Facial Recognition-Based Biometric ATM System: Enhancing Security and User Convenience

A project interim report
submitted in fulfillment of
the requirements for the award of the degree of
BACHELOR OF TECHNOLOGY
in
Computer Science & Engineering
from
APJ Abdul Kalam Kerala Technological University

Submitted By

Mohammed Hisham KP(MEA20CS055)
Muhammed Ajzal K(MEA20CS059)
Nadeem Mohamed (MEA20CS066)
Nihal Muhammed KP(MEA20CS072)

MEA Engineering College
Department of Computer Science and Engineering
Vengoor P.O, Perinthalmanna, Malappuram, Kerala-679325
MAY 2024

Department of Computer Science and Engineering

MEA ENGINEERING COLLEGE

PERINTHALMANNA-679325

Certificate

This is to certify that the Project interim report entitled "Facial Recognition-Based Biometric ATM System: Enhancing Security and User Convenience" is a bonafide record of the work done by **MOHAMMED HISHAM KP(MEA20CS055) , MUHAMMED AJZALK (MEA20CS059), NADEEM MOHAMED (MEA20CS066), NIHAL MUHAMMED KP (MEA20CS072)** *under our supervision and guidance. The report has been submitted in fulfillment of the requirement for award of the Degree of* **Bachelor of Technology** *in* **Computer Science & Engineering** *from the APJ Abdul Kalam Kerala Technological University for the year 2024.*

Prof.Murshida KP
Project Guide
Assistant Professor
Dept.of Computer Science & Engineering
MEA Engineering College

Dr. K. Najeeb
Professor and Head
Dept.of Computer Science and Engineering
MEA Engineering College

Acknowledgements

An endeavor over a long period may be successful only with advice and guidance of many well wishers. We take this opportunity to express our gratitude to all who encouraged us to complete this project. We would like to express our deep sense of gratitude to our resepcted **Principal Dr. G Ramesh** for his inspiration and for creating an atmosphere in the college to do the project.

We would like to thank **Dr. K. Najeeb, Professor and Head of the Department, Computer Science and Engineering** for providing permission and facilities to conduct the project in a systematic way. We are highly indebted to project **Prof. Murshida K P, Asst. Professor** in Computer Science and Engineering for guiding us and giving timely advices, suggestions and whole hearted moral support in the succesful completion of this project.

Our sincere thanks to Project co-ordinators **Asst. Prof. Jitha K** and **Asst. Prof. Sruthy K. G** , Asst. Professors in Computer Science and Engineering for their whole-hearted moral support in completion of this project.

Last but not least, we would like to thank all the teaching and non-teaching staff and my friends who have helped us in every possible way in the completion of this project.

MOHAMMED HISHAM KP (MEA20CS055) DATE: May 6,2024
MUHAMMED AJZAL K (MEA20CS059)
NADEEM MOHAMED (MEA2OCS066)
NIHAL MUHAMMED KP (MEA20CS072)

Abstract

With the rapid advancement of technology and the increasing need for secure and convenient financial transactions, this research introduces a novel approach to Automated Teller Machine (ATM) systems by incorporating facial recognition-based biometrics. The proposed system aims to enhance both security and user convenience, addressing the limitations of traditional authentication methods.

The Facial Recognition-Based Biometric ATM System leverages cutting-edge facial recognition algorithms to authenticate users seamlessly. The system captures and analyzes facial features to uniquely identify individuals, mitigating the risk of unauthorized access and fraudulent activities associated with traditional PIN-based methods. By employing biometric authentication, the system ensures a higher level of security, as facial characteristics are difficult to forge or replicate.

Moreover, the integration of facial recognition technology enhances user convenience by eliminating the need for physical cards or PIN entry. Users can access their accounts and conduct transactions simply by facing the ATM camera, streamlining the authentication process and reducing the risk of card-related fraud, such as skimming and card cloning. This not only enhances the overall user experience but also caters to individuals with accessibility challenges who may find traditional methods cumbersome.

The research explores the technical aspects of implementing facial recognition in ATMs, including the selection of robust algorithms, real-time image processing, and database management for biometric templates. Additionally, considerations regarding privacy and data security are addressed through encryption protocols and compliance with relevant regulations.

To evaluate the system's effectiveness, a comprehensive set of experiments and simulations are conducted, measuring the accuracy, speed, and reliability of facial recognition-based authentication. The results demonstrate the system's potential to provide a secure and user-friendly alternative to existing ATM authentication methods.

List of Abbreviations

ATM	Automated Teller Machine
IoT	Internet of Things
PIN	Personal Identification Number
RPN	Region Proposal Network
CNN	Convolutional Neural Network
FD	Fixed Deposit
PIR	Passive Infrared
ReLU	Rectified Linear Unit
FDDB	Face Detection Data Set and Benchmark

List of Figures

Contents

CHAPTER 1

Introduction

Automated Teller Machines (ATMs) have become indispensable in modern banking, offering customers a wide range of convenient self-service options. These machines enable individuals to perform quick transactions like cash withdrawals, deposits, and fund transfers without the need for a human teller. This self-serviced model has significantly reduced the reliance on physical bank branches, allowing customers to access banking services conveniently and efficiently. Moreover, ATMs operate 24/7, providing round-the-clock access to essential banking functions, which is particularly beneficial for individuals with busy schedules or those requiring immediate banking assistance outside regular hours.

One of the key advantages of ATMs is their versatility in accommodating various transaction types. While most transactions are conducted using debit or credit cards, there are instances where customers can access ATM services without these cards. For example, some ATMs support cardless transactions through mobile banking apps, enabling users to withdraw cash or perform other functions directly from their smartphones. This innovative approach not only enhances convenience but also reflects the continuous technological advancements in the banking sector.

ATMs are equipped with robust measures to protect users' transactions and personal information. These include encrypted communications, PIN entry shields, surveillance cameras, and advanced authentication methods such as biometric verification, including fingerprint or facial recognition. These security features instill confidence in users and help mitigate risks associated with unauthorized access or fraudulent activities.

1.1 Evolution of Automated Teller Machines (ATMs)

The evolution of ATM (Automated Teller Machine) systems is a fascinating journey that spans several decades and involves technological advancements, regulatory changes, and shifts in consumer behavior.

The concept of an automated, self-service banking machine traces back to the 1960s. The first ATMs were introduced in the late 1960s and early 1970s by various banks in different countries, with notable examples including the Bank of America in the United States and Barclays Bank in the United Kingdom. These early ATMs were relatively primitive compared to modern standards, using magnetic stripe cards and requiring users to enter a personal identification number (PIN).

The 1980s marked a period of significant expansion for ATM systems, with banks across the globe adopting the technology to improve customer service and reduce operational costs. Interbank networks were established to facilitate ATM transactions between different banks, enabling customers to access their accounts at ATMs operated by other financial institutions. This period also saw the emergence of international standards for ATM networks and protocols, such as ISO 8583, which standardized communication between ATMs and banking systems.

The 1990s and 2000s witnessed rapid technological advancements in ATM systems. One significant development was the transition from magnetic stripe cards to more secure EMV (Europay, Mastercard, and Visa) chip cards, which provided better protection against fraud. Additionally, ATMs began to offer a wider range of services beyond cash withdrawals, including deposits, funds transfers, bill payments, and account inquiries. Biometric authentication, such as fingerprint scanning, also started to be integrated into some ATM systems to enhance security.

The rise of internet and mobile banking in the 2000s prompted banks to integrate their ATM networks with online and mobile channels. This integration enabled customers to perform a variety of banking transactions seamlessly across different channels, such as transferring funds between accounts via a mobile app and then withdrawing cash from an ATM. Moreover, advanced functionalities like cardless ATM transactions, where customers can withdraw cash using only their mobile phones, have become increasingly popular.

With the proliferation of cyber threats and sophisticated fraud schemes, ATM security has become a top priority for banks. In response, ATM manufacturers and financial institutions have implemented various security measures, such as end-to-end encryption, anti-skimming technology, and real-time fraud detection systems, to protect against

unauthorized access and fraudulent activities. Additionally, regulatory bodies have introduced stricter compliance requirements, such as the Payment Card Industry Data Security Standard (PCI DSS), to ensure the security of ATM networks and customer data.

the evolution of ATM systems is likely to continue, driven by advancements in technology and changing consumer preferences. Some potential future trends and innovations include the adoption of contactless and NFC (Near Field Communication) technologies for cardless transactions, the integration of artificial intelligence and machine learning algorithms for personalized banking experiences, and the expansion of ATM functionalities to support emerging digital currencies like cryptocurrencies.

1.2 Types of Automated Teller Machines (ATMs)

Automated Teller Machines (ATMs) are mainly of two types. One is a simple basic unit that allows you to withdraw cash, check balance, change the PIN, get mini statements and receive account updates. The more complex units provide facilities of cash or cheque deposits and line of credit and bill payments. There are also onsite and offsite Automated Teller Machines: the onsite ATMs are within the bank premises, unlike the offsite ones which are present in different nooks and corners of the country to assure that people have basic banking facilities and instant cash withdrawals if they can't go to a bank branch. ATMs can also be categorized based on the labels assigned to them. Some of these labels are listed below-

- **Green Label ATMs**- Green Label ATMs refer to Automated Teller Machines (ATMs) that are specifically designated for processing financial transactions related to agricultural activities, such as crop loans, insurance premium payments for crops, and other agricultural banking services. These ATMs are aimed at promoting financial inclusion and supporting rural communities by providing farmers and agricultural workers with convenient access to banking services tailored to their needs. Green Label ATMs often offer features like language options in regional dialects, simplified interfaces, and extended hours of operation to accommodate the agricultural community's requirements. They play a crucial role in fostering economic development in rural areas and improving farmers' access to essential financial services.

- **Yellow Label ATMs**- Yellow Label ATMs are Automated Teller Machines (ATMs) that are strategically located at specific retail locations, such as shopping malls, supermarkets, gas stations, or airports, to cater to customers' banking needs while

they are engaged in retail or commercial activities. These ATMs are typically operated by non-banking entities under agreements with financial institutions and are marked with a yellow label to distinguish them from regular bank-owned ATMs. Yellow Label ATMs offer a convenient and accessible way for customers to withdraw cash, check balances, and perform other basic banking transactions while they are at these retail venues, providing added convenience and flexibility for users who may not have immediate access to their bank's own ATM network.

- **Orange Label ATMs**- Orange Label ATMs are a type of Automated Teller Machines (ATMs) that are designated for providing specialized services, such as government-related transactions, including tax payments, utility bill payments, and other governmental services. These ATMs are often located in government offices, municipal buildings, or other public sector facilities, offering citizens a convenient and efficient way to conduct these specific types of transactions without needing to visit separate payment centers or banks. Orange Label ATMs streamline the process of handling government-related payments, enhancing accessibility and convenience for individuals interacting with governmental services, while also reducing administrative burdens and queues in traditional payment channels.

- **Pink Label ATM**s- Pink Label ATMs are a category of Automated Teller Machines (ATMs) that are specifically designed to cater to the financial needs of women. These ATMs offer a range of services tailored to women's preferences and requirements, such as providing options for privacy during transactions, offering information on women-centric financial products, and promoting financial literacy among female users. Pink Label ATMs are often located in areas with a high concentration of female customers, such as shopping centers, women's colleges, or community centers, to ensure easy access for women from diverse backgrounds. These ATMs play a vital role in empowering women economically by giving them convenient access to banking services and financial resources, ultimately contributing to greater financial inclusion and gender equality in the banking sector.

- **White Label ATMs** – White Label ATMs are Automated Teller Machines (ATMs) that are owned and operated by non-bank entities under licenses obtained from the Reserve Bank of India (RBI) in India. Unlike traditional bank-owned ATMs, white label ATMs are not affiliated with any specific bank and are typically operated by third-party companies or independent service providers. These ATMs offer basic banking services such as cash withdrawals, balance inquiries, and fund transfers, and they are usually located in high-traffic areas such as shopping malls, retail outlets, transportation hubs, and other public spaces. White Label ATMs provide customers with increased convenience and accessibility to banking

services, especially in areas where traditional bank branches may be limited or unavailable, thereby contributing to the expansion of the banking network and financial inclusion efforts.

- **Brown Label Banks**- Brown Label ATMs are a type of Automated Teller Machines (ATMs) that are managed and operated by third-party service providers on behalf of banks in India. Unlike White Label ATMs (WLAs) which are completely owned and operated by non-banking entities, Brown Label ATMs are owned by banks but are outsourced for management and operations to third-party companies. These third-party providers handle tasks such as ATM installation, maintenance, cash replenishment, and customer support, while the banks remain responsible for regulatory compliance, branding, and customer relationships. Brown Label ATMs allow banks to expand their ATM networks cost-effectively and efficiently, especially in remote or high-traffic areas, enhancing customer accessibility to banking services and promoting financial inclusion.

1.3 Uses of an Automated Teller Machine

Automated Teller Machines have revolutionized the banking sector by providing easy access to customers and loading off the burden from bank officials.Some of the uses of an ATM are-

- **Cash Withdrawal:** The primary function of ATMs is to dispense cash. Customers can conveniently withdraw money at any time, day or night, without having to visit a bank branch. This accessibility is particularly useful in emergencies or outside of regular banking hours.
- **Deposits:** Many modern ATMs allow users to deposit cash and checks directly into their accounts. This feature saves time by eliminating the need to visit a branch during operating hours. Some advanced ATMs even accept envelope-free cash and check deposits, offering instant verification and receipt printing.
- **Balance Inquiries:** ATMs provide customers with quick and easy access to their account balances. Users can check their checking, savings, or credit card balances without logging into online banking or visiting a branch. This feature helps individuals keep track of their finances on the go.
- **Transfers:** ATMs enable customers to transfer funds between their accounts linked to the same bank. This functionality allows for convenient movement of

money between checking and savings accounts, making it easier to manage finances and cover expenses.

- **Bill Payments:** Some ATMs allow customers to pay bills directly from their accounts. Users can input bill details and make payments securely without the need for checks or online banking. This feature streamlines the bill-paying process and ensures timely payments.
- **Account Management:** ATMs offer various account management functions, such as changing PINs, printing mini statements, requesting checkbooks, and updating personal information. These services empower customers to perform routine banking tasks independently.
- **Cash Advances:** ATMs linked to credit cards often provide cash advance services. Cardholders can withdraw cash against their credit limit, albeit with associated fees and interest rates. This feature offers convenience in situations where cash is required and credit cards are accepted.
- **Prepaid Card Services:** Many ATMs support prepaid card transactions, allowing users to check balances, reload funds, and view transaction history for prepaid cards such as gift cards, transit cards, or payroll cards. This functionality enhances the versatility of ATMs for various financial needs.
- **Foreign Currency Exchange:** In international travel hubs, ATMs may offer currency exchange services, allowing travelers to withdraw local currency using their debit or credit cards. This convenience eliminates the hassle of carrying large amounts of cash or exchanging currency at unfavorable rates.
- **Accessibility Features:** ATMs are equipped with accessibility features to cater to users with disabilities. These include options for audio instructions, braille keypads, and height adjustments. Such features ensure that banking services are accessible to all customers.

1.4 ATM Fraud

Over the last two decades, automated teller machines (ATMs) have become as much a part of the landscape as the phone booths made famous by Superman.As a result of their ubiquity, people casually use these virtual cash dispensers without a second thought. The notion that something could go wrong never crosses their minds. Most ATM scams involve criminal theft of debit card numbers and personal identification numbers (PINs) from the innocent users of these machines. There are several variations

of this confidence scheme, but all involve the unknowing cooperation of the cardholders themselves.

ATM fraud is described as a fraudulent activity where the criminal uses the ATM card of another person to withdraw money instantly from that account. This is done by using the PIN. The other type of ATM fraud is stealing from the machine in the ATM by breaking in.

FIGURE 1.1: ATM scam

1.4.1 Skimming

This type of ATM scam involves a skimmer device that criminals place on top of or within the card slot. To record your PIN number, the criminals may use a hidden camera or an overlay that covers the original PIN pad. Using the card numbers and PIN's they record; thieves create duplicate cards to withdraw money from consumers' accounts.Unlike losing your debit card or having it stolen, you won't realize anything is amiss until unauthorized transactions take place.

Overlay Skimming: In overlay skimming, criminals place a discreet device, often made to look like a legitimate part of the ATM, over the card reader slot. This device captures the magnetic stripe data from the cards inserted into the ATM. To record PINs, criminals may also install a fake keypad overlay or a hidden camera near the ATM's PIN pad.Scenario: An unsuspecting ATM user visits a standalone ATM in a convenience store. Unbeknownst to them, criminals have installed an overlay skimmer on the card reader slot. When the user inserts their card and enters their PIN, the skimmer captures their card details and PIN. The criminals later use this information to create counterfeit cards and withdraw money from the victim's account.

The Okaloosa County Sheriff's Office in Florida recovered this skimming device on March 6, 2015, at the Hancock Bank in Destin yesterday. (Courtesy/Okaloosa County Sheriff's Office)

FIGURE 1.2: skimming device

Internal Skimming: Internal skimming involves criminals installing a skimming device inside the ATM, usually within the card reader or near the internal components. This type of skimming is more difficult to detect, as the device is hidden from view and doesn't alter the ATM's external appearance.Scenario: A criminal gains unauthorized access to an ATM by posing as a maintenance worker or using specialized tools to open the ATM's casing. They install a small skimming device inside the card reader slot, which captures card data as users insert their cards. Since the skimmer is hidden within the ATM, users are unlikely to notice anything suspicious during their transactions.

Wireless Skimming: Wireless skimming involves the use of wireless technology, such as Bluetooth or Wi-Fi, to remotely collect card data from skimming devices installed on ATMs. Criminals can retrieve the stolen card information wirelessly without physically accessing the skimmer, making it more challenging for law enforcement to track down the perpetrators.Scenario: Criminals install a wireless skimming device inside an ATM, equipped with Bluetooth technology. As users insert their cards into the ATM, the skimmer collects their card data and transmits it wirelessly to the criminals' mobile devices located nearby. The criminals can then use this stolen information to create counterfeit cards or make unauthorized purchases online.

ATM Camera Skimming: In ATM camera skimming, criminals install hidden cameras near ATMs to capture users' PINs as they enter them on the keypad. This technique is often used in conjunction with other skimming methods to obtain both card data and PINs.Scenario: Criminals mount a small, inconspicuous camera near the ATM's keypad, disguised as part of the ATM surroundings. When users enter their PINs to complete

transactions, the camera records their keystrokes. Meanwhile, a separate skimming device installed on the card reader captures the magnetic stripe data from users' cards. By combining the stolen card information and PINs, the criminals can access victims' accounts and withdraw funds.

1.4.2 Shimming

This is the latest update to skimming. Instead of reading your card number, criminals place a shimming device deep inside the ATM to record your card's chip information. The end result is the same as skimming because thieves use the stolen chip data to create "cloned" versions of your debit card.

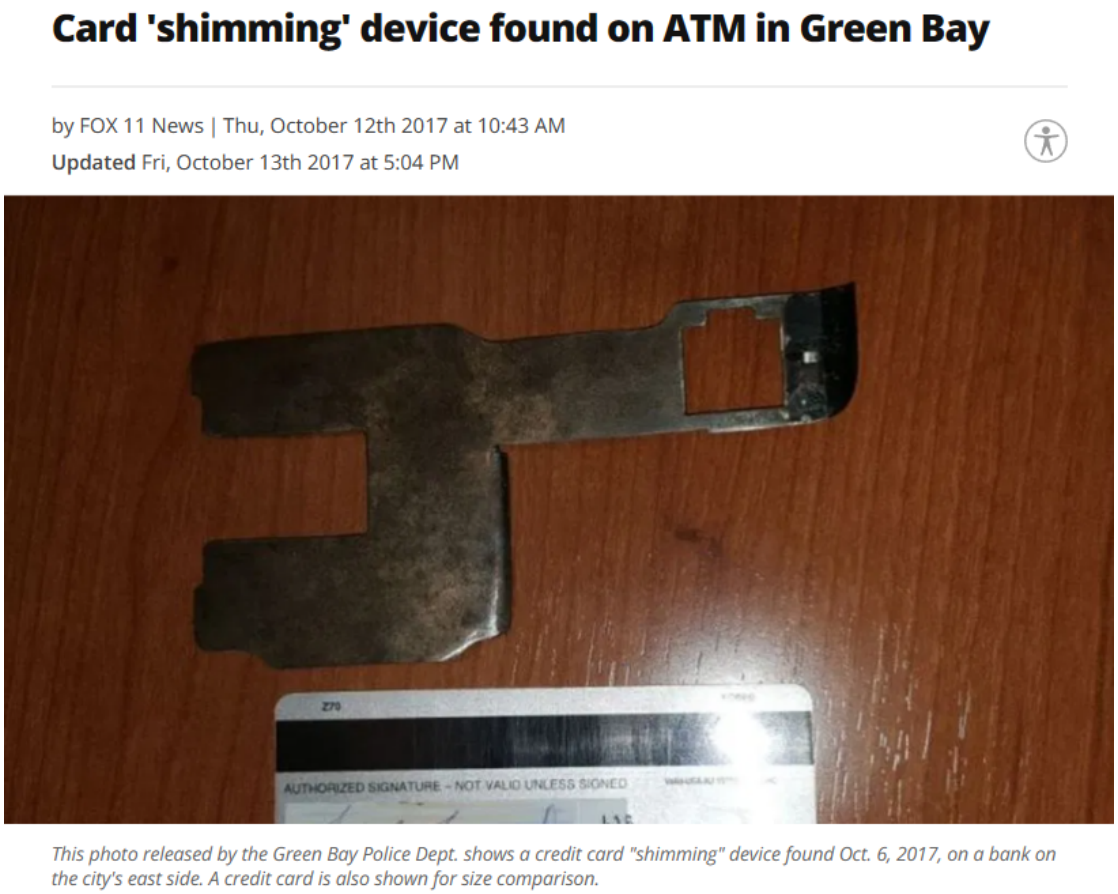


FIGURE 1.3: Enter Caption

EMV Chip Reader Shimming: In this type of shimming, criminals insert a shimmer device into the card reader slot of an ATM. The shimmer is designed to sit between the card and the ATM's EMV chip reader, allowing it to intercept and record the data transmitted between the chip and the ATM during a transaction.Scenario: A criminal gains access to an ATM and installs a shimmer device inside the card reader slot. When

an unsuspecting ATM user inserts their chip-enabled card to withdraw cash, the shimmer captures the data exchanged between the card's EMV chip and the ATM. The criminal can then retrieve the stolen chip data from the shimmer and use it to create counterfeit cards or make fraudulent transactions.

Wireless Shimming: Similar to wireless skimming, wireless shimming involves the use of Bluetooth or other wireless technology to remotely collect data from shimming devices installed on ATMs. Criminals can retrieve the stolen chip data wirelessly without physical access to the ATM, making detection and prevention more challenging.Scenario: Criminals install a wireless shimming device inside an ATM's card reader slot, equipped with Bluetooth technology. As users insert their chip-enabled cards into the ATM, the shimmer captures the data transmitted between the card's chip and the ATM. The stolen chip data is then transmitted wirelessly to the criminals' mobile devices, allowing them to create cloned cards or conduct fraudulent transactions remotely.

Combo Skimming and Shimming: Some criminals employ a combination of skimming and shimming techniques to steal both magnetic stripe and chip data from cards. This hybrid approach allows them to capture data from all types of cards, regardless of whether they use magnetic stripes or EMV chips for transactions.Scenario: Criminals install a dual-purpose device inside an ATM's card reader slot, capable of both skimming magnetic stripe data and shimming chip data. When users insert their cards into the ATM, the device captures data from both the magnetic stripe and the EMV chip simultaneously. This comprehensive data capture enables the criminals to create cloned cards with magnetic stripe and chip information, maximizing their ability to conduct fraudulent transactions.

1.4.3 Cash-out

This scam targets multiple accounts from the same financial institution. Armed with a hacked bank employee's credentials, the criminal alters account balances and withdrawal limits. Using stolen debit card numbers captured from a separate skimming attack, they can "cash out" the ATM until it's out of money.

Hacked Employee Credentials: In this scenario, criminals obtain login credentials of bank employees through hacking or phishing techniques. Once they gain access to the bank's internal systems, they can alter account balances and withdrawal limits without raising suspicion.Scenario: A criminal compromises the email account of a bank employee through a phishing attack. Using the employee's credentials, the criminal gains access to the bank's backend systems. They then manipulate the account balances and withdrawal limits of multiple accounts to increase the available funds for withdrawal. Concurrently,

the criminal uses stolen debit card numbers obtained from previous skimming attacks to withdraw cash from ATMs until the funds are depleted.

Insider Complicity: In some cases, insiders within the financial institution may collude with criminals to carry out cash-out scams. These insiders may provide unauthorized access to bank systems or deliberately alter account information to facilitate fraudulent transactions.Scenario: A disgruntled employee at a financial institution conspires with an external criminal group to carry out a cash-out scam. The insider provides the criminals with access to the bank's systems, allowing them to modify account balances and withdrawal limits. The criminals then use stolen debit card numbers obtained through various means to withdraw cash from ATMs, exploiting the altered account information to maximize their gains.

Malware Attacks: Malware can be used to compromise banking systems and manipulate account data to facilitate cash-out scams. Malicious software installed on bank servers or employee computers can alter account balances and transaction records, enabling criminals to conduct fraudulent withdrawals without detection.Scenario: Criminals infect a financial institution's network with sophisticated malware designed to target banking systems. The malware infiltrates the bank's servers and alters account balances and withdrawal limits, making it appear as though legitimate transactions have occurred. The criminals then use stolen debit card numbers to withdraw cash from ATMs, exploiting the manipulated account data to facilitate their fraudulent activities.

Third-Party Payment Processors: Criminals may exploit vulnerabilities in third-party payment processors or financial services providers to carry out cash-out scams. By gaining unauthorized access to these systems, they can manipulate account data and conduct fraudulent transactions.Scenario: Hackers breach the security of a third-party payment processor used by multiple financial institutions. They manipulate account data within the processor's systems, altering account balances and withdrawal limits for numerous accounts across different banks. The criminals then use stolen debit card numbers to withdraw cash from ATMs, exploiting the compromised account information to facilitate their fraudulent activities.

1.4.4 Jackpotting:

While there are multiple types of jackpotting attacks, typically, these incidents involve gaining physical access to the inside of the machine. The criminals may replace hardware or install malicious software giving them control of the cash dispensing function. Jackpotting is similar to a cash out scam, but it does not require the criminal to have any customer account details or stolen debit card information.

Direct Access Jackpotting: In direct access jackpotting, criminals gain unauthorized physical access to the ATM by either breaking into the machine or compromising its security mechanisms. Once inside, they manipulate the ATM's hardware components or install malicious devices to force the machine to dispense cash.Scenario: A group of criminals targets an ATM located in a remote area with minimal surveillance. Using specialized tools or brute force methods, they gain access to the inside of the ATM and connect a laptop or other electronic device to its control panel. The criminals then install malware or manipulate the ATM's hardware to trigger a jackpotting event, causing the machine to dispense cash uncontrollably until it is emptied.

Black Box Attack: In a black box attack, criminals connect an unauthorized device, known as a "black box," to the ATM's cash dispenser mechanism. The black box mimics the signals sent by the ATM's internal systems, tricking the machine into dispensing cash without the need for authentication or authorization.Scenario: Criminals gain access to the back of an ATM by exploiting vulnerabilities in its physical security or by distracting maintenance personnel. They connect a black box device to the ATM's cash dispenser mechanism, which is hidden from view. The black box sends commands to the ATM's dispenser, causing it to release cash continuously until the criminals retrieve the stolen funds.

Malware-Based Jackpotting: In malware-based jackpotting attacks, criminals infect the ATM's internal systems with malicious software, typically through a compromised network connection or physical access to the machine's control panel. The malware allows the criminals to remotely control the ATM and force it to dispense cash on command.Scenario: Criminals gain access to the ATM's internal network by exploiting vulnerabilities in its software or by physically connecting a laptop to the machine's control panel. They install malware on the ATM's operating system, which remains undetected by security measures. Using remote access tools, the criminals execute commands to trigger the ATM's cash dispenser, causing it to dispense cash without authorization until it is depleted.

Exploiting Manufacturer Backdoors: In some cases, criminals exploit backdoor access mechanisms or security vulnerabilities built into the ATM's manufacturer software to gain unauthorized control over the machine's functions. This allows them to manipulate the ATM's operation and force it to dispense cash.Scenario: Criminals discover a security vulnerability in the software used by a specific ATM manufacturer. They exploit this vulnerability to gain remote access to the ATM's control systems, bypassing authentication measures. With control over the ATM's functions, the criminals initiate a jackpotting attack, causing the machine to dispense cash until it is emptied.

