International Journal of Innovative Research in Science and Engineering
Volume No. 12, Issue No. 05, Year 2026
ISSN : 2454-9665

FAKE CURRENCY DETECTION USING DEEP CONVOLUTIONAL NEURAL NETWORKS

I.Suha Alekhya#1, *Assistant Professor, Department of Computer Science and Engineering, Tirumala Engineering College, Narasaraopet, Andra Pradesh, India - 522601.*
Dammavalam Abhinaya*2, A.V.Sai Rohitha*3, Bathula Harish Kumar *4, Gollapalli Charan*5

Abstract**—**Counterfeit currency poses a significant threat to global economies, leading to financial fraud and instability. Traditional detection methods, such as UV scanning and watermark verification, are costly, time-consuming, and prone to human error. To address these challenges, this project presents an AI-powered **Fake Currency Detection System** using **Convolutional Neural Networks (CNNs)** for accurate and efficient counterfeit detection. A Flask-based web application is developed to provide a user-friendly interface for currency verification. The system supports both image upload-based prediction and live webcam-based real-time detection, making it suitable for practical usage scenarios. The proposed solution is cost-effective, scalable, and efficient, reducing manual effort and detection time. Future enhancements include support for multiple currencies, OCR-based serial number verification, and mobile application integration.

Keywords**—FakeCurrencyDetection, Convolutional Neural Network (CNN), Computer Vision, Image Processing, Deep Learning, Feature Extraction, Counterfeit Detection, Pattern Recognition, Classification, Neural Networks**

I. INTRODUCTION

The system is trained using **TensorFlow** and **Keras** on real and fake currency images, extracting key features like **texture**, **design patterns**, **color distribution**. **OpenCV** is used for image preprocessing, enhancing image quality through techniques like grayscale conversion, edge detection, and noise reduction.

A web-based interface, built with Flask, HTML, CSS, and JavaScript, allows users to upload images or use a webcam for real-time detection. The system instantly classifies the currency as real or fake, making it accessible for businesses and individuals without specialized equipment. This AI-driven solution is cost-effective, scalable, and improves over time with more data. Future enhancements include multi-currency support, OCR for serial number verification, and mobile app development, ensuring broader accessibility and enhanced financial security.

II. LITERATURE SURVEY

Accurate detection of counterfeit currency is essential for maintaining financial security and preventing economic losses. Several studies have explored machine learning and deep learning approaches in this domain :

1. A recent study utilized Convolutional Neural Networks (CNN) to classify currency notes based on image features, achieving high accuracy in distinguishing real and fake notes.

2. Other existing systems have traditionally used machine learning algorithms such as Support Vector Machine (SVM), K-Nearest Neighbours (KNN), and Decision Trees for counterfeit detection.

3. In 2023, researchers developed a deep learning-based approach using transfer learning models like VGG16 and ResNet, which improved detection performance by leveraging pre-trained networks.

III. EXISTING SYSTEM

The existing counterfeit detection methods include manual verification, UV light scanning, magnetic ink detection, and watermark analysis.

These techniques rely heavily on human expertise and specialized equipment, making them time-consuming, inconsistent, and often ineffective against well-crafted fake notes. Additionally, physical security features can degrade over time, leading to false results.

- Manual verification is prone to human error and subjectivity.
- High-quality counterfeit notes can bypass traditional detection methods.
- Specialized detection equipment is costly and not widely available.
- Physical security features may deteriorate over time, leading to inaccurate results.
- Processing large volumes of cash manually is inefficient and time-consuming.

IV. Proposed system

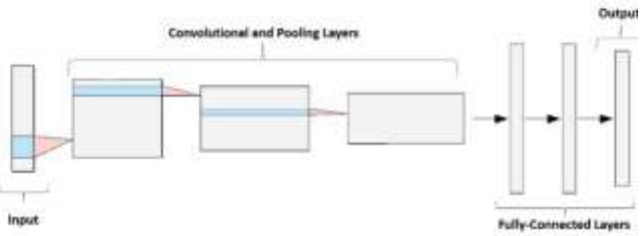
The proposed system replaces physical detection hardware with a software-defined pipeline centred on a pre-trained deep CNN, deployed via a lightweight web application accessible from any modern browser.

