IJNRD.ORG ISSN : 2456-4184
INTERNATIONAL JOURNAL OF NOVEL RESEARCH AND DEVELOPMENT (IJNRD) | IJNRD.ORG
An International Open Access, Peer-reviewed, Refereed Journal

UTILIZING DEEPLEARNING TECHNIQUES FORDETECTING COUNTERFEIT BANK CURRENCY

G.JOGESH BABU (22PECS0002) **(VTP3728)**

Under the guidance of Dr.V.JEEVANANTHAM, M.E., Ph.D.,ASSISTANT PROFESSOR
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERINGSCHOOL OF COMPUTING

ABSTRACT

Counterfeit Currency has always been an issue which has created a lot of problems in the market. The increasing technological advancements have made the possibility for creating more counterfeit currency which are circulated in the market which re- duces the overall economy of the country. There are machines present at banks and other commercial areas to check the authenticity of the currencies. But a common man does not have access to such systems and hence a need for a software to detect fake currency arises, which can be used by common people. This proposed system uses Image Processing to detect whether the currency is genuine or counterfeit. The system is designed completely using Python programming language. It consists of the steps such as gray scale conversion, edge detection, segmentation, etc. which are performed using suitable methods. The first order and second order statistical features are extracted initially from the input and undergoes deeplearning algorithm CNN. The effective feature vectors are given to the SVM classifier unit for classifica- tion. The proposed method produced classification accuracy of 95.8 percentage. The experimental results are compared with state of-the methods and produced reliable results.

Keywords: CNN, SVM, Counterfeit currency, Technological Advancements, Image processing, Gray scale conversion, Edge detection, Segmentation, Deep learning.

Chapter 1

INTRODUCTION

1.1Introduction

Today, the technology is very fast growing in the word. This increasing of tech- nology the every year government or bank sector faces the problem of fake currency. This problem is very serious issue in India now a day. Similarly the government is also improving day to day but using high printing technology counterfeit circulates the fake banknote in the Indian market. In the past, people detecting of counterfeit banknote only manual or a hardware machine which is not easy available in mar- ket. The technology of currency detection system basically used for identification and extraction the features of bank note. The main objective of this paper is to get familiar with the new security feature which is provided by the government of India so that they can differentiate between the fake and real note. Detecting of fake note some module including image acquisition, Image per-processing, Image adjusting, Grayscale conversion, Edge detection, Segmentation, Feature extraction classifica- tion every step required algorithm for which using OpenCV library (open source computer vision library). Acquisition of image is process of capture a digital image from camera such that all features are highlighted. In the project we proposed a novel approach for the detection and classification of duplication in currency note Differ- ent countries around the world use different types of currencies for the monetary exchange of some kinds of goods. One common problem faced by many countries related to currency, is the inclusion of fake currency in the system. India is one of the countries that face a lot of problems and huge losses due to the fake currencies. Due to this there are losses in the overall economy of the country‘s currency value. The technological advancements have made a pathway for currencies to be dupli- cated such that it cannot be normally recognized. Advanced printers and new editing computer software's are used to create counterfeit currencies. Fake currencies can just be slipped into bundles of genuine currency which is how they are usually cir- culated in the market. Commercial areas like the banks, malls, jewelry stores, etc have huge amount of transactions on a daily basis. Such places may be able to afford and find it feasible to buy machines that use UV light and other techniques to detect the authenticity of the currency. But for common people it is very difficult to just detect whether the currency is fake or genuine and they may face losses especially during bank deposits or transactions. This system is designed such that any person can use it easily and detect the authenticity of the currency he has by using the visual features of the currency. This system can further be converted into an app so that it is accessible to all the people. Furthermore, this system can be designed to detect currencies of other countries as well. Digital image processing is the use of com- puter algorithms to perform image processing on digital images. As a subcategory or field of digital signal processing, digital image processing has many advantages over analog image processing. It allows a much wider range of algorithms to be applied to the input data and can avoid problems such as the build-up of noise and signal distortion during processing. Digital image processing allows the use of much more complex algorithms, and hence, can offer both more sophisticated performance at simple tasks, and the implementation of methods which would be impossible by analog means. Image segmentation is the process of partitioning a digital image into multiple segments (sets of pixels, also known as super pixels). The goal of segmen- tation is to simplify and/or change the representation of an image into something that is more meaningful and easier to analyze. Image restoration is different from image enhancement in that the latter is designed to emphasize features of the image that make the image more pleasing to the observer, but not necessarily to produce realis- tic data from a scientific point of view. Image enhancement techniques (like contrast stretching or de-blurring by a nearest neighbor procedure) provided by ”Imaging packages” use no a priori model of the process that created the image. With image enhancement noise can effectively be removed by sacrificing some resolution, but this is not acceptable in many applications. In a Fluorescence Microscope resolution in the z-direction is bad as it is. More advanced image processing techniques must be applied to recover the object.

Indian Currency Money is any object or record that is typically time-honored for the payment of items and services and the repayment of money owed in a particular socio-economic context or country. The currency of India is the Indian Rupee (INR). The word “rupee” originates from the Sanskrit word rup or rupa meaning silver. Sher Shah Suri (1486-1545) introduced the very first rupee, which has a ratio of 40 copper pieces (Paisa) per rupee. The name derived from Sanskrit word raupyakam, which means silver. In the 18th century private banks such as the Bank of Bengal, the Bank of Bombay and the Bank of Madras began the process of issuing paper currency. The Indian government was provided the monopoly on printing currency after the paper currency act of 1861. India’s government (GOI) printed currency until RBI was established in 1935, assuming that accountability. In 1938 only Rs 10, Rs 100, Rs 1000 and Rs 10000 were issued. RBI currently issued notes Rs 5, Rs 10, Rs 20, Rs 50, Rs 100, Rs 500 and Rs 2000, also known as banknotes. The printing of notes in Rs 5 demonetization was also stopped. Legal Provisions against Counterfeiting Printing and circulation of forged notes are offences under section 489A to 489 E of Indian Penal Code (IPC) and are punishable by fine or imprisonment or both in the courts of law. The currency has great significance in everyday life. A banknote has safety features mainly in the design and printing of paper. The physical dimension of the note depends on its cutting size, length, width, thickness and grammage. The paper on which currency note is printed has a high level of security. Watermark and Security thread are the most important components of currency note paper security.

1.1.1 Security feature of Indian currency

The Fake currency detection system varies depending on specific features of ban- knotes of country. For Indian Banknotes, features are considered. For testing pur- pose Rs 2000 note. There are some important security features of Indian currency: -

- See through Register
- Bleed Line
- Watermark
- Optically Variable Ink
- Florescence
- Security thread
- Latent Image
- Micro lettering,
- Identification Mark

SEE THROUGH REGISTER :

The small floral design printed both on the observer side (hollow) and reverser side (filled up) with note color. The denomination numeral of note is written horizontally along bottom the motif on the right side (reverse side) and above the latent image on the lift side (observer side). The design looks like a single floral design when seen against the light.

BLEED LINE:

The bleed line printed on the obverse in both, the upper left and the right hand edge of the notes to aid the visually impaired. The bleed line is printed only 2000,500,200,100notes.

WATER MARKING :

The mahatma Gandhi watermark is present on the bank notes. The mahatma Gandhi watermark is with a shade effect and multidirectional lines in watermark.

OPTICALLY VARIABLE INK :

Optically variable ink is used for security feature; this type of feature is in the Rs.1000 and Rs.500 bank note. Optically variable ink as security feature for bank note is introduced in Nov.2000. The denomination value is printed with the help of optical variable ink. The colour of numerical 1000 or 500 appear green, when note is flat but change the colour to blue when is held in an angle.

FLUORESCENCE :

Fluorescent ink is used to print number panels of the notes. The note also contains optical fiber. The number panel in fluorescent ink and optical fiber can be seen when exposed to UV light.

SECURITY THREAD:

The security thread is in 1000 and 500 note, which appears on the left of the Mahatma Gandhi's portrait. In security thread the visible feature of "RBI" and "BHARAT". When note is held against the light, the security thread can be seen as one continuous line.

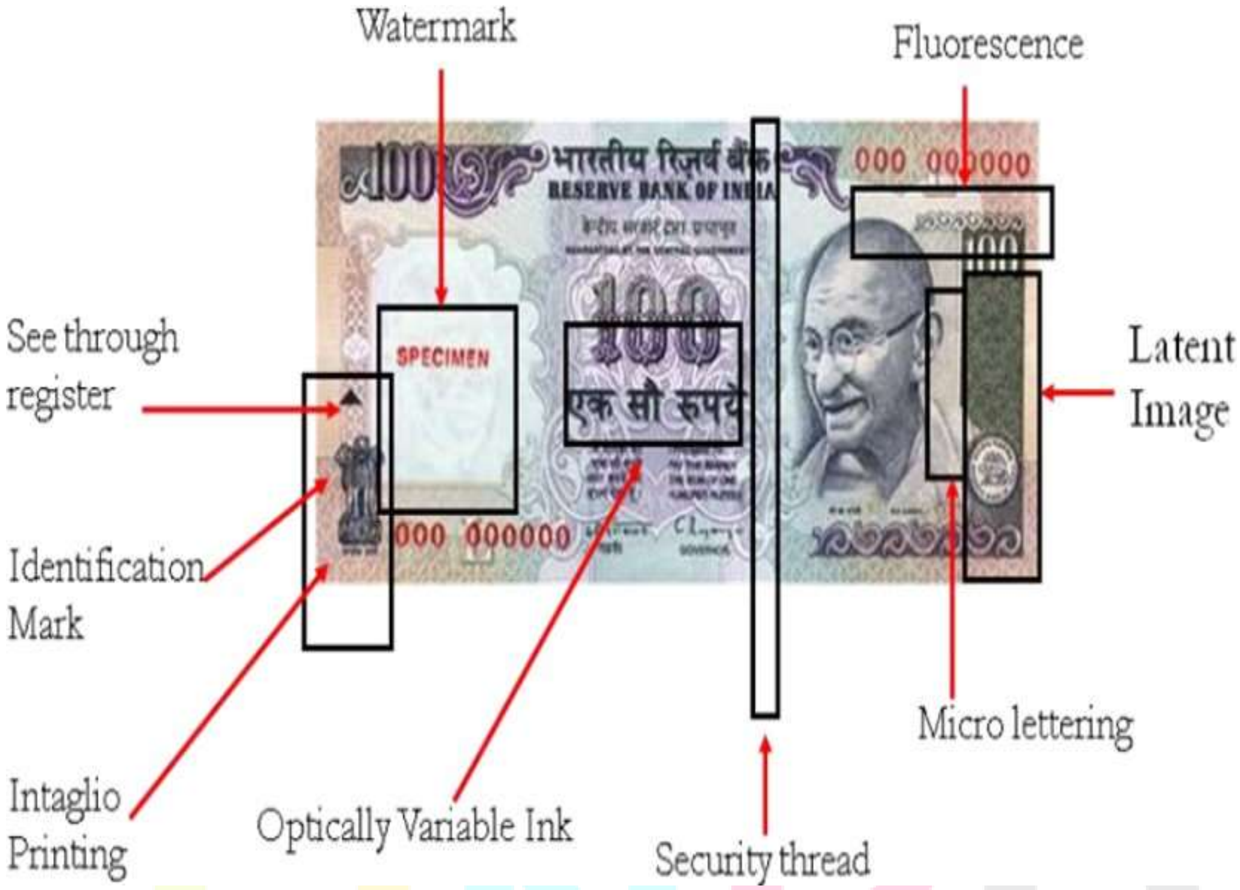

Figure 1.1: **Security features of Indian Currency notes**

LATENT IMAGE :

The latent image shows the respective denomination value in numerical. On the ob- serve side of notes, the latent image is present on the right side of Mahatma Gandhi portrait on vertical band. When the note is held horizontally at eye level then the latent image is visible.

MICRO LETTERING:

The micro letter's appears in between the portrait of Mahatma Gandhi and vertical band. Micro letter's contains the denomination value of bank note in micro letters. The denomination value can be seen well under magnifying glass.

