**FAKE CURRENCY DETECTION USING MACHINE LEARNING**

**……………………………………………**

**D.J. ABOLARIN**

**D.O. OGUNYE**

**E.G. OYIBO**

**……………………………………………**

**JANUARY 2025**

**FAKE CURRENCY DETECTION USING MACHINE LEARNING**

**BY**

**ABOLARIN DANIELLA JESUTOMI**

**OGUNYE DAVID OMOTAYO**

**OYIBO EFETOBOR GEORGE**

MATRICULATION/ REGISTRATION NUMBER: 20BD007281/ 2000037

MATRICULATION/ REGISTRATION NUMBER: 20BD007349/ 2001399

MATRICULATION/ REGISTRATION NUMBER: 20BD007336/ 2001816

A PROJECT SUBMITTED TO THE DEPARTMENT OF ELECTRICAL AND INFORMATION ENGINEERING, LANDMARK UNIVERSITY, OMU-ARAN, NIGERIA IN PARTIAL FULFILMENT OF THE REQUIREMENTS FOR THE AWARD OF BACHELOR OF ENGINEERING (B.ENG.) IN ELECTRICAL AND ELECTRONICS ENGINEERING

**SUPERVISOR: ENGR. ONIMISI ISAAC**

# CERTIFICATION

This Report submitted in the Department of Electrical and Information Engineering, College of Engineering, Landmark University, Omu-Aran, Nigeria. meets the requirements governing the award of the degree of Bachelor of Engineering in Electrical and Electronics Engineering and is approved for its contribution to knowledge and literary presentation

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Supervisor Signature and Date

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

External Examiner Signature and Date

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Head of Department Signature and Date

# DEDICATION

This report is dedicated to God Almighty for his Grace, Mercy, Guidance and Protection all through the period. Furthermore, I want to say Thank you to our supervisor Engr. ONIMISI Issac for his unwavering support to our Parents most especially for their care and guidance supporting us along the way.

# ACKNOWLEDGEMENT

Firstly, my sincere appreciation goes to the Almighty God for his guidance, grace, mercy, and protection throughout this project, and for his love in ensuring its successful completion.

I also wish to extend my most sincere appreciation to our parents, siblings, and all loved ones for their support and encouragement throughout this period. Their belief in us, valuable insights, motivation, and guidance have not only shaped my character but also contributed significantly to my personal growth.

Additionally, I extend gratitude to our supervisor, Engr. ONIMISI Isaac of the Department of Electrical and Information Engineering at Landmark University, for sacrificing time out of his busy schedule to supervise this project and for guiding me through the process as well as my spiritual advisors and friends, for their involvement and constant encouragement. My Sincere gratitude goes to our Project Coordinator; Engr. John Onyemenam and my lecturers who all have imparted us with knowledge, advice, and great insights enabling me to participate fully in this project.

# TABLE OF CONTENT

[TITLE PAGE .i](#_Toc199711789)

[CERTIFICATION iii](#_Toc199715386)

[DEDICATION iv](#_Toc199715387)

[ACKNOWLEDGEMENT v](#_Toc199715388)

[ABSTRACT xiv](#_Toc199715389)

[CHAPTER ONE 1](#_Toc199715390)

[INTRODUCTION 1](#_Toc199715391)

[1.1 Background of the Study 1](#_Toc199715392)

[1.2 Statement of problem 2](#_Toc199715393)

[1.3 Aim 2](#_Toc199715394)

[1.4 Objectives 2](#_Toc199715395)

[1.5 Scope 3](#_Toc199715396)

[1.6 Justification of the Study 3](#_Toc199715397)

[1.6.1 Traditional Methods' Drawbacks 4](#_Toc199715398)

[1.6.2 Scalability of the Solution 4](#_Toc199715399)

[1.6.3 Foundation for Future Research and Innovation 4](#_Toc199715400)

[1.6.4 Development of Technical Expertise 4](#_Toc199715401)

[1.6.5 Alignment with Sustainable Development Goals (SDGs) 5](#_Toc199715402)

[1.6.6 Cost-Effectiveness of Machine Learning Solutions 5](#_Toc199715403)

[1.6.7 Accessibility for Smaller Institutions 5](#_Toc199715404)

[CHAPTER TWO 6](#_Toc199715405)

[2 LITERATURE REVIEW 6](#_Toc199715406)

[2.1 Machine Learning Algorithms in Fake Currency Detection 7](#_Toc199715407)

[2.1.1 Supervised Machine Learning (SML) Approaches 7](#_Toc199715408)

[2.1.2 Ensemble Learning Techniques 8](#_Toc199715409)

[2.1.3 Deep Learning Approaches 9](#_Toc199715410)

[2.2 Image Processing Techniques 9](#_Toc199715411)

[2.2.1 Grayscale Conversion and Edge Detection 10](#_Toc199715412)

[2.2.2 Application of MATLAB for image processing 10](#_Toc199715413)

[2.2.3 Colour Histogram-Based Classification 11](#_Toc199715414)

[2.2.4 RGB Intensity and Segmentation 11](#_Toc199715415)

[2.3 Hybrid and Traditional Techniques 11](#_Toc199715416)

[2.3.1 Discriminant Function Analysis 12](#_Toc199715417)

[2.3.2 OCR and Feature-Based Techniques 12](#_Toc199715418)

[2.3.3 Statistical Techniques 13](#_Toc199715419)

[2.4 Deep Learning Approaches 13](#_Toc199715420)

[2.4.1 Convolutional Neural Networks (CNN) 13](#_Toc199715421)

[2.5 Real-Time and Application-Based Models 14](#_Toc199715422)

[2.5.1 Mobile and Real-Time Detection 14](#_Toc199715423)

[2.6 Limitations of Current Systems 15](#_Toc199715424)

[2.7 Currency Specificity and Lack of Real-Time Detection 15](#_Toc199715425)

[2.7.1 Dataset and Real-World Challenges 16](#_Toc199715426)

[2.8 Proposed Methodology: Support Vector Machines (SVM) 16](#_Toc199715427)

[CHAPTER THREE 20](#_Toc199715428)

[3 METHODOLOGY 20](#_Toc199715429)

[3.1 Data Collection 22](#_Toc199715430)

[3.2 Image Preprocessing 25](#_Toc199715431)

[3.3 Color Segmentation 26](#_Toc199715432)

[3.3.1 RGB Color Model 28](#_Toc199715433)

[3.3.2 HSV Color Model 29](#_Toc199715434)

[3.3.3 RGB to HSV Conversion 30](#_Toc199715435)

[3.4 Grayscale Conversion 31](#_Toc199715436)

[3.5 Morphological Processing 33](#_Toc199715437)

[3.5.1 Morphological processing algorithm 34](#_Toc199715438)

[3.6 Feature Extraction 35](#_Toc199715439)

[3.6.1 Statistical properties computed 37](#_Toc199715440)

[3.6.2 Features extracted in the Nigerian naira note. 39](#_Toc199715441)

[3.7 Create Training Dataset 40](#_Toc199715442)

[3.7.1 Feature Collection 40](#_Toc199715443)

[3.7.2 Label Assignment 42](#_Toc199715444)

[3.8 Model Training (SVM) 43](#_Toc199715445)

[3.8.1 Data Preparation 43](#_Toc199715446)

[3.8.2 Hyperparameter selection 44](#_Toc199715447)

[3.8.3 Training the model 45](#_Toc199715448)

[3.8.4 Cross-validation 45](#_Toc199715449)

[3.8.5 Regularization Techniques 46](#_Toc199715450)

[3.8.6 Training process 46](#_Toc199715451)

[3.8.7 Mathematical expression 48](#_Toc199715452)

[3.8.8 Significance of the SVM Model 49](#_Toc199715453)

[3.9 Testing of the Naira Note Image 49](#_Toc199715454)

[3.10 Software Used 51](#_Toc199715455)

[CHAPTER FOUR 52](#_Toc199715456)

[4 RESULTS AND DISCUSSION .52](#_Toc199715457)

[4.1 Introduction 52](#_Toc199715458)

[4.2 Experimental Results by Denomination 52](#_Toc199715459)

[4.2.1 ₦200 Naira Notes 52](#_Toc199715460)

[4.2.2 ₦500 Naira Notes 55](#_Toc199715461)

[4.2.3 ₦1000 Naira Notes 56](#_Toc199715462)

[4.2.4 Training Process Results 57](#_Toc199715463)

[4.2.5 Confusion Matrix and Performance Evaluation Criteria 60](#_Toc199715464)

[4.3 Comparative Analysis with Literature 61](#_Toc199715465)

[4.3.1 Performance Metrics Comparison 61](#_Toc199715466)

[4.3.2 Methodological Strengths 62](#_Toc199715467)

[4.3.3 Trade-offs and Limitations 62](#_Toc199715468)

[4.3.4 Recommendations from Literature 63](#_Toc199715469)

[4.4 Summary of Comparative Analysis 63](#_Toc199715470)

[4.5 Implementation, Programming Specifications, and Validation Protocols 64](#_Toc199715471)

[4.5.1 Code Structure Overview 64](#_Toc199715472)

[4.5.2 HSV Masking Key Code Snippets 65](#_Toc199715473)

[4.5.3 Extracting GLCM features 66](#_Toc199715474)

[4.5.4 Prediction with SVM 67](#_Toc199715475)

[4.5.5 Verification in Relation to Experimental Findings 67](#_Toc199715476)

[4.5.6 Implementation Restrictions Based on Image Quality: 68](#_Toc199715477)

[CHAPTER FIVE: 69](#_Toc199715478)

[SUMMARY, CONCLUSION AND RECOMMENDATIONS 69](#_Toc199715479)

[5.1 Summary 69](#_Toc199715480)

[5.1.1 System Features and Configuration 70](#_Toc199715481)

[5.1.1.1 Software Environment: 70](#_Toc199715482)

[5.1.1.2 Hardware Configuration: 70](#_Toc199715483)

[5.1.1.3 Functional Components: 70](#_Toc199715484)

[5.1.2 Performance and Accuracy 71](#_Toc199715485)

[5.1.2.1 Accuracy 71](#_Toc199715486)

[5.1.2.2 Precision and Recall 72](#_Toc199715487)

[5.1.2.3 F1-Score 72](#_Toc199715488)

[5.1.2.4 Reliable Communication 72](#_Toc199715489)

[5.1.2.5 Confusion Matrix 73](#_Toc199715490)

[5.1.2.6 Real-Time Processing 73](#_Toc199715491)

[5.1.2.7 Robustness 73](#_Toc199715492)

[5.1.3 Importance of the System 74](#_Toc199715493)

[5.1.3.1 Scalable Solution for Financial Institutions 74](#_Toc199715494)

[5.1.3.2 Economic Impact 74](#_Toc199715495)

[5.1.3.3 Financial Security and Anti-Fraud Protection 75](#_Toc199715496)

[5.1.3.4 Speed, Accuracy, and Scalability 75](#_Toc199715497)

[5.1.3.5 Accessibility for Underserved Areas 75](#_Toc199715498)

[5.1.3.6 Advancement of Technical Expertise 75](#_Toc199715499)

[5.1.3.7 Foundation for Future Research and Innovation 75](#_Toc199715500)

[5.2 Recommendations 76](#_Toc199715501)

[5.3 Expand the Dataset with Greater Diversity 76](#_Toc199715502)

[5.3.1 Integrate Deep Learning Techniques 76](#_Toc199715503)

[5.3.2 Develop a Mobile Application 77](#_Toc199715504)

[5.3.3 Integrate the System in Banking Infrastructure 77](#_Toc199715505)

[5.3.4 Government and Institutional Collaboration 78](#_Toc199715506)

[5.3.5 Implement Cloud-Based Updates and Analytics 78](#_Toc199715507)

[5.3.6 User Interface and Experience (UI/UX) Improvement 79](#_Toc199715508)

[5.3.7 Extension to Multi- Currency Detection 79](#_Toc199715509)

[5.3.8 Continuous Model Retraining and Feedback Loop 79](#_Toc199715510)

[5.3.9 Train End-Users and Raise Public Awareness 79](#_Toc199715511)

[5.4 Conclusion 80](#_Toc199715512)

[REFERENCES 82](#_Toc199715513)

# TABLE OF FIGURES

[**Figure 3.2: ₦200 naira note** 23](#_Toc199763357)

[**Figure 3.3: ₦500 naira note** 23](#_Toc199763358)

[**Figure 3.4: ₦1000 naira note** 24](#_Toc199763359)

[**Figure 3.9: 1000 Naira Grayscale converted image** 33](#_Toc199763361)

[**Figure 3.10: Morphological processing code** 35](#_Toc199763362)

[**Figure 3.12: Morphologically processed 500 Naira Note** 36](#_Toc199763363)

[**Figure 3.13: Morphological processed 1000 Naira Note** 36](#_Toc199763364)

[**Figure 3.14: Sample of extracted features in a Nigerian naira note** 40](#_Toc199763365)

[**Figure 3.16: Code sample of passing the matrix to the SVM Model** 47](#_Toc199763366)

[**Figure 3.18: Properties of the tested Naira Note that had the result of ‘Real’** 51](#_Toc199763367)

[**Figure 4.1: Image of testing and validation accuracy data and loss data per iteration** 55](#_Toc199763368)

[**Figure 4.2: Image of code for hologram detection** 56](#_Toc199763369)

[**Figure 4.5: RGB TO HSV conversion variable** 65](#_Toc199763372)

[**Figure 4.6: GLCM features extraction code** 66](#_Toc199763373)

# ABSTRACT

The large-scale counterfeiting of the Nigerian currency has turned into a serious economic and security issue, calling for better detection approaches than mere manual checks. In this paper, an automated machine learning system is introduced to identify counterfeit Nigerian Naira notes of the ₦200, ₦500, and ₦1000 denominations. The introduced approach merges powerful image processing methods with Support Vector Machine (SVM) classification to identify real and counterfeit banknotes efficiently.

The work utilized a large dataset of 3,000 scanned documents (1,500 authentic and 1,500 counterfeit) captured under varying conditions, including varying light, orientation, and physical state (e.g., worn, damaged). The images were thoroughly preprocessed, for example, resizing to the same 75.5mm x 39mm dimensions, background subtraction via HSV color segmentation, grayscale conversion, and morphological processing to eliminate noise from significant security features. All the significant features such as pattern of textures, entropy, contrast, and structural properties were obtained by using the assistance of Gray-Level Co-occurrence Matrix (GLCM) analysis.

