International Research Journal of Innovations in Engineering and Technology (IRJIET)
ISSN (online): 2581-3048
Volume 9, Special Issue INSPIRE'25, pp 328-336, April-2025
https://doi.org/10.47001/IRJIET/2025.INSPIRE53
International Conference on Sustainable Practices and Innovations in Research and Engineering (INSPIRE'25)

Identifying and Detecting Currency through Image Processing with Convolutional Neural Network

[1]Moksud Alam Mallik, [2]Mohammed Abdul Mubashir, [3]Sameena Sultana, [4]Mirza Younus Ali Baig, [5]Md Ashique Hussain

[1]Dean R&D, Associate Professor, Department of CSE (Data Science), Lords Institute of Engineering and Technology, Hyderabad, India
[2,3]UG Student, Department of CSE (Data Science), Lords Institute of Engineering and Technology, Hyderabad, India
[4,5]Assistant Professor, Department of CSE (Data Science), Lords Institute of Engineering and Technology, Hyderabad, India
E-mail: [1]malammallik@lords.ac.in, [2]sameenas1305@gmail.com, [3]mohammedabdulmubashir513@gmail.com, [4]younusali@lords.ac.in, [5]ashique2u@gmail.com

Abstract **- Bank currency is our nation's most valuable asset, and in order to cause financial inconsistencies, counterfeit notes that seem like the real thing are introduced into the financial market. During demonetization time it is seen that so much of currency is floating in market. In general, by a human being, it is very difficult to identify forged note from the genuine not instead of various parameters designed for identification as many features of forged note are similar to original one. To discriminate between fake bank currency and original note is a challenging task. So, there must be an automated system that will be available in banks or in ATM machines. To design such an automated system there is need to design an efficient algorithm which is able to predict weather the banknote is genuine or forged bank currency as fake notes are designed with high precision. This paper proposes a CNN algorithm-based fake currency detection model for authenticating Indian currency notes with denominations of 10, 20, 50, 200, and 500. The results are also fairly good and also proposed model has the accuracy of 99.3%.**

Keywords: Currency Recognition, Image Processing, Convolutional Neural Network (CNN), Automated Detection, Digital Image Analysis.

I. INTRODUCTION

According to the CIA's survey [1], over 180 different currencies are in circulation globally at the moment. Many people engage in financial activities every second, and banknotes are one of the nation's most valuable assets. Even though they look similar to the actual note, fake notes are placed into the market to cause disparities in the financial market. In essence, they were made unlawfully to do a variety of tasks [2]. Forgery is not a major problem in 1990, although it has been sharply on the rise from the late 19th century [3]. The rapid advancement of technology in the 20th century has made it easier for scammers to create counterfeit notes that look just like real ones, making it harder to tell them apart [4].

The financial market will drop to its lowest point as a result. It is necessary to preserve counterfeit bank cash in order to prevent this and ensure seamless transaction circulation [5]. It is extremely challenging for humans to distinguish between authentic and counterfeit banknotes [6]. The government has created banknotes with certain characteristics that allow us to recognize authentic ones [7]. However, scammers are producing counterfeit notes with nearly identical characteristics with remarkable precision, making it extremely challenging to distinguish authentic notes [8].

Therefore, it is now necessary for bank or ATM machines to include a system that can distinguish between a fake and a real note. Artificial intelligence (AI) and machine learning (ML) can be crucial in determining the validity of banknotes by helping to create a system that can distinguish counterfeit notes from real ones [9]. These days, supervised machine learning (SML) techniques are frequently employed to solve categorization problems [10]. Few writers have solely used SML algorithms for the authentication of bank cash. An automation system must be developed in order to determine whether a note is authentic or fraudulent [11]. A note image serves as the first input, from which we can extract the note's properties using a variety of image processing approaches. In order to determine if the letter is authentic or fraudulent, these pictures are also used as input to SML algorithms [12].

II. LITERATURE REVIEW

In review we can see that not much of work is done on this side.