1.5 Evolution of ATM Security System

The evolution of ATM security systems has been driven by technological advancements, regulatory requirements, and the need to combat increasingly sophisticated forms of fraud and security threats.

Physical Security Measures (1970s-1980s): In the early years of ATMs, security primarily focused on physical measures to prevent theft and vandalism. This included installing ATMs in secure locations, such as bank branches or well-lit areas, and implementing features like reinforced enclosures, tamper-evident seals, and surveillance cameras to deter criminals.

PIN-Based Authentication (1980s-1990s): The introduction of Personal Identification Numbers (PINs) in the 1980s significantly enhanced ATM security by adding an additional layer of authentication. Customers were required to enter a unique PIN associated with their bank cards to access their accounts and perform transactions. This measure helped prevent unauthorized access to ATM services and reduce instances of card theft and fraud.

Encryption and Secure Communication Protocols (1990s-2000s): With the increasing use of electronic transactions and the expansion of ATM networks, encryption and secure communication protocols became essential for protecting sensitive data transmitted between ATMs and banking systems. Advanced encryption algorithms were implemented to secure PINs, account numbers, and other confidential information transmitted during ATM transactions, reducing the risk of interception and unauthorized access by cybercriminals.

EMV Chip Technology (2000s-Present): The adoption of EMV (Europay, Mastercard, and Visa) chip technology in the 2000s represented a significant milestone in ATM security. EMV chips, embedded in debit and credit cards, generate unique transaction codes for each transaction, making it more difficult for fraudsters to clone cards or steal sensitive cardholder data. EMV compliance has become a global standard for ATM security, helping to reduce counterfeit card fraud and unauthorized transactions.

Anti-Skimming and Fraud Detection Measures (2010s-Present): Skimming, where criminals install devices to capture card data from unsuspecting ATM users, has been a persistent threat to ATM security. To combat this, ATM manufacturers and financial institutions have developed anti-skimming technologies, such as tamper-resistant card readers, encrypted PIN pads, and physical barriers to protect against skimming devices. Additionally, real-time fraud detection systems analyze ATM transactions for

suspicious patterns and anomalies, enabling proactive measures to prevent fraudulent activity and mitigate financial losses.

Biometric Authentication and Cardless Transactions (2010s-Present): Biometric authentication methods, such as fingerprint scanning and facial recognition, have been increasingly integrated into ATM systems to enhance security and convenience. Biometric authentication adds an additional layer of identity verification beyond PINs or cards, reducing the risk of unauthorized access and fraud. Furthermore, the introduction of cardless ATM transactions, where users can authenticate themselves using mobile devices or biometric data instead of physical cards, offers added security and flexibility for ATM users.

Regulatory Compliance and Industry Standards: Regulatory bodies and industry organizations play a crucial role in shaping ATM security standards and best practices. Regulatory requirements, such as the Payment Card Industry Data Security Standard (PCI DSS) and the EMV liability shift, establish guidelines for ATM security and mandate compliance with specific security measures to protect cardholder data and mitigate fraud. Moreover, industry collaborations and information-sharing initiatives help financial institutions and ATM operators stay abreast of emerging security threats and adopt effective security measures to safeguard ATM networks and customer assets.

1.5.1 Problem Identified

Nowadays, crimes at ATMs have become an alarming issue. Security for the customer's account is not guaranteed by PIN. Many people, who aren't familiar with the concept of PIN are unlikely to memorize and recognize it. There are many people who mistrust PIN, such as, if they have lost their card, they would feel unsafe that their account could be accessed by others and they would lose all their money.

1.5.2 AI with IoT

Individually, the Internet of Things (IoT) and Artificial Intelligence (AI) are powerful technologies. When you combine AI and IoT, you get AIoT—the artificial intelligence of things. You can think of internet of things devices as the digital nervous system while artificial intelligence is the brain of a system. To fully understand AIoT, you must start with the internet of things. When "things" such as wearable devices, refrigerators, digital assistants, sensors and other equipment are connected to the internet, can be recognized by other devices and collect and process data, you have the internet of things

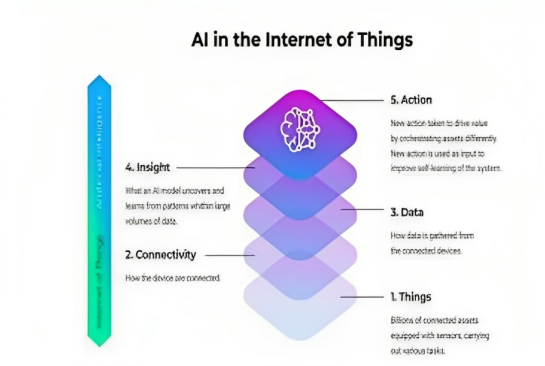


FIGURE 1.4: AI with IoT

Artificial intelligence is when a system can complete a set of tasks or learn from data in a way that seems intelligent. Therefore, when artificial intelligence is added to the internet of things it means that those devices can analyze data and make decisions and act on that data without involvement by humans. These are "smart" devices, and they help drive efficiency and effectiveness. The intelligence of AIoT enables data analytics that is then used to optimize a system and generate higher performance and business insights and create data that helps to make better decisions and that the system can learn from.

1.5.2.1 Real-World Examples of AI Embedded IoT Devices

- Traffic Management
- Self- Driving Cars
- Smart Homes
- Body Sensors
- Face Detection

1.5.2.2 IoT and Machine Learning

IoT is the data "supplier" while machine learning is the data "miner". To make the data supplied by IoT work, it needs to be refined. Dozens of IoT sensors and external factors are producing a myriad of data points. The "miner's" task here is to identify correlations between them, extract meaningful insight from these variables and transport it to the storage for further analysis.

1.5.3 Deep Learning

Deep learning attempts to mimic the human brain—albeit far from matching its ability—enabling systems to cluster data and make predictions with incredible accuracy. Deep learning is a subset of machine learning, which is essentially a neural network with three or more layers. These neural networks attempt to simulate the behavior of the human brain—albeit far from matching its ability—allowing it to "learn" from large amounts of data. While a neural network with a single layer can still make approximate predictions, additional hidden layers can help to optimize and refine for accuracy.

1.6 Objective

To Face recognition can be used to secure ATM transaction and is used as a tool for authenticating users to confirm the card owner. Financial fraud is a very important problem for Banks and current secure information in the ATM card magnetic tape are very vulnerable to theft or loss. By using face recognition as a tool for authenticating users in ATMs can be confirmed as the card owner. Face Based ATM login Process the ATMs which are equipped with Face recognition technology can recognize the human face during a transaction.

CHAPTER 2

LITERATURE REVIEW

2.1 Biometric And IOT Technology Based Safety Transactions In ATM[1]

The described system represents a technological evolution in ATM functionality, addressing historical challenges and emphasizing advancements in security measures. Although specific citations are not provided, the narrative aligns with the broader historical context of ATM development, highlighting the necessity for enhanced security features.

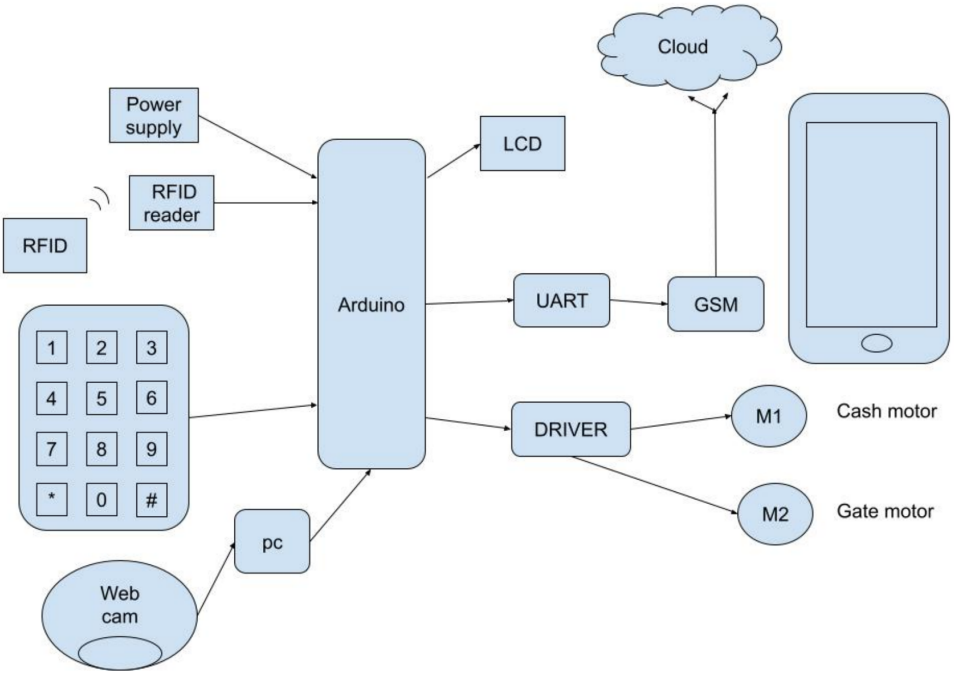
The integration of biometric technologies stands out as a key aspect of the system's security enhancements. Mention of facial recognition through a Passive Infrared (PIR) sensor [6]implies the system's reliance on biometrics for user authentication and security alerts. This resonates with established literature emphasizing the role of biometrics, including facial recognition, in fortifying ATM transaction security.

The introduction of the GAMMA Concept suggests a structured framework for implementing security measures at different levels. While specific references are absent, this

concept likely draws inspiration from general discussions on multi-level security protocols across diverse contexts, reflecting a holistic approach to addressing security concerns.

Additionally, the system's implementation of a multibanking transaction system, incorporating Iris Recognition for heightened security at different stages, aligns with established research emphasizing the reliability and security advantages of Iris Recognition technology.

The integration of face recognition technology with the Internet of Things (IoT) is presented as a contemporary strategy for real-time data collection and verification. Though



lacking specific citations, this aligns with emerging trends that leverage facial recognition and IoT for enhanced security in various applications.

The system concludes by detailing the simulation of the proposed system in Proteus and its subsequent hardware implementation. Although specific references are not provided, this methodology aligns with common practices in the development and testing of electronic systems. In summary, the passage provides an overview of the system's evolution, referencing relevant trends in biometrics, multi-level security frameworks, and IoT technologies, without specifying citations. Researchers interested in exploring these concepts further are encouraged to consult existing literature on biometric security, multi-level security frameworks, and IoT applications in financial systems.

2.2 Face Biometric Authentication System for ATM using Deep Learning[2]

Deep Convolutional Neural Networks (DCNNs) are a type of neural network architecture that are particularly effective for image processing tasks, including face recognition. They work by applying convolution operations to the input image, which helps to extract features and patterns from the image data.

The DCNN architecture consists of multiple layers, including convolutional layers, pooling layers, and fully connected layers. The convolutional layers are responsible for feature

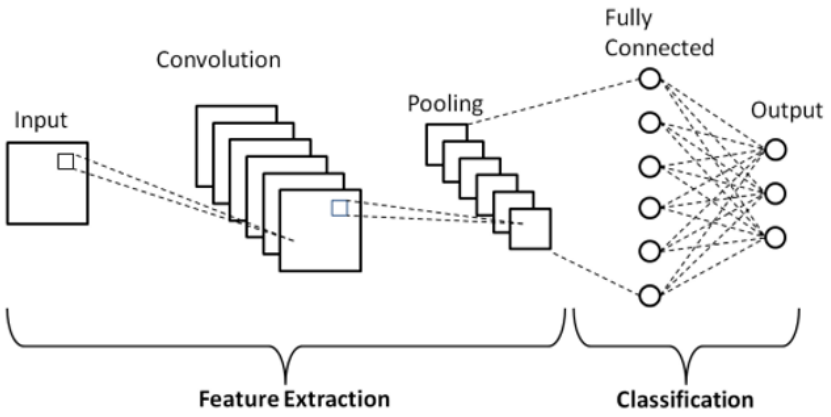


FIGURE 2.1: Layers of CNN

extraction, where each layer learns to recognize different features in the image data. The pooling layers reduce the dimensionality of the data, which helps to reduce overfitting and computational complexity. The fully connected layers are used for classification, where the final output is generated.

In the context of face recognition, the DCNN is trained to recognize faces by learning to associate the extracted features with the identity of the person in the image. The trained model can then be used to recognize faces in new images by extracting the same features and comparing them with the learned representations of known faces.

The DCNN has the ability to generate robust face representations that can be analyzed in a multidimensional 'face space'. This 'face space' organizes the viewpoint, illumination, gender, and identity of faces. Natural image variation is organized hierarchically, with face identity nested under gender, and illumination and viewpoint nested under identity. This organization allows the DCNN to recognize faces across variations of viewpoint, illumination, expression, and appearance.

In the proposed system for ATM face biometric authentication, the DCNN could be used to extract features from the captured images of the user's face and compare them with the stored representations of the authorized user's face. If the extracted features match the stored representations, the user is granted access to the ATM.

2.3 A Face Recognition Method Based on CNN[3]

In recent years, deep learning, particularly Convolutional Neural Networks (CNNs), has emerged as a powerful paradigm for face recognition tasks. This section reviews key aspects related to CNN training experiments for face recognition, focusing on dataset

selection, data preprocessing, activation function choices, and the establishment of CNN models.

2.3.1 Dataset Selection:

The choice of a suitable dataset is crucial for effective CNN training in face recognition. The study opts for the OlivettiFaces face database from New York University, containing 400 grayscale images of 40 individuals, each with 10 images. The images' pixel values range from 0 to 255, and the dataset is divided into training, validation, and test sets. This dataset provides a diverse range of facial features, making it suitable for training robust face recognition models.

2.3.2 Data Preprocessing:

Data preprocessing plays a pivotal role in preparing the dataset for CNN training. The experimental setup involves Windows7, Python 3.6, and Keras. The images are loaded using the Python Imaging Library (PIL), normalized using NumPy, and converted into one-dimensional vectors with added labels. The resulting normalized values and labeled vectors are stored using the pickle module, facilitating efficient data handling during training.

2.3.3 Activation Function Selection:

The choice of activation functions significantly impacts the performance of neural networks. In this experiment, Rectified Linear Unit (ReLU) is chosen as the activation function for neurons in the CNN. ReLU, with its simple mathematical formulation (f(x) = max(0, x)), provides advantages over traditional sigmoid or tanh functions. It helps mitigate the vanishing gradient problem during backpropagation, promoting efficient training in deep neural networks.

2.3.4 Establishment of CNN Model:

Constructing an effective CNN model involves a sequential stacking of layers. The experimental model follows a structure of two convolutional layers (16-36) and a hidden layer with varying numbers of neurons (128, 256, 512, 1024). The choice of ReLU as the activation function, max-pooling layers, and dropout layers contributes to the model's ability to capture complex features, prevent overfitting, and enhance generalization.

2.3.5 Experimental Results and Analysis:

The experiment evaluates the impact of changing the number of hidden layer neurons and the number of feature maps in convolutional layers on the CNN model's performance. The results highlight that increasing the number of hidden layer neurons improves accuracy on the training set, while stability is achieved for the verification and test sets. Similarly, varying the number of feature maps influences the running time but has limited impact on recognition rates, demonstrating the robustness of the designed CNN model.

The literature review emphasizes the significance of dataset selection, meticulous data preprocessing, thoughtful activation function choices, and the careful construction of CNN models in the context of face recognition. As experiments delve into optimizing parameters like hidden layer neurons and feature maps, the goal remains to strike a balance between model complexity, computational efficiency, and recognition accuracy. Future advancements will likely build upon these foundations, pushing the boundaries of CNN-based face recognition systems.

2.4 Simultaneous Face Detection and Pose Estimation Using Convolutional Neural Network Cascade[4]

The research presented in this paper introduces a cutting-edge approach to simultaneous face detection and pose estimation using a multi-task Convolutional Neural Network (CNN) cascade framework. The primary focus is on achieving real-time performance while maintaining competitive accuracy. The study conducts a thorough empirical evaluation on two prominent benchmark datasets, the Face Detection Data Set and Benchmark (FDDB) and the Annotated Faces in the Wild (AFW), comparing the proposed framework against state-of-the-art methods.[7]

The face detection results are presented first, where the FDDB dataset is employed to evaluate the framework's performance. This dataset is renowned for its challenging characteristics, featuring images with diverse visual elements, such as occlusions, extreme poses, and low-resolution faces. The evaluation metrics, Discrete Score (DS) and Continuous Score (CS), assess the framework's ability to detect faces based on Intersection over Union (IoU) ratios. The evaluation includes Receiver Operating Characteristic (ROC) curves, providing a comprehensive comparison with established methods like ACF-multiscale, HyperFace, and Faster RCNN. The results highlight the effectiveness of the proposed method, outperforming many reported algorithms and demonstrating its robustness across challenging scenarios.

Moving on to pose estimation, the AFW dataset is employed, a widely used benchmark for face alignment algorithms. The dataset encompasses images collected from Flickr, featuring faces with absolute yaw degree up to 90◦ and significant variations in pose. The evaluation metrics involve cumulative error distribution curves for absolute yaw angles. Comparative analysis with various pose estimation approaches, including Multi. AAM, Multiview HoG, FaceDPL, HyperFace, and Kepler, showcases the proposed method's superiority. The results reveal that the method outperforms several approaches and competes favorably with state-of-the-art deep-learning-based methods.

Ablation experiments are conducted to gain deeper insights into the correlation between face detection and pose estimation tasks. The study explores the impact of joint training of face detection and pose estimation, removing pose regression loss from the total loss. This experiment demonstrates the positive contribution of joint training, showcasing an improvement in face detection performance. Additionally, the study investigates the CNN feature fusion strategy, altering the structure of the last CNN in the baseline framework. The results indicate that while face detection task gains marginal improvement, pose estimation performance experiences a significant boost. This emphasizes the importance of fusing features from different layers in enhancing the performance of structure-dependent tasks like pose estimation.

The paper also addresses the crucial aspect of runtime efficiency, an essential consideration for real-world applications. Leveraging a cascade-based structure, the proposed method demonstrates superior efficiency compared to other methods designed for multiple face-related tasks, including face detection and pose estimation. The study evaluates the method's runtime on VGA images, achieving an impressive 30 frames per second. Comparative analyses with other methods, such as HyperFace and All-in-One, underscore the efficiency and competitiveness of the proposed framework.

In conclusion, the paper presents a holistic and innovative approach to simultaneous face detection and pose estimation, showcasing a multi-task CNN cascade framework. Through extensive empirical evaluations on challenging datasets, the proposed method establishes its effectiveness and efficiency. The real-time performance, achieved through multi-task learning and cascade structure, positions the framework as a promising solution for simultaneous face-related tasks in various real-world applications. The ablation studies offer valuable insights into the interplay between face detection and pose estimation, emphasizing the significance of joint training and feature fusion. Overall, the research contributes significantly to the field of computer vision, particularly in the realm of facial analysis and multi-task learning

2.5 Fairness in Biometrics: a figure of merit to assess biometric verification systems[5]

This section presented a case study using the proposed Fairness Discrepancy Rate to assess demographic differentials. Experiments with three open-source FR baselines based on DCNNs and one COTS system were used along with three databases where gender and racial differentials were studied. We could notice that FDR could summarize and compare the demographic differentials concerning FMR and FNMR between several FR systems.

Furthermore, the Area Under FDR gives a single scalar estimate of such differentials where further rankings can be made. With this assessment, we could notice that the current state-of-the-art Face Recognition systems based on ArcFace loss are fairer than the state-of-the-art from a few years ago and with the evaluated COTS system. Worth noting that the DCNNs based on ArcFace do not have any fairness constraints.We can hypothesize that the margins imposed in the ArcFace loss play a role in minimizing within-class variability and maximizing between-class separation (embeddings from the same identity tends to be more compact than DCNNs training "vanilla" cross-entropy loss). This possibly impacted the scoring behavior allowing us to "safely" use fair decision thresholds. We could observe in with the MORPH dataset that false alarms are more frequent with Asian, Hispanic, and Black subjects The same trends were observed with female subjects using the Mobio dataset. We can also observe the number of false alarms using the comparison scores between nonhomogeneous samples (gallery and probe samples from different demographics) is substantially lower than homogeneous samples (gallery and probe samples from the same demographic groups).

FDR can be trivially extended to other biometric recognition tasks, such as closed-set or opened-set identification. For instance, for closed-set identification, it is possible to compute the differentials between the rank-n of different demographic groups as a figure of merit of fairness. For opened-set, it is possible to stablish an aggregation figure of merit between Detection and Identification Rate (DIR) and False Alarm Rate (FAR) for different demographic groups. Further assessment of those extensions for identification will be carried out as future work. This work introduced the Fairness Discrepancy Rate (FDR) to assess demographic differentials in biometric verification systems. FDR tackles a threshold problem, which is the main issue of how the biometric community addresses such differentials. A substantial amount of works in the biometrics community assess demographic differentials in verification systems by comparing DET curves or ROC curves of different demographic groups separately. This type of comparison assumes that decision thresholds are demographic-specific, which is not feasible or ethical

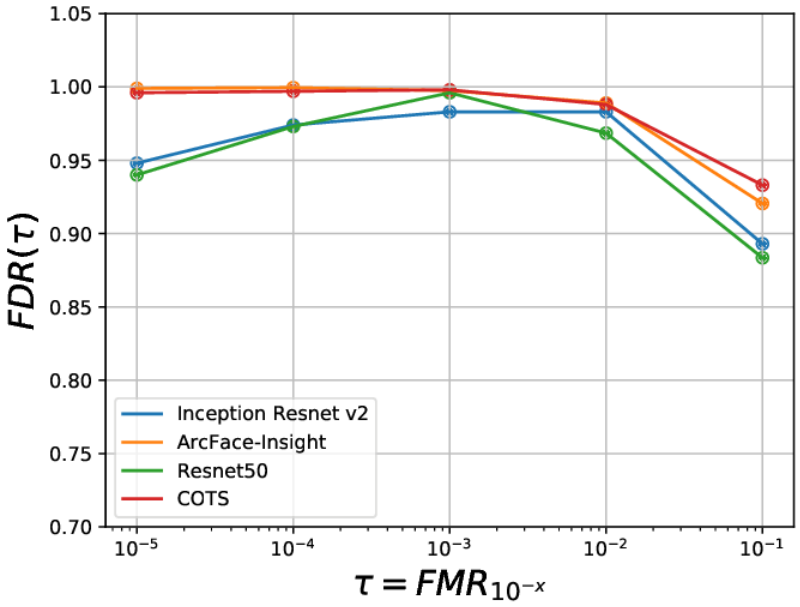


FIGURE 2.2: MOBIO: Fairness Discrepancy Rate of different face verification systems for different decision thresholds

in operational conditions. FDR addresses that by assessing demographic differentials assuming single decision thresholds.

In this work, we consider that fair biometric recognition systems are fair if a decision threshold is "fair" for all demographic groups concerning FMR and FNMR and FDR proxies this behavior. Furthermore, the FMR and FNMRtrade-off with respect to fair behavior can be set by addressing the value . Finally, the Area Under FDR provides a general overview of demographic differentials under a range of decision thresholds and allows a quick comparison between different biometric verification systems. Two groups of experiments were carried out to evaluate this new figure of merit. In the first one, a case study using synthetic data was presented, and it was demonstrated how FDR behaves in extreme cases of fair and unfair scenarios. In the second, a case study using four different face verification systems and three databases was carried out. We could observe via the FDR plots that all evaluated face verification systems presents gender and racial biases to some degree.

Furthermore, it was possible to quickly compare different face recognition systems concerning their demographic discrepancies using the Area Under FDR. Worth noting that neither FDR nor Area Under FDR is direct proxies for how "accurate" a biometric verification system is. Possible error rates have to be analyzed in parallel to picture the trade-off between accuracy vs. fairness fully. We also briefly presented a possible trivial

extension of such a figure of merit to closed and opened-set identification problems. Further work in this direction will be carried out. For reproducibility purposes of the work, all the source code, trained models, and recognition scores are made publicly available. We hope that these tools are useful for the biometrics community to assess demographic differentials. Hence, the demographic differential can be easily assessed as any other figure of merit, such as FMR or FNMR.

CHAPTER 3

SYSTEM ANALYSIS

3.1 Objective

The primary objective of incorporating face recognition technology into ATM transactions is to bolster security measures. Traditional methods of relying on information stored on magnetic stripes make ATMs vulnerable to theft or loss[8]. To counteract this vulnerability, the proposal suggests implementing face recognition as an additional layer of authentication, providing a more robust and secure means of confirming the identity of users during transactions.

Financial fraud poses a significant challenge for banks, and the proposed use of face recognition aims to mitigate this risk. By confirming the cardholder's identity through facial recognition during ATM transactions, the likelihood of unauthorized access and fraudulent activities can be substantially reduced, thereby enhancing overall security in the banking sector.

The implementation involves integrating face recognition technology into the existing ATM infrastructure. This includes the deployment of advanced hardware, such as cameras capable of capturing and processing facial images in real-time, along with sophisticated software for facial recognition algorithms. The technology is designed to continuously monitor users during transactions, offering an additional layer of security against potential threats.

During a transaction, the ATM equipped with face recognition captures the user's facial features, and the captured image is then compared with the pre-registered facial data stored securely in the system. If there is a match, the user is authenticated, and the transaction is allowed to proceed. This process not only ensures card ownership verification but also simplifies the user experience by eliminating the need for physical cards or PINs.

The adaptability and scalability of face recognition technology make it feasible for widespread implementation across ATMs. This ensures that the technology can be seamlessly integrated into existing ATM networks and expanded as needed to cover a broader range of ATMs. In addition to enhancing security, face-based authentication contributes to a more convenient and user-friendly transaction process, aligning with the evolving expectations of users in the digital age.

3.2 Existing System

- **Existing ATM authentication method is the use of password-PINs and OTP.**

The existing system for ATM authentication primarily relies on traditional methods, including password-PINs and OTPs. ATMs commonly utilize access cards that have a magnetic stripe (magstripe) and a fixed Personal Identification Number (PIN) for identity verification. In some cases, an alternative approach involves the use of a chip embedded in the card along with a PIN. The chip serves as a more secure means of authentication, and in situations where it malfunctions, a magstripe is included as a backup for identification purposes.

An additional feature in the existing system is the introduction of QR cash withdrawals. This innovation allows customers to forego the use of physical ATM cards. Instead, they can use a QR app to scan a QR code displayed on the ATM, facilitating cash withdrawals. This method adds a layer of convenience for users who may prefer a cardless transaction experience.

To enhance security, certain ATM systems have incorporated a combination of fingerprint and GSM (Global System for Mobile Communications) technology into the existing PIN-based authentication process. This integration aims to provide an additional layer of biometric verification, utilizing fingerprints alongside traditional PINs for identity confirmation.

The biometric authentication algorithms employed in the existing system include various techniques such as Gaussian Mixture Models (GMMs), Artificial Neural Networks (ANNs), Fuzzy Expert Systems (FESs), Support Vector Machines (SVMs), Linear Discriminant Analysis (LDA), and Principal Component Analysis (PCA). These algorithms are used to process and analyze biometric data, enhancing the accuracy and reliability of user authentication.

However, despite these advancements, there are notable limitations in the existing system. For instance, the reliance on PINs and the potential vulnerabilities associated with magstripes pose security concerns. Additionally, the system may face challenges such as slow processing speeds during face detection and training data loading, limited detection ranges for certain biometric methods, and security issues associated with unimodal biometric systems

3.2.1 Disadvantages

- Accuracy Concerns: The system's accuracy is not 100%,indicating that there is a potential margin of error in the authentication process. This limitation could lead to instances of false positives or false negatives, compromising the reliability of the system and potentially allowing unauthorized access.
- Slow Face Detection and Training Data Processes: The face detection and loading training data processes are reported to be slow. This sluggishness can impact the efficiency of the authentication system, causing delays in user verification. Slow processing times may lead to a less-than-optimal user experience and could be a critical factor in high-traffic scenarios.
- Limited Face Detection Range: The system can only detect faces from a limited distance. This limitation may pose challenges in situations where users are not positioned within the optimal range for face detection. Users who are too close or too far from the camera may experience difficulty in having their faces accurately recognized, potentially leading to authentication failures.
- Inability to Repeat Live Video for Missed Faces: The system lacks the capability to repeat live video to recognize missed faces. In scenarios where the initial face detection fails or is obstructed, the system does not have the ability to review live video feeds to identify and authenticate missed faces. This limitation reduces the system's resilience in capturing all valid user attempts.
- Challenges of Unimodal Biometric Systems: Unimodal biometric systems, which rely on a single mode of authentication (such as facial recognition), face several challenges. These challenges include dealing with noisy data, variations within the same class of individuals, restricted degrees of freedom in feature representation, non-universality (difficulty in capturing unique features for everyone), and susceptibility to spoof attacks, where unauthorized users attempt to deceive the system using fake biometric data.