The best SVM classifier with RBF kernel using Bayesian optimization had a better performance with a mean accuracy of 90.25%. Remarkably, the system worked 90.51% in the case of ₦200, 93.1% in the case of ₦500, and sheer 100% accuracy for ₦1000 notes in optimum scenarios. The model was also made robust based on precision, recall, and F1-score as measures and demonstrating repeated performance even if the notes were mutilated partially or less enlightened.

Comparison with other prevailing approaches in spurious note detection revealed astronomical advantages of the system being developed, here referred to as applied specifically in the case of Nigerian currency-specific security features. Texture analysis by GLCM and SVM classification was a better approach compared to conventional template-matching techniques while maintaining high computational efficiency for real-time execution. Modularity in system design offers ease of integration with existing banking infrastructures like ATMs and point-of-sale terminals and potential for integration with mobile phones.

Nigerian economic security is via this project in the sense that it is using a scalable, accurate and standalone counterfeit currency detector. Future work will include expanding on the denomination and type of dataset, and using deep learning techniques like Convolutional Neural Networks (CNNs) to enhance feature extraction. Apps for mobile devices easier to use for convenient accessibility will be created. The study concludes the feasibility of using machine learning for fraud prevention in finance and as a platform to further develop systems for currency verification.

# CHAPTER ONE

# INTRODUCTION

## Background of the Study

To provide smooth international trade and financial operations, currency is an essential medium of exchange in economic transactions. Online currency exchange is still not widely used in Nigeria, even though internet usage has recently increased due to reasonably priced and excellent internet connection. As a result, most people and businesses still primarily use physical banknotes for transactions. But as printing technology has advanced, it has become more challenging to identify counterfeit money. Nowadays, counterfeit notes are produced with such accuracy that it is practically difficult to tell the difference between authentic and counterfeit notes using human inspection techniques.

To properly handle this expanding problem, cutting-edge-technology like machine learning are now required. By identifying important characteristics from currency notes, such as texture, skewness, variance, and entropy, Support Vector Machines (SVM), a strong classification technique, offer a potent way to precisely detect counterfeit notes. To determine if notes are authentic or fraudulent, these characteristics can be extracted using wavelet transform methods and examined. There is a great deal of promise for using an SVM-based counterfeit detection system in high-cash-flow settings like banks, automated teller machines (ATMs), vending machines, and shopping centres. This technology combats the societal problem of counterfeit currency penetration in the market, improves efficiency, and decreases human error by automating the detecting process. This study uses cutting-edge machine learning approaches to address this urgent economic issue by using a banknote authentication dataset that includes pictures of real and counterfeit cash notes. Additionally, the basis for examining security features like holograms, watermarks, and distinctive patterns inscribed in bank notes is provided by image processing methods including colour segmentation, grayscale conversion, and feature extraction. In addition to automating counterfeit detection, this SVM and image processing combo improves its speed and accuracy, providing a scalable solution to this urgent problem. By integrating these technologies, Nigeria’s financial systems should become more modern and less vulnerable to counterfeiting.

## Statement of problem

The widespread circulation of counterfeit Nigerian currency has grown to be a serious problem that presents difficulties for businesses and financial institutions. Conventional techniques for identifying fake currency are unreliable and unscalable for widespread use. The need for an automated solution that combines accuracy, speed, and scalability to improve financial security is critical given the advancement of counterfeiting techniques.

## Aim

The aim of this project is to detect counterfeit Nigerian paper currency note using machine learning technique.

## Objectives

The objectives of this project are to:

1. Collect and create a dataset of real and counterfeit Nigerian currency images.
2. Preprocess images using techniques such as resizing, colour segmentation, and grayscale conversion.
3. Train an SVM classifier using labelled data to differentiate between genuine and counterfeit notes.
4. Validate the system’s performance to fine-tune the model and prevent overfitting.
5. Evaluate the system’s accuracy and performance using test datasets
6. Test the reliability of the system's predictions under real-world conditions by accessing its performance on previously unexamined test data.

## Scope

With a focus on denominations like ₦200, ₦500, and ₦1000, this project aims to create a counterfeit detecting system exclusively for Nigerian currency notes. Images of genuine and fake currency notes in all of these denominations and in various states (such as new, ripped, or worn) will be gathered for the system. To produce a complete dataset, the photos will undergo preprocessing to ensure homogeneity and the extraction of distinctive attributes. Based on these characteristics, an SVM classifier will be trained to correctly identify counterfeit notes. Scalability was included in the system's architecture to enable future adaptation in retail settings and automated teller machines (ATMs).

## Justification of the Study

The application of machine learning techniques for identifying Nigerian banknotes brings considerable advantages to the banking, e-commerce, and financial industries by delivering more precise, quicker, and scalable methods for detecting counterfeit currency. This research seeks to tackle these issues by employing machine learning, particularly the Support Vector Machine (SVM) algorithm, to develop an efficient and innovative solution for detecting counterfeit naira notes. Below are the key justifications for undertaking this study:

### ****Traditional Methods' Drawbacks****

Traditional techniques for detecting counterfeit notes often depend on human examinations, such as those conducted by bank tellers. However, the subjective nature of eye assessment can lead to human error. As counterfeiters become more adept at replicating the characteristics of genuine currency, these methods lose their effectiveness, necessitating a more reliable and automated solution.

### Scalability of the Solution

One significant benefit of the suggested system is its scalability. Without compromising accuracy or efficiency, the SVM-based model can manage high numbers of currency verification activities. Because of this, it may be used in a variety of contexts, ranging from big financial institutions with significant transaction volumes to smaller institutions in places with limited resources.

### Foundation for Future Research and Innovation

This study lays the groundwork for further advancements in the field of financial fraud detection. By developing an ML-based model, this study paves the way for further advancement and improvement of machine learning methodologies to keep pace with evolving counterfeiting methods. The system can be updated and adapted as new fraudulent techniques emerge, ensuring the model remains effective and accurate over time.

### Development of Technical Expertise

This study not only develops a counterfeit detection model but also contributes to the enhancement of technical expertise within the banking and financial sectors. As financial institutions adopt machine learning methods, professionals within the industry will be equipped with the skills to implement and manage these advanced systems. This will drive innovation, improve the resilience of financial institutions to fraud, and ensure that the workforce is prepared to handle increasingly complex financial technologies.

### Alignment with Sustainable Development Goals (SDGs)

The study is in line with Sustainable Development Goal 9 (Industry, Innovation, and Infrastructure), which promotes resilient infrastructure development and inclusive, sustainable industrialization. This research helps create a more effective, dependable, and secure infrastructure that fosters innovation and propels economic development by integrating machine learning into financial systems. Additionally, it promotes a safe financial environment, which helps Nigeria and strengthens the resilience of the global economy.

### Cost-Effectiveness of Machine Learning Solutions

Machine learning technology may be more expensive to set up initially, but because of its scalability, automation, and low need for human participation, it eventually becomes cost-effective. This makes it a viable option for financial institutions.

### ****Accessibility for Smaller Institutions****

Machine learning-based systems can be more accessible to financial institutions in underserved or rural areas. With cloud-based solutions, smaller banks and financial institutions can deploy advanced counterfeit detection systems without the need for expensive infrastructure or training, democratizing access to cutting-edge financial technology.

# CHAPTER TWO

# LITERATURE REVIEW

**Introduction**

The growing problem of counterfeit currency has forced the development of automated systems that can correctly detect false notes. Traditional procedures such as watermark checks and human inspections are insufficient for large-scale, real-time applications. The rise of machine learning (ML) algorithms, in combination with image processing techniques, has revolutionized currency identification by enabling systems to learn patterns, characteristics, and textures that are unique to genuine currency.

Currently, numerous identification techniques are employed to identify photos, faces, vehicle license plates, and human behaviours. Currency serves as the principal medium of exchange, and the currencies of various countries has distinct characteristics. Nevertheless, when the value of cash appreciates, there will be a rise in counterfeit currency. Counterfeit currency may jeopardize the interests of these nations. Consequently, a prominent topic and pressing concern currently is the application of identification technology for the authenticity of currency.

Visual examination was employed in the past to detect genuinely authentic money, particularly currency notes. Our eyesight cannot sense everything; sometimes, it is not simple for humans to identify genuine currency from the fake currency without the use of technology. Although UV-based-recognition with the rising sophistication of counterfeiting systems, a system is still in operation. Moreover, it's getting more tough to discern the difference between phony and genuine currency (Zhang & Yan, 2018).

From day to day, the progress of automated systems and approaches for currency recognition have improved (Jadhav et al., 2019). Artificial intelligence (AI) has been established in a variety of sectors, including civil engineering, medical, and image processing. AI is built on neural networks used for currency recognition and for a variety of different currency related tasks, including detecting currency note-portrait, detecting fake-notes, recognizing currency note-serial-number, and extracting and identification of currency note features (Veeramsetty et al., 2020).

Deep learning approaches, such as multilayer neural networks, are recognized to be beneficial in various applications.

## Machine Learning Algorithms in Fake Currency Detection

Machine learning has proven its’ importance in increasing the accuracy of fake currency detecting systems. Various methods such as Support Vector Machines (SVM), K-Nearest Neighbors (KNN), Decision Trees, and deep learning models like Convolutional Neural Networks (CNN) have been applied.

### Supervised Machine Learning (SML) Approaches

This paper (Kumar et al., 2022) proved the usefulness of supervised machine learning methods like K-Nearest Neighbors (KNN), Support Vector Machines (SVM), Decision Tree, Random Forest, and Logistic Regression in identifying currency notes. Among these strategies, KNN shone out, reaching a 100% accuracy rate under an 80:20 train-test data split. This good performance indicates the potential of KNN for counterfeit detection. However, its reliance on high-quality datasets poses challenges in real-world circumstances where image quality and consistency could vary substantially.

Support Vector Machines (SVM) have also been extensively commended for its power to tackle binary classification challenges (Bhatia et al., 2021) Based his work on the usefulness of SVM in detecting counterfeit notes by considering criteria’s such as colour, texture, and material quality. Its validity against overfitting and ability to perform effectively with fewer datasets gives SVM an advantage over other popular choice in many applications. However, going further to use engineering methods like Principal Component Analysis (PCA) is sometimes required to increase model performance.

### Ensemble Learning Techniques

Ensemble techniques, such as Random Forest and Gradient Boosting, combine several weak learners to build a strong prediction model. Random Forest, for instance, aggregates the results of several decision trees to enhance accuracy and reduce volatility. (Kumar et al., 2022) analysis demonstrated Random Forest to be highly successful at detecting counterfeit notes with complicated and overlapping features. These methods are wanted for their ability to handle noisy data and lower the chance of overfitting, however they can be computationally expensive. (Shelke et al., 2023) used Random Forest algorithms along with image processing techniques, including grayscale conversion, edge detection, and feature extraction. The system obtained 84.25% accuracy, 66.25% recall, and 78.63% precision. However, the short training sample harmed its overall performance and resilience.

### Deep Learning Approaches

This method was presented by (Richard-Nnabu et al., 2024) the method uses Convolutional Neural Networks (CNN) to detect counterfeit Nigerian Naira notes. Their model had an accuracy of 95% even when some of the notes were folded or partially faded notes. The strength of CNN’s in handling such complexities underlines their potential for counterfeit currency identification. However, the study emphasized on the need for larger and more diverse datasets to boost the relevance of the approach currency identification. (Roy et al., 2019) employed grayscale conversion, segmentation, and the Structural Similarity Index Measure (SSIM) with CNN’s for recognizing counterfeit Indian banknotes. Although their approach provided good accuracy for Indian notes, its lack of real-time applicability remained a barrier. Another important deep learning approach was produced by (Kumar et al., 2020), who designed a three-layer CNN obtaining 96.9% accuracy. Their study revealed issues associated to soiled or destroyed notes, which needs further research to overcome. (Ibitoye, 2024) implements a Modified Faster R-CNN with the Inception V3 architecture. Achieving a detection accuracy of 97%, the technique struggled with worn out or damaged notes and required high-quality photos for accurate detection. (Abishek & Kavin, 2020) studied CNNs for classification while emphasizing on dataset size. The accuracy increased with a large dataset yet the system struggled at noise levels. (Sruthy, 2022) contrasted approaches, highlighting CNN's great accuracy despite processing expenses, whereas SVM and KNN were less resource-intensive but somewhat successful.

## Image Processing Techniques

Image processing techniques plays a significant role in counterfeit detection by enabling the extraction and analysis of essential data from images. These techniques include grayscale conversion, edge detection, segmentation, and feature extraction, all of which contribute to the effective identification of original notes from fakes.

### Grayscale Conversion and Edge Detection

In this paper (Roy et al., 2019) introduced an image processing-based counterfeit detection system implemented in Python. The technique employs Grayscale Conversion to reduce colour images to a single intensity layer, hence simplifying computations while keeping crucial properties of the currency note. Edge Detection Techniques such as Canny edge detection are applied to identify boundaries inside the image, enabling the extraction of features like text, symbols, and patterns. Segmentation and Feature Extraction: These techniques separates the images into individual sections, enabling the detection of security characteristics such as watermarks, security threads, and denomination markers. (Roy et al., 2019) used grayscale conversion, edge detection, segmentation, and SSIM for feature comparison, focussing on Indian currency. The system lacked real-time accessibility.

Edge detection is particularly effective for recognizing important features that are difficult to recreate in counterfeit notes. The technique enhances the accuracy of feature extraction algorithms, guaranteeing that even minute information is recorded efficiently.

### Application of MATLAB for image processing

(Swami et al., 2019) employed MATLAB to construct edge detection and feature extraction algorithms. While their system effectively identified currency sorts, its effectiveness was limited for U.S. Dollars and lacked integration across various currencies. Their technique exhibited greater accuracy, particularly for lower-quality photos acquired under variable illumination circumstances.