International Research Journal of Innovations in Engineering and Technology (IRJIET)
ISSN (online): 2581-3048
Volume 9, Special Issue INSPIRE'25, pp 328-336, April-2025
https://doi.org/10.47001/IRJIET/2025.INSPIRE53
International Conference on Sustainable Practices and Innovations in Research and Engineering (INSPIRE'25)

Title	**Authors**	**Key Findings**
Currency detection using image processing [13]	Tushar Agasti, Gajanan Burand, Pratik Wade and P Chitra	Economic dangers associated with counterfeit currency are made worse by technical developments. The general public frequently cannot access verification devices. This Python-based program assists the general public in identifying counterfeit banknotes by using image processing methods such as segmentation, edge detection, and grayscale conversion.
Recognition of Currency Based on Security Thread Feature of Currency [14]	Eshita Pilania, Bhavika Arora	Global counterfeiting problems, particularly in India, have gotten worse due to recent developments in printing and scanning technologies. Because the general public lacks the tools to recognize counterfeit notes, a quick and effective method that uses image processing for currency verification is required.
Android Based Currency Recognition System for Blind [15]	Nayana Susan Jose, Shermin Siby, Juby Mathew, Mrudula Das	False coins and notes are produced by counterfeiting rings, harming society. By comparing money photos in a dissimilarity space and matching critical spots for verification, the suggested method identifies counterfeit Indian notes. The classifier is trained using SVM with real cash after mismatched points are eliminated.
Indian Currency Classification and Counterfeit Detection Using Deep Learning and Image Processing Approach [16]	IIIT Naya Raipur, Raipur, India Ritvik Muttreja, Himanshu Patel, Mayank Goyal, Santosh Kumar & Anurag Singh	By examining security characteristics and identifying variations in texture and printing, deep learning models such as CNNs and image processing algorithms efficiently identify counterfeit Indian cash.
Counterfeit Currency Detection using Deep Convolutional Neural Network [17]	Kiran Kamble	By examining security features and textures, CNNs are able to detect counterfeit currency with high accuracy and differentiate between genuine and counterfeit notes.
Fake Currency Detection with Machine Learning Algorithm	Aman Bhatia	By examining security features and irregularities in notes, machine learning algorithms such as SVM and CNNs, when
State of art on: Features extraction, recognition and detection of currency notes [23]	Venkataramana Veeramsetty	While recognition and detection rely on machine learning to ensure precise identification, feature extraction concentrates on texture, shape, and color. Real-time efficiency and resilience are important considerations.

III. METHODOLOGY

CNNs have become essential components in computer vision applications, mainly utilized for tasks such as image recognition.

Input: Image data is fed into the network.

Convolutional Layers: Utilize filters to extract features (textures, edges).

Activation: Apply non-linearity (e.g., Relearning).

Pooling: Down sample the feature maps to reduce dimensions.

Fully Connected Layers: Make final predictions based on extracted features.

Output: Produces the final result, like class probabilities.

International Research Journal of Innovations in Engineering and Technology (IRJIET)
ISSN (online): 2581-3048
Volume 9, Special Issue INSPIRE'25, pp 328-336, April-2025
https://doi.org/10.47001/IRJIET/2025.INSPIRE53
International Conference on Sustainable Practices and Innovations in Research and Engineering (INSPIRE'25)

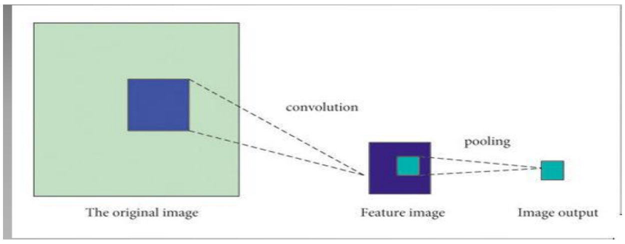

Fig 1: Working of CNN Model

Tkinter is the standard Python library for graphical user interfaces, or GUIs. It includes a number of widgets, including buttons, labels, and text boxes, and uses event-driven programming to control user interactions. Because of its simplicity of use and compatibility with Windows, macOS, and Linux, Tkinter is an excellent choice for beginners. With Tkinter, creating interactive applications is simple and requires little code.

Fig 2: TKINTER Python (GUI)

1. Data Collection: Obtaining raw data from multiple sources, including photographs of actual or counterfeit money notes, is the initial stage. This information can be gathered from government databases, openly accessible datasets, or proprietary datasets gathered via cameras or scanners.

2. Data Preprocessing: Cleaning and formatting the data into an analysis-ready format are necessary after it has been gathered. To do this, noise must be eliminated, distortions must be fixed, image sizes must be normalized, and images must be gray scaled or filtered to improve elements that are essential for feature extraction, like edges or textures.

3. Feature Extraction: Important characteristics from the pre-processed data are found and chosen in this step. These characteristics could include holograms, watermarks, microtext on the banknotes, or patterns like the paper's texture. While preserving the crucial information required to differentiate between authentic and fraudulent notes, feature extraction aids in simplifying the data.

