International Journal of Research
Available at https://journals.pen2print.org/index.php/ijr/
e-ISSN: 2348-6848
p-ISSN: 2348-795X
Volume 07 Issue 06
June 2020

Face-Recognition-Based-Atm-System

Eediga Raghavendra Goud[1], K.Charan Teja[2]
[1,2] Departemnt: Computer Science and Engineering
[1]Research, [2]Professor & Hod Mtech
[1,2] Geethanjali College Of Engineering and Technology ,Nannur, Kurnool
[1]Email:- raghavendragoud777@gmail.com,

Abstract:

Now-a-days, Security plays an important role in day-to-day life. In an ATM, transaction needs to takes place in a secure way. Hence, I shall introduce a Face Recognition method. Face recognition plays an important role in many fields such as to authenticate users, security issues etc., It mainly plays a significant role in real time surveillance systems. By introducing this method, I can help the user to have a secure transaction. So, I shall develop a face recognition-based system over the simple card system.

- The objective of my project yields good results and helps to improve security.
- To make it understandable, I am creating a framework using OpenCV, Tkinker and creating aGUI.

Keywords:- Face Recognition, ATM, Principal Component Analysis

INTRODUCTION

Face recognition plays an important role in the numerous fields such as to authenticate the users, security of homeland, smart home access security, identify the criminal, identifying the user in small scale applications. It is mainly used for authentication purpose. Recent technology development enables the IoT system generates the real-time data which need to process and analyses the data instantaneously.

The detection of object in an image is related to computer technologies and image processing. The main objective of face recognition is to reduce the mis-categorization rate. Face recognition is used in many areas which need more rigorous computation to carry out the face detection. The Haar cascade classifier used to detect the face using Open CV.

Face recognition is such a challenging yet interesting problem that it has attracted researchers from different backgrounds. It is due to this fact that the literature on face recognition is vast and diverse. Face recognition is used in different applications to identify the faces in the digital image or video frames. It matches the image of person bit by bit. Recently, the face recognition used in the education field to detect, analyses and process the captured image(emotion) to measure the positive effects of teaching such as understanding, insight, and sensations. This method comprises of three phases such as feature extraction, subset feature and emotion classifier. The basic features are extracted using Haar Cascades features value.

The aim of all the researches was to make face recognition as automated and accurate as possible through various types of inputs such as static images, video clips, etc. so as to increase its applications in real world. Computational methods of face recognition need to address numerous challenges. These types of difficulties appear because faces need to be represented in such a way that best utilizes the available face information to define a specific face from all the other faces in the database. Also, extracting such detailed facial features can be used in

slandering the search and enhancing recognition.

LITERATURESURVEY

Some subject need's practical assessment. Education deals with more practical knowledge. The students can gain theoretical knowledge by knowing concepts and algorithms but that is not the criteria. The traditional method of teaching does not suit best. To make it more practical and understand the concepts in a better way this problem statement is identified.

AT&T [87] (formerly ORL)

Contains face images of 40 persons, with 10 images of each. For most subjects, the 10 images were shot at different times and with different lighting conditions, but always against a dark background.

Oulu Physics [88]

Includes frontal color images of 125 different faces. Each face was photographed 16 times, using 1 of 4 different illuminants (horizon, incandescent, fluorescent, and daylight) in combination with 1 of 4 different camera calibrations (color balance settings). The images were captured under dark room conditions, and a gray screen was placed behind the participant. The spectral reflectance (over the range from 400 nm to 700 nm) was measured at the forehead, left cheek, and right cheek of each person with a spectrophotometer. The spectral sensitivities of the R, G and B channels of
The camera, and the spectral power of the four illuminants were also recorded over the same spectral range.

XM2VTS [89]

Consists of 1000 GBytes of video sequences and speech recordings taken of 295 subjects at one-month intervals over a period of 4 months (4 recording sessions). Significant variability in appearance of clients (such as changes of hairstyle, facial hair, shape and presence or absence of glasses) is present in the recordings. During each of the 4 sessions a speech" video sequence and a "head rotation" video sequence were captured. This database is designed to test systems designed to do multimodal (video + audio) identification of humans by facial and voice features.