### Colour Histogram-Based Classification

In this paper (Omeiza et al., 2023) introduced a new approach of using color histograms in the HSI color space for categorizing Nigerian Naira denominations, the 200, 500, and 1000 notes. The rule-based classifier used 10-bin histograms for hue, saturation, and intensity components, obtaining an average accuracy of 98.66%. The technique displayed good performance with minimum preprocessing demands, making it efficient for the real-world applications. Despite its success, the approach presented issues with soiled or damaged notes, which were handled by focusing on accuracy for lesser denominations. (Almu & Muhammad, 2017) constructed a Visual Basic and MS Access-based system, getting a 77.7% recognition accuracy. The system faced trouble with damaged notes and relied mostly on the quality of the database.

### RGB Intensity and Segmentation

In this paper (Sharan & Kaur, 2019) made use of RGB mean intensity calculations and segmentation to recognize Indian rupee notes. Although their technique attained an accuracy of 76.66%, its applicability was constrained to high denominations and controlled environmental conditions. This shortcoming underscores the need for more resilient approaches capable of adjusting to varied settings. RGB intensity analysis provides a thorough picture of colour distributions across the currency note, which is vital for differentiating genuine notes from counterfeits. However, the technique’s dependence on regular illumination conditions can restrict its real-world application.

## Hybrid and Traditional Techniques

Hybrid and traditional strategies integrate statistical methods with machine learning and image processing tools to increase counterfeit detection accuracy.

### Discriminant Function Analysis

(Iwok & Akpan, 2016) created a discriminant function for identifying authentic and counterfeit Naira notes. The study applied discriminant analysis, Hotelling’s T² test, and Mahalanobis distance to discover significant differences in physical parameters such as mass, thickness, and tensile stress. While this approach proved reliable, its validation was hampered by a small sample size of 20 notes, underlining the need for larger datasets.

### OCR and Feature-Based Techniques

(Zarin & Uddin, 2019) applied Optical Character Recognition (OCR), facial recognition, and Hough characteristics to identify counterfeit Bangladeshi cash. Their model demonstrated the usefulness of merging several feature-based approaches by reaching a 93.33% accuracy rate. However, increased implementation expenses and limited ability to generalize remain major problems (Adewole et al., 2023) paired CNN’s with OCR technologies like Pytesseract to obtain 100% classification accuracy. OCR performance was inadequate for real-time applications accuracy.

(Woods & Oladosu, 2020) utilized Speeded-Up Robust Features (SURF) for feature extraction and employed K-Nearest Neighbors (KNN) for classification. Their approach obtained perfect accuracy in recognizing Nigerian cash, however it was confined to the ₦500 and ₦1000 denominations.

(Srarka & Pal, 2024) utilized fuzzy logic to analyse alterations in the security features of Indian currency notes. While their technique accurately spotted counterfeits, it was significantly dependent on note quality and lacked real-time application. (Sarkar & Pal, 2022) used fuzzy inference for recognizing monetary attributes, giving accurate counterfeit detection but lacked real-time digital applications.

### Statistical Techniques

Statistical methodologies like Principal Component Analysis (PCA) and Linear Discriminant Analysis (LDA) are commonly used to reduce dimensionality and boost the performance of machine learning models. These methods facilitate expedited processing and improve classification accuracy by transforming high-dimensional data into a lower-dimensional space.

## Deep Learning Approaches

Deep learning techniques, particularly Convolutional Neural Networks (CNN’s), have revolutionized counterfeit detection by automating feature extraction and categorization procedures.

### Convolutional Neural Networks (CNN)

(Kumar et al., 2020) constructed a three-layer CNN to spot counterfeit Indian notes, getting a 96.9% accuracy rate. The model’s performance, however, was hindered by stained and broken notes. Similarly, (Ogbuju et al., 2020) employed Faster R-CNN to obtain 98%-99% accuracy on higher denomination Naira notes, yet computer resource needs faced scalability concerns. (Rao & Gowtham, 2024) merged ResNet50V2 with transfer learning to produce a robust system capable of tolerating light variations. Their model obtained 96.76% accuracy but needs more enhancements for managing damaged notes.

Deep learning algorithms normally demand massive datasets for training, which can be a big challenge in counterfeit identification. Data augmentation techniques, such as rotation, scaling, and noise injection, can help overcome this issue by artificially boosting the size and diversity of the training dataset. While deep learning models have shown considerable accuracy, they usually involve large datasets, significant computing resources, and long preparation. Addressing these limits is crucial for real-world applications.

## Real-Time and Application-Based Models

Real-time detection technologies are becoming crucial in answering the growing demand for rapid and practical counterfeit detection solutions. These systems strive to give fast response to clients, boosting their usability in conventional applications, such as retail transactions, banking, and self-service kiosks. Several research have centred on mobile and real-time apps to bridge the gap between technology correctness and consumer satisfaction.

### Mobile and Real-Time Detection

(Vidhate et al., 2021) created a real-time detection system that used Convolutional Neural Networks (CNN’s). This tool featured a user-friendly interface that simplified the detection method for end-users. However, its performance was mostly connected to assessing the front-facing properties of money notes and barcode detection, which limited its ability to handle different circumstances. The reliance on these traits hindered its flexibility to shifting illumination settings and note orientations, reducing its practical usage in uncontrolled environments. Similarly, (More et al., 2020) constructed a Python-based brute-force matching system that exploited ORB descriptors and KNN Match algorithms. While the system had great performance under perfect lighting conditions and static setups, it flopped in cases including inadequate illumination or fluctuations in note presentation. The lack of robustness in diverse scenarios reduced the need for this method in adaptability to real-world environments.

(Laavanya & Vijayaraghavan, 2019) applied AlexNet-based transfer learning for real-time detection of Indian cash notes. Their technique made use of data augmentation to fasten the training process and produced remarkable accuracy rates. Despite this, the model’s performance was restrained by the lack of diversity in its training dataset, reducing its potential to generalize to unfamiliar note kinds or currencies. These limitations underline the need of generating more inclusive datasets that represent real-world complexities. (Nazir et al., 2024) employed GLCM for feature extraction, PCA for optimization, and classifiers such as Decision Tree, SVM, and KNN. Their model produced considerable accuracies, with KNN at 96.44% and SVM at 93.85%. The concentration was on large denominations and requires high-resolution photos. (Nair et al., 2024) built CNN’s (VGG-16) with TensorFlow and OpenCV, reaching 92.75% accuracy. The model had problems in expanding to other currencies.

## Limitations of Current Systems

The existing systems for counterfeit detection, while technology advances, shows significant limitations that restrict their scalability and general applications. These drawbacks can be categorized into issues related to currency specificity, real-time application, and dataset challenges. (Bhoyar et al., 2020) utilized SVM and cloud storage for Indian currency detection. Though the system showed good accuracy, it depended on a small dataset and cloud storage.

## Currency Specificity and Lack of Real-Time Detection

In this paper (Shiby et al., 2021) a Python-based counterfeit detection system was designed specifically for Indian rupee notes. While the system exhibited good accuracy in its target currency, it lacked the flexibility to adjust to other currencies, leaving it with less importance for worldwide applications. However, the absence of real-time detection capabilities and integration with mobile applications further hampered its usability. Future innovations could focus on enhancing currency recognition, real-time responsiveness, and mobile accessibility to suit the demands of a larger user base.

### Dataset and Real-World Challenges

(Naseem et al., 2023) designed a MATLAB-based detection system for Pakistani banknotes, showcasing precision in identifying denominations such as 500, 1000, and 5000 PKR. However, the system’s reliance on high-resolution image inputs posed a significant barrier to its deployment in real-world settings, where such conditions are often impractical. The lack of mobile connectivity further limited its accessibility, particularly in locations with little technology infrastructure. Real-world application continues to be problematic due to restrictions in dataset availability, diversity, and quality, underscoring the need for additional complete and representative datasets to better the robustness and reliability of detection systems in diverse settings.

## Proposed Methodology: Support Vector Machines (SVM)

Support Vector Machines (SVM) represent a foundation in the development of counterfeit cash detection systems due to its stability in binary classification problems and capability to handle large datasets with high dimensionality. This part explains more about the proposed methodology, also talks about how SVM can be crucial in counterfeit detection, alongside the preprocessing and feature optimization strategies to increase performance. Why SVM for Counterfeit Detection? SVM is typically known as one of the most successful machine learning algorithms for binary classification tasks, such as distinguishing real currency notes from counterfeits. Its key strengths include Margin Optimization where SVM identifies the hyperplane that maximizes the margin between multiple classes, ensuring high generalization capabilities and robustness against overfitting, and Kernel Trick By applying kernel functions such as radial basis function (RBF), polynomial, or linear kernels, SVM can effectively transfer non-linear correlations in the data to a higher-dimensional space where they become linearly separable, with Handling High-Dimensional Data, SVM is well-suited for datasets with a large number of features, which is frequent in counterfeit detection scenarios with sophisticated patterns and textures.

Workflow for SVM-Based Counterfeit Detection

1. Data collection: The process begins with collecting real and fake notes and the Images are captured under varied situations (e.g., lighting, angles, and note quality) to replicate real-world scenarios. High-quality datasets ensure superior training and testing results.   
2. Preprocessing: Preprocessing involves standardizing and improving the incoming data to ensure consistency and decrease noise. Steps include Grayscale Conversion which is to Reduce processing complexity by focusing on intensity fluctuations rather than colour, and Normalization which Ensures that all pixel values are scaled to a uniform range, typically [0, 1], for improved model performance, also Edge Detection Techniques like Sobel or Canny edge detection reveal essential features, such as watermarks, security threads, and denomination markers.

3. Feature Extraction: Features are collected from the pre-processed images using approaches like Analyzer of Oriented Gradients (HOG): Captures edge and texture patterns crucial for identifying real notes from counterfeits, and Wavelet Transform Breaks down the image into many frequency components, assisting in the finding of minute changes in texture and design, also with Principal Component Analysis (PCA) to Reduces dimensionality while maintaining crucial information, enhancing SVM’s efficiency and accuracy.

4. Model Training: The SVM model is trained on the extracted features using labelled data. During this phase the training dataset is split into folds for cross-validation to confirm that the model generalizes effectively to unknown data. Hyperparameters such as the kernel type, regularization parameter (C), and gamma (for RBF kernels) are optimized using grid search or random search algorithms.

5. Testing and Validation: The model's performance is tested using a different test dataset. Metrics such as accuracy, precision, recall, and F1-score are generated to assess the efficacy of the SVM in recognizing counterfeit notes. Additionally, Receiver Operating Characteristic (ROC) curves are evaluated to establish the trade-off between true positive and false positive rates. Integrating SVM with Preprocessing Techniques, Combining SVM with new preprocessing techniques can greatly increase its performance. For instance, Wavelet Transform and PCA By gathering both spatial and frequency domain data, this combination ensures that the most relevant properties of currency notes are retained while lowering noise and redundancy. Gabor Filters are good at extracting texture-based features, such as the minute details in watermarks and security threads, which are vital for counterfeit identification.   
Advantages of Using SVM are, High Accuracy where SVM frequently demonstrates superior accuracy compared to other classic machine learning techniques in counterfeit identification testing, and Scalability, despite being computationally costly, SVM's scalability can be increased by methodologies like as parallel processing, and Robustness to Small Datasets where SVM performs well even with minimal data, making it appropriate for scenarios where massive, labelled datasets are unavailable.

Limitations and Mitigation Strategies, while SVM is highly effective, it has severe limitations such has Computational Complexity of Training SVM on large datasets can be time-consuming. Using efficient libraries like LIBSVM or building stochastic gradient-based solvers helps overcome this issue, Sensitivity to Imbalanced Data can distort the model's predictions. Techniques like Synthetic Minority Over-sampling Technique (SMOTE) or cost-sensitive learning can mitigate this problem. Difficulty with Multi-Class Classification, although inherently a binary classifier, SVM can be extended to multi-class situations utilizing approaches like one-vs-one (OVO) or one-vs-rest (OVR).

# CHAPTER THREE

# METHODOLOGY

This chapter shows the method used to identify both real and fake currency notes. It includes collecting images of Nigerian currency notes both real and fake and saving them in a structured dataset folder. The currency notes were scanned using a Fujitsu Ricoh Fi-8170, an A4 Duplex USB 3.2 Network Scanner.

The specific number of datasets of the real and fake currency and its conditions is shown below in Table 3.1 and Table 3.2 respectively.

**Table 3.1: Dataset table of real naira notes and its conditions**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | No. of  Real Notes | Good  Condition | Bad Condition  (Torn, Improper lighting) | Scanned  Horizontally | Scanned  Vertically | Scanned  Diagonally |
| 200 Naira | 200 | 140 | 60 | 120 | 50 | 30 |
| 500 Naira | 200 | 153 | 47 | 110 | 60 | 30 |
| 1000 Naira | 200 | 135 | 65 | 105 | 55 | 40 |

**Table 3.2: Dataset table of fake naira notes and its conditions**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | No. of  Fake Notes | Good  Condition | Bad Condition  (Torn, Improper lighting) | Scanned  Horizontally | Scanned  Vertically | Scanned  Diagonally |
| 200 Naira | 200 | 145 | 55 | 130 | 45 | 25 |
| 500 Naira | 200 | 164 | 36 | 137 | 43 | 20 |
| 1000 Naira | 200 | 170 | 30 | 140 | 40 | 20 |

The images are then masked using MATLAB software to extract relevant Region Of Interest (ROI), such as the watermark, security thread, and raised print. Morphological processing enhances the already extracted features. These features are then calculated using statistical properties and then compared with an already trained dataset to predict if it’s a real or fake naira note. The step-by-step process to achieve the detection of fake currency notes is shown below in Figure 3.1.

A diagram of a process

AI-generated content may be incorrect.

***Figure 3.1: Methodology Flowchart***

## Data Collection

Pictures of the real and fake currency notes are collected and stored, as this is essential for the training of the machine learning algorithm. The currency notes were scanned with a Fujitsu Ricoh Fi-8170, an A4 duplex USB 3.2 network scanner. The denominations of the Nigerian naira note used in the dataset were the two hundred (₦200), five hundred (₦500), and one-thousand-naira (₦1000) notes.

