Smart ATM Service

Sayan Hazra
Department of Electrical Engineering
University Institute Of technology
The University Of Burdwan
Burdwan, India
hazrasayan201442@gmail.com

***Abstract*— Automated Teller Machine (ATM) is an electronic telecommunications device, which enables customers to perform banking without the need for direct interaction with bank staff. For this, every account holder must have a unique id card for the individual account having a unique pin. On the absence of this card, whatever be the adverse situation the use of this ATM service is not permitted. So, an Internet Of Things and Computer Vision based Smart ATM service is being proposed here, using Raspberrypi microcontroller based embedded system, where each person will be their own identity, where Fingerprint, Face, OTP verifications are key features for security, which in turn reduces the issue of fraud transactions, fraud ATM cards, hence security issue gets resolved.**

Keywords—ATM, IOT, Computer Vision, Raspberrypi, microcontroller, embedded system

I. INTRODUCTION

ATM service [1] stands as one of the most important facility, while it comes to cashless travel. In ATM service every account holder, have individual unique ATM card, provided from the corresponding bank. Using that ATM cards finite limits of transactions are to be permitted within a finite time interval, from any ATM. On the other hand, digitalization becomes very much easier while, having an ATM card, especially during online money transfer or online marketing facility. During ATM service the ATM card is needed to be inserted into the ATM machine along with the unique password or id for that ATM card, and then if the credentials of that ATM card gets matched with the server data, the transaction gets permitted. According to ATM industry association (ATMIA), there are around 3.5 million ATMs are installed worldwide. But, an issue is the execution of these ATM service is bound to the unique ATM card, if it is not there no permission will be granted, whatever be the adverse situation, ATM service will no more permitted, if it gets lost or is not present at that very moment. On the other hand, as more and more people are being globally digitalized, and as it is being so much used, there are being so many cases of having fraud ATM cards, fraud transactions ever since. In this era of global digitalization, ATM cards are linked with most of the digital wallets, account for achieving easiest transaction facility, along with online banking. So, somehow if the card details including a/c number, password, CVV code is known by some third party, it can be misused in thousands way. So, to get rid of these dreadful issues of security of this service, the efficient way out is to go for biometric security, which is proposed here throughout this model. In this present era of massive technological growth, Internet of things, Computer Vision are becoming cutting-edge technologies. Thus, the association of Internet Of Things(IOT) and Computer Vision with the ATM service makes the ATM service much more smart, advanced and user-friendly, too. Since the last few years some works has been done to reduce those problems. Transaction from ATM using mobile banking apart from using ATM is proposed in [4], in order to reduce time of transaction, but there might be security problem, if the system is compared to any biometric security. On the other hand, here the mobile is needed each time for ATM service, when this the proposed model deals with only the user, when it comes to account holder transaction, which is more advantageous than this. The research work described Secured pin authentication (SEPIA) as a service for ATM using mobile, and wearable, in [7] does moreover same as described before, in addition to that a new "mcard" is introduced there to avail m-payment facility, which is mobile banking facility using the same card instead of using ATM card. On the other hand, some facilities like checking authentication from any mobile or wearable devices, to prove co-location with cloud based server and to generate a secure pin for banking. It is much advantageous, but not secured and here, the service becomes bound to a card, which is less secured and advantageous too. In the research work [10] a concept of using multipurpose smart SIM card is proposed, which consists of mobile and all the other available facility along with the ATM service. The issue of this proposed model is implementation, because the ATM services can be changes throughout, if it comes to insecurity and need of advancement, but according to proposed model in [10] a set of new SIM card is needed that avails all the facilities of smart cards into one, here all the SIM cards are needed to be changes, which in turn will need to change a whole system, which is not possible at all, as there are more easier ways out like using biometrics, which is being proposed here in this model. In the paper [9], a new way of ATM service is proposed where, it is aimed to connect all the ATMs using IPv6, where a Near-Field Communication (NFC) is proposed, which would communicate via NFC enabled mobile phone with the ATMs, after inserting the ATM. The process is secured but time taking and bounded to have ATM cards and particular mobile having such facilities. In[11] RFID, GSM, GPS based smart ATM card is proposed for security and authentication, which is not cost effective, and main issue is the model is still that card bounded, and still less secured than biometric ones. In [12] fingerprint based biometric authentication is proposed, which is a great move to achieve to goal of improving security of ATM service, thus advantageous. But, no system is error free, and fingerprint checking is one of those efficient ways, but it can also be bypassed. In addition to that, it can be said that, fingerprint authentication is nothing but a sensor based authentication, like the way a blind person tries to execute perception related tasks, though it's efficient, great but incomplete. So, if vision and sensing are added to the system, any security system gets fulfilled. That is the reason, why here two major efficient biometric facilities are put together, which is no more bound to ATM cards while transaction by a/c holders, and much secured as the account holder is the only one gateway of transaction.

