International Journal of Research Publication and Reviews

Journal homepage: www.ijrpr.com ISSN 2582-7421

Fake Currency Detection Using Image Processing

Ms. K.Vindhiya Vardhini [1], Ms. R.Anushya[2]

[1]Student of II MSC (Computer Science), Department of Computer Science, VLB Janakiammal College of Arts and Science, Kovaipudur, Coimbatore, India.

[2]MCA, M. PHIL, Assistant Professor, Department of Computer Science, VLB Janakiammal College of Arts and Science, Kovaipudur, Coimbatore, India

ABSTRACT:

This project focuses on developing a **"FAKE CURRENCY DETECTION SYSTEM USING IMAGE PROCESSING TECHNIQUES"**, specifically a Convolutional Neural Network (CNN), to classify currency notes as "Real" or "Fake." The model is built using the Keras library with a Sequential architecture comprising convolutional layers, pooling layers, and dense layers. The training process uses data augmentation techniques, including random shear, zoom, and horizontal flips, to improve the model's robustness. The dataset is divided into training and testing sets, and the model is trained to identify fake and real currency notes based on image features. After training, the model is saved as a .h5 file, which is later used to predict the authenticity of new currency images. The system loads the saved model weights, processes the test image, and outputs the classification result, either "Real" or "Fake." This approach is aimed at automating the detection of counterfeit currency, offering an efficient solution to enhance security and prevent fraud.

Keywords: Convolutional Neural Network, data augmentation techniques, Accuracy, comprising convolutional layers, pooling layers, and dense layers

Introduction:

The issue of counterfeit currency is a persistent problem that affects economies globally, resulting in financial losses and disruptions in the circulation of legitimate money. Traditional methods of detecting counterfeit currency, such as manual inspection or the use of security features, are time-consuming and often prone to human error. To address this challenge, this project proposes a *Fake Currency Detection System* powered by *deep learning techniques*, specifically the use of *Convolutional Neural Networks (CNNs)***.** The system is designed to classify currency notes as either "Real" or "Fake" based on the analysis of their visual features. By utilizing CNNs, the model automatically extracts and learns complex patterns from images of currency notes, allowing it to effectively distinguish between authentic and counterfeit bills.The model is developed using the *Keras library* in Python, which facilitates the design of deep learning models through its simple and efficient framework. The architecture of the model consists of several layers, including convolutional layers for feature extraction, pooling layers for dimensionality reduction, and dense layers for classification.

The project incorporates *data augmentation* techniques such as random shearing, zooming, and horizontal flipping to improve the robustness of the model and reduce the likelihood of overfitting. Once trained, the model is saved as a *.h5 file*, making it portable and easy to use for predicting the authenticity of new currency images in real-world applications.

OBJECTIVES:

The primary objective of this project is to develop an automated system that can reliably detect counterfeit currency notes using deep learning methods. The main goals are to create a robust deep learning model that can classify currency notes as "Real" or "Fake" based on their visual features. The project also aims to employ *data augmentation* to increase the diversity of the training dataset, ensuring that the model generalizes well to new and unseen images. Another key objective is to assess the performance of the trained model by evaluating its accuracy and other performance metrics such as precision and recall, ensuring it provides reliable and accurate predictions. Additionally, the project seeks to save the trained model as an *.h5 file*, which can be loaded and deployed in real-world scenarios, enabling businesses, banks, or ATMs to automate the process of counterfeit detection.

SCOPE OF THE STUDY:

The scope of this project is focused on the development and deployment of a *fake currency detection system* using image-based input data. The project primarily focuses on designing and training a *Convolutional Neural Network (CNN)* to recognize distinguishing features of real and fake currency notes. The process begins with collecting a dataset of images representing both real and counterfeit currency notes, which are then preprocessed to ensure uniformity in size and format. To enhance the diversity of the training data, *data augmentation* techniques such as random zoom, horizontal flipping, and shearing are applied, allowing the model to become more resilient to different orientations and variations of the currency notes. The model is trained using this preprocessed data, and its performance is carefully evaluated by testing it on a separate validation set. The evaluation considers accuracy, precision, recall, and other relevant metrics to determine the model's effectiveness in correctly classifying currency notes. Once the model achieves satisfactory performance, it is saved as a *.h5 file*, making it easily deployable for use in practical applications. The system is aimed at enhancing the automation of counterfeit currency detection, providing a more efficient, reliable, and scalable solution compared to traditional methods. While the project is focused on detecting counterfeit bills in 2D images, it does not extend to real-time video or hardware integration, such as working with ATMs or physical scanners. In conclusion, this project aims to create an efficient and automated *Fake Currency Detection System* using deep learning techniques, offering a practical solution to the global problem of counterfeit currency, while ensuring that the system can be deployed in various real-world scenarios.

PROBLEM DEFINITION:

The problem of counterfeit currency is a significant global issue that impacts both consumers and financial institutions. As counterfeiters become more sophisticated in replicating genuine currency, manual detection methods and traditional security features such as watermarks, microtext, and holograms are no longer sufficient to guarantee the authenticity of banknotes. This growing sophistication of counterfeit currency, along with the increase in digital payment methods, has made it increasingly difficult to identify fake currency in real-time using conventional methods. The manual verification of currency notes is time-consuming, prone to human error, and not scalable, especially in environments such as banks, retail stores, and ATMs. Currently, counterfeit currency detection relies heavily on human inspection, which is not only slow but also susceptible to inconsistencies. In many cases, even trained professionals may fail to identify counterfeit notes, especially those of high quality. This problem becomes even more critical in environments where large numbers of currency notes are processed daily, such as in cash handling at banks, vending machines, and other financial institutions. Consequently, there is a need for an automated, reliable, and efficient system that can quickly and accurately identify counterfeit currency.

The objective of this project is to address this problem by developing a Fake Currency Detection System using deep learning techniques, specifically Convolutional Neural Networks (CNNs), to automatically classify currency notes as either "Real" or "Fake" based on their visual features. This system will be able to process currency images and identify fraudulent bills with a high level of accuracy, minimizing human intervention, reducing the time required for verification, and improving the overall efficiency of currency handling processes. The challenge lies in developing a model that can accurately distinguish between real and counterfeit notes by recognizing subtle patterns and features in the images, despite variations in lighting, orientation, and image quality. Additionally, the model needs to generalize well across different currency denominations and regions, making it adaptable for widespread use in various real-world applications, such as ATMs, retail stores, and currency exchange services. The system will aim to provide a scalable, cost-effective solution to automate currency validation, enhance security, and reduce the negative economic impact caused by counterfeit currency.

INTRODUCTION TO BACKEND AND FRONT END:

PYTHON:

Python is an interpreter, object-oriented, high-level programming language with dynamic semantics. Its high-level built in data structures, combined with dynamic typing and dynamic binding; make it very attractive for Rapid Application Development, as well as for use as a scripting or glue language to connect existing components together. Python's simple, easy to learn syntax emphasizes readability and therefore reduces the cost of program maintenance. Python supports modules and packages, which encourages program modularity and code reuse. The Python interpreter and the extensive standard library are available in source or binary form without charge for all major platforms, and can be freely distributed.

Python is Free

The Python interpreter is developed under an OSI-approved open-source license, making it free to install, use, and distribute, even for commercial purposes. A version of the interpreter is available for virtually any platform there is, including all flavors of Unix, Windows, MAC OS, smart phones and tablets, and probably anything else you ever heard of. A version even exists for the half dozen people remaining who use OS/2.

Python is Portable

Because Python code is interpreted and not compiled into native machine instructions, code written for one platform will work on any other platform that has the Python interpreter installed. (This is true of any interpreted language, not just Python.)

Python is Simple

As programming languages go, Python is relatively uncluttered, and the developers have deliberately kept it that way. A rough estimate of the complexity of a language can be gleaned from the number of keywords or reserved words in the language. These are words that are reserved for special meaning by the compiler or interpreter because they designate specific built-in functionality of the language.