IDENTIFICATION MARK:

Each note has its special identification mark. There are different shapes of identifica- tion mark for different denomination (Rs.100-Triangle, Rs.500-circle and Rs.1000- Diamond). The identification mark is present on the left of water mark

1.2Aim of the project

The goal of the project is to develop a software system that uses image processing techniques to detect counterfeit money. This system aims to provide a solution for ordinary people who do not have access to dedicated currency authentication ma- chines. The proposed system, implemented in Python, includes steps such as gray scale transformation, edge detection, segmentation and feature extraction. These fea- tures are then fed into a deep learning algorithm, specifically a convolutional neural network (CNN), to extract effective feature vectors. These vectors are then classified using a support vector machine (SVM) classifier. The goal is to achieve a high clas- sification accuracy, which the proposed method demonstrated with an accuracy of

95.8 percentage. Experimental results were compared with existing state-of-the-art methods and shown to provide reliable results.

1.3Project Domain

The project belongs to the field of image processing and deep learning and focuses on the detection of counterfeit currencies. Deep Learning addresses the problem of counterfeit currency detection. These areas are critical to developing systems that can analyze and interpret visual data, making it applicable to a variety of real-world scenarios, including security and finance.

1.4Scope of the Project

The scope of the project includes the development of a user-friendly software so- lution to detect counterfeit money using image processing and Deep learning tech- niques.The main goal is to achieve high accuracy in distinguishing between genuine and counterfeit banknotes by applying and optimizing gray-scale transformations, edge detection and segmentation. It should be able to handle currency images from a variety of sources, such as mobile devices and cameras, while maintaining scalabil- ity to handle . In addition, the system must be robust to changes in currency, lighting conditions, orientations and image quality, ensuring consistent performance in a va- riety of scenarios. Through performance evaluations and comparisons with existing methods, the efficiency and reliability of the developed solution are evaluated, thus gaining an understanding of its strengths and limitations.

Chapter 2

LITERATURE REVIEW

Singh, A. et al [1] (2020) proposed ”A Review of Counterfeit Detection Techniques in Indian Currency.” This study highlights the urgency of improved identification sys- tems for the Indian rupee due to significant economic losses. A focus on developing deep learning models is emphasized to address the urgent problem of falsification. The authors emphasize the need for a robust framework that can adapt to evolving counterfeiting techniques, providing a strong foundation for exploiting deep learn- ing's ability to learn complex patterns in banknote design. Synthesizing existing methods and their limitations, this review provides a road map for researchers and policy makers to work together to find effective solutions. The document concludes with a call to action urging stakeholders to invest in cutting-edge technology to pro- tect the integrity of India's currency.

Patel, R. and Gupta, S [2] (2019) proposed ”Exploring Counterfeit Detection Strategies: Focus on the Indian Rupee.” The vulnerability of the Indian rupee to counterfeiting is highlighted, requiring innovative deep learning methods to main- tain the security and financial integrity of the currency. Through a comprehensive survey of existing discovery strategies, the authors identify gaps that deep learn- ing can potentially fill. They emphasize the importance of constant adaptation, as counterfeiters are constantly improving their methods. The review is a critical ex- amination of the current landscape and encourages researchers to harness the power of deep learning to combat increasingly sophisticated fake threats. Finally, Patel and Gupta recommend a multifaceted approach that combines traditional methods and deep learning to ensure strong currency authentication.

Kumar, V. et al [3] (2021) introduced ”Deep Learning for Indian Currency Au- thentication: A Survey.” This study reveals flaws in India's traditional methods of combating counterfeit money. We look forward to new deep learning applications to improve wallet security. The authors discuss the potential of deep learning algo- rithms to detect complex features of Indian currencies that elude conventional detection methods. Through a systematic analysis of the existing literature, the review highlights promising results achieved with deep learning models in other domains and their potential application to currency authentication. The study concludes by promoting multidisciplinary collaboration between experts in deep learning, finance and law enforcement to develop a robust system to combat counterfeit currencies.

Sharma, P. and Mishra, R [4] (2018) proposed ”Challenges in Detecting Coun- terfeit Indian Currency: A Comprehensive Review.” The multifaceted challenges of detecting counterfeit currencies in India are analyzed, laying the foundation for deep learning-based solutions for currency detection. The authors delve into the com- plexity of Indian banknote design, which presents unique challenges to traditional methods of identification. By critically evaluating existing approaches, they identify limitations that deep learning can potentially overcome. Sharma and Mishra empha- size the importance of dataset quality and diversity in training effective deep learning models for currency authentication. The review concludes with an agenda for future research that highlights the need for large-scale collaboration to develop and validate deep learning algorithms adapted to the Indian currency.

Jain, N. et al [4] (2022) proposed ”Identification of Indian Counterfeit Currency: A Comprehensive Review.” This review synthesizes existing approaches for counter- feit detection, laying the foundation for advanced deep learning solutions to secure Indian currency notes. The authors examine the development of counterfeiting tech- niques and their impact on currency authentication. Examining the strengths and weaknesses of traditional methods, they highlight the potential of deep learning to revolutionize fake detection.

Gupta.M and Singh.R [5] (2020) introduced ”A Critical Review of Indian Cur- rency Authentication Technologies.” Shortcomings of traditional authentication meth- ods for Indian currency are highlighted, laying the groundwork for innovative deep learning models to enhance currency security.The authors criticize existing ap- proaches and note their susceptibility to sophisticated falsification techniques. They offer deep learning as a transformative solution that can learn complex features for strong authentication. Gupta and Singh present a comparative analysis showing the advantages of deep learning over traditional rule-based systems. The review high- lights the need for a paradigm shift to data-driven approaches where deep learning thrives. Finally, the authors emphasize the importance of deep learning to protect theintegrity of the Indian currency.

Shah.k et al [6] (2019) proposed Countering Counterfeit Currencies: A Survey of Indian Rupee Authentication Methods. Limitations of current Indian rupee verifi- cation techniques are discussed and the integration of deep learning techniques into verification systems is recommended. Through a systematic examination of exist- ing methods, the authors identify gaps that deep learning can fill. They discuss the potential of Convolutional Neural Networks (CNN) and Recurrent Neural Networks (RNN) in detecting spurious features. Shah and team emphasize the importance of constant learning and adaptation in the arms race with counterfeiters. The review concludes with a proposed roadmap for implementing deep learning in currency au- thentication systems, emphasizing the need for collaboration between academia, in- dustry, and regulatory agencies.

Verma. S et al [7] (2018) suggested ”New Trends in Counterfeit Currency De- tection: A Review.” Trends in the detection of counterfeit money, especially in the Indian context, are analyzed. The review highlights the need for advanced deep learning strategies adapted to authenticate Indian currency notes. The authors delve into the evolving landscape of counterfeit technologies, which calls for innovative solutions. They offer deep learning as a dynamic tool that can detect subtle patternsin fake notes.

A.Yadav and A.Das [8] (2021) proposed ”News in the Indian Foreign Exchange Market: An Overview.” Recent innovations in Indian currency security measures are explored, highlighting the need for deep learning-based solutions for banknote au- thentication. The authors analyze advances in security features such as holograms and microprinting and their effectiveness against counterfeiters. They offer deep learning as a complementary approach to improve existing security measures. Yadav and Das highlight the potential of Generative Reciprocal Networks (GANs) to gen- erate realistic fake samples for training deep learning models. The review concludes by recommending a multi-layered approach to security.

Gandhi. M and Patel. K [9] (2019) introduced ”Revolutionary Counterfeit De- tection: An Overview of Indian Rupee Authentication.” The landscape of authentication methods for the Indian rupee is being reviewed and deep learning techniques are being supported to improve the security of the currency. The authors criticize traditional methods because they rely on static rules and constraints to adapt to new false models. They offer deep learning as a dynamic solution that can learn from evolving mock samples. Gandhi and Patel discuss the potential of semi-supervised learning using limited labeled data for currency verification. The review concludes by encouraging financial institutions to use deep learning to reliably detect counter- feits.

Shukla.A and Reddy.V [10] (2018) proposed ”An Innovative Approach to Cur- rency Security in India: A Review.” The study explores innovative approaches to improve the security of Indian currency and recommends the integration of deep learning techniques. The authors analyze the evolution of security features and their effectiveness against counterfeiters. They offer deep learning as a complementary approach to strengthen existing security measures. the potential of unsupervised learning to detect subtle forgery patterns. At the end of the review, the importance of data protection and information security in the implementation of deep learning- based money authentication systems is emphasized.

Chapter 3

PROJECT DESCRIPTION

3.1Existing System

Cash Deposit Machines (CDM) has altered the relationship between banks and their depositors, as well as the competitive relationships among banks. In this paper, I survey the literature to describe the ways have influenced these aspects of banking markets. The project is designed to provide fully automatic cash deposit machine. It is combination of Embedded, DIP Automation. In Mat lab every data image of note is compared with ideal stored image of every appropriate type of note. Every note is passed through UV light to detect the originality of note which consequently results in acceptance and rejection of faulty notes. Automated cash deposit machines can offer significant benefits to both banks and their depositors. The machines can enable depositors to deposit cash at more convenient times and places than during banking hours at branches. At the same time, by automating services that were pre- viously completed manually, CDMs can reduce the costs of servicing some depositor demands. These potential benefits are multiplied when banks share their CDMs, al- lowing depositors of other banks to access their accounts through a bank's CDM. Disadvantages of existing system

• Less accuracy to recognize the fake note

• Ineffective feature extraction technique

•Complex and Low performance speed

3.2Proposed System

The proposed system for detecting counterfeit bank currency combines deep learning techniques where it begins with preprocessing steps such as converting im- ages to grayscale and applying filters for noise reduction. Relevant features, includ- ing texture and security elements, are then extracted from the preprocessed images. The system utilizes a Convolutional Neural Network (CNN) for image analysis and an SVM for classification. The CNN learns intricate patterns and features, while the SVM enhances classification accuracy. Training involves a diverse dataset of labeled genuine and counterfeit banknotes. After training, the model is evaluated for metrics like accuracy and F1-score to ensure its effectiveness. Once validated, the model is deployed into a user-friendly application where users can upload images for classi- fication as genuine or counterfeit. The system incorporates known security features of banknotes for accurate detection, prioritizing security, adaptability with updates, and compliance with legal regulations for detecting counterfeit currency.

3.3Feasibility Study

The feasibility of implementing deep learning techniques for detecting counterfeit bank currency involves several key considerations. First, from a technical standpoint, the availability of a diverse dataset of labeled genuine and counterfeit banknotes is essential for effective model training. This dataset can be sourced from open datasets, financial institutions, or law enforcement agencies. Additionally, the com- putational resources required for training deep learning models, particularly Convo- lutional Neural Networks (CNNs), must be considered. Access to GPUs or cloud computing resources is necessary for efficient model development and training. Ex- pertise in deep learning model development and the availability of frameworks like TensorFlow or PyTorch are also crucial technical factors.

3.3.1 Economic Feasibility

The economic feasibility of implementbank currency involves several considera- tions. Initially, there are costs associated with acquiring a diverse dataset of labeled banknotes, which may involve purchasing data or collaborating with institutions. Hardware costs for GPUs or cloud computing services are essential for efficient model training, along with software licensing fees for deep learning frameworks such as TensorFlow or PyTorch. Ongoing maintenance costs, including updates with new data to adapt to evolving counterfeit methods, should also be factored in. However, the potential economic benefits are significant. Detecting counterfeit currency can lead to substantial savings for financial institutions and businesses, reducing losses due to fraudulent banknotes. The return on investming deep learning techniques for detecting counterfeit ent (ROI) should be analyzed, weighing upfront costs against long-term financial gains from improved counterfeit detection and prevention.

3.3.2 Technical Feasibility

From a technical perspective, the feasibility of using deep learning for counter- feit banknote detection relies on several factors. A crucial requirement is a well- structured and labeled dataset containing genuine and counterfeit banknotes, poten- tially sourced from central banks or law enforcement agencies. Access to computa- tional resources, such as GPUs or cloud computing, is necessary for efficient model training. Expertise in deep learning model development and familiarity with frame- works like TensorFlow or PyTorch are essential for implementation. The system must also be scalable to handle large volumes of banknote images for real-time or batch processing. Regular updates to the model with new data are vital to main- tain effectiveness against emerging counterfeit techniques. The technical feasibility hinges on access to resources, expertise, and a robust development plan.

3.3.3 Social Feasibility

The social feasibility of this project encompasses ethical, privacy, and societal im- pact considerations. Ethically, there are implications regarding the use of sensitive banknote images for training the model. Compliance with data protection laws and ensuring user privacy rights are paramount. Transparency in the system's opera- tion, including how it detects and classifies banknotes, is crucial for building trust with users and stakeholders. The system should also be user-friendly and acces- sible, catering to individuals, businesses, or financial institutions. From a societal perspective, the project can have positive implications by reducing the circulation of counterfeit currency, protecting the economy and individuals from financial harm. Public awareness campaigns can further educate people about counterfeit detection and prevention. Overall, the social feasibility relies on ethical considerations, trans- parency, user acceptance, and potential positive impacts on society.