978-1-5386-6722-4/19/$31.00 ©2019 IEEE

II. System Architecture and Implementation

In this proposed model, no ATM card is needed for transaction, so for security purpose, face-recognition along with the aliveness checkup of face, fingerprint verification, OTP (one time password) verification security checks have been taken into consideration. A brief of face recognition concept will be described here.

A. Face Recognition

In this proposed architecture face-recognition plays an important role. Face-recognition is one of most advanced biometric technology, where a digital image of an individual customer will be taken, and compared to pre-trained system images in a database. The face-recognition algorithm in this proposed system, consists of few major sections-
1. Face-detection and building the dataset, 2. Building a recognition model, 3. Recognize faces in live video feed

All these sections will be explained using the block diagram shown in Fig. 1.

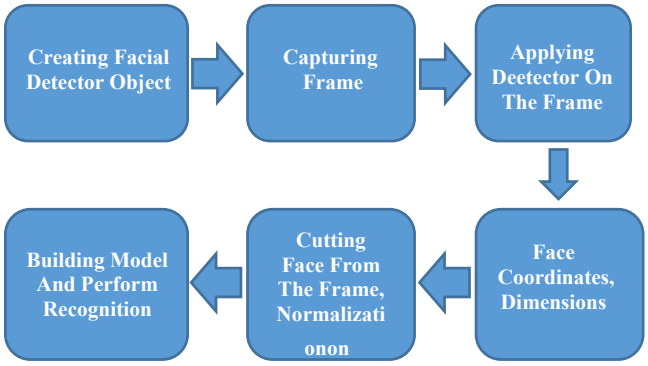

Fig. 1. Block diagram of the workflow of face recognition

1. Face-detection:

In this proposed system, face-detection is done by using Haar-featured based Cascade classifiers, as proposed by Paul Viola and Michole-Jones, in "rapid Object Detection" [2], which is consisted of Haar features extraction: where different square shaped shapes are traversed throughout the images or frames, to analyze the frame. Each of those squares, which can be rotated or extended in all possible ways during traversal, consists of two sections black and white. In this entire process of traversal feature extraction is done by finding difference between overall pixel intensity between black and white regions of the squared shapes, like:
Σ White intensity- Σ Black intensity= Output,
high output is obtained when regions are similar. In haar feature extraction edge, line and center surrounded features are extracted.
In haar feature extraction method, some limitations are observed, so Integral image and Adaboost methods are used. In this **Adaboost** method from linear combination of various weak classifiers, a strong classifiers are obtained from the feature detection process, which is defined by:

$$F(x) = \alpha_1 f_1(x) + \alpha_2 f_2(x) + \ldots + \alpha_n f_n(x),$$

where f(x), F(x) are weak classifier and strong classifier, α_i is the weight corresponds to i th weak classifier, the bigger the weight, the more relevant the feature will be. Depending upon the value of these weights the classifiers are ranked.

In **Integral image**, each of the pixel intensity is replaced by the sum of all the pixel intensities around it. Then comes the one of most efficient step, called cascading.

Cascading, consists of traversing the input image through various separated classifiers, which actually finds the facial-object from the image, features are distinguished by the classifiers, after traversing through all the classifiers the face is finally detected inside the image or frame.
The demonstration of the proposed model has implemented using **python**, and **OpenCV [3]** computer vision tool for executing image manipulation, facial recognition. Various pre-trained models are there in OpenCV.
"haarcascade_frontalface_alt", is such pre-trained model for face detection, is used in this system for face-detection.
"haarcascade_frontalface_alt", is such pre-trained model for face detection, is used in this system for face-detection.
The algorithm of face-detection is represented by the block diagram in Fig.1. According to the block diagram shown in the figure, firstly the **CascadeClassifier,** which creates the classifier is called and thereafter, the Facial detector object is created using the haarcascade frontal face standard face detection model.
Then the image or frames are capured containg faces and, detector is applied on each of the frames. While applying detector objects over the frames, some parameters are passed, which includes

scale-factor: To detect faces inside image, the detector first takes an area and applies classifier to detect faces inside that, if not then it increases or reduces the area by the scale_factor times of the previous area, here scale_factor is taken as 1.2.