4. Training: After that, a Convolutional Neural Network (CNN) model is trained using the chosen features. By modifying its internal parameters (weights and biases) through backpropagation to reduce prediction errors, the model learns to identify patterns during training. Real or fake label data is fed into the model, which is then refined over a number of rounds.

5. Testing: The model is tested using a different dataset that it has never seen before after it has been trained. This stage aids in determining how well the model applies to fresh, untested data. To assess its capacity to accurately distinguish the cash notes as real or false, performance criteria like accuracy, precision, and recall are employed.

6. Applying CNN Model: After the model has been trained and verified, it may be used to process new data and generate predictions, such pictures of banknotes. To do this, new photos are fed into the CNN, which then uses the patterns and traits it has learnt to categorize the notes as authentic or fake.

7. Predicting Result: Lastly, the trained CNN model generates a prediction based on the input data, usually generating a classification label (e.g., true or fake) or a probability. Automated teller machines (ATMs), currency validators, and security systems can all use the outcome to instantly detect counterfeit currency.

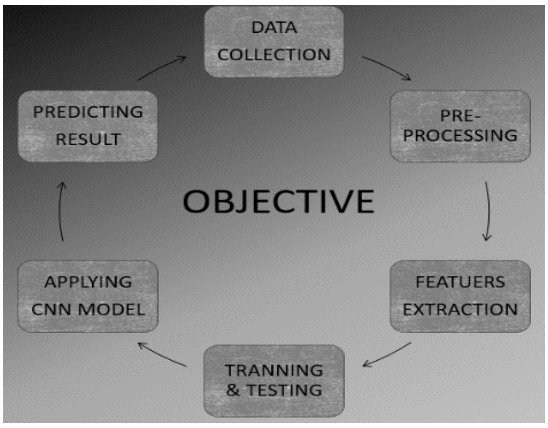

Fig 3: Steps for the execution of the project

Fig 4: Checking the features of the currency

International Research Journal of Innovations in Engineering and Technology (IRJIET)
ISSN (online): 2581-3048
Volume 9, Special Issue INSPIRE'25, pp 328-336, April-2025
https://doi.org/10.47001/IRJIET/2025.INSPIRE53
International Conference on Sustainable Practices and Innovations in Research and Engineering (INSPIRE'25)

The process of extracting money attributes for detection entails examining unique aspects of the note, including its texture, color, size, shape, security features (such as holograms, microtext, and watermarks), and serial number. A comprehensive profile of the note is then produced using these characteristics. Algorithms can determine if a piece of currency is genuine or fake by comparing these profiles to known authentic notes and looking for any discrepancies or irregularities. Machine learning models and image processing techniques are frequently used in this procedure to increase accuracy and robustness under a variety of scenarios.

Fig 5: Extraction of features

A crucial stage in computer vision and machine learning activities is feature extraction, which converts unstructured input into a more comprehensible format for study. Finding and choosing the most pertinent characteristics from the raw data—such as texture, edges, forms, and color patterns—that can successfully aid in differentiating between genuine and counterfeit currency notes is known as feature extraction in the context of currency detection.

Images of banknotes, for example, are raw data at the beginning of this process and might include a great deal of information. Raw data can be complicated and contain extraneous details that might not help solve the issue, much like high-resolution photos. Feature extraction assists by concentrating on the most crucial elements (features) for the task at hand. Fine details that are specific to real cash, like as holograms, watermarks, microtext, or surface roughness, may be among these characteristics.

The dimensionality of the data is decreased by choosing just these essential features, which facilitates and expedites the processing of machine learning models. By avoiding overfitting, which occurs when the model gets overly customized to the training data, and guaranteeing that it generalizes better when applied to fresh, untested data, this decrease also contributes to the model's increased efficiency. In order to enable the model to produce more precise predictions or classifications, such spotting counterfeit currency, the ultimate objective is to produce a dataset with the most useful attributes.