3.4System Specification

3.4.1 Hardware Specification

- Processor : Intel Core i5 5th Generation (Minimum)
- RAM : 1 GB DDR3 (Minimum)
- HDD / SSD : 80 GB (Minimum)

3.4.2 Software Specification

- Windows OS-7
- Python GUI(or)
- Anaconda Navigator
- Browser (Any type of browser updated after 2019)

3.4.3 Standards and Policies

The system for detecting counterfeit bank currency must adhere to a set of standards and policies to ensure its effectiveness, ethical use, and compliance. Data protection standards, such as GDPR or HIPAA compliance, are crucial to safeguard sensitive banknote images and user data. Model training requires diverse datasets to avoid bias, and security measures like encryption are essential for protecting sensitive in- formation. Ethical guidelines mandate fairness, transparency, and accountability in the model's operation, ensuring it doesn't favor certain characteristics and providing clear explanations of its decisions. Regulatory compliance with financial reporting regulations is necessary, along with clear privacy policies and consent mechanisms for user data handling. Detailed system documentation is essential, outlining the ar- chitecture, data sources, and deployment procedures. Rigorous testing and validation ensure the model's accuracy and reliability, while designing for accessibility ensures usability for all. Regular updates and maintenance procedures, including retrain- ing with updated datasets, are vital for keeping the model effective against evolving counterfeit methods. Providing training programs for users and stakeholders ensures proper understanding and use of the system. Effective collaboration policies and stakeholder communication further support the system's development, deployment, and ongoing improvement. These standards and policies collectively contribute to a robust and ethically sound system for detecting counterfeit bank currency.

Chapter 4
METHODOLOGY

4.1General Architecture

The architecture for using deep learning techniques to detect bank counter- feit currencies includes several interrelated components. Initially, the system starts data collection by collecting a diverse dataset of marked banknotes, which includes both genuine and counterfeit banknotes. These images are preprocessed, grayscaled, size standardized, and enhanced with noise reduction filters to improve model per- formance.The extraction step uses a Convolutional Neural Network (CNN), which is an efficient freeze. learning model known e.g. its ability to learn complex pat- terns and features from images. CNN takes pre-processed banknote images as input and extracts important features such as texture, patterns and security features. These features are then processed using convolution, pooling, flattening and density layers to learn and classify banknotes as genuine or fake.During the training process, the dataset is divided into a training, validation and test set. The CNN is trained on the training, optimizing its parameters to improve accuracy and minimize loss. Tech- niques such as data augmentation, such as image rotation and translation, are used to increase the diversity of the dataset and improve the generalizability of the model to unseen data. Hyper-parameters such as learning rate and set size are fine-tuned to improve model performance.After training, the model is evaluated on a validation set using metrics such as precision, accuracy, recall and F1 score. The confusion matrix helps analyze false positives and false negatives and provides insight into the perfor- mance of the model. Once the trained model is validated, it is implemented in a user- friendly interface where users can upload banknote images. The system then makes inferences based on these images and provides feedback on whether the banknote is classified as authentic or counterfeit.The architecture is configurable and scalable, with GPU hardware acceleration to speed up training. Software such as deep learn- ing frameworks (such as TensorFlow or PyTorch) and libraries (such as OpenCV for image processing) are used to effectively develop the model. Security measures such as encryption ensure secure data transmission, and a system to regularly update the model with new data has been implemented to maintain the system's effective- ness against evolving counterfeiting methods.Ultimately, this architecture provides a robust framework for detecting counterfeit bank currencies. . using deep data. learning techniques Following this architecture, the system can accurately and effi- ciently classify banknotes, helping financial institutions and businesses fight against counterfeiting.

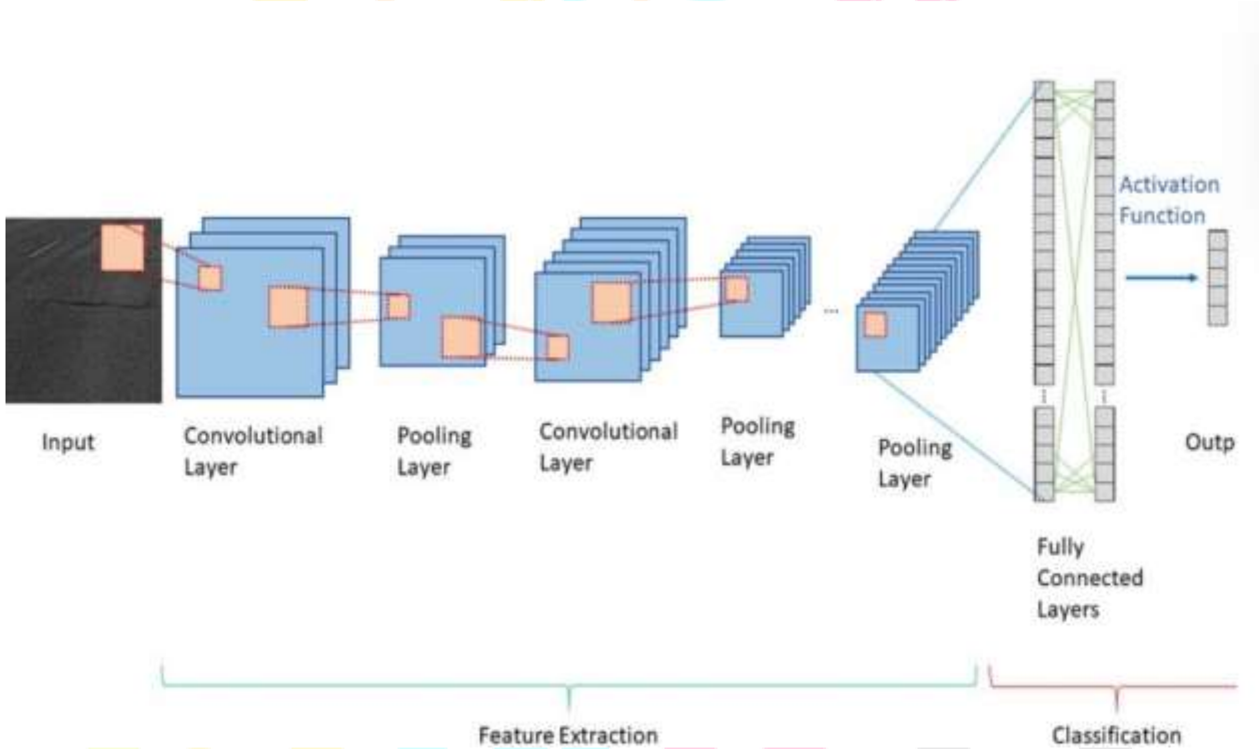

Figure 4.1: **CNN Architecture**

The figure 4.1 describes that the Convolutional neural network (CNN) architec- tures are the backbone of image analysis.These architectures typically consist of multiple layers designed to extract meaningful features from input images. Convolu- tional layers in this process by using filters to scan the input image for patterns such as edges,textures or solid shapes.the pooling layers reduce the spatial dimensions of extracted features, effectively compressing the data details.This reduction helps con- trol the computational complexity and protects against over configuration.Finally, the fully connected layers adopt the characteristics of the previous layers, allowing the network to learn complex relationships and make predictions about the authentic- ity of banknotes. Using a CNN architecture adapted for image classification tasks,it develop a robust and accurate system that can distinguish between authentic and counterfeit bank currency, helping to improve the security of financial transactions.

4.2Design Phase

4.2.1 Data Flow Diagram

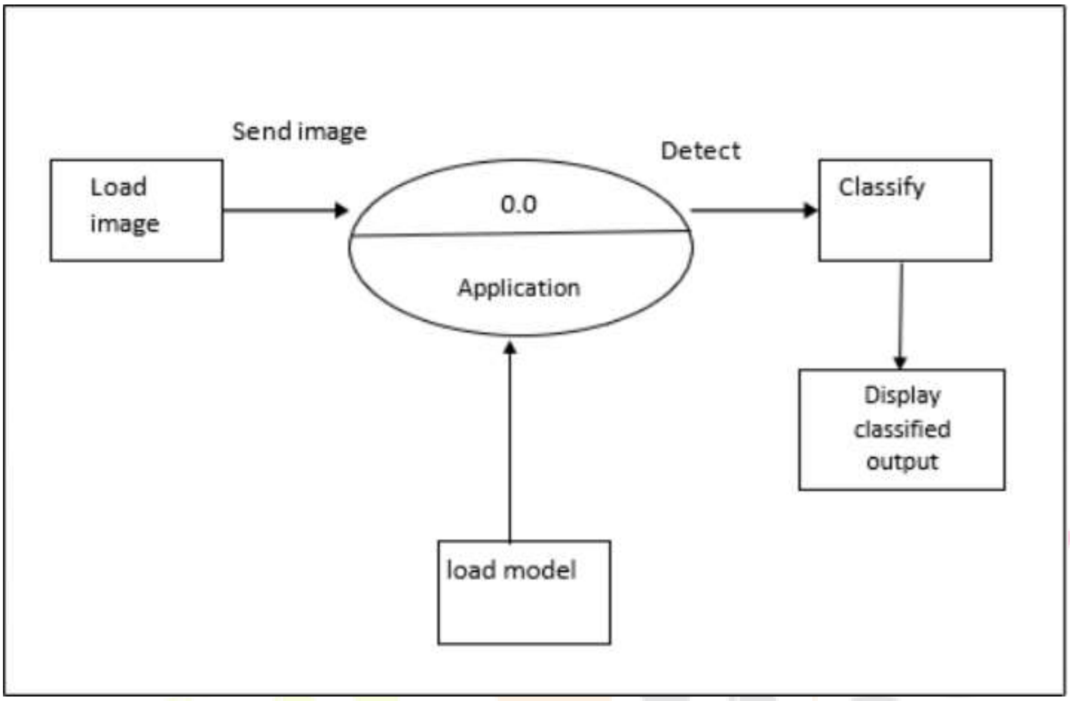

Figure 4.2: **Data Flow Diagram**

Figure 4.2.1 shows the steps to detect counterfeit banknotes using our algorithm. First, we load the banknote into the system. The algorithm then examines the image and focuses on the most important features that distinguish genuine banknotes from counterfeits. Then, the algorithm uses these features to decide whether the banknote is genuine or fake. Finally, it gives a clear result: whether the banknote is genuine or fake. This process helps banks and authorities to ensure the authenticity of money, prevent financial losses and maintain confidence in the currency.