min_neighbors: It signifies, no. of neighboring points in the image is to be flagged positive to detect the face.

min_size: this is in the format (width, heights) which signifies the minimum size of the face that can be detected inside the frame.

flag: it means the proerty of the images, which signifies how many features are atleast needed to be satisfied in oder to detect the faces inside the frame.

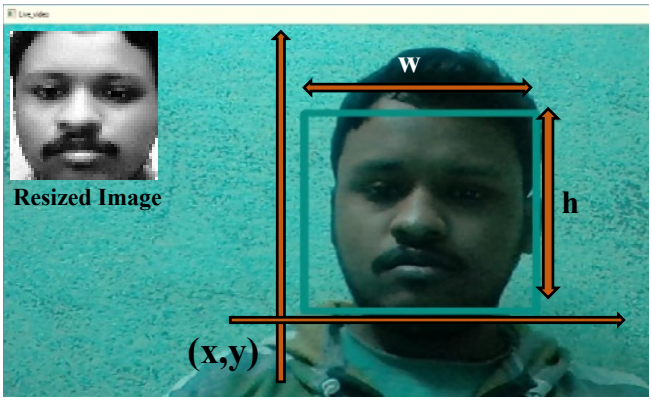

Fig. 2. Demo of the detected face along with the resized face before

After applying the detector, a numpy array is obtained which, consists of (x,y), i.e. the coordinate of the left bottom corner of the rectangular region, which covers the face inside the frame, along with the width(w) and height(h) of the rectangle. After applying the concepts, face is detected and, using the obtained parmeters regarding the position of face inside frame, rectangle is drawn around the faces as demonstrated in the Fig. 2.

2. Building a recognition model:

After the face in a frame gets detected, the face is cut from the frame. The dimension of the region or area which is cut from the frame, that contains the face is

height=h, and
width=w-(2*(w_rm))

Where, **w_rm=absolute value of (0.2*w/2).** So, w_rm is the portion of width deducted from the actual obtained width by the detector, as it can be seen that, some of the width is noisy, so to remove this portion, width gets modulated. After the face is cut, it is first converted into grey-scale image, there after the face is **normalized**, by which, the contrast of the face gets enhanced in the complete greyscale spectrum range, which is between 0 to 255.After the grey scale facial image gets normalized, the face is **resized** to a finite size having standard pixel dimension, here it is (50, 50). The demo of this resized form of a facial image is shown in the Fig. 2. Now, in this way final resized facial data from various consecutive is stored in a particular folder to construct a recognizer model. Workflow of recognization section

Resized dataset→ Pass the recognizer algorithm→ Train the system→ Obtain frame from live video→ apply reconizer→ Check prediction result→ Show the status of recognition

After obtaining the resized facial image set, to build the recognizer model, **Local Binary Pattern Histogram (LBPH)** algorithm, as proposed in **[6]** is used here. In this lbph algorithm, the whole image is not treated at once, it takes a small section of the image, where pixel at the middle becomes the threshold to the neighbors, if the neighboring pixel greater than threshold, then it gets replaced by 1 else replaced by zero, by doing this, a certain local binary pattern on the image gets formed. Then, spatial information are incorporated by lbph. So, this way on applying the lbph over the resized dataset of particular person, face recognizer model of the dataset of that particular person gets built, and trained at the same time. Final step is recognition.

3. Recognize faces in live video feed:

In the previous way, an unknown detected face from a frame is detected, normalized, resized and then gray scale resized image is passed through a predictor, which uses the pre-trained saved recognized model of saved database for checking with the input image, and returns the prediction value, if the prediction value along with the corresponding dataset name with which it's prediction is close, of the model less than the pre-calibrated threshold, then face got successfully recognized with the corresponding dataset of stored database. In Fig. 3. A demo of such recognized face is shown in python live video window.

