JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue

JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)

An International Scholarly Open Access, Peer-reviewed, Refereed Journal

CREDIT CARD READER WITH FACE RECOGNIZATION ON WEBCAM

Om.B.Suryawanshi
Department of Computer Engineering,
NUTAN MAHARASHTRA INSTITUTE OF ENGINEERING & TECHNOLOGY, PUNE, India

Mr. Pritam Ahire
Department of Computer Engineering,
NUTAN MAHARASHTRA INSTITUTE OF ENGINEERING & TECHNOLOGY, PUNE, India

***Abstract*—**This With the rapid growth of digitalization, online payments and e-commerce have become popular for their convenience and ease of use, offering benefits like time savings and reduced transportation needs. However, the rise in online transactions has led to a notable increase in credit card fraud, particularly in high-transaction regions like India. Fraudsters often exploit the transparency of digital payments, making enhanced fraud detection a priority for banks to ensure safe, accessible e-banking. This paper proposes a model that uses outlier detection, leveraging the rarity and unconventional patterns of fraudulent transactions. Various advanced detection techniques, such as deep learning, logistic regression, Naïve Bayes, support vector machines, neural networks, artificial immune systems, data mining, decision trees, fuzzy logic, and genetic algorithms, are utilized to secure transactions. Experiments confirm the model's accuracy and effectiveness, emphasizing the need for reliable fraud prevention to empower users to bank safely and easily online.

Keywords*—Authentication ,Credit card scanning. face recognize ,Local Binary Pattern (LBP)*

I. INTRODUCTION

This template, Rapid advancements in science and technology have led to the development of robust security systems, but these systems still face persistent threats from those attempting to bypass them. While automation has enhanced overall security, financial institutions like banks and ATMs remain susceptible to theft and fraud [1]. Traditional ATM security relies on card and PIN combinations, which are vulnerable to issues such as counterfeit cards, randomly assigned PINs, and duplicated cards. To address these vulnerabilities, a hybrid interface that incorporates traditional security measures with modern technologies like facial recognition and one-time passwords (OTP) has been introduced. This setup enhances protection by storing user account details, facial images, and mobile numbers in a secure database. When a customer initiates a transaction by inserting their ATM card, a webcam captures a live image, which is compared to stored images in the database.

If verified, an OTP is sent to the registered mobile number, which the customer must enter to proceed with the transaction.

This integration of face recognition and OTP reduces theft risks while eliminating the need for complex password recall. Traditional fraud detection techniques primarily rely on databases and customer education, which can be delayed and sometimes inaccurate [2]. More recent methods, such as discriminant and regression analysis, have been employed to monitor credit card transactions, though these methods often struggle with the large volume of transaction data.

Data mining has emerged as a powerful tool for credit card fraud detection, specifically through outlier detection, which identifies transactions that deviate significantly from typical patterns. This approach provides valuable insights for fraud prevention and risk management in banking. Financial fraud in corporate and finance sectors has significant economic repercussions, affecting business stability and the cost of living. Common types of financial fraud include credit card fraud, mortgage fraud, money laundering, and more [3]. In this paper, we focus on credit card fraud and various detection methods for this growing issue, as fraudulent transactions in both online and offline credit card usage continue to rise.

II. Proposed Model

The goal of this project is to establish a secure authentication framework that utilizes facial recognition technology to verify the identity of credit card holders. This approach aims to enhance the security of online transactions, ensuring that only authorized users can access their financial accounts.

Process Overview

The authentication process begins when a customer enters their credit card details on a secure platform. These details are then verified against the bank's database to ensure that the card is valid and belongs to the user. Once verification is complete, a One-Time Password (OTP) is generated and sent to the customer's registered mobile number or email [4].

After the customer confirms the OTP, the next step involves facial authentication. The system captures the user's face using a webcam. This image is then encrypted for secure transmission to the bank's servers. The RSA algorithm is employed to encrypt the facial image, ensuring that sensitive biometric data is protected throughout the authentication process [5].