Yale [90]

Contains frontal grayscale face images of 15 people, with 11 face images of each subject, giving a total of 165 images. Lighting variations include left-light, centre-light, and right-light. Spectacle variations include with-glasses and without-glasses. Facial expression variations include normal, happy, sad, sleepy, surprised, and wink

Yale B [91]

Contains grayscale images of 10 subjects with 64 different lighting angles and 9 different poses angles, for a total of 760 images. Pose 0 is a frontal view, in which the subject directs his/her gaze directly into the camera lens. In poses 1,2, 3, 4, and 5 the subject is gazing at 5 points on a semicircle about 12 degrees away from the camera lens, in the left visual field. In poses 6, 7, and 8 the subject is gazing at 3 different points on a semicircle about 24 degrees away from the camera lens, again in the left visual field. The images were captured with an overhead lighting structure which was fitted with 64 computer-controlled xenon strobe lights. For each pose, 64 images were captured of each subject at a rate of 30 frames/sec, over a period of about 2 seconds.

MIT [92]

Contains 16 subjects. Each subject sat on a couch and was photographed 27 times, while varying head orientation. The lighting direction and the camera zoom were also varied during the sequence. The resulting 480 x 512 grayscale images were then filtered and sub sampled by factors of 2, to produce six levels of a binary Gaussian pyramid. The six pyramid levels" are annotated by an X-by-Y pixel count, which ranged from 480x512 down to 15x16.

CMU Pose, Illumination, and Expression (PIE) [93]
Contains images of 68 subjects that were captured with 13 different poses, 43 different illumination conditions, and 4 different facial expressions, for a total of 41,368 color images with a resolution of 640 x 486. Two sets of images were captured – one set with ambient lighting present, and another set with ambient lighting absent.

UMIST [94]
Consists of 564 grayscale images of 20 people of both sexes and various races. (Image size is about 220 x 220.) Various pose angles of each person are provided, ranging from profile to frontal views.

Bern University face database [63]
Contains frontal views of 30 people. Each person has 10 gray-level images with different head pose variations (two front parallel pose, two looking to the right, two looking to the left, two looking downwards, and two looking upwards). All images are taken under controlled/ideal conditions.

Purdue AR [64]
Contains over 4,000 color frontal view images of 126 people's faces (70 men and 56 women) that were taken during two different sessions separated by 14 days. Similar pictures were taken during the two sessions. No restrictions on clothing, eyeglasses, make-up, or hair style were imposed upon the participants. Controlled variations include facial expressions (neutral, smile, anger, and screaming), illumination (left light on, right light on, all side lights on), and partial facial occlusions (sun glasses or a scarf).