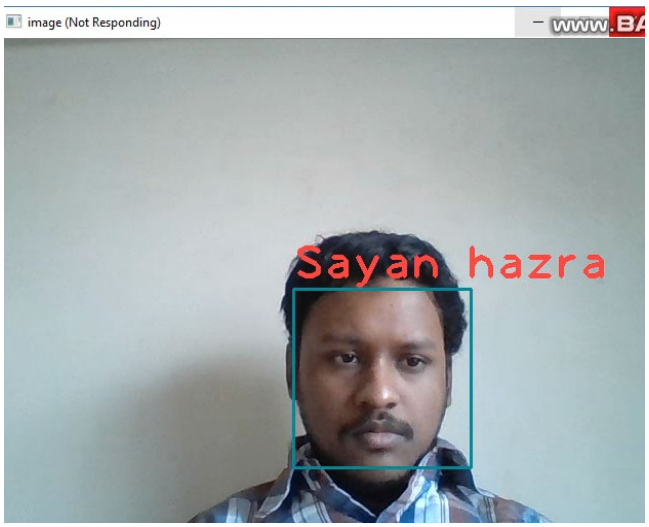

Fig. 3. Demo of the final recognized face from live video

After a face got recognized it is needed to validate to aliveness of that face to distinguish between actual face and image of that face. In order to check aliveness, smile feature is needed to be deteceted on the recognized face, when face got recogized, user will be requested to smile for aliveness checkup.

B. Aliveness Check

After the face gets recognized, to check the aliveness of the face, further smile is detected on the cropped resized facial grey scale image. To **detect smile, haar cascade based classifier,** whose concept was proposed in [5], is constructed using **"haarcascade_smile.xml"**, a pre-trained model, which returns a numpy array consisting location of the smiled portion inside the facial image, if array returns greater than zero, then smile detected on the face and aliveness.
Note: In order to check aliveness of a recognized face, both recognition function and aliveness function should be run in a thread.

C. System Setup

Fig. 4 shows the schematic diagram of the system setup. In this system, **Raspberrypi 3** is used as a microcontroller as shown in the figure. Keypad is connected to enter input from the user during ATM service, fingerprint sensor is used to check the fingerprint of users and operation result will be displayed on the display screen. **Raspberrypi camera module**, which having good light sensitivity, is used to face-recognition. **Global System for Mobile Communication (GSM)** module is used here to send the required message to the user phone number, whether it may be one time password (OTP) or any other confirmation message. **L293d motor driver module** is used to operate the motor setup of the cash transaction slit, using digital output from the analog to digital converter chip depending upon the instruction from Raspberrypi.

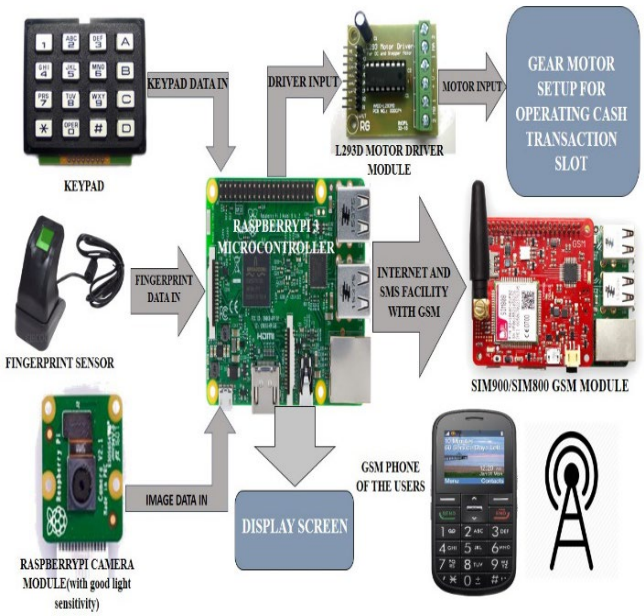

Fig. 4. Schematic of the system setup

D. *Banking Process*

Banking process is consisted of three categories, these are

1. Banking by account holder: When account holder selects the account holder banking option, then the workflow is explained by the block diagram shown in Fig. 5. If both fingerprint and face got recognized, user can continue the transaction, if anything wrong happens, a message of wrong attempt will be sent to the registerer mobile number of the account holder and the photograph will be stored in the cloud database of bank, if user wants to see the photograph and other details of wrong entry, then details of wrong entry along with photograph can be seen after logging in to the user account.