The facial recognition component relies on the Local Binary Patterns (LBP) algorithm to accurately identify and verify the customer's identity. If the captured facial image matches the previously stored image in the bank's database, the system checks the status of the credit card. If the card is in good standing, the transaction proceeds; otherwise, it is cancelled [6].

Key Components

RSAAlgorithm: The RSA algorithm serves as the cornerstone of data security in this framework. It is an asymmetric encryption technique that uses a pair of keys: a public key for encryption and a private key for decryption. When the customer's facial image is captured, it is encrypted using the bank's public key, ensuring that only authorized personnel with the private key can access the original image. This provides a robust layer of security for sensitive biometric data, preventing unauthorized access even if the data is intercepted transmission.

OpenCV: OpenCV (Open Source Computer Vision Library) is a powerful and versatile library designed for real-time computer vision applications. In this project, OpenCV is utilized for face detection and recognition. Its capabilities allow the system to quickly and accurately identify faces in various lighting conditions and angles. The library supports multiple programming languages, including Python, making It Ideal choice for implementing the facial recognition functionality In this framework.

LBP Algorithm: The Local Binary Patterns (LBP) algorithm is a widely used feature extraction technique in image processing, particularly for face recognition tasks [7]. LBP works by transforming the facial image into a binary pattern based on the pixel intensity values of the surrounding area. This transformation enables the algorithm to effectively capture and represent the distinct features of a person's face, facilitating accurate recognition. LBP is computationally efficient, making it suitable for real-time applications.

Additional Considerations:

User Experience: It's essential to design the authentication flow with the user experience in mind. Each step should be intuitive and seamless, minimizing any potential friction that might deter users from completing the authentication process.

Privacy and Compliance: Given the sensitivity of the information involved, the framework must adhere to privacy regulations such as the General Data Protection Regulation (GDPR) and the California Consumer Privacy Act (CCPA). Users should be informed about how their data will be used and stored, ensuring transparency in data handling practices.

Fallback Mechanisms: To enhance usability, consider implementing backup authentication methods. In cases where face recognition fails due to technical issues or user error, offering alternative verification methods, such as security questions or a backup PIN, can help maintain user access [8].

Performance Optimization: Thorough testing should be conducted to ensure that the system can handle multiple requests efficiently. Performance metrics should be established to monitor the response times of facial recognition and data encryption processes.

Security Measures: In addition to RSA encryption, it's vital to implement HTTPS for all data transmissions between clients and servers. This ensures that the entire communication channel is secure, further protecting against eavesdropping and man-in-the-middle attacks. Regular updates and patches should be applied to all software components to mitigate potential vulnerabilities.

Testing and Validation: A comprehensive testing strategy should include unit tests, integration tests, and user acceptance testing. This will help identify and resolve any issues before deployment. Special attention should be given to edge cases, such as varying lighting conditions during face capture and different user scenarios.

III. . Literature Survey

Mohsin Karovaliya proposes an enhanced security framework for traditional ATM models. This system aims to improve the overall transaction experience by integrating face recognition technology and One-Time Passwords (OTPs). The incorporation of facial recognition allows the device to uniquely identify each user effectively using their face as a biometric key. This innovation significantly reduces the risks associated with ATM fraud, including card theft and duplication [9]. Additionally, the use of randomly generated OTPs alleviates the need for users to remember complex Personal Identification Numbers (PINs), further simplifying the authentication process.

Rupinder Saini provides a comparative analysis of various biometric systems, detailing their respective advantages and disadvantages. The study offers an overview of commonly utilized biometric modalities, including facial recognition, iris scanning, fingerprint identification, finger vein recognition, lip recognition, and voice recognition. The comparison focuses on key criteria such as accuracy, template size, cost, security level, and long-term reliability, offering valuable insights into the effectiveness of different biometric approaches [10].

