

AI BASED CARDLESS ATM TRANSACTION USING FACE RECOGNIZATION

A PROJECT REPORT

Submitted by

RAJAMANICKAM.R (95071912070)

SATHISH KUMAR.P (95071912087)

SUDHAKAR.S (95071912097)

**in partial fulfilment for the award of the
degree of**

BACHELOR OF ENGINEERING

in

COMPUTER SCIENCE

FRANCIS XAVIERENGINEERINGCOLLEGE

(Autonomous)

TIRUNELVELI –627 003

APRIL 2023

FRANCIS XAVIER ENGINEERING COLLEGE
(Autonomous)
TIRUNELVELI 627 003
BONAFIDE CERTIFICATE

Certified that this project report “AI based Cardless ATM transaction using face recognition” is the bonafide work of “Rajamanickam.R (95071912070), Sathish Kumar.P (95071912087), Sudhakar.S (95071912097)” who carried out the project work under my supervision. Certified further that to the best of my knowledge the work reported herein does not form part of any others thesis or dissertation on the basis of which a degree or was conferred on an earlier occasion on this or any other candidate.

SIGNATURE

Mrs.J.PRISKILLA ANGEL

RANI M.E, PH.D.,

SUPERVISOR

ASSITANT PROFESSOR

DEPARTMENT OF COMPUTER

SCIENCE AND ENGINEERING

Francis Xavier Engineering College,

Vannarapettai, Tirunelveli.

SIGNATURE

**Dr.G.ARAVIND
SWAMINATHAN,**

M.TECH, PH.D.,

HEAD OF THE DEPARTMENT

ASSOCIATE PROFESSOR

DEPARTMENT OF COMPUTER

SCIENCE AND ENGINEERING

Francis Xavier Engineering

College, Vannarapettai, Tirunelveli.

Submitted for the B.E Degree Project Viva Voce held on

INTERNAL EXAMINER

EXTERNAL EXAMINER

ACKNOWLEDGEMENT

It gives us immense pleasure and satisfaction in presenting this report of the project undertaken during the final year of B.Tech. As it is the first step into our Professional Life, we would like to take this opportunity to express our sincere thanks to several people, without whose help and encouragement, it would have been impossible for us to carry out the desired work. We would like to express our sincere thanks to our Head of Department **Dr.G.Aravind Swaminathan** the bottom of our heart, who gave us an opportunity to undertake such a great challenging and innovative work. The gratitude is also extended to **Mrs.J.PRISKILLA ANGEL RANI** Professor, Info computer science and engineering Department for sharing her pearl of wisdom with us during the course of development of this project.

We place on record our sincere thanks to Professor **Dr. L.R Priya** for being a constant source of motivation for us. By her uncompromising demand for quality and her insistence for meeting the deadlines, we could do such an excellent work.

We would like to thank all faculty members of our college, our friends and our family members for providing their support and continuous encouragement throughout the project.

Finally, we thank our college Francis Xavier Engineering College for providing us supporting environment. We thank them for providing us such a warm atmosphere to make our Project work development experience delightful and memorable.

AI Based Card less ATM Transaction using Face Recognition

ABSTRACT

The current Automated Teller Machine (ATM) system relies on the use of ATM cards and Pin Identification Numbers (PINs) for authentication. However, this method is prone to various security threats such as theft of ATM cards, skimming, and the Lebanese loop, which compromise the security of users' financial information and funds. To address these issues, we propose a new system that utilizes face recognition technology for authentication instead of traditional ATM cards. The system combines facial recognition with the use of a PIN, providing a more secure and reliable method of accessing one's bank account. The system leverages the use of Convolutional Neural Network (CNN) model, a deep learning technique, to accurately recognize and match a user's face with their account information. This innovative solution offers a higher level of security, ensuring that users' financial information is protected from potential threats, and offers a convenient and seamless banking experience.

CHAPTER NO.	TITLE	PAGE NO.
	ACKNOWLEDGEMENT	3
	ABSTRACT	4
1.	INTRODUCTION	7
	1.1 ATM (Automatic Teller Machine)	7
	1.2 ATM HARDWARE	9
	1.3 FACE RECONGNITION	11
	1.4 Techniques for face recognition	12
	1.5 Human Identification at a distance (HID)	13
	1.6 ID Verification	14
2.	LITERATURE SURVEY	15
3.	SYSTEM IMPLEMENTATION	22

	3.1 EXISTING SYSTEM	22
	3.2 PROPOSED SYSTEM	23
	3.3 PREPROCESSING	24
4.	SIMULATION RESULTS& DISCUSSION	34
	4.1 SOFTWARE DESCRIPTION	34
	4.1.1 Why Python?	34
	4.1.2 Installation	36
	4.1.3 Python file formats	41
	4.1.4 Syntax and semantics	42
	4.1.5 Python programming examples	45
5.	CONCLUSION	47

CHAPTER – 1

INTRODUCTION

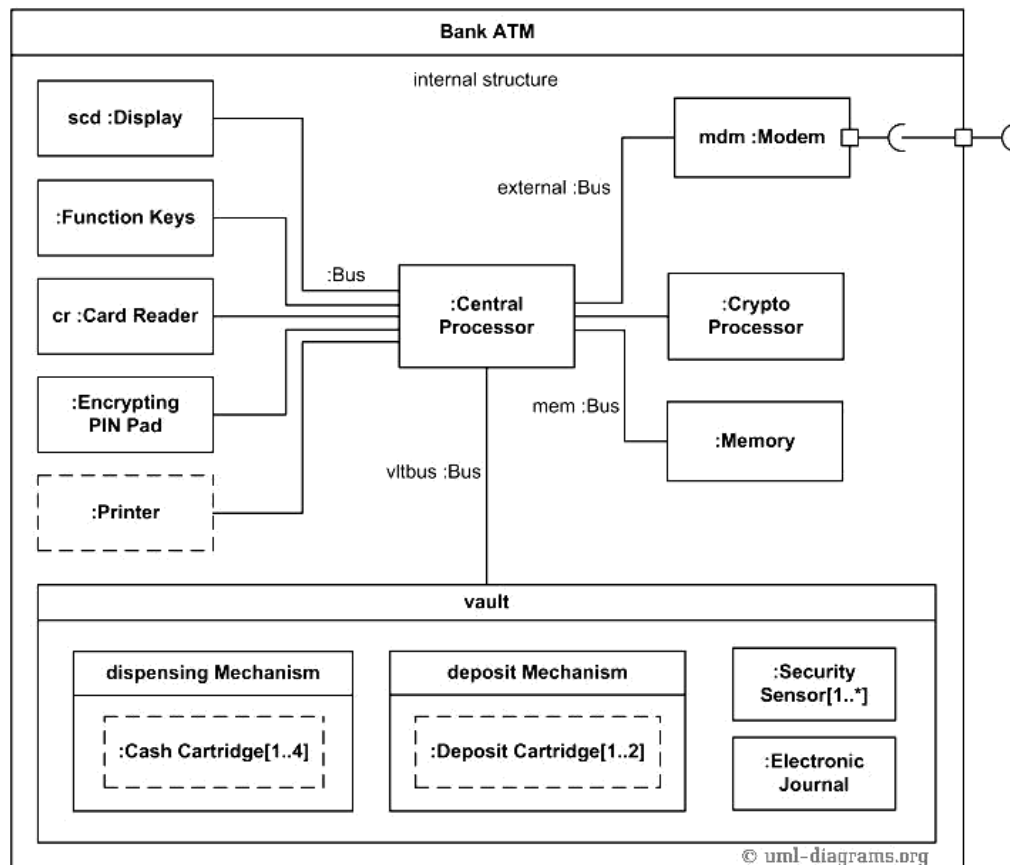
1.1 ATM (Automatic Teller Machine)

An automated teller machine (ATM) or cash machine (in British English) is an electronic telecommunications device that enables customers of financial institutions to perform financial transactions, such as cash withdrawals, deposits, funds transfers, or account information inquiries, at any time and without the need for direct interaction with bank staff. ATMs are known by a variety of names, including automatic teller machine (ATM) in the United States (sometimes redundantly as "ATM machine"). In Canada, the term automated banking machine (ABM) is also used, although ATM is also very commonly used in Canada, with many Canadian organizations using ATM over ABM. In British English, the terms cashpoint, cash machine, cashline and hole in the wall are most widely used. Other terms include any time money, cashline, tyme machine, cash dispenser, cash corner, bankomat, or bancomat. Many ATMs have a sign above them indicating the name of the bank or organisation that owns the ATM,

and possibly including the networks to which it can connect. ATMs that are not operated by a financial institution are known as "white-label" ATMs.

Using an ATM, customers can access their bank deposit or credit accounts in order to make a variety of financial transactions, most notably cash withdrawals and

balance checking, as well as transferring credit to and from mobile phones. ATMs can also be used to withdraw cash in a foreign country. If the currency being withdrawn from the ATM is different from that in which the bank account is denominated, the money will be converted at the financial institution's exchange rate. Customers are typically identified by inserting a plastic ATM card (or some other acceptable payment card) into the ATM, with authentication being by the customer entering a personal identification number (PIN), which must match the PIN stored in the chip on the card (if the card is so equipped), or in the issuing financial institution's database. According to the ATM Industry Association (ATMIA), as of 2015, there were close to 3.5 million ATMs installed worldwide. However, the use of ATMs is gradually declining with the increase in cashless payment systems



1.2 ATM HARDWARE

An ATM is typically made up of the following devices:

- CPU (to control the user interface and transaction devices)
- Magnetic or chip card reader (to identify the customer)
- a PIN pad for accepting and encrypting personal identification number EPP4 (similar in layout to a touch tone or calculator keypad), manufactured as part of a secure enclosure
- Secure cryptoprocessor, generally within a secure enclosure
- Display (used by the customer for performing the transaction)
- Function key buttons (usually close to the display) or a touchscreen (used to select the various aspects of the transaction)
- Record printer (to provide the customer with a record of the transaction)
- Vault (to store the parts of the machinery requiring restricted access)
- Housing (for aesthetics and to attach signage to)

- Sensors and indicators

Due to heavier computing demands and the falling price of personal computer-like architectures, ATMs have moved away from custom hardware architectures using microcontrollers or application-specific integrated circuits and have adopted the hardware architecture of a personal computer, such as USB connections for peripherals, Ethernet and IP communications, and use personal computer operating systems.

Business owners often lease ATMs from service providers. However, based on the economies of scale, the price of equipment has dropped to the point where many business owners are simply paying for ATMs using a credit card.

The vault of an ATM is within the footprint of the device itself and is where items of value are kept. Scrip cash dispensers do not incorporate a vault.

Mechanisms found inside the vault may include:

- Dispensing mechanism (to provide cash or other items of value)
- Deposit mechanism including a cheque processing module and bulk note acceptor (to allow the customer to make deposits)
- Security sensors (magnetic, thermal, seismic, gas)
- Locks (to ensure controlled access to the contents of the vault)

- Journaling systems; many are electronic (a sealed flash memory device based on in-house standards) or a solid-state device (an actual printer) which accrues all records of activity including access timestamps, number of notes dispensed, etc. This is considered sensitive data and is secured in similar fashion to the cash as it is a similar liability.

ATM vaults are supplied by manufacturers in several grades. Factors influencing vault grade selection include cost, weight, regulatory requirements, ATM type, operator risk avoidance practices and internal volume requirements. Industry standard vault configurations include Underwriters Laboratories UL-291 "Business Hours" and Level 1 Safes, RAL TL-30 derivatives, and CEN EN 1143-1 - CEN III and CEN IV.

ATM manufacturers recommend that a vault be attached to the floor to prevent theft, though there is a record of a theft conducted by tunnelling into an ATM floor.

1.3 FACE RECOGNITION

A facial recognition system is a technology capable of matching a human face from a digital image or a video frame against a database of faces, typically employed to authenticate users through ID verification services, works by pinpointing and measuring facial features from a given image. While initially a form of computer application, facial recognition systems have seen wider uses in recent times on smart phones and in other forms of technology, such as robotics. Because computerized facial recognition involves the measurement of a human's physiological characteristics facial recognition systems are categorised as biometrics. Although the accuracy of facial recognition systems as a biometric

technology is lower than iris recognition and fingerprint recognition, it is widely adopted due to its contactless process. Facial recognition systems have been deployed in advanced human-computer interaction, video surveillance and automatic indexing of images. They are also used widely by law enforcement agencies.

1.4 Techniques for face recognition

While humans can recognize faces without much effort, facial recognition is a challenging pattern recognition problem in computing. Facial recognition systems attempt to identify a human face, which is three-dimensional and changes in appearance with lighting and facial expression, based on its two-dimensional image. To accomplish this computational task, facial recognition systems perform four steps. First face detection is used to segment the face from the image background. In the second step the segmented face image is aligned to account for face pose, image size and photographic properties, such as illumination and grayscale. The purpose of the alignment process is to enable the accurate localization of facial features in the third step, the facial feature extraction. Features such as eyes, nose and mouth are pinpointed and measured in the image to represent the face. The so established feature vector of the face is then, in the fourth step, matched against a database of faces.

Some face recognition algorithms identify facial features by extracting landmarks, or features, from an image of the subject's face. For example, an algorithm may analyze the relative position, size, and/or shape of the eyes, nose, cheekbones, and jaw. These features are then used to search for other images with matching features.

Other algorithms normalize a gallery of face images and then compress the face data, only saving the data in the image that is useful for face recognition. A probe image is then compared with the face data. One of the earliest successful systems is based on template matching techniques applied to a set of salient facial features, providing a sort of compressed face representation.

Recognition algorithms can be divided into two main approaches: geometric, which looks at distinguishing features, or photo-metric, which is a statistical approach that distills an image into values and compares the values with templates to eliminate variances. Some classify these algorithms into two broad categories: holistic and feature-based models. The former attempts to recognize the face in its entirety while the feature-based subdivide into components such as according to features and analyze each as well as its spatial location with respect to other features.

Popular recognition algorithms include principal component analysis using eigenfaces, linear discriminant analysis, elastic bunch graph matching using the Fisherface algorithm, the hidden Markov model, the multilinear subspace learning using tensor representation, and the neuronal motivated dynamic link matching.

1.5 Human Identification at a distance (HID)

To enable human identification at a distance (HID) low-resolution images of faces are enhanced using face hallucination. In CCTV imagery faces are often very small. But because facial recognition algorithms that identify and plot facial features require high resolution images, resolution enhancement techniques have

been developed to enable facial recognition systems to work with imagery that has been captured in environments with a high signal-to-noise ratio. Face hallucination algorithms that are applied to images prior to those images being submitted to the facial recognition system utilise example-based machine learning with pixel substitution or nearest neighbour distribution indexes that may also incorporate demographic and age related facial characteristics. Use of face hallucination techniques improves the performance of high resolution facial recognition algorithms and may be used to overcome the inherent limitations of super-resolution algorithms. Face hallucination techniques are also used to pre-treat imagery where faces are disguised. Here the disguise, such as sunglasses, is removed and the face hallucination algorithm is applied to the image. Such face hallucination algorithms need to be trained on similar face images with and without disguise. To fill in the area uncovered by removing the disguise, face hallucination algorithms need to correctly map the entire state of the face, which may be not possible due to the momentary facial expression captured in the low resolution image.

1.6 ID Verification

The emerging use of facial recognition is in the use of ID verification services. Many companies and others are working in the market now to provide these services to banks, ICOs, and other e-businesses. Face recognition has been leveraged as a form of biometric authentication for various computing platforms and devices; while Microsoft introduced face recognition login to its Xbox 360 video game console through its Kinect accessory, as well as Windows 10 via its "Windows Hello" platform (which requires an infrared-illuminated camera). In

2017 Apple's iPhone X smartphone introduced facial recognition to the product line with its "Face ID" platform, which uses an infrared illumination system.

Face ID

Apple introduced Face ID on the flagship iPhone X as a biometric authentication successor to the Touch ID, a fingerprint based system. Face ID has a facial recognition sensor that consists of two parts: a "Romeo" module that projects more than 30,000 infrared dots onto the user's face, and a "Juliet" module that reads the pattern. The pattern is sent to a local "Secure Enclave" in the device's central processing unit (CPU) to confirm a match with the phone owner's face. The facial pattern is not accessible by Apple. The system will not work with eyes closed, in an effort to prevent unauthorized access. The technology learns from changes in a user's appearance, and therefore works with hats, scarves, glasses, and many sunglasses, beard and makeup. It also works in the dark. This is done by using a "Flood Illuminator", which is a dedicated infrared flash that throws out invisible infrared light onto the user's face to properly read the 30,000 facial points.

CHAPTER – II

LITERATURE SURVEY

1. Title: Detection of single-trial EEG of the neural correlates of familiar faces recognition using machine learning algorithms

Author name: Abdulmajeed, Alsufyani

Year: 2019

We analyze Electroencephalograph EEG data with several classification algorithms to classify probe and irrelevant data. Out of eight algorithms, five found to perform poorly: Decision Tree, Random Forest, Neural Network, SVM RBF and Adaboost, while KNN, SVM Linear and Naive Bayes Gaussian yielded satisfactorily. Analysis is carried with 14 different subjects. Various metrics like accuracy, precision and

recall are calculated to establish best performing algorithms with Electroencephalogram EEG data. Further work is needed on this area by increasing the number of subjects and experiments, with an idea to eliminate intersubjective variability. Also, work on algorithms tuning for better mental states capturing.

2. Title: Facial Verification Technology for Use in ATM Transactions

Author name: Aru, philO.Ezeand I.Gozie

Year: 2013

There is an urgent need for improving security in banking region. With the birth of the Automatic Teller Machines, banking became a lot easier though with its own troubles of insecurity. Due to tremendous increase in the number of criminals and their activities, the ATM has become insecure. ATM systems today use no more

than an access card and PIN for identity verification. The recent progress in biometric identification techniques, including finger printing, retina scanning, and facial recognition has made a great efforts to rescue the unsafe situation at the ATM. This research looked into the development of a system that integrates facial recognition technology into the identity verification process used in ATMs. An ATM model that is more reliable in providing security by using facial recognition software is proposed .The development of such a system would serve to protect consumers and financial institutions alike from intruders and identity thieves. This paper proposes an automatic teller machine security model that would combine a physical access card, a PIN, and electronic facial recognition that will go as far as withholding the fraudster's card. If this technology becomes widely used, faces would be protected as well as PINs.

3) Title: Face Recognition Application for Automatic Teller Machines (ATM)

Author name: H.R.Babaei, O.Molalapata

Year: 2012

In this article about biometric systems the general idea is to use facial recognition to reinforce security on one of the oldest and most secure piece of technology that is still in use to date thus an Automatic Teller Machine. The main use for any biometric system is to authenticate an input by Identifying and verifying it in an existing database. Security in ATM's has changed little since their introduction in the late 70's. This puts them in a very vulnerable state as technology has brought in a new breed of thieves who use the advancement of technology to their advantage. With this in mind it is high time something should be done about the security of this technology beside there cannot be too much security when it comes to people's money.

4) Title- Short Term Face Recognition for Automatic Teller Machine (ATM)

Author name - E.Derman, Y.K.Gecici

Year: 2013

Automatic Teller Machines (ATMs) are widely used in our daily lives due to their convenience, wide-spread availability and time-independent operation. Automatic retraction of forgotten card or cash by ATMs is a problem with serious consequences (lost time and money), typically caused by user inattention/negligence. In this work, we propose a scheme in which the retraction rate of an ATM is decreased using face detection and recognition methods via ATM's built-in camera. The short time frame of ATM usage and severe motion artifacts make this problem very different from an ordinary face authentication or face recognition problem. We evaluate the proposed system under challenging conditions of real ATM usage. The experimental results on multiple databases reveal that our proposed system is promising for mitigating card/cash forgetting issue and improving ATM user experience.

5) Title: Multiple Level Information Security Using Image Steganography and Authentication.

Author name: Marilou O. Espina¹

Year: 2019

The security of information during transmission in the open network is crucial. While sharing digital data on the internet, it is essential to observe information security goals: confidentiality, integrity, authenticity, and accuracy. This paper presents a multiple tier information security through image steganography technique using a novel puzzle in YCbCr color space, and digitally signed stego

image for authentication. Experimental results show that the resemblance between the stego-images and cover images is high. The recognition of the stego image is small, with the use of the Human Visual System (HVS) having an average PSNR value of 47.72db. The probability of detection of the stego-images using ChiSquare Analysis is low, with the average value of 4.39E05.

6) Title: Recognition Application for Automatic Teller Machines (ATM)

Author name: H.R.Babaei, O.Molalapata and A.A.Pandor

Year: 2012

In this project about biometric systems the general idea is to use facial recognition to reinforce security on one of the oldest and most secure piece of technology that is still in use to date thus an Automatic Teller Machine. The main use for any biometric system is to authenticate an input by Identifying and verifying it in an existing database. Security in ATM's has changed little since their introduction in the late 70's. This puts them in a very vulnerable state as technology has brought in a new breed of thieves who use the advancement of technology to their advantage. With this in mind it is high time something should be done about the security of this technology beside there cannot be too much security when it comes to people's money.

7) Title: SEPIA: Secure-PIN-authentication-as-a-service for ATM using Mobile and wearable devices.

Author name: JinfangXu, Khan, Rasib and RasibHasan

Year: 2015

Credit card fraud is a common problem in today's world. Financial institutions have registered major losses till today due to users being exposed of their credit card information. Shoulder-surfing or observation attacks, including card skimming and video recording with hidden cameras while users perform PIN-based authentication at ATM terminals is one of the common threats for common users. Researchers have struggled to come up with secure solutions for secure PIN authentication. However, modern day ubiquitous wearable devices, such as the Google Glass have presented us with newer opportunities in this research area. In this paper, we propose Secure-PIN-Authentication-as-a-Service (SEPIA), a secure obfuscated PIN authentication protocol for ATM and other point-of-service terminals using cloud-connected personal mobile and wearable devices. Our approach protects the user from shoulder-surfers and partial observation attacks, and is also resistant to relay, replay, and intermediate transaction attacks. A SEPIA user utilizes a Google Glass or a mobile device for scanning a QR code on the terminal screen to prove co-location to the cloud-based server and obtain a secure PIN template for point-of-service authentication. SEPIA ensures minimal task overhead on the user's device with maximal computation offloaded to the cloud. We have implemented a proof-of-concept prototype to perform experimental analysis and a usability study for the SEPIA architecture.

8) Title: Overview of Techniques for Face Recognition,

Author name: M.Murugesan,S.Thilagamani,

Year: 2019

Individuals perform face acknowledgment naturally consistently and for all intents and purposes with no exertion. In spite of the fact that it sounds like a straightforward undertaking for us, it has turned out to be a mind boggling task for

a PC, as it has numerous factors that can impede the exactness of the strategies, for instance: enlightenment variety, low goals, impediment, among other. In software engineering, face acknowledgment is essentially the assignment of perceiving an individual dependent on its facial picture. It has turned out to be extremely prevalent over the most recent two decades, for the most part in view of the new techniques created and the high caliber of the present recordings/cameras. There is different face recognition algorithm which is used for various uses in their identification. In this paper, there is an overview of some face recognition techniques for different kind of usage.

9) Title: Pedestrian Re-Identification Monitoring System Based on Deep Convolutional Neural Network

Author name: Wenzheng Qu, Zhiming Xu, Bei Luo, Haihua Feng, Zhiping Wan

Year: 2019

The gradual establishment of large-scale distributed camera networks and the rapid development of “Internet +” have resulted in the recent popularization of massive video surveillance systems. As pedestrians are the key monitoring targets in video surveillance systems, many studies are focusing on pedestrian re-identification monitoring algorithms across cameras. At present, the pedestrian re-identification model is not only faced with the difficulty of training the network model due to the huge quantity difference between different types of training samples, but also needs to reduce the impact of the large difference in visual performance on the model identification accuracy. To solve these difficulties, this paper proposed a deep learning model and designed a system based on a deep convolutional neural network for pedestrian re-identification. In particular, we determined the difference

between the system input neighborhoods in order to derive the local relationship between the two input images, thus reducing the effects of illumination and perspective. The proposed method was implemented in our developed end-to-end monitoring system for pedestrian re-identification. The hardware component of the system design framework was composed of a digital matrix, streaming media storage server and a network high-speed dome, with the ability to extend to additional tasks in the future. Our approach reduces the effects of data imbalances and visual performance differences, with a score of 76.0% for rank-1 and 99.5% for rank-20 on large data sets (CUHK03).

10) Title: ATM- Security using machine learning technique in IoT

Author name Udhaya Kumar N., Sri Vasu R., Subash S, Sharmila Rani D

Year: 2019

The idea of designing and implementation of the real-time ATM security project came with the incidents of accessing the ATM by the unauthorized users instead of the authorized user. This project will give access to the user only after identifying the image of the user taken by the CCTV in the ATM and compare the identified image with the image of the user that was stored in the database created during the account creation which comes under the banking session of banks. In some cases the authorized user is not able to use the ATM for some emergency purposes, in such cases, the OTP is sent to the users registered mobile number and the person who came instead of the authorized user have to enter the OTP that the authorized user received. This method will reduce the risk in ATM usage by the common people. The face detection and face recognition are done using deep learning techniques and machine learning. The IOT components like Camera, RFID reader, Tag, Relay, Motor were used.

CHAPTER-III

SYSTEM IMPLEMENTATION

3.1 EXISTING SYSTEM

The ATM using Face Recognition System is indicate the way to a lot of forgery attempt and abuse through card theft and pin theft of customer account details. In this system they are used many components like Face Detector, Face Recognizer, 2-D, 3-D Technique and Surface Texture Analysis. In the existing System they use some of the machine learning techniques to predict the facial expression but the output accuracy is very low it may be $< 70\%$.

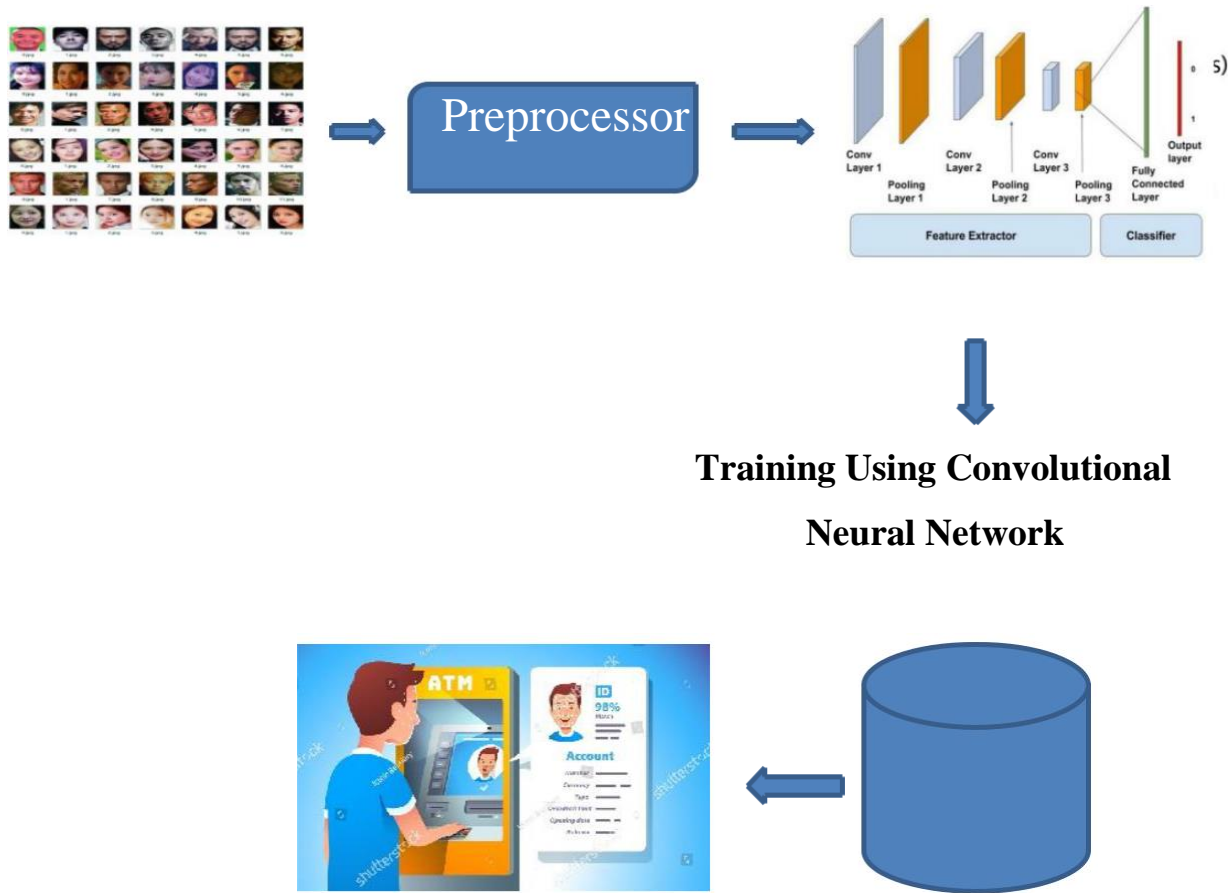
3.2 PROPOSED SYSTEM

The proposed system we use Convolutional Neural Network to predict the face recognition.

Its consists of the following steps:

- i. the face in image is detected and cropped,
- ii. the cropped image is pre-processed in order to provide further illumination invariant,
- iii. the convolutional neural network is applied to predicted features.

PROPOSED SYSTEM ARCHITECTURE



Testing using the trained model and move forward for ATM Transaction

Fig 3.1 Proposed Block Diagram

3.3 PREPROCESSING

A pre-processing or filtering step is applied to minimize the degradation related to the noise. There has been a lot of work in structuring the efficient noise suppression filters. The noise such as the shadow in the input images are removed using the pre-processing filters such as average filter. This stage is necessary to enhance the lungs image quality and made the feature extraction component more reliable for the improvement of broad and narrow input image.

DEEP LEARNING

Deep learning (also known as deep structured learning) is part of a broader family of machine learning methods based on artificial neural networks with representation learning. Learning can be supervised, semi-supervised or unsupervised.

Deep learning architectures such as deep neural networks, deep belief networks, recurrent neural networks and convolution neural networks have been applied to fields including computer vision, machine vision, speech recognition, natural language processing, audio recognition, social network filtering, machine translation, bioinformatics, drug design, medical image analysis, material inspection and board game programs, where they have produced results comparable to and in some cases surpassing human expert performance.

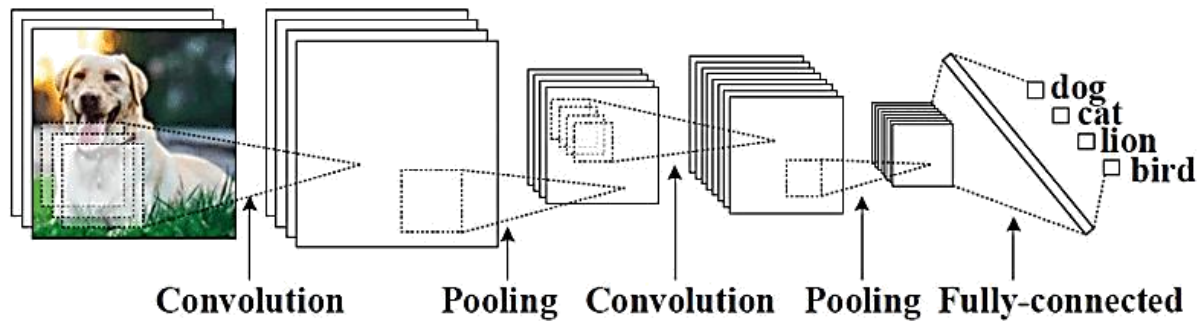
The adjective "deep" in deep learning comes from the use of multiple layers in the network. Early work showed that a linear perceptron cannot be a universal classifier, and then that a network with a nonpolynomial activation function with one hidden

layer of unbounded width can on the other hand so be. Deep learning is a modern variation which is concerned with an unbounded number of layers of bounded size, which permits practical application and optimized implementation, while retaining theoretical universality under mild conditions. In deep learning the layers are also permitted to be heterogeneous and to deviate widely from biologically informed connectionist models, for the sake of efficiency, trainability and understandability, whence the "structured" part.

CONVOLUTIONAL NEURAL NETWORK

In deep learning, a convolutional neural network (CNN, or ConvNet) is a class of deep neural networks, most commonly applied to analyzing visual imagery.[1] They are also known as shift invariant or space invariant artificial neural networks (SIANN), based on their shared-weights architecture and translation invariance characteristics.[2][3] They have applications in image and video recognition, recommender systems,[4] image classification, medical image analysis, natural language processing,[5] and financial time series.[6]

CNNs are regularized versions of multilayer perceptrons. Multilayer perceptrons usually mean fully connected networks, that is, each neuron in one layer is connected to all neurons in the next layer. The "fully-connectedness" of these networks makes them prone to overfitting data. Typical ways of regularization include adding some form of magnitude measurement of weights to the loss function. CNNs take a different approach towards regularization: they take advantage of the hierarchical pattern in data and assemble more complex patterns using smaller and simpler patterns. Therefore, on the scale of connectedness and complexity, CNNs are on the lower extreme.



Architecture of Convolutional neural network

Steps in CNN

- **Step 1: Convolution**
- **Step 2: Max pooling**
- **Step 3: Flattening**
- **Step 4: Fully connection**

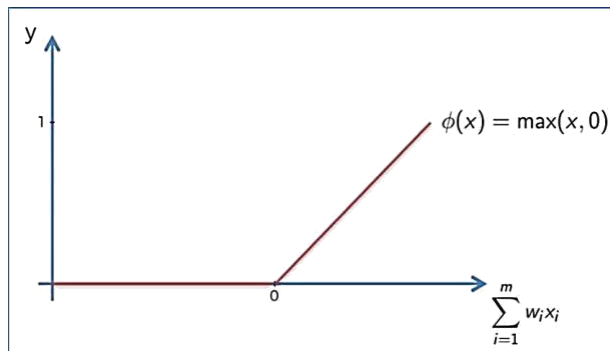
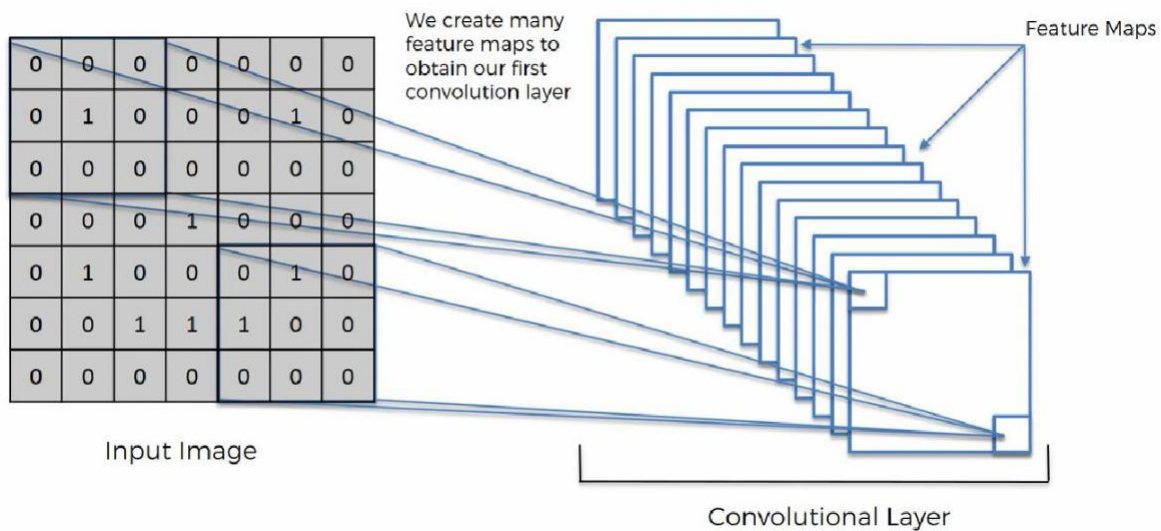
Convolution Layer

When programming a CNN, the input is a tensor with shape (number of images) x (image height) x (image width) x (image depth). Then after passing through a convolutional layer, the image becomes abstracted to a feature map, with

shape (number of images) x (feature map height) x (feature map width) x (feature map channels). A convolutional layer within a neural network should have the following attributes:

- Convolutional kernels defined by a width and height (hyper-parameters).
- The number of input channels and output channels (hyper-parameter).
- The depth of the Convolution filter (the input channels) must be equal to the number channels (depth) of the input feature map.

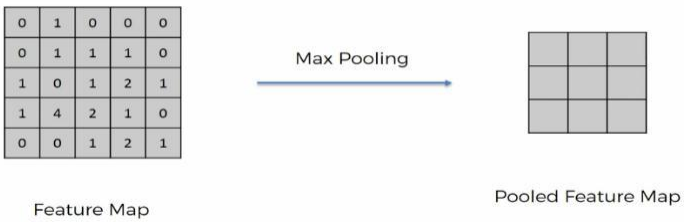
Convolutional layers convolve the input and pass its result to the next layer. This is similar to the response of a neuron in the visual cortex to a specific stimulus.[12] Each convolutional neuron processes data only for its receptive field. Although fully connected feedforward neural networks can be used to learn features as well as classify data, it is not practical to apply this architecture to images. A very high number of neurons would be necessary, even in a shallow (opposite of deep) architecture, due to the very large input sizes associated with images, where each pixel is a relevant variable. For instance, a fully connected layer for a (small) image of size 100 x 100 has 10,000 weights for each neuron in the second layer. The convolution operation brings a solution to this problem as it reduces the number of free parameters, allowing the network to be deeper with fewer parameters.[13] For instance, regardless of image size, tiling regions of size 5 x 5, each with the same shared weights, requires only 25 learnable parameters. By using regularized weights over fewer parameters, the vanishing gradient and exploding gradient problems seen during backpropagation in traditional neural networks are avoided.[14][15]



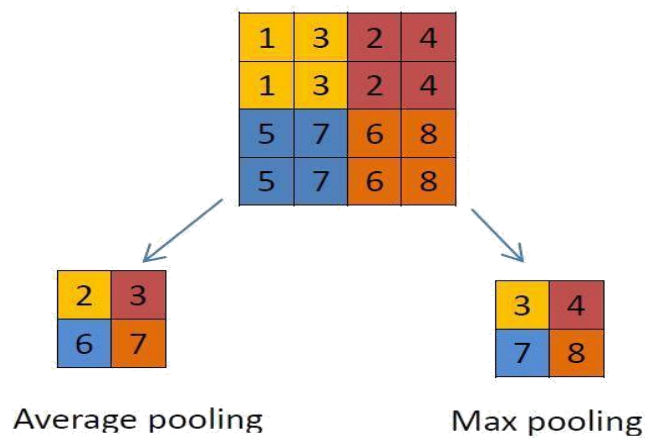
Applying ReLu Activation function to decrease the linearity in the image, because the image originally nonlinear

Pooling Layer

A **pooling** layer is another building block of a **CNN**. Its function is to progressively reduce the spatial size of the representation to reduce the amount of parameters and computation in the network. **Pooling** layer operates on each feature map independently. The most common approach used in **pooling** is **max pooling**.



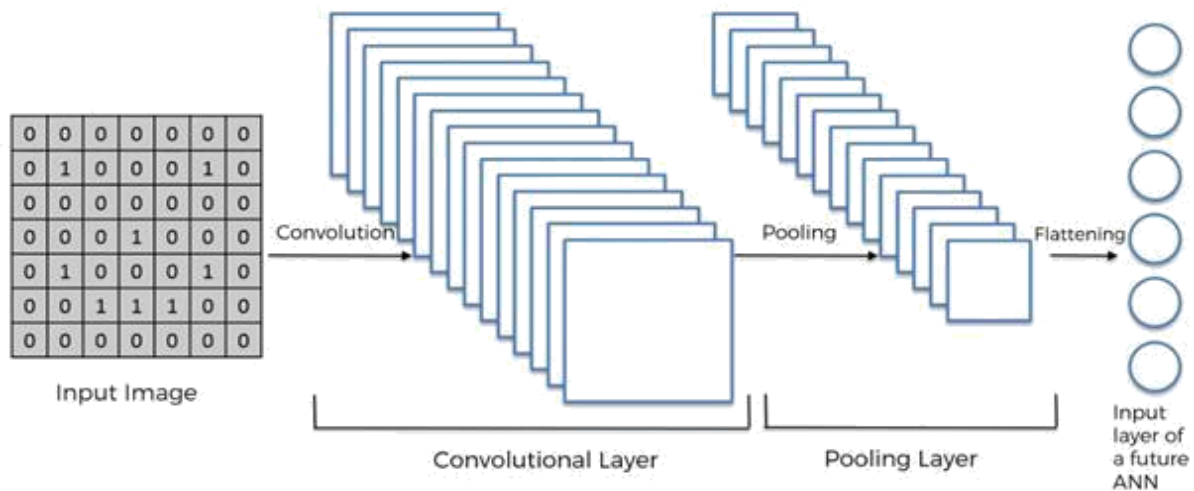
Max / Avg. Pooling



Flattening

Flattening is converting the data into a 1-dimensional array for inputting it to the next layer. We **flatten** the output of the convolutional layers to create a single long feature vector. And it is connected to the final classification model, which is called a fully-connected layer





Fulling Connection

Fully connected layers connect every neuron in one layer to every neuron in another layer. It is in principle the same as the traditional multi-layer perceptron neural network (MLP). The flattened matrix goes through a fully connected layer to classify the images.

CHAPTER IV

SIMULATION RESULTS& DISCUSSION

4.1 SOFTWARE DESCRIPTION

The Python language had a humble beginning in the late 1980s when a Dutchman Guido Von Rossum started working on a fun project, which would be a successor to ABC language with better exception handling and capability to interface with OS Amoeba at Centrum Wiskunde and Informatica. It first appeared in 1991. Python 2.0 was released in the year 2000 and Python 3.0 was released in

the year 2008. The language was named Python after the famous British television comedy show Monty Python's Flying Circus, which was one of Guido's favorite television programmes. Here we will see why Python has suddenly influenced our lives and the various applications that use Python and its implementations.

In this chapter, you will be learning the basic installation steps that are required to perform on different platforms (that is Windows, Linux, and Mac), about environment variables, setting up of environment variables, file formats, Python interactive shell, basic syntaxes and finally printing out formatted output.

4.1.1 Why Python?

Now you might be suddenly bogged with the question, why Python? According to Institute of Electrical and Electronics Engineers (IEEE) 2016 ranking Python ranked third after C and Java. As per Indeed.com's data of 2016, the Python job market search ranked fifth. Clearly, all the data points to the ever rising demand in the job market for Python. It's a cool language if you want to learn just for fun or if you want to build your career around Python, you will adore the language. At school level, many schools have started including Python programming for kids.

With new technologies taking the market by surprise Python has been playing a dominant role. Whether it is cloud platform, mobile app development, Big Data, IoT with Raspberry Pi, or the new Block chain technology, Python is being seen as a niche language platform to develop and deliver a scalable and robust applications. Some key features of the language are:

- Python programs can run on any platform, you can carry code created in Windows machine and run it on Mac or Linux
- Python has inbuilt large library with prebuilt and portable functionality, also known as the standard library
- Python is an expressive language
- Python is free and open source
- Python code is about one third of the size of equivalent C++ and Java code
- Python can be both dynamically and strongly typed--dynamically typed means it is a type of variable that is interpreted at runtime, which means, in Python, there is no need to define the type (int or float) of the variable

Python applications

One of the most famous platforms where Python is extensively used is YouTube. The other places where you will find Python being extensively used are the special

effects in Hollywood movies, drug evolution and discovery, traffic control systems, ERP systems, cloud hosting, e-commerce platform, CRM systems, and whatever field you can think of.

Versions

At the time of writing this book, two main versions of the Python programming language were available in the market, which are Python 2.x and Python 3.x. The stable release as of writing the book were Python 2.7.13 and Python 3.6.0.

Implementations of Python

Major implementations include CPython, Jython, IronPython, MicroPython, and PyPy.

4.1.2 Installation

Here we will look forward to the installation of Python on three different OS platforms, namely, Windows, Linux, and Mac OS. Let's begin with the Windows platform.

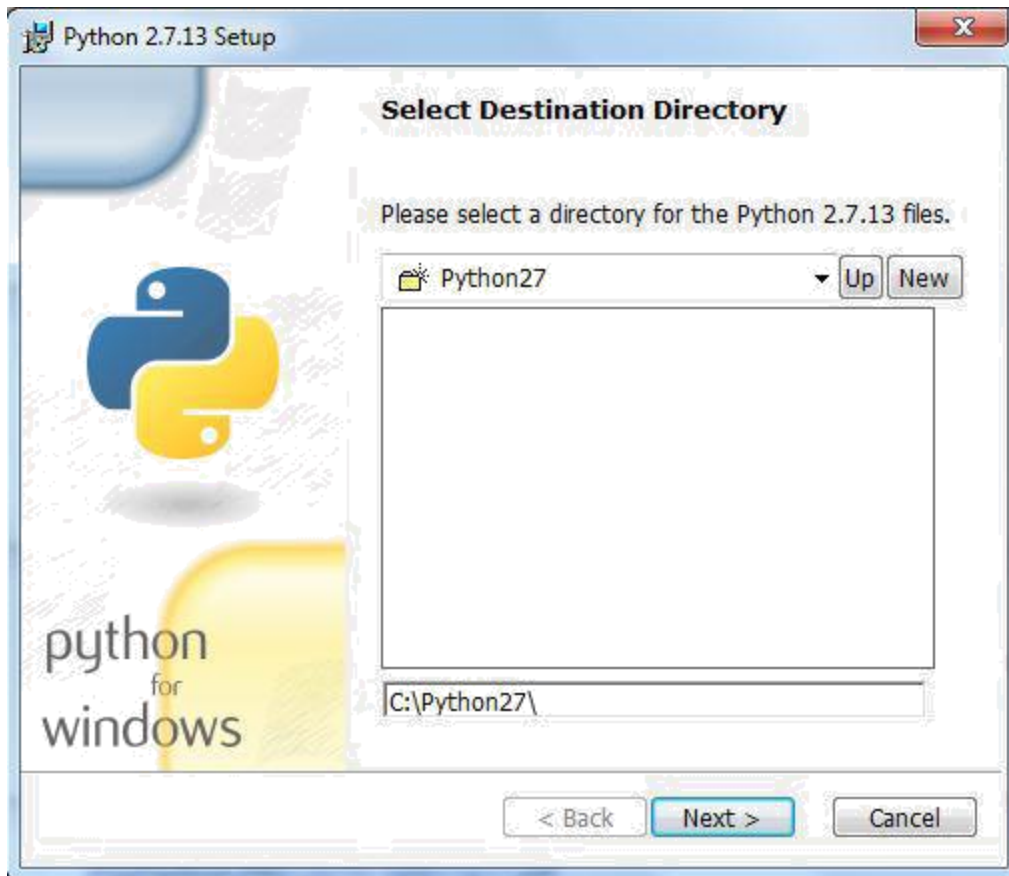
Installation on Windows platform

Python 2.x can be downloaded from <https://www.python.org/download/s>. The installer is simple and easy to install. Perform the following steps to install the setup:

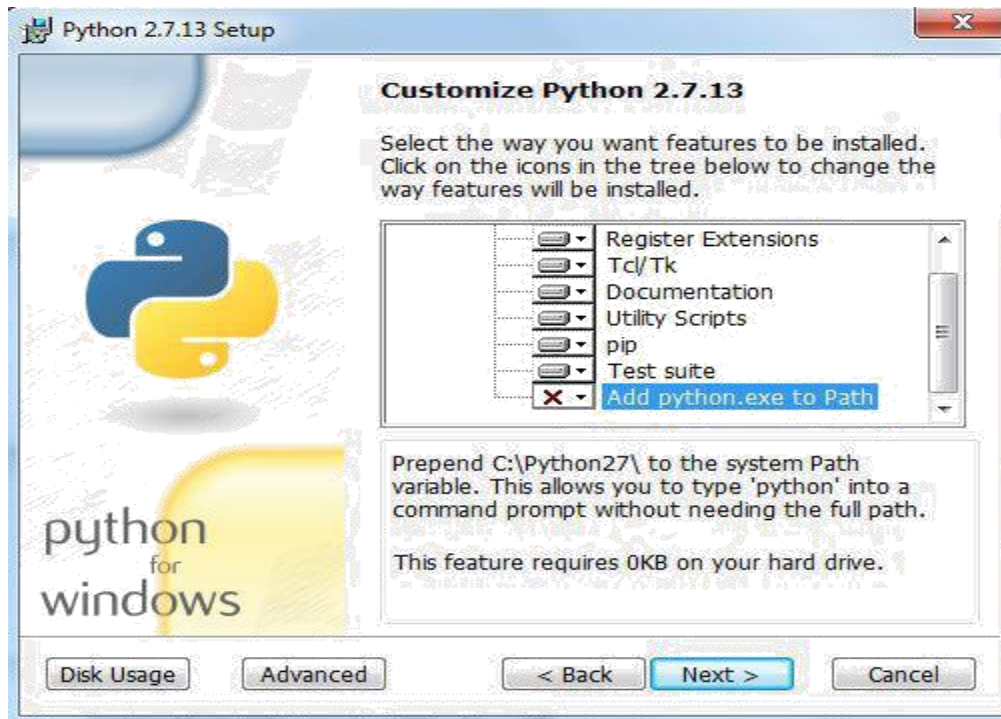
1. Once you click on setup installer, you will get a small window on your desktop screen as shown here; click on **Next**:



2. Provide a suitable installation folder to install Python. If you don't provide the installation folder, then the installer will automatically create an installation folder for you, as shown in the following screenshot. Click on **Next**:



After completion of step 2, you will get a window to customize Python as shown in the preceding screenshot. Notice that the **Add python.exe to Path** option has been marked **x**. Select this option to add it to system path variable (which will be explained later in the chapter), and click on **Next**:




4. Finally, click on **Finish** to complete the installation:



Installation on Linux platform

These days most of the Linux-based systems come preloaded with Python, so in most cases, you do not need to install it separately. However, if you do not find your desired version of Python on the Linux platform, you can download your desired version for a particular Linux platform from the site <https://www.python.org/downloads/source/>. Perform the following steps:

1. Extract the compressed file using the `tar -xvzfpython_versionx.x` command.
2. Browse the directory of the compressed file as shown in the screenshot:



The screenshot shows a terminal window titled "root@localhost:~/Python-2.7.12". The terminal output shows the following commands and results:

```
[root@localhost ~]# cd Python-2.7.12
[root@localhost Python-2.7.12]# ls
aclocal.m4  Demo      Lib        Modules  pyconfig.h.in  Tools
config.guess  Doc      LICENSE    Objects  Python
config.sub   Grammar  Mac        Parser   README
configure    Include  Makefile.pre.in  PC       RISCOS
configure.ac install-sh Misc       PCbuild  setup.py
```

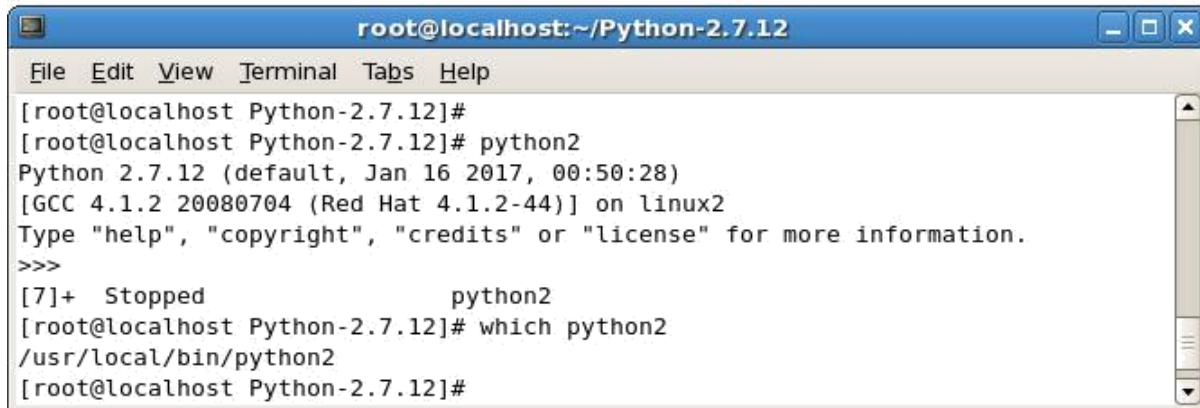
3. Run the following commands:

```
[root@localhost Python-2.7.12]# ./configure
```

```
[root@localhost Python-2.7.12]# make
```

```
[root@localhost Python-2.7.12]# make install
```

4. Use the command as shown in screenshot to ensure that Python is running:



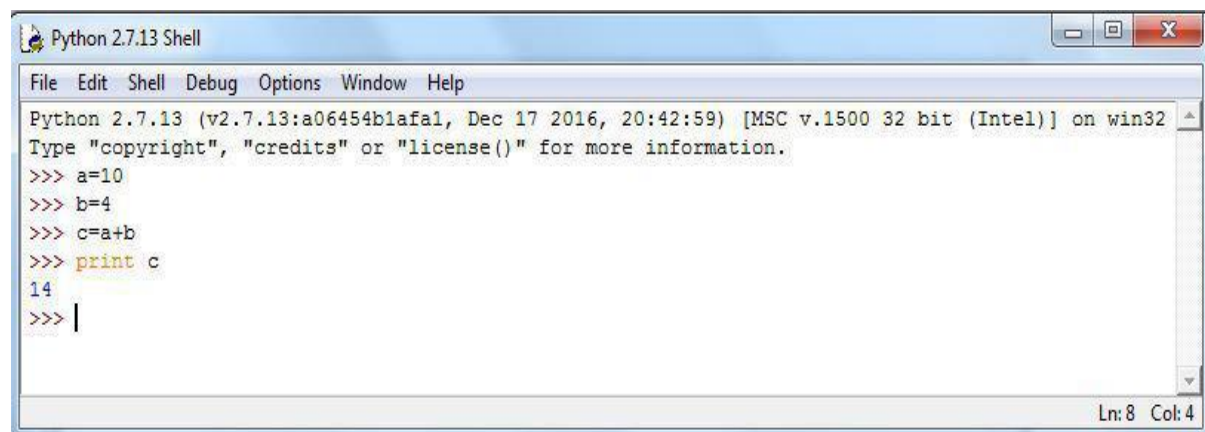
```
root@localhost:~/Python-2.7.12
File Edit View Terminal Tabs Help
[root@localhost Python-2.7.12]#
[root@localhost Python-2.7.12]# python2
Python 2.7.12 (default, Jan 16 2017, 00:50:28)
[GCC 4.1.2 20080704 (Red Hat 4.1.2-44)] on linux2
Type "help", "copyright", "credits" or "license" for more information.
>>>
[7]+ Stopped python2
[root@localhost Python-2.7.12]# which python2
/usr/local/bin/python2
[root@localhost Python-2.7.12]#
```

4.1.3 Python file formats

Every language understands a file format, for example, like the C language file extension is .c likewise java language has a file extension .java. The Python file extension is .py while bytecode file extension is .pyc.

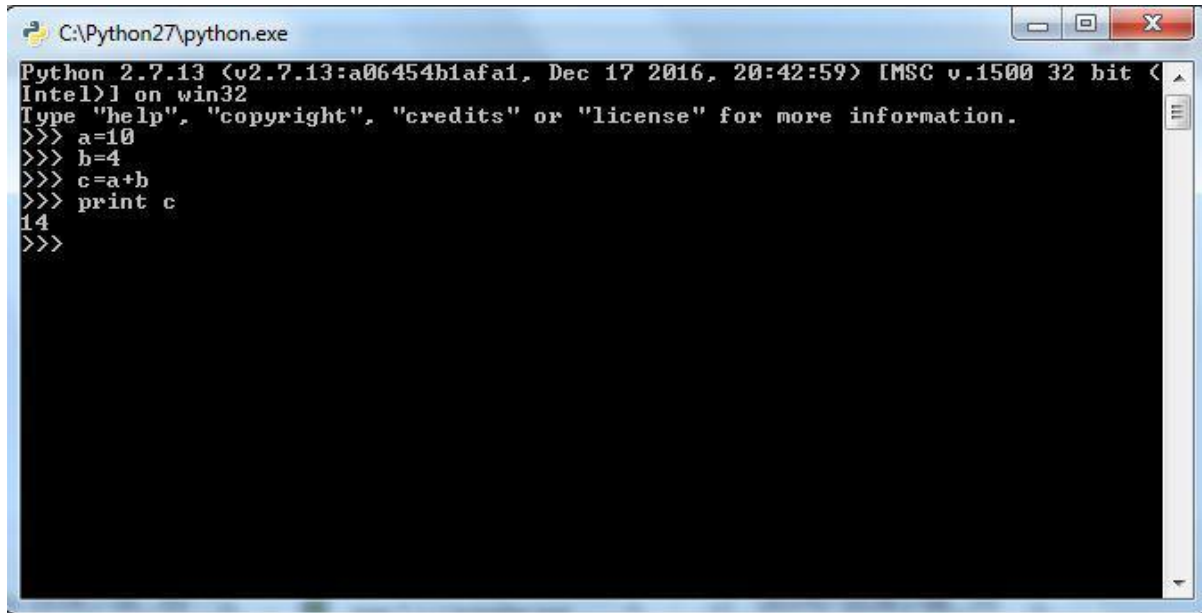
Python interactive shell

Python interactive shell is also known as **Integrated Development Environment (IDLE)**. With the Python installer, two interactive shells are provided: one is IDLE (Python GUI) and the other is Python (command line). Both can be used for running simple programs. For complex programs and executing large files, the windows command prompt is used, where after the system variables are set automatically, large files are recognized and executed by the system.



```
Python 2.7.13 Shell
File Edit Shell Debug Options Window Help
Python 2.7.13 (v2.7.13:a06454b1afaf1, Dec 17 2016, 20:42:59) [MSC v.1500 32 bit (Intel)] on win32
Type "copyright", "credits" or "license()" for more information.
>>> a=10
>>> b=4
>>> c=a+b
>>> print c
14
>>> |
```

The preceding screenshot is what we call Python IDLE, which comes bundled with the Python installation. The next screenshot is of the command line that also comes bundled with the Python installation, or we can simply launch the Python command through the windows command line and get Python command line. For most of our programming instructions, we will be using the Python command line:



```
C:\Python27\python.exe
Python 2.7.13 (v2.7.13:a06454b1afa1, Dec 17 2016, 20:42:59) [MSC v.1500 32 bit <
Intel)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> a=10
>>> b=4
>>> c=a+b
>>> print c
14
>>>
```

4.1.4 Syntax and semantics

Python is meant to be an easily readable language. Its formatting is visually uncluttered, and it often uses English keywords where other languages use punctuation. Unlike many other languages, it does not use curly brackets to delimit blocks, and semicolons after statements are optional. It has fewer syntactic exceptions and special cases than C or Pascal.

Indentation

Python uses whitespace indentation, rather than curly brackets or keywords, to delimit blocks. An increase in indentation comes after certain statements; a decrease in indentation signifies the end of the current block. Thus, the program's visual structure accurately represents the program's semantic structure. This feature is sometimes termed the off-side rule, which some other languages share, but in most languages indentation doesn't have any semantic meaning.

Statements and control flow

Python's statements include (among others):

- The **assignment** statement (token '=', the equals sign). This operates differently than in traditional imperative programming languages, and this

fundamental mechanism (including the nature of Python's version of variables) illuminates many other features of the language. Assignment in C, e.g., `x = 2`, translates to "typed variable name `x` receives a copy of numeric value 2". The (right-hand) value is copied into an allocated storage location for which the (left-hand) variable name is the symbolic address. The memory allocated to the variable is large enough (potentially quite large) for the declared type. In the simplest case of Python assignment, using the same example, `x = 2`, translates to "(generic) name `x` receives a reference to a separate, dynamically allocated object of numeric (int) type of value 2." This is termed binding the name to the object. Since the name's storage location doesn't contain the indicated value, it is improper to call it a variable. Names

may be subsequently rebound at any time to objects of greatly varying types, including strings, procedures, complex objects with data and methods, etc. Successive assignments of a common value to multiple names, e.g., `x = 2; y = 2; z = 2` result in allocating storage to (at most) three names and one numeric object, to which all three names are bound. Since a name is a generic reference holder it is unreasonable to associate a fixed data type with it. However at a given time a name will be bound to some object, which will have a type thus there is dynamic typing.

- The **if** statement, which conditionally executes a block of code, along with `else` and `elif` (a contraction of `else-if`).
- The **for** statement, which iterates over an iterable object, capturing each element to a local variable for use by the attached block.
- The **while** statement, which executes a block of code as long as its condition is true.
- The **try** statement, which allows exceptions raised in its attached code block to be caught and handled by `except` clauses; it also ensures that clean-up code in a `finally` block will always be run regardless of how the block exits.
- The **raise** statement, used to raise a specified exception or re-raise a caught exception.
- The **class** statement, which executes a block of code and attaches its local namespace to a class, for use in object-oriented programming.

- The **def** statement, which defines a function or method.
- The **with** statement, from Python 2.5 released in September 2006, which encloses a code block within a context manager (for example, acquiring a lock before the block of code is run and releasing the lock afterwards, or opening a file and then closing it), allowing Resource Acquisition Is Initialization (RAII)-like behavior and replaces a common try/finally idiom.
- The **break** statement, exits from the loop.
- The **continue** statement, skips this iteration and continues with the next item
-
- The **pass** statement, which serves as a NOP. It is syntactically needed to create an empty code block.
- The **assert** statement, used during debugging to check for conditions that ought to apply.
- The **yield** statement, which returns a value from a generator function. From Python 2.5, yield is also an operator. This form is used to implement coroutines.
- The **import** statement, which is used to import modules whose functions or variables can be used in the current program. There are three ways of using import: import <module name> [as <alias>] or from <module name> import

* or from <module name> import <definition 1> [as <alias 1>], <definition 2> [as <alias 2>],

- The **print** statement was changed to the print() function in Python 3.

4.1.5 Python programming examples

Hello world program:

```
print('Hello, world!')
```

Program to calculate the factorial of a positive integer:

```
n = int(input('Type a number, then its factorial will be printed: '))
```

```
if n < 0:
```

```
    raise ValueError('You must enter a positive number')
```

```
fact = 1
```

```
i = 2
```

```
while i <= n:
```

```
    fact = fact * i
```

```
    i += 1
```

```
print(fact)
```

5. CONCLUSION

This project can overcome the issue of impersonation of a cardholder. This is like a Two factor authentication method which is used to confirm that the transaction is done by the card owner or the persons trusted by the owner using face recognition. It limits the card usage of the unauthorized users who hold the password of someone's card. Thus, this ATM model provides security against exploitation of identity, by using a verification system using face recognition to the identity and confirm the user and it will scale back forced transactions to an excellent extent.

REFERENCES

- [1]Abdulmajeed, Alsufyani¹, Alroobaea¹, Ahmed, Roobaea, **Detection of single-trial EEG of the neural correlates of familiar faces recognition using machinelearning algorithms**, International Journal of Advanced Trends in Computer Science and Engineering, Volume 8, No.6, November – December 2019, pp.2855-2860.
- [2]Aru, O.Ezeand I.Gozie, **Facial Verification Technology for Use in ATM Transactions**, in American Journal of Engineering Research (AJER), [Online] 2013, pp. 188-193. <https://doi.org/10.30534/ijatcse/2019/28862019>
- [3] H.R.Babaei, O.Molalapata and A.A.Pandor, **Face** MurugesanM.*et al.*, International Journal of Advanced Trends in Computer Science and Engineering, 9(2), March - April 2020, 1295 – 1299 1299 **Recognition Application for Automatic Teller Machines (ATM)**, in ICIKM, 3rdvol.45, November – December 2012, pp.211-216.
- [4] E.Derman, Y.K.Gecici and A.A.Salah, **Short Term Face Recognition for Automatic Teller Machine (ATM) Users**, in ICECCO 2013, Istanbul, Turkey, pp.111-114. <https://dx.doi.org/10.21172/1.841.20>
- [5]JinfangXu, Khan, Rasib and RasibHasan, **SEPIA: Secure-PIN-authentication-as-a-service for ATM using Mobile and wearable devices**, 3rdIEEE

International Conference on Mobile Cloud Computing, Services, and Engineering
IEEE, June 2015, pp. 41-50.

[6] Marilou O. Espina¹, Arnel C. Fajardo, Bobby D. Gerardo, RujiP. Medina,

Multiple Level Information Security Using Image Steganography and Authentication, International Journal of Advanced Trends in Computer Science and Engineering, Volume 8, No.6, November – December 2019, pp.3297-3303.
<https://doi.org/10.30534/ijatcse/2019/100862019>

[7] M.Murugesan, R.Elankeerthana, **Support vector machine the most fruitful algorithm for prognosticating heart disorder** , International, Journal of Engineering and Technology, Volume 7, pp.48 – 52,

2018. <https://doi.org/10.14419/ijet.v7i2.26.12533>

[8] M.Murugesan, S.Thilagamani, **Overview Of Techniques For Face Recognition**, International Journal Of Life Science and Pharma Reviews , pp.66 - 71 , 2019 , ISSN 2250 – 0480. <https://dx.doi.org/10.22376/ijpbs/10.SP01/Oct/2019>

[9] M.Murugesan, R.Elankeerthana, **Pedestrian Re- Identification Using Deep Learning**, International Journal Of Life Science and Pharma Reviews, pp.71 - 78 , 2019 , ISSN 2250 – 0480.

[10] P.Pandiaraja, N. Deepa, **A Novel Data Privacy- Preserving Protocol for Multi-data Users by using genetic algorithm**, Journal Soft Computing Volume 23 Issue 18, pp8539-8553, 2019.