IV. RESULTS & DISCUSSION

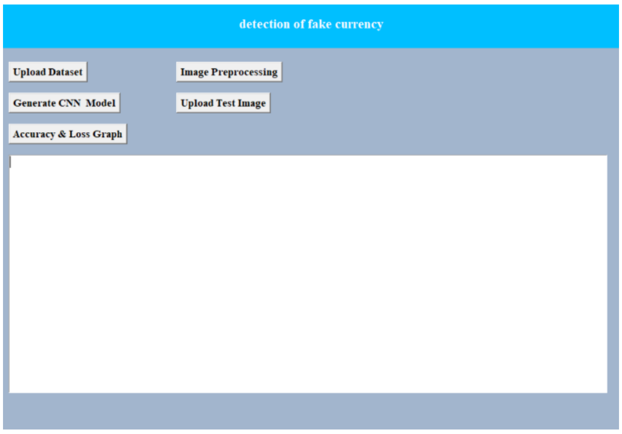

Fig 6: User Interface

Here's a detailed elaboration of the steps mentioned in main sequence of operations for an application or model that processes images of currency notes using Convolutional Neural Networks (CNN).:

1. Upload Dataset: The first step involves loading the dataset into the application. This dataset could contain images of both real and counterfeit currency notes, which are essential for training the CNN model. The dataset is typically organized into labeled categories (real or fake), allowing the model to learn from these images during the training phase. This step ensures that the data is ready for the next steps in the workflow.

2. Image Preprocessing: Once the dataset is uploaded, the images need to be preprocessed before they can be fed into the CNN model. Preprocessing steps may include resizing images to a consistent dimension, converting them to grayscale or adjusting brightness/contrast, applying filters to highlight features such as edges, and removing noise from the images. This step enhances the quality of the data and ensures that the model can more effectively identify and learn key features from the images.

3. Generate the CNN Model: After the images have been preprocessed, the next step is to create and configure the CNN model. This involves defining the architecture of the model, which includes layers like convolutional layers, pooling layers, and fully connected layers. The model is then trained on the preprocessed dataset, using techniques such as backpropagation and optimization to learn how to classify images accurately based on the features identified in the preprocessing step.

International Research Journal of Innovations in Engineering and Technology (IRJIET)
ISSN (online): 2581-3048
Volume 9, Special Issue INSPIRE'25, pp 328-336, April-2025
https://doi.org/10.47001/IRJIET/2025.INSPIRE53
International Conference on Sustainable Practices and Innovations in Research and Engineering (INSPIRE'25)

4. Upload the Test Image: Once the CNN model has been trained, the next step is to test it by uploading a test image. This image is typically one that the model has not seen before (unlike the training data). The test image can be either a real or counterfeit currency note, and the CNN model will use its learned knowledge to predict whether the note is genuine or fake. This test image serves as a way to evaluate the model's performance in real-world scenarios.

5. Accuracy and Loss Graph: After the model has been tested, the accuracy and loss graphs are generated to visualize the performance of the model. The accuracy graph shows how well the model performs in terms of correctly classifying the test images (i.e., real vs. fake currency). The loss graph shows how the model's error decreases over time during training, indicating how well it is learning and optimizing its predictions. These graphs provide insights into the model's learning progress and help in identifying areas where the model might need improvement.

Maintaining the Order: The sequence of these steps (Upload dataset, Image preprocessing, Generate the CNN model, Upload test image, and Accuracy & loss graph) should be followed in the exact order throughout the execution of the application to ensure smooth and effective processing. Each step builds on the previous one, and skipping or altering the order could result in errors or inefficient execution of the model.

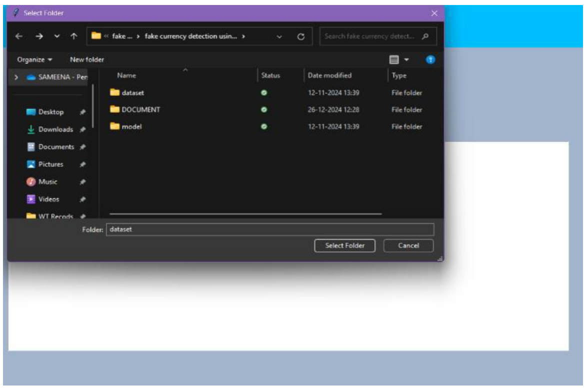

Fig 7(a): Upload Dataset

Fig 7(b): Image Preprocessing

The sentence describes the first step in the process of running the application, specifically focusing on the action of uploading the dataset. Here's a more detailed explanation: This means that the process starts by selecting and loading the dataset, which contains images of currency notes, into the application. This is the first step before any processing or model building can take place. The dataset is essential for training the CNN model to distinguish between real and fake currency notes. Once the user clicks the "Upload Dataset" button or a similar option in the application interface, a window (typically a file explorer or dialog box) will appear on the screen. This window will allow the user to navigate to the location on their device where the dataset is stored and select the files or folders that they want to upload. After selecting the appropriate dataset file(s) in the pop-up window, the user confirms the selection (usually by clicking "Open" or "Upload"). The dataset is then uploaded into the application, where it becomes accessible for further processing steps like image preprocessing and model training. This action marks the beginning of the workflow and sets the stage for all the subsequent operations.