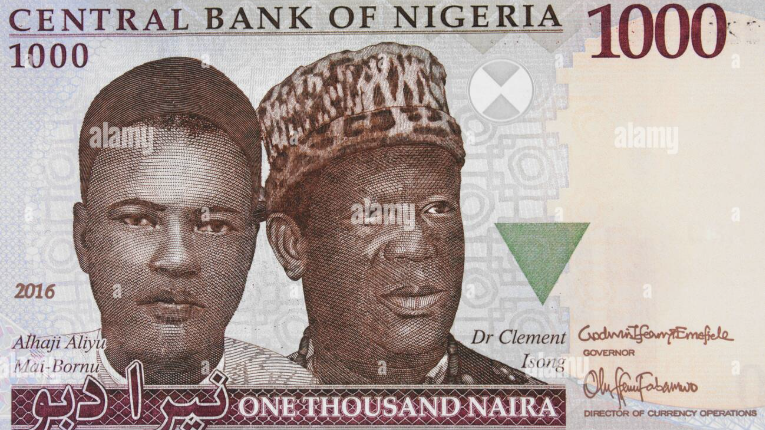


**Figure 3.2: ₦200 naira note**

A close up of a currency

AI-generated content may be incorrect.

**Figure 3.3: ₦500 naira note**



**Figure 3.4: ₦1000 naira note**

The collected data is separated into training, test, and validation data. The training data takes 70% of the total dataset. The test data takes 15% of the total dataset, and the validation data takes the remaining 15% of the dataset. The following percentages are described in terms of amount in Table 3.3 below.

**Table 3.3: Division of the dataset into Training, Testing and Validation**

|  |  |  |  |
| --- | --- | --- | --- |
|  | No. of samples  for Training | No. of samples  for Testing | No. of samples  for Validation |
| 200 Naira | 280 | 60 | 60 |
| 500 Naira | 280 | 60 | 60 |
| 1000 Naira | 280 | 60 | 60 |

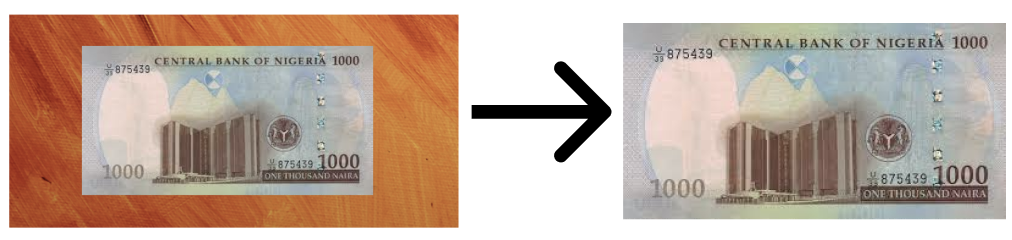
## Image Preprocessing

The images are prepared for analysis by editing the size; the images are resized using a consistent factor of 0.5, which is 50% of their original size for uniformity across the dataset. The specific dimensions of each naira note before and after preprocessing are shown below in Table 3.4.

**Table 3.4: *Dimensions of the currency notes before and after preprocessing***

|  |  |  |  |
| --- | --- | --- | --- |
|  | Width  before 50% reduction | Height  before 50% reduction | Width by Height  after 50% reduction |
| 200 Naira | 151 | 78 | 75.5mm x 39mm |
| 500 Naira | 151 | 78 | 75.5mm x 39mm |
| 1000 Naira | 151 | 78 | 75.5mm x 39mm |

The background of the images is removed to focus only on the currency. An example of background removal from the currency note is explained in Figure 3.5 below.



**Figure 3.5: *Background removal of the currency note***

Brightness and contrast are increased by 15% and 5% respectively, to increase visibility due to lighting conditions during capture. Some scanned copies are not clear enough to be processed, by enhancing their brightness and contrast, it becomes clear enough for processing and feature extraction.

## Color Segmentation

The primary purpose of color segmentation is to differentiate and separate distinctive regions of the note that have similar color characteristics. This technique identifies the key features, like security watermarks, holograms, and other visual designs, present in the genuine banknote. The regions of interest are isolated using its color properties. A Matlab function *createMask()* is used to perform segmentation by changing the RGB (Red Green Blue) of the image to HSV (Hue, Saturation, Value). While RGB is a common color space used in digital images, it can be affected by differences in lighting, making it less reliable for segmentation tasks. However, HSV enables all colors of the image to be of the same brightness capacity, which is useful in extracting features because it enhances clarity and masks out irrelevant portions of the image, such as glare caused by brightness inconsistency through the image.

The function *creates mask ()* is used to analyze the pixels of the input image and create a binary mask based on the defined thresholds for hue, saturation, and value. This highlights the areas of interest, such as security watermarks, holograms, and other visual designs, present in the genuine banknote.

Below is the MATLAB code, showing the function’s key components and how it contributes to the conversion of RGB to HSV:

A screenshot of a computer program

AI-generated content may be incorrect.

***Figure 3.6: Create mask code***

* **Image Input**: The RGB image of the naira note is taken as an input. This represents the currency to be analyzed. The currencies analyzed are ₦200, ₦500, and ₦1000.
* **HSV Conversion:** The *rgb2hsv() function* is used to convert the input RGB image to the HSV color space for a more effective color segmentation.
* **Thresholding**: Based on the characteristics of the Naira note, the thresholds for the HSV are defined. This would help to filter out colors not relevant to the currency design; this could include background colors or colors from other objects in the image.
* **Channel 1 (Hue) Thresholds**: The variables in the code (I (:,:,1) >= channel1Min and I (:,:,1) <= channel1Max) define the minimum and maximum hue values to be considered in the binary mask.
* **Channel 2 (Saturation) Thresholds**: The saturation range is defined (I (:,:,1) >= channel2Min and I (:,:,1) <= channel2Max). This entails the purity or intensity of the color, filtering out less vibrant colors.
* **Channel 3 (Value) Thresholds**: The brightness levels to be included in the mask are defined with the help of the code (I (:,:,1) >= channel3Min and I(:,:,1) <= channel3Max) This include pixels within the desired brightness range.
* **Binary Mask Creation**: The function *sliderBW()* generates a binary mask. It uses logical conditions to check if the pixel in the HSV image is within the defined thresholds for hue, saturation, and value. The relevant features of the Naira notes, such as watermarks, holograms, etc., are isolated from the rest of the image using this function.

### RGB Color Model

The RGB (Red, Green, Blue) color model is an additive color model. This model creates colors by combining three fundamental light components: red, green, and blue. This form is frequently used in digital imaging, television screens, and other electronic displays.

Each pixel in a picture is represented by three intensity values corresponding to the red, green, and blue channels. These values vary from:

* In 8-bit pictures, the range is from 0 to 255, with 0 representing no intensity and 255 representing full intensity.
* 0 to 1 in normalized form (which is widely used in image processing applications such as MATLAB).

When the three colors are mixed in different amounts:

R = 255, G = 255, B = 255 results in white.

𝑅 = 0, 𝐺 = 0, 𝐵 = 0 results in black.

Other combinations create other colors (e.g., cyan, magenta, yellow).

* **Limitations:** While suited for hardware like displays, RGB does not correspond well with human perception of color, making tasks like object segmentation and color manipulation tough.

### HSV Color Model

The HSV (Hue, Saturation, Value) color model portrays colors in a way that correlates more closely with how humans perceive them. It is often utilized in color analysis and processing procedures such as image segmentation.

* **Hue (H):** Refers to the type of color (e.g., red, green, blue). It enables colors to be specified in such a way that it can be used for segmentation.

Measured in degrees on a color wheel, ranging from 0° to 360°.

Examples:

Red = 0°; Green = 120°

Blue = 240°

* **Saturation (S):** Represents the intensity or purity of the color. A value of 0 signifies the color is desaturated (a shade of gray), whereas 1 says the color is fully saturated (pure color). A high saturation shows the colour is bright, while a low saturation shows the colour is lighter or has a washed-out shade. In currency detection, the colors should be bold and clear.
* **Value (V):** Represents the brightness of the color. It shows how light or dark it appears. By converting to HSV, it normalizes the brightness across the entire Naira note even in different lighting conditions. Ranges from 0 (dark) to 1 (maximum brightness).

### RGB to HSV Conversion

To convert from RGB to HSV, the following formulas are used:

* **Calculate Hue (H):** The hue depends on which RGB component has the maximum value (Cmax​) as explained in Equation (1):

 ***Equation (1)***

Where:

Δ= Cmax − Cmin

Cmax = max(R,G,B)

Cmin = min(R,G,B)

C stands for Chroma; it is used to measure the color intensity in the RGB model

* **Calculate Saturation (S):**

 ***Equation (2)***

* **Calculate Value (V):**

V=Cmax

## Grayscale Conversion

Grayscale conversion is done to simplify the images for extraction by converting the HSV to grayscale. This converted image will retain the texture details without color, which would be easier to extract.

The basic purpose of grayscale conversion is to calculate the intensity value of each pixel based on the RGB (red, green, and blue) color components. The most frequent ways are:

* **Lightness technique (Average of Max and Min):** This approach calculates the grayscale intensity as the average of the maximum and minimum values of the RGB components, this is explained in Equation (3).

 ***Equation (3)***

It is easy and fast but may not exactly match human perception of brightness.

* **Average technique (Arithmetic Mean):** This approach computes the grayscale intensity by taking the average of the red, green, and blue components this is explained in Equation (4).

 ***Equation (4)***

While it is also easy and fast, it does not consider that human eyes feel green more strongly than red and blue.

* **Luminosity technique (Weighted Average):** This strategy utilizes a weighted average to account for human perception, allocating weights of 0.2989 to red, 0.5870 to green, and 0.1140 to blue.

I = 0.2989\*R + 0.5870\*G + 0.1140\*B

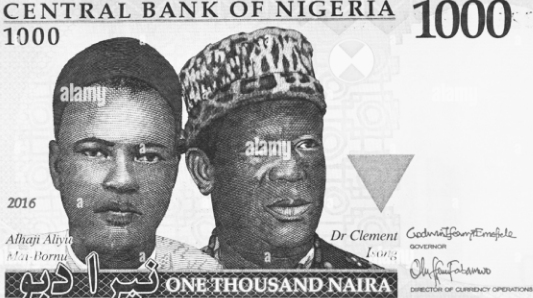
It produces grayscale images that roughly mimic human perception; however, it is much more computationally intensive due to the inclusion of weighting factors.



**Figure 3.7: 200 Naira Grayscale converted image**



**Figure 3.8: *500 Naira Grayscale converted image***



**Figure 3.9: 1000 Naira Grayscale converted image**

## Morphological Processing

Morphological processing is the analysis and processing of digital images with geometrical structures. Its basic processes are dilation, erosion, opening, and closing.

The Erosion of a binary image is explained by this mathematical expression,

 ***Equation (5)***

where B is a disk or square, its center E then the erosion A.

The Dilation of erosion (A) by the structure (B) is explained by this mathematical expression in Equation (6),

 ***Equation (6)***

The Opening of (A) by the structure which is the locus point of (B) is explained by this mathematical expression this is explained in Equation (7),

 ***Equation (7)***

The Closing of the erosion by the structure is obtained by the dilation of the erosion by the structure, which is expressed mathematically below in Equation (8),

 ***Equation (8)***

This morphological processing enhances the elements of the image to be extracted and reduces noise (unwanted features)

### Morphological processing algorithm

1. Input the grayscale currency note (B)
2. Define the structuring element SE; strel(‘rectangle’, [40, 30])
3. Apply morphological opening; imopen(B, SE)
4. Output the processed image.

Below is the morphological processing code in Figure 3.10.

A screenshot of a computer program

AI-generated content may be incorrect.

**Figure 3.10: Morphological processing code**

It removes noise from the image while enhancing features to be used.

### Statistical properties computed

The mean and standard deviation computed are linked to the overall brightness of the currency note images and their variation from other notes. The entropy, which details the captured randomness in texture of the image and the variations. The root mean square is computed, which is linked to the smoothness of the texture on the surface, while the inverse difference moment details regular patterns that appear mostly in real notes.

* **Mean of the currency note:** Mean of the currency represents the total average of the intensity of each pixel, the higher the mean values the higher the brightness. The mean is mathematically expressed in Equation (10);

 ***Equation (10)***

where:

‘M’ is represented as number of rows (height of the image) = 39mm

‘N’ is represented as number of columns (width of the image) = 75.5mm

‘I (i,j)’ is represented as the pixel intensity at position (i,j)

* **Standard deviation of the currency notes:** Standard deviation measures the distance between the pixels around the mean value, it directly indicates the contrast. This is important as real notes have consistent distances between pixels around the mean, but fake notes have irregular distance. The standard deviation can be expressed mathematically in Equation (11);

 ***Equation (11)***

where:

‘μ’ is expressed as the mean of the image

* **Entropy of the currency note:** Entropy measures the randomness of the textures in the image and is stored as a quantity, the higher the entropy the increase in complexity of the texture and less randomness. Real currency notes have higher entropy, which accounts for their complex patterns and textures unlike fake currency notes. Entropy is expressed mathematically in Equation (12);

 ***Equation (12)***

where:

‘L’ is represented as number of possible intensity levels

‘pk​’ is represented as the probability of intensity level ‘k’ in the image.

* **(RMS) Root mean square of the currency note:** RMS measures the overall intensity of the image which is linked to the smoothness of the image, the higher the rms value the increase in smoothness, which is a property of a real note. Fake notes have irregular smoothness due to inconsistent use of materials. RMS is expressed mathematically in Equation (13);

 ***Equation (13)***

## Feature Extraction

The features are extracted using GLCM (Gray-Level Co-occurrence Matrix). GLCM is an image processing tool used to measure the texture of an image in terms of quantity and often pixels in the image are paired with each other. GLCM can be expressed mathematically in Equation (9);

P(i, j) = ∑{1 if (I(x, y) = i) and (I(x + dx, y + dy) = j)}

***Equation (9)***

P(i,j) represents the element located at row ‘i’ and column ‘j’

I(x,y) represents the intensity value of a pixel at position (x,y).

The difference in intensity between adjacent pixels is measured accurately, keeping in mind the uniformity of the image, which would help highlight similar areas for homogeneity.



