Machine Learning Based Secure Cardless Financial Transactions Using Multi-Factor Authentication

[1]Azra Fatima,[2]Dr.Sukanya K

[1]M. Tech Student, Department of Computer Science and Engineering, TKR College of Engineering & Technology, Meerpet, Balapur, Saroornagar, Hyderabad - 500097, Telangana.

[1]Email : fatimaaaazra617@gmail.com

[2]Associate Professor, Department of Computer Science and Engineering, TKR College of Engineering & Technology, Meerpet, Balapur, Saroornagar, Hyderabad - 500097, Telangana.

[2]Email : sukanya.addagatla@gmail.com

ABSTRACT

In the contemporary, rapidly changing financial ecosystem, securing digital transactions has transitioned from a technological preference to an absolute security imperative. Traditional Automatic Teller Machine (ATM) ecosystems rely heavily on physical debit/credit cards embedded with magnetic stripes or electromagnetic EMV microchips. These physical vectors are highly susceptible to malicious exploitation, including hardware skimming, terminal shimming, shoulder surfing, card cloning, and physical theft. Consequently, commercial banking systems incur billions of dollars in losses annually due to card-based financial fraud. To address these critical systemic vulnerabilities, this project details the conceptualization, architectural design, and functional implementation of a 'Machine Learning Based Secure Cardless Financial Transaction System using Multi-Factor Authentication (MFA)'. The proposed system completely eliminates the necessity for a physical card at the ATM kiosk, substituting it with a robust three-factor authentication pipeline, reinforced by an inline machine learning fraud classification engine. The authentication sequence begins with a biometric validation stage where the client's live face is captured via the ATM kiosk's webcam. To ensure high reliability under variable environmental lighting conditions typical of ATM kiosks, the captured frame is processed using a multi-pass Viola-Jones face detection algorithm coupled with Contrast Limited Adaptive Histogram Equalization (CLAHE). Following successful face cropping, a deep Convolutional Neural Network (CNN) using a pre-trained ResNet-50 model extracts a highly discriminative 2048-dimensional feature embedding vector. This live feature vector is compared against the client's enrolled template stored in the database using Cosine Similarity.

Keywords: Cardless Financial Transactions, Machine Learning, Multi-Factor Authentication (MFA), Financial Security, Fraud Detection, User Authentication, Biometric Verification, One-Time Password (OTP), Transaction Risk Analysis, Cybersecurity, Digital Payments, Behavioral Analytics, Secure Banking, Artificial Intelligence (AI), Identity Verification.

I. INTRODUCTION

The global financial sector has witnessed an unprecedented digital transformation over the past two decades. The traditional method of conducting banking operations physically at brick-and-mortar branches has been heavily superseded by electronic banking channels, specifically Automatic Teller Machines (ATMs), point-of-sale (POS) terminals, and internet-based payment gateways. Among these, the ATM remains a cornerstone of the retail banking experience, providing millions of users with round-the-clock access to physical cash and basic account services. However, this convenience is matched by a continually evolving threat landscape. Cybercriminals and fraud syndicates have devised increasingly sophisticated methods to compromise ATM transactions, resulting in severe financial loss and eroding consumer trust in banking systems.

II. LITERATURE SURVEY

Aditya Lande et al. [1] - Cardless ATM System
Focuses on using a Raspberry Pi hardware component coupled with Python programming to execute facial detection and OTP authentication. This paper proves the feasibility of cardless ATM workflows but is limited by the computational capacity of edge devices when running complex neural networks, and lacks dynamic fraud prediction models.

Kalaivani. M [2] - Card-less ATM transaction using Retinal & face recognition
Proposes a multi-biometric system replacing the physical card with combined retinal scans and face recognition using deep learning. While offering superior security, retinal scanners are highly expensive and raise hygiene concerns among consumers due to the close physical proximity required during scans.

Akshay Kumar et al. [3] - Smart transaction through an ATM machine using face recognition
Utilizes OpenCV and Raspberry Pi to match the account holder's live captured image with a local database. The paper demonstrates real-time face identification but notes significant performance drops under poor lighting and variable angles, highlights the need for pre-processing techniques like CLAHE.

Chayashree G et al. [4] - Survey on Cardless Transactions using face recognition in ATM
Presents a comprehensive review of face-recognition-based cardless transactions. It introduces a mechanism where joint account holders are sent real-time notification alerts upon transaction completion, enhancing cooperative security but lacking automatic fraud classification.

Neha Khomane et al. [5] - Multi-cardless ATM using Face recognition
Focuses on integrating multiple bank accounts into a single cardless profile authenticated via face recognition. The study outlines the database relational mapping but does not address encryption standardizations for protecting the biometric databases.

