As per UGC guidelines an electronic bar code is provided to secure your paper

International Journal for Modern Trends in Science and Technology
Volume 12, Issue 04, pages 918-924
ISSN: 2455-3778 online
Available online at: http://www.ijmtst.com/vol12issue04.html
DOI: https://doi.org/10.5281/zenodo.19644340

Design and Implementation of a Deep Learning-Based Fake Currency Detection System with Scalable Deployment Architecture

Dr. B. V. S. Varma, G. S. S. Vara Prasad, A. Suresh, A. Bhargavi, B. A. V. Karthik

Department of Computer Science and Engineering, D.N.R. College of Engineering & Technology, Balusumudi, Bhimavaram, Andhra Pradesh, India

To Cite this Article
Dr. B. V. S. Varma, G. S. S. Vara Prasad, A. Suresh, A. Bhargavi & B. A. V. Karthik (2026). Design and Implementation of a Deep Learning-Based Fake Currency Detection System with Scalable Deployment Architecture. International Journal for Modern Trends in Science and Technology, 12(04), 918-924. https://doi.org/10.5281/zenodo.19644340

Article Info
Received: 17 March 2026; Revised: 07 April 2026; Accepted: 10 April 2026.

KEYWORDS

Fast API, RESTful API, Convolutional Neural Network (CNN), Currency Authentication, Deep Learning, TensorFlow/Keras, Image Preprocessing (OpenCV), Asynchronous Processing (Uvicorn).

ABSTRACT

This project implements a RESTful API using Fast API for authenticating currency notes through deep learning. The core functionality revolves around a Currency Classifier model a Convolutional Neural Network (CNN) trained to distinguish between genuine and counterfeit currency alongside a Data Preprocessor for preparing image inputs before inference. The primary endpoint /api/predict accepts image files (JPG, JPEG, PNG, up to 16 MB), validates the file type and size, preprocesses the image using OpenCV, and feeds it to the loaded TensorFlow/Keras model to predict whether the currency note is genuine or counterfeit, returning a confidence score and detailed prediction metadata in a structured JavaScript Object Notation (JSON) response. The application is designed for scalability and robustness, leveraging Uvicorn for asynchronous processing and Cross Origin Resource Sharing (CORS) Middleware for cross-origin resource sharing, which is crucial for web-based deployments. A /api/health endpoint monitors model loading status, preprocessor readiness, and system memory utilization in real time.

1. INTRODUCTION

1.1 BRIEF INFORMATION

Currency plays a fundamental role in daily financial transactions, and ensuring its authenticity is essential for maintaining economic stability and preventing fraud. With the rapid advancement in printing and reproduction technologies, counterfeit currency has become increasingly sophisticated, making it difficult to detect using traditional manual methods. Conventional approaches for currency verification, such as visual inspection and hardware-based detection systems, are often limited by human error, high cost, and lack of accessibility. These methods are not suitable for large-scale or real-time applications, especially for small businesses and individual users. To overcome these limitations, this project introduces a Deep Learning-Based Fake Currency Detection System that leverages Convolutional Neural Networks (CNN) and computer vision techniques. The system is designed as a web-based application using Fast API, enabling users to upload images of currency notes and receive instant predictions regarding their authenticity. The application processes input images through a preprocessing pipeline and feeds them into a trained CNN model, which classifies the currency as genuine or counterfeit. The result is presented along with a confidence score and additional metadata. The system also includes scalable deployment features, making it suitable for integration into real-world financial environments. This approach enhances accuracy, reduces dependency on manual verification, and provides a fast, reliable, and accessible solution for counterfeit currency detection.

1.2 PURPOSE

The purpose of this project is to develop an automated system for detecting counterfeit currency using deep learning techniques. It allows users to upload currency images and receive real-time authenticity predictions. The system uses a CNN model to improve accuracy in classification. It reduces dependency on manual verification and costly hardware devices. The solution is scalable and suitable for real-world financial applications.