HTML:

HTML (Hyper Text Markup Language) is the most basic building block of the Web. It defines the meaning and structure of web content. Other technologies besides HTML are generally used to describe a web page's appearance/presentation (CSS) or functionality/behaviour (JavaScript). "Hypertext" refers to links that connect web pages to one another, either within a single website or between websites. Links are a fundamental aspect of the Web. By uploading content to the Internet and linking it to pages created by other people, you become an active participant in the World Wide Web.

It is the standard markup language used for creating web pages and web applications. HTML forms the backbone of most websites, providing the basic structure upon which CSS and JavaScript are applied to manipulate appearance and functionality.

HTML DOCUMENT STRUCTURE

HTML document contains the text (the content of the page) with embedded tags, which provide instruction, appearance and function of the content. The HTML document is divided into two major portions: the head and the body. The head contains information about the document such as the title and "meta" information describing the content. The body contains the actual contents of the document (the parts that is displayed in the browser window).

CSS (CASCADING STYLE SHEETS)

Cascading Style Sheets, fondly referred to as CSS, is a simple design language intended to simplify the process of making web pages presentable.

CSS handles the look and feel part of a web page. Using CSS, you can control the color of the text, the style of fonts, the spacing between paragraphs, how columns are sized and laid out, what background images or colors are used, as well as a variety of other effects. It is a style sheet language used for describing the presentation of a document written in HTML or XML (including XML dialects like SVG or XHTML). CSS describes how elements should be rendered on screen, on paper, in speech, or on other media.

JAVASCRIPT

JavaScript is a lightweight, interpreted programming language. It is designed for creating network-centric applications. It is complimentary to and integrated with Java. JavaScript is very easy to implement because it is integrated with HTML. It is open and cross-platform. JavaScript is a programming language that enables interactive web pages. It is an essential part of web applications, allowing for client-side script to interact with the user, control the browser, communicate asynchronously, and alter document content that is displayed.

METHODOLOGY:

The methodology of the *Fake Currency Detection System* is designed to utilize deep learning techniques, specifically *Convolutional Neural Networks (CNNs)*, to automatically classify currency notes as "Real" or "Fake." This approach leverages modern image processing and machine learning techniques to analyse the visual features of currency notes and distinguish between authentic and counterfeit bills. The methodology is divided into several key stages, including data collection, preprocessing, model design, training, evaluation, and deployment.

1) Data Collection

The first critical step in the methodology is the collection of a comprehensive *dataset* consisting of images of both real and counterfeit currency notes. The dataset needs to cover different denominations, currencies, and various types of counterfeit bills to ensure the model generalizes well. The dataset should ideally include high-resolution images captured under varied conditions to include variations in lighting, orientation, and quality. If a pre-existing dataset is not available, it will be necessary to create one by taking controlled photographs of real and counterfeit notes to accurately represent the visual characteristics that differentiate authentic bills from fake ones.

2) Data Preprocessing

After collecting the dataset, the next step is *data preprocessing*, which is crucial for preparing the data for model training. First, the images are *resized* to a uniform dimension to ensure compatibility with the neural network input requirements. The pixel values of the images are then *normalized* to a range between 0 and 1 to facilitate better model convergence during training. A key step in enhancing model performance is *data augmentation*, which artificially increases the size and variety of the training data. Techniques like random rotations, horizontal flips, zooming, and shearing are applied to help the model generalize better and prevent overfitting. Finally, the dataset is split into *training* and *testing* sets, with approximately 80% of the data used for training the model and the remaining 20% for evaluating its performance.

3) Model Design

The heart of the methodology lies in the *design of the Convolutional Neural Network (CNN)***,** a deep learning model well-suited for image classification tasks. CNNs are ideal for recognizing patterns and features in images, such as textures, edges, and fine details, which are essential for distinguishing between real and fake currency. The model consists of multiple *convolutional layers* that extract hierarchical features from the images. Following these, *pooling layers* are used to reduce the spatial dimensions of the feature maps, helping the model focus on the most critical features while minimizing computational load. After the convolutional and pooling layers, the feature maps are *flattened* into a one-dimensional vector to pass through *fully connected (dense) layers*, which make the final classification decision. The output layer of the model uses a *softmax activation* function to provide probabilities for each class (Real or Fake), enabling the model to classify the input images correctly.