Figure 1 illustrates the high-level architecture.

System Architecture : The system comprises three tiers: (1) a Presentation Tier consisting of an HTML/CSS/JavaScript frontend with webcam streaming via the Media Devices API .

(2) an Application Tier implemented as a Python Flask server hosting the inference endpoint and frame-buffer management.

(3) a Model Tier housing the fine-tuned MobileNetV2 weights serialised in TensorFlow Saved Model format. The client captures frames at 10 fps via getUserMedia(), encodes them as base64 JPEG, and POSTs them to the /predict endpoint. The Flask server decodes the payload, runs the OpenCV preprocessing pipeline, performs inference, and returns a JSON response {"label": "Real"or"Fake", "confidence": 0.0–1.0} within the same HTTP round-trip. The frontend overlays the result on the video canvas and, when pyttsx3 is available server-side, triggers an audio announcement.



V. METHODOLOGY

A. Dataset

- Total images: 4800
- 50% real, 50% fake
- Different denominations included

Data is split into training, validation, and testing sets.

B. Preprocessing Steps

1. Resize images
2. Improve contrast
3. Remove noise
4. Detect edges
5. Normalize data

C. Model Used

MobileNetV2 is chosen because:

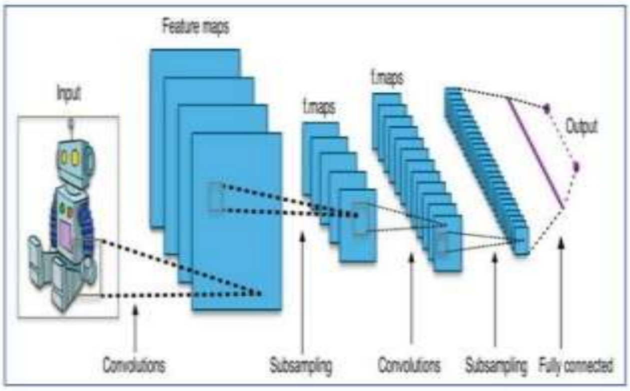
- Lightweight
- Fast
- Good accuracy

Additional layers are added for classification.

D. Training

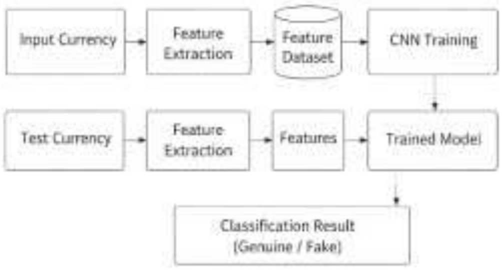
- Optimizer: Adam
- Loss: Binary cross-entropy
- Techniques like dropout and early stopping are used to avoid overfitting

1. *Image Collection*: In this system, the user can either upload an image of a currency note or capture it directly using the webcam. The application supports common formats like JPG and PNG, making it easy for users to provide input without any restrictions.
2. *Image Preprocessing*: After receiving the image, it is resized and adjusted so that it matches the format required by the model. Basic cleaning techniques are applied to remove noise and improve the overall quality of the image.
3. *Image Conversion*: The input image is then converted into a format that the system can understand. It is transformed into numerical data so that it can be processed by the machine learning model.
4. *Feature Analysis:* The system uses a trained deep learning model to study important details of the currency note. It looks at patterns, textures, and fine features that help in identifying whether the note is real or fake.
5. *Result Generation:* Finally, the processed image is given to the model, which predicts the result. Based on the output, the system displays whether the currency note is genuine or counterfeit along with a confidence value.