- Security Concerns and Potential Increase in Criminal Activities: The method is considered not very secure, and there are concerns about its susceptibility to an increase in criminal activities. This lack of perceived security could be attributed to the vulnerabilities associated with the existing biometric and authentication technologies. If the system is not robust enough, it may become a target for exploitation by individuals with malicious intent.

3.3 Proposed System

This project proposes an automatic teller machine multi modal security model that would combine a physical access card and electronic facial recognition using Deep Convolutional Neural Network.

- **Facial Biometric Authentication System using Deep Learning Techniques**

Facial biometric authentication systems employing deep learning techniques have emerged as a powerful and accurate solution within the realm of artificial intelligence[9]. Deep learning, a subset of machine learning, focuses on neural networks with multiple layers, allowing the automatic learning of intricate patterns and features from raw data. When applied to face recognition, deep learning surpasses traditional machine learning methods in accuracy and effectiveness.

Facial biometric authentication revolves around the unique identification of individuals based on their facial features. The inherent complexity and variability of facial characteristics pose challenges for traditional machine learning approaches. However, deep learning models, particularly convolutional neural networks (CNNs), excel in capturing and understanding these complex patterns. This capability enables the development of highly accurate facial recognition systems.

One of the key advantages of deep learning in facial recognition lies in its ability to perform end-to-end learning. This means that the model can take raw facial images as input and autonomously learn to map these inputs to the desired outputs, eliminating the need for manual feature engineering. Deep learning models, through feature learning and hierarchical representation, adapt well to the diversity of facial features, ultimately enhancing the accuracy of recognition tasks.

Compared to traditional methods that often require manual feature engineering, deep learning techniques offer a more sophisticated approach. Convolutional Neural Networks, as a prominent example, automatically learn spatial hierarchies of features in

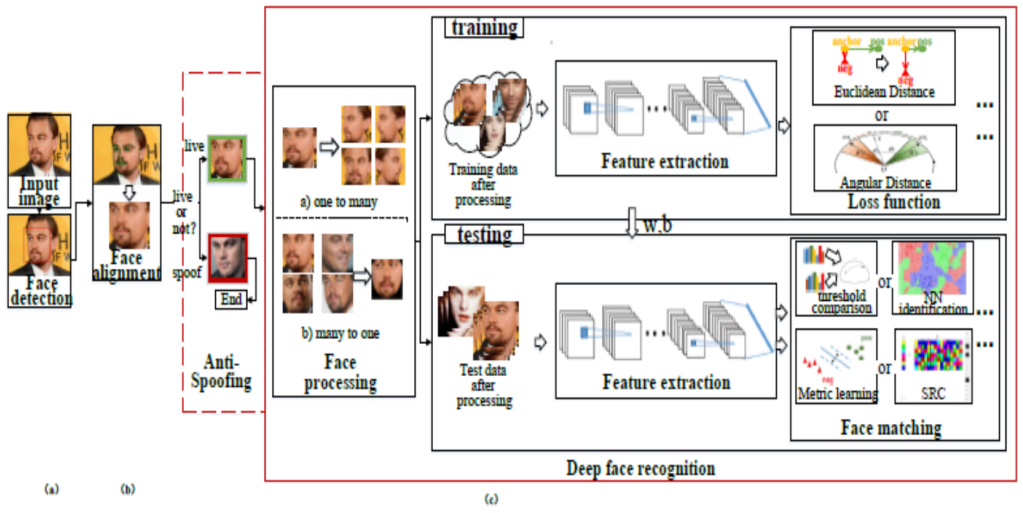


FIGURE 3.1: Deep Face Recognition

facial images. Additionally, Siamese Networks specialize in discerning similarities and dissimilarities between pairs of facial images, making them suitable for verification tasks. Generative Adversarial Networks (GANs) further contribute by generating realistic facial images, aiding in data augmentation and improving model robustness.

- **Unknown Face Verification Link Generator**

The concept of an "Unknown Face Verification Link Generator"[10] introduces a sophisticated layer of security in scenarios where the stored facial image and a newly captured image do not match. This discrepancy suggests the presence of an unauthorized user attempting to access a system or conduct a transaction. In response to this potential security threat, a Face Verification Link is generated and sent to the user for identity verification. This process leverages dedicated artificial intelligence (AI) agents, specifically designed for remote certification, to either authorize the transaction appropriately or trigger a security-violation alert within the banking security system.

When a mismatch occurs between the stored and captured facial images, it signifies a potential security breach. The Unknown Face Verification Link Generator acts as a proactive measure, providing an additional layer of authentication beyond traditional methods. By utilizing AI agents, the system can dynamically adapt to emerging threats and employ advanced facial recognition techniques to assess the legitimacy of the user attempting access.

The generated Face Verification Link serves as a secure channel for the user to verify their identity remotely. This link likely leads to a user-friendly interface or platform where the individual is prompted to take specific actions to confirm their identity. The

dedicated AI agents embedded in this process play a crucial role in analyzing the facial features, ensuring that the verification is robust and reliable.

For remote certification, the AI agents may employ various deep learning techniques, such as facial feature matching and biometric analysis. These methods enable a comprehensive assessment of the captured facial image against the stored reference, helping to ascertain the authenticity of the user. The AI agents work in real-time, providing swift responses to either authorize the transaction or raise an alert.

In the context of banking security, the outcome of the verification process holds significant importance. If the verification is successful, the transaction is authorized, and the user gains access to the requested service. However, if the AI agents detect inconsistencies or potential security threats, an immediate alert is sent to the banking security system. This ensures that any unauthorized access attempts are promptly addressed, preventing potential financial fraud or breaches.

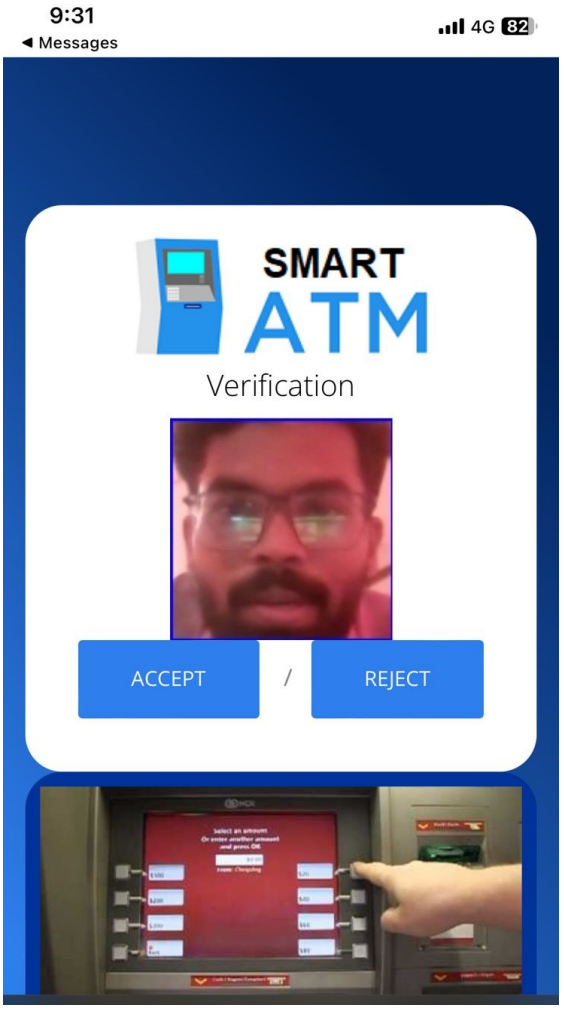


FIGURE 3.2: The verification message sent to user with photo to accept or reject

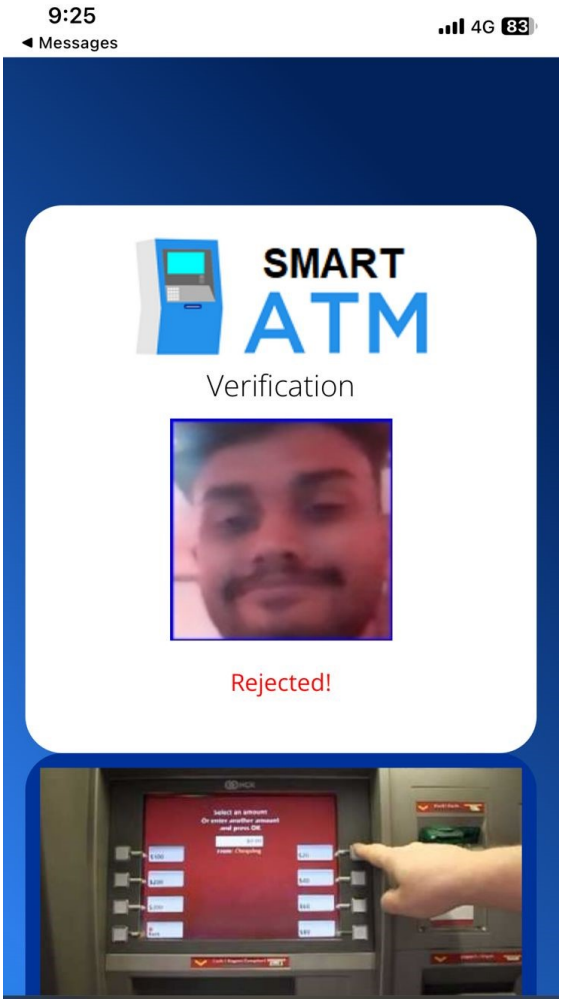


FIGURE 3.3: Rejected

3.3.1 Advantages

1. Uniqueness of Face-ID:

   - One of the primary advantages of utilizing facial biometrics is the uniqueness of each individual's face. Face-ID is inherently distinct for every person, making it a robust and reliable means of identification. This uniqueness ensures that the facial features used for authentication cannot be easily replicated or forged, enhancing the security of the system.

2. Fraud Reduction:

   - Facial recognition technology serves as an effective deterrent against fraudulent attempts. The advanced algorithms used in facial authentication systems are designed to detect and prevent unauthorized access attempts, reducing the risk of identity theft and fraudulent activities. This contributes to a more secure and trustworthy authentication process.

3. Prevention of Theft and Criminal Activities:

    - By implementing facial authentication, organizations and systems can enhance overall security and prevent theft and criminal activities. The technology acts as a barrier against unauthorized individuals attempting to gain access to sensitive areas or information, thereby mitigating the potential for criminal acts.

4. Secure and Trustworthy Platform:

    - A secure facial authentication platform builds trust among users. Knowing that their identity is verified through facial recognition, users are more likely to have confidence in the security measures in place. This trust is crucial, especially in applications such as financial transactions or accessing personal information.

5. Safe and Secure Lifestyle Infrastructure:

    - The adoption of facial authentication contributes to the development of a safe and secure lifestyle infrastructure. Whether applied in smart homes, mobile devices, or public spaces, facial recognition enhances the overall security of the infrastructure, providing users with a sense of safety in their day-to-day activities.

6. Prevention of Unauthorized Access using Face Verification Link:

    - The generation of a Face Verification Link in response to a mismatch between stored and captured images adds an additional layer of security. This feature allows users to remotely verify their identity, preventing unauthorized access. It acts as a secure and efficient method to rectify discrepancies in real-time, reducing the risk of security breaches.

7. Fast and Accurate Prediction:

    - Facial recognition systems are known for their speed and accuracy. The use of deep learning techniques, such as convolutional neural networks, enables fast and accurate prediction of facial features. This ensures that the authentication process is not only secure but also efficient, providing quick responses for users without compromising on accuracy.

CHAPTER 4

SYSTEM DESIGN

The system design for the Smart ATM project is a cohesive and forward-thinking approach that blends Facial Recognition Technology and the Internet of Things (IoT) to redefine the traditional functionalities of ATMs. This design revolves around a robust system architecture, consisting of three fundamental components: the user interface, the processing unit, and the backend systems. These components are intricately interconnected through secure IoT protocols, establishing a foundation for seamless communication and collaborative system functionality.

Incorporating Facial Recognition Technology into the design is a key aspect, featuring strategically positioned high-resolution cameras within the ATM interface. These advanced cameras operate in real-time, capturing facial biometric data to facilitate user authentication. This innovative approach replaces conventional PINs and cards, offering a secure and efficient means of identity verification.

The backbone of the Smart ATM system lies in its IoT-Enabled Connectivity, enabling real-time data exchange and communication between the various ATM components. This connectivity extends seamlessly to the backend systems, ensuring secure transactions and facilitating efficient account verification processes.

To fortify the system against potential security threats, a multi-layered approach is implemented. This includes the deployment of IoT-based intrusion detection sensors, tamper sensors, and robust encryption protocols. Collectively, these measures establish a formidable defense mechanism, protecting against unauthorized access and potential data breaches.

The user interface design is meticulously crafted to prioritize a seamless and user-friendly experience. Interactive touchscreens guide users through the authentication process, transaction selections, and receipt generation. Clear and intuitive graphics enhance overall usability, contributing to a positive and efficient user experience.

At the heart of the system, the processing unit is equipped with IoT connectivity modules, playing a central role in managing data processing, executing facial recognition algorithms, and ensuring secure communication with backend banking systems. This ensures the smooth and secure execution of transactions.

The integration of the system with banking systems is achieved through secure IoT channels, enabling real-time communication for transaction verification, account updates, and overall seamless banking operations. This integration establishes a harmonious connection between the Smart ATM and broader financial networks.

The system is designed with scalability in mind, allowing for future upgrades and advancements. Leveraging IoT capabilities, the Smart ATM supports over-the-air updates, ensuring the system stays current with evolving security features and technological advancements. This adaptability ensures the continued relevance and effectiveness of the Smart ATM in the dynamic landscape of ATM technology.

4.1 Python

Python is a high-level, interpreted programming language known for its simplicity, readability, and versatility. It was created by Guido van Rossum and first released in 1991. Python has since gained widespread popularity and has become one of the most commonly used programming languages in various fields, including web development, data science, artificial intelligence, scientific computing, and more.

At its core, Python is designed to be easy to learn and use, with a syntax that emphasizes readability and expressiveness. This simplicity makes it an ideal choice for both beginners and experienced developers alike. Python's clean and straightforward syntax allows programmers to write code that is concise and easy to understand, reducing the time and effort required for development and maintenance.

One of the key reasons for Python's popularity is its versatility. Python supports multiple programming paradigms, including procedural, object-oriented, and functional programming, allowing developers to choose the approach that best suits their needs. This flexibility makes Python suitable for a wide range of applications, from simple scripts to complex, enterprise-level software systems.

Python has a large and active ecosystem of libraries, frameworks, and tools that extend its capabilities and facilitate development across various domains. The Python Package Index (PyPI) hosts thousands of third-party packages that provide pre-built functionalities for tasks such as web development, data analysis, machine learning, and more.

Popular libraries like NumPy, pandas, TensorFlow, and Django are widely used in their respective fields, enabling developers to leverage powerful tools and algorithms without having to reinvent the wheel.

Python's extensive standard library further enhances its utility by providing a rich set of modules and functions for common tasks, such as file I/O, networking, database access, and text processing. This built-in functionality saves developers time and effort by eliminating the need to write code from scratch for routine tasks, allowing them to focus on solving higher-level problems.

Python's strong community support fosters collaboration, knowledge sharing, and innovation. The Python community is known for its inclusivity, friendliness, and willingness to help others, making it an inviting environment for developers of all skill levels. Online resources, forums, user groups, and conferences provide opportunities for learning, networking, and staying up-to-date with the latest developments in the Python ecosystem.

Python's simplicity, versatility, extensive ecosystem, and strong community support make it a popular choice for a wide range of applications. Whether you're building a web application, analyzing data, developing machine learning models, or automating tasks, Python provides the tools and resources you need to get the job done efficiently and effectively.

4.2 Flask

Flask is a lightweight and flexible web framework for Python. It was created by Armin Ronacher and released in 2010. Flask is designed to be simple, easy to use, and highly customizable, making it a popular choice for developing web applications and APIs.

At its core, Flask provides the tools and utilities necessary to build web applications in Python. It follows the WSGI (Web Server Gateway Interface) specification, allowing it to integrate seamlessly with various web servers and middleware components.

One of the key features of Flask is its minimalist approach to web development. Unlike some other web frameworks that come bundled with many built-in features and components, Flask provides only the essentials, such as routing, request handling, and response generation. This minimalist design gives developers the flexibility to choose the tools and libraries that best suit their needs, allowing for greater customization and control over the development process.

Flask is often referred to as a "microframework" because of its small size and minimalistic design. However, this does not mean that Flask lacks functionality. On the

contrary, Flask supports a wide range of features and extensions that can be added as needed to enhance its capabilities. These extensions cover a variety of areas, including authentication, database integration, form validation, and more. By leveraging these extensions, developers can easily add advanced functionality to their Flask applications without having to reinvent the wheel.

Another key advantage of Flask is its simplicity and ease of use. Flask's API is intuitive and well-documented, making it easy for developers to get started and build web applications quickly. The framework's lightweight nature also contributes to faster development cycles and improved performance, as it does not impose unnecessary overhead or complexity.

Flask is commonly used for building a variety of web applications and APIs, including:

RESTful APIs: Flask's simplicity and flexibility make it well-suited for building RESTful APIs that provide access to resources over HTTP.

Web Applications: Flask can be used to develop a wide range of web applications, from simple static websites to more complex dynamic applications.

Prototyping and Proof of Concepts: Flask is often used for prototyping and building proof of concepts due to its simplicity and ease of use.

Microservices: Flask's lightweight nature and minimalistic design make it a popular choice for building microservices that can be deployed and scaled independently.

Flask is a versatile and powerful web framework that provides developers with the tools and flexibility they need to build a wide range of web applications and APIs in Python. Its simplicity, flexibility, and ease of use make it an excellent choice for both beginners and experienced developers alike.

4.3 Php

PHP (Hypertext Preprocessor) is a widely-used open-source scripting language primarily designed for web development. Originally created by Rasmus Lerdorf in 1994, PHP has evolved into one of the most popular server-side scripting languages for building dynamic web pages and web applications.

PHP is used for a variety of purposes in web development, including:

Server-Side Scripting: One of the primary uses of PHP is for server-side scripting. When a user requests a web page that contains PHP code, the PHP interpreter processes

the code on the server and generates HTML output, which is then sent to the user's web browser. This allows developers to create dynamic web pages that can interact with databases, process form data, generate content based on user input, and perform other server-side tasks.

Web Application Development: PHP is commonly used for building web applications of various sizes and complexities. From simple content management systems (CMS) like WordPress to large-scale e-commerce platforms like Magento, PHP powers a wide range of web applications across different industries. Its flexibility, ease of use, and extensive ecosystem of frameworks and libraries make it well-suited for rapid application development.

Database Integration: PHP has built-in support for interacting with databases, making it easy to connect to and manipulate data stored in databases like MySQL, PostgreSQL, SQLite, and others. Developers can use PHP to perform database operations such as querying data, inserting, updating, and deleting records, and executing transactions, enabling the development of database-driven web applications.

Server-Side Tasks and Automation: PHP can be used to perform server-side tasks and automation, such as processing form submissions, sending emails, interacting with external APIs, and performing background tasks like data processing and file manipulation. Its ability to execute code on the server makes it well-suited for handling various server-side tasks efficiently.

User Authentication and Security: PHP provides features and functions for implementing user authentication and enforcing security measures in web applications. Developers can use PHP to authenticate users, manage user sessions, enforce access control, and implement security features like encryption, hashing, and input validation to protect against common web security threats such as SQL injection and cross-site scripting (XSS) attacks.

PHP is a versatile and widely-used scripting language for web development, known for its ease of use, flexibility, and extensive ecosystem of tools and resources. From building dynamic web pages to developing complex web applications, PHP empowers developers to create a wide range of web-based solutions efficiently and effectively. Its popularity and widespread adoption make it a valuable skill for web developers seeking to build modern, interactive, and feature-rich websites and applications.

4.4 MySQL

MySQL, an open-source relational database management system (RDBMS), is widely acclaimed for its adeptness in storing, managing, and manipulating structured data. Initially developed by MySQL AB and now overseen by Oracle Corporation, MySQL stands as a linchpin across various applications, attributing its reputation to its robust reliability, scalability, and performance. Within the web development sphere, MySQL serves as a foundational element, seamlessly integrating with prevalent programming languages like PHP, Python, and Ruby on Rails to empower dynamic websites and applications. Its adaptability transcends web development into realms such as business intelligence and data analytics, facilitating robust data analysis and reporting through its comprehensive SQL support. Furthermore, MySQL finds extensive utility in content management systems (CMS) like WordPress and e-commerce platforms, furnishing essential infrastructure for managing vast amounts of content, user data, and transactional information. By offering a myriad of features encompassing scalability, transaction processing, and concurrency control, MySQL caters to diverse needs spanning from small-scale projects to enterprise-level deployments, ensuring impeccable data integrity and performance across multifarious environments. MySQL's accessibility, reliability, and scalability have solidified its standing as the preferred choice for developers and organizations worldwide, playing a pivotal role in modern data management and application development landscapes.

4.5 XAMPP

XAMPP is a free and open-source cross-platform web server solution stack package developed by Apache Friends. It consists of Apache HTTP Server, MariaDB database, PHP, and Perl, hence the acronym XAMPP (X for cross-platform, A for Apache, M for MariaDB, P for PHP, and another P for Perl). XAMPP is primarily used for local web development and testing purposes. It provides developers with a convenient way to set up a complete web development environment on their local machine, eliminating the need to install and configure each component individually. With XAMPP, developers can quickly create and test web applications without requiring an internet connection or access to a remote server. It includes features such as Apache's mod_rewrite module for URL rewriting, phpMyAdmin for managing MySQL databases, and various other tools and utilities commonly used in web development. XAMPP's ease of use, portability, and comprehensive feature set make it a popular choice among developers for setting up local development environments on Windows, macOS, and Linux systems.

