

Signature Verification System: Real-Time Forgery Detection Using Deep Learning and Computer Vision

N. Madhuri¹; M. Manasa²; K. Uday Satya Sri Manikanta³; K. Srinivasulu⁴; K. Lakshmi Sainadh⁵

^{1,2,3,4,5}Aditya College of Engineering and Technology

Publication Date: 2026/10/10

Abstract: Recent years have witnessed a significant rise in the application of deep convolutional neural networks (CNNs) in computer vision and pattern recognition tasks. Offline handwritten signature verification, which remains a challenging problem, is one of the most widely used biometric techniques in banking systems, administrative processes, and financial applications. This project explores CNN-based approaches for signature verification and evaluates the effectiveness of deep learning models as feature extractors using a transfer learning strategy. The proposed system presents a robust deep learning-based framework for detecting forged signatures by leveraging the ResNet50 architecture, a pre-trained convolutional neural network widely used for image classification tasks. ResNet50 incorporates residual connections, which enable efficient information flow across layers and prevent gradient vanishing, thereby improving the performance of deeper networks. In this approach, signature images are first preprocessed using OpenCV techniques such as resizing and normalization, and then passed through a fine-tuned ResNet50 model for feature extraction and classification into genuine or forged categories. To enhance usability, a web-based interface is developed using Django, allowing users to upload signature images and obtain real-time verification results. The system is designed to operate on standard computational resources, making it cost-effective, scalable, and suitable for real-world deployment in banking, legal documentation, and identity verification systems. Experimental evaluation on a dataset containing both genuine and forged signatures demonstrates that the proposed model achieves an average classification accuracy of approximately 95%, indicating strong robustness against variations in writing styles and image conditions. Furthermore, the system addresses key limitations of traditional methods, such as reliance on handcrafted features, poor generalization, and lack of real-time verification capability.

Keywords: Signature Verification, Forgery Detection, Deep Learning, CNN, ResNet50, Transfer Learning, Image Classification, OpenCV, Django.

How to Cite: N. Madhuri; M. Manasa; K. Uday Satya Sri Manikanta; K. Srinivasulu; K. Lakshmi Sainadh (2026) Signature Verification System: Real-Time Forgery Detection Using Deep Learning and Computer Vision. *International Journal of Innovative Science and Research Technology*, 11(3), 4281-4285. <https://doi.org/10.38124/ijisrt/26mar2120>

I. INTRODUCTION

Signature verification remains a critical component of identity authentication in financial transactions, legal documentation, and administrative systems. Despite the rapid advancement of digital authentication technologies, handwritten signatures continue to be widely accepted due to their simplicity, convenience, and legal validity. However, verifying signatures—especially in offline scenarios—poses significant challenges due to natural variations in writing styles, presence of skilled forgeries, and subjective human judgment. Manual verification is often time-consuming, inconsistent, and prone to errors when distinguishing between genuine and forged signatures.

Traditional automated signature verification systems primarily rely on handcrafted feature extraction and classical machine learning techniques. Although these methods offer moderate performance, they often fail to generalize across diverse datasets and are sensitive to variations in scale, stroke patterns, and image quality. Additionally, systems based on single-model architectures may lack robustness, leading to increased misclassification, particularly in complex forgery cases.

Recent advancements in artificial intelligence, particularly deep learning and Convolutional Neural Networks (CNNs), have significantly improved performance in image-based classification tasks. CNNs enable automatic feature extraction and hierarchical representation learning, eliminating the need for manual feature engineering while

capturing both local and global patterns in images. However, single deep learning models may still struggle to capture the diverse feature distributions present in real-world signature datasets.

To address these challenges, this work proposes a multi-model deep learning-based signature verification system that combines custom CNN architectures with transfer learning techniques. The system leverages multiple pre-trained models, including ResNet50, VGG16, and VGG19, which are trained on large-scale datasets such as ImageNet and fine-tuned for the signature verification task. Among these, ResNet50 plays a crucial role due to its deep residual learning framework, which incorporates identity shortcut connections to enable efficient information flow and mitigate the vanishing gradient problem. This allows the model to achieve higher accuracy and better generalization compared to traditional deep networks.