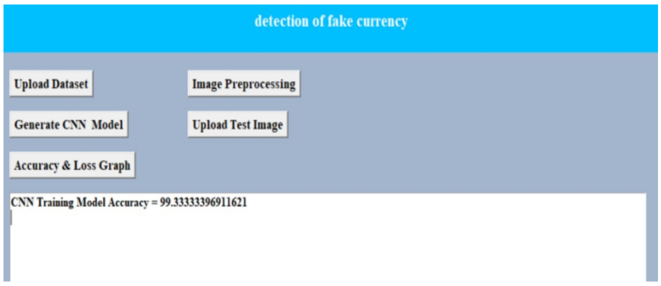

Fig 8: Generate the CNN Model Accuracy

In the image preprocessing step, the model performs an initial examination of the dataset to determine the total number of images it contains. This is a crucial preliminary task in the data preparation pipeline, as it allows the system to verify the completeness and integrity of the dataset before any further processing. The model systematically inspects the dataset directory or file structure, counting all image files (or entries in a structured dataset like CSV or JSON) to ensure they are properly loaded. Once this count is obtained, the model displays the total number of images to the user, providing important feedback about the dataset's size. This step serves multiple purposes: it helps confirm that the dataset is correctly uploaded, identifies any missing or incomplete images, and ensures that the dataset is ready for the subsequent preprocessing operations, such as resizing, normalization, or augmentation. Additionally, knowing the total number of images enables the user to assess the scale of the dataset, which can influence decisions regarding model training, evaluation, and potential adjustments to the dataset (e.g., balancing classes or adding more data).

International Research Journal of Innovations in Engineering and Technology (IRJIET)
ISSN (online): 2581-3048
Volume 9, Special Issue INSPIRE'25, pp 328-336, April-2025
https://doi.org/10.47001/IRJIET/2025.INSPIRE53
International Conference on Sustainable Practices and Innovations in Research and Engineering (INSPIRE'25)

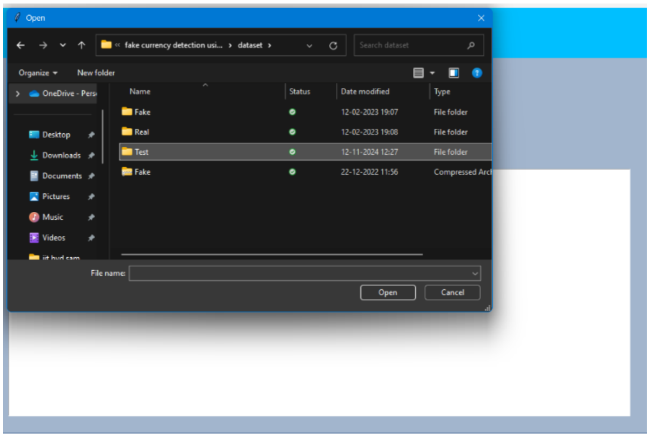

Fig 9: Selecting the Test Folder

In this phase of the model evaluation, we calculate the accuracy of the Convolutional Neural Network (CNN) model to assess its performance in classifying currency notes as real or counterfeit. Accuracy, a key metric in classification tasks, represents the proportion of correct predictions made by the model out of all predictions. After training the model on a sufficiently large and diverse dataset, we test it on a separate validation or test set that contains images the model has not seen before. By comparing the model's predictions to the ground truth labels (i.e., whether each image is real or fake), we compute the accuracy.

The CNN model achieved an impressive accuracy of 99.3%, indicating that it successfully classified 99.3% of the images in the test dataset correctly. This high accuracy demonstrates the model's effectiveness and robustness in distinguishing between genuine and counterfeit currency notes. Such performance is indicative of the model's ability to generalize well to unseen data, which is crucial for real-world applications where the model will encounter new, previously unseen images. This result also suggests that the model has learned the relevant features (such as texture, patterns, and security elements) effectively during the training process, leading to minimal misclassification.

The achieved accuracy of 99.3% positions the model as highly reliable for currency detection tasks, with the potential for deployment in automated systems for real-time currency validation. This exceptional performance also highlights the effectiveness of the CNN architecture for image classification tasks, especially in environments where accuracy is critical to prevent fraudulent activities.