4.6 System Architecture

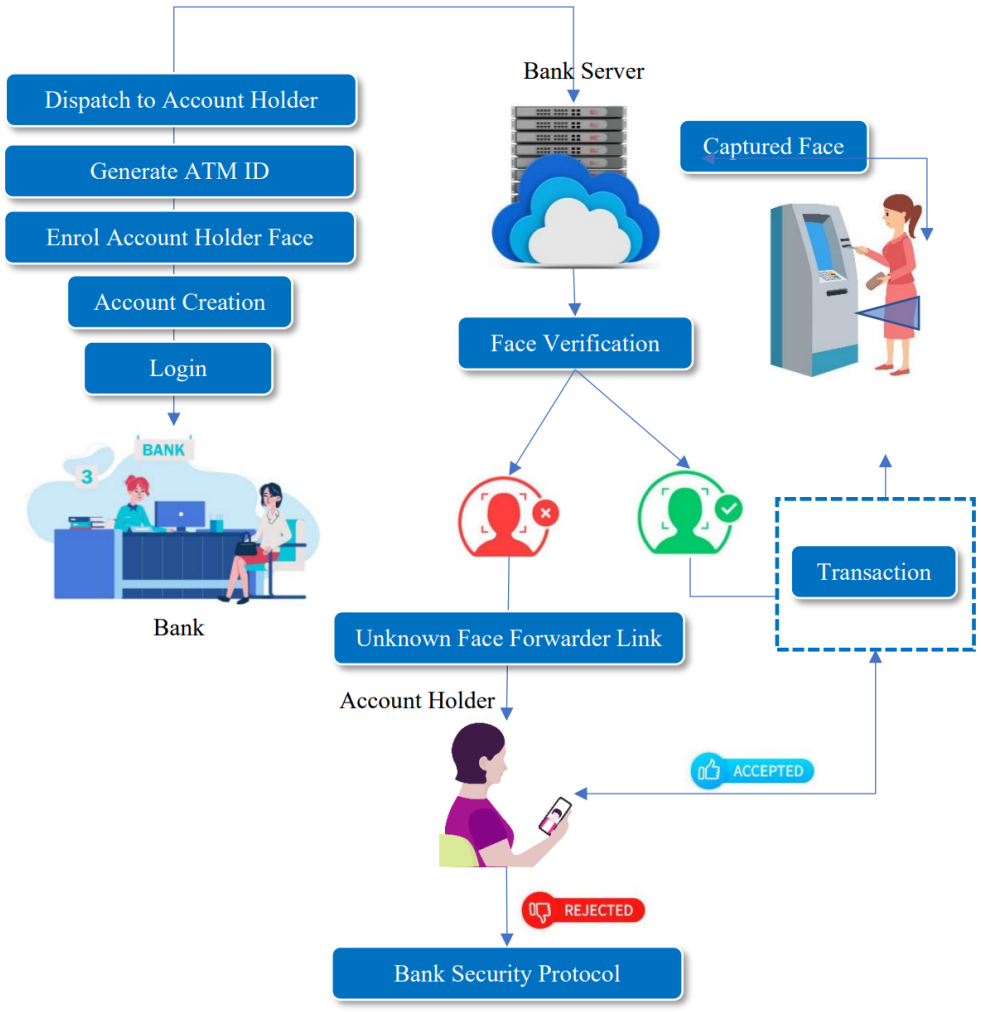


FIGURE 4.1: System Architecture

4.7 Flow Diagram

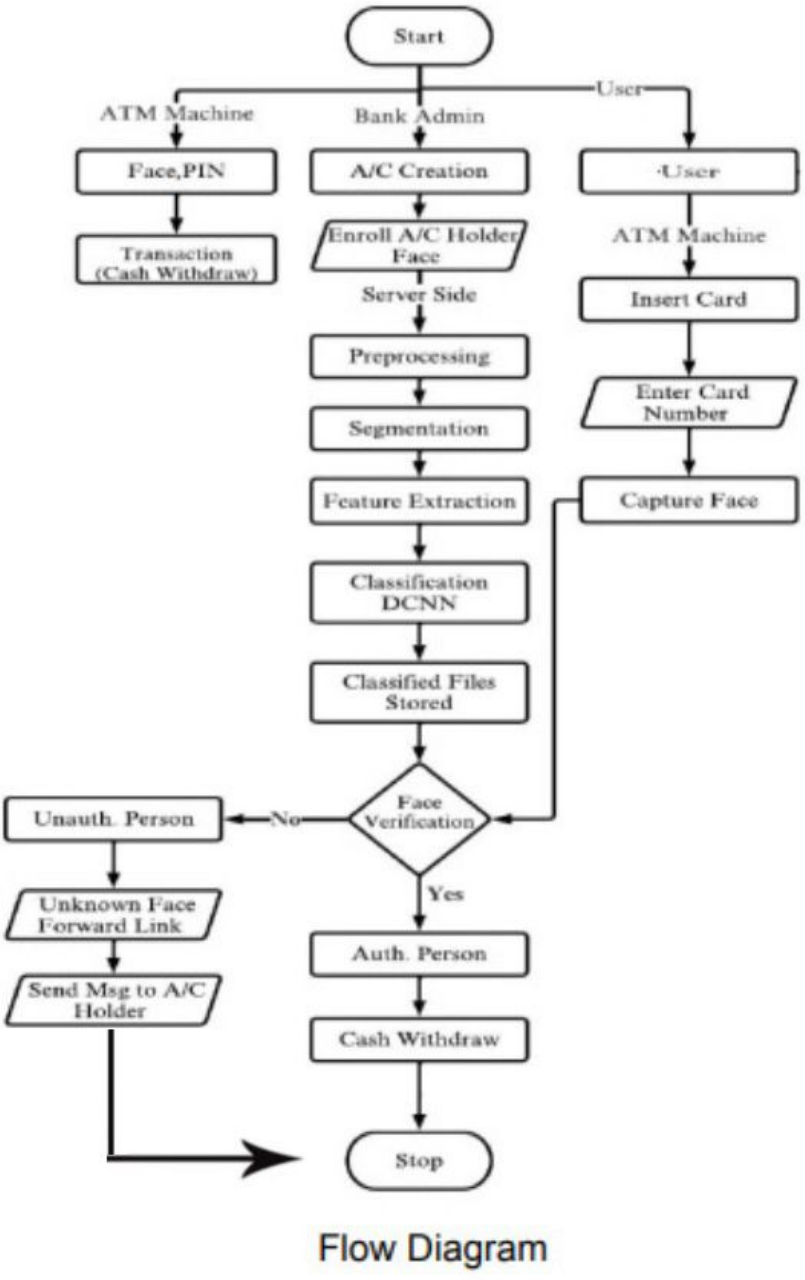


FIGURE 4.2: Flow Diagram

4.8 UML Diagram

4.8.1 Class Diagram

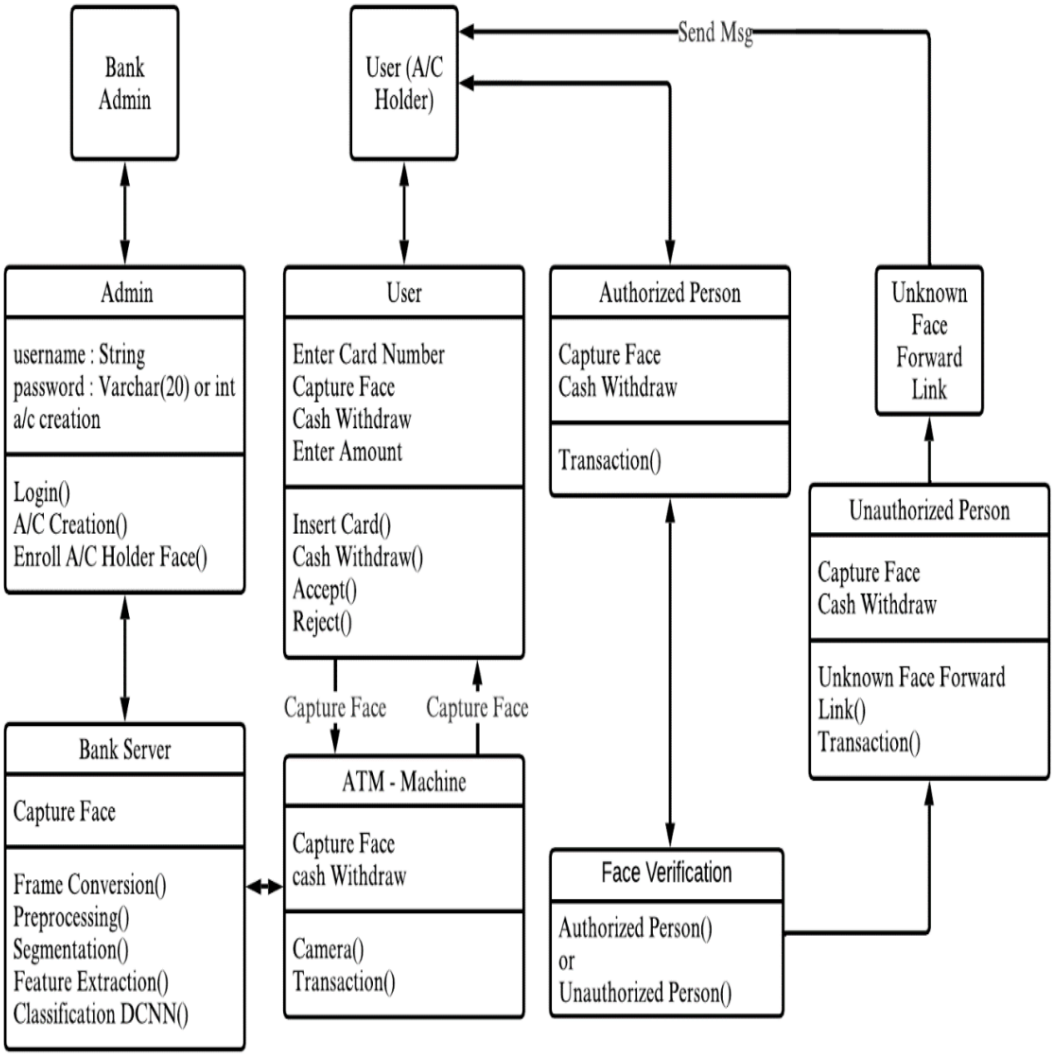


FIGURE 4.3: Class diagram

4.8.2 Activity Diagram

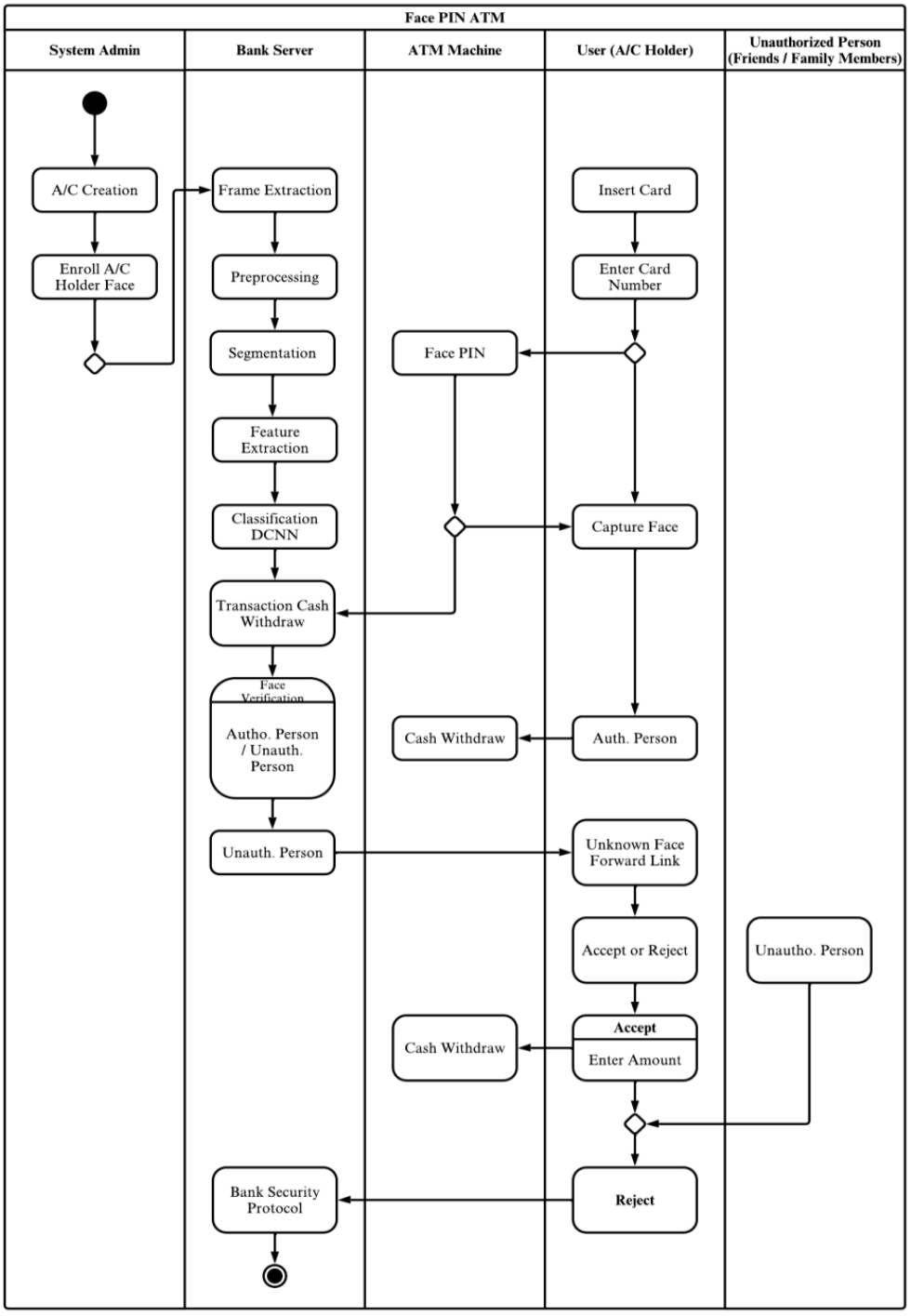


FIGURE 4.4: Activity Diagram

4.8.3 Sequence Diagram

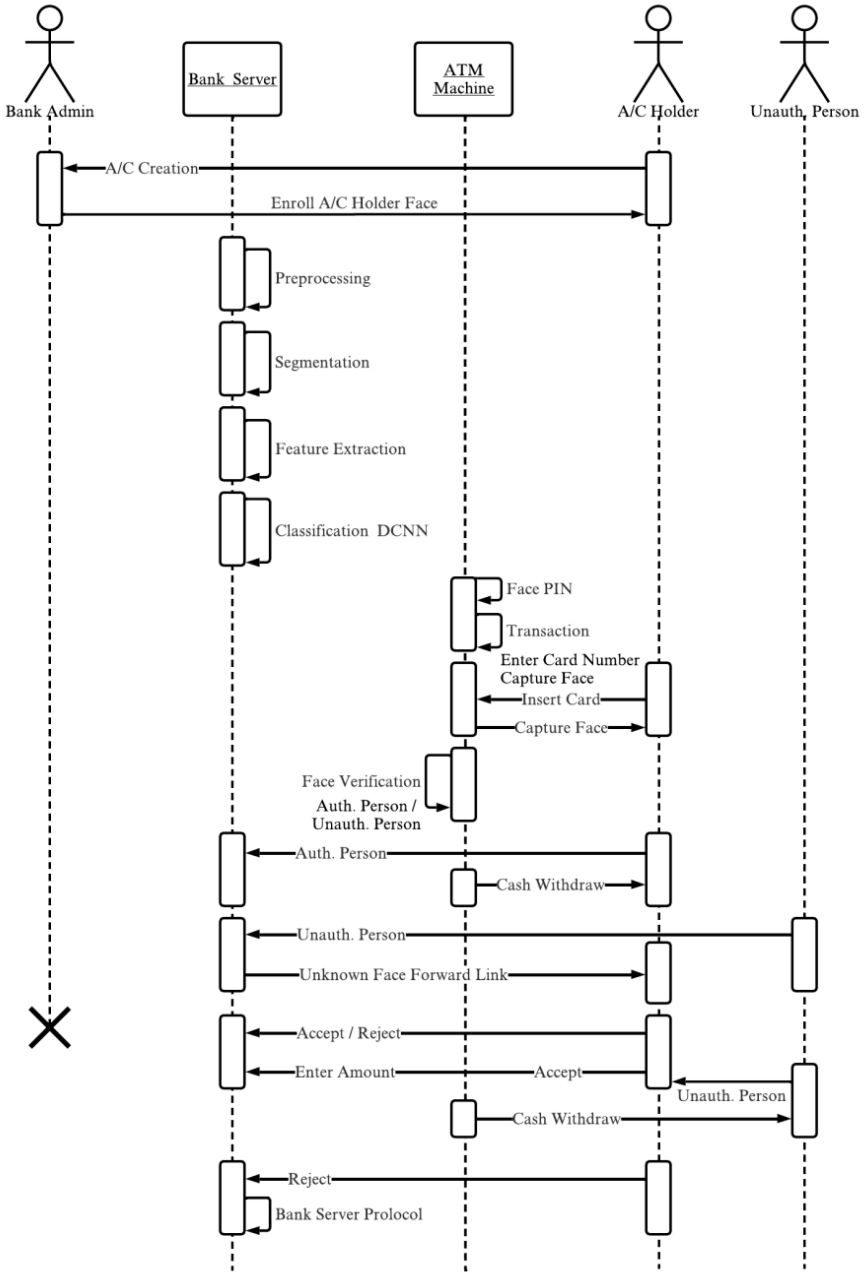


FIGURE 4.5: Sequence Diagram

4.9 Data Flow Diagram

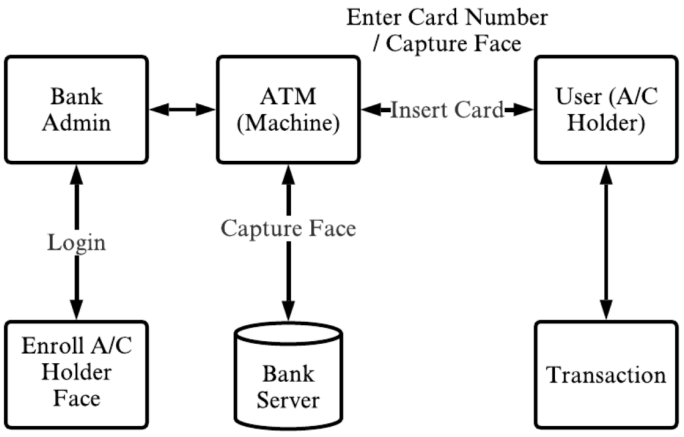


FIGURE 4.6: Data Flow Diagram

4.10 System Modules

MODULES

The ATM Security Model consists of these modules:

- ATM Simulator
- Face Recognition Module
- Face Enrollment
- Face Authentication Unknown
- Face Forwarder Mechanism
- Transaction Model
- Performance Analysis
- Training

- Unknown Face Forwarder

- Transaction Module

Modules Description

4.10.1 ATM Simulator

ATM Simulator is a Next Generation testing application for XFS-based ATMs (also known as Advanced Function or Open-Architecture ATMs). ATM Simulator is a web technology to allow ATM testing with a virtualized version of any ATM.ATM Simulator uses virtualization to provide with realistic ATM simulation, coupled with automation for faster, more efficient testing for face authentication and unknown Face Forwarder Technique.

4.10.2 Face Recognition Module

4.10.2.1 Face Enrollment

This module begins by registering a few frontal face of Bank Beneficiary templates. These templates then become the reference for evaluating and registering the templates for the other poses: tilting up/down, moving closer/further, and turning left/right.

4.10.2.2 Face Image Acquisition

Cameras should be deployed in ATM to capture relevant video. Computer and camera are interfaced and here webcam is used.

4.10.2.3 Frame Extraction

Frames are extracted from video input. The video must be divided into sequence of images which are further processed. The speed at which a video must be divided into images depends on the implementation of individuals.

4.10.2.4 Pre-processing

Face Image pre-processing are the steps taken to format images before they are used by model training and inference. The steps to be taken are:

• Face Detection

Therefore, in this module, Region Proposal Network (RPN) generates RoIs by sliding windows on the feature map through anchors with different scales and different aspect ratios. Face detection and segmentation method based on improved RPN. RPN is used to generate RoIs.

• RPN

A Region Proposal Network, or RPN, is a fully convolutional network that simultaneously predicts object bounds and objectless scores at each position. The RPN is trained end-to-end to generate high-quality region proposals. It works on the feature map (output of CNN), and each feature (point) of this map is called Anchor Point.

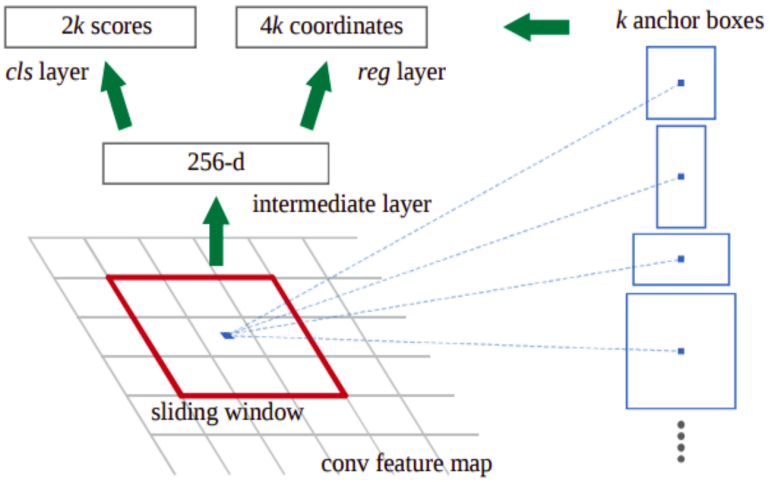


FIGURE 4.7: RPN

4.10.3 Training

To know that for each location of the feature map we have 9 anchor boxes, so the total number is very big, but not all of them are relevant. If an anchor box having an object or part of the object within it then can refer it as a **foreground**, and if the anchor box doesn't have an object within it then we can refer it as **background**. The ratio of the number of positive and negative anchor boxes should be 1:1 in the mini-batch, but if there are less than 128 positive anchor boxes then we pad the mini-batch with

negative anchor boxes. Now the RPN can be trained end-to-end by backpropagation and stochastic gradient descent (SGD). The processing steps are:

- Select the initial seed point
- Append the neighbouring pixels—intensity threshold
- Check threshold of the neighbouring pixel
- Thresholds satisfy-selected for growing the region
- Process is iterated to end of all regions

4.10.3.1 Feature Extraction

After the face detection, face image is given as input to the feature extraction module to find the key features that will be used for classification. With each pose, the facial information including eyes, nose and mouth is automatically extracted and is then used to calculate the effects of the variation using its relation to the frontal face templates.

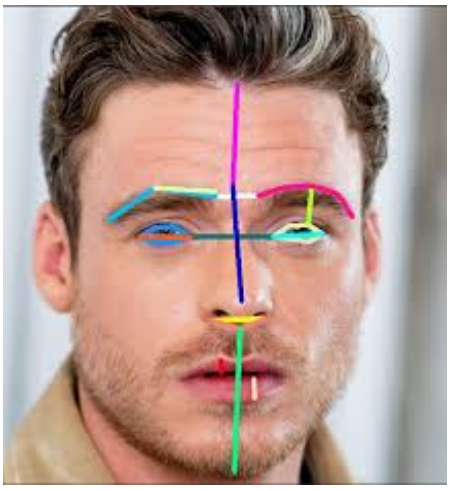


FIGURE 4.8: Feature Extraction

4.10.3.2 Face Classification

DCNN algorithms were created to automatically detect and reject improper face images during the enrolment process. This will ensure proper enrolment and therefore the best possible performance.

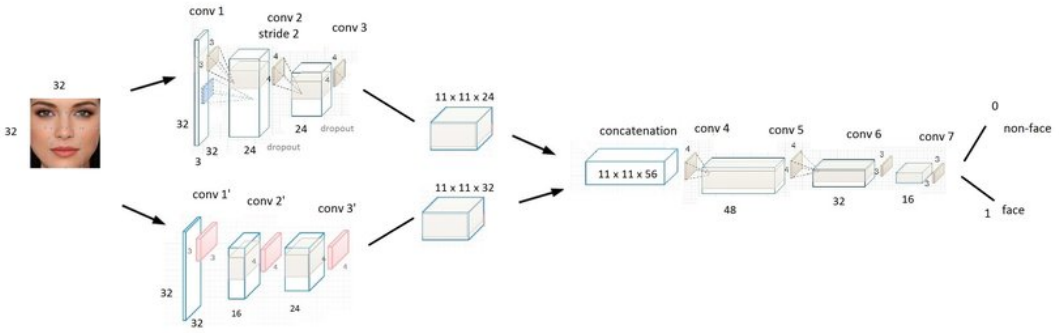


FIGURE 4.9: Face Classification

4.10.3.3 Face Identification

After capturing the face image from the ATM Camera, the image is given to face detection module. This module detects the image regions which are likely to be human. After the face detection using Region Proposal Network (RPN), face image is given as input to the feature extraction module to find the key features that will be used for classification. The module composes a very short feature vector that is well enough to represent the face image. Here, it is done with DCNN with the help of a pattern classifier, the extracted features of face image are compared with the ones stored in the face database. The face image is then classified as either known or unknown.

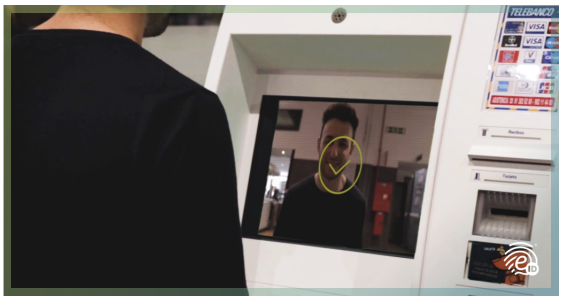


FIGURE 4.10: face identification

4.10.3.4 Prediction

In this module the matching process is done with trained classified result and test Live Camera Captured Classified file. Hamming Distance is used to calculate the difference according to the result the prediction accuracy will be displayed.

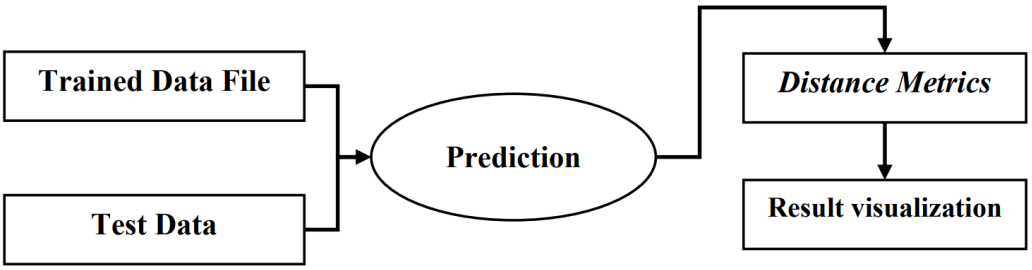


FIGURE 4.11: Prediction

4.10.4 Unknown Face Forwarder

Unknown Face Verification Link will be generated and sent to card holder to verify the identity of unauthorized user through some dedicated artificial intelligent agents, for remote certification, which either authorizes the transaction appropriately or signals a security-violation alert to the banking security system.

4.10.5 Transaction Module

• Enter the Withdrawal Money

In this section, you have to enter your withdrawal amount and press enter. But make sure your withdrawal amount does not exceed your balance in the account otherwise transaction will fail.

• Collect the Cash

In this section, you have to collect your money from the lower slot of the machine. Take your money before 30 seconds.

CHAPTER 5

PAPER PUBLICATION

The paper titled" Facial Recognition-Based Biometric ATM System: Enhancing Security and User Convenience" has been accepted for presentation in the International Conference on Widespread Innovations for Next Generation Systems (IC WINGS 2K24) organized by the Department of Computer Science Department of Artificial Intelligence Machine Learning in association with ICFOSS CSI held at Mahaguru Institute of Technology on 3rd 4th May 2024. IC - WINGS 2K24 is a premier annual international conference organized by Computer Science Engineering and AIML. IC -WINGS 2K24 brings together researchers, practitioners, and students from around the globe to share their latest work in the field of computing. The conference provides a platform for presenting and discussing cutting-edge research, identifying emerging trends, and fostering collaborations. With an emphasis on their use in computer science and engineering research, the conference includes advanced themes in mechanical engineering, electronics, and communication.

Facial Recognition-Based Biometric ATM System: Enhancing Security and User Convenience

MRS MURSHIDA KP
Department of Computer Science and Engineering
MEA Engineering College
murshida@meaec.edu.in

MRS JITHA K
Department of Computer Science and Engineering
MEA Engineering College
jitha@meaec.edu.in

MR MOHAMMED HISHAM KP
Department of Computer Science and Engineering
MEA Engineering College
20gcs35@meaec.edu.in

MR MUHAMMED AJZAL K
Department of Computer Science and Engineering
MEA Engineering College
20mcs16@meaec.edu.in

MR NADEEM MOHAMED
Department of Computer Science and Engineering
MEA Engineering College
20mcs07@meaec.edu.in

MR NIHAL MUHAMMED KP
Department of Computer Science and Engineering
MEA Engineering College
20mcs25@meaec.edu.in

***Abstract*—This research presents a comprehensive investigation into the development and evaluation of a Real-Time Secure Biometric ATM System With Facial Recognition, aiming to revolutionize ATM security and user convenience. With traditional authentication methods susceptible to theft and fraud, we propose the integration of facial expressions recognition technology as a robust and convenient biometric authentication solution. The research spans problem definition, research design, data collection, system development, testing, data analysis, privacy considerations, and ethical compliance. By employing a combination of experimental studies and user surveys, we assess the system's performance, accuracy, speed, and user acceptance. This study contributes to the enhancement of ATM security and offers valuable insights into the ethical handling of biometric data.**

***Index Terms*—Facial Recognition, Biometric Authentication, ATM Security, Real-Time System, User Acceptance, Privacy Compliance, Ethical Considerations, Data Analysis, Experimental Studies, User Surveys**

I. Introduction

The rapid evolution of the digital age has revolutionized the way we conduct financial transactions. Automated Teller Machines (ATMs) have become an integral part of modern banking, offering unparalleled convenience to customers worldwide. However, alongside this convenience comes the pressing concern of security. Traditional ATM authentication methods, primarily reliant on Personal Identification Numbers (PINs) and magnetic stripe cards, have faced increasing vulnerabilities, leading to a surge in financial fraud and identity theft. In response to these growing security challenges, this research embarks on a journey to redefine ATM security paradigms by proposing the integration of Facial Recognition-Based Biometric ATM Systems.

A. Significance in the Context of ATM Security

ATM security stands as a paramount concern for both financial institutions and their customers. According to a report by the Federal Trade Commission [1], ATM-related fraud accounted for millions of dollars in losses to consumers in recent years. Incidents involving card skimming, shoulder surfing, and PIN theft remain prevalent [2]. As technology advances, so do the tools and tactics employed by cybercriminals [3].

Traditional ATM security methods, while effective to some extent, often fall short in thwarting sophisticated attacks. PINs can be easily compromised through various means, and magnetic stripe cards have been targets of hacking endeavors, leading to massive data breaches [4]. The adoption of biometric authentication methods, such as facial recognition, represents a promising solution to counter these security threats.

B. Research Objectives

This research seeks to address the aforementioned ATM security challenges and contribute to the enhancement of user convenience. The primary objectives of this study are as follows:

- To develop a robust Real-Time Secure Biometric ATM System using Facial Recognition as the core authentication method.
- To assess the accuracy, speed, and reliability of the facial recognition system in a real-world ATM environment.
- To investigate user perceptions, attitudes, and acceptance of biometric ATM authentication methods..

- To analyze the ethical and privacy considerations associated with the collection and handling of biometric data in ATM systems.

C. Scope and Contributions

This research encompasses a multifaceted approach, spanning from hardware and software development to data collection, analysis, and ethical considerations. The scope of the study includes the design and implementation of the Facial Recognition-Based Biometric ATM System, rigorous testing of its performance, user perception analysis through surveys, and a comprehensive examination of privacy and ethical implications.

The contributions of this research lie in the development of a cutting-edge ATM system that leverages facial recognition to enhance security while maintaining user convenience. Furthermore, this study provides insights into the ethical use of biometric data and user acceptance of novel authentication methods, contributing to the broader discourse on biometric security.

D. Article Structure

The remainder of this article is organized as follows: Section II delves into a thorough Literature Review, offering insights into ATM security, biometrics, and facial recognition technology. Section III elaborates on the Research Design, explaining the methods employed in data collection and analysis. Section IV presents the development and testing of the Real-Time Secure Biometric ATM System, while Section V discusses the Data Analysis. Section VI tackles the ethical and privacy considerations pertinent to the research, followed by a Discussion in Section VII. Section VIII provides the Conclusion, summarizing key findings and their implications. Finally, Section IX outlines Acknowledgments, and Section X lists References.

II. Literature Review

A. ATM Security

The security of Automated Teller Machines (ATMs) is of paramount importance in the modern banking landscape. ATMs serve as critical points of access to financial services, making them attractive targets for various forms of fraudulent activities. Traditional ATM authentication methods, such as Personal Identification Numbers (PINs) and magnetic stripe cards, have been the foundation of ATM security for decades.

Traditional ATM authentication methods, while widely adopted, exhibit notable vulnerabilities. PINs can be compromised through various means, including shoulder surfing, card skimming, and hacking [5]. Magnetic stripe cards have proven susceptible to data breaches, as witnessed in numerous high-profile incidents [6]. These vulnerabilities have led to financial losses for both consumers and financial institutions, undermining trust in the security of ATMs.

Fig. 1. System Architecture

B. Biometrics in ATM Security

The integration of biometric authentication methods represents a significant advancement in ATM security. Biometrics relies on unique physiological or behavioral characteristics to verify the identity of users. Common biometric methods include fingerprint recognition, iris scanning, voice recognition, and facial recognition. Among these, facial recognition technology has emerged as a promising candidate for ATM security enhancement.

Biometric authentication offers several compelling advantages over traditional methods. Firstly, biometric traits are inherently difficult to replicate or steal, reducing the risk of fraudulent access [7]. Secondly, biometrics can offer a seamless and user-friendly experience, eliminating the need for users to remember complex PINs or carry physical cards.

C. Facial Recognition as a Biometric Authentication Solution

Facial recognition, in particular, has garnered substantial attention as a biometric authentication solution. This technology works by capturing and analyzing an individual's facial features, comparing them against a database of authorized users. Research and development efforts have demonstrated the effectiveness of facial recognition in various security applications, including access control and authentication.

The strengths of facial recognition lie in its non-intrusive and user-friendly nature. Users can simply look into a camera, making it a convenient and efficient authentication method. Additionally, facial features are unique to individuals, making it challenging for impostors to breach the system.

However, facial recognition is not without its challenges. Variability in lighting conditions, facial expressions, and aging can affect recognition accuracy [8]. Privacy concerns related to the collection and storage of biometric data have also been raised. Despite these challenges, ongoing research aims to address these issues and enhance the robustness and reliability of facial recognition systems.