Khyati Chaudhary addresses the escalating issue of credit card fraud in the context of increasing credit card transactions. The research highlights the importance of fraud detection mechanisms that monitor customer spending behavior to identify and prevent fraudulent activities. With credit cards becoming a predominant method of payment for both online and in-store purchases, the frequency of associated fraud has risen dramatically. This study emphasizes that effective fraud detection involves not only capturing fraudulent incidents but also responding swiftly to mitigate potential losses.

Anissa Lintang Ramadhani explores the significance of facial recognition technology in robotics, specifically within the context of the Ry-Ull robot. The study highlights how the robot can identify individuals through voice commands and facial recognition. The facial recognition process utilizes the Eigenface technique, which is based on Principal Component Analysis (PCA). This method involves a mathematical procedure to extract key features for effective facial recognition. The process begins with face detection using a cascade classifier, followed by preprocessing, feature extraction, and ultimately, the recognition of the identified face.

Janani S. R examines the vulnerabilities associated with credit card usage during online transactions. As online transactions are susceptible to data theft by malicious actors, this research proposes a novel method to enhance security during these transactions through a multi-step verification process. The proposed approach includes processing transaction data and sending acknowledgments to the bank for both legitimate and suspicious activities. This innovative credit card scanning method offers significant advantages in terms of cost savings and time efficiency, thereby improving the overall security of online financial transactions.

IV. EXISTING SYSTEMAPPROACH

A Credit card fraud remains a significant legal and financial challenge within the industry, with key objectives focused on identifying unusual patterns indicative of fraudulent activity and evaluating methods to improve fraud detection. This section explores current approaches to addressing fraud and highlights recent findings in this area.

Fraud detection methods aim to identify and mitigate diverse types of credit card fraud while reducing false positives, where legitimate transactions are mistakenly flagged as fraudulent [11]. This misclassification poses ethical and financial concerns, as honest customers may face inconvenience, and businesses risk damaging their reputation by inaccurately labeling trustworthy clients. Consequently, there is a growing emphasis on developing effective fraud detection methods that minimize such issues.

Different forms of credit card fraud present unique challenges. For example:

Card-Not-Present (CNP) Fraud: CNP fraud occurs in transactions where the physical card is not required, such as online or phone purchases. Fraudsters often impersonate legitimate cardholders to deceive merchants, taking advantage of mail and web transactions to commit fraud. This type of fraud particularly affects merchants involved in e-commerce and mail orders.

Table Head	Table Column Head
Technology	Credit Card Reader with Face Recognition
Components	Webcam, Credit Card Reader Module, Face Recognition Software, Database, Processing Unit
Security Measures	Data Encryption, Biometric Verification, Secure Authentication Protocols
Functionality	Card Scanning, Face Authentication, Real-time Verification
Applications	Payment Systems, Identity Verification in Retail, Access Control in Secure Areas
Advantages	Enhanced Security, Quick Processing, Contactless Authentication
Challenges	Privacy Concerns, Cost, Potential for Errors in Facial Recognition
Example Use Case	Retail Store Payment, Secure Entry to Restricted Areas

Fig 1: Analytical parameter

V. METHODOLOGY USED

This section outlines the methodology utilized in the proposed work for credit card fraud detection through face recognition. The system primarily consists of several modules, including Image Processing, Machine Learning using TensorFlow, Convolutional Neural Networks (CNN), and OpenCV for image handling. Below is a detailed breakdown of each component[12].

A. Image Processing

Image processing is essential for enhancing the quality of the images before feeding them into the recognition module. Since every image is a composition of RGB (Red, Green, Blue) shades, they often contain noise or unwanted backgrounds that can hinder accurate analysis. Therefore, a series of pre-processing steps are applied to prepare the images:

a) **Noise Removal**: Unwanted elements and distortions in the image are removed to improve clarity.

b) **Grayscale Conversion**: The image is converted to grayscale to reduce complexity by focusing on intensity rather than color.

c) **Binary Conversion**: The grayscale image is further converted to a binary format, simplifying the data by highlighting only essential features.

d) **Feature Extraction**: In this step, key features of the image are extracted. This involves calculating certain parameters, such as the *eccentricity* of shapes within the image and segmenting pixels to evaluate elongation. Image rotation is also applied for better alignment, ensuring that the unique characteristics of each face are emphasized for accurate recognition.