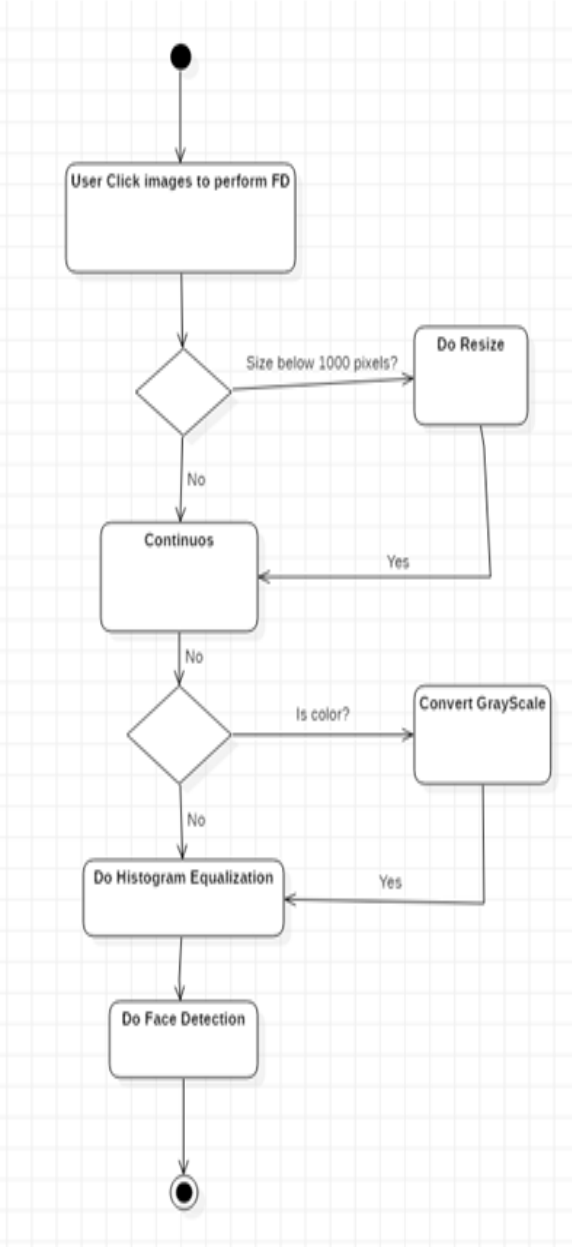
The University of Stirling online database [95]
It was created for use in psychology research, and contains pictures of faces, objects, drawings, textures, and natural scenes. A web-based retrieval system allows a user to select from among the 1591 face images of over 300 subjects
Based on several parameters, including male, female, grayscale, color, profile view, frontal view, or 3/4 view.

The FERET [96]
Contains face images of over 1000 people. It was created by the FERET program, which ran from 1993 through 1997. The database was assembled to support government monitored testing and evaluation of face recognition algorithms using standardized tests and procedures. The final set of images consists of 14051 grayscale images of human heads with views that include frontal views, left and right profile views, and quarter left and right views. It contains many images of the same people taken with time-gaps of one year or more, so that some facial features have changed. This is important for evaluating the robustness of face recognition algorithms over Tim.

Kuwait University face database (KUFDB)[97]
The in-house built database consists of 250 face acquired from 50 people with five images per face. There is a total 250 gray level images (5 images x 50

people). Facial images are normalized to sizes 24 x 24, 32 x 32, and 64 x 64). Images were acquired without any control of the laboratory illumination. Variations in lighting, facial expression, size, and rotation, are considered.

ACTIVITY DIAGRAM



SYSTEMSTUDY

It is always necessary to study and recognize the problems of the existing system, which will help in finding out the requirements for the new system. System study helps in finding different alternatives for better solution. The project study basically deals with different operations and steps involved in it.First, I need to open Python-IDE and select the location of the project and choose the file in which code is present. Then, Run the file, I will get a framework where I need to enter ID, Name of the user and click on “upload Images” and then after successful uploading of the images, I will get a display on the screen stating that , Images are Uploaded along with the Id and Name of the person I previously entered. Then click on “Train Images”, after training of images is done, I will get a display on the page that “Images are trained”. Then click on “Verify Images”, once the identified images are verified against the pre-loaded images by the user, I will be able to have a look over the person’s login date and time. Once I was done with the usage of theframeworkI can close the framework by clicking on “Quit” button or by clicking letter “Q” on the keyboard.

PCA APPROACH:

“Fast face recognition using eigen faces” their approach signifies face recognition as a two-dimensional problem. In this approach, face reorganization is done by Principal Component Analysis (PCA). Face images are faced onto a space that encodes best difference among known face images. The face space is created by eigen face methods which are eigenvectors of the set of faces, which may not link to general facial features such as eyes, nose, and lips. The eigenface method uses the PCA for recognition of the images. The system performs by facing pre-extracted face image onto a set of face space that shows significant difference among known face images. Face will be categorized as known or unknown face after imitating it with the present database. From the obtained results,

it was concluded that, for recognition, it is sufficient to take about 10% eigenfaces with the highest eigenvalues. It is also clear that the recognition rate increases with the number of training images.

The information which defines the face is represented as Eigen functions and the faces are referred as Eigen faces. Recognition system is implemented using these Eigen faces with principal component analysis (PCA). PCA is used to recognize the faces in the image. It involves the calculation of the Eigen value decomposition of a data covariance matrix. It compresses a set of high dimensional vectors into a set of lower dimensional vectors and reconstructs the original sets

EXISTINGSYSTEM:

In the existing system, I have a facility of using the general PIN option in order to authenticate the user before making access to his/her account. Customers can be recognizedby inserting their assigned cards by the bank into the machine, where if the authentication is satisfied by the pin entered by the customer, which must match the PIN stored in the chip on the card. It is possible that if some hackers know or steal the password and they can make use of the general password to perform the transactions without the knowledge of the owner of the account. Thereby security through this general password is not up to the mark that means I cannot ensure maximum security through the general password mode of transaction

PROPOSED SYSTEM:

In the proposed system, I have created a framework using tinker for recognizing face. In this framework I will take the details of the user such as Id, name. Then the user images are uploaded into the system and images are trained for further recognition. Then the images are verified against the user, if the authentication is accessed then it will return a message that “Login Success”. I will develop this project in Python.

The face detector detects the face by means of testing each part of the image. In face recognition, the first stage is face detection which has more challenges to recognize the face due to different features in face such as skin texture, color, shape of the face, wearing glasses, and moustache/beard. The photographic environment also affects the occurrence of digital image such as pose of head, facial expression, and lighting conditions. Furthermore, the face detection algorithm used for detecting the other objects such as vehicle number plat identification, pedestrians etc.

PROPOSED SYSTEM:

In the proposed system, I have created a framework using tinker for recognizing face. In this framework I will take the details of the user such as Id, name. Then the user images are uploaded into the system and images are trained for further recognition. Then the images are verified against the user, if the authentication is accessed then it will return a message that “Login Success”. I will develop this project in Python.

The face detector detects the face by means of testing each part of the image. In face recognition, the first stage is face detection which has more challenges to recognize the face due to different features in face such as skin texture, color, shape of the face, wearing glasses, and moustache/beard. The photographic environment also affects the occurrence of digital image such as pose of head, facial expression, and lighting conditions. Furthermore, the face detection algorithm used for detecting the other

objects such as vehicle number plat identification, pedestrians etc.

Testing

Testing is the process of finding differences between the expected behavior specified by system models and the observed behavior of the system.

They are two types of testing. They are:

1. White box testing.
2. Black box testing.

1.1. White box testing:

White box testing focuses on internal structure of the component.

- In the white box testing the designer can derive test cases that guarantee that all independent paths within a module have been exercised at least once.
- Exercise all logical decision on their true and false sides.
- Exercise all loops at their boundaries and within their operational bounds.
- Exercise internal data structures to ensure their validity.

Black box testing:

Black box testing focus on input or output behavior of the component. That is for a swinger the black box testing will give the desired results for what it is meant for, it means that he has to exercise all functional requirements for a program. The black box testing is done after the software is developed.

Black box testing attempts to find errors in the following:

- Incorrect or missing functions.
- Interface errors.
- Errors in data structures or external database access.
- Performance errors.
- Initialization and termination errors.

Testing approach:

Testing is the process of detecting errors. Testing performs a very critical role for quality assurance and for ensuring the reliability of software. The results of testing are used later on during maintenance also.

Psychology of Testing:

The aim of testing is often to demonstrate that a program works by showing that it has no errors. The basic purpose of testing phase is to detect the errors that may be present in the program. Hence one should not start testingwiththeintentofshowingthataprogramworks,buttheintentshouldbetoshowthataprogramdoesn't work.

Testing is the process of executing a program with the intent of finding errors.

Testing Objectives:

The main objective of testing is to uncover a host of errors, systematically and with minimum effort and time. Stating formally, I can say, Testing is a process of executing a program with the intent of finding an error.