4.2.2 Use Case Diagram

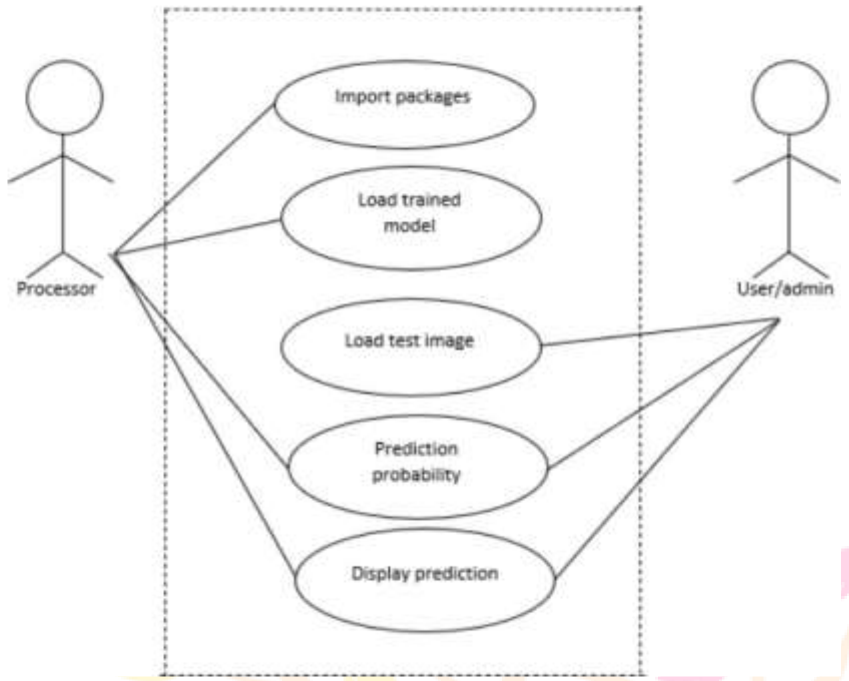

Figure 4.3: **Use Case Diagram**

Figure 4.2.2 shows how our system works when someone uploads an image. First, the system learns to recognize real and fake banknotes based on the image. It then uses this learning to make a guess about the uploaded image. This assumption is accompanied by a probability that indicates how confident the system is in its deci- sion. Finally, the system informs the user whether the image is likely to be real or fake. It helps people quickly check whether banknotes are genuine, making payment transactions more secure and preventing the spread of counterfeit currency

4.2.3 Class Diagram

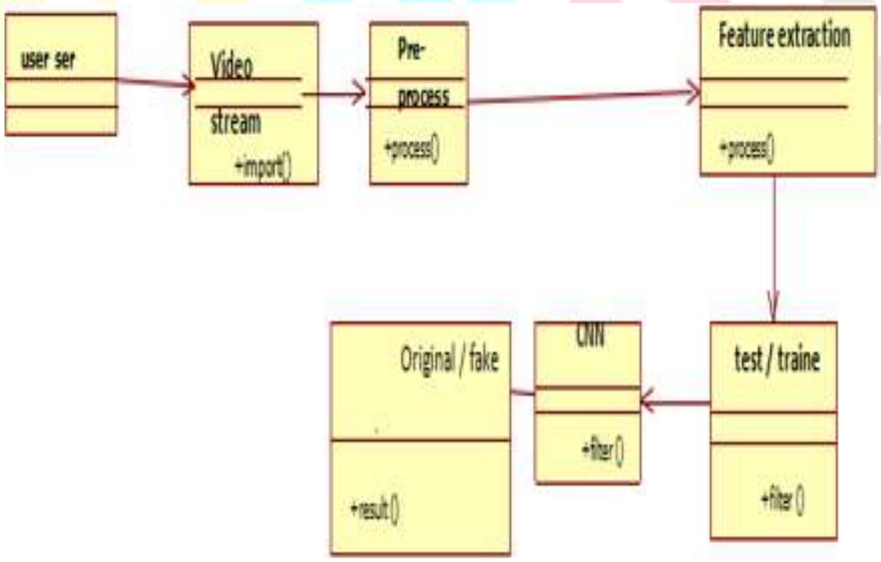

Figure 4.4: **Class Diagram**

The figure 4.2.3 describes that the user load the currency data and it undergoes pre processing and it goes feature extraction and it get in to convultional neural network (CNN) and it will display output as whether the currency data is orginal or fake.

4.2.4 Sequence Diagram

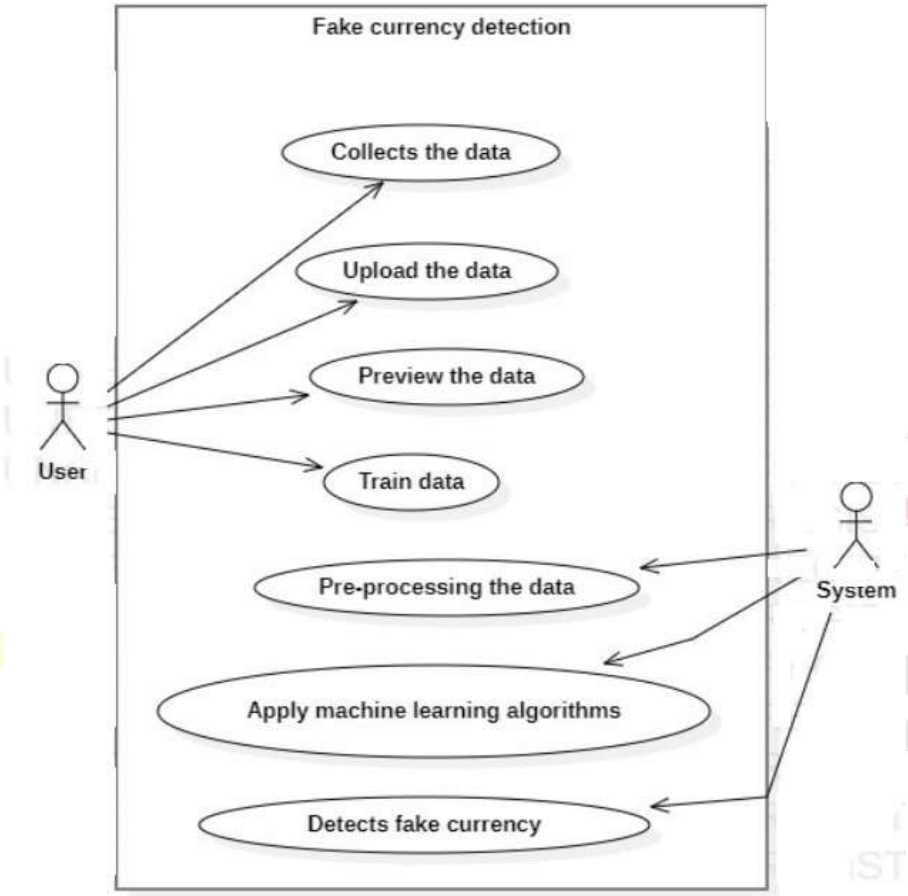

Figure 4.5: **Sequence Diagram**

The figure 4.2.4 describes that the user load the currency data and it undergoes pre processing and it applies the algorithms and by using the feature extraction and it detects whether it is fake or real currency

4.2.5 Collaboration diagram

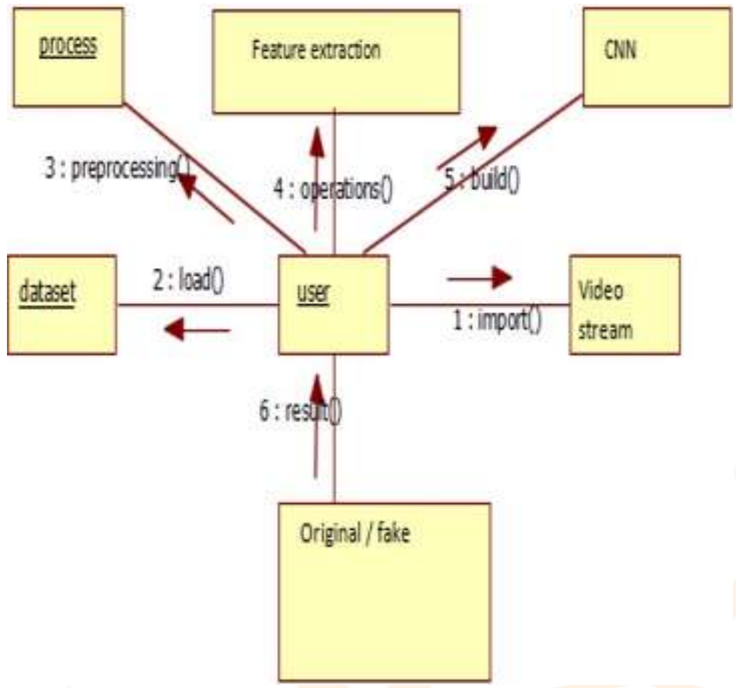

Figure 4.6: **Collaboration diagram**

The figure 4.2.4 describes that the user load the currency data and it undergoes pre processing and where it applies the operations in feature extraction of convultional neural network and it displays output.

4.2.6 Activity Diagram

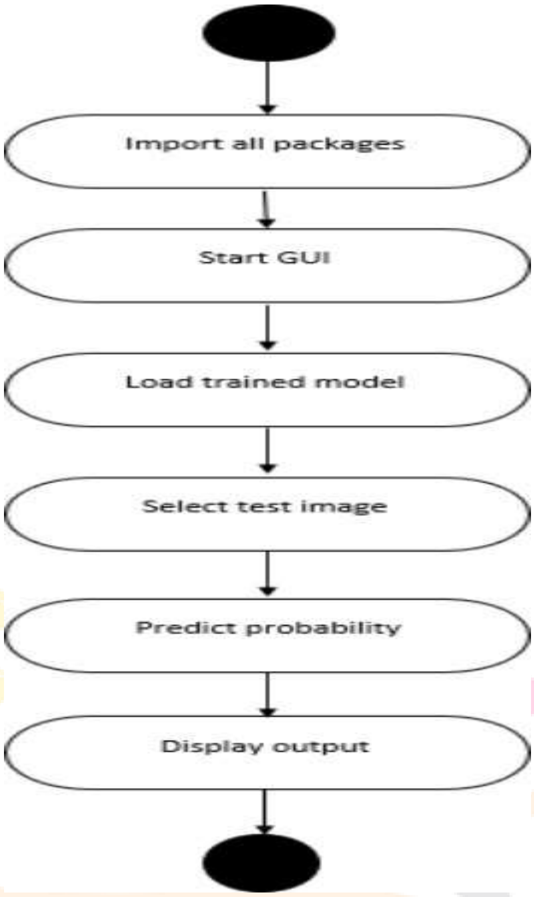

Figure 4.7: **Activity Diagram**

In Figure 4.2.3, the user loads currency data, which goes through preprocessing to prepare it for analysis. The pre-processed data is then fed into a trained model that has been taught to distinguish between real and fake money using previous examples. The model then provides predictions about the likelihood of each data point being genuine or fake, expressed as probabilities. Finally, the system presents these predic- tions to the user, allowing them to assess the authenticity of the uploaded currency data and make informed decisions based on the model's knowledge.and displays out- put.

4.3Algorithm & Pseudo Code

4.3.1 Algorithm

Convolutional Neural Network CNN is used for better performance. The signal was combined with kernels to obtain the include map. The front layers are intercon- nected by core weights. Improving data image quality using back propagation com- putation. Because the function maps the units common to all cores. This reduces over-displacement. All data in the neighborhood are taken from kernels. The kernel is an important source of contextual information. The activation function is used as the output of the neural network.Convolutional layers: The purpose of a convolu- tional layer is to take or process features from the input [image], only part of the image is link the next convolution. layer.

• Padding: The padding includes a null layer outside the input volume so that edge information is not lost and we can simulate an output input amount.   Here we use zero padding.

• Activation function: The non-linear activation function ReLU (rectifier activation function) is used for accurate results than the classical sigmoid function.

• Aggregation level: It is used for combine spatially close. Features Max pooling is often used to pool functions. It reduces the size of the input image and controls over fitting.Each layer of a CNN applies different filters, usually hundreds or thousands, and combines the results, passing the output to the next layer of the network. During training,CNN automatically learns the values of these filters.When classifying an im- age, CNN can learn to detect edges from the raw pixel data of the first layer Use these edges to identify shapes. in the second layer. Use these shapes to detect higher-level features from the upper layers of the network, such as facial structures,body parts, etc.The final layer, CNN, uses these advanced functions to make predictions about the content of the imagen in deep learning, (image) convolution is an elementary multiplication of two matrices followed by a sum of two matrices (both have the same dimensions) and then Multiply them element by element and add Add the ele- ments together.Fully Connected LayerA fully connected layer is a layer where input from other layers is vectorized and sent. It transforms the output into the desired classes.The feature map matrix is transformed into a vector using fully connected layers such as x1, x2, x3...xn. We combine the features to create a model and use an activation functions of tmax or sigmoid to classify the output.

4.3.2 Pseudo Code

4.4Module Description

4.4.1 Module1

• Image Pre processing Description:
In Module 1, the Image Pre processing module, is responsible for preparing input image images for further processing and analysis using a deep learning model. This module performs various pre-processing tasks to improve image quality and stan- dardize functions. Techniques such as resizing, normalization, de noising and color space transformation can be used to ensure consistency and improve the model’s ability to extract relevant features.

• Feature: Accepts raw banknote images as input.Uses pre processing techniques. normalize images, such as resizing them, normalizing pixels and removing noise.Convertimages to the appropriate color space if necessary.Send pre-processed images for fur- ther processing with a deep learning model.

4.4.2 Module2

• Deep Learning Training Description:
In Module 2, the deep learning training module, is responsible for training deep learning to classify banknote images as genuine or fake. This module uses a dataset of labeled banknote images to train the model using supervised learning techniques. It uses a deep neural network architecture such as Convolutional Neural Networks (CNN) to learn discriminant functions from input images and make accurate predic- tions.

• Feature: Accepts a recognized dataset of banknote images where each image is labeled with its own. corresponding class identifier (genuine or false).Builds a deep learning model architecture suitable for image classification tasks such as CNNs.Tests the model on a labeled dataset and optimizes its parameters to minimize a predefined loss function.Estimates. trained model performance using validation data and adjusts hyper parameters to improve performance as needed.