1.3 MOTIVATION

The increasing circulation of counterfeit currency poses a serious threat to economic stability and financial security. Traditional verification methods are time-consuming, costly, and prone to human error. There is a growing need for an automated, accurate, and easily accessible solution. Advancements in deep learning and computer vision provide an opportunity to solve this problem efficiently. This project is motivated by the need to develop a fast, reliable, and scalable system for real-time currency authentication.

1.4 PROBLEM STATEMENT

Counterfeit currency has become increasingly sophisticated, making it difficult to detect using traditional manual and hardware-based methods. Existing solutions are either time-consuming, expensive, or not easily accessible to common users. Manual verification is prone to human error, while hardware devices lack portability and scalability. There is a need for an automated system that can accurately and efficiently detect fake currency in real time. This project aims to develop a deep learning-based solution that provides fast, reliable, and user-friendly currency authentication

2. LITERATURE REVIEW

Various studies propose deep learning and image processing-based solutions to address the problem of counterfeit currency detection, emphasizing the need for automated and accurate verification systems. Traditional approaches use machine learning algorithms with handcrafted features, but they often lack robustness and adaptability. Recent research highlights the effectiveness of Convolutional Neural Networks (CNNs) in extracting complex visual patterns from currency images. Systems based on deep learning provide higher accuracy, faster processing, and better generalization across different note conditions. These approaches significantly reduce human effort and improve reliability in detecting fake currency compared to manual and hardware-based methods.

2.1. Deep Learning for Image Classification

AUTHORS: Yann LeCun, Yoshua Bengio, Geoffrey Hinton

This work highlights the importance of deep learning in image recognition tasks. Convolutional Neural Networks (CNNs) are capable of automatically learning hierarchical features from images, making them highly effective for classification problems. The study demonstrates how deep neural networks outperform traditional machine learning methods in terms of accuracy and feature extraction, especially in complex visual tasks.

2.2. Machine Learning in Pattern Recognition

AUTHORS: Tom Mitchell

This study defines machine learning as a system's ability to learn from data and improve performance over time. It explains various algorithms such as decision trees and Bayesian models used in pattern recognition. However, these methods require manual feature extraction and are less efficient compared to deep learning models for image-based applications.

2.3. Image Processing Techniques for Detection Systems

AUTHORS: Gonzalez and Woods

This research focuses on traditional image processing techniques such as edge detection, filtering, and histogram analysis. These methods are useful for extracting basic features from images but lack the capability to handle complex variations in realworld data, making them less reliable for counterfeit currency detection.

2.4. Deep Learning in Computer Version Application
AUTHORS: Greenspan et al.

This study emphasizes the use of deep learning in computer vision tasks. It shows how CNN models improve detection accuracy by automatically learning features instead of relying on handcrafted inputs. The research demonstrates significant improvements in classification performance across various domains.

2.5 Automated Model Optimization Techniques

AUTHORS: Bergstra and Bengio

This work discusses optimization techniques such as random search and hyperparameter tuning to improve model performance. These methods help in selecting the best parameters for deep learning models, resulting in better accuracy and efficiency in prediction systems.

3. PROPOSED SYSTEM

The proposed system, titled "Currency Note Authentication using Deep Learning", introduces a novel, AI-powered approach to identifying counterfeit currency notes. It bridges the limitations of traditional methods by utilizing convolutional neural networks (CNNs), web technologies, and real-time image processing to detect fake currency notes with high accuracy and usability.