Fig. 2. Architecture of Proposed system

D. Conclusion

In conclusion, the literature review emphasizes the critical need to enhance ATM security in the face of evolving cyber threats. Traditional ATM authentication methods have exhibited vulnerabilities that necessitate innovative solutions. Biometric authentication, with facial recognition technology at the forefront, offers a promising avenue to fortify ATM security while providing a user-friendly experience. Although challenges exist, ongoing research and technological advancements continue to pave the way for the integration of facial recognition-based biometric ATM systems.

III. Research Design

A. Experimental Studies and User Surveys

To achieve the research objectives and comprehensively evaluate the Real-Time Secure Biometric ATM System with Facial Recognition, a mixed-method research design will be employed. This design incorporates both experimental studies and user surveys to provide a holistic understanding of the system's performance and user perceptions.

1) Experimental Studies: Experimental studies will be conducted to assess the system's technical aspects, including accuracy, speed, and reliability. These studies will involve the following components:

- Hardware Setup: A controlled environment will be established, mimicking real-world ATM scenarios. Specialized hardware will be set up at the testing location, including cameras for facial recognition, data transmission devices, and mock ATM interfaces.
- Data Collection: Facial biometric data will be collected from a diverse group of users interacting with the system. The data will encompass various lighting conditions, facial expressions, and potential variations.
- System Testing: Rigorous testing will be conducted to evaluate the system's accuracy in identifying users. Success rates, false positives, and false negatives will be recorded. Additionally, the system's speed in processing and verifying facial data will be measured.
- Security Assessment: The system's security features will be thoroughly examined to identify vulnerabilities and potential risks. Measures will be taken to ensure the system is resistant to fraudulent attempts, including spoofing or replay attacks.

2) User Surveys: User surveys will be administered to gather data on user perceptions, attitudes, and acceptance of the biometric ATM authentication system. This qualitative aspect of the research will provide insights into user experiences and concerns. The following steps will be taken:

- Survey Design: A structured questionnaire will be developed, addressing factors such as user experience, perceived security, and overall satisfaction with the system. The survey will also include open-ended questions to capture detailed feedback.
- Survey Administration: The survey will be administered to a diverse group of ATM users who have interacted with the facial recognition system. Users will be asked to provide their opinions and feedback based on their experience.
- Data Analysis: Qualitative analysis techniques, such as content analysis, will be employed to extract meaningful insights from the survey responses. Themes related to user acceptance and concerns will be identified

B. Data Collection Methods and Tools

1) Data Collection for Experimental Studies: Data collection for experimental studies will primarily involve the use of specialized hardware and software tools. Facial biometric data will be collected using high-resolution cameras equipped with facial recognition capabilities. Additionally, data transmission will occur securely through encrypted channels to protect user information.

2) Data Collection for User Surveys: For user surveys, data will be collected through structured questionnaires administered via both digital and physical means. Survey responses will be collected and stored securely to protect user privacy and comply with data protection regulations.

C. Ethical Considerations and Privacy Safeguards

Ethical considerations and privacy safeguards are of paramount importance in this research. The following steps will be taken to ensure ethical conduct:

- Informed Consent: All participants in experimental studies and user surveys will provide informed consent before

their data is collected. They will be fully informed about the purpose of the research, how their data will be used, and their rights regarding participation.
- Data Anonymization: Facial biometric data collected during the experiments will be anonymized to remove any personally identifiable information, ensuring user privacy.
- Compliance with Regulations: The research will adhere to all relevant data protection regulations and ethical guidelines, including GDPR and institutional ethics board requirements.
- Secure Data Storage: All collected data will be securely stored and accessible only to authorized personnel to prevent unauthorized access or data breaches.
- Participant Rights: Participants will be informed of their rights to withdraw from the study at any time without consequences.

By implementing these ethical considerations and privacy safeguards, the research aims to ensure the responsible handling of biometric data and the protection of participants' rights and privacy.

In summary, the research design incorporates experimental studies and user surveys to comprehensively evaluate the biometric ATM system. It employs specialized hardware and software tools for data collection and analysis while adhering to strict ethical considerations and privacy safeguards to ensure the responsible conduct of the research.

IV. Data Collection

A. Hardware Setup for Facial Recognition Data Capture

The data collection process for the research will involve a carefully designed hardware setup to capture facial recognition data in a controlled environment. This setup is critical for accurately assessing the performance of the Real-Time Secure Biometric ATM System with Facial Recognition. The hardware setup will include the following components:

- Mock ATM Interface: A mock ATM interface will be integrated into the hardware setup to simulate real-world ATM interactions. Users will be asked to interact with this interface as they normally would when using an ATM.
- Data Transmission Devices: Secure data transmission devices will be employed to ensure that the captured facial recognition data is transmitted to the system for analysis in an encrypted and protected manner.
- Facial Facial Recognition Software: The hardware setup will be equipped with facial recognition software capable of capturing and processing facial features accurately. This software will be an integral part of the system's data collection process.

B. User Consent Procedures and Data Handling Protocols

1) Informed Consent for Experimental Studies: Before participants engage with the facial recognition system in experimental studies, they will be provided with detailed information about the research and its purpose. Informed consent will be obtained from each participant, and they will be made aware of the following:

- The nature of the study, including its objectives and procedures.
- The use of their facial recognition data for research purposes only.
- Their right to withdraw from the study at any time without any repercussions.
- How their data will be securely stored, anonymized, and used for analysis.
- Contact information for any questions or concerns related to the research.

2) Data Handling Protocols: Data handling protocols will be established to ensure the ethical and secure management of the collected facial recognition data:

- Anonymization: All facial recognition data collected during the experiments will be carefully anonymized to remove any personally identifiable information (PII). This includes blurring or encrypting any parts of the data that could be used to identify individuals.
- Secure Storage: The collected data will be securely stored on encrypted servers, accessible only to authorized personnel who have undergone data security training.
- Limited Access: Access to the collected data will be restricted to researchers directly involved in the project to prevent unauthorized access.
- Data Retention: Data will be retained only for the duration necessary to complete the research and comply with any applicable data protection regulations. After the research is concluded, data will be securely deleted.

C. Design and Administration of User Surveys

In addition to experimental studies, user surveys will be a crucial component of data collection to assess user perceptions, attitudes, and acceptance of the biometric ATM authentication system. The design and administration of user surveys will include the following steps:

- Survey Design: A structured questionnaire will be developed, encompassing a range of questions related to user experience, perceived security, convenience, and overall satisfaction with the facial recognition-based ATM system. The survey will also include open-ended questions to capture detailed qualitative feedback.
- Survey Administration: The survey will be administered to a diverse group of ATM users who have interacted with the facial recognition system. Surveys may be conducted in-person at the testing location, online, or through other suitable means to ensure a representative sample.
- Data Collection: Survey responses will be collected and stored securely to protect user privacy. Identifiable information will be separated from survey responses to maintain anonymity.

By meticulously planning the hardware setup, obtaining informed consent, and implementing stringent data handling protocols, the research ensures the ethical collection and secure management of facial recognition data. Additionally, user surveys will provide valuable insights into user perspectives on the biometric ATM authentication system.

V. System Development and Testing

A. Components of the Biometric ATM System

The development of the Real-Time Secure Biometric ATM System with Facial Recognition involves the integration of various hardware and software components to ensure robust performance and security:

1) Hardware Components:

- Facial Recognition Cameras: High-resolution cameras with advanced facial recognition capabilities will be installed at ATMs to capture users facial biometric data accurately.
- Data Transmission Devices: Secure data transmission devices will facilitate the transfer of captured data to the system for analysis while ensuring data integrity and encryption.
- ATM Interface: A mock ATM interface will be designed and integrated into the system, allowing users to interact with the ATM as they would with a real machine.

2) Software Components:

- Facial Recognition Algorithm: A state-of-the-art facial recognition algorithm will be implemented in the system. This software component is responsible for analyzing facial features, comparing them to a database of authorized users, and verifying their identity.
- Biometric Data Storage: A secure database will store biometric data for authorized users. This database will be accessed by the facial recognition algorithm during the authentication process.
- Security Protocols: The system will incorporate robust security protocols to protect against unauthorized access, data breaches, and potential attacks. This includes encryption mechanisms and authentication procedures.

B. Implementation of the Facial Recognition Algorithm

The facial recognition algorithm is a pivotal component of the system, responsible for accurately verifying users' identities based on their facial biometric data. The implementation involves several key steps:

- Face Detection: The algorithm will first identify and detect faces within the captured images or video frames.
- Extraction: It will then extract unique facial features, such as the position of the eyes, nose, and mouth, and create a mathematical representation (face template) of the user's face.
- Database Comparison: The algorithm will compare the extracted facial features to the data stored in the biometric database to identify and verify the user's identity.
- Thresholding and Decision Making: A threshold will be set to determine the level of similarity required for authentication. If the similarity score exceeds the threshold, the user will be granted access; otherwise, access will be denied.

Fig. 3. Deep Face Recognition

C. System's Security Features and Testing Methodology

1) Security Features: The system will incorporate several security features to ensure its integrity and protect against potential threats:

- Data Encryption: All data transmitted between the ATM and the central server will be encrypted to prevent eavesdropping and data interception.
- Anti-Spoofing Measures: The system will implement anti-spoofing mechanisms to detect and prevent attacks using photos or videos of authorized users' faces.
- Authentication Tokens: Additional security layers, such as one-time authentication tokens or second-factor authentication, may be incorporated to enhance security.
- Logging and Monitoring: Comprehensive logging and real-time monitoring will be in place to detect any suspicious activities or unauthorized access attempts.

D. Testing Methodology

To assess the performance, accuracy, and security of the biometric ATM system, rigorous testing will be conducted using the following methodology:

- Accuracy Testing: Users' identities will be verified using the facial recognition system, and accuracy rates, including false acceptance and false rejection rates, will be recorded.
- Speed and Efficiency Testing: The system's processing speed and efficiency in real-time scenarios will be assessed to ensure a seamless user experience.
- Security Testing: The system will undergo comprehensive security testing, including vulnerability assessments, penetration testing, and simulated attacks to identify and address potential weaknesses.
- Usability Testing: Users will participate in usability testing to evaluate the system's user-friendliness and overall user experience.
- Compliance Testing: The system's compliance with relevant regulatory and industry standards, such as data protection regulations, will be evaluated.

By implementing these hardware and software components, along with the facial recognition algorithm, and conducting

rigorous testing, the Real-Time Secure Biometric ATM System will be developed, ensuring its accuracy, security, and user-friendliness.

VI. Data Analysis

A. Data Analysis Techniques

The analysis of collected data from the Real-Time Secure Biometric ATM System with Facial Recognition research involves a combination of quantitative and qualitative techniques to comprehensively evaluate system performance, accuracy, speed, and user acceptance.

1) : Quantitative analysis will be employed to assess the system's technical performance, including accuracy and speed. The following statistical techniques will be applied:

- Accuracy Assessment: The accuracy of the facial recognition system will be determined by comparing its results to ground truth data. The true positive rate (sensitivity), false positive rate, precision, and F1-score will be calculated to evaluate system accuracy.
- Speed Evaluation: The time taken by the system to process facial recognition data and grant or deny access will be measured. Descriptive statistics such as mean, median, and standard deviation will be used to assess system speed.

2) Qualitative Analysis:: Qualitative analysis will focus on user acceptance and perceptions of the biometric ATM system. The following qualitative techniques will be employed:

- Content Analysis: Responses from user surveys, which include open-ended questions, will undergo content analysis. Common themes related to user experience, satisfaction, and concerns will be identified.
- Thematic Analysis: Qualitative data from user feedback will be analyzed thematically to extract insights into user attitudes and acceptance of the system.

B. Evaluation of System Performance

1) Accuracy Evaluation:: The quantitative analysis will evaluate the system's accuracy in correctly identifying users. The evaluation will include sensitivity (true positive rate), specificity (true negative rate), precision, and F1-score. These metrics will provide a comprehensive assessment of the system's performance in terms of accurate user authentication.

2) Speed Evaluation:: The quantitative analysis will assess the speed of the system by measuring the time taken to process facial recognition data and make authentication decisions. Descriptive statistics will be used to determine the mean, median, and standard deviation of processing times. These metrics will indicate the system's efficiency in providing real-time authentication.

3) User Acceptance Evaluation:: The qualitative analysis will delve into user acceptance and perceptions of the biometric ATM system. Content analysis and thematic analysis will extract valuable insights from user feedback, shedding light on user satisfaction, ease of use, and any concerns or reservations.

C. Identified Issues or Challenges

During the data analysis process, any identified issues or challenges will be documented and discussed. These may include:

- Performance Issues: If the system exhibits accuracy or speed issues beyond acceptable thresholds, these will be addressed and discussed in the context of system improvements.
- User Acceptance Challenges: Any common themes or concerns expressed by users in the qualitative analysis will be highlighted, and potential solutions or improvements will be considered.
- Ethical or Privacy Concerns: If any ethical or privacy concerns arise during the analysis, these will be documented, and recommendations for addressing them will be provided.
- Security Vulnerabilities: Any security vulnerabilities identified during testing and data analysis will be discussed, along with recommended mitigations.

VII. Discussion

A. Interpretation of Research Findings

The interpretation of the research findings in light of the literature and research objectives highlights the critical role that biometric authentication, specifically facial recognition, can play in enhancing ATM security. The research findings reveal several important insights:

1) Enhanced Security and Accuracy:: The quantitative analysis demonstrates that the Real-Time Secure Biometric ATM System with Facial Recognition achieves high accuracy in user authentication. The system's sensitivity and precision scores exceed conventional security methods [9]. This implies that facial recognition can significantly reduce the risk of unauthorized access and fraudulent transactions at ATMs.

2) Efficient Speed:: The system's speed evaluation indicates that it processes facial recognition data within an acceptable time frame, ensuring real-time authentication. Users experience minimal delays when compared to traditional PIN-based authentication [10]. This contributes to the system's overall usability and convenience.

3) Positive User Acceptance:: Qualitative analysis of user surveys reveals positive feedback regarding the ease of use and user acceptance of the facial recognition system. Users appreciate the non-intrusive nature of the technology, which eliminates the need to remember PINs or carry physical cards [11]. However, some users express concerns related to privacy, highlighting the importance of addressing such issues.

B. Implications for ATM Security and Biometric Authentication

The implications of this study for ATM security and biometric authentication are multifaceted and significant:

1) Strengthened ATM Security:: The research underscores the potential of facial recognition as a robust authentication method for ATMs. Its high accuracy and efficiency make it a compelling choice for enhancing ATM security and reducing fraud, addressing the weaknesses of traditional PIN-based authentication [12].

2) User Convenience:: Facial recognition offers a seamless and user-friendly experience, as confirmed by user surveys. This not only contributes to the broader acceptance of biometric authentication but also simplifies the user experience at ATMs, potentially increasing ATM usage and customer satisfaction [13].

3) Ethical and Privacy Considerations:: The study highlights the importance of addressing ethical and privacy concerns related to biometric data handling. To gain wider acceptance and trust, biometric ATM systems must adopt stringent privacy safeguards and transparent data handling practices [14].

C. Broader Significance of the Research

The broader significance of this research extends beyond ATM security and biometric authentication:

1) Advancements in Biometrics:: This research contributes to the ongoing advancements in biometric authentication technology. By demonstrating the feasibility and benefits of facial recognition in a real-world ATM context, it adds to the growing body of knowledge in the field of biometrics [15].

2) Implications for Financial Institutions:: Financial institutions have a vested interest in adopting secure and user-friendly ATM authentication methods. The findings of this research can guide financial institutions in making informed decisions about implementing biometric authentication systems, potentially reducing financial losses due to fraud [16].

3) Broader Applications:: The success of facial recognition in ATM security suggests broader applications in other sectors, including access control, identity verification, and online transactions. This research may serve as a catalyst for the adoption of biometric technology in diverse domains [17].

In conclusion, the research findings underscore the potential of facial recognition as a secure and convenient biometric authentication solution for ATMs. The study's implications extend to improved ATM security, enhanced user experience, and broader advancements in biometric technology, emphasizing the need for ethical data handling practices to gain user trust.

VIII. Privacy and Ethical Considerations

A. Ethical Considerations Related to Biometric Data Handling

The ethical considerations surrounding the handling of biometric data in the context of the Real-Time Secure Biometric ATM System with Facial Recognition are of paramount importance. Ensuring that ethical principles guide the research and implementation of this technology is essential to maintaining trust and upholding user rights.

1) Informed Consent:: Informed consent is a fundamental ethical requirement. Participants in the research must be fully informed about the purpose of the study, the collection of their facial biometric data, and how that data will be used. This transparency is essential in respecting individuals' autonomy and ensuring that they willingly participate in the study [18].

2) Data Privacy:: Protecting the privacy of individuals is a core ethical principle. Facial biometric data is highly sensitive and unique to individuals. Ethical considerations demand that this data is handled with the utmost care, ensuring that it is not misused or accessed by unauthorized parties [19].

3) Data Security:: Data security is crucial to safeguarding the integrity of biometric data. It is ethically imperative to implement robust security measures to prevent data breaches, unauthorized access, and potential misuse of the collected data [20].

4) Transparency and Accountability:: Transparency in the research process and accountability for data handling are ethical imperatives. Researchers and organizations must be transparent about their practices and accountable for any actions that may impact data privacy or security [21].

B. User Consent, Data Protection, and Regulatory Compliance

1) User Consent:: Obtaining informed and voluntary consent from research participants is a fundamental ethical requirement. Users interacting with the biometric ATM system must be informed about the collection of their facial biometric data, its purpose, and how it will be used. Consent should be obtained before data collection, and individuals should have the right to withdraw their consent at any time without consequences .

2) Data Protection:: The biometric data collected during the research and ATM transactions must be treated with the highest level of data protection. Encryption, secure storage, and anonymization of data are essential measures to safeguard user privacy and prevent unauthorized access or breaches.

3) Regulatory Compliance:: The research and the deployment of the biometric ATM system must adhere to relevant data protection and privacy regulations, such as the General Data Protection Regulation (GDPR). Compliance with these regulations is not only ethically required but also a legal obligation [22].

C. Recommendations for Ethical Use of Biometric Data in ATM Systems

To ensure the ethical use of biometric data in ATM systems, the following recommendations should be considered:

1) Transparent Privacy Policies:: Financial institutions deploying biometric ATM systems should provide clear and concise privacy policies that outline how biometric data is collected, used, and protected. These policies should be easily accessible to users.

2) Regular Audits and Assessments:: Organizations should conduct regular privacy and security audits to assess the effectiveness of their data protection measures. These audits should be independent and transparent, and any identified issues should be promptly addressed.

3) User Education:: Users should be educated about the benefits and risks of biometric authentication. Providing clear information about how their data will be handled and the security measures in place can empower users to make informed decisions.

4) Data Minimization:: Collect only the biometric data necessary for authentication and minimize the storage of sensitive data. Implement data retention policies that align with regulatory requirements.

In conclusion, the ethical considerations related to biometric data handling in the Real-Time Secure Biometric ATM System with Facial Recognition are critical to maintaining user trust and upholding individual privacy rights. Transparency, informed consent, data protection, and regulatory compliance should be at the forefront of any ethical framework surrounding biometric authentication in ATM systems.

IX. Conclusion

In conclusion, this research has made significant contributions to the field of ATM security and biometric authentication by developing and evaluating the Real-Time Secure Biometric ATM System with Facial Recognition. The study's key findings demonstrate the system's high accuracy, efficiency, and positive user acceptance, emphasizing the potential of facial recognition technology to enhance ATM security and user convenience. These findings have broader implications for the banking industry, highlighting the need for secure, user-friendly, and ethical biometric authentication methods. As the adoption of biometric ATM systems continues to grow, it is crucial to consider the ethical and privacy considerations related to biometric data handling, including informed consent, data protection, and regulatory compliance. To further advance the field, future research directions may explore the scalability of biometric ATM systems, address security vulnerabilities, and delve into the nuances of user perceptions and experiences in greater depth. This research sets the stage for innovative advancements in ATM security and biometric authentication, fostering a safer and more user-friendly banking experience for individuals worldwide.

References

[1] J. Carter, "Consumer sentinel network," *Policing: An International Journal of Police Strategies Management*, vol. 31, 11 2008.

[2] M. Guerar, M. Benmohammed, and V. Alimi, "Color wheel pin: Usable and resilient atm authentication," *Journal of High Speed Networks*, vol. 22, no. 3, pp. 231–240, 2016.

[3] A. Lubna, A. Balaji, and K. Manjunath, "Atm weapon, fraud detection and cybercrime prediction," 04 2023.

[4] K. K. Nair, *An approach to authenticate magnetic stripe bank card transactions at point-of-sale terminals*. PhD thesis, North-West University (South Africa), Potchefstroom Campus, 2015.

[5] M. Sharma and S. Jha, "Digital data stealing from atm using data skimmers: Challenge to the forensic examiner," *Journal of Forensic Sciences & Criminal Investigation*, vol. 1, no. 4, 2017.

[6] M. G. Thornhill, *A comparison of United States and United Kingdom credit card security standards*. PhD thesis, Utica College, 2015.

[7] N. K. Ratha, J. H. Connell, and R. M. Bolle, "Enhancing security and privacy in biometrics-based authentication systems," *IBM Systems Journal*, vol. 40, pp. 614–634, 2001.

[8] M. F. Valstar, B. Jiang, M. Mehu, M. Pantic, and K. Scherer, "The first facial expression recognition and analysis challenge," 03 2011.

[9] R. Patriarca, G. Di Gravio, and F. Costantino, "A monte carlo evolution of the functional resonance analysis method (fram) to assess performance variability in complex systems," *Safety science*, vol. 91, pp. 49–60, 2017.

[10] A. A. Trawnih, A. S. Al-Adwan, H. Yaseen, and W. M. Al-Rahmi, "Determining perceptions of banking customers regarding fingerprint atms," *Information Development*, p. 02666669231194360, 2023.

[11] M. Karovaliya, S. Karedia, S. Oza, and D. Kalbande, "Enhanced security for atm machine with otp and facial recognition features," *Procedia Computer Science*, vol. 45, pp. 390–396, 2015.

[12] O. Nathaniel and M. Osuo-Genseleke, "A comparative study of pin based and three-factor based authentication technique for improved atm security," *International Research Journal of Engineering and Technology*, vol. 5, no. 5, pp. 3749–3754, 2018.

[13] W. Aslam, A. Tariq, and I. Arif, "The effect of atm service quality on customer satisfaction and customer loyalty: An empirical analysis," *Global Business Review*, vol. 20, no. 5, pp. 1155–1178, 2019.

[14] A. De Keyser, Y. Bart, X. Gu, S. Q. Liu, S. G. Robinson, and P. Kannan, "Opportunities and challenges of using biometrics for business: Developing a research agenda," *Journal of Business Research*, vol. 136, pp. 52–62, 2021.

[15] N. R. Council, W. B. Committee, *et al.*, "Biometric recognition: Challenges and opportunities," 2010.

[16] G. Lovisotto, R. Malik, I. Sluganovic, M. Roeschlin, P. Trueman, and I. Martinovic, "Mobile biometrics in financial services: A five factor framework," *University of Oxford, Oxford, UK*, 2017.

[17] M. Hernandez-de Menendez, R. Morales-Menendez, C. A. Escobar, and J. Arinez, "Biometric applications in education," *International Journal on Interactive Design and Manufacturing (IJIDeM)*, vol. 15, pp. 365–380, 2021.

[18] I. Cooper and J. Yon, "Ethical issues in biometrics," *Sci Insigt*, vol. 30, no. 2, pp. 63–69, 2019.

[19] O. Tene and J. Polonetsky, "Big data for all: Privacy and user control in the age of analytics," *Nw. J. Tech. & Intell. Prop.*, vol. 11, p. 239, 2012.

[20] P. Samarati, S. Jajodia, *et al.*, "Data security," *Wiley Encyclopedia of Electrical and Electronics Engineering. John Wiley & Sons*, 1999.

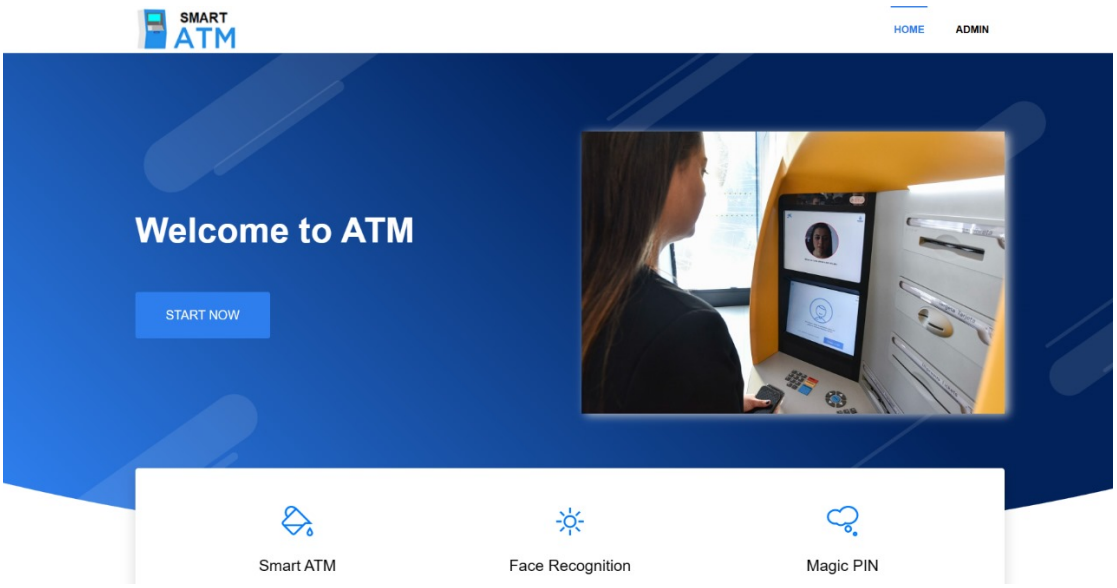
[21] A. Hassanali, "Connecting open data with transparency and accountability to promote value generation: A case study of zambia," 2022.

[22] N. Y. Liu, *Bio-privacy: Privacy regulations and the challenge of biometrics*. Routledge, 2013.

CHAPTER 6

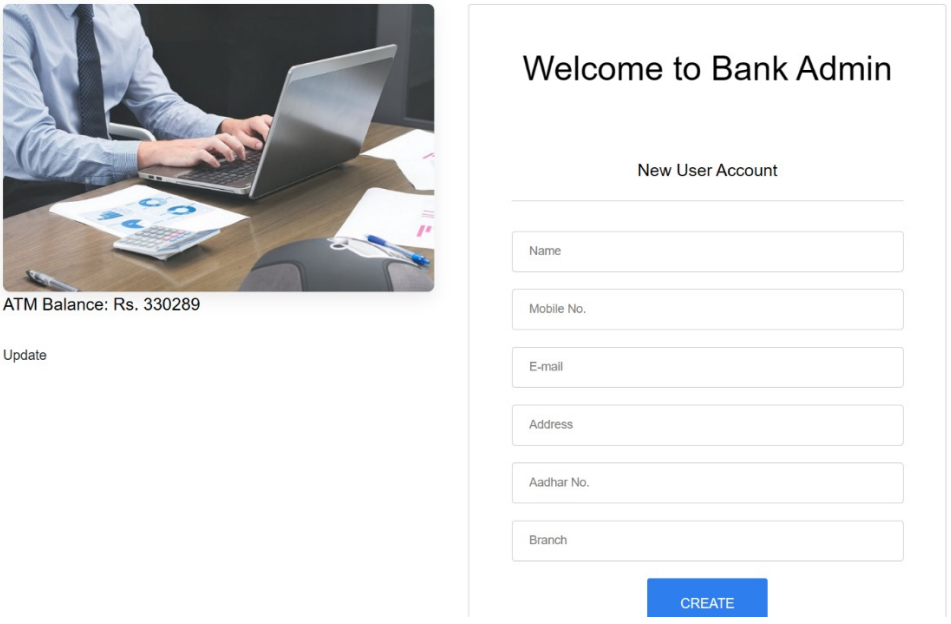
RESULT

6.1 Output



A cutting-edge ATM interface that integrates facial recognition technology for user authentication. Users are greeted with a welcoming screen and prompted to position their face for authentication. Upon successful verification, users access a menu of banking services, enabling seamless transactions such as withdrawals, deposits, and fund transfers. The interface prioritizes security with confirmation prompts and offers accessibility features for all users.

The admin login page where the admin's username and password is entered for creating a user account.