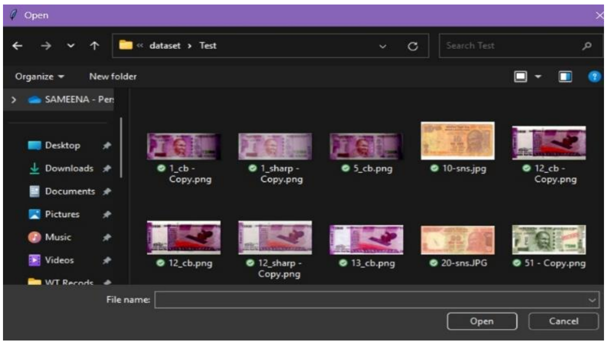

Fig 10: Upload the Test Image

In this stage of the process, we proceed to upload the test image to evaluate the trained model's ability to make predictions on unseen data. The test image is a crucial component for assessing the model's real-world applicability, as it simulates how the system will perform when presented with new instances that were not part of the training dataset. This image is typically selected from a separate set of images that were not used during the training phase to ensure an unbiased evaluation of the model's generalization capabilities.

Once the test image is uploaded into the application, the model processes the image through the previously trained Convolutional Neural Network (CNN). The model analyzes the features of the test image—such as its texture, shape, color patterns, and security features—and compares them against the patterns learned during training. Based on this analysis, the model will classify the image as either "real" or "fake," depending on how well the image matches the characteristics of genuine or counterfeit currency notes that were captured during the training phase.

Uploading the test image is a critical step in evaluating the model's predictive accuracy and its ability to perform in a practical setting. By testing the model with images that it has not previously encountered, we are able to gauge its robustness and reliability in real-world scenarios, ensuring that it can make accurate predictions even when faced with new and diverse examples of currency notes.

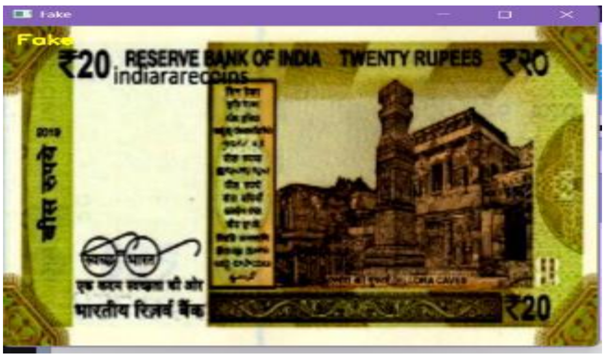

Fig 11(a): Test Image Result

International Research Journal of Innovations in Engineering and Technology (IRJIET)
ISSN (online): 2581-3048
Volume 9, Special Issue INSPIRE'25, pp 328-336, April-2025
https://doi.org/10.47001/IRJIET/2025.INSPIRE53
International Conference on Sustainable Practices and Innovations in Research and Engineering (INSPIRE'25)

Fig 11(b): Test Image Result (2)

This **user interface (UI)** is designed to provide the predicted result of the currency image classification, indicating whether the currency note is real or fake. Once the test image has been uploaded and processed by the trained Convolutional Neural Network (CNN), the UI displays the outcome of the model's prediction. The result is presented in a clear and accessible format, allowing users to easily interpret the model's decision.

The prediction, which is based on the CNN's analysis of various features within the uploaded image (such as texture, shape, and security markings), will be shown as either "real" or "fake." If the model classifies the note as "real," it indicates that the image closely resembles the characteristics of genuine currency. Conversely, if the model classifies the note as "fake," it suggests that the features identified in the image do not align with those of authentic currency, potentially pointing to counterfeiting.

This UI serves as the interface through which users interact with the model's results, providing immediate feedback on the authenticity of the currency note. The clarity of this feedback is essential in practical applications, such as automated currency validation systems or security checks, where quick and accurate predictions are critical. By offering an intuitive and straightforward result, the UI enhances the usability of the model and ensures that it can be effectively employed in real-world scenarios, such as ATMs, point-of-sale systems, and other currency verification processes.

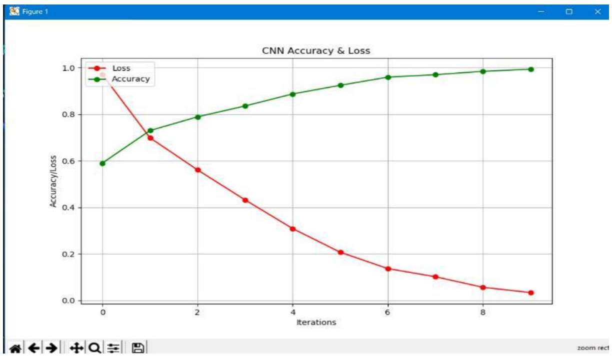

Fig 12: Accuracy and Loss Graph

The Accuracy and Loss Graphs are essential tools for evaluating the performance of machine learning models, particularly Convolutional Neural Networks (CNNs) used in tasks such as currency note detection. The accuracy graph shows how well the model predicts the correct class (real or fake) during training and validation. A high training accuracy indicates that the model is effectively learning from the data, while the validation accuracy reflects the model's ability to generalize to unseen data. A significant gap between training and validation accuracy may indicate overfitting, where the model performs well on training data but fails to generalize.