3.1. System Architecture Overview

The system is designed with a modular and layered architecture consisting of the following components:

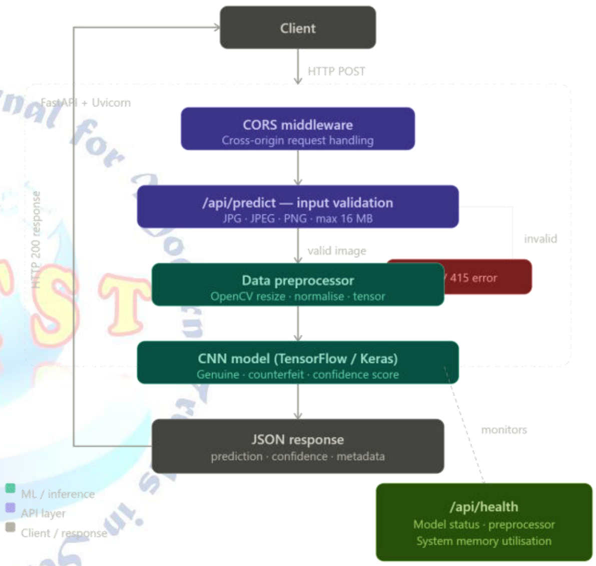


Fig 1: System Architecture

1. **User Interface (UI):**
   - Developed using HTML, CSS, and JavaScript.
   - Allows users to upload or drag-and-drop currency images.
   - Displays prediction results with confidence score and visual indicators.
2. **Backend Server:**
   - Built using **FastAPI**, a high-performance Python web framework.
   - Hosts the image preprocessing, model inference, and result formatting logic.
3. **Deep Learning Model:**

- TensorFlow-based Convolutional Neural Network (CNN).
- Trained on labeled datasets of genuine and counterfeit notes.
- Provides binary classification with probability output.

4. **Image Preprocessing Engine:**
   - Uses OpenCV to resize, normalize, and convert image formats.
   - Ensures consistent input for the deep learning model.
5. **Result Handling and Visualization:**
   - Frontend interprets and displays JSON response.
   - Renders prediction labels, confidence bars, and error messages.

3.2. Functional Workflow

1. User uploads or drags a currency note image to the interface.
2. The image is sent to the FastAPI backend via a POST request.
3. The backend validates the file and temporarily saves it.
4. The image is pre-processed (resized, normalized, color corrected).
5. The preprocessed image is fed to the trained CNN model.
6. The model returns a prediction (genuine or counterfeit) with confidence score.
7. The result is returned to the frontend and displayed to the user.

3.3 UML Diagrams

3.3.1 **Use Case Diagram**

- Actors: User, System (API)