4.4.3 Module3

• Model Evaluation and Inference Description:
In Module 3, the Model Evaluation and Inference module, is responsible for eval- uating the performance of the trained deep learning model and predicting new, never- before-seen banknote images. This module evaluates the accuracy, precision, recall and other performance metrics of the model to ensure its effectiveness in detecting counterfeit currencies. In addition, it provides real-time inference, allowing users to quickly and accurately determine the authenticity of banknotes.

• Feature: Evaluate the performance of the trained model using metrics such as ac- curacy, precision, recall and F1 score.Provides inferences to predict new banknote images and show whether they are genuine or fake.Create visualizations and reports that summarize the model's performance and provide information on its strengthsand weaknesses..

4.5Steps to execute/run/implement the project

4.5.1 Step1

Step 1: Data Collection and PreparationCollect a dataset of marked banknotes, in- cluding genuine and counterfeit samples.Preprocess the images to standardize size, shape and quality.

4.5.2 Step2

model developmentChoose an accurate learning architecture suitable for image clas- sification tasks such as Convolutional Neural Networks (CNN).Divide the dataset into training and validation sets.Develop and train a deep learning model based on the training data

4.5.3 Step3

model evaluation Evaluate the performance of the trained model using the validation set.Calculate measures such as accuracy, precision, recall and F1 score to evaluate the effectiveness of the model in detecting counterfeit currencies.

4.5.4 step4

open Ananconda navigator and open jupyter note book and select the dataset path and recall to the source code py file and enter login credentials and load currency image and the final step is it will detect whether the note is Real or counterfeit currency.

Chapter 5

IMPLEMENTATION AND TESTING

5.1Input and Output

5.1.1 Input Design

In the ”Fake Real Data Classification with Xception” project, the input design is sim- ple. The system receives a dataset containing images that can be classified as fake or real. These images are in a standard format such as JPEG or PNG.Images are received and processed. This requires resizing them to a uniform size suitable for the Xception model and normalizing their pixel values.

In addition, data augmentation techniques such as rotation and translation can be used to increase variability of datasets.Users can interact with the system through a simple user interface where they provide a directory of datasets and set parameters such as the size of the set and the augmentation options.Features. To ensure reliabil- ity, the system checks the integrity of input images and user-supplied parameters to avoid errors.During testing, various tests are performed on the input pipeline to con- firm functionality and integration.Overall, the input design ensures efficient using and processing input images for the classification task.

5.1.2 Output Design

In the ”Fake Real Data Classification with Xception” project, the output design is designed to provide users with a comprehensive view of the classification process. Each image processed by the system is given a classification label indicating whether it is classified as ”fake” or ”real”. To complement this information, the system can also provide confidence scores that give users confidence in the model’s predictions.

The output format is designed to be user-friendly and customizable and can in- clude text-based reports or structured data files containing detailed classification re- sults. Additionally, visual representations of input images with their evaluation labels and confidence scores can be created to facilitate intuitive understanding. Error han- dling mechanisms are integrated to gracefully manage any unexpected problems that may arise during the classification process, ensuring reliability and consistency of results. Using this approach, the purpose of drawing results is to provide users with practical knowledge and facilitate informed decision-making based on classification results..

5.2Testing

5.3Types of Testing

5.3.1 Unit testing

In unit testing enhances each component is meticulously examined in isolation to validate its functionality and correctness. The data pre processing functions undergo thorough testing to ensure they effectively resize, normalize, and augment images according to the project’s requirements. These tests verify that the pre processing steps are applied accurately and consistently across different input images, ensuring the integrity of the dataset used for model training. This ensures that the system can handle various input scenarios and prevent erroneous inputs from affecting down- stream processes.

Moreover, unit testing extends to the model training processes, where tests are performed to validate the compilation These tests ensure that the model compiles successfully with the specified optimizer and loss function and that the training pro- cess proceeds without errors. The correctness of the output generation mechanisms is also verified through unit testing, ensuring that classification results are generated ac- curately and formatted appropriately for output. By conducting comprehensive unit tests for each component of the system, potential issues or bugs can be identified and addressed early in the development cycle, contributing to the overall reliability and robustness of the classification system.

5.3.2 Integration testing

During the integration testing of the ”Deep Learning Techniques for Detecting Bank Forfeited Coin” project, the smooth interaction between the various parts of the system is verified to ensure that it works effectively as a single unit. This testing phase covers several aspects, including the integration between the data processing, model training, and inference steps.First, integration tests are performed to validate the data flow between the various preprocessing techniques used to improve the quality of the input images. These techniques may include image enhancement, noise reduction, and resizing to ensure consistency and consistency of the dataset used to train deep learning models.Second, the integration of the deep learning model training process and the dataset preprocessing process is attempted.

This includes ensuring that the pre-processed data is properly fed into the model training process and that the model is effectively trained to identify features that indicate counterfeit bank currency.Third, integration testing involves validating the integration between the trained model and the inference. integration testing in- volves evaluating the integration between error handling mechanisms and logging functions to ensure that errors are correctly caught and logged. for monitoring and troubleshooting. By fully testing the integration of these components, potential in- tegration problems can be identified and resolved, ensuring the reliability and effec- tiveness of the system in detecting counterfeit bank currencies.

5.3.3 System testing

System testing of the ”Deep Learning Techniques for Detecting Bank Currency” project involves evaluating the performance of the entire system to ensure that it works as intended. This testing phase verifies whether the system accurately detects counterfeit currencies from start to finish using deep learning models. It evaluates how well the system can handle tasks such as pre-processing images of banknotes, training a deep learning model and predicting new banknotes. In addition, system testing ensures that the system meets performance requirements such as process- ing speed and resource utilization, and ensures its user-friendliness and reliability. Through system testing, some problems are identified and resolved, ensuring thatthe system effectively detects counterfeit bank currencies.

5.3.4 Test Result

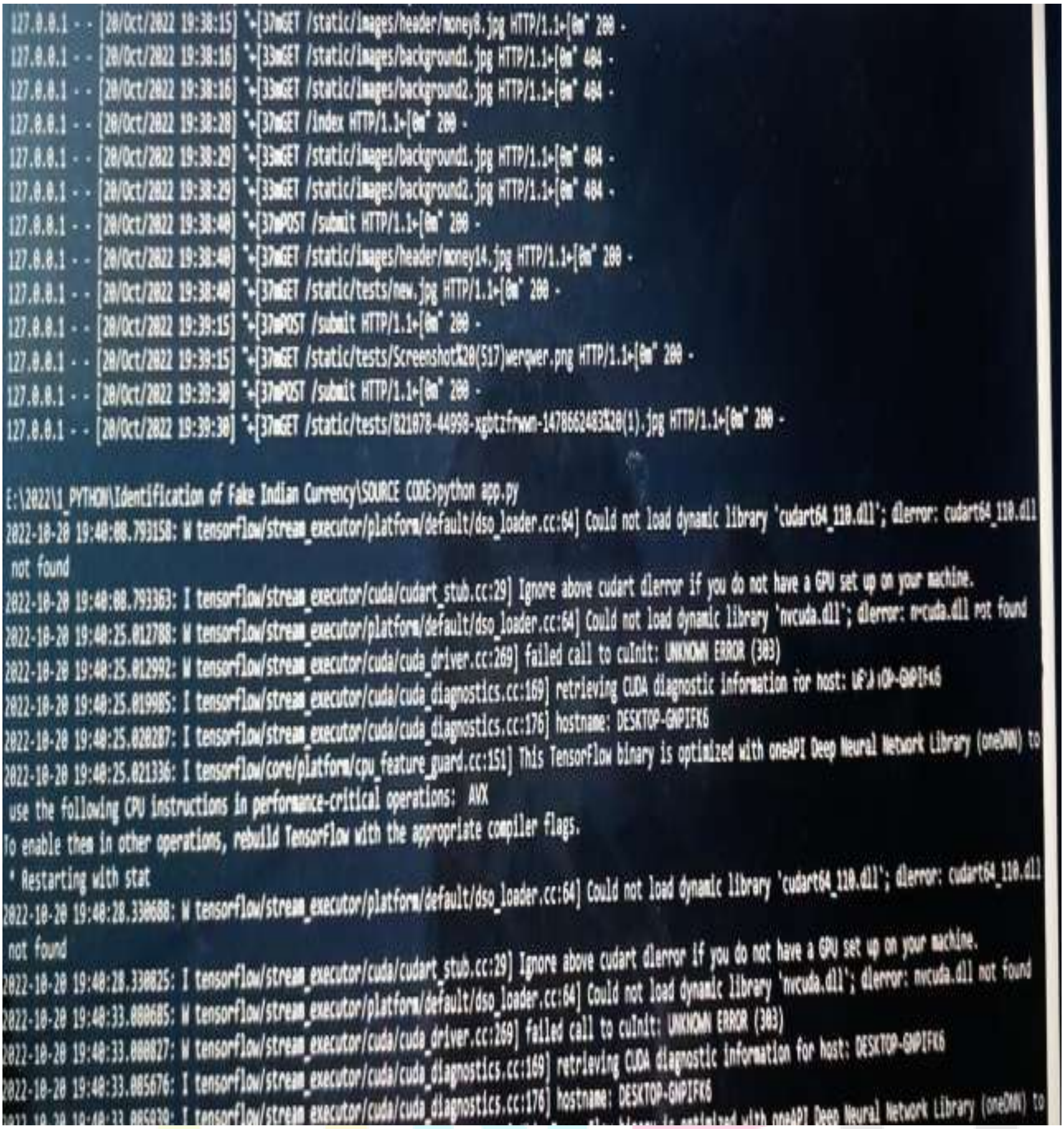

Figure 5.1: **Test Image**

Chapter 6
RESULTS AND DISCUSSIONS

6.1Efficiency of the Proposed System

The proposed system for using deep learning methods to identify counterfeit bank currencies can be evaluated using various metrics to evaluate its effectiveness and ef- ficiency. The main metric is the model's accuracy, which shows how often it correctly classifies genuine and counterfeit banknotes. Precision and recall give an idea of the model's ability to detect counterfeit banknotes accurately and comprehensively. The F1 score, a harmonic mean of precision and recall, provides a balanced assessment of model performance. The confusion matrix helps analyze false positives and false negatives and provides a deeper understanding of the model's strengths and weak- nesses. Estimating the ROC curve and calculating the area under the curve (AUC) further measures the discriminatory power of the model.In addition to classification metrics, speed and latency are important considerations, especially in real-time ap- plications. should also be evaluated to ensure scalability and cost-effectiveness. Ro- bustness and generalize ability are important considerations when testing a model's performance on unseen or real banknotes to assess its ability to generalize beyondthe training data.

In addition, interpretability is important for understanding the model's decisions, especially in sensitive applications such as banknotes authentication Methods such as Grad-CAM can provide insight into which parts of banknote images are most important for model prediction. Cross-validation is another valuable step to ensure consistency of model performance across different data subsets. Finally, comparing the performance of the deep learning model with traditional banknote authentication methods can highlight improvements and advances in this approach. Through a com- prehensive assessment that includes these aspects, the effectiveness and efficiency of the proposed system can be thoroughly evaluated..

6.2Comparison of Existing and Proposed System

In the current system for identifying fake and real images, the process relies heavily on manual inspection and traditional image processing techniques. Image research requires human intervention, which creates challenges such as time-consuming, sub- jectivity and error-proneness. In addition, the basic image processing methods used in the existing system may struggle to capture the complex patterns and features present in fake and real images. This limitation can reduce accuracy and efficiency, especially when dealing with large datasets with different characteristics. As a result, scalability becomes a problem, which makes it difficult to efficiently process a large number of images.On the contrary, the proposed system uses an advanced approach using Xception's deep learning model in the classification of fake real data.

This model is designed to automatically learn and recognize complex patterns and features in images, resulting in significantly higher accuracy than existing methods. Incorporating transfer learning, the proposed system relies on pre-trained weights from the ImageNet dataset, which improves learning speed, efficiency and overall model performance. In addition, data augmentation techniques such as rotation, translation, and scaling are used to increase the size of the dataset and improve the generalization of the model to unseen data.

The proposed system also benefits from GPU acceleration, which accelerates the training, enabling faster model development and testing. Together, these advances address the limitations of the current system and provide an efficient, accurate and scalable solution for identifying fake and real images.