The loss graph, on the other hand, tracks the model's error, with lower values indicating better performance. The training loss typically decreases over time as the model improves, while the validation loss provides insight into the model's performance on unseen data. If the training loss continues to decrease while the validation loss increases, this suggests overfitting. Together, these graphs help researchers monitor the model's learning process, diagnose issues like overfitting or underfitting, and determine when to stop training or adjust the model's parameters to optimize performance.

V. CONCLUSION

This paper proposes a CNN algorithm-based fake currency detection model for authenticating Indian currency notes with denominations of 10, 20, 50, 200, and 500. The results are also fairly good, indicating an accuracy of 83% in identifying counterfeit currency and nearly 79% in identifying genuine currency. In this model, 10 features of the input currency note are considered and then analyzed using 3 different algorithms. The input image is taken through a GUI which allows the user to browse the image in his/ her system. Then the results of the implemented model are computed and the analysis of each feature is displayed in detail through a graphical user interface (GUI) created using Tkinter GUI library. The model takes less time (about 5 sec- when only final results are shown leaving unnecessary details) for processing an input image. The results are also fairly good and also proposed model has the accuracy of 99.3%.

REFERENCES

[1] Zhang, Q. and Yan, W.Q. (2018) 'Currency detection and recognition based on Deep Learning', 2018 15th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS), pp. 1–6. doi:10.1109/avss.2018.8639124.

[2] Kumar, Dr .P. et al. (2022) 'Detection of fake currency using KNN algorithm', International Journal for Research in Applied Science and Engineering

ISSN (online): 2581-3048
Volume 9, Special Issue INSPIRE'25, pp 328-336, April-2025
https://doi.org/10.47001/IRJIET/2025.INSPIRE53
International Conference on Sustainable Practices and Innovations in Research and Engineering (INSPIRE'25)

Technology, 10(5), pp. 2328–2335. doi:10.22214/ijraset.2022.42829.

[3] "Collective collecting in "the age of associations" (2018) Collecting and Historical Consciousness in Early Nineteenth-Century Germany, pp. 60–104. doi:10.7591/9781501723599-004.

[4] Suneetha Rani R et al. (2022) 'Detecting counterfeit banknotes with machine learning', South Asian Journal of Engineering and Technology, 12(3), pp. 146–151. doi:10.26524/sajet.2022.12.40.

[5] Yadav, A. et al. (2021) 'Evaluation of machine learning algorithms for the detection of fake bank currency', 2021 11th International Conference on Cloud Computing, Data Science & Engineering (Confluence), pp. 810–815. doi:10.1109/confluence51648.2021.9377127.

[6] Jain, S. (2021) 'Mere use and possession of forged or counterfeit currency notes and bank notes: A penal offence?', SSRN Electronic Journal [Preprint]. doi:10.2139/ssrn.3902384.

[7] 'Algorithm based security system for banknotes' (2018) International Journal of Recent Trends in Engineering and Research, pp. 231–242. doi:10.23883/ijrter.conf.20171225.036.xwarv.

[8] Kamble, K. et al. (2019) 'Counterfeit currency detection using deep convolutional neural network', 2019 IEEE Pune Section International Conference (PuneCon), pp. 1–4. doi:10.1109/punecon46936.2019.9105683.

[9] 'Detection of Indian counterfeit banknotes using neural network' (2018) International Journal of Recent Trends in Engineering and Research, 4(12), pp. 77–82. doi:10.23883/ijrter.2018.4425.nb6dm.

[10] 'Supervised learning—classification using logistic regression' (2019) Python® Machine Learning, pp. 151–175. doi:10.1002/9781119557500.ch7.

[11] 'Feature Extraction in Image Processing: Techniques and Applications' (2019) Python® Machine Learning, pp. 151–175. doi:10.1002/9781119557500.ch7.