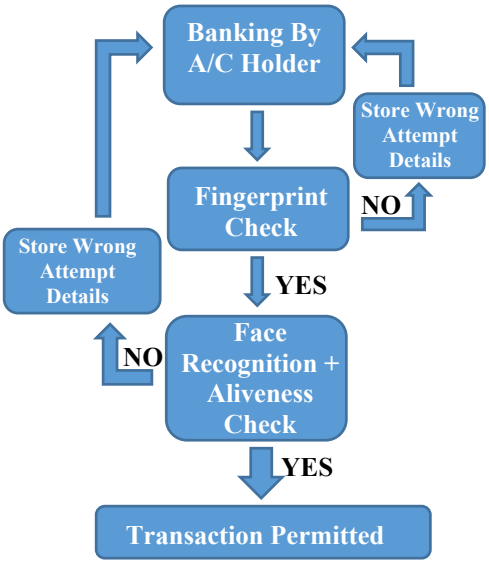

Fig. 5. Block diagram of workflow of a/c holder banking

Fig. 6 shows the demo of account holder banking transaction on the python shell output window.

```
Python 3.6.4 (v3.6.4:d48eceb, Dec 19 2017, 06:54:40) [MSC v.19
Type "copyright", "credits" or "license()" for more informatio
>>>
========== RESTART: D:\7th sem\final seminar\main code.py
WELCOME TO 24 HOURS ATM SERVICE
1.ENTER 1 FOR USER BANKING,
2.ENTER 2 FOR PROVIDING SUB-USER DETAILS,
3.ENTER 3 FOR BANKING BY SUB-USER
1
AS A SUBSTITUTE OF FINGERPRINT WE ARE TAKING ACCOUNT NUMBER:
79801504211234
['ATANU PAL', 98765432112345.0, 'ATUL']
['NABARUN SENGUPTA', 12345678912345.0, 'NABA']
['SAYAN HAZRA', 79801504211234.0, 'Sayan0506']
PROCEED
LOOK AT THE CAMERA FOR FACE RECOGNITION:
face count 1
LBPH faces:->Sayan hazra124.94813873724755 1
YOUR CREDENTIALS ARE
text:'SAYAN HAZRA'
number:79801504211234.0
YOU CAN CONTINUE YOUR TRASACTION
THANKS FOR BANKING!
1.ENTER 1 FOR USER BANKING,
2.ENTER 2 FOR PROVIDING SUB-USER DETAILS,
3.ENTER 3 FOR BANKING BY SUB-USER
```

Fig. 6. Demo of account holder banking on python shell

2. Banking by sub-user: Sub-user is addressed here, to one who is apart from account holder, who needs to conduct transaction from account. Here, to do such

2.1 The first step is sub-user entry, which will be explained using the block diagram as shown in Fig. 6. After logging in into the registered account of the account holder created online using OTP verification, the account holder needed to enter the name of sub-user, their phone number, and the max amount that an individual sub-user can transact. Max, three sub-user entry from the account per day is permitted, and the sub-user details will be valid for only 24 hours, after which these details will be reset to null.

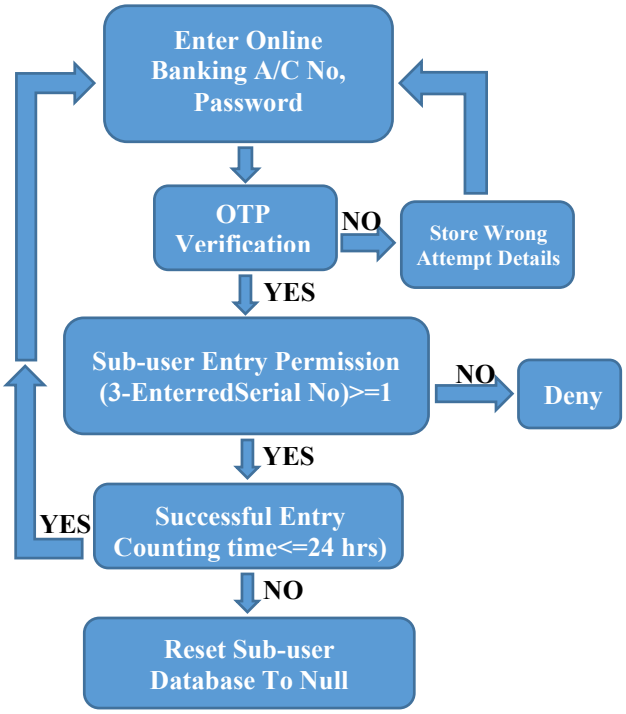

Fig. 6. Block diagram of workflow of sub-user entry by the a/c holder

2.2 Sub-user banking workflow is shown in Fig. 7. During banking by sub-user, sub-user first needed to ask the account number, sub-user details consisting name, phone number and amount to be transacted. If phone number matched with corresponding sub-user details in the database, and transaction amount if less than the calculated transaction limit of that sub-user, a random OTP will be sent to the entered phone number using GSM by Rasppberrypi. Within a finite

time from then, if the OTP is entered by the sub-user correctly, then the transaction by sub-user is permitted.