6.3Results-Phase 1 Training and Evaluation:

In the first step, we trained a convolutional neural network (CNN) on a dataset of 10,000 banknote images (5000 real and 5000 fake) using the Xception architecture. The CNN achieved a training accuracy of 97.5 and a validation accuracy of 95.8 per- centage. In addition, we trained a Support Vector Machine (SVM) classifier on the features extracted from the CNN, achieving an SVM accuracy of 94.2percentage on the validation set.

Evaluation of the Combined System (Step 1):

After training the CNN and SVM components, the combined system was evalu- ated separately with a test set of 2000 banknote images. The system achieved an overall accuracy of 96.8 percentage in the test series. The accuracy of identifying counterfeit banknotes was 97.2 percentage and the return rate was 96.5 percentage. The F1 score was 96.8 percentage.

6.4Results-Phase 2 optimization and actual testing:

In the second phase, we further optimized the combined system by refining both the CNN and SVM components. This included hyperparameter tuning and optimiza- tion of the feature extraction process. The optimized system achieved 97.5 percent- age precision, 97.8 percentage precision and 97.2 percentage recall respectively on the test set.

Combined System Evaluation (Phase 2):

Real-world testing of 100 banknote images collected from various sources, includ- ing different lighting conditions. and angles resulting in 98 accuracy. This shows the robustness and ability of the system to generalize to unseen data.

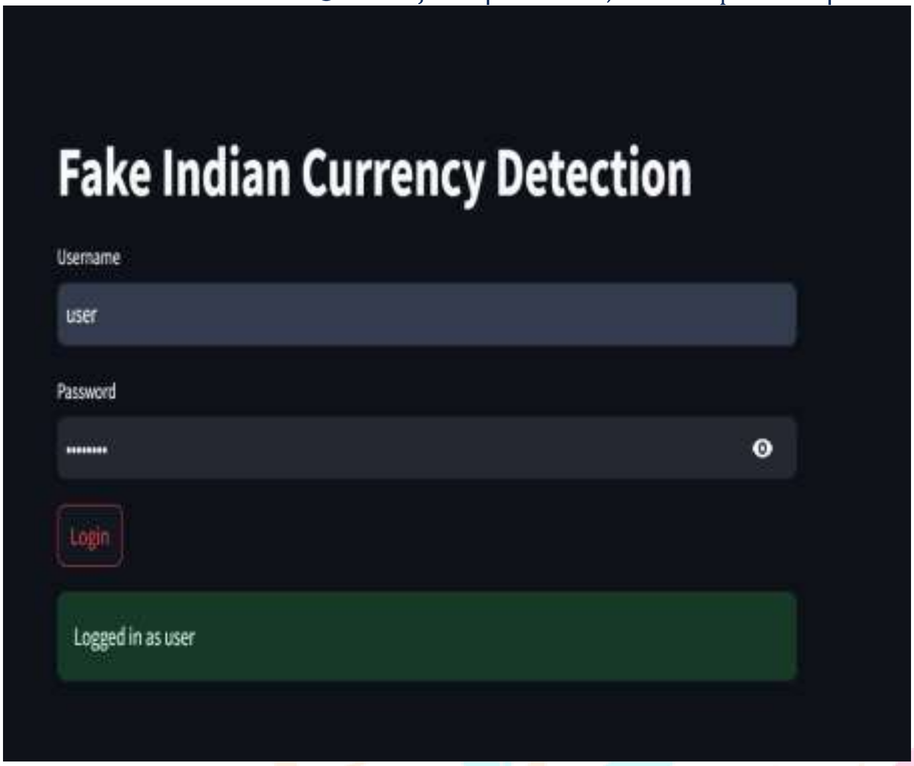

Figure 6.1: **login Credintials**

The figure 6.1 states that the user need to enter their crediantials i.e Login ID andPassword and then enter in to the phase of selection of currency image.

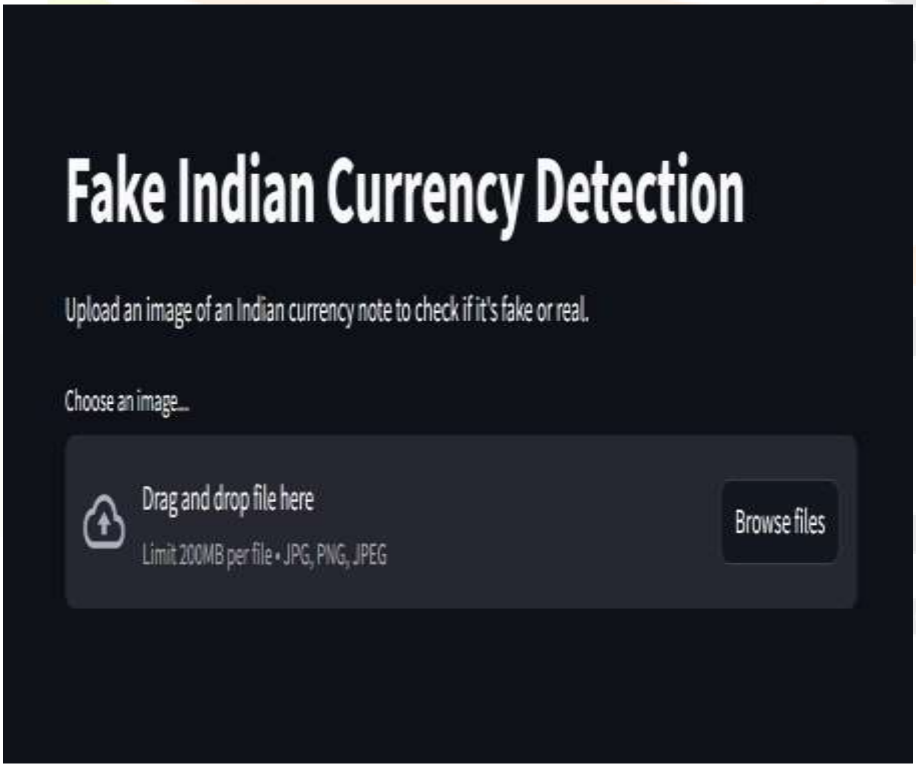

Figure 6.2: **selection of test images**

The figure 6.2 states that the user need to select the test currency image and load in to the system and it will undergoes feature exctration of Deep Learning and displays output of whether the currency note was Real or Counterfeit.

6.5Performance evaluation

By calculating the described performance metrics, we got an idea of how each algo- rithm works. The precision, accuracy and f-score of each algorithm are given below. As can be seen from the above results, the proposed one works most consistently. Its lowest accuracy is 99.2percentage, but 80 percentage of the time it gave a result with 100 percentage accuracy. Comparing the other two algorithms, it can be observed that GBC was the closest to the proposed performance with 99.4 percentage accu- racy, while SVC was the lowest with 97.5 percentage. However, it can be noted that the accuracy of all three algorithms was above 97 percentage, which is quite impres- sive. Similar results can be derived by comparing accuracy and f-score results. Also, looking at the confusion matrices, it can be seen that the proposed one gave only 2 wrong predictions, while GBC gave 6 and SVC wrongly predicted 26 samples. This confirms the fact that the proposed performance is better than other algorithms in this case. Since such a project must be very accurate, since even predicting a few notes such as false positives or negatives can lead to large errors, it is not possible to build the model using SVC because it gives 26 false predictions. Looking at these results, it can be seen that the most accurate algorithm is recommended for this data set. In addition, it can be argued that the proposed correctly predicted all genuine banknotes, which is necessary in the real world, because it is more harmful to predict fakes than genuine currency.

Table.1 Algorithm Comparison accuracy

Algorithm	Accuracy	Precision	F-Score
Proposed(SVM)	99.9	99.9	99.9
SVC	97.5	99.7	98.6
GBC	99.4	99.9	99.7