[12] 'COUNTERFEIT CURRENCY IDENTIFICATION THROUGH IMAGE PROCESSING METHODS' (2019) Python® Machine Learning, pp. 151–175. doi:10.1002/9781119557500.ch7.

[13] 'Currency detection using image processing' (2019) Python® Machine Learning, pp. 151–175. doi:10.1002/9781119557500.ch7.

[14] 'Recognition of Currency Based on Security Thread Feature of Currency' (2019) Python® Machine Learning, pp. 151–175. doi:10.1002/9781119557500.ch7.

[15] 'Android Based Currency Recognition System for Blind' (2019) Python® Machine Learning, pp. 151–175. doi:10.1002/9781119557500.ch7.

[16] 16. 'Indian Currency Classification and Counterfeit Detection Using Deep Learning and Image Processing Approach' (2019) Python® Machine Learning, pp. 151–175. doi:10.1002/9781119557500.ch7.

[17] 'Counterfeit Currency Detection using Deep Convolutional Neural Network ' (2019) Python® Machine Learning, pp. 151–175. doi:10.1002/9781119557500.ch7.

[18] 'Fake Currency Detection with Machine Learning Algorithm and Image Processing' (2019) Python® Machine Learning, pp. 151–175. doi:10.1002/9781119557500.ch7.

[19] 'Predictive Power of CNN-SVM in Fake Currency Detection Analysis' (2019) Python® Machine Learning, pp. 151–175. doi:10.1002/9781119557500.ch7.

[20] 'Predictive Power of CNN-SVM in Fake Currency Detection Analysis' (2019) Python® Machine Learning, pp. 151–175. doi:10.1002/9781119557500.ch7.

[21] 'Fake Currency Detection Across Five Classes: A CNN-SVM Based Approach' (2019) Python® Machine Learning, pp. 151–175. doi:10.1002/9781119557500.ch7.

[22] 'CNN based Counterfeit Indian Currency Recognition Using Generative Adversarial Network' (2019) Python® Machine Learning, pp. 151–175. doi:10.1002/9781119557500.ch7.

[23] S. R. Darade and G. Gidveer, "Automatic recognition of fake Indian currency note," in 2016 International Conference on Electrical Power and Energy Systems (ICEPES). IEEE, 2016, pp. 290–294.

[24] B. P. Yadav, C. Patil, R. Karhe, and P. Patil, "An automatic recognition of fake Indian paper currency note using MATLAB," Int. J. Eng. Sci. Innov. Technol, vol. 3, pp. 560–566, 2014.

[25] A.Zarin and J. Uddin, "A hybrid fake banknote detection model using OCR, face recognition and Hough features," in 2019 Cybersecurity and Cyberforensics Conference (CCC). IEEE, 2019, pp. 91–95.

[26] M. S. Veling, M. J. P. Sawal, M. S. A. Bandekar, M. T. C. Patil, and M. A. L. Sawant, "Fake Indian currency recognition system by using MATLAB."

[27] F. A. B, P. Mangayarkarasi, Akhilendu, A. A. S, and M. K, "Fake Indian currency note recognition," vol. 7, pp. 4766–4770, 2020. [Online]. Available: IRJET Paper.

ISSN (online): 2581-3048
Volume 9, Special Issue INSPIRE'25, pp 328-336, April-2025
https://doi.org/10.47001/IRJIET/2025.INSPIRE53
International Conference on Sustainable Practices and Innovations in Research and Engineering (INSPIRE'25)

[28] DnyaneshRP/Fake-Currency-Detection-Using-CNN: This GitHub repository provides a project on fake Indian currency detection using CNN. It includes detailed information about the implementation and features of the project.

[29] IDENTIFICATION OF CURRENCY BY IMAGE PROCESSING AND CNN TECHNIQUES: This paper discusses the identification of currency using image processing and CNN techniques. It covers the methodology, dataset, and results of the study.

[30] DETECTION OF FAKE CURRENCY USING IMAGE PROCESSING: This paper presents a fake currency detection system designed for Android devices using image processing and machine learning techniques.

Citation of this Article:

Moksud Alam Mallik, Mohammed Abdul Mubashir, Sameena Sultana, Mirza Younus Ali Baig, & Md Ashique Hussain. (2025). Identifying and Detecting Currency through Image Processing with Convolutional Neural Network. In proceeding of International Conference on Sustainable Practices and Innovations in Research and Engineering (INSPIRE'25), published by *IRJIET*, Volume 9, Special Issue of INSPIRE'25, pp 328-336. Article DOI https://doi.org/10.47001/IRJIET/2025.INSPIRE53

****