Transfer learning is employed to adapt pre-trained models to the signature verification domain, significantly reducing training time and improving performance, especially when labeled data is limited. Instead of training models from scratch, pre-trained weights are fine-tuned to capture domain-specific features of handwritten signatures, enhancing classification capability.

The proposed system follows an ensemble-based approach, where predictions from multiple models are combined to produce a final decision. This multi-model strategy improves robustness and accuracy by leveraging the strengths of different architectures. The system processes offline signature images through several stages, including preprocessing, feature extraction, and classification. Key components of the system include image preprocessing (resizing, normalization, and noise reduction), deep feature extraction using CNN and transfer learning models, independent predictions from multiple architectures, and an ensemble decision mechanism for final classification. Furthermore, the system is implemented using widely adopted deep learning frameworks and deployed through a web-based interface, enabling real-time signature verification. It operates efficiently on standard computational resources, making it scalable and suitable for real-world applications without requiring specialized hardware.

➤ *The Main Contributions of this Work are Summarized as Follows:*

- A unified multi-model framework combining custom CNN and transfer learning architectures for signature verification;
- An ensemble-based decision mechanism that enhances classification reliability and robustness;
- A scalable, cost-effective, and real-time implementation using a web-based deployment model.

The remainder of this paper is organized as follows: Section II reviews related work. Section III describes the proposed methodology. Section IV presents the system architecture. Section V discusses experimental results and

performance evaluation. Section VI concludes the paper and outlines future work.

II. RELATED WORK

Signature verification has been extensively studied in the domains of pattern recognition and biometric authentication. Existing approaches can broadly be categorized into traditional feature-based methods, deep learning-based methods, and hybrid ensemble approaches.

➤ *Feature-Based Methods*

Early research in signature verification relied on handcrafted feature extraction techniques combined with classical machine learning algorithms. Features such as stroke direction, curvature, pixel density, and geometric attributes were extracted and used with classifiers like Support Vector Machines (SVM) and K-Nearest Neighbors (KNN). While these approaches are computationally efficient, their performance is highly dependent on feature engineering and lacks robustness against variations in writing style and skilled forgeries.

➤ *Deep Learning-Based Methods*

Recent advancements in deep learning have significantly improved the performance of signature verification systems. Convolutional Neural Networks (CNNs) enable automatic feature extraction and hierarchical representation learning directly from raw image data. Architectures such as VGGNet and ResNet have been widely adopted due to their ability to capture complex spatial patterns in signature images.

Transfer learning techniques further enhance performance by leveraging pre-trained models on large-scale datasets such as To overcome the limitations of single-model approaches, ensemble learning techniques have been introduced. These methods combine predictions from multiple models to improve classification accuracy and robustness. Hybrid systems integrating CNN-based feature extraction with ensemble decision mechanisms have shown promising results in reducing misclassification rates, particularly in cases involving skilled forgeries.

Despite these advancements, existing systems exhibit several limitations: (1) dependency on a single model, leading to reduced generalization capability; (2) sensitivity to image variations such as noise, scale, and orientation; and (3) lack of robust decision mechanisms for handling ambiguous cases. The proposed system addresses these challenges by integrating multiple deep learning models within a unified ensemble framework, enabling improved accuracy and reliability in signature verification tasks.

III. MATHEMATICAL FORMULATION

➤ *Image Preprocessing and Normalization*

Let the input signature image be represented as $I(x, y)$, where x and y denote spatial coordinates. The preprocessing stage transforms the image into a normalized format:

$$I' = \frac{I - \mu}{\sigma}$$

Where μ and σ represent the mean and standard deviation of pixel intensities. This normalization ensures consistent input distribution across models.

➤ CNN-Based Feature Extraction

Each deep learning model learns a mapping function:

$$f(I') = y$$

Where f represents the neural network and $y \in \{0,1\}$ denotes the classification output (genuine or forged). The convolutional layers extract spatial features using learned filters:

$$Fi = I' * Ki$$

Where Ki represents the convolution kernel and $*$ denotes convolution operation.