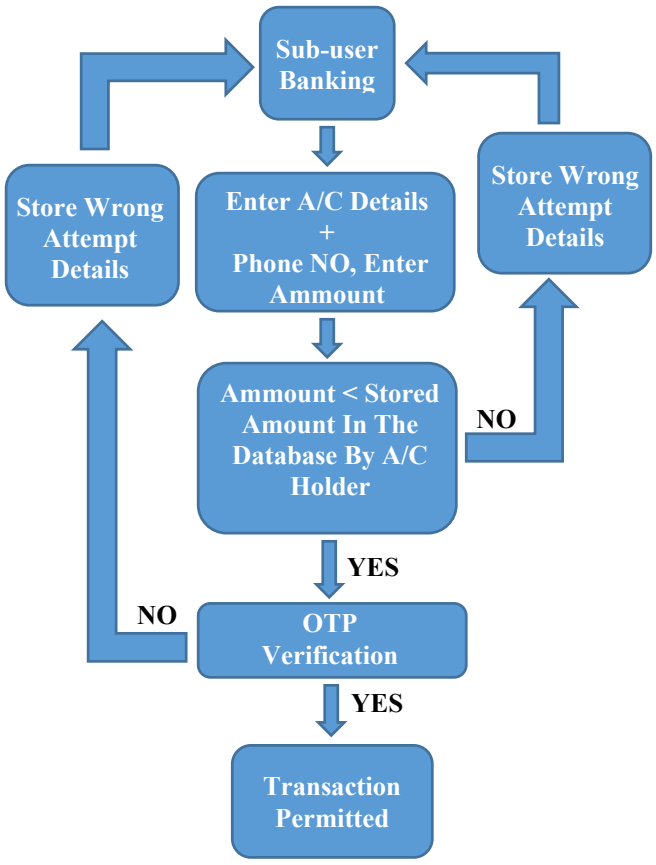

Fig. 7. Block diagram of workflow of sub-user banking

The demo of this sub-user banking along with sub-user details entry in python shell window is shown in Fig. 8.
The Fig. 9 shows the screenshot of OTP sent from the system to the sub-user phone.

```
WELCOME TO 24 HOURS ATM SERVICE
1.ENTER 1 FOR USER BANKING,
2.ENTER 2 FOR PROVIDING SUB-USER DETAILS,
3.ENTER 3 FOR BANKING BY SUB-USER
2
ENTER THE A/C PASSWORD GIVEN FROM THE BANK!
Sayan0506
YOUR CREDENTIALS ARE AS FOLLOWS
A/C HOLDER NAME: text:'SAYAN HAZRA'
A/C NUMBER: number:79801504211234.0
ENTER NUMBER OF SUB-USERS(IT MUST BE WITHIN 3 PER DAY)
2
ENTER SUB-USER NAME
TANDRA HAZRA
ENTER SUB-USER CONTACT NUMBER
7980150421
ENTER MAX AMMOUNT TO BE TRNSACTED
500
ENTER SUB-USER NAME
BIKASH HAZRA
ENTER SUB-USER CONTACT NUMBER
9831535142
ENTER MAX AMMOUNT TO BE TRNSACTED
500
SUB-USER ENTRY SUCCESSFUL
['TANDRA HAZRA', 'number:79801504211234.0', '7980150421', '500']
['BIKASH HAZRA', 'number:79801504211234.0', '9831535142', '500']
THANKS FOR BANKING!
1.ENTER 1 FOR USER BANKING,
2.ENTER 2 FOR PROVIDING SUB-USER DETAILS,
3.ENTER 3 FOR BANKING BY SUB-USER
3
ENTER YOUR NAME IN CAPS!
TANDRA HAZRA
ENTER YOUR REGISTERED PH_NO
7980150421
ENTER A/C NO
79801504211234
HOW MUCH YOU WANT TO TRANSACT!
500
{"return":true,"request_id":"aodmtqvx8413hr6","message":["Messag
ENTER THE OTP SENT TO THE GIVEN MOBILE NUMBER
7434
CONTINUE THE TRANSACTION
```