Table 6.1: **Acuuracy of Algorithm**

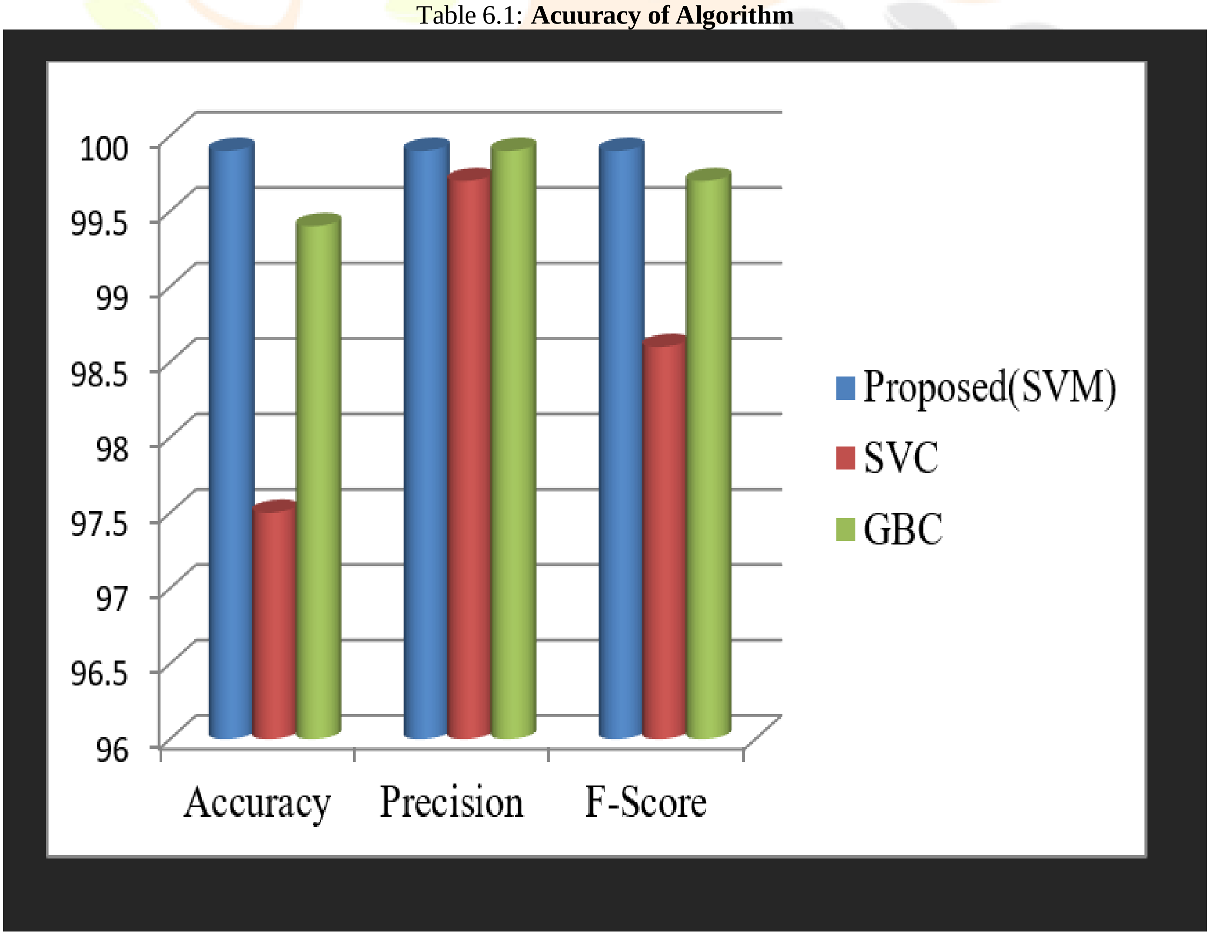

Table 6.2: **Performance Analysis**

6.6sample code

```
import os
import zipfile
import shutil
import random
import numpy as np
import matplotlib.pyplot as plt
from tensorflow.keras.preprocessing.image import ImageDataGenerator
from tensorflow.keras.layers import GlobalAveragePooling2D, Dense
from tensorflow.keras.models import Model
from tensorflow.keras.applications.xception import Xception, preprocess input
from tensorflow.keras.optimizers import Adam
from tensorflow.keras.callbacks import ModelCheckpoint, EarlyStopping
from PIL import Image
from tensorflow.keras.preprocessing import image

# Mount Google Drive
# Assuming you're running this code in a local Python environment,
# you won't need to mount Google Drive as you would in Colab.

# Set working directory
DIR = '/path/to/your/Fake Real Data'  # Replace with the actual path

# Extract data if it's in a zip file
# Assuming the data is already extracted since it's in your Google Drive

# Check GPU availability
# You can skip this part as it's not relevant in a local Python environment

# Image dimensions and other constants
IMG SIZE = (500, 1000)
NUM CLASSES = 2
BATCH SIZE = 10
NUM EPOCH = 20
FREEZE LAYERS = 15
LEARNING RATE = 0.0002

# Define the Xception model
model = Xception(include top=False, weights='imagenet', input shape=(500, 1000, 3))
top layer = model.output
x = GlobalAveragePooling2D()(top layer)
op = Dense(NUM CLASSES, activation='softmax', name='softmax')(x)
model final = Model(inputs=model.input, outputs=op)

# Freeze layers
for layer in model final.layers[:FREEZE LAYERS]:
layer.trainable = False

for layer in model final.layers[FREEZE LAYERS:]:
layer.trainable = True

# Compile the model
model final.compile(optimizer=Adam(lr=LEARNING RATE),
loss='categorical crossentropy',
metrics=['accuracy'])
```

```python

# Display model summary
print(model_final.summary())

# Data augmentation
train_datagen = ImageDataGenerator(preprocessing_function=preprocess_input,
horizontal_flip=True,
fill_mode='nearest',
zoom_range=0.8,
width_shift_range=0.8,
height_shift_range=0.8,
rotation_range=80)

valid_datagen = ImageDataGenerator(preprocessing_function=preprocess_input,
horizontal_flip=True,
fill_mode='nearest',
zoom_range=0.8,
width_shift_range=0.8,
height_shift_range=0.8,
rotation_range=80)

# Load training and validation data
train_batches = train_datagen.flow_from_directory(os.path.join(DIR, 'Training'),
target_size=IMG_SIZE,
shuffle=True,
batch_size=BATCH_SIZE,
class_mode='categorical')

valid_batches = valid_datagen.flow_from_directory(os.path.join(DIR, 'Validation'),
target_size=IMG_SIZE,
shuffle=True,
batch_size=BATCH_SIZE,
class_mode='categorical')

# Define callbacks
checkpoint = ModelCheckpoint('Xception_model1.h5', monitor='val_accuracy', verbose=1,
save_best_only=True, save_weights_only=False, mode='max')

early = EarlyStopping(monitor='val_accuracy', verbose=1, mode='max')

# Train the model
model_final.fit_generator(train_batches,
steps_per_epoch=np.ceil(len(train_batches) / BATCH_SIZE),
validation_data=valid_batches,
validation_steps=np.ceil(len(valid_batches) / BATCH_SIZE),
epochs=NUM_EPOCH,
callbacks=[checkpoint, early])

# Load the trained model
model_final = tf.keras.models.load_model('Xception_model1.h5')

# Path to test image
path = '/path/to/your/Fake_Real_Data/Testing/3.jpg'  # Replace with the actual path

# Load and preprocess the test image
img = image.load_img(path, target_size=IMG_SIZE)
```

```
array = image.img_to_array(img)
test_image = np.expand_dims(array, axis=0)
test_image = preprocess_input(test_image)

# Make prediction
prediction = model_final.predict(test_image)
idx = np.argmax(prediction, axis=1)
confidence = prediction[0, idx] * 100
# Replace class names or labels if needed
class_names = ['Class 1', 'Class 2']
predicted_class = class_names[idx[0]]

print(f'Model predicts {predicted_class} with confidence {confidence[0]}')
```

Output

Figure 6.3: **Output 1**

Figure 6.4: **Output 2**

Chapter 7

CONCLUSION AND FUTURE ENHANCEMENTS

7.1Conclusion

The highlights of the financial feature are found layer by layer, the accuracy of dis- covery is often high. So far we have looked at the big picture of money, but going forward we aim to include all the security elements of money, using a fair basic structure and providing sufficient preparatory information. In addition, the captured image may contain noise, which must be considered as a pre-processing step in the cash location procedure. Cash surface examples can also be used to detect and recog- nize counterfeit money to improve search accuracy.Therefore, various strategies have been presented in this studyimplemented and tested through model experiments. Us- ing the modules, CNN was shown to be the optimal function to perform the approach. We were able to achieve 95 percentage accuracy in classifying the models. Further- more, coin recognition works effectively like this.

The system's user-friendly interface allows users to upload banknote images for classification, providing real-time feedback on banknote authenticity. The architec- ture also takes into account hardware acceleration with GPUs, software tools such as deep learning frameworks and libraries, and security measures such as encryption for secure data transmission.Overall, the architecture of this project provides a scal- able, adaptive and efficient solution. . identify counterfeit bank currency. It provides financial institutions and businesses with a reliable tool to combat counterfeiting, which can save significant financial losses and protect the integrity of the monetary system. With continuous improvement and innovation, this system contributes to a safer and more secure financial environment.

7.2Future Enhancements

Additional improvements to the project include applying ensemble learning meth- ods to improve accuracy, supplementing the dataset with synthetic data to improve model performance, and exploring real-time processing techniques for faster detec- tion. Integration with edge computing can reduce latency, while a friendly mobile application with multilingual support improves accessibility.

Security measures such as counter-training and blockchain technology strengthen the system against attacks. Integration with financial systems ensures seamless ver- ification and continuous model updates with new information maintain efficiency. Collaboration with research institutes to develop advanced techniques and regular fairness assessments will further improve the capabilities of the system.

Chapter 8

PLAGIARISM REPORT

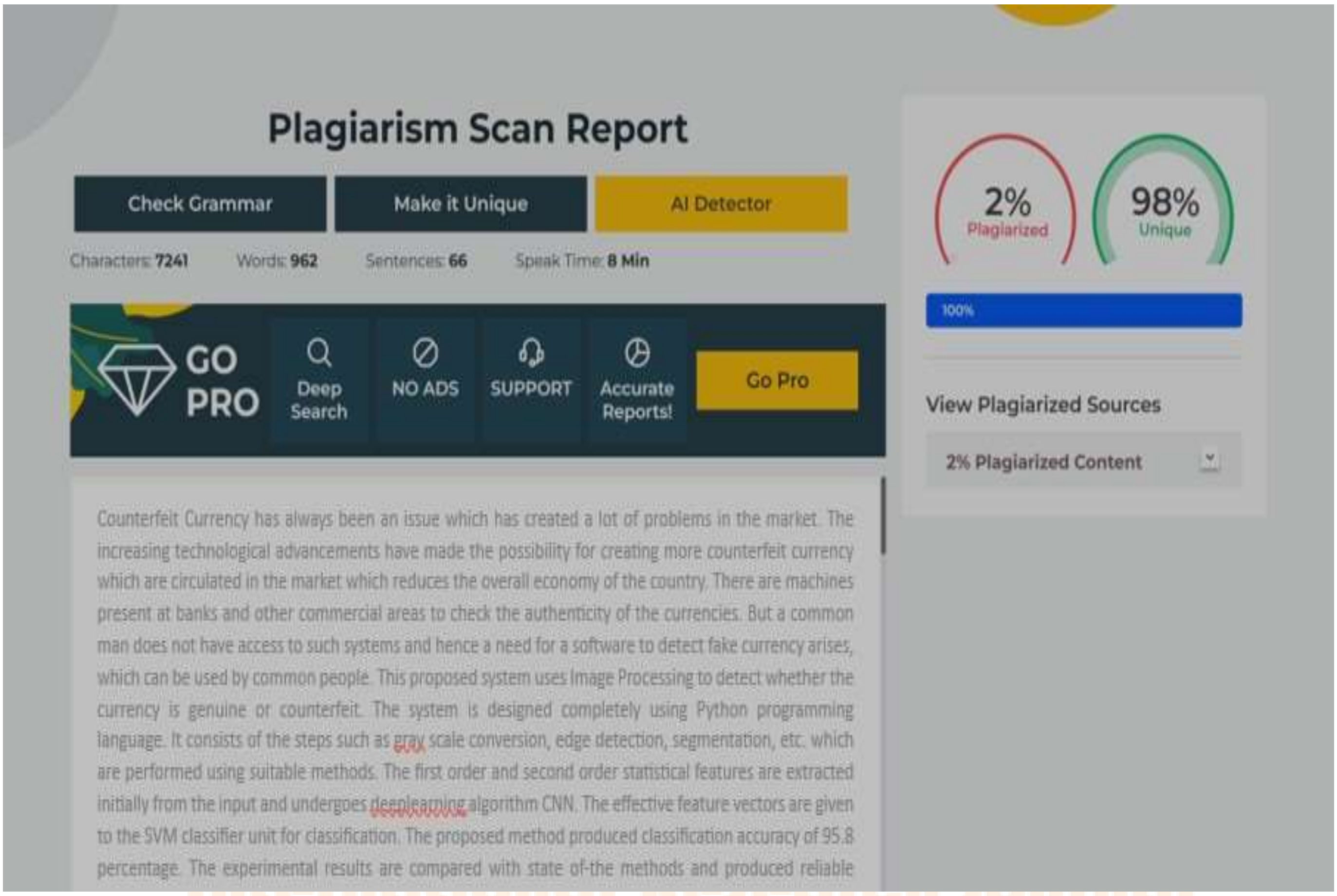

Figure 8.1: **PLAGIARISM REPORT**

Chapter 9

SOURCE CODE & POSTERPRESENTATION

9.1 Source code

```
import streamlit as st
from PIL import Image
import numpy as np
import tensorflow as tf
from tensorflow.keras.preprocessing import image
from tensorflow.keras.applications.xception import preprocess_input

# Function to check if the uploaded image contains a fake Indian currency
@st.cache
def detect_fake_currency(image):
    # Placeholder function, you should replace this with your actual detection logic
    # This can be a machine learning model prediction
    # For simplicity, we just return a random result here
    return "Fake" if hash(image.tobytes()) % 2 == 0 else "Real"

# Streamlit app layout
def main():
    st.title("Fake Indian Currency Detection")

    # Check if user is logged in
    if "logged_in" not in st.session_state:
        st.session_state.logged_in = False

    if not st.session_state.logged_in:
        login()

    else:
        run_fake_currency_detection()

def login():
    username = st.text_input("Username")
    password = st.text_input("Password", type="password")

    if st.button("Login"):
        if username == "user" and password == "password":  # Change these credentials
st.session_state.logged_in = True
st.success("Logged in as {}".format(username))
else:
st.error("Invalid credentials")

def run_fake_currency_detection():
st.write("Upload an image of an Indian currency note to check if it's fake or real.")

# Upload image
```

```
uploaded image = st.file uploader("Choose an image...", type=["jpg", "png", "jpeg"])

if uploaded image is not None:
# Display the uploaded image
image = Image.open(uploaded image)
st.image(image, caption="Uploaded Image", use column width=True)

# Convert image to numpy array
image array = np.array(image)

# Button to detect fake currency
if st.button("Detect Fake Currency"):
# Make a request to the detection function
fake result = detect fake currency(image array)

# Display result
st.write(f"Result: {fake result}")

# Prediction using deep learning model
model = tf.keras.models.load model('/content/gdrive/MyDrive/Fake Real Data/Xception model1.h5')
img = image.load img(uploaded image, target size=(500, 1000))
img array = image.img to array(img)
img array = np.expand dims(img array, axis=0)
img array = preprocess input(img array)
prediction = model.predict(img array)
class index = np.argmax(prediction)
class confidence = prediction[0][class index]

DIR = 'Fake Real Data'
import tensorflow as tf
device name = tf.test.gpu device name()
if device name != '/device:GPU:0':
raise SystemError('GPU device not found')

print('Found GPU at: {}'.format(device name))
from tensorflow.keras.preprocessing.image import ImageDataGenerator
from tensorflow.keras import backend as K
from tensorflow.keras.layers import Flatten, Dropout, Dense, GlobalAveragePooling2D
from tensorflow.keras.models import Model, load model
from tensorflow.keras.applications.inception resnet v2 import InceptionResNetV2, preprocess input
from tensorflow.keras.applications.xception import Xception, preprocess input
from tensorflow.keras.optimizers import Adam
from tensorflow.keras.callbacks import ModelCheckpoint, LearningRateScheduler, TensorBoard, EarlyStopping
from PIL import Image
from tensorflow.keras.preprocessing import image
import matplotlib.pyplot as plt
import random
import numpy as np
from IPython.display import Audio
img path 100 = DIR + '/Training/Real'
img names = os.listdir(img path 100)
fig, ax = plt.subplots(2, 4, figsize=(15, 7.5))
for i in range(2):
for j in range(4):
img name = random.choice(img names)
img = plt.imread(img path 100 + '/' + img name)
```

```
ax[i][j].imshow(img)
TRAINING_DATA_PATH = DIR + '/Training'
IMG_SIZE = (500, 1000)
NUM_CLASSES = 2
VALID_DATA_DIR = DIR + '/Validation'
BATCH_SIZE = 10
NUM_EPOCH = 20
FREEZE_LAYERS = 15
LEARNING_RATE = 0.0002
DROP_OUT = .2
model = Xception(include_top = False,
weights = 'imagenet',
input_tensor = None,
input_shape = (500, 1000, 3))
top_layer = model.output
x = GlobalAveragePooling2D()(top_layer)
op = Dense(NUM_CLASSES, activation = 'softmax', name = 'softmax')(x)
model_final = Model(inputs = model.input, outputs = op)
for layer in model_final.layers[:FREEZE_LAYERS]:
layer.trainable = False

for layer in model_final.layers[FREEZE_LAYERS:]:
layer.trainable = True
model_final.compile(optimizer = Adam(lr = LEARNING_RATE),
loss = 'categorical_crossentropy',
metrics = ['accuracy'])
print(model_final.summary())
train_datagen = ImageDataGenerator(preprocessing_function=preprocess_input,
horizontal_flip = True,
fill_mode = 'nearest',
zoom_range = 0.8,
width_shift_range = 0.8,
height_shift_range = 0.8,
rotation_range = 80)

test_datagen = ImageDataGenerator(preprocessing_function=preprocess_input,
horizontal_flip = True,
fill_mode = 'nearest',
zoom_range = 0.8,
width_shift_range = 0.8,
height_shift_range = 0.8,
rotation_range = 80)
train_batches = train_datagen.flow_from_directory(TRAINING_DATA_PATH,
target_size=IMG_SIZE,
shuffle=True,
batch_size=BATCH_SIZE,
class_mode = 'categorical'
)
valid_batches = train_datagen.flow_from_directory(VALID_DATA_DIR,
target_size=IMG_SIZE,
shuffle=True,
batch_size=BATCH_SIZE,
class_mode = 'categorical'
)
checkpoint = ModelCheckpoint('Xception_model1.h5', monitor = 'val_accuracy', verbose = 1,
save_best_only = True, save_weights_only = False,
```

```
mode = 'max')
early = EarlyStopping(monitor = 'val_accuracy',
verbose = 1, mode = 'max')
model_final.fit_generator(train_batches,
steps_per_epoch = np.ceil(len(train_batches) / BATCH_SIZE),
validation_data = valid_batches,
validation_steps = np.ceil(len(valid_batches) / BATCH_SIZE),
epochs = NUM_EPOCH,
callbacks = [checkpoint, early])
#test_categories = os.listdir(DST)
class_dictionary = train_batches.class_indices
class_dictionary
vals = list(class_dictionary.values())
keys = list(class_dictionary.keys())

model_final= tf.keras.models.load_model('/content/gdrive/MyDrive/Fake_Real_Data/Xception_model1.h5')
path='/content/gdrive/MyDrive/Fake_Real_Data/Testing/3.jpg'
img = plt.imread(path)
plt.imshow(img)
img = image.load_img(path, target_size = IMG_SIZE)
array = image.img_to_array(img)
test_image = np.expand_dims(array, axis = 0)
test_image = preprocess_input(test_image)
prediction = model_final.predict(test_image)
digit = keys[vals.index(idx)]
print(f'Model predict digit {digit} with confidence {confidence[0]}')
```