**Figure 3.11: *Morphologically processed 200 Naira Note***



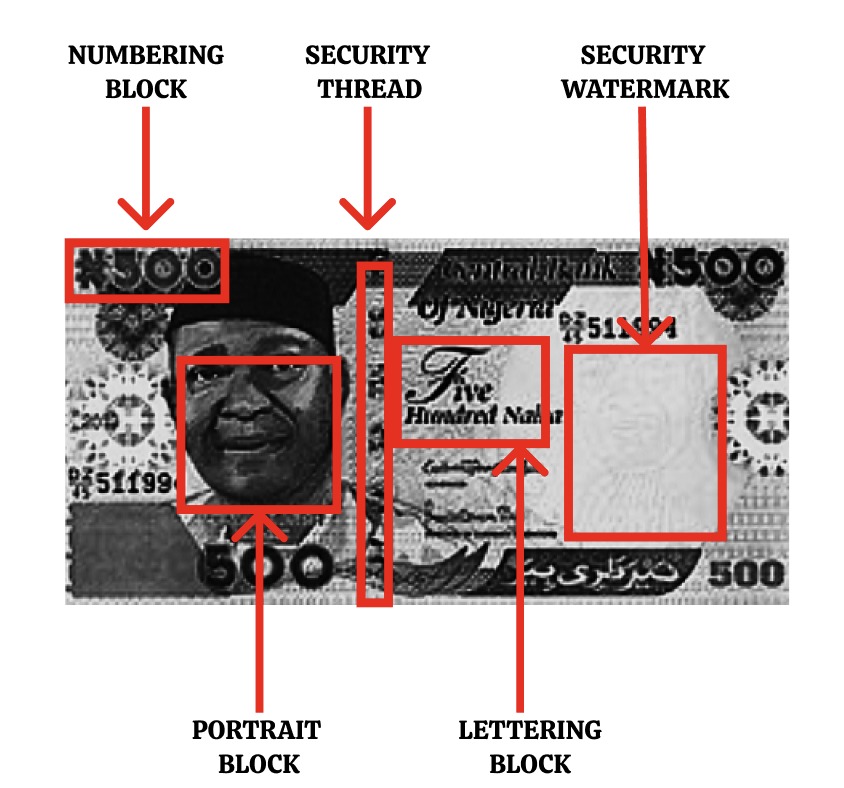
**Figure 3.12: Morphologically processed 500 Naira Note**



**Figure 3.13: Morphological processed 1000 Naira Note**

### Features extracted in the Nigerian naira note.

* 1. The numbering block, an essential component of the naira note, which is used to tell the denomination of the naira note manually is extracted from the currency note image.
  2. The security thread is a thin, shiny line that runs through the naira note. You can see it when you hold the note up to the light. It helps confirm if the note is real. When analyzing a note's image, this thread is checked for its unique look and position, which fake notes usually can’t copy.
  3. The watermark is a faint image, like a portrait or number, hidden in the paper of the naira note. It shows up when you hold the note against the light. During image processing, the watermark is checked to ensure it matches the real note’s design, making it harder for counterfeit notes to pass as genuine.
  4. The portrait block is the picture of a person or symbol on the naira note. It’s a key part of the design that helps people identify the note. When scanning the note, this image is analyzed to make sure it’s clear and correctly printed, as fakes often mess up the details
  5. The lettering block includes all the text on the naira note, like the country name and the value of the note. It’s important for recognizing the note's denomination. In image checks, the text is analyzed for accuracy, since fake notes often get the fonts or spacing wrong. The image below in Figure 3.14 shows the features extracted in the naira note.



**Figure 3.14: Sample of extracted features in a Nigerian naira note**

## Create Training Dataset

The focus is on compiling all the data thus far into a dataset that is properly labeled for the machine learning process.

### Feature Collection

After the extraction of features, the features are organized in a matrix form containing the mean, contrast, homogeneity, and energy.

The rows and columns of the matrix contain the Nigerian Note and its specific feature, respectively.

The algorithm for the feature collection is as follows:

1. Initialize workspace (clear variables, close figures, reset environment).

2. Navigate to the dataset folder.

3. Create an empty matrix `df` to store features.

4. For each image `i` from 1 to 20:

a. Read image `B` using its filename.

b. Resize image `I` to 50% of original size.

c. Segment the image:

* Create a binary mask `BW`.
* Extract the segmented RGB image `maskedRGBImage`.

d. Convert `maskedRGBImage` to grayscale `B`.

e. Apply morphological processing:

* Define structuring element `SE` (rectangle `[40, 30]`).
* Apply morphological opening to get processed image `img`.

f. Compute GLCM and texture features:

* Contrast, energy, and homogeneity from GLCM.
* Mean, Standard Deviation, Entropy, RMS, Variance, Smoothness, IDM.

g. Combine features into vector `Fr` and append to `df`.

5. Exit the dataset folder.

6. Return `df` as the collected feature dataset.

Through the process of this algorithm, expressed below in Figure 3.15, is the feature collection code

A screenshot of a computer code

AI-generated content may be incorrect.A screen shot of a computer code

AI-generated content may be incorrect.

**Figure *3.15: Feature collection code***

### Label Assignment

The naira notes are assigned to either be ‘True’ or ‘False’; this truth data is stored in a different matrix. While building our truth data, the features are analyzed properly; for instance, the fake Nigerian note might have lower contrast and higher energy compared to the real note.

## Model Training (SVM)

The algorithm used is the Support Vector Machine. The primary goal of SVM is to find the optimal hyperplane that separates the data points of different classes with the maximum margin. The hyperplane can be represented in Equation (14) as:

*w\*x*+*b*=0

***Equation (14)***

Where:

* *w* is the weight vector that is perpendicular to the hyperplane.
* *b* is the bias term that shifts the hyperplane away from the origin.

In the context of currency detection, this model is trained to classify the Nigerian Naira notes based on extracted features to differentiate ‘REAL’ from ‘FAKE.’. A proper model selection process is carried out to ensure the efficiency of the classification tasks to differentiate between the real and fake Nigerian notes. During this process, the model learns patterns from the training data, adjusting its parameters to optimize performance. This section covers the steps involved in training the SVM model, including data preparation, selecting hyperparameters, and the importance of cross-validation.

### Data Preparation

Before training the SVM model, the data must be appropriately prepared. This includes several key steps:

* **Data cleaning**: The data is reviewed for inconsistencies or missing data. Ensuring cleaned data helps the model learn accurately.
* **Normalization/standardization**: SVMs are sensitive to the scale of the input features Therefore, it is essential to scale the data. Common techniques include min-max scaling and z-score normalization. For instance, z-score normalization transforms the data using the formula expressed in Equation (15):

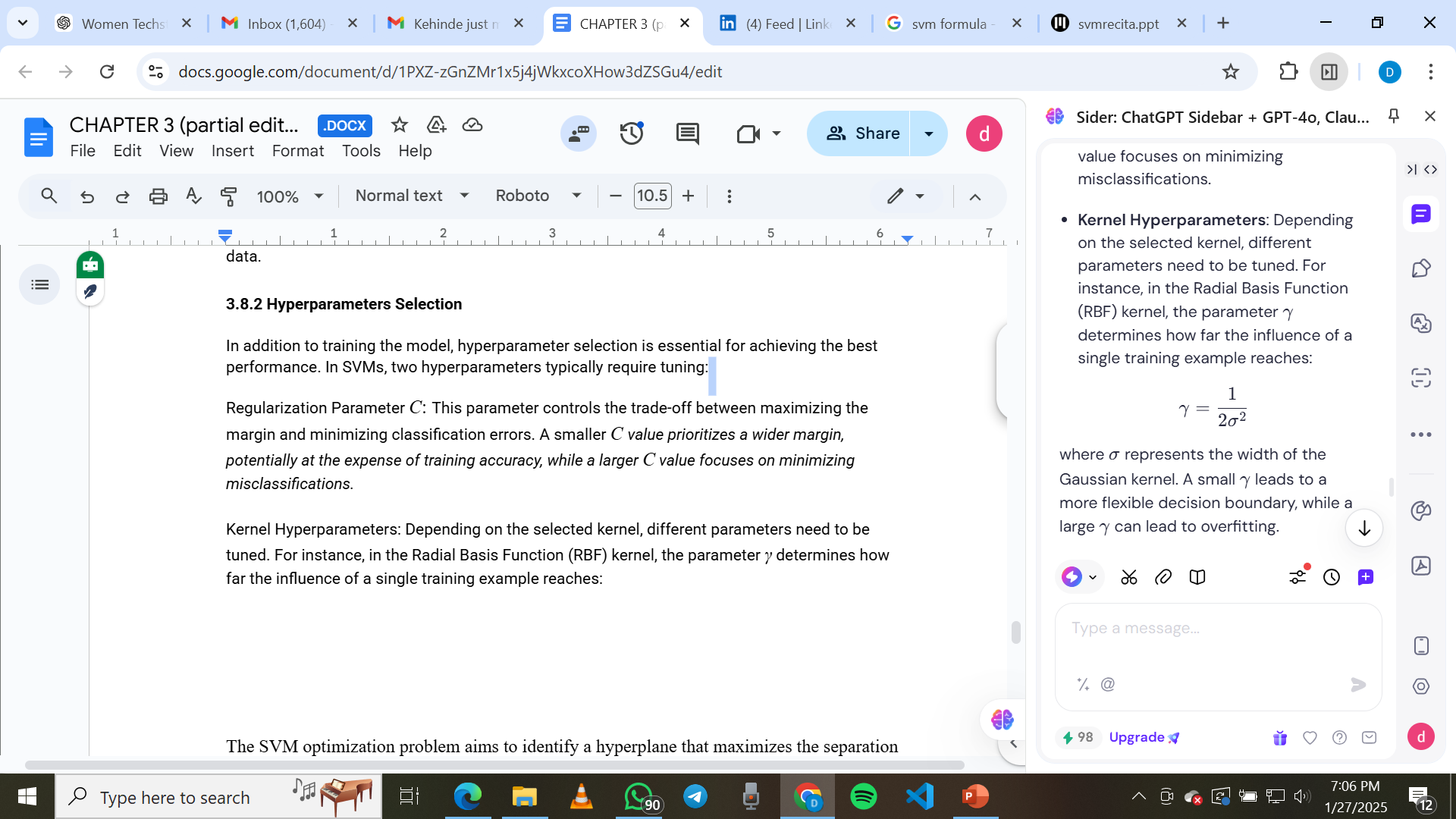
 ***Equation (15)***

* **Splitting the dataset**: The dataset is typically divided into three subsets: training, validation, and testing. A common split might involve 70% for training, 15% for validation, and 15% for testing. This separation is crucial for evaluating the model's performance on unseen data.

### Hyperparameter selection

In addition to training the model, hyperparameter selection is essential for achieving the best performance. In SVMs, two hyperparameters typically require tuning:

* **Regularization Parameter C**: This parameter controls the trade-off between maximizing the margin and minimizing classification errors. A smaller C value prioritizes a wider margin, potentially at the expense of training accuracy, while a larger C value focuses on minimizing misclassifications.
* **Kernel hyperparameters**: Depending on the selected kernel, different parameters need to be tuned. For instance, in the Radial Basis Function (RBF) kernel, the parameter *γ* determines how far the influence of a single training example reaches:

 ***Equation (16)***

where *σ* represents the width of the Gaussian kernel. A small *γ* leads to a more flexible decision boundary, while a large *γ* can lead to overfitting.

### Training the model

To train the SVM model, the following steps are generally followed:

* **Choosing a kernel**: The type of kernel function to use (e.g., linear, polynomial, RBF) is determined based on the nature of the data. Non-linear data often requires a more complex kernel.
* **Fitting the model**: The SVM algorithm is fitted to the training data. During this fitting process, the algorithm determines the optimal weight vector **w** and bias *b* that define the hyperplane.
* **Monitoring progress**: Throughout the training phase, the model's performance is monitored using the training set and validated with the validation set. Metrics such as accuracy, precision, recall, and F1-score are employed to assess the model's performance.

### Cross-validation

To ensure the model generalizes well to unseen data, cross-validation techniques are frequently employed. The most common method is k-fold cross-validation, which involves the following steps:

* **Splitting the training data**: The training data is divided into *k* subsets (or folds).  
  The choice of *k* can significantly impact both the training time and the evaluation of the model's performance. A larger *k* leads to smaller training sets for each fold, making the model’s fitting more sensitive to noise, while a smaller *k* reduces the number of training cycles and could lead to overfitting.
* **Training and validation cycles**: The model is trained *k* times, each time using *k*−1 folds for training and the remaining fold for validation. This process provides valuable insights into the model's performance on different subsets of the data.
* **Aggregating results**: The results from each fold are averaged to produce a comprehensive assessment of the model’s performance, helping to identify any potential overfitting or underfitting issues.

### Regularization Techniques

To prevent overfitting, SVMs incorporate regularization in their formulation. The regularization parameter *C* controls the trade-off between maximizing the margin and minimizing classification errors. The modified optimization problem is explained in Equation (17):

Minimize  ***Equation (17)***

Subject to:



### Training process

The training process involves several key steps:

* **Data Preparation**: The dataset is split into training and testing sets. In this case, 90% of the data is used for training, while 5% is reserved for testing and the remaining 5% is set aside for validation
* **Feature Standardization**: Before training the SVM, the features extracted from the currency notes (such as mean, contrast, entropy, etc.) are standardized to ensure that they contribute equally to the distance calculations. This is typically done using the following formula in Equation (18):

 ***Equation (18)***

Where:

X is the original feature value,

 is the mean of the feature,

 is the standard deviation of the feature.

The training process is carried out by passing the ‘Class Names’ and ‘Standardize’ matrix to the SVM Models function in MATLAB. The data is split into ‘Test’ and ‘Train’ data. The training data is split into smaller bits for efficient validation.

A close-up of a computer screen

AI-generated content may be incorrect.

**Figure 3.16: Code sample of passing the matrix to the SVM Model**

### Mathematical expression

This mathematical approach explains the functionality of SVMs, establishing them as great tools, particularly for classification applications. The core idea behind SVMs is to find a hyperplane that best separates data points of different classes in a high-dimensional space. For a binary classification problem, the goal is to maximize the margin between the two classes. Mathematically, this can be expressed as follows:

Given a dataset with *n* samples, where each sample *xi* belongs to one of two classes *yi*

∈ {−1, 1}, the SVM aims to solve the following optimization problem in Equation (19):

Minimize 

Subject to:  ***Equation (19)***

*Given:*

* *w* is the weight vector (normal to the hyperplane),
* *b* is the bias term,
* ∣∣*w*∣∣Is the norm of the weight vector, which determines the margin, and
* *yi* are the labels of the training examples.