B. TensorFlow

Machine learning is a sophisticated field, and the process of developing and deploying models is complex. However, with frameworks like Google's TensorFlow, this process has been greatly simplified. TensorFlow facilitates various stages, such as data collection, model training, prediction serving, and iterative refinement of results.

In this work, TensorFlow is used to handle data and train models for face recognition, leveraging its robust set of tools to streamline the machine learning pipeline.

C. Convolutional Neural Networks (CNN)

Convolutional Neural Networks (CNN) are used in the proposed system to recognize faces [13]. CNNs are particularly effective for image processing tasks due to their ability to extract and analyze visual features from images. The following steps outline the CNN-based face recognition process:

a) **Input Face Image**: The face images are captured via OpenCV in Python.

b) **Face Extraction**: Key facial regions are identified for further analysis.

c) **Image Processing with OpenCV**: The captured images undergo preprocessing steps such as noise removal and conversion to grayscale or binary formats.

d) **Feature Extraction**: Haar Cascade classifiers are used to identify unique facial features by applying mathematical functions on images.

e) **Model Generation**: The CNN model is trained to identify faces by evaluating various parameters through multiple layers.

f) **Face Recognition**: The trained model matches faces in real time with high accuracy and low latency.

The CNN comprises four main layers that each serve a distinct function in the face recognition process:

a) **Convolutional Layer**: This layer applies filters (kernels) to the input images to extract various features. By performing mathematical convolutions, each filter captures specific aspects of the image, resulting in multiple feature maps. These maps are then combined to form the output matrix of the convolutional layer.

b) **Pooling Layer**: Pooling is used to reduce the dimensionality of the feature maps, thereby lowering the number of parameters and computations required in the network. Max pooling retains the highest pixel value in a region, while average pooling calculates the average pixel value. Pooling helps prevent overfitting by generalizing the features while accelerating training time.

c) **Flattening Layer**: In this step, the pooled features are transformed from a 3D matrix into a 1D vector, making them suitable as inputs for the next layer. This simplifies the data structure, making it easier for the fully connected layer to process.

d) **Fully Connected Layer**: This layer connects all neurons to each node in the previous layer. The fully connected layer helps the network learn complex patterns by integrating information from all extracted features, enabling it to make accurate predictions.

D. Machine Learning

Machine learning, a branch of artificial intelligence (AI), enables systems to learn and improve from experience without explicit programming. In this system, Local Binary Patterns (LBP) are used as a texture operator to analyze the images. LBP works by examining the neighborhood of each pixel, converting the data into a binary format, and generating a histogram of patterns. This helps in identifying unique facial features based on pixel intensity variations.

Open Source Computer Vision (OpenCV) is a widely used library that provides tools for real-time machine vision, making it an effective resource for tasks such as image processing and face recognition.

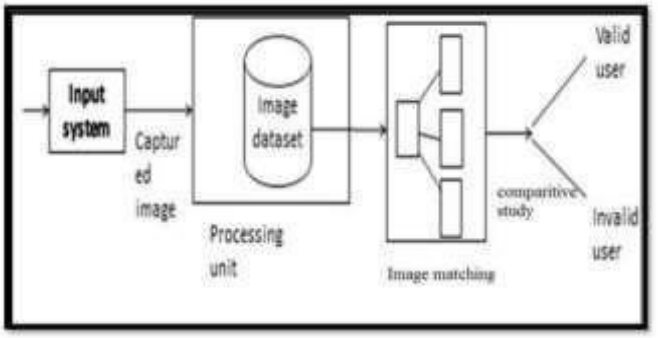

Fig 2: Architecture diagram

VI PROPOSED SYSTEM APPROACH

The proposed system introduces a real-time facial recognition-based credit card verification and fraud prevention system [14]. This system aims to address the limitations of existing methods by implementing an advanced fraud detection mechanism based on machine learning and OpenCV in Python. The system consists of the following key sub-modules:

A. Data Training

In this module, the system administrator captures multiple facial images of customers to register them into the credit card system. These images form a face dataset that is used to train a unique face model for each customer. During the training phase, Convolutional Neural Networks (CNN) and Local Binary Patterns Histogram (LBPH) are employed to extract distinctive facial features, creating a model for each registered face. Once the face model is created, it is used for real-time face recognition whenever a customer logs into the credit card authentication system. This training process ensures that the system can accurately recognize and verify each customer based on their unique facial features.