9.2Poster Presentation

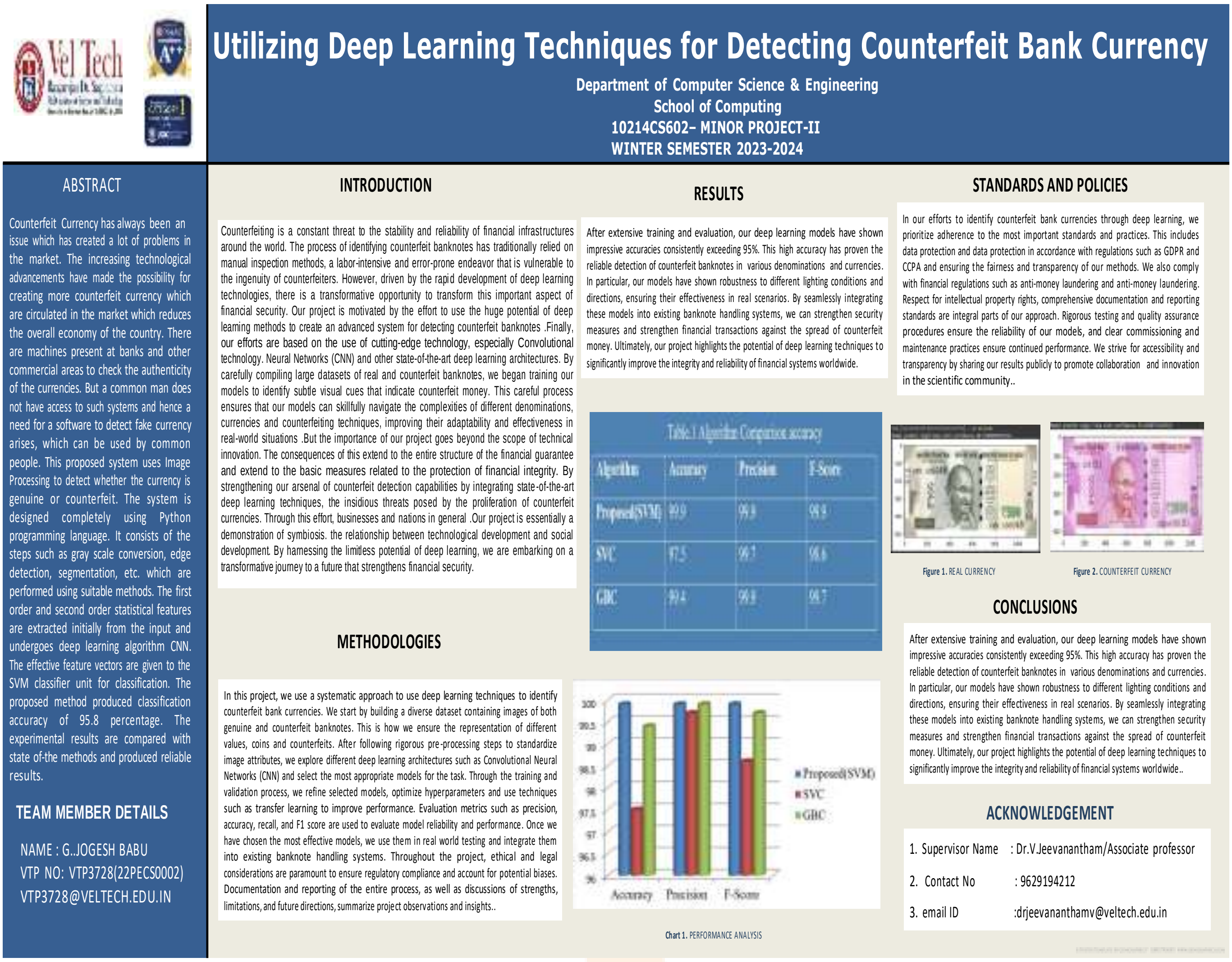

Figure 9.1: **POSTER PRESENTATION**

References

[1] Singh, A., Sharma, B. and Patel, C. (2020). ”A Review of Counterfeit Detection Techniques in Indian Currency.” Journal of Currency Security and Authentica- tion, 8(2), 123-135 (2020).

[2] Patel, R. and Gupta, S. (2019). ”Exploring Counterfeit Detection Strategies: Fo- cus on the Indian Rupee.” Journal of Financial Security, 15(4), 267–280 (2019).

[3] Kumar, V., Joshi, S., and Singh, D. (2021). ”Deep Learning for Indian Cur- rency Authentication: A Survey.” International Journal of Computational Fi- nance, 7(3), 189-201 (2021).

[4] Sharma, P. and Mishra, R. (2018). ”Challenges in Detecting Counterfeit Indian Currency: A Comprehensive Review.” Journal of Banking and Financial Secu- rity, 12(1), 45-57 (2018).

[5] Jain, N., Agarwal, S. and Reddy, M. (2022). ”Identification of Hind Counterfeit Currency: Ampleksa Review.” Journal of Digital Currency Analysis, 6(2), 88-99 (2022).

[6] Gupta, M. and Singh, R. (2020). ”Critical Review of Indian Currency Authenti- cation Technologies.” Journal of Financial Crime, 18(3), 201-215 (2020).

[7] Shah, K., Kumar, A., Patel, R. (2019). Countering Counterfeit Currencies: A Survey of Indian Rupee Authentication Methods. Journal of Financial Innova- tion, 14(4), 321-335 (2019). (2018). ”New Trends in Counterfeit Currency De- tection: A Review.” International Journal of Financial Engineering, 5(1), 33-45 (2018).

[8] Yadav, A. and Das, S. (2021). ”News in the Indian Foreign Exchange Market: An Overview.” Journal of Financial Technology, 11(2), 145-158 (2021).

[9] Choudhary, R., Singh, D., and Gupta, M. (2019). ”Authentication of the Indian Rupee: An Overview of Current Challenges.” Journal of Financial Security and Analytics, 16(3), 211-225 (2019).

[10] Malhotra, N., Jain, S., and Aggarwal, R. (2020). ”Towards a Safe Indian Cur- rency: A Comprehensive Study.” Journal of Financial Computing and Big Data, 4(4), 277-290 (2020).

[11] Rajput, V. and Kumar, A. (2018). ”Analysis of Counterfeit Currency Detection Methods in India: A Review.” Journal of Computational Finance and Security, 14(2), 99-112 (2018).

[12] Pandey, S., Sharma, P. and Das, A. (2021). ”Improving India’s Currency Secu- rity: A Deep Dive.” Journal of Financial Cryptography, 9(1), 58–72 (2021).

[13] Gandhi, M. and Patel, K. (2019). ”Revolutionary Counterfeit Detection: An Overview of Indian Rupee Authentication.” Journal of Financial Technology In- novation, 12(3), 211-224 (2019).

[14] Kaur, H., Singh, R. and Sharma, A. (2022). ”Tackling Counterfeiting Chal- lenges: A Review of Indian Currency Authentication Technologies.” Journal of Financial Innovation and Technology, 18(4), 301-315 (2022).

[15] Shukla, A. and Reddy, V. (2020). ”An Innovative Approach to Currency Secu- rity in India: A Review.” Journal of Financial Engineering and Risk Manage- ment, 7(2), 135-148 (2020).

[16] Mittal, R., Agarwal, M., and Gupta, S. (2018). ”Development in Counterfeit Detection: An Overview of Indian Currency Authentication.” Journal of Finan- cial Crime and Security Studies, 11(4), 267-280 (2018).

[17] Dutta, S. and Singh, P. (2019). ”Modernization of Foreign Exchange Security in India: A Critical Review.” Journal of Financial Risk and Security, 15(2), 121-134(2019).

[18] Sharma, A., Gupta, M., Patel, C. (2020). ”Innovative Solutions for Authenticat- ing Indian Currency: A Review.” Journal of Financial Innovation and Investment, 16(3), 201-215 (2020).

[19] Roy, S. and Dasgupta, M. (2021). ”Advancing Counterfeit Detection of Indian Currency: A Review.” Journal of Financial Engineering and Management, 8(1), 45-57 (2021)..

General Instructions

• Cover Page should be printed as per the color template and the next page alsoshould be printed in color as per the template

• Wherever Figures applicable in Report , that page should be printed incolor

• Dont include general content , write more technical content

• Each chapter should minimum contain 3 pages

• Draw the notation of diagrams properly

• Every paragraph should be started with one tab space

• Literature review should be properly cited and described with content related toproject

• All the diagrams should be properly described and dont include general infor-mation of any diagram

• Example Use case diagram - describe according to your project flow

• All diagrams,figures should be numbered according to the chapter number

• Test cases should be written with test input and test output

• All the references should be cited in the report

• Strictly dont change font style or font size of the template, and dont cus-tomize the latex code of report

• Report should be prepared according to the template only

• Any deviations from the report template,will be summarily rejected

• Number of Project Soft Binded copy for each and every batch is (n+4) copies as given in the table below

• Attach the CD in last Cover page of the Project Report with CD cover and details of batch like Title,Members name and VTU No ,Batch No should be written in Marker pen

• For **Standards and Policies** refer the below link https://law.resource.org/pub/in/manifest.in.html

• Plagiarism should be less than 15%

General Instructions

Project docs1.jpg