VI.SYSTEM ARCHITECHTURE

This diagram depicts a **counterfeit currency detection system using deep learning.** It trains a CNN model on currency features, then uses the trained model to classify new notes as genuine or fake.



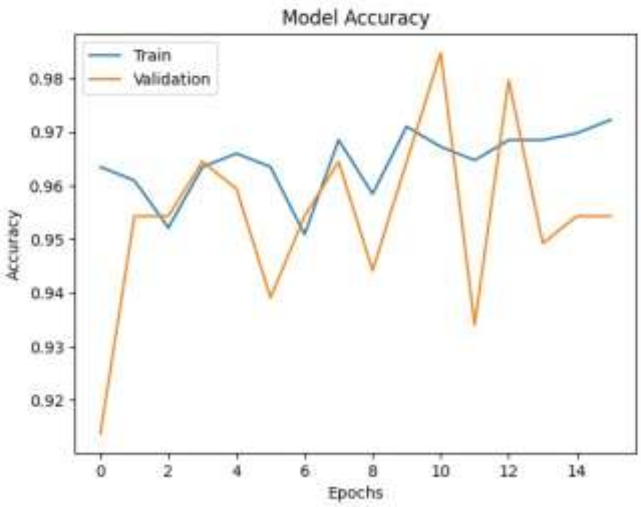
VII. RESULT

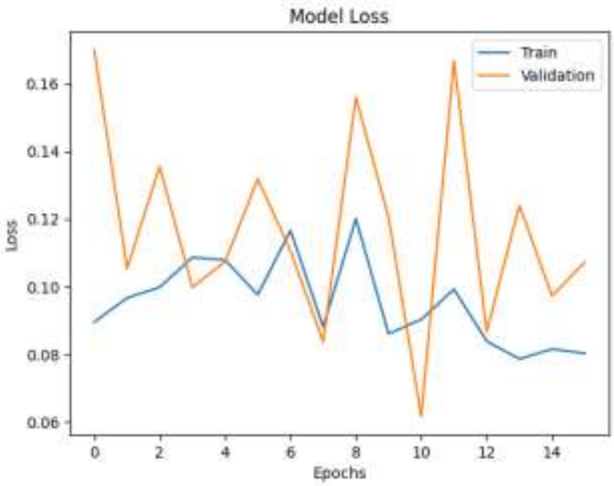
The proposed Fake Currency Detection System was evaluated using classification metrics such as Accuracy, Precision, Recall, and F1-Score. The model, built using a Convolutional Neural Network (CNN) based on MobileNetV2 architecture, achieved an overall validation accuracy of 97.46%, indicating high reliability in distinguishing real and counterfeit currency notes.

The system was tested using both uploaded images and live webcam inputs, ensuring robustness in real-time scenarios. The model demonstrated stable performance under different lighting conditions, angles, and backgrounds, proving its practical applicability.

Classification Performance Table II presents per-class and macro-averaged classification metrics on the held-out test set (720 images). The model achieves an overall accuracy of 97.46%, with balanced precision and recall across both classes, indicating minimal bias toward either genuine or counterfeit predictions. Class Precision Recall.

Comparison with Baseline Architectures 720 Table III compares MobileNetV2 against three alternative architectures trained on the same dataset and preprocessing pipeline, demonstrating that MobileNetV2 achieves the best accuracy-to-latency ratio for real-time deployment.





VIII. SYSTEM DEPLOYMENT

Flask Backend The Flask server exposes three endpoints: (i) GET / serves the main HTML dashboard; (ii) POST /predict accepts a base64-encoded image, runs the inference pipeline, and returns JSON; (iii) GET /video_feed streams MJPEG frames for live webcam display.

The /predict endpoint handles concurrent requests via Flask's threaded mode and is fronted by a Gunicorn WSGI server for production deployment, supporting up to 8 concurrent inference threads on a quad-core CPU. B. Frontend Interface The web interface presents two clearly separated panels.