➤ Transfer Learning Representation

For transfer learning models such as ResNet50, VGG16, and VGG19, pre-trained weights are used to initialize the network. The feature extraction process can be represented as:

$$z = g(I'; \theta)$$

Where g denotes the pre-trained network and θ represents learned parameters. The extracted features z are passed through fully connected layers for classification.

➤ Ensemble Decision Mechanism

The final classification is determined by aggregating predictions from multiple models. Let p_i denote the prediction probability from the i^{th} model. The ensemble output is computed as:

$$P_{final} = \frac{1}{N} \sum_{i=1}^N p_i$$

➤ Ensemble and Hybrid Methods

ImageNet. These models are fine-tuned for signature classification tasks, reducing training time and improving generalization. However, most deep learning approaches rely on a single architecture, which may not effectively capture diverse variations present in real-world datasets.

Where N is the total number of models. The decision rule is defined as:

$$\text{Genuine} \Leftrightarrow P_{final} \geq \tau$$

Where τ is a predefined threshold (typically 0.5).

IV. SYSTEM ARCHITECTURE

➤ Pipeline Overview

The proposed signature verification system operates as a multi-stage pipeline:

- Image Acquisition — user uploads signature image through web interface.
- Preprocessing — resizing, normalization, and noise removal.
- Feature Extraction — deep features extracted using CNN and transfer learning models.
- Model Prediction — independent predictions from multiple architectures.
- Ensemble Decision — aggregation of outputs for final classification.

➤ Model Integration

The system integrates four models:

- Custom CNN model for domain-specific feature learning
- ResNet50 for deep residual feature extraction
- VGG16 and VGG19 for hierarchical representation learning

Each model processes the input independently and produces a confidence score.

➤ Web-Based Deployment

The system is deployed using a Django-based web framework, enabling real-time interaction. Users upload signature images, and the backend processes them through trained models to generate predictions. The results are displayed along with confidence scores.

➤ Decision Mechanism

A centralized decision module aggregates outputs from all models. The system ensures robustness by reducing dependency on a single prediction and improving classification stability.

➤ Hardware and Software Requirements

- Processor: Intel i3 or above
- RAM: Minimum 4 GB (8 GB recommended)
- Storage: 20 GB free space

➤ Software Requirements

- Operating System: Windows / Linux
- Language: Python 3.x
- Libraries: TensorFlow, Keras, OpenCV, NumPy
- Framework: Django

➤ Detection Performance

The performance of the proposed ResNet50-based signature verification system was evaluated using standard classification metrics, including Accuracy, Precision, Recall, and F1-Score. A true positive is defined as correctly identifying a signature as genuine or forged.

Table 1 Performance Metrics of Resnet50 Model

Metrics	Values
Accuracy	95.0%
Precision	94.6%
Recall	95.3%
F1-Score	94.9%

The model achieves high accuracy due to the strong feature extraction capability of ResNet50. The balanced Precision and Recall values indicate that the system effectively minimizes both false positives and false negatives, making it reliable for practical applications.

➤ Processing Latency

The system performance was evaluated on standard hardware configurations. On a system with Intel Core i5 processor and 8 GB RAM, the average inference time per image was approximately 30–40 ms. On a lower-end system (Intel Core i3, 4 GB RAM), the inference time increased to

around 60 ms per image. These results demonstrate that the system is capable of near real-time verification and is suitable for deployment in practical environments.

➤ Comparison with Existing Systems

The proposed system was compared with traditional signature verification approaches based on handcrafted features and classical machine learning algorithms. Unlike earlier systems, which rely on manual feature extraction, the proposed approach leverages deep learning for automatic feature representation.

Table 2 Comparison with Existing Methods

System Type	Feature Extraction	Accuracy	Real-Time
SVM-based Methods	Manual	Moderate	Limited
KNN-based Methods	Manual	Moderate	Limited
CNN (Basic)	Automatic	High	Yes
ResNet50 (Proposed)	Automatic	95%	Yes

V. EXPERIMENTAL RESULTS

➤ Dataset

The proposed system was evaluated on a dataset consisting of genuine and forged signature images collected from multiple individuals. The dataset includes variations in writing styles, stroke thickness, and image quality to simulate real-world conditions. To improve robustness, additional preprocessing was applied to normalize image size and intensity variations. The dataset was divided into training and testing sets to ensure unbiased evaluation, with annotations verified for correctness.