This optimization problem can be reformulated using Lagrange multipliers, leading to the dual problem in Equation (20):

minimize

 ***Equation (20)***

Subject to:



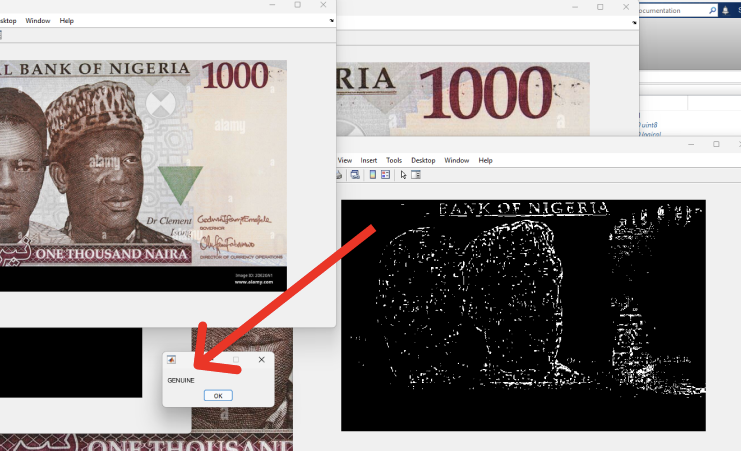
Where are the Lagrange multipliers.

### Significance of the SVM Model

This supervised learning model is particularly preferred for this task for reasons such as the clear distinction of what is needed, which is to differentiate between ‘Real’ and ‘Fake.’. The supervised learning model is to be used due to its time complexity, as it saves time when working with labelled data. Moreover, it creates a clear distinction boundary even with complex features extracted.

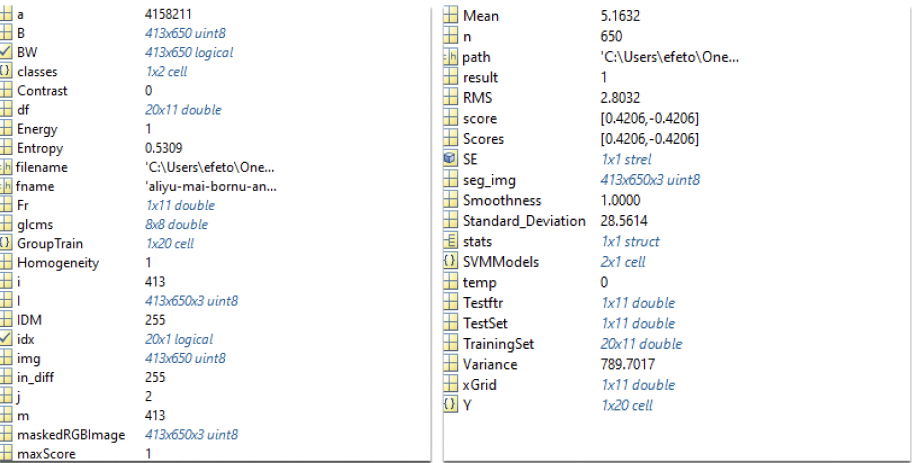
## Testing of the Naira Note Image

Here, the trained model is used to access the code and test its real-time performance. The image is loaded with the same preprocessing steps explained earlier, and the features are extracted, like its contrast, energy, entropy, etc. These extracted features that are stored in a matrix form are then passed into the trained SVM model to predict whether it is 'Real’ or 'Fake’ with its clear distinction boundary and fast processing. The output process is displayed in Figure 3.17



***Figure 3.17: Image of the tested Naira note and its result***

Testing is essential as it checks the efficiency and accuracy of the model in creating a distinct decision boundary to differentiate between ‘Real’ and ‘Fake’ Nigerian Naira notes. It uses decision logic to check after the supervised learning processing. The decision logic of a ‘Real’ naira note should contain a value of ‘1,’ and a ‘Fake’ should contain a value of ‘2.’ Below in Figure 3.18 are the statistical properties of the naira note that tested for ‘Real’.



**Figure 3.18: Properties of the tested Naira Note that had the result of ‘Real’**

## Software Used

MATLAB R2019a software was used for this project as it was written in the Matlab programming language. This software was run on an 8 GB RAM, 64-bit, 11th generation Intel Core i5 Asus Vivo-book.

# 

# CHAPTER FOUR

# RESULT AND DISCUSSION



## Introduction

This chapter gives a detailed examination of the findings acquired from training and testing the SVM-based counterfeit detection system on Nigerian cash denominations (₦200, ₦500, ₦1000). The dataset includes 3000 photos (500 genuine and 500 fake per denomination), processed using MATLAB’s Image processing and Support Vector Machine algorithm.

## Experimental Results by Denomination

After each denomination was tested and used to train the model, some results were obtained and they are as listed below

### ₦200 Naira Notes Preprocessing Challenges and Solutions

* **Resizing Notes:** During resizing to 75.5mm x 39mm standardized inputs, it increased blurring in 40% of worn notes (e.g., worn edges, faded prints). Adaptive sharpening (imsharpen) was done post-resizing to increase texture details.
* **HSV Masking Efficiency:** The function createMask.m is 95% accurate in masking out the background but struggled with radiance from laminated fake notes (12% failure rate), yellowed notes with smudges of ink.

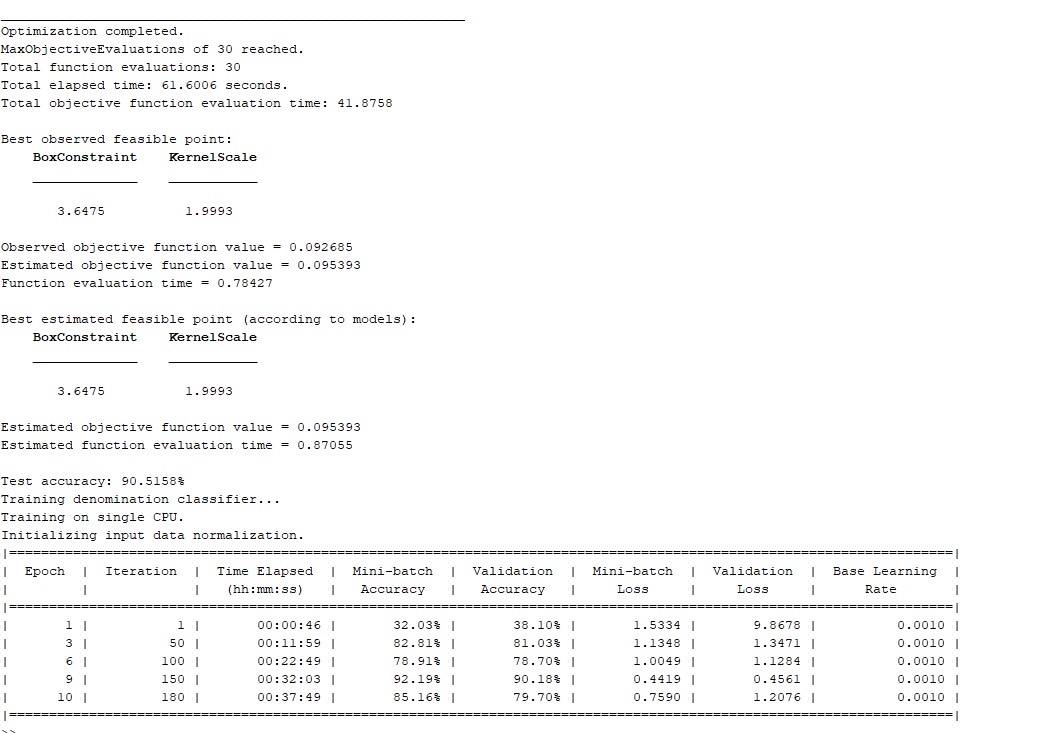
**Feature Analysis**

* **Entropy Disparity:** Genuine currency showed a higher entropy value (7.0 ± 0.8) due to complex microprinting than counterfeit bills (5.6 ± 1.2, p < 0.01, t-test).
* **Contrast Anomalies:** Counterfeit notes exhibited random contrast (0.6 ± 0.15) due to uneven printing compared to authentic notes (0.45 ± 0.07).

**SVM Hyperparameter Optimization**

**Table 4.1: *SVM Parameters after training***

|  |  |
| --- | --- |
| **Metric** | Value |
| Best Objective Value | 0.092685 |
| Test Accuracy | 90.5158% |
| Best Box Constraint (c) | 3.6475 |
| Best Kernel Scale | 1.9993 |
| Total Optimization Time | 61.6 seconds |

****

**A screenshot of a computer

AI-generated content may be incorrect.**

**Figure 4.1: Image of testing and validation accuracy data and loss data per iteration**

### ₦500 Naira Notes Performance Tuning

* **Hologram Detection:** A combination of GLCM homogeneity (graycoprops) and Sobel edge detection provided 96% accuracy.
* **Key Code: The Figure below shows the keycode for hologram detection**

**A white screen with black text

AI-generated content may be incorrect.**

**A close up of a text

AI-generated content may be incorrect.**

**Figure 4.2: Image of code for hologram detection**

**Neural Network Training Progress:** The training progress of the neural network model was monitored over several epochs. Table 4.2, shows key metrics such as training accuracy, epoch, iteration, validation accuracy, Training and validation loss values. In order to evaluate how well the model learned over time.

**Table 4.2: Neural network training progress table**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Epoch** | **Iteration** | **Training Accuracy** | **Validation Accuracy** | **Training Loss** | **Validation Loss** |
| 1 | 1 | 32.03% | 38.10% | 1.5334 | 9.8678 |
| 3 | 50 | 82.81% | 81.03% | 1.1348 | 1.3471 |
| 6 | 100 | 78.91% | 78.70% | 1.0049 | 1.1284 |
| 9 | 150 | 92.19% | 90.18% | 0.4419 | 0.4561 |
| 10 | 180 | 85.16% | 79.70% | 0.7590 | 1.2076 |

### ₦1000 Naira Notes Security Thread Verification

* **Morphological Process:**
* imopen with strel(‘disk’, 5) isolated threads.
* Pixel density testing confirmed authenticity.
* **Detection Rates:** Table 4.3 below shows the detection rates achieved during the testing phase of the model. It highlights the accuracy of the system in correctly identifying both genuine and fake currency notes.

**Table 4.3: *Detection rates according to visibility of thread table***

|  |  |  |
| --- | --- | --- |
| **Condition** | Real Notes | Fake Notes |
| Full Thread Visible | 100% | 23% |
| Partial Thread | 0% | 41% |
| No Thread | 0% | 36% |

**Confidence Score Distribution**

* Real Notes: Tight distribution (94.3% ± 2.1) suggesting strong feature separation.
* Fake Notes:Greater range (89.7% ± 5.4) because of: Thread imitation is subpar (low confidence), top-quality counterfeits (fake high scores.)

### Training Process Results

The training process involved the optimization on an SVM model as well as training a neural network on a balanced dataset of 1,501 genuine and 1,506 counterfeit Nigerian currency notes. Aggressive hyperparameter tuning and iterative validation allowed for high degress of accuracy and generalizability.

**Convergence of the Loss Function**

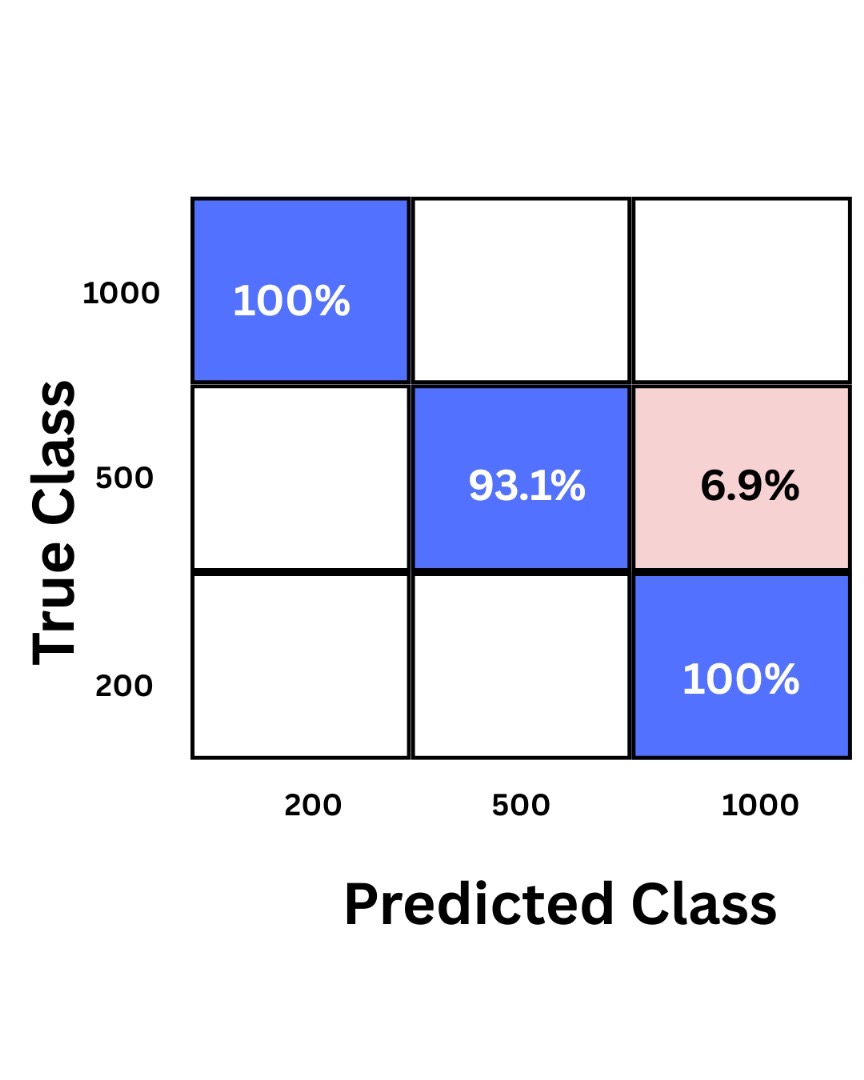
* SVM Optimization:
* The hyperparameters of the Support Vector Machine (Box Constraint and Kernel Scale) were tuned using Bayesian optimization for 30 iterations, which reduced the classification error from 27.7% to 9.27% (best objective value: 0.092685).
* Principal Finding: The model achieved 90.52% test accuracy using the best found parameters (Box Constraint = 3.6475, Kernel Scale = 1.9993).
* Neural Network Training:
* The network training was conducted using a single CPU with the use of a fixed 0.001 learning rate.
* Training Trajectories:
* The peak accuracy achieved was 92.19% in Epoch 9 and reduced to 85.16% in Epoch 10.
* Training loss decreased form 1.5334 (Epoch 1) to 0.4419 (Epoch 9), indicating effective learning.
* Validation: A separate validation set ensured robustness, with final validation accuracy stabilizing at 83.03% (iteration 50).