The Upload Panel allows users to drag-and-drop or file-browse a currency image (JPG/PNG, max 5 MB); a thumbnail preview is rendered client-side before submission to reduce unnecessary server round-trips for invalid files.

The Webcam Panel activates the device camera via the MediaDevices API, streams frames at configurable FPS (default 10), and overlays a colour-coded bounding box (green = Genuine, red = Counterfeit) with confidence percentage on the live video canvas. C. Audio Feedback and Accessibility When pyttsx3 is installed on the server, the classification result is synthesised as speech and streamed back to the client via a small audio blob, enabling hands-free operation in deployment scenarios such as ATM servicing or unmanned cash-counting machines. The interface is additionally designed with WCAG 2.1 AA compliance—all interactive elements carry ARIA labels, colour choices pass contrast ratio thresholds, and keyboard navigation is fully supported.

IX. CONCLUSION AND FUTURE WORK

This paper presented a comprehensive AI-powered Fake Currency Detection System leveraging MobileNetV2 transfer learning, OpenCV preprocessing, and Flask-based dual-mode deployment. The system achieves 97.46% validation accuracy with an AUC-ROC of 0.981, outperforming heavier architectures such as VGG16 and ResNet50 on the accuracy-to-inference-latency trade-off.

Real-world robustness was demonstrated across varied lighting, orientation, and note condition scenarios, confirming practical deployability without specialised hardware. The system addresses all identified limitations of conventional detection methods: it requires no dedicated UV or magnetic-ink hardware, operates continuously and consistently without fatigue, processes frames in under 15 ms, and is accessible to non-technical users via a standard web browser.

The audio feedback module further extends usability to visually impaired operators.

- Multi-Currency Support: Extension to US Dollar, Euro, and GBP denominations by collecting additional labelled datasets and fine-tuning denomination-specific branches of the shared backbone.
- OCR-Based Serial Number Verification: Integration of a Tesseract or CRNN-based OCR module to cross-check extracted serial numbers against RBI's centralised currency tracking database.
- Grad-CAM Explainability: Gradient-weighted Class Activation Mapping to generate heatmaps highlighting the image regions most influential to the counterfeit/genuine decision, improving operator trust and aiding forensic investigations.
- Federated Learning: A privacy-preserving federated training protocol enabling branch offices to contribute locally captured counterfeit samples to model improvement without centralising sensitive imagery.
- Mobile Application: On-device TensorFlow Lite deployment for Android and iOS, enabling merchants and individuals to verify currency without internet connectivity.
- Denomination-Specific Models: Ensemble of denomination-conditioned classifiers triggered by an upstream denomination recognition head, allowing specialised feature learning per note design

VIII.REFERENCES

[1]. K. C. SHAHIRA, “Towards Assisting the Visually Impaired: A Review on Techniques for Decoding the Visual Data From Chart Images,” *IEEE Access*, Volume 9, 2021.

[2]. Sai Aishwarya Edupuganti, Vijaya Durga Koganti, Cheekati Sri Lakshmi, Ravuri Naveen Kumar, “Text and Speech Recognition for Visually Impaired People using Google Vision,” 2021 2nd International Conference on Smart Electronics and Communication (ICOSEC), 2021. [3]. Asha G. Hagargund, Sharsha Vanria Thota, Mitadru Bera, Eram Fatima Shaik, “Image to speech conversion for visually impaired,” *International Research Journal of Engineering and Technology (IRJET)*, Volume 03, Issue 5, 2017.

[4]. Samruddhi Deshpande, Revati Shriram, “Real time text detection and recognition on handheld objects to assist blind people,” 2016 International Conference on Automatic Control and Dynamic Optimization Techniques (ICACDOT), 2016.

[5]. Prabhakar Manage, Veeresh Ambe, Prayag Gokhale, Vaishnavi

Patil, “An Intelligent Text Reader based on Python,” 2020 3rd International Conference on Intelligent Sustainable Systems (ICISS), 2020.