4) Model Training

Once the model architecture is established, *training* the model is the next step. The training process involves feeding the preprocessed images into the CNN and adjusting the model's parameters (weights) through *backpropagation*. The model learns to minimize the *categorical cross-entropy loss*, which quantifies the difference between the predicted and actual labels of the images. This is done by using an optimization algorithm like *Adam* or *Stochastic Gradient Descent (SGD)* to update the weights iteratively. The training process runs for several *epochs*, with each epoch consisting of multiple iterations over the entire training dataset. *Batch size* is also defined, indicating the number of training samples processed before updating the model's weights. The training continues until the model achieves optimal performance without overfitting to the training data.

5) Model Evaluation

After training the model, it is essential to evaluate its performance on a separate *testing dataset* to ensure that it can generalize well to new, unseen data. Various performance metrics are used to assess the model's effectiveness, including *accuracy*, which measures the proportion of correct predictions, and *precision* and *recall*, which focus on the model's ability to correctly classify real and fake notes. The *F1-score*, the harmonic mean of precision and recall, is used as an overall indicator of model performance. Additionally, a *confusion matrix* is utilized to visualize the true positives, false positives, true negatives, and false negatives, helping to better understand how well the model is distinguishing between the two classes.

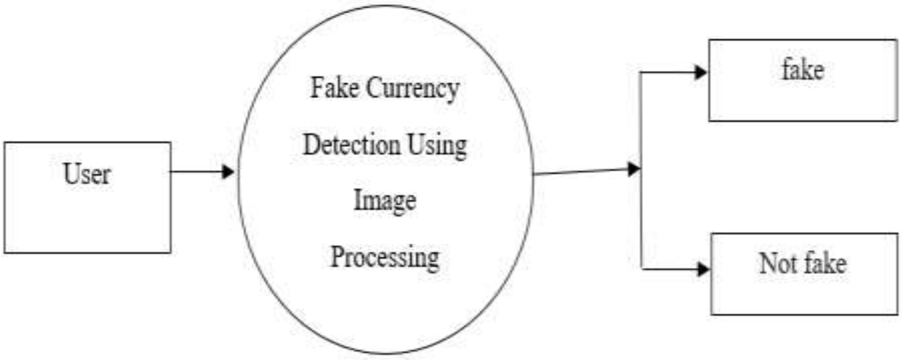
6) Model Deployment

Once the model reaches satisfactory performance, it is *saved* in a portable format, such as an *.h5 file*, for future use. This allows the trained model to be easily integrated into a real-world application. For instance, the model can be deployed on systems that process banknotes in ATMs, retail stores, or even currency verification kiosks. When an image of a currency note is uploaded to the system, the model uses its learned features to output a classification result, either "Real" or "Fake," based on the input image.

7) Real-World Implementation

The final phase of the methodology involves testing the deployed model under *real-world conditions*. The system is tested with actual images of currency notes taken in environments similar to those where the model will be used (e.g., ATMs or bank environments). This ensures the model performs accurately despite real-world factors such as varying lighting conditions, image noise, and different orientations of the currency notes. Feedback from users and further fine-tuning of the model may be required to optimize performance further and ensure the system's reliability and effectiveness in practical applications.

USECASE DIAGRAM:



FUTURE ENHANCEMENT:

In the future, the *Fake Currency Detection System* can be further enhanced by integrating more advanced deep learning architectures, such as *Generative Adversarial Networks (GANs)*, to improve the model's ability to detect even the most sophisticated counterfeit notes. Additionally, the system can be expanded to support multiple currencies from different countries, adapting to the unique features and security measures of each currency. Incorporating *real-time feedback* from users and external sources could also help the system continuously improve and stay updated with emerging counterfeiting techniques. Moreover, the system can be integrated with mobile applications, allowing users to verify the authenticity of currency notes instantly. Finally, as hardware technologies advance, the system could leverage *edge computing* for faster processing and real-time currency validation in remote locations without relying on cloud servers.