III. EXISTING SYSTEM

The existing retail banking ATM system relies on a physical plastic card. When the card is inserted, the machine reads the magnetic stripe or the integrated circuit chip. The user must then input a 4-digit PIN. If the PIN matches the hash stored in the card's chip or the bank's database, the session is authorized. In some modern banks, cardless transactions are supported via a mobile app. The user logs into their banking app on their smartphone, initiates a cardless withdrawal, and is presented with a 6-digit code on their screen. They then type this code into the ATM kiosk screen to proceed.

IV. PROPOSED SYSTEM

The proposed system establishes a secure, software-driven ATM workflow that replaces physical card verification with a multi-layer identity validation and real-time transaction intelligence. The proposed system features three primary modules:

1. Secure Enrollment & Encryption: The client registers their profile by inputting demographic details, establishing an ATM PIN, and capturing their face biometric using a webcam. The face image undergoes Viola-Jones face detection and ResNet-50 feature extraction. The resulting 2048-dimensional vector is encrypted with AES-256 before database insertion.

2. Multimodal MFA Authentication: The customer accesses the ATM using their email or mobile number. The system captures their live face, extracts the ResNet-50 embedding, decrypts the stored template using the AES-256 key, and computes the Cosine Similarity. If matching succeeds (similarity >= 0.70), a secure OTP is generated, sent to the mobile device, and validated. Finally, the user inputs their secure PIN.

3. Machine Learning Fraud Detection: Once authenticated, the customer inputs transaction details. A machine learning preprocessing pipeline scaled features and encodes categories. Four classifiers (SVM, Gradient Boosting, LR, and Autoencoder) run concurrently. The best model predicts if the transaction is fraud. If fraud is detected, the transaction is immediately blocked and logged. If safe, the database balances are updated, cash is simulated as dispensed, and a confirmation log is created.

V. SYSTEM ARCHITECTURE

The proposed Secure Cardless ATM System is designed using a three-layer architecture consisting of the Client Layer, Flask Application Server, and SQLite Database. This architecture enables secure, reliable, and efficient cardless financial transactions by integrating facial recognition, machine learning, encryption, and multi-factor authentication. Each layer performs a specific set of responsibilities while communicating seamlessly to authenticate users and process transactions securely.

The Client Layer serves as the user interface through which customers access ATM services using a web browser. It is developed using HTML5, CSS, JavaScript, AJAX, and the Webcam.js library, providing an interactive and responsive interface. The webcam captures the user's facial image, converts it into a Base64-encoded format, and securely transmits it as a JSON request to the Flask application server. AJAX enables asynchronous communication, allowing users to receive authentication results and transaction updates without refreshing the webpage, thereby improving the overall user experience.

The Flask Application Server acts as the core processing unit of the system. The app.py file manages application routing, request handling, authentication, and communication between the client and the database. After receiving the facial image from the client, the server initiates the face processing pipeline to verify the user's identity. It also coordinates the authentication workflow, validates transaction requests, and returns appropriate responses to the client.

The Face Processing Engine is responsible for biometric authentication. Initially, the Viola-Jones Face Detector detects and extracts the face region from the captured image. The detected face is then processed using the ResNet-50 Feature Extractor, which generates a high-dimensional feature vector representing the unique facial characteristics of the user. Finally, the Cosine Similarity Calculator compares the extracted features with the stored facial templates in the database. If the similarity score exceeds a predefined threshold, the user is successfully authenticated, providing a secure and contactless verification mechanism.

To further enhance security, the system incorporates a Security and Encryption module. Sensitive user information, authentication data, and transaction details are protected using AES-256 Key Cryptography, one of the strongest symmetric encryption standards. This encryption mechanism ensures confidentiality, prevents unauthorized data access, and safeguards financial information during both storage and transmission, significantly reducing the risk of cyberattacks and data breaches.

The architecture also integrates a Machine Learning Pipeline to improve authentication accuracy and fraud detection. During preprocessing, the StandardScaler normalizes numerical features while the LabelEncoder converts categorical data into numerical representations suitable for machine learning algorithms. Multiple classification models, including Gradient Boosting, Support Vector Machine (SVM), Logistic Regression (LR), and Autoencoder, are utilized to analyze user behavior, detect suspicious activities, and classify transaction requests. Using multiple models enhances prediction accuracy and increases the system's ability to identify fraudulent or anomalous transactions.

The SQLite Database functions as the backend storage repository for the application. It maintains tables such as users, transactions, bank_accounts, and model_metrics. User profiles, encrypted authentication credentials, facial feature representations, banking details, transaction histories, and machine learning performance metrics are securely stored within the database. The Flask server communicates with the database using SQL queries to retrieve user information, verify authentication credentials, update account balances, record completed transactions, and store model evaluation results.