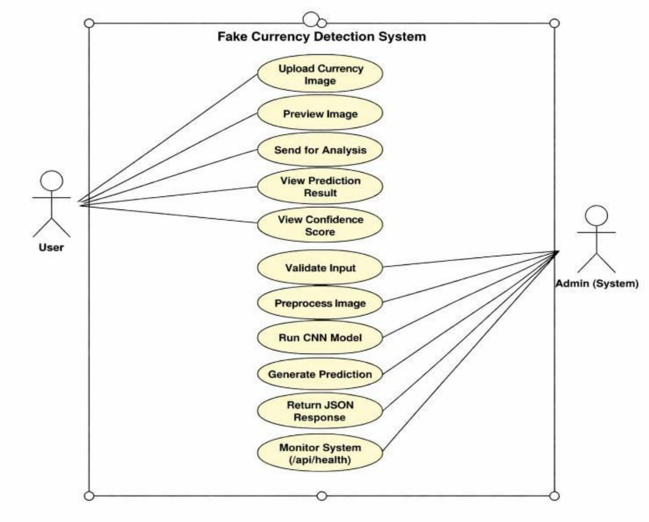


Fig 2: **Use Case Diagram**

3.3.4 **Class Diagram (Simplified)**

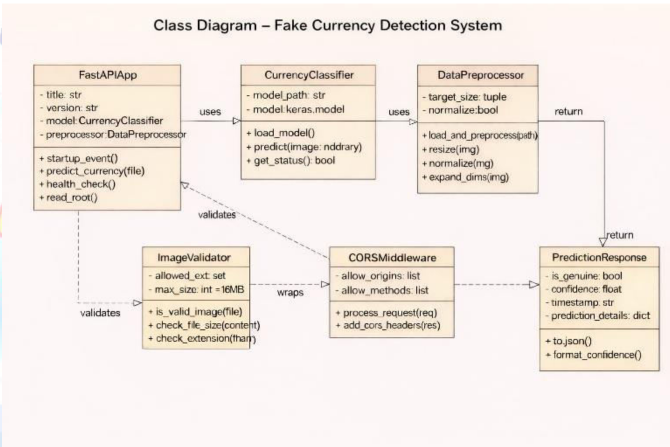


Fig 3: **Class Diagram**

Currency Guard AI – Deep Learning-Based Currency Authentication System

Auto-Optimization Pipeline for DL-Based Currency Authentication

Figure 1 illustrates the back-end pipeline of the Currency Guard AI system that performs automated deep learning model optimization. The architecture receives an input image classification task and an initial deep learning (DL) model architecture (e.g., CNN), and applies an Auto-Optimization Scheme to derive:

- The optimal model architecture (e.g., number of layers, filters, activation functions),
- The best hyperparameter configuration (e.g., learning rate, batch size, optimizer),

- And ultimately, the maximum performance in terms of validation accuracy, loss, and generalization ability.

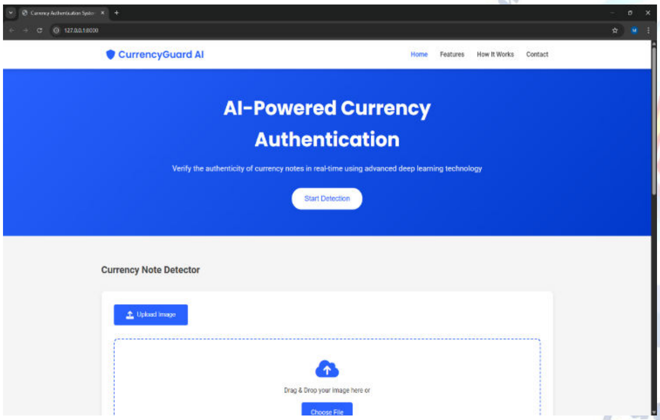
This optimization block can be implemented using:

- Bayesian optimization, random/grid search, or evolutionary algorithms,
- Automated Machine Learning (AutoML) frameworks such as Optuna, Keras Tuner, or AutoKeras,
- With an objective function defined over cross-validated performance metrics.

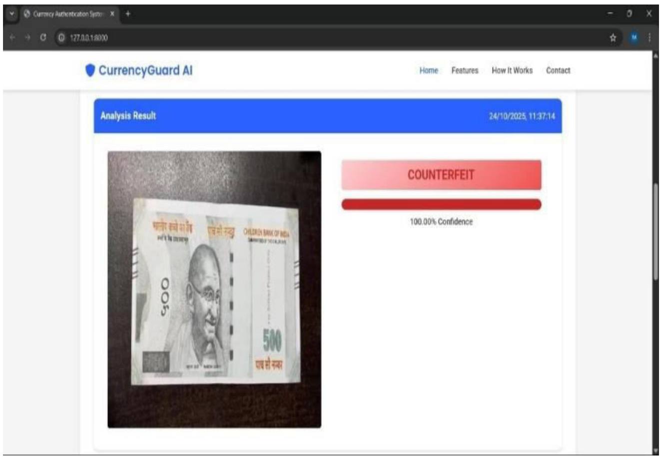
This approach enables a meta-learning loop where the model continuously adapts or self-tunes for the given dataset, thereby making the system extensible to different currency types or visual contexts without manual reconfiguration.