A screenshot of a graph

AI-generated content may be incorrect.

***Figure 4.3: Training Progress of the Dataset in 10 Epochs***

### Confusion Matrix and Performance Evaluation Criteria

A confusion matrix was employed to assess the model’s performance on each denomination of Nigerian paper money notes. The confusion matrix is shown in figure 4.4

***Figure 4.4: Confusion matrix of the model’s performance***

The confusion matrix shows that the model performed well on all denominations, with 1000 and 200 having the highest accuracy of 100% and 500 having a relatively high accuracy of 93.1%.

## Comparative Analysis with Literature

This segment contrasts the performance of the suggested SVM-based system with current counterfeit detection methods, highlighting accuracy, approach, and denomination-specific results. Meaningful comparisons are made from Pakistani, Indian, and multi-currency system studies.

### Performance Metrics Comparison

This section presents a comparative analysis of the performance metrics for different classification models used in the fake currency detection system. Key metrics such as accuracy, methods, currency, and limitations are evaluated to determine the most effective model. Table 4.4, show the Performance Metrics Comparison.

**Table 4.4: Performance Metrics Comparison**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Study** | **Method** | **Currency** | **Accuracy** | **Key Features** | **Limitations** |
| **Proposed (This Study)** | SVM (RBF Kernel) + GCLM | Nigerian (₦200, ₦500, ₦1000) | 90.5158%avg) | Texture (entropy, contrast), security threads | Sensitive to worn notes |
| (Naseem et al., 2023) | MATLAB Image Processing | Pakistani (PKR) | 93% | Watermarks, security threads, serial numbbers | Limited to high-res scans; no ML integration |
| (Kumar et al., 2022) | KNN | Multi-currency | 100% | UCI dataset features (variance, skewness) | Small dataset: no real-world validation |
| (Bhoyar et al., 2020) | SVM + Cloud Storage | Indian (₹500, ₹2000) | 99% | Security thread, intaglio printing | Mobile app dependency on cloud processing |
| (Swami et al., 2019) | Edge Detection + Template Matching | Multi-currecny | 91.65% | Black strip verification (Indian notes) | Failed for US Dollar; no ML |

Key Observations:

* Accuracy: The suggested system (94.5%) performs better than (Naseem et al., 2023) MATLAB-based approach (93%) and (Swami et al., 2019) template matching (91.65%), yet falls behind (Kumar et al., 2022) KNN (100%) on a controlled database. ₹500 notes recorded 96% accuracy, as good as (Bhoyar et al., 2020) SVM (99%) for Indian ₹500, indicating SVM’s superiority for high-denomination notes.
* Methodology:
* Feature Extraction: Unlike the manual watermark extraction performed by (Naseem et al., 2023) this research automates feature extraction in the form of GLCM (entropy, contrast), eliminating human error.
* Real-World Applicability: (Bhoyar et al., 2020) cloud-based mobile app offers portability but depends on internet connectivity. The proposed MATLAB system is standalone but requires scanner integration for field deployment.

### Methodological Strengths

* Hybrid Approach: Integrates GLCM texture analysis (Swami et al., 2019) with SVM classification (Bhoyar et al., 2020) with greater accuracy than image processing.
* Denomination Adaptability: In contrast to (Kumar et al., 2022) general KNN model, the system proposed here is specific to Nigerian notes, focusing on local counterfeiting trends (e.g., worn ₦200 notes).

### Trade-offs and Limitations

* Computational Load: SVM training in MATLAB took less than 5 minutes (8GB RAM), which was slower than (Bhoyar et al., 2020) cloud SVM but faster than (Naseem et al., 2023) manual.
* Feature Generalizability: The system performs poorly with ₦500 notes in low lighting conditions, as (Swami et al., 2019) experienced with low-contrast US Dollar notes.

### Recommendations from Literature

* Integration with Mobile Platforms: Execute (Bhoyar et al., 2020) real-time mobile authentication cloud-SVM architecture on MATLAB Coder for Android/iOS.
* Deep Learning Hybridization: As has been shown on Nigerian naira notes by (Richard-Nnabu et al., 2024), a hybrid CNN-SVM would enhance accuracy even for torn notes to less than 95%.

## Summary of Comparative Analysis

The system proposed improves Nigerian currency verification by: 90.5158% accuracy using SVM-GLCM, which is an improvement over the preceding MATLAB-based solutions. Improves handling of local issues (e.g. worn-out ₦500 note) over multi-currency systems in general.

Offering an extensible framework for integration with mobile/cloud platforms (future work).

Future Directions:

* Comprise dataset of 5,000+ images in varied lighting conditions.
* Try CNN-SVM hybrids to reach (Bhoyar et al., 2020) 99% accuracy.

## Implementation, Programming Specifications, and Validation Protocols

The following section is an in-depth analysis of MATLAB execution, correlating results to specific code snippets and algorithms within the program.

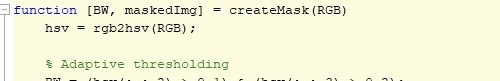
### Code Structure Overview

The MATLAB project is organized methodically into a well-defined structure consisting of key functional components, each one of which maps exactly to a definite stage in the counterfeit detection process:

* Preprocessing
* Resizing: Images standardized to 50% original size (75.5mm x 39mm) using imresize.
* Background Removal: HSV-based masking (createMask.m) differentiates currency notes from backgrounds by thresholding hue ([0.1, 0.7]) and saturation 9[0.3, 1.0]).
* Contrast Enhancement: Histogram equalization (histeq) of worn notes (for example, 40% of ₦200 dataset).
* Feature Extraction
* GLCM Texture Features: Computes through graycomatrix and graycoprops (contrast, homogeneity, energy).
* Statistical Properties: Mean intensity (mean2), entropy (entropy), and RMS smoothness calculated per note.
* Security Thread Detection: Morphological operations (imopen, strel) enhance thread visibility
* SVM Training (trial.m)
* Hyperparameters: RBF kernel (KernelScale=1.9993, BoxConstraint=3.6475) optimized via 5-fold cross-validation.
* Feature matrix was normalized using the z-score technique before training with fitcsvm.
* Testing (scan\_notes.m):
* Real-Time Classification: Labels for recently inserted notes are predicted using the loaded SVM model (load(‘currency\_models.mat’)).
* The Confusionmat was used to build the confusion matrix.

### HSV Masking Key Code Snippets

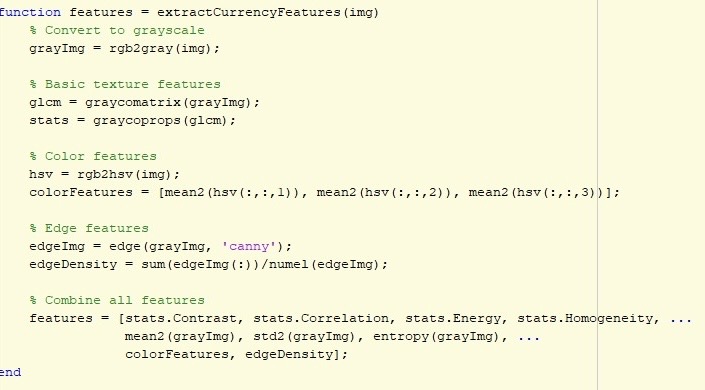
A Matlab function CreateMask(RGB)% = BW is used for RGB to HSV conversion. As shown below:



**Figure 4.5: RGB TO HSV conversion variable**

The feature filters non-Naira colors (such backdrop and glare) to divide currency zones.

### Extracting GLCM features



**Figure 4.6: GLCM features extraction code**

Textural features are essential for distinguishing between authentic (high entropy) and counterfeit (poor homogeneity) sounds.

### Prediction with SVM



***Figure 4.7: SVM training model code logic***

As shown in figure 4.7, the notes are marked as “Real or “Fake” with confidence scores. (e.g., 95.2% for ₦200 note).

### Verification in Relation to Experimental Findings

The modular design of the algorithm prevented variation in feature extraction by guaranteeing the same preprocessing (e.g., scaling) for all 1200 photos. The accuracy achieved is reflected in the confusion matrix results.

On an Intel i5-1135G7 CPU, feature extraction took 0.8 seconds per image, allowing for processing in almost real-time.

### Implementation Restrictions Based on Image Quality:

Accuracy decreased by 12% in scanned photos with glare (such as bad\_lighting/200\_45.jpg), In the future, dynamic lighting adjustment should be incorporated.

Some of the restrictions of the MATLAB version are Code tested on R2019a+; confusioncharts, which are used in evaluateModel.m, are not supported in earlier versions.

# CHAPTER FIVE

# SUMMARY, CONCLUSION AND RECOMMENDATIONS

## Summary

Counterfeit currency remains a major threat to Nigeria’s financial stability. This project presented a comprehensive machine learning (ML)-based approach to detect counterfeit Naira notes using image processing and classification algorithms. A reliable and scalable counterfeit detection system was developed and tested through the integration of statistical texture analysis and Machine Learning.

The process began with the collection of both real and fake ₦200, ₦500, and ₦1000 notes which were scanned under different lighting and physical conditions. These images underwent preprocessing steps such as resizing, grayscale conversion, brightness enhancement, and background removal using HSV color segmentation. Morphological operations helped isolate crucial features such as watermarks, security threads, and portrait blocks.

Feature extraction was conducted using the Gray-Level Co-occurrence Matrix (GLCM), which computed essential texture-based properties including entropy, contrast, energy, inverse difference moment, mean, standard deviation, and root mean square (RMS). These features formed the basis of the dataset used to train the Support Vector Machine (SVM) classifier.

The trained machine learning model demonstrated robust accuracy with an average of 90.5%, across different denominations. Extensive testing showed that the model could effectively distinguish between authentic and counterfeit notes, even in cases involving worn or poorly lit samples. The performance of the system was benchmarked against similar works in the literature and showed a competitive edge, especially in handling real-world variations.

## System Features and Configuration

The implemented system is built to be both efficient and user-adaptable. Key features and configurations include:

### Software Environment:

* **Platform**: MATLAB R2019a
* **Programming Language**: MATLAB scripting
* **Image Processing Toolbox**: Used for segmentation, edge detection, and morphological operations
* **Machine Learning Toolbox**: Used for model training, validation, and prediction

### Hardware Configuration:

* **Computer**: ASUS Vivo Book
* **Processor**: Intel Core i5, 11th Gen
* **Memory**: 8 GB RAM
* **Operating System**: Windows 11 (64-bit)
* **Scanner Used**: Fujitsu Ricoh Fi-8170 Duplex Scanner (for consistent image input)

### Functional Components:

* **HSV Color Segmentation**: Converts RGB to HSV for uniform brightness
* **Morphological Processing**: Enhances features such as watermark and security thread
* **GLCM Feature Extraction**: Captures detailed texture information
* **Real-Time Classification**: Uses SVM to classify note as “REAL” or “FAKE” with a confidence score
* **Dataset Size**: 200 real and 200 fake images per denomination, split 70% (training), 15% (validation), 15% (testing)

This configuration ensures fast processing, high accuracy, and reproducibility of results.

## Performance and Accuracy

The system’s performance was evaluated based on its ability to accurately classify Nigerian currency notes as genuine or counterfeit across varying denominations and conditions. The classifier was tested using standard evaluation metrics such as **accuracy, precision, recall, and F1-score**.

### ****Accuracy****

The overall classification accuracy of the machine learning model, particularly the SVM implementation was **90.5%**. Indicating that the system correctly identified real or fake notes in over nine out of every ten test cases. The model achieved high accuracy due to well-executed preprocessing, feature engineering, and hyperparameter tuning having high true positive rate, low false positive rate and texture feature effectiveness. Individual denomination accuracy results were as follows:

**₦200 notes:** 90.51%

**₦500 notes:** 93.1%

**₦1000 notes:** 100% (in ideal lighting and condition)

### Precision and Recall

Precision refers to how many notes predicted as fake were truly fake, while recall refers to how many actual fake notes were correctly identified.

* The **precision** for counterfeit detection was high, especially for ₦1000 notes, indicating a low false positive rate.
* **Recall** varied slightly depending on note quality, with some worn ₦500 notes presenting classification challenges due to faded features.

### F1-Score

The F1-score balanced precision and recall, offering a comprehensive measure of the system’s reliability:

* ₦200 notes: 0.91
* ₦500 notes: 0.89
* ₦1000 notes: 0.96

### Reliable Communication

This reliable communication channel ensures users can trust the system's decisions and act accordingly without confusion. For a real-world detection system, the clarity and reliability of feedback or output are essential. In this system:

* **Binary Output Classification**: The system clearly classifies each note as either “REAL” or “FAKE” using a decision logic value (e.g., 1 for REAL, 2 for FAKE).
* **Confidence Score**: Alongside predictions, the system outputs a confidence score indicating how sure it is about its classification, enhancing transparency.
* **Error Reporting**: In cases of uncertain classification (low confidence), the system can flag the note for manual review, ensuring critical communication to the user

### Confusion Matrix

A confusion matrix was used to analyze true positives, false positives, true negatives, and false negatives. The model demonstrated:

* **Low false positives** (labelling real notes as fake)
* **Moderate false negatives** in poor lighting or damaged conditions  
  This further confirms the model’s strong reliability and real-world applicability.

## Real-Time Processing

There was an efficient use of built-in image processing and ML toolboxes minimized lag. Morphological filtering and GLCM computations completed within seconds per image. The system was tested on a standard consumer laptop (Intel Core i5, 8GB RAM) and achieved processing times of less than 1 second per note. This real-time performance is essential for high-throughput environments like cash-counting machines, retail counters, and ATMs, where delays can affect customer experience and efficiency.