Fig. 8. Demo of the sub-user banking in the python shell output window

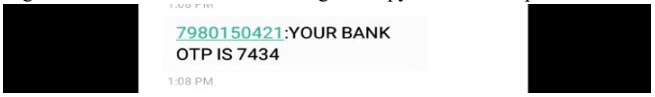

Fig. 9. Screenshot from the sub-user mobile after receiving the OTP

III. Conclusion

It can be concluded that, this article provides a solution for card less more secured less time taking user-friendly cash transaction service where face-recognition along with aliveness checkup is used for security purpose followed by fingerprint checkup and for sub-user transaction can also be conducted on the basis of stored information by account holder, where OTP verification has included for validity of sub-users. The demonstration result of working model verifies the proposed smart secured efficient fast ATM service concept.

IV. Future Development

In this proposed model of ATM service, for security haar-cascade classifier based face recognition along with smile detection has been proposed, in future rather neural network can be incorporated to increase the accuracy of the face recognition system. Again, 3-D biometric face recognition can also be incorporated in future, which will be much more efficient for security checkup. The system should be properly maintained. Some special classifiers or algorithms must be included with the system in order to detect mobbing activity or weapon carried by user in the ATM service room.

References

[1] V. Cuervo, “Automated teller machine dispenser of debit cards,” U.S. Patent 6,105,009, August 2000

[2] P. Viola and M. Jones, “Rapid Object Detection using a Boosted Cascade of Simple Features,” Proc. IEEE Comp. Soc. Conf. USA, vol. 1, pp. 1-1, December 2001

[3] G. Bradski and A. Kaehler, ”Learning OpenCV: Computer vision with the OpenCV library,” O’Reilly Med. Inc. USA, 2008

[4] N. Bansal and N. Singla, “Cash withdrawl from ATM machine using Mobile banking,” Int. Conf. Computational The. Inform. And Communication Tech. (ICCTICT) India, pp. 535-539, March 2016

[5] J. Whitehill, G. Littlewort, I. Fasel, M. Bartlett and J. Movellan, “Toward Practical Smile Detection,” IEEE Trans. Pattern Analysis and Intelligence IEEE Comp. Soc., vol. 31, pp. 2106-2111, November 2009

[6] T. Ahonen, A. Hadid and M. Pietikainen, “Face Description with Local Binary Patterns: Application to Face Recognition,” IEEE Trans. Pattern Analysis and Machine Intelligence IEEE Comp. Soc., vol. 28, pp. 2037-2041, December 2006

[7] R. Khan, R. hasan, J. Xu, “SEPIA: Secure-PIN-Authentication-as-a-Service for ATM Using Mobile and Wearable Devices,” IEEE 3rd Int. Conf. Mobile Cloud Computing, Services, and Engg. , pp. 41-50, March 2015

[8] M. S. Uddin, N. C. Das and A. Barua, “The mCard approach for Bangladesh: A smart phone based Credit/Debit/ATM card,” 16th Int. Conf. Computer and Inform. Tech. Bangladesh, pp. 209-212, March 2014

[9] S. Sridharan and K. Malladi, “New generation ATM terminal services,” Int. Conf. Computer Communication and Inform. (ICCCI) India, pp. 1-6, January 2016

[10] H. R. F. Najafabadi and M. R. F. Derakhshi, “Multipurpose smart SIM card based on mobile database and location dependent query,” 6th Int. Conf. Application Inform. and Communication Tech. (AICT) Georgia, pp. 1-5, October 2012

[11] Nelligani, B. M. Reddy, NV U. reddy and N. Awasti, “Smart ATM security system using FPR, GSM, GPS”, Int. Conf. Inventive Computation Tech.(ICICT) India, vol. 3, pp. 1-5, August 2016

[12] Christiawan, B. A. Sahar, A. F. Rahardian, and E. Muchtar, “Fingershield ATM – ATM Security System using Fingerprint Authentication,” Int. Symposium Electronics and Smart Devices (ISESD) Indonesia, January 2019.