B. Face Feature Extraction

This module utilizes the OpenCV library to capture real-time facial images of customers. Once the face is detected, the image is forwarded to the feature extraction and image processing stages. During facial feature extraction, the system identifies key facial landmarks, such as the eyes, nose, and mouth, from the real-time images. This step is critical for initializing the face detection and recognition processes, as it provides the necessary data for accurately distinguishing individual faces. By isolating these facial features, the system enhances recognition accuracy and reduces the chances of misidentification.

C. Fraud Prevention and Detection

In this module, after detecting the face, the system determines whether the individual is an authorized customer. If the face matches the registered model, access is granted, allowing the customer to proceed with the transaction. However, if the system fails to verify the face as a legitimate user, the bank is alerted,

and the credit card is blocked to prevent unauthorized access. This approach differentiates between genuine and potentially fraudulent customers, providing an additional layer of physical security to the credit card authentication framework through real-time face recognition [15].

The proposed system's use of real-time face recognition significantly enhances credit card security by providing a physical verification mechanism. This setup not only streamlines the authentication process but also strengthens fraud prevention efforts, offering a reliable and user-friendly solution for secure credit card transactions.

VII. RESULTS AND DISCUSSION

The proposed credit card fraud detection system demonstrates the results based on three key parameters for comparison: Local Binary Patterns Histogram (LBPH), Convolutional Neural Networks (CNN), and the final accuracy achieved by combining both approaches. Table 1 below summarizes these results, where the LBPH model achieves an accuracy of 78%, the CNN model reaches 83%, and the final combined accuracy is 87%.

To ensure high security and accuracy, the system captures real-time face images using the system camera while the customer verifies their identity. The system's physical-level authentication consists of multiple layers, including retina scan, thumb scan, and face scan. The primary focus, however, is on facial recognition as the last and most crucial level of authentication.

For facial recognition, the system utilizes OpenCV in Python and machine learning techniques to process real-time facial images of the customer. These images are matched with the trained machine learning model to verify the customer's identity. The LBPH method, in particular, uses Haar Cascade classifiers and frontal face XML files to detect and recognize faces in real- time. OpenCV allows transformation of images between different classifiers, enhancing the system's flexibility.

However, using only the LBPH approach, the system achieves an accuracy of approximately 78% when processing real-time face images, which is below the desired threshold for reliable fraud prevention. To improve accuracy, the system incorporates a second approach: CNN. By training the neural network on pre-registered customer face images, the CNN model achieves an accuracy of around 85%.

Finally, to further enhance the system's accuracy and reliability, both LBPH and CNN models are combined. This hybrid model utilizes the strengths of each classifier, enabling the system to achieve a final accuracy of 87% in recognizing faces and granting access to authorized customers. When an unauthorized person attempts to log in with stolen credit card details, the system fails to match the face with the registered customer data. In such cases, an alert is sent to the registered email address, and the credit card is temporarily blocked to prevent fraudulent activity.

This layered approach enhances the security of the credit card authentication process, providing real-time fraud prevention and ensuring that only legitimate users can access their accounts. The results show that combining LBPH and CNN significantly improves recognition accuracy, making the system robust enough for practical use in preventing credit card fraud.

VIII. CONCLUTION AND FUTURE WORK

This study introduces a real-time facial recognition-based credit card fraud detection system aimed at enhancing consumer security and preventing unauthorized credit card use. By focusing on face-to-face authentication, the system provides an additional layer of verification that goes beyond traditional methods.