4. RESULTS

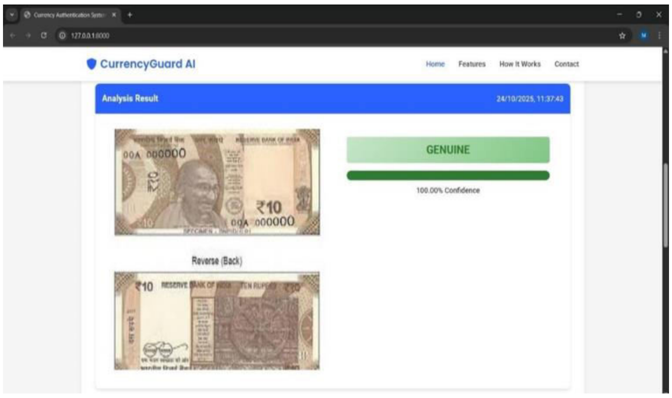
4.1 HOME PAGE



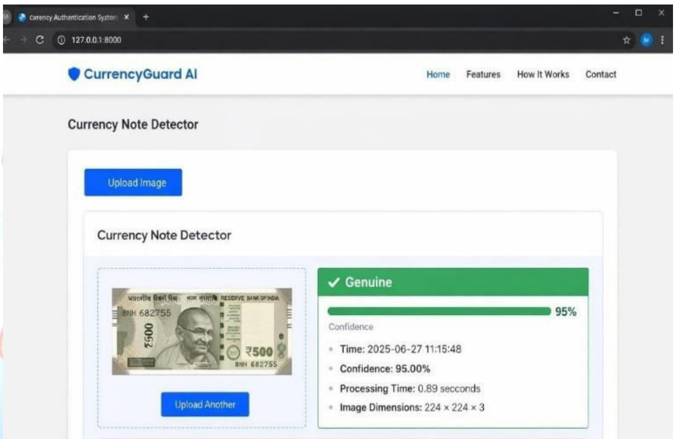
4.2 PREDICTION RESULT – counterfeit



4.3 PREDICTION RESULT – Genuine



4.4 API Response (JSON Output)



CONCLUSION:

The project "Deep Learning-Based Fake Currency Detection System with Scalable Deployment Architecture" successfully demonstrates the application of modern artificial intelligence techniques in solving a critical real-world problem identifying counterfeit currency. The system integrates a Convolutional Neural Network (CNN) model with a Fast API-based RESTful backend, enabling accurate and real-time classification of currency notes as genuine or counterfeit. Through efficient image preprocessing, optimized model inference, and structured response handling, the application achieves reliable performance within a short processing time, fulfilling the requirement of near real-time verification. A major strength of the project lies in its scalable and modular architecture, which separates concerns into well-defined components such as image validation, preprocessing, model inference, and result visualization. This design not only improves maintainability but also allows easy integration with web interfaces and future deployment platforms like cloud or containerized environments. Additionally, the implementation of health monitoring APIs and robust error handling mechanisms ensures system stability and usability in practical scenarios. The system also provides a user-friendly frontend interface with visual feedback, making it accessible to non-technical users. By eliminating the need for specialized hardware and enabling verification through simple image uploads, the project increases accessibility and practical applicability in areas such as banks, retail sectors, and public use.

FUTURE SCOPE:

1. Mobile Application Development: A dedicated mobile application can be developed for Android and iOS platforms to make the system more accessible. Users will be able to capture images of currency notes directly using their smartphone cameras. This eliminates the need for external devices and improves usability in real-world scenarios. It is especially useful for shopkeepers and the general public.

2. Real-Time Detection: The system can be enhanced to support real-time detection using live video streams from cameras. Instead of uploading static images, the model can process continuous frames and provide instant results. This feature is highly beneficial in banks, ATMs, and retail environments. It improves speed and operational efficiency.

3. Multi-Currency Support: Currently, the system is limited to a specific currency dataset, but it can be expanded to support multiple currencies. By training the model with diverse datasets, it can recognize notes from different countries. This makes the system more versatile and globally applicable. It is useful for international financial environments.

4. Improved Deep Learning Models: Future improvements can include using advanced deep learning models like transfer learning techniques. Models such as ResNet or EfficientNet can provide higher accuracy and better feature extraction. These models perform well even under challenging conditions like poor lighting. This leads to more reliable detection results.