The results show that the proposed ResNet50-based system outperforms traditional methods in terms of accuracy and robustness, while also supporting real-time verification.

➤ Failure Cases and Limitations

Despite achieving high accuracy, the system exhibits certain limitations. Performance may degrade when signature images are of extremely low quality or contain significant noise. Variations such as incomplete strokes or highly skilled forgeries can also affect classification accuracy. Additionally, the model performance depends on the diversity of the training dataset. Future improvements include enhancing dataset diversity and incorporating advanced architectures to further improve robustness.

VI. CONCLUSION

In this paper, a deep learning-based signature verification system was presented for accurate detection of forged signatures in offline scenarios. The proposed approach

utilizes a ResNet50-based transfer learning model to perform robust feature extraction and classification. The system processes input signature images through preprocessing steps and generates predictions indicating whether a signature is genuine or forged.

The developed framework is implemented using TensorFlow, Keras, and OpenCV, and deployed through a web-based interface, enabling real-time verification on standard computational hardware. Experimental evaluation demonstrates that the proposed system achieves an accuracy of 95%, indicating strong performance under variations in writing styles and image conditions. The system effectively addresses key limitations of traditional methods, including reliance on handcrafted features and limited generalization capability.

Future work will focus on improving robustness by incorporating larger and more diverse datasets, enhancing performance under noisy and low-quality image conditions, and exploring advanced architectures such as Vision Transformers. Additionally, deployment on lightweight edge devices and integration with mobile applications will be considered to enable broader real-world adoption.

ACKNOWLEDGMENT

The author sincerely expresses gratitude to the project guide for their valuable guidance, continuous support, and insightful feedback throughout the development of this work. Appreciation is also extended to the Department of Artificial Intelligence and Machine Learning for providing the necessary resources and infrastructure to successfully complete this project.

REFERENCES

- [1.] K. He, X. Zhang, S. Ren, and J. Sun, “Deep Residual Learning for Image Recognition,” in Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR), 2016, pp. 770–778.
- [2.] K. Simonyan and A. Zisserman, “Very Deep Convolutional Networks for Large-Scale Image Recognition,” arXiv preprint arXiv:1409.1556, 2014.
- [3.] Y. LeCun, Y. Bengio, and G. Hinton, “Deep Learning,” *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.
- [4.] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “ImageNet Classification with Deep Convolutional Neural Networks,” in Proc. Advances in Neural Information Processing Systems (NIPS), 2012, pp. 1097–1105.
- [5.] S. J. Pan and Q. Yang, “A Survey on Transfer Learning,” *IEEE Transactions on Knowledge and Data Engineering*, vol. 22, no. 10, pp. 1345–1359, 2010.
- [6.] M. Diaz, M. A. Ferrer, D. Impedovo, and G. Pirlo, “A Perspective Analysis of Handwritten Signature Technology,” *ACM Computing Surveys*, vol. 51, no. 6, pp. 1–39, 2019.
- [7.] R. Plamondon and G. Lorette, “Automatic Signature Verification and Writer Identification — The State of the Art,” *Pattern Recognition*, vol. 22, no. 2, pp. 107–131, 1989.
- [8.] A. Hafemann, R. Sabourin, and L. Oliveira, “Learning Features for Offline Handwritten Signature Verification using Deep Convolutional Neural Networks,” in Proc. International Joint Conference on Neural Networks (IJCNN), 2017, pp. 2576–2583.
- [9.] G. Bradski, “The OpenCV Library,” *Dr. Dobb’s Journal of Software Tools*, 2000.
- [10.] TensorFlow, “An End-to-End Open Source Machine Learning Platform,” Google, 2015.
- [11.] F. Chollet, “Keras,” <https://keras.io>, 2015.