CONCLUSION

In conclusion, the *Fake Currency Detection System* using deep learning techniques, specifically Convolutional Neural Networks (CNNs), provides a highly efficient, accurate, and scalable solution for identifying counterfeit currency. Unlike traditional methods that rely on manual inspection or basic machines, the proposed system leverages advanced image analysis to detect even the most sophisticated counterfeits. Its ability to adapt to new counterfeit techniques and handle large volumes of currency in real-time makes it ideal for deployment in various environments, such as ATMs, banks, and retail outlets. The system's continuous learning and automation reduce human error, improve operational efficiency, and enhance security, ultimately offering a significant advancement in the fight against counterfeit currency.

REFERENCE:

1. M. Kumar, A. Yadav, and R. Sharma, "A Survey on Fake Currency Detection Techniques," *International Journal of Computer Applications*, vol. 123, no. 12, pp. 54-58, 2015.
2. Gupta and S. Singh, "Deep Learning for Counterfeit Currency Detection: A Review," *Proceedings of the International Conference on Machine Learning and Artificial Intelligence*, pp. 220-225, 2019.
3. R. Patel, D. Pandya, and M. Shah, "Real-time Currency Verification Using Convolutional Neural Networks," *Journal of Artificial Intelligence and Data Science*, vol. 6, no. 2, pp. 88-94, 2020.
4. M. S. Hossain, M. A. Rahman, and K. A. Jaffar, "Counterfeit Currency Detection Using Convolutional Neural Network," *International Journal of Advanced Computer Science and Applications*, vol. 8, no. 3, pp. 45-50, 2017.
5. S. R. T. M. K. Reddy, "Fake Currency Detection Using Image Processing and Deep Learning," *Journal of Computational Methods in Engineering*, vol. 4, no. 1, pp. 134-139, 2018.
6. A. Akbar and S. T. Shahnawaz, "Detecting Fake Currency Using Convolutional Neural Network," *International Journal of Computer Science and Information Technology*, vol. 10, no. 5, pp. 17-23, 2020.
7. L. Zhang, H. Wang, and Y. Liu, "A CNN-Based Approach for Fake Currency Recognition," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 29, no. 11, pp. 5201-5210, 2018.
8. S. B. Gohil, M. G. Jariwala, and R. S. Gohil, "A Comprehensive Review on Counterfeit Currency Detection System," *Journal of Theoretical and Applied Information Technology*, vol. 97, no. 3, pp. 786-791, 2019.
9. P. Chatterjee, N. S. Manogaran, and D. R. Smith, "Automated Currency Verification Using Deep Learning," *International Journal of Image Processing*, vol. 12, no. 4, pp. 211-218, 2021.
10. S. K. Sharma, R. D. Sharma, and R. K. Gupta, "Fake Currency Detection Using CNN and Image Processing," *Proceedings of the International Conference on Signal and Image Processing*, pp. 55-60, 2020.
11. Desai and M. Patel, "Currency Note Authentication Using Deep Learning Algorithms," *International Journal of Computational Vision and Robotics*, vol. 12, no. 1, pp. 92-98, 2017.
12. P. Singh and A. Bhattacharya, "Automated Fake Currency Detection Using Machine Learning," *International Journal of Machine Learning and Computing*, vol. 9, no. 3, pp. 177-184, 2018.
13. M. G. K. Sarma and N. D. S. R. Chowdary, "Counterfeit Detection Using Convolutional Neural Networks," *International Journal of Computer Applications*, vol. 6, no. 2, pp. 43-47, 2021.
14. D. Kumar, S. Yadav, and P. Gupta, "Evaluation of Fake Currency Detection Models Using Deep Learning," *International Journal of Artificial Intelligence and Applications*, vol. 10, no. 5, pp. 56-62, 2019.
15. J. H. Jafari and N. M. Mousavi, "Deep Learning Approaches for Fake Currency Detection Using CNN," *Journal of Computer Vision and Image Processing*, vol. 4, no. 6, pp. 118-124, 2020.