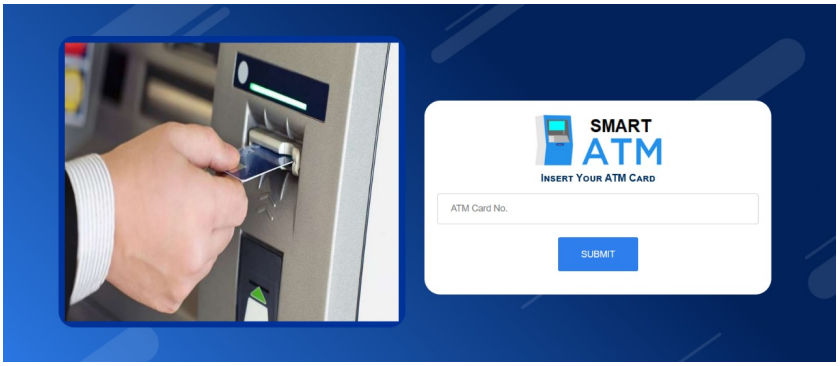
The page where the details of the Account holder such as name,mobile no,email,address,aadhar no,bank branch is entered and the account is created.

Bank Account Holder Details

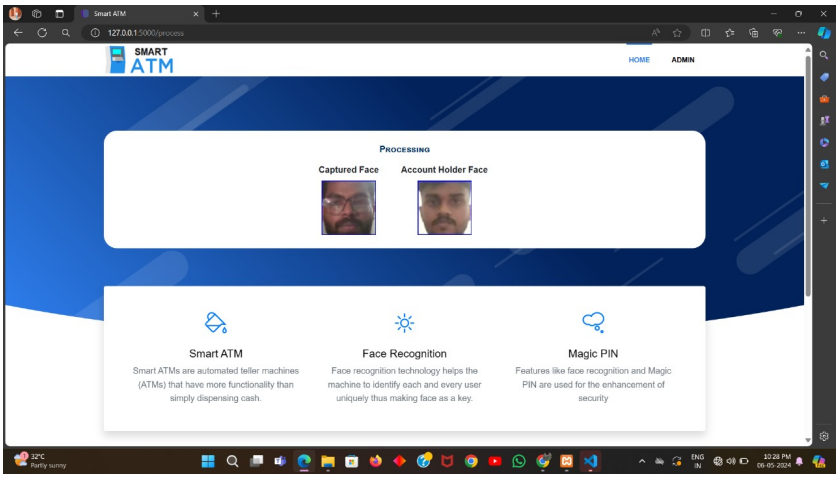
Sno	Name	Address	Mobile	E-mail ID	Account No.	Card No.	Date	Face Template
1	nihal	mea engineering college	9207411618	murshida@meaec.edu.in	2233440001	281400019511	23-04-2024	Add / View
2	nihal	mea engineering college	9207411618	murshida@meaec.edu.in	2233440002	782200027659	23-04-2024	Add / View
3	nadeem	mea engineering college	9207411618	murshida@meaec.edu.in	2233440003	197900033648	23-04-2024	Add / View
4	nadeem	kphouse1	9747952778	kvnihal@gmail.com	2233440004	760000044324	23-04-2024	Add / View
5	ajzal	mea engineering college	8593949006	ajzalmuhammed15@gmail.com	2233440005	923200052350	23-04-2024	Add / View
6	hisham	poolantharakkal	9526080637	20mcs07@meaec.edu.in	2233440006	251100066426	23-04-2024	Add / View
7	nihal mohd	vengooooor	9633820904	jitha@meaec.edu.in	2233440007	164400074527	23-04-2024	Add / View

The page where the whole list of the bank account holder details is available and where the face template of the account user is added.

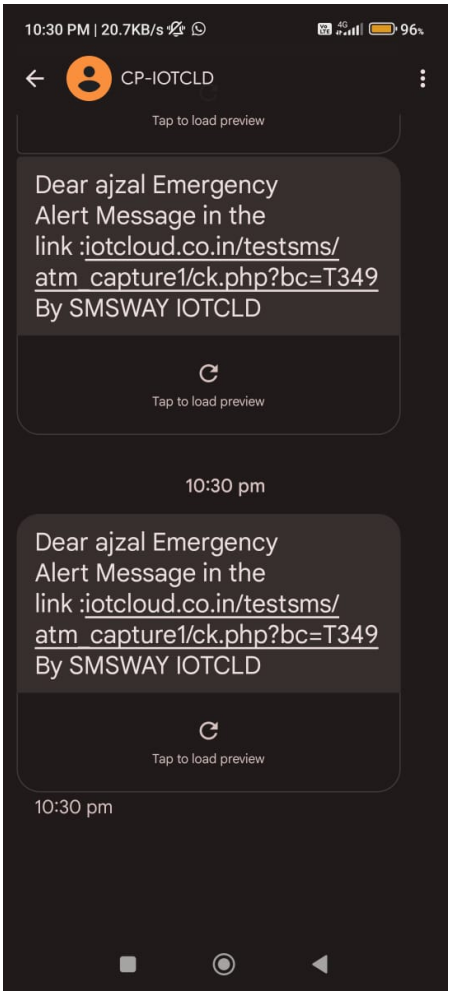
The face templates obtained by the system after the facial recognition.



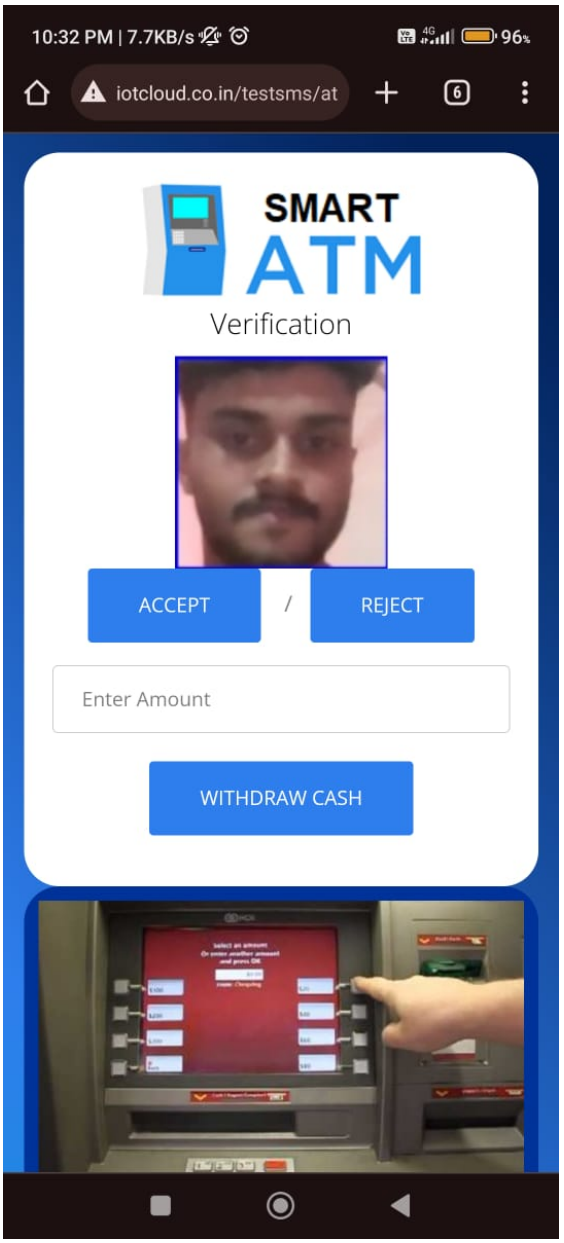
The interface in which the ATM Card number of the user is entered for interaction with the bank server.



The processing page where the image of the user is compared with the account holder.

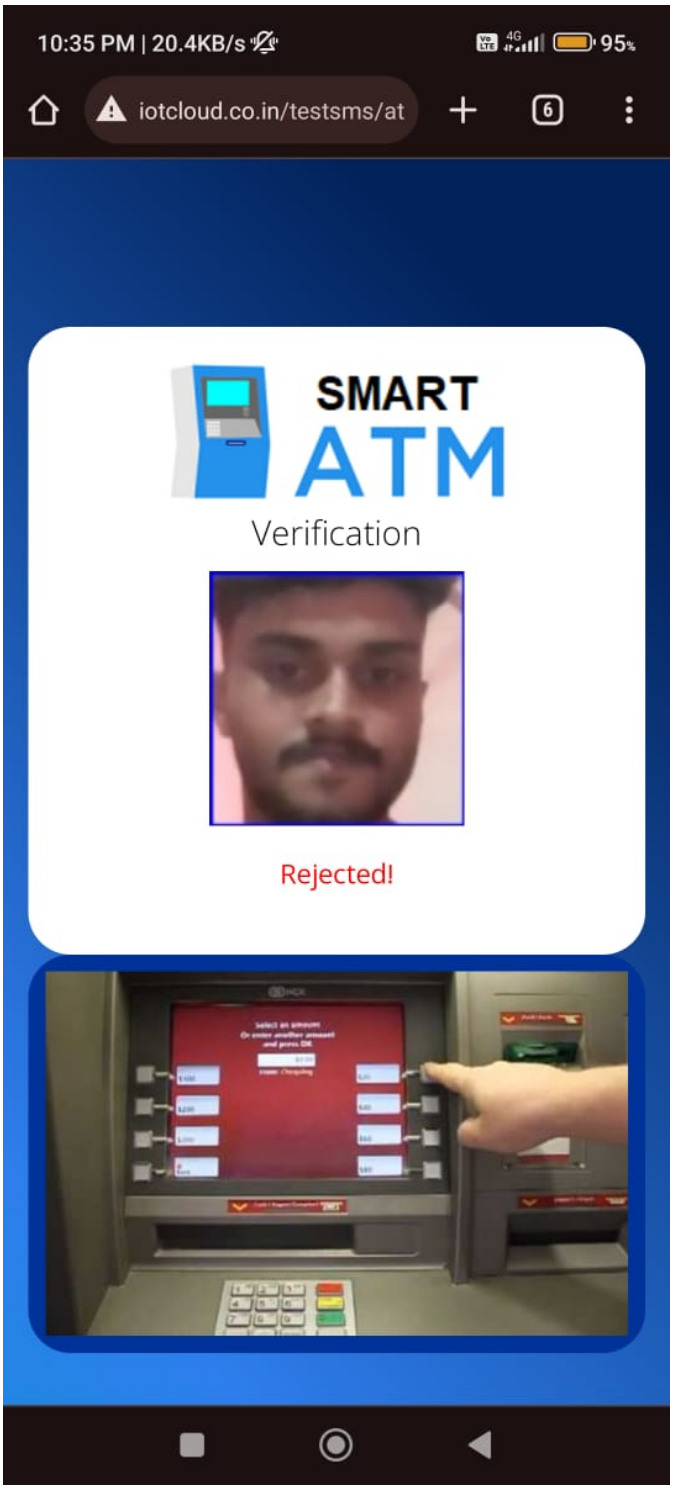


Alert message send to the account holder if the face is not recognized with the link to accept and reject the transaction.



The interface in which the account holder can accept the request and enter the amount to be withdrawn.

The page where Cash withdraw success message displays.



The page where rejected message displays.

6.2 Evaluation Matrices

Evaluation metrics are essential tools for assessing the performance of machine learning models and systems. They provide quantitative measures to gauge how well a model or system performs its intended task. Here's a brief note on some commonly used evaluation metrics.

6.2.1 Precision

- Precision measures the proportion of correctly identified legitimate account holders (true positives) out of all cases identified as positive by the facial recognition system.
- In our project, precision assesses the system's ability to accurately identify genuine users during ATM transactions, minimizing the risk of false positives (incorrectly identifying unauthorized individuals as legitimate account holders).

$$\text{Precision} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Positives}}$$

6.2.2 F1-Score

- F1-score is the harmonic mean of precision and recall, providing a balanced evaluation of both metrics.
- It considers both false positives and false negatives, making it useful for assessing the overall performance of the system.
- In our project, F1-score helps to gauge the system's ability to balance precision and recall, ensuring a reliable and effective biometric authentication process at ATMs.

$$F1 = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

6.2.3 Accuracy

- Accuracy measures the overall correctness of the system's predictions, including both true positives and true negatives, relative to all cases.
- It evaluates the system's ability to correctly authenticate account holders and prevent unauthorized access to ATM services.
- In our project, accuracy reflects the system's reliability in distinguishing between legitimate users and fraudulent attempts, ensuring the security and integrity of ATM transactions.

$$\text{Accuracy} = \frac{\text{True Positives} + \text{True Negatives}}{\text{Total Number of Cases}}$$

6.2.4 Recall

- Recall (also known as sensitivity or true positive rate) measures the proportion of actual legitimate account holders (true positives) that were correctly identified by the facial recognition system.
- It assesses the system's ability to capture all positive instances accurately, minimizing the risk of false negatives (incorrectly rejecting legitimate users).
- In our project, recall ensures that the system effectively identifies and authenticates legitimate users, providing a seamless and secure banking experience at ATMs.

$$\text{Recall} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Negatives}}$$

6.2.5 Evaluation Matrices Results

- Precision: 0.714
- Accuracy: 0.8
- F1-score: 0.83
- Recall: 1.0

CHAPTER 7

IMPLEMENTATION

Python is the programming language chosen for implementation. Python is an interpreted, high-level, and general-purpose programming language that offers several advantages for developing software applications.

One of the key advantages of Python is its simplicity and readability. Python has a clean and easy-to-understand syntax that emphasizes code readability. This readability reduces system maintenance costs by making the code easier to comprehend and debug. The straightforward syntax of Python also facilitates faster development and prototyping, allowing developers to write code more efficiently. Python supports the concept of modules and packages, which promote a modular and organized system layout. Modules allow developers to break down their code into separate, reusable components, enhancing code organization and promoting code reuse. This modularity not only improves the maintainability of the code but also contributes to a more scalable and flexible project structure.

Another aspect mentioned is that Python saves space in terms of code length. Python code tends to be concise and expressive, which means developers can achieve the desired functionality with fewer lines of code compared to other programming languages. This concise nature of Python contributes to improved code readability and reduces the overall complexity of the project.

However, it is worth noticing that Python's interpreted nature can result in slower execution compared to compiled languages. Python code is executed line by line, which can lead to relatively slower performance when dealing with computationally intensive tasks. Nonetheless, Python provides various optimization techniques and libraries that can help mitigate performance concerns when necessary.

Flask is a popular web framework for Python used for developing web applications. Flask is known for its simplicity and minimalism. It doesn't come with built-in bells and whistles, which makes it easy to learn and understand. Developers appreciate its straightforward and unopinionated nature, allowing them to have more control over their code and project structure.

Flask offers developers a high degree of flexibility in building web applications. It doesn't impose a strict architecture or design pattern, allowing developers to choose the tools and libraries that best suit their project requirements. This flexibility enables Flask to be used for a wide range of applications, from simple prototypes to complex, large-scale projects.

Flask is built on Python, which means it embraces Python's philosophy and idioms. If you're already familiar with Python, you'll find Flask's syntax and conventions intuitive and easy to work with. This Pythonic approach makes Flask an attractive choice for Python developers who want to leverage their existing skills and knowledge.

Flask follows a modular design, with core functionality kept to a minimum. Instead, additional features can be added through extensions. This modular architecture allows developers to keep their applications lightweight by only including the components they need. It also encourages code reuse and makes it easy to integrate third-party libraries and tools.

While Flask is lightweight and suitable for small projects, it can also scale up to handle larger applications with ease. Its simplicity and flexibility make it adaptable to changing requirements and evolving project needs. Additionally, Flask can be combined with other tools, such as SQL Alchemy for database management and Celery for task scheduling, to further enhance scalability and performance.

Flask's simplicity, flexibility, Pythonic nature, modularity, scalability, and strong community support make it an excellent choice for building web applications in Python. Whether you're developing a small prototype or a large-scale production application, Flask provides the tools and flexibility you need to get the job done efficiently and effectively.

MySQL is a widely used open-source relational database management system (RDBMS) that provides a robust and scalable platform for storing, organizing, and managing data.

MySQL allows you to create databases to store your data. A database is a collection of related tables that hold structured information. Each table consists of rows and columns, where each row represents a single record and each column represents a specific attribute or field of that record.MySQL provides a powerful set of Data Definition Language (DDL) commands for defining and managing database objects such as tables, indexes, views, and triggers MySQL supports Data Manipulation Language (DML) commands for querying and manipulating data within tables.

MySQL provides robust security features to protect your data, including user authentication, access control, and encryption. You can create multiple user accounts with different levels of privileges, allowing you to control who can access and modify the data in your databases. MySQL offers tools and mechanisms for backing up and restoring databases, ensuring that your data is protected against accidental loss or corruption. You can perform full or incremental backups, schedule automated backups, and restore data to a previous state in case of emergencies.

MySQL provides a comprehensive set of features and capabilities for creating and managing databases, making it a popular choice for a wide range of applications, from small-scale projects to enterprise-level systems. Its ease of use, reliability, performance, and scalability make it a reliable option for storing and accessing data effectively.

7.1 Implementation code

7.1.1 Camera

```
# camera.py

import cv2
import PIL.Image
from PIL import Image
class VideoCamera(object):
    def __init__(self):
        # Using OpenCV to capture from device 0. If you have trouble capturing
        # from a webcam, comment the line below out and use a video file
        # instead.

        self.video = cv2.VideoCapture(0)
        self.k=1
        #cap = self.video
        # If you decide to use video.mp4, you must have this file in the folder
        # as the main.py.
        #self.video = cv2.VideoCapture('video.mp4')

        # Check if camera opened successfully
        #if (cap.isOpened() == False):
        #  print("Unable to read camera feed")

        # Default resolutions of the frame are obtained.The default resolutions are system dependent.
        # We convert the resolutions from float to integer.
        #frame_width = int(cap.get(3))
        #frame_height = int(cap.get(4))

        # Define the codec and create VideoWriter object.The output is stored in 'outpy.avi' file.
        #self.out = cv2.VideoWriter('video.avi',cv2.VideoWriter_fourcc('M','J','P','G'), 10, (frame_width,frame_height))


    def __del__(self):
        self.video.release()


    def get_frame(self):
        success, image = self.video.read()
        #self.out.write(image)
        face_cascade = cv2.CascadeClassifier('haarcascade_frontalface_default.xml')

        # Read the frame
        #_, img = cap.read()

        # Convert to grayscale
        gray = cv2.cvtColor(image, cv2.COLOR_BGR2GRAY)

        # Detect the faces
        faces = face_cascade.detectMultiScale(gray, 1.1, 4)

        # Draw the rectangle around each face
        j = 1

        ff=open("user.txt","r")
        uu=ff.read()
        ff.close()
```

```
ff=open("user.txt","r")
uu=ff.read()
ff.close()

ff1=open("photo.txt","r")
uu1=ff1.read()
ff1.close()



for (x, y, w, h) in faces:
    mm=cv2.rectangle(image, (x, y), (x+w, y+h), (255, 0, 0), 2)
    cv2.imwrite("myface.jpg", mm)


    image = cv2.imread("myface.jpg")
    cropped = image[y:y+h, x:x+w]
    gg="f"+str(j)+".jpg"
    cv2.imwrite("faces/"+gg, cropped)

    ###
    if self.k<=40:
        self.k+=1
        fnn=uu+"_"+str(self.k)+".jpg"

        ff2=open("det.txt","w")
        ff2.write(str(self.k))
        ff2.close()
        if uu1=="2":
            cv2.imwrite("static/frame/"+fnn, cropped)
        #cv2.imwrite("https://iotcloud.co.in/testsms/upload/"+fnn, cropped)
        ###
        mm2 = PIL.Image.open('faces/'+gg)
        rz = mm2.resize((100,100), PIL.Image.Resampling.LANCZOS)
        rz.save('faces/'+gg)
        #rz.save("https://iotcloud.co.in/testsms/upload/"+gg)
    j += 1

ff4=open("img.txt","w")
ff4.write(str(j))
ff4.close()




# We are using Motion JPEG, but OpenCV defaults to capture raw images,
# so we must encode it into JPEG in order to correctly display the
# video stream.
ret, jpeg = cv2.imencode('.jpg', image)
# return image.nbytes()
return jpeg.tobytes()
```

