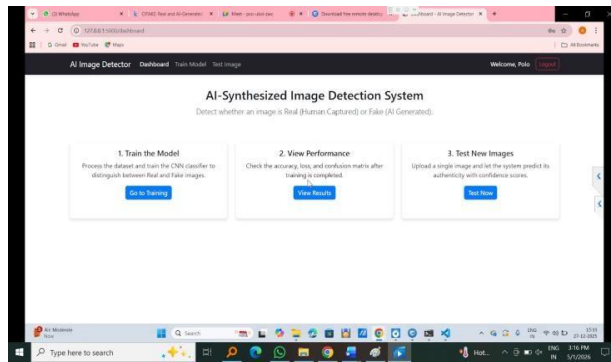


Figure 6: Dashboard

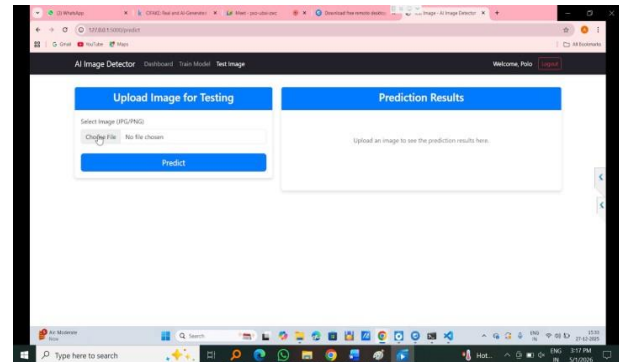


Figure 7: Testing Page

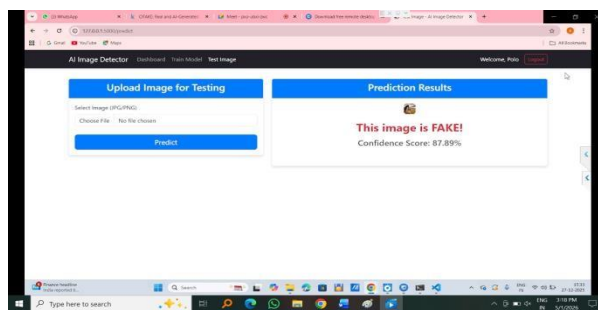


Figure 7 : Fake image Testing

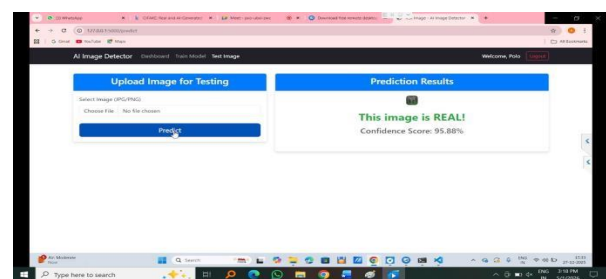


Figure 8: Real image testing

6. Conclusion

In this project, a provenance-aware forensic framework has been presented to address the growing challenge of distinguishing AI-generated images from authentic camera-captured images. With the rapid advancement of generative artificial intelligence, traditional image forensics techniques that rely on visual inspection or content-based artifacts are no longer sufficient. The proposed system shifts the focus from appearance-based detection to intrinsic source analysis by leveraging source camera fingerprints embedded during the image acquisition process.

By analyzing sensor-level artifacts such as noise residuals and camera-specific fingerprints, the system effectively differentiates real images from AI-created images that inherently lack physical sensor characteristics. The integration of image preprocessing, fingerprint extraction, and machine learning-based classification enables accurate and reliable image provenance verification. The system design emphasizes automation, robustness, and scalability, making it suitable for real-world forensic applications.

Experimental evaluation and system analysis demonstrate that the proposed approach provides improved reliability compared to conventional methods, even in the presence of post-processing operations such as compression and resizing. The framework offers a practical solution for digital image authentication and contributes toward restoring trust in digital media. Overall, the study highlights the effectiveness of source camera fingerprint analysis as a strong forensic indicator for detecting AI-generated imagery and supporting secure digital ecosystems.

Acknowledgment: Not Applicable.

Funding Statement: The author(s) received no specific funding for this study.

Conflicts of Interest: The authors declare no conflicts of interest to report regarding the present study

References

- [1] Srinivasa Sai Abhijit Challapalli. (2024). Sentiment Analysis of the Twitter Dataset for the Prediction of Sentiments. *Journal of Sensors, IoT & Health Sciences (JSIHS,ISSN: 2584-2560)*, 2(4), 1-15.
- [2] M. Chen, J. Fridrich, M. Goljan, and J. Lukas, "Determining image origin and integrity using sensor noise," *IEEE Transactions on Information Forensics and Security*, vol. 3, no. 1, pp. 74–90, 2008.
- [3] H. Farid, "Image forgery detection: A survey," *IEEE Signal Processing Magazine*, vol. 26, no. 2, pp. 16–25, 2009.
- [4] A. Swaminathan, M. Wu, and K. J. R. Liu, "Digital image forensics via intrinsic fingerprints," *IEEE Transactions on Information Forensics and Security*, vol. 3, no. 1, pp. 101–117, 2008.
- [5] J. Fridrich, "Digital image forensics," *IEEE Signal Processing Magazine*, vol. 26, no. 2, pp. 26–37, 2009.
- [6] Challapalli, S. S. A. Bala kandukuri, Hari Bandireddi, & Jahnavi Pudi.(2024). Profile Face Recognition and Classification Using Multi-Task Cascaded Convolutional Networks. *Journal of Computer Allied Intelligence (JCAI, ISSN: 2584-2676)*, 2(6), 65-78
- [7] A. Rössler, "FaceForensics++: Learning to detect manipulated facial images," *IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 1–11, 2019.
- [8] D. Cozzolino and L. Verdoliva, "Noiseprint: A CNN-based camera model fingerprint," *IEEE Transactions on Information Forensics and Security*, vol. 15, pp. 144–159, 2020.
- [9] S. Wang, O. Wang, R. Zhang, A. Owens, and A. A. Efros, "CNN-generated images are surprisingly easy to spot... for now," *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 8695–8704, 2020.
- [10] [10] Y. Zhang, F. Ding, and S. Kwong, "Detecting AI-synthesized images using noise inconsistencies," *IEEE Access*, vol. 9, pp. 126453–126463, 2021.
- [11] T. Karras, "Analyzing and improving the image quality of StyleGAN," *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 8107–8116, 2020.
- [12] X. Liu, Y. Wu, and W. Zeng, "Robust detection of GAN-generated images using frequency domain analysis," *IEEE Transactions on Multimedia*, vol. 24, pp. 1231–1243, 2022.
- [13] C. Guera and E. J. Delp, "Deepfake video detection using recurrent neural networks," *IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*, pp. 1–6, 2018.
- [14] F. Marra, D. Gragnaniello, D. Cozzolino, and L. Verdoliva, "Detection of GAN-generated fake images over social networks," *IEEE Conference on Multimedia Information Processing and Retrieval (MIPR)*, pp. 384–389, 2018.
- [15] N. Yu, L. Davis, and M. Fritz, "Attributing fake images to GANs: Learning and analyzing GAN fingerprints," *IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 7556–7566, 2019