5. Cloud Deployment & Scalability: Deploying the system on cloud platforms can enhance scalability and availability. It allows the application to handle multiple users and large volumes of requests simultaneously. Cloud infrastructure also supports easy updates and maintenance. This is essential for enterprise-level implementation.

6. Advanced Feature Detection: The system can be improved by detecting physical security features of currency notes. These include watermarks, security threads, and microprinting patterns. Combining these features with deep learning increases detection accuracy. It also makes the system more robust against sophisticated counterfeit methods.

Integration with Banking Systems: The system can be integrated with banking and financial systems such as ATMs and POS machines. This allows automatic verification of currency during transactions. It reduces manual effort and enhances security. Such integration makes the solution more practical and widely usable.

Conflict of interest statement

Authors declare that they do not have any conflict of interest.

REFERENCES

[1] Hamet, P.; Tremblay, J. Artificial intelligence in medicine. Metabolism 2017, 69, S36–S40. [CrossRef] [PubMed]

[2] Nichols, J.A.; Herbert Chan, H.W.; Baker, M.A. Machine learning: Applications of artificial intelligence to imaging and diagnosis. Biophys. Rev. 2019, 11, 111–118. [CrossRef] [PubMed]

[3] Nilsson, N.J. Principles of Artificial Intelligence. IEEE Trans. Pattern Anal. Mach. Intell. 1981, PAMI-3, 112. [CrossRef]

[4] Zhang, X.-D. Machine Learning; Springer: Singapore, 2020; pp. 223–440. [CrossRef]

[5] Alpaydin, E. Introduction to Machine Learning; MIT Press: Cambridge, MA, USA, 2020.

[6] El Naqa, I.; Murphy, M.J. What is Machine Learning? In Machine Learning in Radiation

[7] Oncology; Springer: Cham, Switzerland, 2015; pp. 3–11. [CrossRef]

[8] Mitchell, T. Machine Learning; McGraw-Hill Science: London, UK, 1997.

[9] Mohammed, M.; Khan, M.B.; Bashier, E.B.M. Machine Learning: Algorithms and Applications; CRC Press: Cambridge, MA, USA, 2017.

[10] Greenspan, H.; Van Ginneken, B.; Summers, R.M. Guest Editorial Deep Learning in Medical Imaging: Overview and Future Promise of an Exciting New Technique. IEEE Trans. Med. Imaging 2016, 35, 1153–1159. [CrossRef]

[11] LeCun, Y.; Bengio, Y.; Hinton, G. Deep learning. Nature 2015, 521, 436–444. [CrossRef]

[12] He, X.; Zhao, K.; Chu, X. AutoML: A survey of the state-of-the-art. Knowl.-Based Syst. 2021, 212, 106622. [CrossRef]

[13] Bergstra, J.; Bengio, Y. Random search for hyper-parameter optimization. J. Mach. Learn. Res. 2012, 13, 281–305.

[14] Hesterman, J.Y.; Caucci, L.; Kupinski, M.A.; Barrett, H.H.; Furenlid, L.R. Maximumlikelihood estimation with a contracting-grid search algorithm. IEEE Trans. Nucl. Sci. 2010, 57, 1077–1084. [CrossRef]

[15] Dewancker, I.; McCourt, M.; Clark, C.; Peter, P.E. Hart, Bayesian Optimization Primer; Wiley: Hoboken, NJ, USA; p. 654. Available online: https://static.sigopt.com/b/20a144d208ef255d3b981ce419667ec25d8412e2/static/pdf/SigO pt_Bayesian_Optimization_ Primer.pdf (accessed on 1 June 2022).

[16] Shahriari, B.; Swersky, K.; Wang, Z.; Adams, R.P.; de Freitas, N. Taking the human out of the loop: A review of bayesian optimization. Proc. IEEE 2015, 104, 148–175. [CrossRef]