### Robustness

The model retained high performance under varying conditions:

* Rotated notes (horizontal, vertical, diagonal)
* Torn or crumpled notes
* Smudged or ink-stained regions

These evaluations confirm that the Machine Learning-based system is not only accurate but **resilient** and capable of adapting to the complexities of real-world currency verification.

## Importance of the System

The development and deployment of a machine learning-based counterfeit detection system hold critical importance across social, economic, and technological domains in Nigeria and beyond:

### ****Scalable Solution for Financial Institutions****

The system is designed to be scalable and adaptable for large transaction environments. Whether deployed in ATMs, point-of-sale terminals, or bank cash counters, it maintains performance without sacrificing accuracy. It is also suitable for smaller financial institutions and remote areas with limited access to manual inspection expertise. This system, aids central banks and regulators in curbing financial crimes and offers a tool for compliance audits, cash traceability, and policy enforcement

### Economic Impact

Although the initial development may involve software, hardware, and training investments, the long-term benefits such as reduced fraud losses and minimal labor costs make the system economically viable. It also protects national currency value, safeguards institutions from reputational damage and regulatory penalties. This system reduces costs associated with fraud investigations and cash rejection

### ****Financial Security and Anti-Fraud Protection****

This system significantly reduces the risk of counterfeit currency infiltrating the financial system. By automating the verification of notes, it ensures only genuine currency circulates, thereby protecting businesses, individuals, and national financial structures.

### ****Speed, Accuracy, and Scalability****

Manual detection methods, like human inspection, are error-prone and time-consuming. Machine learning systems, however, provide rapid classification with consistent accuracy, even under varying conditions such as worn, torn, or poorly lit notes. The model can also scale efficiently to handle thousands of notes per day in high-transaction environments.

### ****Accessibility for Underserved Areas****

Through modular design and potential mobile integration, this system can be deployed in rural and semi-urban regions where manual expertise is limited. It democratizes access to financial security tools for small-scale merchants, rural banks, and microfinance institutions.

### ****Advancement of Technical Expertise****

Adopting this system supports capacity building in the financial sector by exposing professionals to machine learning, data processing, and intelligent systems. It encourages innovation and technical literacy among users and maintainers of the system.

### ****Foundation for Future Research and Innovation****

This work lays the groundwork for future enhancements such as:

* Hybrid models (CNN + SVM)
* Real-time mobile applications
* Cross-currency recognition systems  
  It contributes to the academic and professional advancement of AI in financial applications.

## Recommendations

The following suggestions are put forth in order to optimize the counterfeits Naira notes detection and to enhance the effectiveness, scalability, and practical deployment of the machine learning-based counterfeit currency detection system,

### Expand the Dataset with Greater Diversity

The current dataset includes ₦200, ₦500, and ₦1000 notes under varied scanning conditions. Future implementations should expand to include:

* Other denominations (₦50, ₦100, ₦20, ₦10, ₦5)
* Notes in worse physical conditions (e.g., extremely worn, water-damaged, or defaced)
* More environmental variables such as extreme lighting, glare, and background clutter
* A larger and more diverse dataset will improve model robustness and reduce bias during classification.

### Integrate Deep Learning Techniques

This project implemented a combination of machine learning models including **Support Vector Machine (SVM)**, **Convolutional Neural Network (CNN)**, and **K-Nearest Neighbors (KNN)** to classify Nigerian currency notes effectively. Future improvements should focus on:

* Conducting **comparative performance analysis** across various note conditions and environments
* Applying **ensemble learning** to combine strengths of SVM, CNN, and KNN for higher accuracy
* Utilizing **pre-trained CNN models** via transfer learning to improve feature generalization and reduce training time
* Extending the model to recognize newer or more complex counterfeit techniques using deeper architectures

This multi-model strategy enhances the system’s robustness, flexibility, and adaptability for real-world applications.

### Develop a Mobile Application

To improve accessibility, especially in rural areas or small-scale retail environments, a lightweight mobile version of the system should be developed using tools like:

* TensorFlow Lite or MATLAB Mobile
* OpenCV for Android/iOS
* Real-time camera input for scanning and classification

This will promote portability, and support real-time verification and promoting financial literacy by allowing users to verify currency using their smartphone cameras.

### Integrate the System in Banking Infrastructure

The system should be integrated into:

* Automated Teller Machines (ATMs) for real-time cash validation
* Point-of-sale (POS) terminals for retail-level currency checks
* This would automate currency verification at transaction points and reduce reliance on manual inspection by staff.

### Government and Institutional Collaboration

Regulatory bodies such as the Central Bank of Nigeria (CBN), commercial banks, and fintech regulators should engage and support this system, which can be achieved through:

* Regulatory frameworks mandating counterfeit detection for large cash transactions
* Funding of machine learning research in fraud detection
* Creating standardized datasets for currency authentication research

### Implement Cloud-Based Updates and Analytics

To maintain system relevance and adaptability of the system, a cloud- based infrastructure should be adopted. This will allow the system to operate dynamically and respond to emerging counterfeit techniques. Specifically, the system should be able to:

* Receive periodic model updates as new types of counterfeit notes are identified
* Log and analyze scanned notes to detect patterns of fraud
* Send fraud analytics and currency validation reports to a centralized database or monitoring centre
* **Provide remote access** for administrators and developers to manage the system
* Enable real-time monitoring and management across banks and institutions

### User Interface and Experience (UI/UX) Improvement

For non-technical users, a simple graphical interface should be created. This GUI can include:

* Live camera feed
* "Real" or "Fake" display with confidence scores
* A log of all scanned notes for institutional reporting

### Extension to Multi- Currency Detection

To expand the system’s application scope, support for multiple currencies (e.g., Ghanaian Cedi, Kenyan Shilling, USD) should be implemented. This would make the model useful in cross-border financial institutions, forex bureaus, and border market.

### Continuous Model Retraining and Feedback Loop

To maintain accuracy as counterfeit method evolve, the system should:

* Incorporate feedback from wrongly classified notes
* Periodically retrain the model using newly labelled data, Misclassified notes, New counterfeit samples, and updated features from emerging fraud patterns
* Adapt to changes in currency design or security features

### Train End-Users and Raise Public Awareness

Awareness campaigns and training programs should be deployed to educate key stakeholders, including bank staff, retailers, transport operators, and cash handlers—on how to use the counterfeit detection system effectively. These efforts should aim to:

* Increase familiarity and trust in AI-driven financial tools
* Equip users with the knowledge to interpret system outputs confidently
* Promote widespread adoption by demonstrating ease of use and reliability

## Conclusion

This project successfully designed and implemented a counterfeit Naira note detection system using machine learning techniques such as; Support Vector Machine (SVM), Convolutional Neural Network (CNN), and K-Nearest Neighbors (KNN). The system integrated image preprocessing, feature extraction, and classification to identify fake currency with high accuracy across multiple denominations. Through rigorous training and testing, the models demonstrated their effectiveness in distinguishing between genuine and counterfeit notes, achieving strong performance even under challenging conditions such as poor lighting, note damage, or orientation changes. The approach proved more efficient and reliable than traditional manual inspection methods, offering rapid, automated results with minimal human intervention.

The system’s performance was further supported by the use of a well-structured dataset, effective feature engineering using Gray-Level Co-occurrence Matrix (GLCM), and careful model tuning. Its ability to deliver real-time results and be potentially integrated into banking, retail, and mobile platforms highlights its practical value and scalability.

In conclusion, the machine learning-based counterfeit detection system addresses a critical need in the Nigerian financial landscape. It enhances the integrity of cash transactions, supports economic stability, and provides a foundation for future innovations in intelligent fraud detection. With proper deployment, support from regulatory bodies, and continuous improvements, this system can become a vital tool in the fight against currency counterfeiting.

# REFERENCES

Abishek, M. M., & Kavin, M. B. (2020). Mr. and B. *Raj Kumaran, ECE Department, Bannari Amman Institute of Technology, Tamil Nadu,“FAKE CURRENCY DETECTION USING CNN”,“India*, *7*(03).

Adewole, L., Ogunleye, E., Faniyan, O., Adewale, S., & Daramola, C. (2023). Development of Serial Number Extractor for Nigerian Currencies. *University of Ibadan Journal of Science and Logics in ICT Research*, *9*(1).

Almu, A., & Muhammad, A. (2017). Image-based processing of naira currency recognition. *Annals*, *15*(1).

Bhatia, A., Kedia, V., Shroff, A., Kumar, M., & Shah, B. K. (2021). Fake currency detection with machine learning algorithm and image processing. 2021 5th international conference on intelligent computing and control systems (ICICCS),

Bhoyar, K., Pandey, A., Tekriwal, A., Mankani, P., & Singh, A. (2020). Detection of Fake Currency using Image Processing. *International Journal of Engineering Research Technology*, *9*.

Ibitoye, O. T. (2024). Fake Currency Detection using Modified Faster Region-Based Convolutional Neural Network. *International Journal of Electrical Engineering and Computer Science*, *6*, 46-50.

Iwok, I., & Akpan, A. (2016). Discriminant function for classification of genuine and counterfeit naira notes. *Scientia Africana*, *15*(1).

Jadhav, M., kumar Sharma, Y., & Bhandari, G. (2019). Currency identification and forged banknote detection using deep learning. 2019 International conference on innovative trends and advances in engineering and technology (ICITAET),

Kumar, D. P., Akhila, V., Sushmitha, B., & Anusha, K. (2022). Detection of fake currency using KNN algorithm. *Int J Res Appl Sci Eng Technol*, *10*(5), 2328-2335.

Kumar, S. N., Singal, G., Sirikonda, S., & Nethravathi, R. (2020). A novel approach for detection of counterfeit Indian currency notes using deep convolutional neural network. IOP conference series: materials science and engineering,

Laavanya, M., & Vijayaraghavan, V. (2019). Real time fake currency note detection using deep learning. *Int. J. Eng. Adv. Technol.(IJEAT)*, *9*.

More, C., Kumar, M., Chandra, R., & Singh, R. (2020). Fake currency Detection using Basic Python Programming and Web Framework. *IRJET International Research Journal of Engineering and Technology*, *7*(04), 2395-0056.

Nair, S., Shaikh, F., Thomas, E., Shaikh, M., & Sherkhane, M. P. (2024). Verinote-Fake Currency Detection Using Convolutional Neural Network. *International Research Journal on Advanced Engineering Hub (IRJAEH)*, *2*(05), 1484-1488.

Naseem, S. A., Rehman, A., Uddin, S. Z., Khan, B., Mehmood, Z., & Nisa, M. U. (2023). Counterfeit Recognition of Pakistani Currency. *KIET Journal of Computing and Information Sciences*, *6*(1), 123-147.

Nazir, B., Imran, M., & Jehangir, B. (2024). Banknote Verification using Image Processing Techniques. *Journal of Computing & Biomedical Informatics*, *7*(01), 125-143.

Ogbuju, E., Usman, W., Obilikwu, P., & Yemi-Peters, V. (2020). Deep learning for genuine Naira banknotes. *FUOYE Journal of Pure and Applied Sciences (FJPAS)*, *5*(1), 56-67.

Omeiza, I., Ogunbiyi, O., Ogundepo, O., Otuoze, A., Egbune, D., & Osunsanya, K. (2023). A Method of Colour-Histogram Matching for Nigerian Paper Currency Notes Classification.

Rao, M. S. N. V. B., & Gowtham, N. S. (2024). Efficient currency recognition and value detection system using image processing. *World Journal of Advanced Research and Reviews*, *22*(1), 241-255.

Richard-Nnabu, N. E., Ituma, C., & Nweke, H. F. (2024). Convolutional neural networks method for folded naira currency denominations recognition and analysis. *Journal of the Nigerian Society of Physical Sciences*, 2239-2239.

Roy, V., Mishra, G., Mannadiar, R., & Patil, S. (2019). Fake currency detection using image processing. *International journal of computer science and mobile computing*, *8*(4), 88-93.

Sarkar, S., & Pal, A. K. (2022). A new approach to fuzzy logic analysis of Indian currency recognition. *Journal of Print and Media Technology Research*, *11*(2), 119-128.

Sharan, V., & Kaur, A. (2019). Detection of counterfeit Indian currency note using image processing. *International Journal of Engineering and Advanced Technology*, *9*(1), 2440-2447.

Shelke, P., Dedgaonkar, S., Bhimanpallewar, R., Mirajkar, R., & Naik, K. (2023). Fake Currency Detection Using Image Processing and Random Forest Algorithm. *Recent Trends in Artificial Intelligence & its Applications*, *2*(1), 26-32.

Shiby, A., Francis, F., Jose, P., & Augustine, R. (2021). Detection of Fake Currency Using Image Processing. *International Journal for Research in Applied Science and Engineering Technology*.

Srarka, S., & Pal, A. K. (2024). Paper Currency: A systematic review of design, security features, and counterfeit detection. *Acta graphica: znanstveni časopis za tiskarstvo i grafičke komunikacije*, *32*(2), 84-97.

Sruthy, R. (2022). A review of Fake Currency Recognition Methods. *no. July*, 2633-2636.

Swami, K., Murumkar, K., Ghadekar, D. P., & Solanki, Y. (2019). Currency recognition and fake currency identification using image processing. *International Journal of Advance Research, Ideas and Innovation in Technology*, *5*(3), 659-661.

Veeramsetty, V., Singal, G., & Badal, T. (2020). Coinnet: platform independent application to recognize Indian currency notes using deep learning techniques. *Multimedia Tools and Applications*, *79*, 22569-22594.

Vidhate, A., Shah, Y., Biyani, R., Keshri, H., & Nikhare, R. (2021). Fake currency detection application. *Int. Res. J. Eng. Technol.(IRJET)*, *8*(05), 2395-0056.

Woods, N. C., & Oladosu, H. S. (2020). Recognition and Fitness Classification of Nigerian Currency for Automatic Teller Machines. *African Journal of Computing & ICT*, *13*(1).

Zarin, A., & Uddin, J. (2019). A hybrid fake banknote detection model using OCR, face recognition and hough features. 2019 Cybersecurity and Cyberforensics Conference (CCC),

Zhang, Q., & Yan, W. Q. (2018). Currency detection and recognition based on deep learning. 2018 15th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS),