TABLE OF CONTENTS

ABSTRACT:

A Point of Sale (POS) is an electronic banking outlet that enables customers to complete the basic financial transactions using debit or credit cards without the aid of bank representative or teller. It is an electronic banking outlet that enables customers to complete the basic financial transactions using debit or credit cards without the aid of bank representative or teller.

POS systems generally accept credit and debit cards to process bills. It is found that there is increase in levels of fraud among the users of credit and debit card. So there is a requirement to build a system that can store the data securely and also do the transactions without giving scope to any fraudulent activities. For the effective and efficient operations of security systems, accurate and automatic recognition of persons is becoming increasingly important.

As a solution our proposed system ‘Cloud Based Point of Sale System with Face Recognition and Password with Cloud implementation’ is proposed as a method of payment making POS systems both card less and cashless.

1. INTRODUCTION

A point-of-sale system, or POS, is the place where a customer makes a payment for product or services at a store. Simply put, every time a customer makes a purchase at a store, they are completing a point-of-sale transaction. It is a combination of hardware and software and used primarily by a business to process customer purchases.

1.1. Problem Definition

POS systems generally accept cash, credit and debit cards to process bills. In recent times in credit card transactions many fraud and security issues are present. Cryptographic keys were fetched and are utilized to achieve privileged access and sensitive information of card holders are stolen. Also, credit card thefts are becoming quite common. It is found that there is increase in levels of fraud among the users of credit and debit card. Information is being stolen by the hackers and used to make fraud transaction. It has been seen that the card information gets stored in POS systems which helps hackers to easily get the card data by hacking the POS system.

In order to decrease the crime happening in massive level a protection profile for Point of Sales System, there is need to enhance the security of the users. Now there exists no Point-of-Sale system that gives full security to the users. Thus, a Point-of-Sale System with an innovative feature such as secure payment gateway by two step authentications, i.e. facenet. recognition using Facenet and password verification, that make forecasts at a real store considering the real time data specific to the domain and at the same time providing users security with the credit card data is needed, to improve the current models.

1.2. Methodologies

Face recognition system is a computer-based application having a capable of identifying or verifying a person by pre-training the selected features from a digital image of a person and thus makes facial database. We use a FaceNet Model for the recognition of faces as a part of Security at the payment Gateway. The complete Face Recognition system can be divided into three categories:

1. Face Detection 2. Feature Extraction 3. Feature Matching

Basically, the output of one method is taken as an input for the next method. And then there will be verification of Password. Cloud computing has given to the possibility of electronic point of sale (EPOS) systems that can be accessed directly from the Internet using any internet browser. Cloud-based POS systems are different from traditional POS largely because user data is not stored locally, but in a remote server. They are independent from platform and operating system limitations. Integrating Cloud Computing with the system makes it more secure.

1.3. Outline of the results

- Cloud Based Point of Sale System with Face Verification and Password Verification is developed.
- For the new customer who wants to use this system should store the details in cloud.
- For this he needs to capture his image and enter details.
- For the existing customer it allows to capture his image and verify password along with the image.
- Then after entering the account details it allows for the transaction.
- If the enough amount is present in the account, then the transaction will be successful else it gets failed.
- Then the customer gets an SMS to his mobile.

1.4. Scope of the project

POS systems generally accept cash, credit and debit cards to process bills and therefore they have the capability to extract all the sensitive data of the customer. They act as a basic interface for the card transaction and also include many additional features. But it is found that there is increase in levels of fraud among the users of credit and debit card. Information is being stolen by the hackers and used to make fraud transaction. There exists no POS system that gives full (100%) security. In practice, AMSR (Advanced Microwave Scanning Radiometer) a minor hardware dongle reads magnetic stripes on cards and then hides the sensitive information of the card and then sends the outcome to the application. It hides all the sensitive information on card by encrypting and stores them on cloud. But few mobile applications can deactivate AMSR permanently and fetch cryptographic keys and can hack the data. It is therefore necessary to provide a secure payment gateway. And also, the data is usually stored on an on- site PC or server. But this system process and store data online i.e., in cloud as Cloud computing shows great potential in this regard opposed to locally on our computer or on-premise server.

2. PROJECT

2.1. LITERATURE SURVEY

2.1.1. Introduction to the Problem Domain Terminology

Artificial intelligence (AI) refers to the simulation of human intelligence in machines that are programmed to think like humans and mimic their actions. Facial recognition technology is a set of algorithms that work together to identify people in a video or in a static image. We can integrate artificial intelligence (AI) within facial recognition systems. Intelligent, AI-based software can instantaneously search databases of faces and compare them to one or multiple faces that are detected in a scene. In an instant, you can get highly accurate results – typically, systems deliver 99.5% accuracy rates on public standard data sets. AI face recognition software has the following advantages: Real-time identification, Anti-spoofing measures, lessened racial or gender bias due to model training across millions of faces, can be used across multiple cameras.

Deep learning is a subset of AI's machine learning, and it has networks that can learn from unstructured or unlabelled data – and it can do so without supervision. It imitates the processing power and pattern-creation capabilities of the human brain and uses those abilities to make decisions. It depends on level of training the neural network, how accurately a well-trained AI facial recognition system can recognize the faces in real world context.

Cloud computing means storing and accessing data and programs over the internet instead of our computer's hard drive. It provides hardware and software services from a provider on the internet. Cloud is described as data centres available to many users over the Internet. Cloud computing offers flexibility, data recovery, little to no maintenance, easy access and a higher level of security.

2.1.2Existing Solutions and Related work

There are many existing solutions available for the secure payment at the payment gateway. The incorporation of fingerprint biometric recognition as an additional layer of protection to the customary pin and password requirements to gain permission to pay for goods purchased and services rendered using point of sale device can guarantee the secured transaction.

But it is possible that our fingerprints can be identified from public photographs that show our hands. Therefore, fingerprint scanning can't guarantee the secured transactions. And hence, Iris scan was introduced at the authentication level of the existing POS machine. The experimental results shows that the system could significantly minimize cardholder fraud at the POS machine but not at a high rate. Currently, face recognition is used as a technology to provide multiple security in various practices likes verification of identity, access authority, observation, to replace passwords and identity cards that are no longer safe. The use of face recognition has the benefit of verifying personal data because, inhuman faces things like

irises, retinas, faces are very unique to each other. For the effective and efficient operations of security systems, accurate and automatic recognition of persons is becoming increasingly important.

Face recognition is one branch of computer science is an ability to recognize or identify the person's identity by analysing the pattern-based facial contours of human faces. Face recognition has many methods in its application today. For example, Haar Cascade, Principal Component Analysis (PCA), Linear Discriminant Analysis (LDA), and Deep Learning. Haar feature is a wavelet-based feature that decomposes image. The principle of the process is, the parameter of the image is detected whether it has face characteristic or not. If it has the separation stage of face and background, the image will be processed for further classification. The formula of Haar feature is, the average value of the result is above the threshold, it means the Haar feature exists. The closer the value to 1, the more likely we have found the face. Haar cascade face detection can detect images at different scales. But the major drawback is that it gives false results as well as it doesn't work on non-frontal images.

PCA is a multivariate technique that analyse a data table in which observations are described by several inter-correlated quantitative dependent variables. It is a dimensional reduction method that is often used to reduce dimensions on a large enough dataset. You do this by changing variables in large datasets to smaller parts but still using the information on large datasets. Eigenface is one of the facial recognition methods based on the Principal Component Analysis (PCA) algorithm. One of the PCA methods is Eigenfaces. Eigenfaces is unsupervised and it ignores all the class labels. However, Eigenfaces are not optimized for class separation. Eigenvectors are derived from covariance matrix which has a high probability distribution and vector space dimension to recognize the possibility of a face by reconstructing the face. The disadvantage of using PCA is a covariance matrix that is difficult to evaluate accurately, the difficulty of capturing invariance on PCA unless the data trainee provides information explicitly. The results of face pre-process will be compared with the training model. PCA results in loss of information if the number of Principal Components is not selected wisely.

LDA is a supervised data that is very strong against dimension reduction. By using supervised, LDA is able to work quite well on a dataset containing a greater number of face images. The purpose of the LDA method is to detect groups of test objects based on the closest average. Each group has an average vector of each vector characteristic for each object. The closest measurement is using a distance metric. The advantages of LDA are the simple, fast and portable method, and are good for the initial project. The disadvantage of LDA is an old algorithm. PCA is mainly used for feature extraction while the LDA is used for classification.

There are many face recognition methods in Deep Learning, such as the use of Convolutional Neural Networks (CNN) and Artificial Neural Network (ANN). Another efficient technique for face recognition system based on Deep Learning using Convolutional Neural Network (CNN) with Dlib face alignment was then introduced. D-lib was used for face alignment of the frontal faces; the alignment could be effective for 45 degrees. This system uses computer vision technology for frontal faces detection by facial landmark detection. The major disadvantage is that it is trained on a minimum face size of 80*80 so it can't detect small faces in images. It is also very slow on the CPU.

FaceNet is one of the uses of face recognition based on Deep Learning. That is a one-shot learning method using Euclidean space to calculate the similarity distance for each face. FaceNet is a fairly new method, introduced by Google researches in 2015, using Deep Convolutional Network method. The advantage of using the FaceNet method is that this model only requires minimal alignment in terms of cutting the face area quite tightly. FaceNet is able to provide accuracy of up to 99.63% from dataset Labelled Faces in the Wild (LFW) and 95.12% on the YouTube Faces DB.

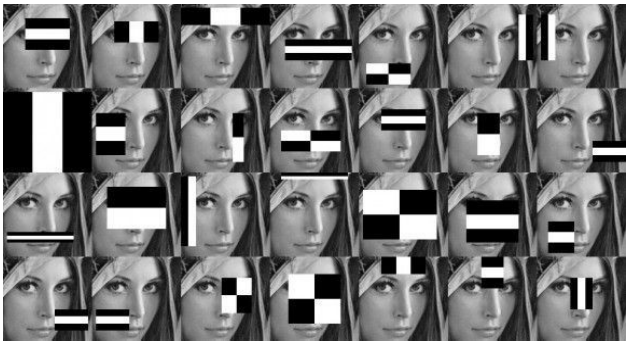


FIG : HAAR CASCADE

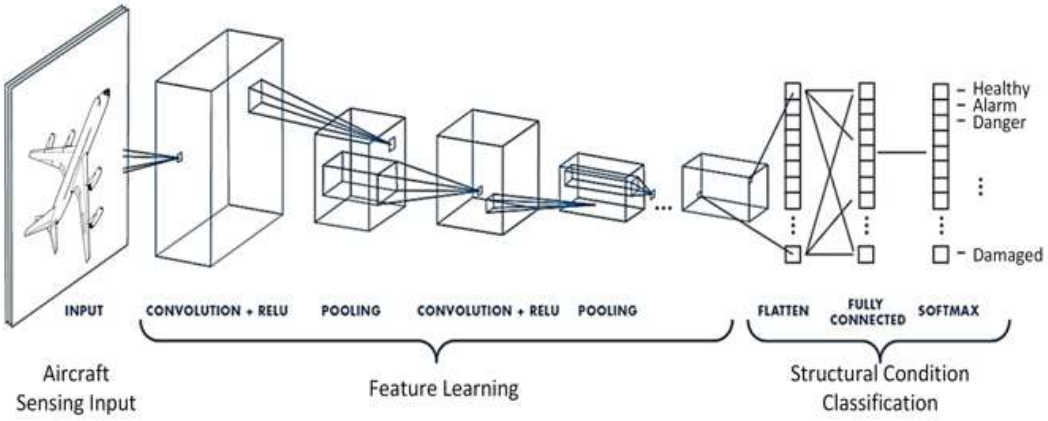


FIG : CONVOLUTION

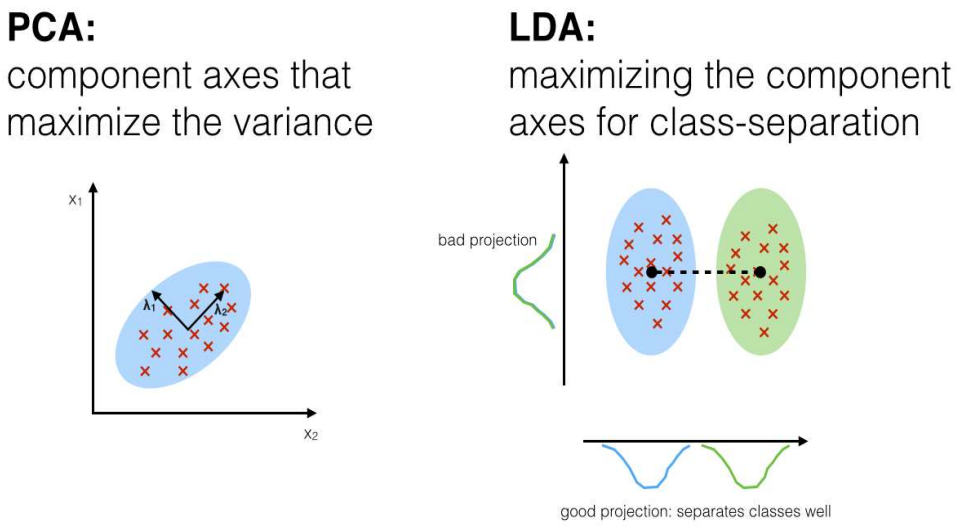


FIG : PRINCIPAL COMPONENT ANALYSIS VS LINEAR DISCRIMINANT ANALYSIS

Deep neural networks are able to extract more meaningful features than machine learning models. The downfall of these big networks is however the need for a huge amount of data. We managed to cope with this issue by using a pre-trained model, a model that has been trained on a way bigger dataset in order to retain knowledge on how to encode face images, which we then used for our purposes in this challenge. In addition, fine-tuning SVM really helped us to push beyond the accuracy of 95%.

2.2. Tools/Technologies used

2.2.1. Amazon Web Services :

AWS is one of the largest cloud providers along with Microsoft Azure and Google Cloud Platform. There are a lot of services offered by Amazon including AWS S3. Amazon S3 or Amazon Simple Storage Service is a service offered by Amazon Web Services (AWS) that provides object storage through a web service interface.

History:

Founding (2000–2005)

The genesis of AWS came in the early 2000s. Prior experience with building Merchant.com, Amazon's e-commerce-as-a-service platform for third-party retailers to build their own web-stores, led them to pursue as a means to scale their engineering operationsled by the then Allan Vermeulen. Around the same time frame, Amazon was frustrated with the speed of its software engineering, and sought to implement various recommendations put forth by Matt Round, an engineering leader at the time, including maximization of autonomy for engineering teams, adoption of REST, standardization of infrastructure, removal of gate-keeping decision-makers (bureaucracy), and continuous development. He also called for increasing the percentage of the time engineers spent building the software rather than doing other tasks.
In November 2004, AWS launched its first infrastructure service for public usage Simple Queue Service (SQS).

S3, EC2, and other first generation services (2006–2010)

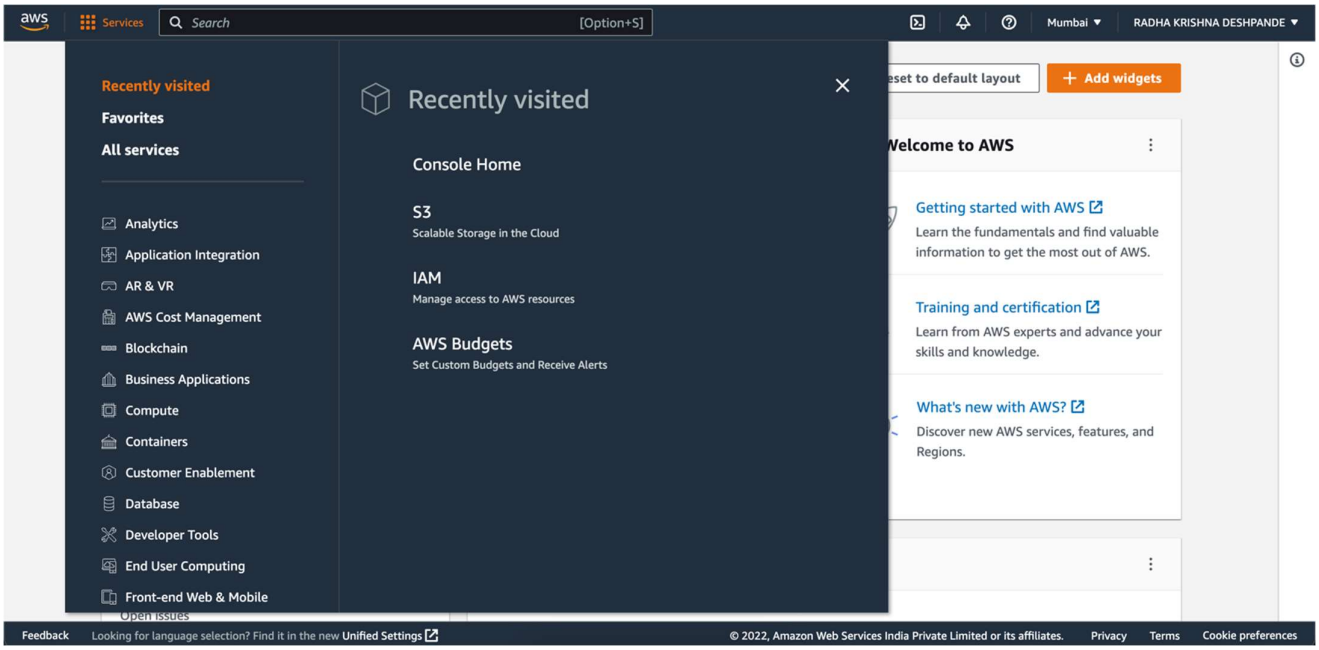
In September 2007, AWS announced its annual Start-up Challenge, a contest with prizes worth $100,000 for entrepreneurs and software developers based in the US using AWS services such as S3 and EC2 to build their businesses.

Growth (2010–2015)

In November 2010, it was reported that all of Amazon .com's retail sites had migrated to AWS. Prior to 2012, AWS was considered a part of Amazon.com and so its revenue was not delineated in Amazon financial statements. In that year industry watchers for the first time estimated AWS revenue to be over $1.5 billion. In April 2015, Amazon.com reported AWS was profitable, with sales of $1.57 billion in the first quarter of the year and $265 million of operating income.

Market leadership (2016–present)

In June 2022, Amazon announced and displayed a preview of their latest AI tools for programmers called Code Whisperer. While currently only available to people who received an invitation though the AWS IDE Toolkit, Code Whisperer is an IDE plugin that will examine the users code and comments and present users with syntactically correct suggestions. The Suggestions will be based on coding style and variable names and are not just snippets. Set to be free for the review period, the company will be charging for the product on a subscription bases upon full release.



2.2.2. FLASK AND PYTHON

What is Flask Python?

Flask is a web framework, it's a Python module that lets you develop web applications easily. It's has a small and easy-to-extend core: it's a microframework that doesn't include an ORM (Object Relational Manager) or such features.

It does have many cool features like url routing, template engine. It is a WSGI(Web Server Gateway Interface) web app framework.

What is Web Framework?

A Web Application Framework or a simply a Web Framework represents a collection of libraries and modules that enable web application developers to write applications without worrying about low-level details such as protocol, thread management, and so on.

What is FLASK?

Flask is a web application framework written in Python. It was developed by Armin Ronacher, who led a team of international Python enthusiasts called Poocco. Flask is based on the Werkzeg WSGI toolkit and the Jinja2 template engine. Both are Poocco projects.

WSGI

The Web Server Gateway Interface (Web Server Gateway Interface, WSGI) has been used as a standard for Python web application development. WSGI is the specification of a common interface between web servers and web applications.

Werkzeug

Werkzeug is a WSGI toolkit that implements requests, response objects, and utility functions. This enables a web frame to be built on it. The Flask framework uses Werkzeg as one of its bases.

jinja2

jinja2 is a popular template engine for Python.A web template system combines a template with a specific data source to render a dynamic web page.

Why is FLASK is a good web framework choice?

Unlike the Django framework, Flask is very Pythonic. It's easy to get started with Flask, because it doesn't have a huge learning curve. On top of that it's very explicit, which increases readability. To create the "Hello World" app, you only need a few lines of code.

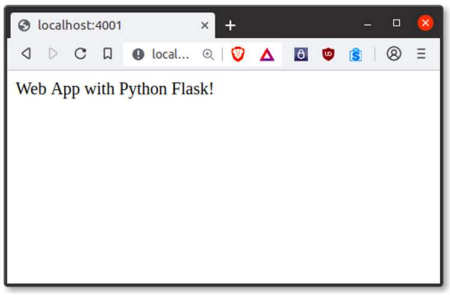
This is a boilerplate code example.

```
from flask import Flask
app = Flask(__name__)

@app.route('/')
def hello_world():
    return 'Hello World!'

if __name__ == '__main__':
    app.run()
```

If you want to develop on your local computer, you can do so easily. Save this program as server.py and run it with python server.py.

```
$ python server.py
 * Serving Flask app "hello"
 * Running on http://127.0.0.1:5000/ (Press CTRL+C to quit)
```

It then starts a web server which is available only on your computer. In a web browser open localhost on port 5000 (the url) and you'll see "Hello World" show up.



It’s a microframework, but that doesn’t mean your whole app should be inside one single Python file. You can and should use many files for larger programs, to handle complexity.

Micro means that the Flask framework is simple but extensible. You may all the decisions: which database to use, do you want an ORM etc, Flask doesn’t decide for you.

Flask is one of the most popular web frameworks, meaning it’s up-to-date and modern. You can easily extend its functionality. You can scale it up for complex applications.

Many more tools used are given below

Above are the HTML, CSS, JS uses for the Framework with BASH language.

Tensor Flow JS is the elemental language which is uses for the WEBCAM.

2.2.3. USING FACE NET

To really push the limits of face detection we will look at some state-of-the-art methods. Modern-day face extraction techniques have made use of Deep Convolution Networks. As we all know that features created by modern deep learning frameworks are really better than most handcrafted features. We checked 4 deep learning models namely, FaceNet (Google), DeepFace (Facebook), VGGFace (Oxford), and OpenFace (CMU). Out of these 4 models FaceNet was giving us the best result. In general, FaceNet gives better results than all the other 3 models.

FaceNet is considered to be a state-of-art model developed by Google. It is based on the inception layer, explaining the complete architecture of FaceNet is beyond the scope of this blog. Given below is the architecture of FaceNet. FaceNet uses inception modules in blocks to reduce the number of trainable parameters. This model takes RGB images of 160×160 and generates an embedding of size 128 for an image. For this implementation, we will need a couple of extra functions. But before we feed the face image to FaceNet we need to extract the faces from the images.

TABLE II. BEST SPLIT POINT FOR THRESHOLD IN MODEL METRIC PAIRS

	Cosine	Euclidean	Euclidean L2
VGGFace	Threshold: 0.31 Accuracy: 89.28 Precision: 97.41 Recall: 80.71 F1: 88.28	Threshold: 0.47 Accuracy: 81.42 Precision: 97.82 Recall: 64.28 F1: 77.58	Threshold: 0.79 Accuracy: 89.28 Precision: 97.41 Recall: 80.71 F1: 88.28
FaceNet	Threshold: 0.40 Accuracy: 98.21 Precision: 100 Recall: 96.42 F1:98.18	Threshold: 11.26 Accuracy: 98.57 Precision: 100 Recall: 97.14 F1: 98.55	Threshold: 0.90 Accuracy: 98.21 Precision: 100 Recall: 96.42 F1: 98.18
OpenFace	Threshold: 0.11 Accuracy: 57.85 Precision: 95.83 Recall: 16.42 F1: 28.04	Threshold: 0.47 Accuracy: 57.85 Precision: 95.83 Recall: 16.42 F1: 28.04	Threshold: 0.47 Accuracy: 57.85 Precision: 95.83 Recall: 16.42 F1: 28.04
DeepFace	Threshold: 0.13 Accuracy: 54.64 Precision: 100 Recall: 9.28 F1: 16.99	Threshold: 42.21 Accuracy: 52.50 Precision: 100 Recall: 5.00 F1: 9.52	Threshold: 0.51 Accuracy: 54.64 Precision: 100 Recall: 9.28 F1: 16.99

2.3.DESIGN OF THE PROPOSED SYSTEM

2.3.1.Block Diagram

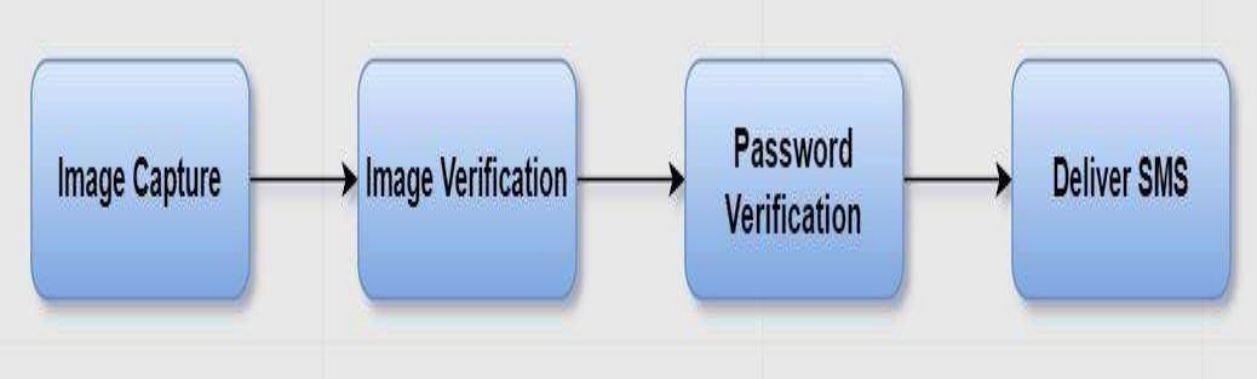


Fig: Block Diagram of proposed system

1. The above figure is the block diagram of proposed system containing four main steps.
2. First, whoever the customer maybe, whether new or existing customer, one needs to capture their image.
3. After capturing image, if the customer is new then the image gets stored in the cloud along with the credentials the persons gave.
4. If the customer is existing one, then simply the image gets stored in a local file.
5. Then, after Capturing the image, it gets verified with the images present in the cloud.
6. If a match is found for the image, then it proceeds for the Password Verification. If no match found transaction gets failed.
7. Then Payment is done and a message is sent to customers mobile number regarding this.

2.3.2Module Description

2.3.2.1.Customer Type Module:

In this module the customer can choose whether he's a new customer or an existing customer. It contains a page which consists of two buttons. One is for the new user and the other is for the existing user. Upon clicking the buttons it will re direct to the respected modules to finish off the transactions or to store the details of new customer.

2.3.2.2.New User Module

On choosing new user in the customer type module, it redirects to the New user module. Here the system takes the details of the new customer and store them in the cloud. First Page consists of a button that allows to capture the image of the customer. Then on pressing upload image, customer’s captured image is stored in the cloud along with an Id. Then a button create password is viewed. It asks to create a password and then after stored in the cloud with respect to the customer Id. After successfully uploading the details, a page saying customer details successfully uploaded is viewed.

2.3.2.3.Existing User Module

On choosing new user in the customer type module, it redirects to the New user module. It consists of a page that contains a capture button to capture the customer's face. Then it gets stored in a local file. After pressing the verify image button, system starts to verify the image from the local file with the images present in the cloud. If no match is found or face verification gets failed, a page appears saying face verification failed. On successful verification, a button appears to enter the password and password gets verified with the password stored in the customer with respect to the customer's id as a second step of authentication. If the password doesn't match, transaction halts and a page saying password verification failed will appear. On successful verification of password it redirects to the payment module for processing payments.

2.3.2.4.Payment Module

On successful verification of password existing user module redirects to this module for processing payments. Initially it asks for the customer to enter the account id, amount to pay and the phone number. After pressing the pay button, the system checks whether sufficient amount for paying is available or not. If not a page appears saying not enough money. If yes, amount is deducted from the account and a message is sent to the entered mobile number. And a page appears saying transaction successful.

2.4.Theoretical Foundation

2.4.1.FaceNet

FaceNet is a method that uses deep convolutional networks to optimize its embedding, compared to using intermediate bottleneck layers as a test of previous deep learning approaches. This method is called one-shot learning. In more detail, this method can use a small sample of face images to produce the initial model, and when there are new models, the initial model can be used without retraining. FaceNet directly trains the face using the Euclidean space where the distance consists of similarities between facial models. When the results of similarities between face models are obtained, it will be easy to carry out face recognition and classification using FaceNet attached become feature vectors.

In the training process, FaceNet applies triplets by matching face to face with the online novel triplet mining method. Of course, this triplet consists of a collection of anchor images, where each image consists of positive and negative images. (Fig.1) shows the structural model used in FaceNet. FaceNet consists of batch layers as input and deep architecture which is deep CNN followed by L2 normalization, that become the result of face embedding. FaceNet also pursued by the triplet loss when the training process (Fig.2).

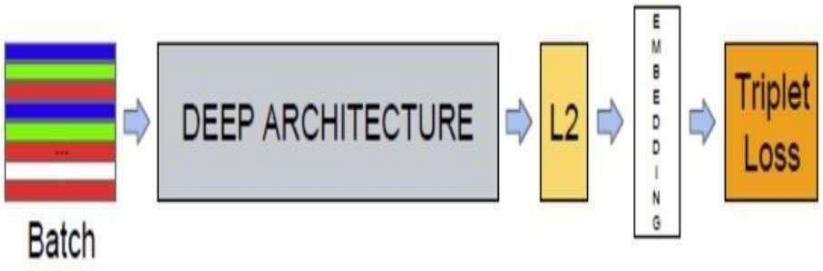


Fig 1: Model structure of FaceNet

Triplet loss training methods have three main elements namely anchor, positive and negative as shown in Fig 3.3. This triplet loss works by minimizing the distance between anchors positively and maximizing the distance between anchors negatively. Where this positive has the same identity as the anchor and negative has a different identity from the anchor. FaceNet trains its output directly into concise 128- dimensional embedding by applying triplet-based loss method depend on LMNN. It formed by two thumbnails of compared faces and thumbnails that do not match and the loss aim to distinguish between positive and negative pairs using a range of limit. Thumbnails were cut tightly on the face field, it didn't need 2D or 3D adjustment, apart from the ratio and translation implemented.

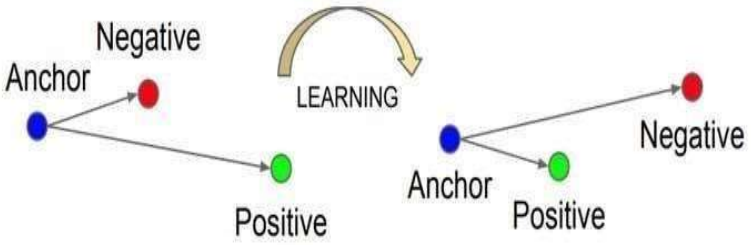


Fig 2: Triplet Loss training

2.4.2.Multi-task Cascaded Convolutional Neural Networks (MTCNN)

Convolutional Neural Network is a neural network design that aims to rate visual images and datasets. MTCNN or Multi-task Cascaded Neural Networks is a developed version of CNN . Shows that the proposed MTCNN model show the innate bond between disclosure and the adjustment of the usage frames to increase the performance. Purpose of the proposed MTCNN is to form an avalanched structure and use it as material for multi-task knowledge to forecast the location of the face and it is marked in a coarse-to-fine method. Also, MTCNN aims the bond of 2 tasks. And in its application, MTCNN is able to detect real-time with fairly high accuracy.

The MTCNN model made from three networks as shown in (Fig.3) first is the Proposal Network (P-Net) which functions to get the face area and give some bounding boxes to the face. Second is the Refine Network (R-Net), which functions to remove some bounding boxes on the face by calibrating them and leaving only an accurate bounding box. And the last is the Output Network (O-Net). The workings of the O-Net are different from the previous layers, the O-Net takes the result of the R-Net in the form of a boundary box and divides it into three different layers: the first layer for face probabilities in the box, the second layer to give the boundary coordinates in the box, the last layer for the coordinates of the five landmarks of the face.

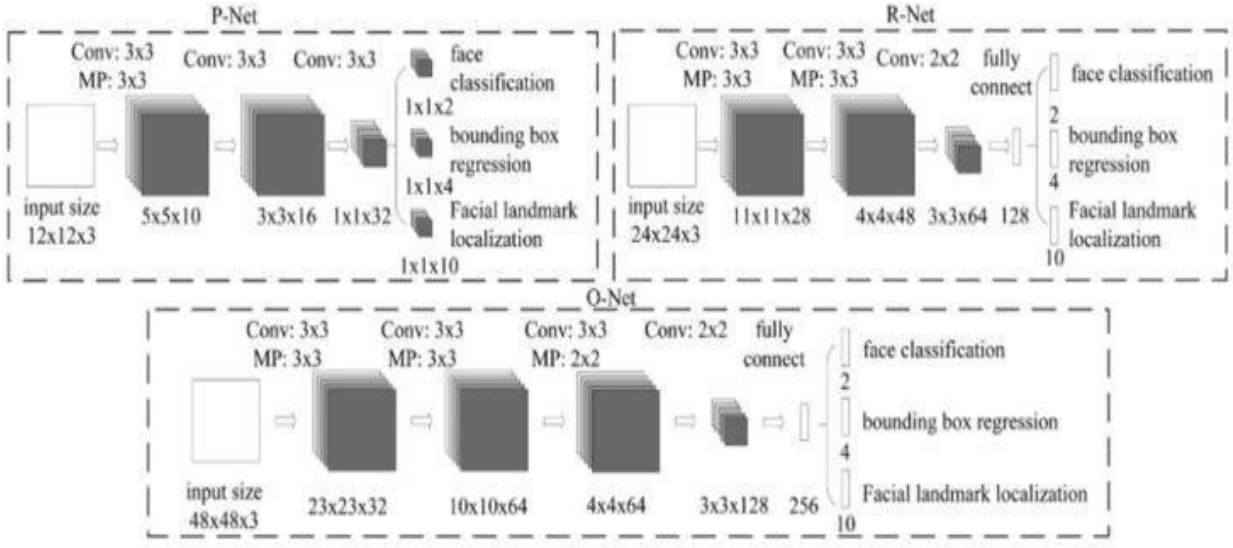
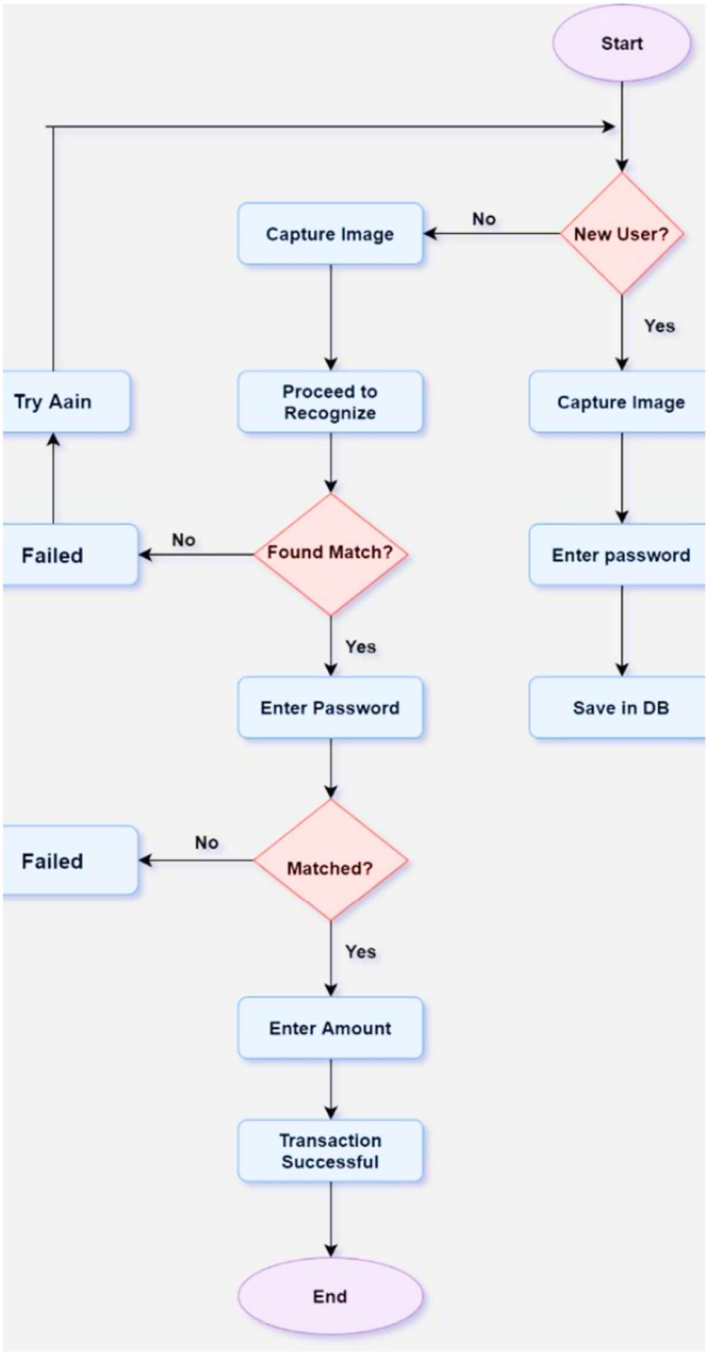


Fig 3: MTCNN Architecture

2.5. Implementation of Proposed System

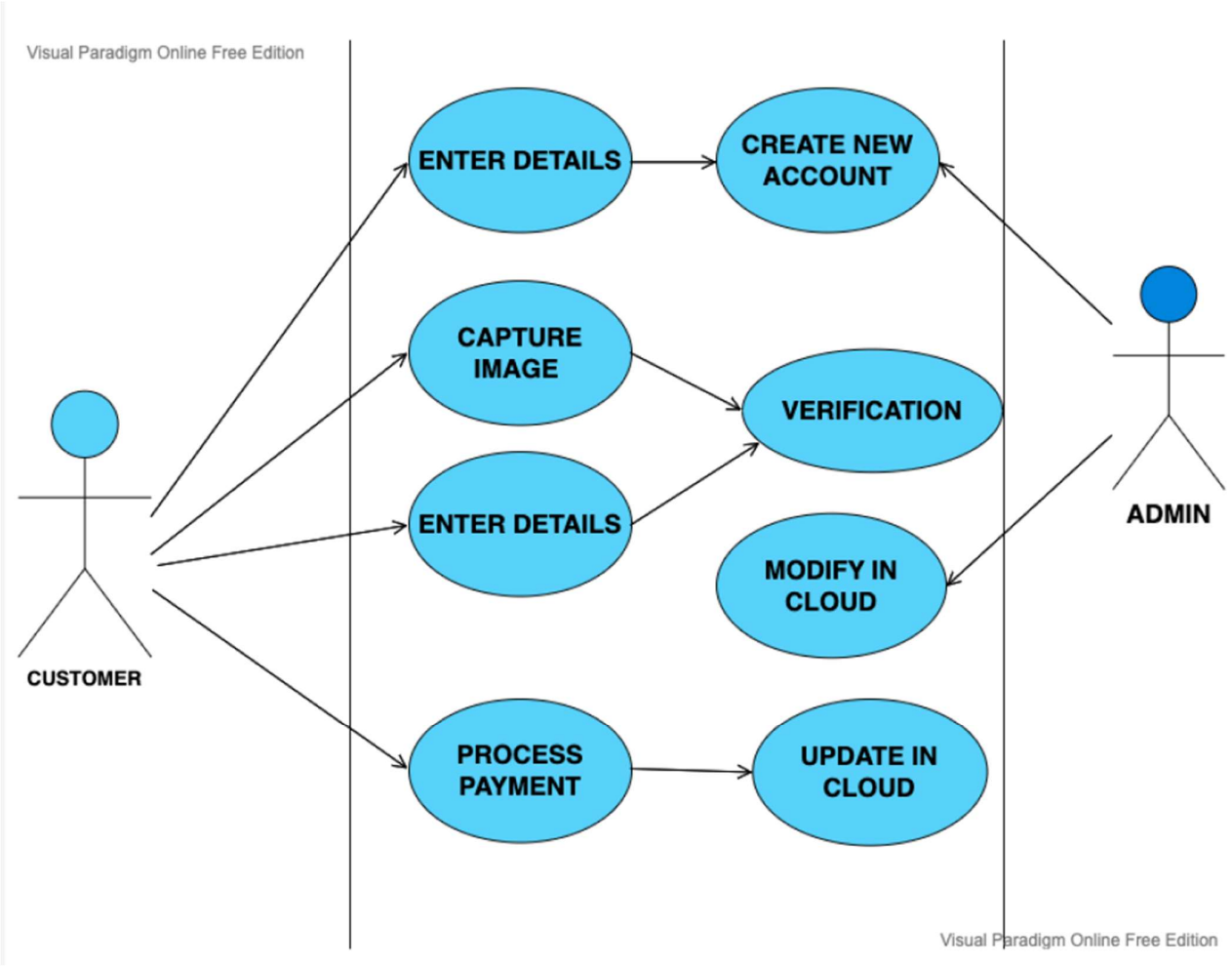
2.5.1.Flow Chart

The proposed System with two factor authentication will take the flow as shown in Fig.



1. The system will read input to verify if a customer is new user or an existing user.
2. If the customer is a new user, the system will read user personal details along with his image and store the sensitive data in the cloud.
3. If the customer is an existing user, the system will perform face verification followed by password verification and only after being authenticated can proceed with transaction.

2.5.2.Use Case Diagram



The actors for the model Cloud Based POS System with Face Recognition from Fig will be the following entities:

1. Administrator: The administrator uploads the customer images and passwords to cloud. He will take up the task for maintaining the data on cloud and updating whenever required.
2. Customer: The customer will be the person who wants to complete a payment for a purchase.

Major function performed by the actors:

3. Maintaining data in cloud: This will involve managing all customer image, password files on cloud.
4. Face authenticator: This will involve capturing customer image and executing the face verifier function.
5. Password authenticator: This will involve calling password verifier function along with reading actual input from user and fetching expected input from cloud.

2.6. Algorithm

Step 0: Start Phase.
Step 1: Import cv2 module of python
Step 2: Capture customer image and save it
Step 3: Read the type of user. If user is an existing user jump to Step 5
Step 4: (For new user) Read a new password from user and upload captured image along with password to cloud. Terminate the process
Step 5: (For existing user) Import MTCNN module of python
Step 6: Save the cropped and aligned image.
Step 7: Import FaceNet module of python
Step 8: Calculate embeddings for the current customer.
Step 9: Find embedding distance with all embeddings of all other customer images in the cloud.
Step 10: If there exists a pair with distance less than expected threshold return success. Else terminate the process.
Step 11: Redirect to payment gateway that accepts Password as second authentication layer to proceed with the transaction.
Step 12: On receiving “Success” from payment gateway close transaction.
Step 13: Send a message to the customers mobile Number.
Step 14: Stop.

3. TESTING PROCESS

3.1. Bounding Box Coordinates

Print the coordinates of bounding box by calling detect faces function.
Output: The detector returns a list of JSON objects as shown in Fig . Each JSON object contains three main keys: 'box', 'confidence' and 'key points':

- The bounding box is formatted as [x, y, width, height] under the key 'box'.
- The confidence is the probability for a bounding box to be matching a face.
- The key points are formatted into a JSON object with the keys 'left eye', 'right eye', 'nose', 'mouth left', 'mouth right'. Each key point is identified by a pixel position (x, y).

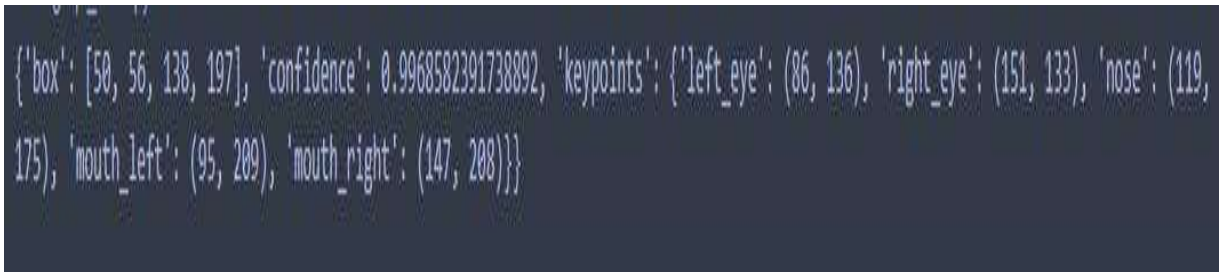


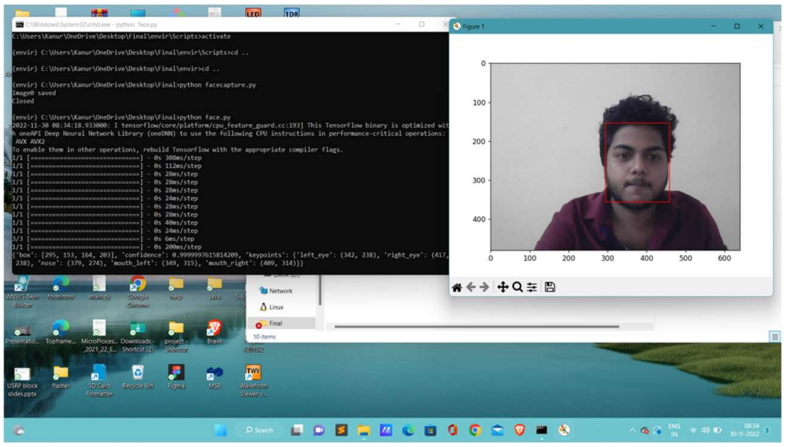
Fig : Bounding Box Coordinates

3.2.Rectangular plot on the face

- Draw the rectangle of the bounding box
- By passing in the coordinates obtained by calling detect faces function. By using

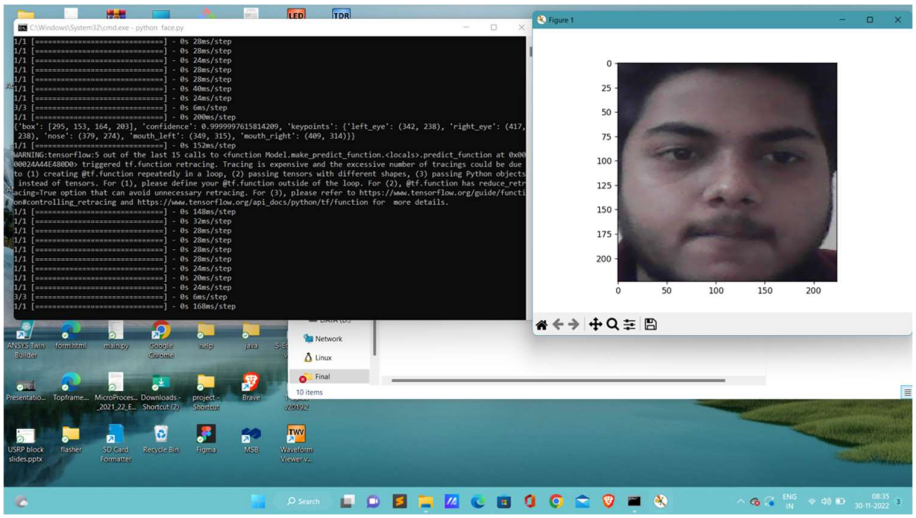
  matplotlib the co-ordinates obtained by the bounding box is plotted on the image

- A box is drawn around the face lining with the co-ordinates.
- Output of the testcase: The output will have a rectangular plot around the face as shown in Fig.



3.3.Extracted Face

- Plot the extracted face eliminating the background noise.
- From the output obtained by rectangular plot, the image inside the plot is needed to be extracted from the captured image.
- The output image will contain only the main face as shown in Fig 4.6 from the input, consisting of only face with eyes, nose, lips and forehead and jawline.



3.4.Embedded Vector

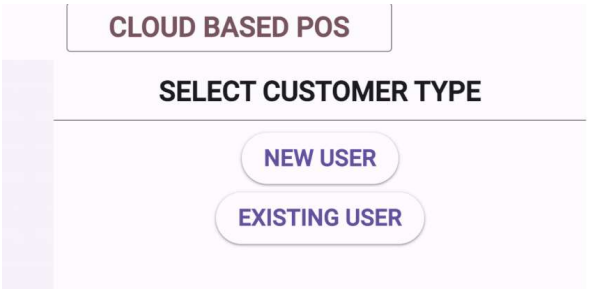
- Print the embedding vector of a facial image using FaceNet Classifier.
- The output will contain embedding vectors of an input face image.
- The input face image is the extracted face obtained.

```
([[-4.7446e-02,  1.1238e-02, -7.7405e-02, -1.0071e-01, -3.9388e-02,
   4.5476e-02, -2.9879e-02,  5.1630e-03,  1.9694e-02,  1.1588e-02,
  -2.0864e-02,  3.4005e-02,  7.0845e-03, -2.5673e-02, -4.7758e-02,
  -3.1949e-02,  1.8853e-02,  5.6145e-03,  2.6090e-02, -1.3143e-01,
  -6.4763e-02,  4.6367e-02,  7.9812e-02, -3.2136e-02,  5.0248e-02,
  -3.8745e-02, -8.3709e-02, -3.6629e-02,  9.0813e-03, -2.5537e-02,
  -6.4530e-02,  1.0447e-02,  2.1252e-02, -1.3456e-02,  4.1936e-02,
   1.1284e-01,  3.9611e-02,  1.4769e-03, -1.7361e-02,  7.8464e-03,
  -2.1736e-02, -5.4847e-02, -6.4537e-03,  8.4894e-03,  2.2331e-02,
  -3.5829e-02, -3.7084e-02, -4.1738e-02, -3.1373e-02, -2.7309e-03,
   3.0114e-02, -5.3937e-02, -2.0671e-02,  2.0716e-02, -1.9630e-02,
   9.5843e-02, -4.0588e-02,  4.6252e-02, -2.3585e-02, -9.7828e-03,
   5.0773e-02,  3.9131e-02, -7.4135e-02,  2.2864e-02, -4.1212e-02,
   7.8805e-03,  4.4853e-02,  1.1676e-01, -1.2790e-02,  2.7636e-02,
   1.4557e-02, -4.2889e-02,  3.7055e-02,  3.4139e-04,  2.3169e-02,
  -1.0955e-01, -6.2067e-02,  1.0967e-01,  1.4153e-02, -1.8695e-02,
  -2.9038e-02,  5.7727e-03,  7.6315e-03, -4.5169e-02, -4.2627e-02,
   1.8517e-02, -9.9954e-02,  6.8754e-03, -9.2949e-03,  3.9462e-02,
   7.8731e-02,  2.3698e-02,  3.8656e-02, -4.3716e-02, -2.3006e-02,
   2.1008e-02.  2.5000e-02.  1.3549e-02. -5.9497e-02.  1.1141e-02.
```

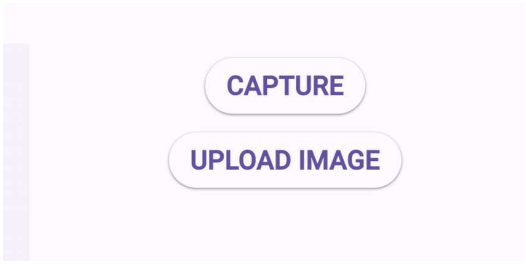
Fig : Embedded Vector

4. RESULTS

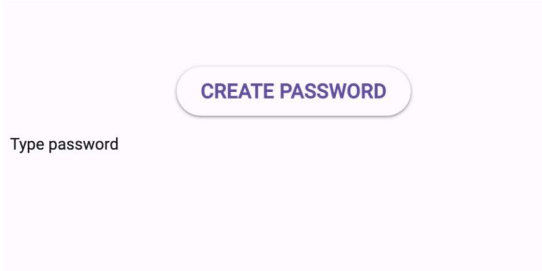
Snippets of the Point-of-Sale System and a brief description about them is described below.



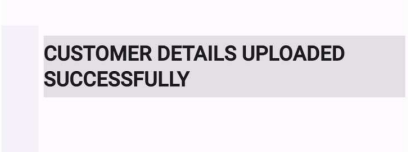
Frame 1



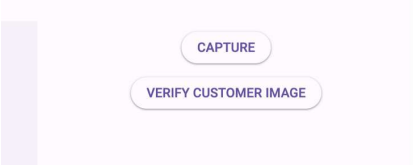
Frame 2



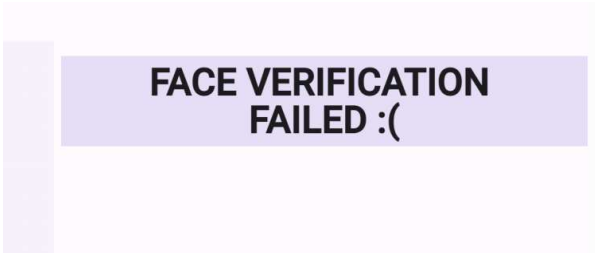
Frame 3



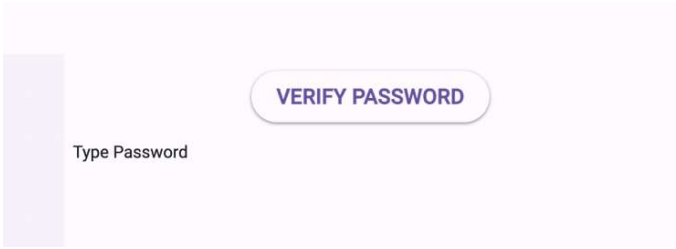
Frame 4



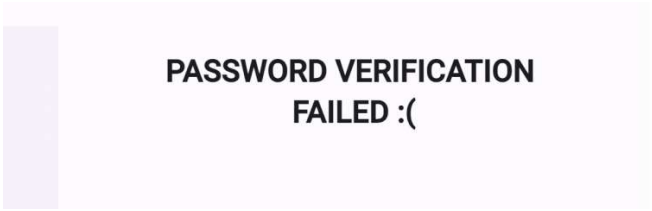
Frame 5



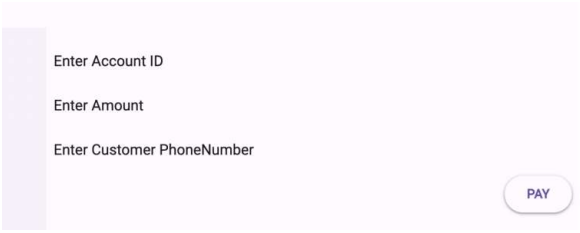
Frame 6



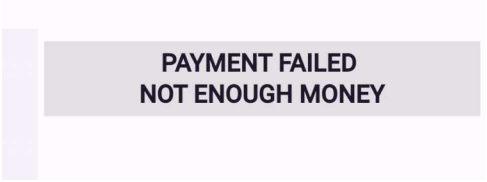
Frame 7



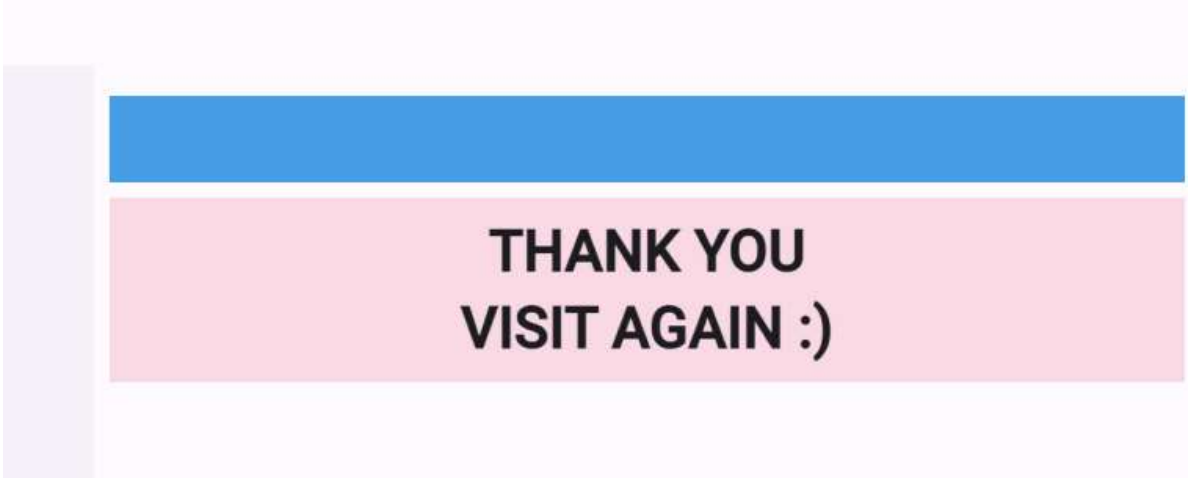
Frame 8



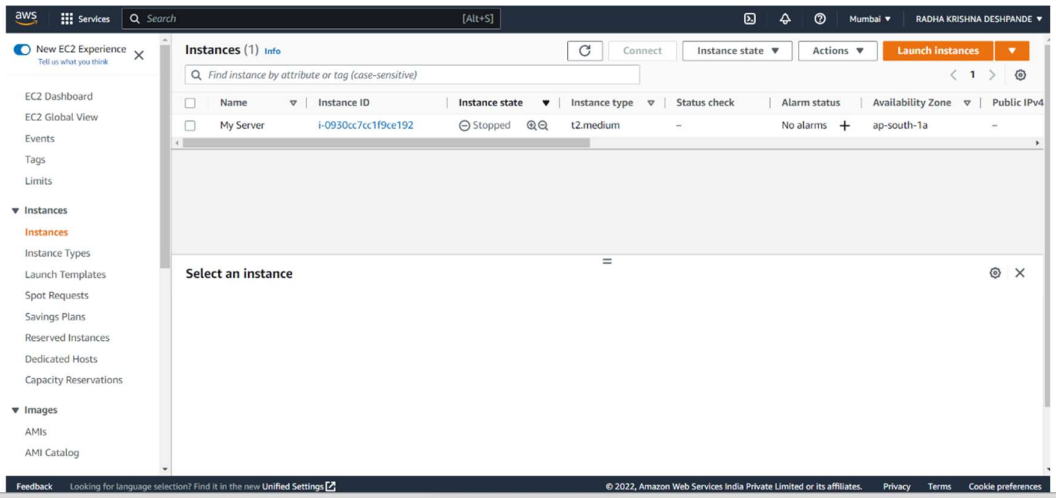
Frame 9

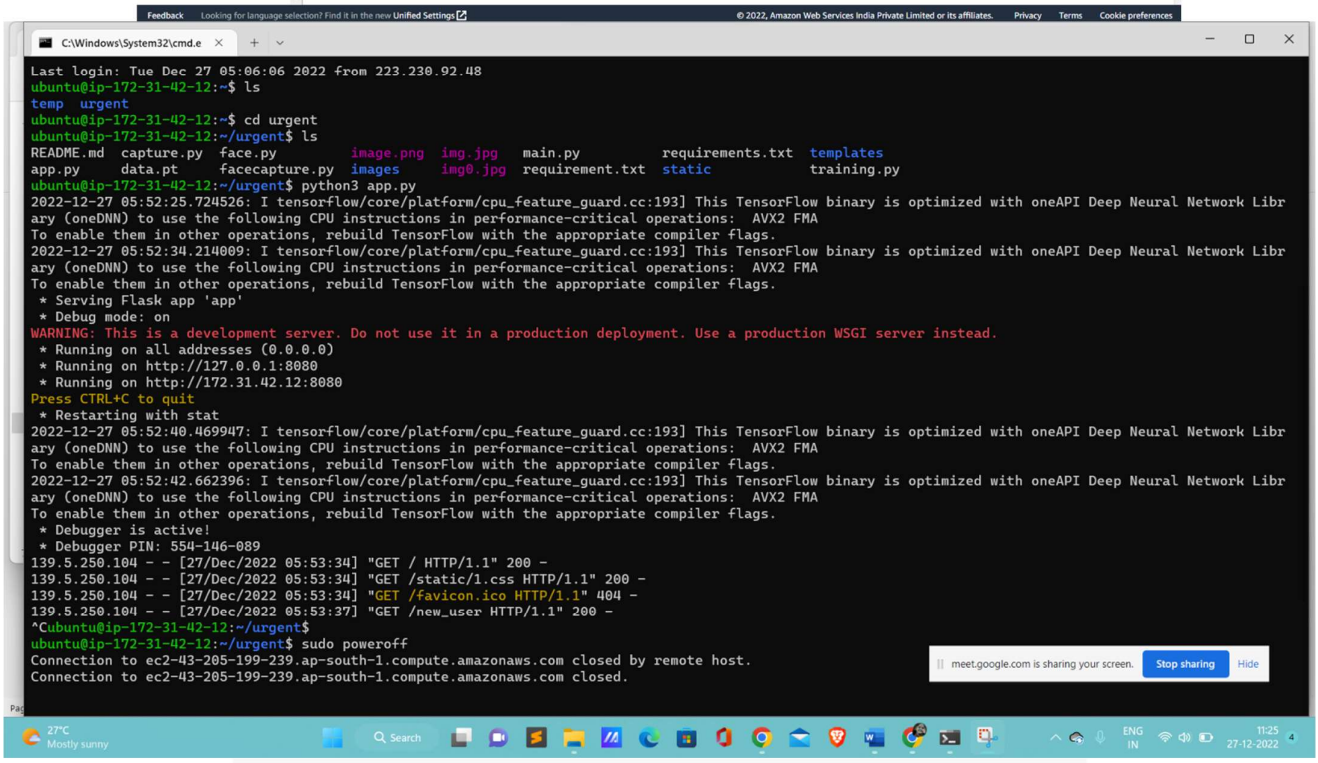


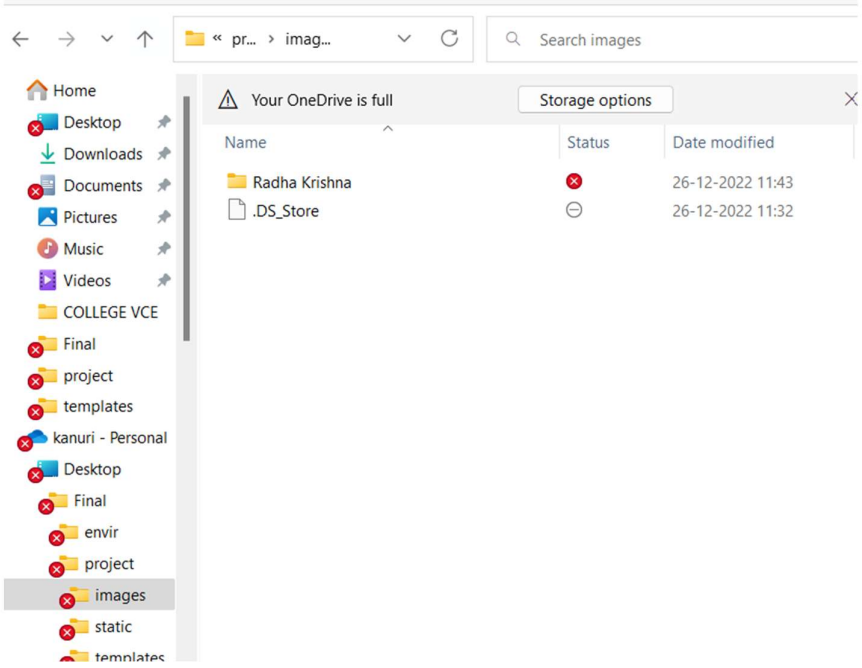
Frame 10



Frame 11


Instances (1) Info
Connect
Instance state
Actions
Launch instances
Find instance by attribute or tag (case-sensitive)
Name
Instance ID
Instance state
Instance type
Status check
Alarm status
Availability Zone
Public IPv4
My Server
i-0930cc7cc1f9ce192
Stopped
t2.medium
No alarms
ap-south-1a
Select an instance
EC2 Dashboard
EC2 Global View
Events
Tags
Limits
Instances
Instance Types
Launch Templates
Spot Requests
Savings Plans
Reserved Instances
Dedicated Hosts
Capacity Reservations
Images
AMIs
AMI Catalog


C:\Windows\System32\cmd.e
Last login: Tue Dec 27 05:06:06 2022 from 223.230.92.48
ubuntu@ip-172-31-42-12:~$ ls
temp urgent
ubuntu@ip-172-31-42-12:~$ cd urgent
ubuntu@ip-172-31-42-12:~/urgent$ ls
README.md capture.py face.py image.png img.jpg main.py requirements.txt templates
app.py data.pt facecapture.py images img0.jpg requirement.txt static training.py
ubuntu@ip-172-31-42-12:~/urgent$ python3 app.py
2022-12-27 05:52:25.724526: I tensorflow/core/platform/cpu_feature_guard.cc:193] This TensorFlow binary is optimized with oneAPI Deep Neural Network Library (oneDNN) to use the following CPU instructions in performance-critical operations: AVX2 FMA
To enable them in other operations, rebuild TensorFlow with the appropriate compiler flags.
2022-12-27 05:52:34.214009: I tensorflow/core/platform/cpu_feature_guard.cc:193] This TensorFlow binary is optimized with oneAPI Deep Neural Network Library (oneDNN) to use the following CPU instructions in performance-critical operations: AVX2 FMA
To enable them in other operations, rebuild TensorFlow with the appropriate compiler flags.
* Serving Flask app 'app'
* Debug mode: on
WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead.
* Running on all addresses (0.0.0.0)
* Running on http://127.0.0.1:8080
* Running on http://172.31.42.12:8080
Press CTRL+C to quit
* Restarting with stat
2022-12-27 05:52:40.469947: I tensorflow/core/platform/cpu_feature_guard.cc:193] This TensorFlow binary is optimized with oneAPI Deep Neural Network Library (oneDNN) to use the following CPU instructions in performance-critical operations: AVX2 FMA
To enable them in other operations, rebuild TensorFlow with the appropriate compiler flags.
2022-12-27 05:52:42.662396: I tensorflow/core/platform/cpu_feature_guard.cc:193] This TensorFlow binary is optimized with oneAPI Deep Neural Network Library (oneDNN) to use the following CPU instructions in performance-critical operations: AVX2 FMA
To enable them in other operations, rebuild TensorFlow with the appropriate compiler flags.
* Debugger is active!
* Debugger PIN: 554-146-089
139.5.250.104 - - [27/Dec/2022 05:53:34] "GET / HTTP/1.1" 200 -
139.5.250.104 - - [27/Dec/2022 05:53:34] "GET /static/1.css HTTP/1.1" 200 -
139.5.250.104 - - [27/Dec/2022 05:53:34] "GET /favicon.ico HTTP/1.1" 404 -
139.5.250.104 - - [27/Dec/2022 05:53:37] "GET /new_user HTTP/1.1" 200 -
^Cubuntu@ip-172-31-42-12:~/urgent$
ubuntu@ip-172-31-42-12:~/urgent$ sudo poweroff
Connection to ec2-43-205-199-239.ap-south-1.compute.amazonaws.com closed by remote host.
Connection to ec2-43-205-199-239.ap-south-1.compute.amazonaws.com closed.
meet.google.com is sharing your screen.
Stop sharing
Hide


Search images
Home
Desktop
Downloads
Documents
Pictures
Music
Videos
COLLEGE VCE
Final
project
templates
kanuri - Personal
Desktop
Final
envir
project
images
static
templates
Your OneDrive is full
Storage options
Name
Status
Date modified
Radha Krishna
26-12-2022 11:43
.DS_Store
26-12-2022 11:32

5. CONCLUSION

5.1. Conclusions

Cloud Based Point of Sale System with Face Verification is successfully developed. It can bring down credit card frauds as no issuance of credit card would require. Further, we emphasize that it is a method to provide convenience to people and bring down frauds. Convenience is provided by means of not required to carry credit card, debit card or cash to the market for shopping. Use of FaceNet is suggested as part of solution for Face Recognition in our system. Cloud implementation would make it more reliable and energy efficient as well. We can conclude by saying that, if this system is implemented in real life will give huge benefit to society and humans by way of bringing convenience, security, energy power saving and
reducing various card-based frauds.

5.2 Limitations

• The system cannot identify if the person is wearing any face mask of the other person.
• System runs on the Internet Connection. If there is loss of connection system may not work.

5.3 Future Scope

• There is chance that a person might use a face mask and get his face verified.
• The project can be further be extended to recognize a face mask and raise a false alarm.

6. REFERENCE

[1] Arnab Dey, Sudhanshu Jain, Shovan Nandi (2019) "Automated POS System based on Face Recognition and Password", International Conference on Recent Advances in Energy-efficient Computing and Communication (ICRAECC).
[2] Florian Schroff, Dmitry Kalenichenko, James Philbin (2015) "FaceNet: A Unified Embedding for Face Recognition and Clustering" IEEE Conference on Computer Vision and Pattern Recognition (CVPR).
[3] Ramadhani, Anissa Lintang; Musa, Purnawarman; Wibowo, Eri Prasetyo (2017). "Human Face Recognition Application Using PCA and Eigenface Approach" [IEEE 2017 Second International Conference on Informatics and Computing (ICIC).
[4] Sharma S, Karthikeyan Shanmugasundaram, Sathees Kumar Ramasamyc (2016), "FAREC - CNN Based Efficient Face Recognition Technique using Dlib", International Conference on Advanced Communication Control and Computing Technologies (ICACCCT).
[5] Diajeng, Cindykia, Putri Winda, Dr. M.Arief Soeleman (2018), "Face Detection Using Haar Cascade in Difference Illumination" 8 International Seminar on Application for Technology of Information and Communication (iSemantic).
[6] Chu, Delin; Liao, Li-Zhi; Ng, Michael Kwok-Po; Wang, Xiaoyan (2015). "Incremental Linear Discriminant Analysis: A Fast Algorithm and Comparisons" IEEE Transactions on Neural Networks and Learning Systems.
[7] Hendrik, Irving Vitra Paputungan, Tomi Budi Susilo, Hari Setiaji, (2018) "Designing a Cloud-Based System for Small and Medium Enterprises with Multiple Branches", 3rd International Conference on Computer and Communication Systems.
[8] Adi Suryaputra Paramita, (2019) "Cloud Computing-Based Point-of-Sales Readiness for Surabaya's Small/Medium Enterprises" International Journal of Advanced Trends in Computer Science and Engineering.
[9] Kaipeng Zhang, Zhanpeng Zhang, Zhifeng Li and Yu Qiao (2016) "Joint Face Detection and Alignment Using Multitask Cascaded Convolutional Networks" IEEE Signal Processing Letters, VOL. 23, NO. 10.
[10] Nyein, Thida; Oo, Aung Nway (2019) "University Classroom Attendance System Using FaceNet and Support Vector Machine" IEEE International Conference on Advanced Information Technologies (ICAIT).
[11] Kennedy Okokpujie, Etinosa Noma-Osaghae, Olatunji Okesola, Osemwegie Omoruyi, Chinonso Okereke, Samuel John and Imhade P. Okokpujie (2018) "Fingerprint Biometric Authentication Based Point of Sale Terminal" Information Science and Applications 2018.
[12] Abikoye O. C, Afolabi, G. K, Aro, T. O (2019) "Biometric Based Point- of- Sale Authentication System" International Journal of Software Engineering and Computer Systems (IJSECS) ISSN: 2289-8522, Volume 5 Issue 1, pp. 36-51.