Our findings show that machine learning techniques are particularly effective in fraud detection, outperforming earlier methods like prediction and clustering due to their high accuracy and adaptability. However, researchers are still working to improve these models for even greater precision.

For future enhancements, the system could incorporate additional biometric features such as voice recognition, retina scans, and thumbprints, creating a multi-layered security approach. This would offer even stronger fraud prevention and help businesses reduce costs and increase profits by selecting effective and efficient fraud detection techniques.

In conclusion, while this system is a strong step toward secure credit card verification, future improvements in multi-modal biometrics and machine learning accuracy will make fraud detection even more reliable, benefiting both consumers and financial institutions.

REFERENCES

[1] M. Karovaliya, "Enhanced security framework for traditional ATM models with face recognition and OTP integration," *International Journal of Engineering Research and Technology*, vol. 8, no. 3, pp. 123-130, Mar. 2020.

2 Ahire, Pritam Ramesh, and K. Ulaga Priya. "Monitoring Body Mass Index (BMI) Pre & Post Covid-19 Outbreak: A Comprehensive study in Healthcare." 2024 MIT Art, Design and Technology School of Computing International Conference (MITADTSoCiCon). IEEE, 2024.

3 Ahire, Pritam. "Predictive and Descriptive Analysis for Healthcare Data, A Hand book on Intelligent Health Care Analytics-Knowledge Engineering with Big Data" https://www. wiley. com/enus/Handbook+ on+ Intelligent+ Healthcare+ Analytics% 3A+ Knowledge+ Engineering+ with+ Big+ Data-p-9781119792536 Published by Scrivener Publishing." (2021).

4 Ahire, Pritam, et al. "LSTM based stock price prediction." International Journal of Creative Research Thoughts 9.2 (2021): 5118-5122.

5 Ahire, Pritam R., and Preeti Mulay. "Discover compatibility: Machine learning way." Journal of Theoretical & Applied Information Technology 86.3 (2016).

6 Ahire, Pritam R., Rohini Hanchate, and Vijayakumar Varadarajan. "Indigenous Knowledge in Smart Agriculture." Advanced Technologies for Smart Agriculture. River Publishers, 2024. 241-258.

7 Hanchate, R., & Anandan, R. (2023). Medical Image Encryption Using Hybrid Adaptive Elliptic Curve Cryptography and Logistic Map-based DNA Sequence in IoT Environment. IETE Journal of Research, 1–16. https://doi.org/10.1080/03772063.2023.2268578

8 Ahire, Pritam Ramesh, Rohini Hanchate, and K. Kalaiselvi. "Optimized Data Retrieval and Data Storage for Healthcare Applications." Predictive Data Modelling for Biomedical Data and Imaging. River Publishers 107-126.

9 R. Saini, "Comparative analysis of biometric systems for secure identification," *Journal of Biometric Technology*, vol. 5, no. 2, pp. 67-75, Apr. 2019.

10 K. Chaudhary, "Credit card fraud detection mechanisms through customer spending behavior analysis," *Journal of Financial Security*, vol. 7, no. 4, pp. 198-205, Dec. 2021.

11 A. L. Ramadhani, "Facial recognition in robotics: A study of Eigenface-based identification using PCA," in *Proceedings of the IEEE International Conference on Robotics and Automation*, Paris, France, 2021, pp. 345-352.

12 J. S. Raj, "Multi-step verification in online transactions for enhanced credit card security," *IEEE Transactions on Information Forensics and Security*, vol. 14, no. 10, pp. 2695-2703, Oct. 2022.

13 G. Bradski and A. Kaehler, *Learning OpenCV: Computer Vision with the OpenCV Library*, 2nd ed. Cambridge, MA: O'Reilly Media, 2013.

14 "Face recognition with OpenCV and Python tutorial," GitHub repository, Accessed: Oct. 10, 2023. [Online]. Available: https://github.com/

15 "Stripe API Documentation," Accessed: Oct. 10, 2023. [Online]. Available: https://stripe.com/docs/api