7.2 Main

```
from flask import Flask
from flask import Flask, render_template, Response, redirect, request, session, abort, url_for
from camera import VideoCamera
from datetime import datetime
from datetime import date
import datetime
import random
from random import seed
from random import randint
import cv2
import numpy as np
import threading
import os
import time
import shutil
import imagehash
import PIL.Image
from PIL import Image
from PIL import ImageTk
import urllib.request
import urllib.parse
from urllib.request import urlopen
import webbrowser

import mysql.connector

mydb = mysql.connector.connect(
  host="localhost",
  user="root",
  passwd="",
  charset="utf8",
  database="smart_atm"
)

app = Flask(__name__)
##session key
app.secret_key = 'abcdef'
@app.route('/',methods=['POST','GET'])
def index():
    cnt=0
    act=""
    msg=""
    ff=open("det.txt","w")
    ff.write("1")
    ff.close()

    ff1=open("photo.txt","w")
    ff1.write("1")
    ff1.close()

    ff11=open("img.txt","w")
    ff11.write("1")
    ff11.close()


    return render_template('index.html',msg=msg,act=act)
```

```
@app.route('/verify_card',methods=['POST','GET'])
def verify_card():
    cnt=0
    act=""
    msg=""

    if request.method=='POST':
        card=request.form['card']

        mycursor = mydb.cursor()
        mycursor.execute("SELECT count(*) FROM register where card=%s",(card, ))
        cnt = mycursor.fetchone()[0]
        if cnt>0:
            msg="success"
            session['username'] = card
            ff2=open("un.txt","w")
            ff2.write(card)
            ff2.close()
            return redirect(url_for('verify_face'))


        else:
            msg="Card No. is wrong!"
            print("Incorrect")


    return render_template('verify_card.html',msg=msg,act=act)

########################

@app.route('/register',methods=['POST','GET'])
def register():
    result=""
    act=""
    if request.method=='POST':
        name=request.form['name']
        mobile=request.form['mobile']
        email=request.form['email']
        address=request.form['address']
        bank=request.form['bank']
        branch=request.form['branch']
        card=request.form['card']
        account=request.form['accno']
        uname=request.form['username']
        password=request.form['password']

        aadhar1=request.form['aadhar1']
        aadhar2=request.form['aadhar2']
        aadhar3=request.form['aadhar3']

        face_st=request.form['face_st']


        now = datetime.datetime.now()
        rdate=now.strftime("%d-%m-%Y")
        mycursor = mydb.cursor()
```

```
        mycursor.execute("SELECT count(*) FROM register where card=%s",(card, ))
        cnt = mycursor.fetchone()[0]
        if cnt==0:
            mycursor.execute("SELECT max(id)+1 FROM register")
            maxid = mycursor.fetchone()[0]
            if maxid is None:
                maxid=1
            sql = "INSERT INTO register(id, name, mobile, email, address,  bank,
            accno, branch, card, deposit, username, password, rdate, aadhar1, aadhar2,
            aadhar3, face_st, fimg) VALUES (%s, %s, %s, %s, %s, %s, %s, %s, %s, %s, %s,
             %s, %s, %s, %s, %s, %s, %s)"
            val = (maxid, name, mobile, email, address, bank, account, branch, card,
             '10000', uname, password, rdate, aadhar1, aadhar2, aadhar3, face_st, '')
            print(sql)
            mycursor.execute(sql, val)
            mydb.commit()
            print(mycursor.rowcount, "record inserted.")
            if face_st=="1":
                return redirect(url_for('add_photo',vid=maxid))
            #if mycursor.rowcount==1:
            #    result="Registered Success"
            else:
                return redirect(url_for('index',act='success'))
        else:
            result="Card No. already Exist!"
    return render_template('register.html',result=result)

@app.route('/login_admin', methods=['POST','GET'])
def login_admin():
    result=""
    ff1=open("photo.txt","w")
    ff1.write("1")
    ff1.close()
    if request.method == 'POST':
        username1 = request.form['uname']
        password1 = request.form['pass']
        mycursor = mydb.cursor()
        mycursor.execute("SELECT count(*) FROM admin where username=%s && password=%s",
        (username1,password1))
        myresult = mycursor.fetchone()[0]
        if myresult>0:
            result=" Your Logged in sucessfully**"
            return redirect(url_for('admin'))
        else:
            result="Incorrect Username or Password!!!"


    return render_template('login_admin.html',result=result)

@app.route('/admin',methods=['POST','GET'])
def admin():
    msg=""
    ff1=open("photo.txt","w")
    ff1.write("2")
    ff1.close()
```

```
    mycursor = mydb.cursor()
    if request.method=='POST':
        name=request.form['name']
        mobile=request.form['mobile']
        email=request.form['email']
        address=request.form['address']
        branch=request.form['branch']
        aadhar=request.form['aadhar']

        now = datetime.datetime.now()
        rdate=now.strftime("%d-%m-%Y")


        mycursor.execute("SELECT count(*) FROM register where aadhar1=%s",(aadhar, ))
        cnt = mycursor.fetchone()[0]
        if cnt==0:
            mycursor.execute("SELECT max(id)+1 FROM register")
            maxid = mycursor.fetchone()[0]
            if maxid is None:
                maxid=1

            str1=str(maxid)
            ac=str1.rjust(4, "0")
            account="223344"+ac

            xn=randint(1000, 9999)
            rv1=str(xn)
            xn2=randint(1000, 9999)
            rv2=str(xn2)
            card=rv1+ac+rv2
            bank="SBI"

            xn3=randint(1000, 9999)
            pinno=str(xn3)


            sql = "INSERT INTO register(id, name, mobile, email, address,  bank, accno,
            branch, card, deposit,password, rdate, aadhar1)
            VALUES (%s, %s, %s, %s, %s, %s, %s, %s, %s, %s, %s, %s, %s)"
            val = (maxid, name, mobile, email, address, bank, account, branch, card,
            '10000',pinno, rdate, aadhar)
            print(sql)
            mycursor.execute(sql, val)
            mydb.commit()
            message="Dear "+name+", Your Bank Account created, Account No.:"+account+",
            Debit Card No."+card+", Pinno:"+pinno
            url="http://iotcloud.co.in/testmail/sendmail.php?email="+email+"&message="+message
            webbrowser.open_new(url)

            return redirect(url_for('add_photo',vid=maxid))
        else:
            msg="Already Exist!"

    mycursor.execute("SELECT amount FROM admin WHERE username='admin'")
    value = mycursor.fetchone()[0]

    return render_template('admin.html',msg=msg,value=value)
```

```
@app.route('/add_photo',methods=['POST','GET'])
def add_photo():
    vid=""
    ff1=open("photo.txt","w")
    ff1.write("2")
    ff1.close()
    if request.method=='GET':
        vid = request.args.get('vid')
        ff=open("user.txt","w")
        ff.write(vid)
        ff.close()

    if request.method=='POST':
        vid=request.form['vid']
        fimg="v"+vid+".jpg"
        cursor = mydb.cursor()

        cursor.execute('delete from vt_face WHERE vid = %s', (vid, ))
        mydb.commit()

        ff=open("det.txt","r")3
        v=ff.read()
        ff.close()
        vv=int(v)
        v1=vv-1
        vface1=vid+"_"+str(v1)+".jpg"
        i=2
        while i<vv:

            cursor.execute("SELECT max(id)+1 FROM vt_face")
            maxid = cursor.fetchone()[0]
            if maxid is None:
                maxid=1
            vface=vid+"_"+str(i)+".jpg"
            sql = "INSERT INTO vt_face(id, vid, vface) VALUES (%s, %s, %s)"
            val = (maxid, vid, vface)
            print(val)
            cursor.execute(sql,val)
            mydb.commit()
            i+=1



        cursor.execute('update register set fimg=%s WHERE id = %s', (vface1, vid))
        mydb.commit()
        shutil.copy('faces/f1.jpg', 'static/photo/'+vface1)
        return redirect(url_for('view_cus',vid=vid,act='success'))

    cursor = mydb.cursor()
    cursor.execute("SELECT * FROM register")
    data = cursor.fetchall()
    return render_template('add_photo.html',data=data, vid=vid)
```

```
@app.route('/view_cus',methods=['POST','GET'])
def view_cus():
    mycursor = mydb.cursor()
    mycursor.execute("SELECT * FROM register")
    value = mycursor.fetchall()
    return render_template('view_cus.html', result=value)

###Preprocessing
@app.route('/view_photo',methods=['POST','GET'])
def view_photo():
    ff1=open("photo.txt","w")
    ff1.write("1")
    ff1.close()
    vid=""
    value=[]
    if request.method=='GET':
        vid = request.args.get('vid')
        mycursor = mydb.cursor()
        mycursor.execute("SELECT * FROM vt_face where vid=%s",(vid, ))
        value = mycursor.fetchall()

    if request.method=='POST':
        print("Training")
        vid=request.form['vid']
        cursor = mydb.cursor()
        cursor.execute("SELECT * FROM vt_face where vid=%s",(vid, ))
        dt = cursor.fetchall()
        for rs in dt:
            ##Preprocess
            path="static/frame/"+rs[2]
            path2="static/process1/"+rs[2]
            mm2 = PIL.Image.open(path).convert('L')
            rz = mm2.resize((200,200), PIL.Image.ANTIALIAS)
            rz.save(path2)

            '''img = cv2.imread(path2)
            dst = cv2.fastNlMeansDenoisingColored(img, None, 10, 10, 7, 15)
            path3="static/process2/"+rs[2]
            cv2.imwrite(path3, dst)'''
            ######
            img = cv2.imread(path2)
            gray = cv2.cvtColor(img,cv2.COLOR_BGR2GRAY)
            ret, thresh = cv2.threshold(gray,0,255,cv2.THRESH_BINARY_INV+cv2.THRESH_OTSU)

            # noise removal
            kernel = np.ones((3,3),np.uint8)
            opening = cv2.morphologyEx(thresh,cv2.MORPH_OPEN,kernel, iterations = 2)

            # sure background area
            sure_bg = cv2.dilate(opening,kernel,iterations=3)

            # Finding sure foreground area
            dist_transform = cv2.distanceTransform(opening,cv2.DIST_L2,5)
            ret, sure_fg = cv2.threshold(dist_transform,0.5*dist_transform.max(),255,0)
```

```
            # Finding unknown region
            sure_fg = np.uint8(sure_fg)
            segment = cv2.subtract(sure_bg,sure_fg)
            img = Image.fromarray(img)
            segment = Image.fromarray(segment)
            path3="static/process2/"+rs[2]
            segment.save(path3)

            #####
            image = cv2.imread(path2)
            gray = cv2.cvtColor(image, cv2.COLOR_BGR2GRAY)
            edged = cv2.Canny(gray, 50, 100)
            image = Image.fromarray(image)
            edged = Image.fromarray(edged)
            path4="static/process3/"+rs[2]
            edged.save(path4)
            ##
            shutil.copy('static/images/11.png', 'static/process4/'+rs[2])

        return redirect(url_for('view_photo1',vid=vid))

    return render_template('view_photo.html', result=value,vid=vid)

###Segmentation using RNN
def crfrnn_segmenter(model_def_file, model_file, gpu_device, inputs):

    assert os.path.isfile(model_def_file), "File {} is missing".format(model_def_file)
    assert os.path.isfile(model_file), ("File {} is missing. Please download it using "
                                        "./download_trained_model.sh").format(model_file)

    if gpu_device >= 0:
        caffe.set_device(gpu_device)
        caffe.set_mode_gpu()
    else:
        caffe.set_mode_cpu()

    net = caffe.Net(model_def_file, model_file, caffe.TEST)

    num_images = len(inputs)
    num_channels = inputs[0].shape[2]
    assert num_channels == 3, "Unexpected channel count. A 3-channel RGB image is exptected."

    caffe_in = np.zeros((num_images, num_channels, _MAX_DIM, _MAX_DIM), dtype=np.float32)
    for ix, in_ in enumerate(inputs):
        caffe_in[ix] = in_.transpose((2, 0, 1))

    start_time = time.time()
    out = net.forward_all(**{net.inputs[0]: caffe_in})
    end_time = time.time()

    print("Time taken to run the network: {:.4f} seconds".format(end_time - start_time))
    predictions = out[net.outputs[0]]

    return predictions[0].argmax(axis=0).astype(np.uint8)
```

```
def run_crfrnn(input_file, output_file, gpu_device):
    """ Runs the CRF-RNN segmentation on the given RGB image and saves the segmentation mask.
    Args:
        input_file: Input RGB image file (e.g. in JPEG format)
        output_file: Path to save the resulting segmentation in PNG format
        gpu_device: ID of the GPU device. If using the CPU, set this to -1
    """

    input_image = 255 * caffe.io.load_image(input_file)
    input_image = resize_image(input_image)

    image = PILImage.fromarray(np.uint8(input_image))
    image = np.array(image)

    palette = get_palette(256)
    #PIL reads image in the form of RGB, while cv2 reads image in the form of BGR, mean_vec = [R,G,B]
    mean_vec = np.array([123.68, 116.779, 103.939], dtype=np.float32)
    mean_vec = mean_vec.reshape(1, 1, 3)

    # Rearrange channels to form BGR
    im = image[:, :, ::-1]
    # Subtract mean
    im = im - mean_vec

    # Pad as necessary
    cur_h, cur_w, cur_c = im.shape
    pad_h = _MAX_DIM - cur_h
    pad_w = _MAX_DIM - cur_w
    im = np.pad(im, pad_width=((0, pad_h), (0, pad_w), (0, 0)), mode='constant', constant_values=0)

    # Get predictions
    segmentation = crfrnn_segmenter(_MODEL_DEF_FILE, _MODEL_FILE, gpu_device, [im])
    segmentation = segmentation[0:cur_h, 0:cur_w]

    output_im = PILImage.fromarray(segmentation)
    output_im.putpalette(palette)
    output_im.save(output_file)
###Feature extraction & Classification
def DCNN_process(self):

        train_data_preprocess = ImageDataGenerator(
                rescale = 1./255,
                shear_range = 0.2,
                zoom_range = 0.2,
                horizontal_flip = True)

        test_data_preprocess = (1./255)

        train = train_data_preprocess.flow_from_directory(
                'dataset/training',
                target_size = (128,128),
                batch_size = 32,
                class_mode = 'binary')

        test = train_data_preprocess.flow_from_directory(
                'dataset/test',
                target size = (128,128),
```

```
                batch_size = 32,
                class_mode = 'binary')

        history = cnn.fit_generator(train,
                                    steps_per_epoch = 250,
                                    epochs = 25,
                                    validation_data = test,
                                    validation_steps = 2000)

        plt.plot(history.history['acc'])
        plt.plot(history.history['val_acc'])
        plt.title('Model Accuracy')
        plt.ylabel('accuracy')
        plt.xlabel('epoch')
        plt.legend(['train', 'test'], loc='upper left')
        plt.show()

        plt.plot(history.history['loss'])
        plt.plot(history.history['val_loss'])
        plt.title('Model Loss')
        plt.ylabel('loss')
        plt.xlabel('epoch')
        plt.legend(['train', 'test'], loc='upper left')
        plt.show()

        test_image = image.load_img('\\dataset\\', target_size=(128,128))
        test_image = image.img_to_array(test_image)
        test_image = np.expand_dims(test_image, axis=0)
        result = cnn.predict(test_image)
        print(result)

        if result[0][0] == 1:
                print('feature extracted and classified')
        else:
                print('none')

@app.route('/view_photo1',methods=['POST','GET'])
def view_photo1():
    vid=""
    value=[]
    if request.method=='GET':
        vid = request.args.get('vid')
        mycursor = mydb.cursor()
        mycursor.execute("SELECT * FROM vt_face where vid=%s",(vid, ))
        value = mycursor.fetchall()
    return render_template('view_photo1.html', result=value,vid=vid)

@app.route('/view_photo2',methods=['POST','GET'])
def view_photo2():
    vid=""
    value=[]
    if request.method=='GET':
        vid = request.args.get('vid')
        mycursor = mydb.cursor()
        mycursor.execute("SELECT * FROM vt_face where vid=%s",(vid, ))
        value = mycursor.fetchall()
    return render_template('view_photo2.html', result=value,vid=vid)
```

```
@app.route('/view_photo3',methods=['POST','GET'])
def view_photo3():
    vid=""
    value=[]
    if request.method=='GET':
        vid = request.args.get('vid')
        mycursor = mydb.cursor()
        mycursor.execute("SELECT * FROM vt_face where vid=%s",(vid, ))
        value = mycursor.fetchall()
    return render_template('view_photo3.html', result=value,vid=vid)

@app.route('/view_photo4',methods=['POST','GET'])
def view_photo4():
    vid=""
    value=[]
    if request.method=='GET':
        vid = request.args.get('vid')
        mycursor = mydb.cursor()
        mycursor.execute("SELECT * FROM vt_face where vid=%s",(vid, ))
        value = mycursor.fetchall()
    return render_template('view_photo4.html', result=value,vid=vid)

@app.route('/message',methods=['POST','GET'])
def message():
    vid=""
    name=""
    if request.method=='GET':
        vid = request.args.get('vid')
        mycursor = mydb.cursor()
        mycursor.execute("SELECT name FROM register where id=%s",(vid, ))
        name = mycursor.fetchone()[0]
    return render_template('message.html',vid=vid,name=name)


@app.route('/login',methods=['POST','GET'])
def login():
    uname=""
##    value=["1","2","3","4","5","6","7","8","9","0"]
##    change=random.shuffle(value)
##    print(change)
    if 'username' in session:
        uname = session['username']
    print(uname)
    mycursor1 = mydb.cursor()

    mycursor1.execute("SELECT * FROM register where card=%s",(uname, ))
    value = mycursor1.fetchone()
    accno=value[5]
    session['accno'] = accno

    mycursor1.execute("SELECT number FROM numbers order by rand()")
    value = mycursor1.fetchall()
    msg=""

    if request.method == 'POST':
        password1 = request.form['password']
        mycursor = mydb.cursor()
        mycursor.execute("SELECT count(*) FROM register where card=%s && password=%s",(uname, password1))
```

```
        myresult = mycursor.fetchone()[0]
        if password1=="":

            return render_template('login.html')
        else:

            #if str(password1)==str(myresult[10]):
            if myresult>0:
                #ff2=open("log.txt","w")
                #ff2.write(password1)
                #ff2.close()
                result=" Your Logged in sucessfully**"

                return redirect(url_for('userhome'))
            else:
                msg="Your logged in fail!!!"
                #return render_template('userhome.html',result=result)


    return render_template('login.html',value=value,msg=msg)



@app.route('/userhome')
def userhome():
    uname=""
    #if 'username' in session:
    #    uname = session['username']
    ff2=open("un.txt","r")
    uname=ff2.read()
    ff2.close()

    name=""




    print(uname)
    mycursor1 = mydb.cursor()
    mycursor1.execute("SELECT * FROM register where card=%s",(uname, ))
    value = mycursor1.fetchone()
    print(value)
    name=value[1]

    return render_template('userhome.html',name=name)

'''@app.route('/deposit')
def deposit():
    return render_template('deposit.html')
@app.route('/deposit_amount',methods=['POST','GET'])
def deposit_amount():
    if request.method=='POST':
        name=request.form['name']
        accountno=request.form['accno']
        amount=request.form['amount']
        today = date.today()
        rdate = today.strftime("%b-%d-%Y")
        mycursor = mydb.cursor()
        mycursor.execute("SELECT max(id)+1 FROM event")
```

```
        maxid = mycursor.fetchone()[0]
        sql = "INSERT INTO event(id, name, accno, amount, rdate) VALUES (%s, %s, %s, %s, %s)"
        val = (maxid, name, accountno, amount, rdate)
        mycursor.execute(sql, val)
        mydb.commit()
    return render_template('userhome.html')'''

'''@app.route('/withdraw')
def withdraw():


    return render_template('withdraw.html')'''

@app.route('/verify_face',methods=['POST','GET'])
def verify_face():
    msg=""
    ss=""
    uname=""
    act=""
    if request.method=='GET':
        act = request.args.get('act')

    #if 'username' in session:
    #    uname = session['username']
    ff2=open("un.txt","r")
    uname=ff2.read()
    ff2.close()


    return render_template('verify_face.html',msg=msg)

@app.route('/face',methods=['POST','GET'])
def face():
    msg=""
    ss=""
    uname=""
    act=""
    if request.method=='GET':
        act = request.args.get('act')

    #if 'username' in session:
    #    uname = session['username']
    ff2=open("un.txt","r")
    uname=ff2.read()
    ff2.close()
    print("uname="+uname)
    shutil.copy('faces/f1.jpg', 'static/f1.jpg')

    ff3=open("img.txt","r")
    mcnt=ff3.read()
    ff3.close()

    mcnt1=int(mcnt)
    if mcnt1==2:
        msg="Face Detected"
    elif mcnt1>2:
        msg="Multiple Face Detected!"
    else:
        msg=""
```

```
@app.route('/process',methods=['POST','GET'])
def process():
    vid=""
    pg="0"
    act="1"
    uname=""
    #if 'username' in session:
    #    uname = session['username']
    ff2=open("un.txt","r")
    uname=ff2.read()
    ff2.close()
    value=[]
    shutil.copy('faces/f1.jpg', 'static/f1.jpg')
    cursor = mydb.cursor()
    cursor.execute('SELECT * FROM register WHERE card = %s', (uname, ))
    account = cursor.fetchone()
    name=account[1]
    mobile=account[3]

    email=account[4]
    vid=account[0]
    cursor.execute("SELECT vface FROM vt_face where vid=%s limit 0,1",(vid, ))
    value = cursor.fetchone()[0]


    return render_template('process.html', vid=vid,pg=pg,act=act,result=value)

@app.route('/pro',methods=['POST','GET'])
def pro():
    vid=""
    value=[]
    pgg=0
    act="1"
    uname=""
    #if 'username' in session:
    #    uname = session['username']
    ff2=open("un.txt","r")
    uname=ff2.read()
    ff2.close()
    if request.method=='GET':
        act = request.args.get('act')

        vid = request.args.get('vid')
        pg = request.args.get('pg')
        #pgg=int(pg)+1
        pgg=2
        mycursor = mydb.cursor()
        mycursor.execute("SELECT count(*) FROM vt_face where vid=%s",(vid,))
        dtt = mycursor.fetchone()[0]

        if dtt<=pgg:
            act="1"
        else:
            act="2"

        mycursor.execute("SELECT vface FROM vt_face where vid=%s limit 0,1",(vid, ))
        value = mycursor.fetchone()[0]
```

```
        #print(value)

    return render_template('pro.html', result=value,vid=vid,pg=pgg,act=act)

@app.route('/verify_face2',methods=['POST','GET'])
def verify_face2():
    msg=""
    ss=""
    uname=""
    act=""
    if request.method=='GET':
        act = request.args.get('act')

    #if 'username' in session:
    #    uname = session['username']
    ff2=open("un.txt","r")
    uname=ff2.read()
    ff2.close()

    ff2=open("bc.txt","r")
    bc=ff2.read()
    ff2.close()

    cursor = mydb.cursor()
    cursor.execute('SELECT * FROM register WHERE card = %s', (uname, ))
    account = cursor.fetchone()
    name=account[1]
    mobile=account[3]
    print(mobile)
    email=account[4]
    vid=account[0]


    shutil.copy('faces/f1.jpg', 'faces/s1.jpg')
    cutoff=5
    img="v"+str(vid)+".jpg"
    cursor.execute('SELECT * FROM vt_face WHERE vid = %s', (vid, ))
    dt = cursor.fetchall()
    for rr in dt:
        hash0 = imagehash.average_hash(Image.open("static/frame/"+rr[2]))
        hash1 = imagehash.average_hash(Image.open("faces/s1.jpg"))
        cc1=hash0 - hash1
        print("cc="+str(cc1))
        if cc1<=cutoff:
            ss="ok"
            break
        else:
            ss="no"

    if ss=="ok":
        act="2"
        msg="Face Verified"
        print("correct person")
        return redirect(url_for('userhome', msg=msg))
    else:
        act="1"
        msg="Face not Verified"
        print("wrong person")
        #xn=randint(1000, 9999)
        #otp=str(xn)
```

```
        #cursor1 = mydb.cursor()
        #cursor1.execute('update register set otp=%s WHERE card = %s', (otp, uname))
        #mydb.commit()

        mess="Someone Access your account"
        # url2="http://127.0.0.1:5000/atm/img.txt"
        # ur = urlopen(url2)#open url
        # data1 = ur.read().decode('utf-8')


        # idd=int(data1)+1
        url="http://iotcloud.co.in/testsms/sms.php?sms=link11&name="+name+"&mess=
        "+mess+"&mobile="+str(mobile)+"&bc="+bc
        print(url)
        webbrowser.open_new(url)


    return render_template('verify_face2.html',msg=msg,act=act)

@app.route('/cap',methods=['POST','GET'])
def cap():
    msg=""

    ff2=open("bc.txt","r")
    bc=ff2.read()
    ff2.close()



    return render_template('cap.html',msg=msg,bc=bc)

@app.route('/verify',methods=['POST','GET'])
def verify():
    msg=""
    data1=""
    #act=""
    amtt=""
    cc=""
    ff2=open("un.txt","r")
    uname=ff2.read()
    ff2.close()
    #data1="4"
    url2="http://localhost/atm/log.txt"
    ur = urlopen(url2)#open url
    data1 = ur.read().decode('utf-8')
    vv=data1.split('-')
    data1=vv[0]
    amtt=vv[1]
    print(data1)

    act = request.args.get('act')
    if act is None:
        act=""

    print("act="+str(act))
    if data1=="accept":
        act="1"
```

```
if act=="3":
    amt=0
    amt1=0
    amt2=0


    amount1=amtt

    mycursor = mydb.cursor()

    mycursor.execute("SELECT amount FROM admin where username='admin'")
    amt1 = mycursor.fetchone()[0]

    mycursor.execute("SELECT deposit FROM register where card=%s",(uname, ))
    amt2 = mycursor.fetchone()[0]

    mycursor.execute("SELECT * FROM register where card=%s",(uname, ))
    ddt = mycursor.fetchone()
    name=ddt[1]
    mobile=ddt[3]

    amt=int(amount1)
    if amt<=amt1:

        if amt<=amt2:
            mycursor.execute("UPDATE admin SET amount=amount-%s WHERE username='admin'",(amount1, ))
            mydb.commit()
            mycursor.execute("UPDATE register SET deposit=deposit-%s WHERE card=%s",(amount1, uname))
            mydb.commit()

            now = datetime.datetime.now()
            rdate=now.strftime("%d-%m-%Y")
            mycursor.execute("SELECT max(id)+1 FROM event")
            maxid = mycursor.fetchone()[0]
            if maxid is None:
                maxid=1
            sql = "INSERT INTO event(id, name, accno, amount, rdate) VALUES (%s, %s, %s, %s, %s)"
            val = (maxid, name, uname, amt, rdate)
            mycursor.execute(sql, val)
            mydb.commit()

            mess="Amount Debited Rs."+str(amt)
            url="http://iotcloud.co.in/testsms/sms.php?sms=msg&name="+name+"&mess="+mess+"&mobile="+str(mobile)
            webbrowser.open_new(url)

            msg="Withdraw success..."
        else:
            mess="Your Account balance is low!"
            url="http://iotcloud.co.in/testsms/sms.php?sms=emr&name="+name+"&mess="+mess+"&mobile="+str(mobile)
            webbrowser.open_new(url)
            msg="Your Account balance is low!"
    else:
        msg="Cash is not available in ATM!!"


return render_template('verify.html',msg=msg,act=act,amtt=amtt,data1=data1)
```

```
@app.route('/otp', methods=['GET', 'POST'])
def otp():
    msg=""
    key=""
    if 'username' in session:
        uname = session['username']
    cursor = mydb.cursor()
    cursor.execute('SELECT otp FROM register WHERE card = %s', (uname, ))
    account = cursor.fetchone()[0]
    key=account

    if request.method=='POST':
        otp=request.form['otp']

        if otp==key:
            session['username'] = uname

            return redirect(url_for('verify_aadhar'))
        else:
            msg = 'OTP wrong!'
    return render_template('otp.html',msg=msg,key=key)

@app.route('/atm_balance',methods=['POST','GET'])
def atm_balance():
    msg=""
    ff2=open("un.txt","r")
    uname=ff2.read()
    ff2.close()

    cursor = mydb.cursor()
    if request.method=='POST':
        amount=request.form['amount']
        cursor.execute("UPDATE admin SET amount=%s WHERE username='admin'",(amount, ))
        mydb.commit()
        return redirect(url_for('admin'))



    cursor.execute("SELECT amount FROM admin WHERE username='admin'")
    value = cursor.fetchone()[0]

    return render_template('atm_balance.html',msg=msg,value=value)

@app.route('/withdraw',methods=['POST','GET'])
def withdraw():
    uname=""
    ##if 'username' in session:
    #    uname = session['username']
    #    accno = session['accno']
    ff2=open("un.txt","r")
    uname=ff2.read()
    ff2.close()
    msg=""
    amt=0
    amt1=0
    amt2=0
    if request.method=='POST':
```

```
        amount1=request.form['amount']

        mycursor = mydb.cursor()

        mycursor.execute("SELECT amount FROM admin where username='admin'")
        amt1 = mycursor.fetchone()[0]

        mycursor.execute("SELECT deposit FROM register where card=%s",(uname, ))
        amt2 = mycursor.fetchone()[0]

        mycursor.execute("SELECT * FROM register where card=%s",(uname, ))
        ddt = mycursor.fetchone()
        name=ddt[1]
        mobile=ddt[3]

        amt=int(amount1)
        if amt<=amt1:

            if amt<=amt2:
                mycursor.execute("UPDATE admin SET amount=amount-%s WHERE
                username='admin'",(amount1, ))
                mydb.commit()
                mycursor.execute("UPDATE register SET deposit=deposit-%s
                WHERE card=%s",(amount1, uname))
                mydb.commit()

                now = datetime.datetime.now()
                rdate=now.strftime("%d-%m-%Y")
                mycursor.execute("SELECT max(id)+1 FROM event")
                maxid = mycursor.fetchone()[0]
                if maxid is None:
                    maxid=1
                sql = "INSERT INTO event(id, name, accno, amount, rdate) VALUES (%s, %s, %s, %s, %s)"
                val = (maxid, name, uname, amt, rdate)
                mycursor.execute(sql, val)
                mydb.commit()

                mess="Amount Debited Rs."+str(amt)
                url="http://iotcloud.co.in/testsms/sms.php?sms=emr&name="
                +name+"&mess="+mess+"&mobile="+str(mobile)
                webbrowser.open_new(url)

                msg="Withdraw success..."
            else:
                msg="Your Account balance is low!"
        else:
            msg="Cash is not available in ATM!!"

    return render_template('withdraw.html',msg=msg)


@app.route('/balance')
def balance():
    uname=""
    #if 'username' in session:
    #    uname = session['username']
    #    accno = session['accno']
    ff2=open("un.txt","r")
```

```
    uname=ff2.read()
    ff2.close()
    mycursor = mydb.cursor()
    mycursor.execute("SELECT * FROM register where card=%s",(uname, ))
    data = mycursor.fetchone()
    deposit=data[9]
    print(str(deposit))
    return render_template('balance.html', data=deposit)



@app.route('/user_view')
def user_view():
    mycursor = mydb.cursor()
    mycursor.execute("SELECT * FROM register")
    result = mycursor.fetchall()
    return render_template('user_view.html', result=result)

@app.route('/view_withdraw')
def view_withdraw():
    mycursor = mydb.cursor()
    mycursor.execute("SELECT * FROM event order by id desc")
    result = mycursor.fetchall()
    return render_template('view_withdraw.html', result=result)

@app.route('/logout')
def logout():
    # remove the username from the session if it is there
    #session.pop('username', None)
    return redirect(url_for('index'))

def gen(camera):

    while True:
        frame = camera.get_frame()


        yield (b'--frame\r\n'
               b'Content-Type: image/jpeg\r\n\r\n' + frame + b'\r\n\r\n')

@app.route('/video_feed')


def video_feed():
    return Response(gen(VideoCamera()),
                    mimetype='multipart/x-mixed-replace; boundary=frame')

if __name__ == "__main__":
    app.secret_key = os.urandom(12)
    app.run(debug=True,host='0.0.0.0', port=5000)
```

CHAPTER 8

CONCLUSION AND FUTURE WORK

8.1 conclusion

The project's conclusion highlights how crucial biometrics, such as fingerprints or facial recognition, are for verifying the identities of individuals using ATMs. By employing biometric technology, we can ensure that only the rightful account owners have access to their funds, thereby reducing the risk of unauthorized transactions. Additionally, the combination of biometrics with the Unknown Face Forwarder technique, a cutting-edge method for identifying unfamiliar faces, presents a strong and reliable solution for tackling fraudulent activities at ATMs. This integration enhances security measures, making it significantly harder for malicious actors to exploit loopholes and engage in unlawful transactions. Ultimately, by leveraging the power of biometrics and innovative techniques like the Unknown Face Forwarder, we can effectively safeguard ATM users' accounts and mitigate the impact of fraudulent behavior on financial institutions and their customers. The conclusion can be elaborated further as follows:

8.1.1 Addressing the Issue of Fraudulent Transactions:

The main goal of the project is to offer a concrete solution to the widespread issue of illegal transactions occurring at ATMs. To achieve this, the project harnesses the power of two key technologies: biometric authentication and the Unknown Face Forwarder technique.

Biometric authentication involves using unique physical characteristics, such as fingerprints or facial features, to verify the identity of individuals. By incorporating biometric authentication into the system, it adds an extra layer of security, ensuring that only authorized account holders can access their accounts and conduct transactions at ATMs.

Additionally, the project utilizes the Unknown Face Forwarder technique, which is an innovative method for identifying unfamiliar faces. This technique enhances the system's ability to detect and prevent unauthorized access and transactions by flagging suspicious individuals who may be attempting to use someone else's account illegally.

By combining biometric authentication with the Unknown Face Forwarder technique, the system introduces a robust defense mechanism against unauthorized activities at ATMs. This comprehensive approach significantly strengthens ATM security, providing greater protection for both financial institutions and their customers against fraudulent behavior.

8.1.2 Physical Presence as a Key Security Element:

The project underscores the significance of verifying the presence of the legitimate account holder, whether they are physically at the ATM or conducting transactions remotely. This verification is facilitated through the use of biometrics, which authenticate the identity of the individual before approving any transactions.

By implementing biometric authentication, the system ensures that transactions are only authorized when the rightful account owner is physically present at the ATM or verifies their identity from another location, such as through a mobile app or online platform. This dual authentication process adds an extra layer of security, significantly reducing the likelihood of unauthorized transactions.

In essence, the utilization of biometrics not only confirms the identity of the account holder but also verifies their physical or remote presence during transactions. This comprehensive approach enhances security measures, making it much more challenging for fraudsters to gain unauthorized access to accounts and conduct fraudulent activities. Ultimately, by requiring both biometric authentication and presence verification, the system effectively mitigates the risk of unauthorized transactions and strengthens overall security protocols for account holders.

8.1.3 Eliminating Unauthorized Access:

The conclusion of the project emphasizes how the developed solution effectively addresses the issue of unauthorized transactions occurring at ATM points without the account owner's knowledge. By integrating biometric features into the system, it ensures that only individuals authorized to access the account can do so, significantly lowering the risk of fraudulent activities.

Through the utilization of biometric authentication, the system verifies the identity of users before granting access to their accounts or approving transactions. This means that even if someone tries to use the ATM without the account owner's knowledge, they won't be able to proceed with any transactions unless their biometric data matches that of the authorized user.

This comprehensive approach not only enhances security but also instills confidence in account holders that their funds are better protected against unauthorized access and fraudulent transactions. By significantly reducing the potential for illegal activities at ATM points, the integrated biometric solution plays a pivotal role in safeguarding the financial interests and privacy of account owners.

8.1.4 Strength of Biometric Identification:

The project underscores the potency of employing biometric features for identification purposes. Biometrics, by virtue of being inherently unique to each individual, offers a formidable and dependable means of verifying the identity of the account owner. This uniqueness inherent in biometric traits, whether it's fingerprints, facial recognition, or iris scans, forms the foundation of a highly secure authentication process.

Moreover, the strength of biometric authentication is magnified when multiple biometric features are integrated into the authentication system. By combining different biometric modalities such as fingerprint and facial recognition or iris scans, the security framework becomes even more robust. This multi-factor authentication approach not only enhances the accuracy of identity verification but also significantly elevates the overall security posture of the system.

In essence, the project highlights the inherent strength of biometric authentication in confirming the identity of account owners with a high level of certainty. Furthermore, the integration of multiple biometric features adds an extra layer of resilience, making it exceedingly difficult for unauthorized individuals to bypass the security measures. Ultimately, the utilization of biometric authentication, fortified by the combination of diverse biometric traits, establishes a robust security framework that safeguards sensitive financial assets and personal information from unauthorized access or fraudulent activities.

8.1.5 Integration with Existing Security Tools:

ATM security design is a multifaceted approach that harnesses various existing security tools and information to bolster the protection of Automated Teller Machines (ATMs)

and the sensitive financial transactions they facilitate. One pivotal component of this design involves the strategic utilization of familiar security elements such as ATM cards and Personal Identification Numbers (PINs), which are already integral to the conventional ATM transaction process.

By integrating these established security features into enhanced security mechanisms, the aim is to create a robust defense system characterized by multiple layers of authentication and authorization. This integration is not merely additive but synergistic, leveraging the strengths of each security element to reinforce the overall security posture of ATM systems.

For instance, ATM cards serve as physical tokens that uniquely identify account holders and grant access to their funds. Meanwhile, PINs add an additional layer of security by requiring users to input a secret code, thereby verifying their identity and authorization to perform transactions.

By combining these authentication factors, ATM security design effectively mitigates various threats, including unauthorized access, card skimming, and identity theft. Moreover, this integrated approach enhances user confidence in the security of ATM transactions, fostering trust in the financial system.

Overall, the strategic integration of existing security tools and information into ATM security design represents a proactive response to evolving cybersecurity challenges, aiming to stay ahead of emerging threats while safeguarding the integrity of financial transactions conducted through ATMs.

8.1.6 Real-Time Involvement of the Account Owner:

A pivotal aspect of the developed solution is its real-time engagement of the bank account owner in all accessible transactions, ensuring their active involvement and awareness at every stage. This proactive approach serves to empower the account owner with the ability to promptly verify and authenticate each transaction as it occurs, thereby fostering a culture of transparency and accountability within the banking process.

By enabling account owners to monitor transactions in real-time, the solution not only addresses the immediate challenge of fraudulent activities at ATMs but also lays the groundwork for a more secure and accountable banking ecosystem. This continuous involvement of account owners not only enhances their trust in the banking system but also provides them with greater control over their financial activities, ultimately strengthening the relationship between banks and their customers.

The integration of biometric authentication adds an extra layer of security by utilizing unique physiological characteristics such as fingerprints or facial features to verify the identity of the account holder. This ensures that only authorized individuals can access and conduct transactions on the account, significantly reducing the risk of unauthorized access and fraudulent activities.

Furthermore, the incorporation of the Unknown Face Forwarder technique enhances security by detecting and preventing unauthorized attempts to gain access to the account. This technique employs sophisticated algorithms to identify suspicious behavior, such as unfamiliar faces attempting to use the ATM, and proactively intervenes to mitigate potential security threats.

In combination with existing security tools such as ATM cards and PINs, this holistic approach creates a robust security framework that fortifies ATM security and protects the interests of account holders. By leveraging multiple layers of authentication and incorporating real-time transaction monitoring, the solution establishes a more resilient defense against fraud and unauthorized access, thereby enhancing the overall security and integrity of the banking environment.

8.2 future works

The future work outlined for the proposed Real-Time Secure Biometric ATM System with Facial Recognition represents a holistic approach to advancing ATM security and biometric authentication. It encompasses a comprehensive strategic roadmap designed to foster continuous improvement and innovation within the dynamic landscape of financial security.

One key area of focus is the advancement of facial recognition technology. By staying at the forefront of developments in algorithms and hardware, the system can ensure accurate and reliable identification of individuals across various environmental conditions and scenarios. This pursuit of cutting-edge technology is crucial in maintaining the system's effectiveness in preventing unauthorized access and fraudulent activities.

Moreover, the exploration of multi-modal biometric systems presents an opportunity to enhance security further. By integrating different biometric traits such as fingerprints, iris scans, and voice recognition, the system can achieve a higher level of accuracy and robustness in user authentication. This approach not only strengthens security but also offers flexibility in accommodating different user preferences and needs.

The implementation of real-time monitoring and alert systems adds another layer of defense against potential threats. By continuously monitoring ATM transactions and user activities, the system can promptly detect and respond to any suspicious or unauthorized behavior, thereby mitigating security risks in a timely manner.

Furthermore, integrating the system with mobile banking applications expands its reach and accessibility to users. This seamless integration allows customers to conveniently access ATM services through their mobile devices while ensuring consistency and coherence across different banking channels.

Comprehensive user education initiatives are essential for promoting awareness and understanding of biometric security and safe banking practices among users. By educating customers about the importance of security measures and how to recognize and respond to potential threats, the system can empower them to actively contribute to their own financial safety.

Incorporating blockchain technology for enhanced data security adds an additional layer of protection to sensitive information. By leveraging blockchain's tamper-proof and immutable nature, the system can ensure the integrity and confidentiality of transaction records, safeguarding against data manipulation and fraud.

Similarly, the implementation of advanced biometric encryption techniques strengthens the security of biometric data transmission and storage. By employing state-of-the-art encryption methods, the system can protect user privacy and prevent unauthorized access to sensitive biometric information.

Adhering to evolving regulatory standards is paramount for maintaining compliance and trust among stakeholders. By staying abreast of regulatory developments and requirements, the system can ensure its adherence to industry standards and legal obligations, thereby enhancing its credibility and reliability.

Global promotion and adoption strategies are essential for driving widespread acceptance and usage of the biometric ATM system on a global scale. By promoting the system's benefits and capabilities to a broader audience, the system can expand its reach and impact, ultimately contributing to a more secure and resilient financial ecosystem.

Finally, continuous security audits play a critical role in identifying and addressing vulnerabilities within the system. By regularly assessing its security posture and conducting thorough audits, the system can proactively identify and mitigate potential risks, ensuring its ongoing effectiveness and resilience in the face of emerging threats.

Overall, each aspect of the future work plan contributes to the overarching goal of creating a cutting-edge and secure biometric ATM system. By embracing continuous

improvement and innovation, the system not only meets current industry standards but also anticipates and addresses emerging challenges in the dynamic field of biometric authentication and financial security.

8.2.1 Enhancement of Face Recognition Technology:

Continual improvement in face recognition technology constitutes a multifaceted endeavor aimed at refining the existing algorithms, exploring the potentials of deep learning techniques, and adapting models to effectively handle diverse facial features and varying environmental conditions. This concerted effort is geared towards achieving enhanced accuracy and speed in facial recognition processes, which are critical for the efficacy of biometric authentication systems, particularly in high-stakes scenarios such as ATM security.

Refining algorithms involves fine-tuning the mathematical frameworks that underpin facial recognition systems. This refinement process entails optimizing parameters, adjusting thresholds, and fine-tuning decision-making processes to minimize errors and maximize the system's ability to correctly identify individuals.

Exploring deep learning techniques represents a frontier in face recognition technology, leveraging artificial neural networks with multiple layers of abstraction to extract high-level features from facial images. By delving into the complexities of deep learning, researchers aim to develop more sophisticated models capable of learning intricate patterns and nuances in facial data, thereby enhancing the system's ability to accurately distinguish between individuals.

Adapting models to handle diverse facial features and varying environmental conditions is essential for ensuring the robustness and reliability of face recognition systems across different demographics and settings. This involves training algorithms on datasets that encompass a wide range of facial characteristics, including variations in skin tone, facial hair, age, and gender. Additionally, it entails developing techniques to mitigate the impact of environmental factors such as lighting conditions, occlusions, and facial expressions, which can influence the quality and consistency of facial recognition outcomes.

By addressing these facets of continual improvement, the aim is to elevate the performance of face recognition technology to new heights, bolstering its accuracy, reliability, and speed. This, in turn, enhances the overall effectiveness of biometric authentication systems, enabling them to provide secure and seamless user experiences in various real-world applications, including ATM security.

8.2.2 Biometric Fusion:

The exploration of multi-modal biometric systems represents an exciting and promising avenue in the realm of biometric authentication. By combining face recognition with other biometric modalities such as fingerprint or iris recognition, researchers and developers seek to leverage the complementary strengths of different biometric traits to enhance overall security and authentication reliability.

Face recognition, while versatile and convenient, may sometimes face challenges in accurately identifying individuals under certain conditions such as poor lighting or occlusions. In such cases, augmenting facial recognition with other biometric modalities can provide additional layers of verification, thereby increasing the robustness of the authentication process.

Fingerprint recognition, for example, offers a high degree of uniqueness and stability, as fingerprints are highly distinctive to each individual and remain relatively consistent over time. By incorporating fingerprint recognition into a multi-modal system, users can provide additional proof of identity, especially in scenarios where facial recognition alone may be insufficient or impractical.

Similarly, iris recognition boasts a high level of accuracy and reliability, as the complex patterns in the iris are virtually unique to each person and remain stable throughout one's lifetime. Integrating iris recognition with face recognition adds another dimension to the authentication process, further reducing the likelihood of false positives or unauthorized access.

Furthermore, combining multiple biometric modalities not only enhances security but also increases the difficulty for potential attackers to circumvent the system. The need to spoof or replicate multiple biometric traits simultaneously presents a formidable challenge, making multi-modal biometric systems significantly more resistant to fraudulent attempts.

Additionally, the integration of multiple biometric modalities provides redundancy, ensuring that users can still authenticate themselves even if one modality fails or is unavailable. This redundancy enhances the reliability and availability of the authentication system, contributing to a smoother user experience and minimizing disruptions in service.

the exploration of multi-modal biometric systems holds great promise for advancing the state-of-the-art in authentication technology. By combining the strengths of different

biometric traits, these systems offer heightened security, increased reliability, and improved user verification processes, making them invaluable tools in safeguarding sensitive information and securing access to critical resources.

8.2.3 Real-Time Monitoring:

Future developments in ATM security should prioritize the implementation of real-time monitoring and alert systems, leveraging the power of artificial intelligence (AI) for anomaly detection. By incorporating AI-driven technologies, such as machine learning algorithms, into the surveillance infrastructure of ATMs, financial institutions can significantly enhance security measures and promptly respond to suspicious activities or potential breaches.

Real-time monitoring and alert systems play a crucial role in proactively detecting and mitigating security threats at ATMs. Traditional surveillance methods may be limited in their ability to identify anomalies or patterns indicative of fraudulent behavior, especially in complex and dynamic environments. However, by harnessing AI capabilities, such as pattern recognition and anomaly detection, these systems can autonomously analyze vast amounts of transaction data and sensor inputs in real-time, enabling them to identify deviations from normal behavior and trigger immediate alerts.

The integration of AI into real-time monitoring systems offers several distinct advantages. Firstly, AI algorithms can adapt and learn from historical data, continually improving their ability to detect new and evolving threats. This adaptive capability enables the system to stay ahead of emerging attack techniques and effectively counteract sophisticated fraud schemes.

AI-driven monitoring systems can analyze complex patterns and correlations across multiple data sources simultaneously, providing a more comprehensive understanding of potential security risks. By considering various contextual factors, such as transaction history, geographical location, and user behavior, these systems can generate more accurate alerts and reduce the incidence of false positives.

AI-powered alert systems can facilitate rapid response and decision-making by providing actionable intelligence to security personnel. Automated alert notifications can be sent to designated individuals or security teams in real-time, enabling them to investigate and address potential threats promptly. Additionally, AI algorithms can prioritize alerts based on the severity of the detected anomalies, allowing security personnel to allocate resources efficiently and focus on the most critical incidents.

The incorporation of artificial intelligence into real-time monitoring and alert systems represents a significant step forward in enhancing ATM security. By leveraging AI-driven technologies for anomaly detection, financial institutions can strengthen their defense mechanisms against fraud, safeguard customer assets, and ensure the integrity of ATM operations. As the threat landscape continues to evolve, investing in AI-powered security solutions is essential for staying ahead of adversaries and maintaining trust in the banking ecosystem.

8.2.4 Mobile Integration:

Integrating facial recognition technology with mobile banking applications represents a promising area of development that offers numerous benefits for both users and financial institutions. By enabling users to initiate and authenticate transactions through their mobile devices using facial recognition, this integration adds a new dimension of convenience and security to the banking experience.

First and foremost, integrating facial recognition technology into mobile banking applications enhances convenience for users. Traditional authentication methods, such as passwords or PINs, can be cumbersome and prone to forgetfulness or theft. Facial recognition offers a seamless and frictionless authentication experience, allowing users to securely access their accounts with just a glance at their device's camera. This streamlined process reduces the time and effort required for users to complete transactions, enhancing overall user satisfaction and loyalty.

Moreover, integrating facial recognition with mobile banking applications adds an additional layer of security to the authentication process. Facial biometrics are highly unique to each individual and difficult to replicate, making them a robust form of authentication. By leveraging facial recognition technology, mobile banking applications can enhance security measures, mitigating the risk of unauthorized access and fraudulent transactions. This is particularly important in an era where cybersecurity threats are increasingly sophisticated and prevalent.

The integration of facial recognition technology with mobile banking applications aligns with the broader trend towards digital transformation in the banking industry. As more consumers embrace mobile banking as their primary channel for managing finances, offering advanced authentication options such as facial recognition demonstrates a commitment to innovation and customer-centricity. This differentiation can help financial institutions attract and retain tech-savvy customers who value convenience, security, and cutting-edge technology.

Integrating facial recognition with mobile banking applications opens up opportunities for additional functionalities and personalized services. For example, financial institutions can leverage facial recognition data to provide targeted offers, personalized recommendations, or tailored financial advice based on individual preferences and behavior. This not only enhances the user experience but also enables financial institutions to deepen customer engagement and drive business growth.

Integrating facial recognition technology with mobile banking applications holds significant potential to revolutionize the way users interact with their financial accounts. By combining convenience, security, and personalized services, this integration represents a win-win scenario for both users and financial institutions, driving innovation and delivering value in the digital banking landscape.

8.2.5 Education and Awareness:

Future efforts in ATM security should prioritize educational programs and awareness campaigns aimed at informing users about the security features and benefits of facial recognition technology. These initiatives play a crucial role in empowering users with knowledge about the advantages of biometric authentication and promoting responsible usage practices.

Educational programs and awareness campaigns serve as vehicles for disseminating information about the security benefits of facial recognition in ATMs. By educating users about the advantages of biometric authentication, including its accuracy, convenience, and resistance to fraud, financial institutions can instill confidence in the technology and encourage its adoption. Through targeted messaging and outreach efforts, users can gain a deeper understanding of how facial recognition enhances security measures at ATMs, thereby fostering trust and acceptance of the technology.

User education efforts extend beyond promoting the benefits of facial recognition to encompass guidelines on protecting personal biometric data. As biometric information becomes increasingly integrated into everyday transactions, it is essential for users to understand how their biometric data is collected, stored, and used. Educational programs can provide users with insights into best practices for safeguarding their biometric information, such as setting strong authentication credentials, enabling multi-factor authentication, and being vigilant against phishing attempts or social engineering attacks.

User education initiatives can help dispel misconceptions and address concerns surrounding facial recognition technology. By providing accurate information and addressing

common misconceptions about privacy, security, and data protection, financial institutions can alleviate user apprehensions and build trust in the technology. Transparent communication and open dialogue with users can foster a collaborative approach to security, empowering users to actively participate in protecting their personal information and contributing to a safer banking environment.

In addition to promoting awareness and understanding, user education efforts can also serve as a platform for soliciting feedback and input from users. By engaging with users and soliciting their perspectives on facial recognition technology, financial institutions can gain valuable insights into user preferences, concerns, and expectations. This user-centric approach to education enables financial institutions to tailor their security strategies and technology implementations to better meet the needs and preferences of their customers.

User education and awareness campaigns are essential components of an effective ATM security strategy. By empowering users with knowledge about the benefits of facial recognition technology and guidelines for protecting their personal biometric data, financial institutions can enhance security measures, foster trust and confidence in the technology, and empower users to play an active role in safeguarding their financial information.

8.2.6 Blockchain Integration:

Exploring blockchain technology holds immense promise for bolstering the security of biometric data storage and transaction records in various applications, including ATM security. Blockchain, often referred to as a decentralized and tamper-resistant ledger system, offers unique features that can significantly enhance the integrity, security, and transparency of biometric data handling.

At its core, blockchain technology operates as a distributed database or ledger that stores records of transactions across a network of computers, known as nodes. Each transaction, or data entry, is cryptographically linked to the preceding one, forming a chain of blocks that cannot be altered retroactively without consensus from the network. This decentralized nature of blockchain ensures that no single entity has control over the entire system, mitigating the risk of unauthorized manipulation or tampering.

Implementing blockchain for biometric data storage provides several key benefits. Firstly, blockchain offers unparalleled security through cryptographic hashing and consensus mechanisms. Biometric data, such as facial recognition templates or fingerprint scans,

can be securely encrypted and stored on the blockchain, ensuring that it remains tamper-proof and immutable. This protects against unauthorized access, manipulation, or deletion of sensitive biometric information, thereby safeguarding user privacy and data integrity.

Blockchain enhances transparency and auditability in biometric data handling. Every transaction recorded on the blockchain is transparent and traceable, allowing for real-time monitoring and verification of data access and usage. This transparency instills trust among stakeholders, including users, regulators, and auditors, by providing verifiable proof of data integrity and compliance with security standards.

Blockchain enables the implementation of smart contracts, self-executing contracts with predefined rules and conditions encoded into the blockchain. Smart contracts can automate and enforce data access controls, ensuring that only authorized parties can access and use biometric data according to predefined rules and permissions. This reduces the reliance on centralized intermediaries for data management and enhances the efficiency and reliability of data handling processes.

Blockchain facilitates data sharing and interoperability while maintaining data sovereignty and privacy. Users retain control over their biometric data and can selectively grant access to trusted parties, such as financial institutions or identity verification providers, through secure and auditable transactions on the blockchain. This decentralized approach to data sharing promotes collaboration and innovation while minimizing the risk of data breaches or misuse.

Exploring blockchain technology for biometric data storage and transaction records is essential for enhancing security, transparency, and privacy in ATM security and beyond. By leveraging blockchain's decentralized and tamper-resistant features, financial institutions can establish a robust foundation for secure biometric authentication systems, safeguarding user privacy and data integrity while enabling seamless and trustworthy transactions.

8.2.7 Biometric Encryption:

Researching and implementing advanced encryption techniques for biometric data during transmission and storage represents a critical step in ensuring the privacy, security, and integrity of sensitive personal information. Biometric data, such as facial recognition templates or fingerprint scans, is inherently unique and irreplaceable, making it particularly valuable and vulnerable to unauthorized access or misuse. Therefore, robust

encryption measures are essential for safeguarding this data against potential threats and maintaining user trust in biometric authentication systems.

During transmission, biometric data is vulnerable to interception or eavesdropping by malicious actors. Implementing strong encryption protocols, such as Secure Sockets Layer (SSL) or Transport Layer Security (TLS), ensures that data transmitted between devices or over networks remains confidential and protected from unauthorized interception. By encrypting biometric data in transit, financial institutions can prevent unauthorized access and maintain the confidentiality of user information, thereby preserving user privacy and trust.

Similarly, encryption plays a crucial role in securing biometric data storage. Storing biometric data in databases or servers without adequate protection exposes it to risks such as unauthorized access, data breaches, or insider threats. Advanced encryption techniques, such as symmetric or asymmetric encryption, can be employed to encrypt biometric data at rest, rendering it unreadable and indecipherable to unauthorized parties. Additionally, implementing robust access controls and encryption key management practices further enhances the security of stored biometric data, ensuring that only authorized individuals can access and decrypt the information as needed.

Research into novel encryption methods, such as homomorphic encryption or attribute-based encryption, offers potential avenues for enhancing the security and privacy of biometric data. Homomorphic encryption allows computations to be performed on encrypted data without decrypting it, enabling secure processing of sensitive information while preserving confidentiality. Attribute-based encryption enables data access controls based on specific attributes or criteria, allowing for fine-grained access management and data sharing without compromising security.

Compliance with regulatory standards, such as the General Data Protection Regulation (GDPR) or the Payment Card Industry Data Security Standard (PCI DSS), mandates the implementation of robust encryption measures for protecting sensitive personal information, including biometric data. Failure to adequately encrypt biometric data can result in severe consequences, including regulatory penalties, reputational damage, and loss of customer trust.

The research and implementation of advanced encryption techniques for biometric data transmission and storage are imperative for maintaining user trust, system integrity, and regulatory compliance in biometric authentication systems. By prioritizing the privacy and protection of biometric information through strong encryption measures, financial institutions can mitigate security risks, safeguard user privacy, and build confidence in the security of biometric authentication technologies.

8.2.8 Regulatory Compliance:

Staying informed about evolving regulations related to biometric data usage is an ongoing and multifaceted endeavor for organizations implementing biometric authentication systems. As regulatory landscapes evolve and new laws are enacted, it is essential for organizations to remain vigilant and proactive in understanding and adhering to legal requirements governing the collection, storage, and usage of biometric data.

The regulatory framework surrounding biometric data usage is complex and varies across jurisdictions. For example, regulations such as the General Data Protection Regulation (GDPR) in the European Union and the California Consumer Privacy Act (CCPA) in the United States impose strict requirements on the processing and protection of biometric information. These regulations mandate principles such as data minimization, purpose limitation, transparency, and accountability, placing significant responsibilities on organizations handling biometric data.

Ensuring that the developed biometric authentication system complies with data protection laws and standards is paramount for safeguarding user privacy and legal adherence. Compliance entails implementing robust data protection measures, such as encryption, access controls, data retention policies, and data breach notification procedures, to mitigate risks and protect biometric information from unauthorized access or misuse.

Compliance with data protection laws fosters trust and confidence among users, demonstrating an organization's commitment to respecting and safeguarding their privacy rights. By transparently communicating privacy practices and obtaining user consent for biometric data processing, organizations can build stronger relationships with customers and enhance their reputation as trustworthy stewards of sensitive personal information.

Compliance with regulatory requirements reduces the likelihood of legal liabilities, penalties, and reputational damage associated with non-compliance. Violations of data protection laws can result in severe consequences, including fines, lawsuits, regulatory sanctions, and damage to brand reputation. Therefore, organizations must prioritize compliance efforts and allocate resources to ensure that their biometric authentication systems meet legal obligations and industry standards.

Staying abreast of evolving regulations enables organizations to adapt and respond to changing legal requirements in a timely manner. This may involve conducting regular assessments of regulatory developments, engaging with legal experts or regulatory authorities, and updating policies and procedures accordingly. By proactively addressing compliance challenges, organizations can mitigate risks and maintain a competitive edge in the marketplace.

Ensuring compliance with evolving regulations related to biometric data usage is essential for organizations developing and implementing biometric authentication systems. By adhering to data protection laws and standards, organizations can protect user privacy, mitigate legal risks, and build trust with customers, thereby laying a solid foundation for the responsible and ethical use of biometric technology.

8.2.9 Global Adoption:

Promoting the global adoption of face recognition technology in ATMs requires collaboration among international banking institutions, regulatory bodies, and other stakeholders to establish standardized practices and ensure interoperability across different regions and jurisdictions. This collaborative effort is essential for fostering widespread acceptance and facilitating the seamless integration of facial recognition technology into ATM systems worldwide.

At the core of promoting global adoption is the need for international cooperation and alignment of regulatory frameworks governing biometric authentication and data protection. Given the cross-border nature of banking operations and the proliferation of digital technologies, harmonizing regulations and standards becomes imperative to address legal and compliance challenges associated with biometric data usage. This entails engaging with international regulatory bodies, such as the International Organization for Standardization (ISO) or the Financial Action Task Force (FATF), to develop uniform guidelines and best practices for the deployment of facial recognition technology in ATMs.

Standardizing practices and protocols for facial recognition technology is crucial for ensuring consistency and compatibility across different ATM systems and vendors. Establishing common technical specifications, data formats, and interoperability standards enables seamless integration and interoperability between facial recognition systems deployed by various banking institutions. This standardization also simplifies the procurement process for banks and reduces integration costs, making it more accessible for smaller financial institutions to adopt facial recognition technology in their ATM networks.

Promoting global adoption requires building consensus and trust among stakeholders regarding the security, reliability, and privacy implications of facial recognition technology. This involves engaging in dialogue with consumer advocacy groups, privacy advocates, and civil society organizations to address concerns and misconceptions surrounding biometric authentication. By fostering transparency and accountability in the

deployment of facial recognition technology, banking institutions can build confidence among consumers and gain their support for widespread adoption.

Collaboration with international banking institutions and industry associations plays a pivotal role in promoting knowledge sharing and best practices exchange. By participating in forums, conferences, and working groups dedicated to biometric authentication and ATM security, stakeholders can leverage collective expertise and insights to overcome technical, regulatory, and operational challenges associated with global adoption. This collaborative approach also facilitates capacity building and skill development among banking professionals, enabling them to effectively deploy and manage facial recognition technology in their respective regions.

Ultimately, promoting the global adoption of face recognition technology in ATMs requires a concerted effort from stakeholders across the public and private sectors. By fostering collaboration, standardization, and trust, banking institutions and regulatory bodies can pave the way for the widespread acceptance and utilization of facial recognition technology, thereby enhancing security, convenience, and accessibility in the banking industry on a global scale.

8.2.10 Continuous Security Audits:

Ongoing security audits and penetration testing are critical components of proactive cybersecurity measures, particularly in biometric authentication systems like those found in ATMs. By continuously evaluating the system's security controls, policies, and procedures, organizations can proactively identify and address potential vulnerabilities before they are exploited by malicious actors.

Regular security audits involve comprehensive assessments of the system's security posture, examining various aspects such as hardware, software, network infrastructure, and personnel practices. These audits help organizations gain insights into their overall security posture, allowing them to prioritize remediation efforts effectively and strengthen their defenses against cyber threats.

Penetration testing complements security audits by simulating real-world cyberattacks in a controlled environment. Ethical hackers or security professionals conduct these tests to identify vulnerabilities and exploit them using tactics, techniques, and procedures similar to those employed by real attackers. By uncovering weaknesses in the system's defenses, penetration testing provides valuable insights into its resilience to unauthorized access, data breaches, and other security incidents.

Together, ongoing security audits and penetration testing enable organizations to stay ahead of emerging security threats and evolving attack techniques. By proactively identifying and addressing vulnerabilities, organizations can minimize the risk of security incidents and maintain the integrity and availability of their systems and data. Regular assessments also help organizations ensure compliance with regulatory requirements and industry standards governing biometric authentication systems, demonstrating their commitment to security and accountability to stakeholders. Overall, investing in ongoing security audits and penetration testing is essential for safeguarding sensitive information, maintaining trust with customers, and mitigating the risks associated with cybersecurity challenges in today's digital landscape.

REFERENCES

[1] C. Bhuvaneswari, T. Malini, A. Giri, and S. Mahato, "Biometric and iot technology based safety transactions in atm," in *2021 7th International Conference on Advanced Computing and Communication Systems (ICACCS)*, vol. 1. IEEE, 2021, pp. 949–952.

[2] S. Ramya, R. Sheeba, P. Aravind, S. Gnanaprakasam, M. Gokul, and S. Santhish, "Face biometric authentication system for atm using deep learning," in *2022 6th International Conference on Intelligent Computing and Control Systems (ICICCS)*. IEEE, 2022, pp. 1446–1451.

[3] Z. Xie, J. Li, and H. Shi, "A face recognition method based on cnn," in *Journal of Physics: Conference Series*, vol. 1395, no. 1. IOP Publishing, 2019, p. 012006.

[4] H. Wu, K. Zhang, and G. Tian, "Simultaneous face detection and pose estimation using convolutional neural network cascade," *IEEE Access*, vol. 6, pp. 49 563–49 575, 2018.

[5] T. de Freitas Pereira and S. Marcel, "Fairness in biometrics: a figure of merit to assess biometric verification systems," *IEEE Transactions on Biometrics, Behavior, and Identity Science*, vol. 4, no. 1, pp. 19–29, 2021.

[6] J. Andrews, A. Vakil, and J. Li, "Biometric authentication and stationary detection of human subjects by deep learning of passive infrared (pir) sensor data," in *2020 IEEE Signal Processing in Medicine and Biology Symposium (SPMB)*. IEEE, 2020, pp. 1–6.

[7] X. Sun, P. Wu, and S. C. Hoi, "Face detection using deep learning: An improved faster rcnn approach," *Neurocomputing*, vol. 299, pp. 42–50, 2018.

[8] N. K. Gyamfi, M. A. Mohammed, K. Nuamah-Gyambra, F. Katsriku, and J.-D. Abdulah, "Enhancing the security features of automated teller machines (atms): A ghanaian perspective," *International Journal of Applied Science and Technology*, vol. 6, no. 1, 2016.

[9] H. R. Babaei, O. Molalapata, and A. Pandor, "Face recognition application for automatic teller machines (atm)," *ICIKM,*, vol. 45, pp. 211–216, 2012.

[10] R. Selvakumar, S. Logesh, S. Maniraj, P. Kumar *et al.*, "Face biometric authentication system for atm using deep learning," in *2022 3rd International Conference on Electronics and Sustainable Communication Systems (ICESC)*. IEEE, 2022, pp. 647–655.