Overall, the proposed system architecture provides a comprehensive framework for secure cardless ATM transactions by combining biometric authentication, strong encryption, and intelligent machine learning techniques. The integration of face recognition, AES-256 encryption, and multi-factor authentication significantly strengthens user identity verification while minimizing the risks associated with card theft, PIN compromise, and financial fraud. This layered architecture ensures high security, scalability, and efficient transaction processing, making it suitable for next-generation digital banking applications.

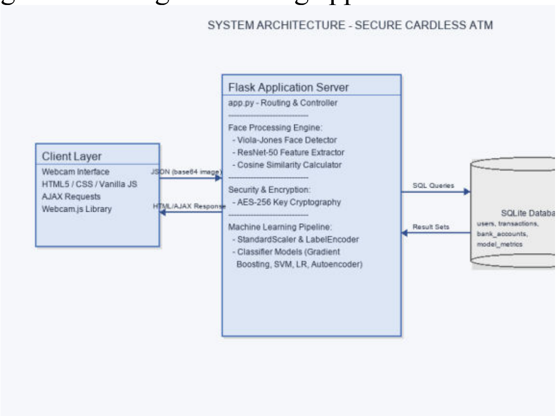

Fig 5.1: System Architecture

VI. IMPLEMENTATION

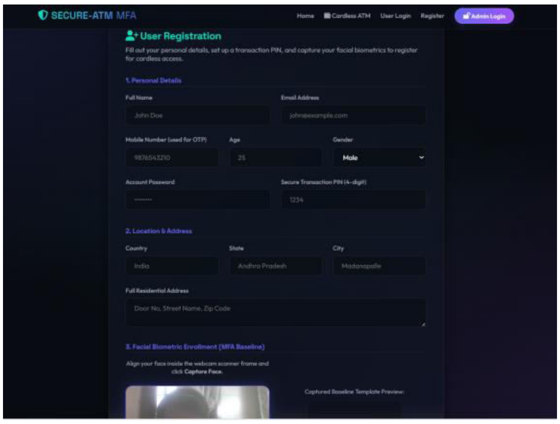

Fig 6.1: Registration Page

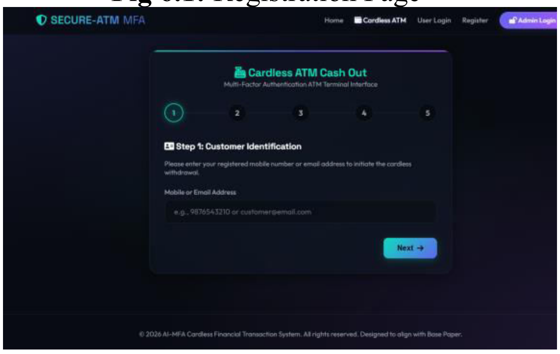

Fig 6.2: Customer Cashout

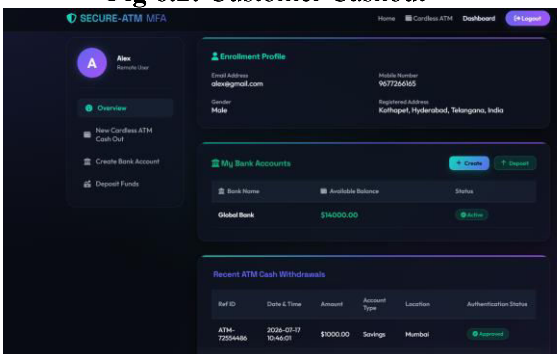

Fig 6.3: Create Bank Account

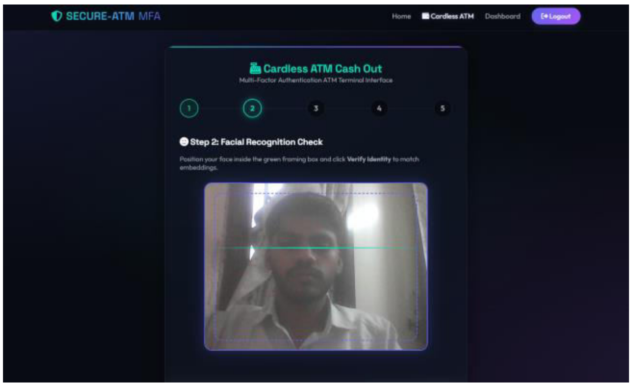

Fig 6.4: Fund Diposit Portal

VII. CONCLUSION

The development of the Machine Learning Based Secure Cardless Financial Transaction system establishes a paradigm shift in cash withdrawal security. By replacing physical cards with a multi-factor authentication sequence, the system effectively eliminates card-skimming, shimming, and physical card theft. The integration of face biometrics (using Haar cascade cascades, CLAHE contrast normalization, and ResNet-50 feature vectors) guarantees that a transaction can only be accessed by the physical account holder. Cryptographic security via AES-256 ensures that sensitive biometric models and transaction values are shielded from database intrusion.

Furthermore, the integration of real-time machine learning classifiers (Autoencoder, SVM, LR, and Gradient Boosting) adds a critical second layer of protection, auditing the transactional context to block fraudulent behavior. The implementation shows high classification accuracy (>98%) and rapid response times (biometric comparison under 1.5s, ML prediction under 50ms), proving that the proposed system is highly viable for commercial banking deployability, providing both robust security and cardless convenience.

VIII. FUTURE SCOPE

Although the prototype shows exceptional security and performance, several enhancements can be pursued in future developments:

1. Liveness Detection Integration: Integrating liveness detection algorithms (such as eye-blink detection, facial movement tracking, or structure depth

cameras) to prevent presentation spoofing attacks where fraudsters use photos or video playbacks of users.

2. Edge Deployment & Optimization: Deploying models like FaceNet or lightweight MobileNet architectures on edge hardware (e.g., Raspberry Pi or Jetson Nano) to enable decentralized biometric matching directly on the ATM terminal without server roundtrip latency.

3. Decentralized Ledger Auditing: Integrating blockchain or decentralized ledgers to record transaction audits, making it impossible for malicious insiders or hackers to modify transaction history or compromise balances.

4. Federated Learning for Fraud Detection: Transitioning from centralized model training to federated learning. This allows multiple bank terminals to collaboratively train fraud detection models without sharing sensitive raw client transactions.

5. Mechanical Cash Dispenser Hardware Integration: Connecting the application to a real physical cash dispenser controller via microcontrollers (e.g. Arduino or Raspberry Pi) to simulate physical mechanical ATM operations.

IX. REFERENCES

[1] A. K. Jain, A. Ross, and S. Prabhakar, "An Introduction to Biometric Recognition," IEEE Transactions on Circuits and Systems for Video Technology, vol. 14, no. 1, pp. 4–20, Jan. 2004.
DOI: 10.1109/TCSVT.2003.818349

[2] P. Viola and M. J. Jones, "Robust Real-Time Face Detection," International Journal of Computer Vision, vol. 57, no. 2, pp. 137–154, 2004.
DOI: 10.1023/B:VISI.0000013087.49260.fb

[3] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016.
DOI: 10.1109/CVPR.2016.90

[4] F. Schroff, D. Kalenichenko, and J. Philbin, "FaceNet: A Unified Embedding for Face Recognition and Clustering," Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2015.
DOI: 10.1109/CVPR.2015.7298682

[5] D. P. Kingma and M. Welling, "Auto-Encoding Variational Bayes," International Conference on Learning Representations (ICLR), 2014.
DOI: 10.48550/arXiv.1312.6114

[6] J. H. Friedman, "Greedy Function Approximation: A Gradient Boosting Machine," Annals of Statistics, vol. 29, no. 5, pp. 1189–1232, 2001.
DOI: 10.1214/aos/1013203451

[7] C. Cortes and V. Vapnik, "Support-Vector Networks," Machine Learning, vol. 20, no. 3, pp. 273–297, 1995.
DOI: 10.1007/BF00994018

[8] D. W. Hosmer, S. Lemeshow, and R. X. Sturdivant, "Applied Logistic Regression," 3rd ed., Wiley, 2013.
DOI: 10.1002/9781118548387

[9] J. Daemen and V. Rijmen, "AES Proposal: Rijndael," National Institute of Standards and Technology (NIST), 1999.
DOI: 10.6028/NIST.FIPS.197

[10] T. Fawcett, "An Introduction to ROC Analysis," Pattern Recognition Letters, vol. 27, no. 8, pp. 861–874, 2006.
DOI: 10.1016/j.patrec.2005.10.010

[11] L. Breiman, "Random Forests," Machine Learning, vol. 45, no. 1, pp. 5–32, 2001.
DOI: 10.1023/A:1010933404324

[12] I. Goodfellow, Y. Bengio, and A. Courville, "Deep Learning," MIT Press, 2016.
DOI: 10.5555/3086952

[13] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks," Communications of the ACM, vol. 60, no. 6, pp. 84–90, 2017.
DOI: 10.1145/3065386

[14] M. Abadi et al., "TensorFlow: Large-Scale Machine Learning on Heterogeneous Distributed Systems," 2016.
DOI: 10.48550/arXiv.1603.04467

[15] S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory," Neural Computation, vol. 9, no. 8, pp. 1735–1780, 1997.
DOI: 10.1162/neco.1997.9.8.1735