- A successful test is one that uncovers an as yetun discovered error.
- A good test case is one that has a high probability of finding error, if it exists.
- The tests are inadequate to detect possibly present errors.
- The software more or less confirms to the quality and reliable standards

Test cases

International Journal of Research
Available at https://journals.pen2print.org/index.php/ijr/
e-ISSN: 2348-6848
p-ISSN: 2348-795X
Volume 07 Issue 06
June 2020

ID	TC01
TITLE	User Log-In
PREREQUISTE	Username ID
TEST ACTION	Takes the username and ID
EXPECTED RESULT	After taking username and ID, it stores the data in an excel sheet.

ID	TC02
TITLE	Upload Images
PREREQUISTE	Username ID
TEST ACTION	A Face Recognizer (Camera) opens takes images of the user in perfect lighting conditions
EXPECTED RESULT	Upon clicking Upload Images, Camera opens, and user images are uploaded into data base. After taking images, the camera closes automatically

ID	TC03
TITLE	Train Images
PREREQUISTE	Username ID Upload Images
TEST ACTION	Images will be trained
EXPECTED RESULT	Upon Clicking Train Images, Images will be trained for recognition

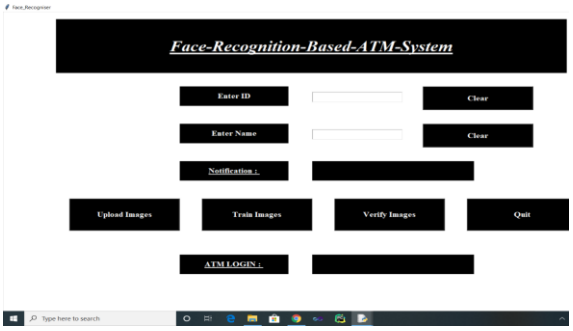
ID	TC04
TITLE	Verify Images
PREREQUISTE	User name ID Train Images
TEST ACTION	A Face Recognizer opens and the images are verified against the stored images in the training set and after recognizing image, the corresponding name, ID will be appeared on the screen. It returns a message "Login Success"
EXPECTED RESULT	Upon clicking Verify Images, the ID and name of the user are returned

ID	TC05
TITLE	Verify Images
PREREQUISTE	-
TEST ACTION	If the user is already registered then Id, name are printed on the screen. It returns a message "Login Success"
EXPECTED RESULT	Upon Clicking Verify Images directly, it gives the name, id of the user, if already registered.

ID	TC06
TITLE	Verify Images
PREREQUISTE	-
TEST ACTION	"Unknown" is printed, if the user has not registered before
EXPECTED RESULT	Upon clicking Verify Images, "Unknown" is printed
ID	TC07
TITLE	Quit (Q)
PREREQUISTE	-
TEST ACTION	The frame work stops the action
EXPECTED RESULT	The frame work stops the action it is being performing

Test Results:

TC01			
ITERATION	EXPECTED RESULT	ACTUAL RESULT	STATUS
1	On Log In, it should store the entered details in an excel sheet	On Log In, it should store the details in an excel sheet	PASS
TC02			
ITERATION	EXPECTED RESULT	ACTUAL ESULT	STATUS
2	On clicking Upload images, it should take the pictures of the user.	Takes pictures of the user	PASS
TC03			
ITERATION	EXPECTED RESULT	ACTUAL RESULT	STATUS
3	On clicking Upload Images, under dim lighting conditions it takes pictures of the user	It may not recognize the perfect images of the user for recognition	FAIL
TC04			
ITERATION	EXPECTED RESULT	ACTUAL RESULT	STATUS
4	On Clicking Train Images, the images are trained and stored in the data base	The images will be trained and upon successful training of images, A message will be displayed with user ID, Name	PASS
TC05			
ITERATION	EXPECTED RESULT	ACTUAL RESULT	STATUS
5	On Clicking Verify Images, Images will be verified against the authenticated user.	Camera will be opened and verifies the images of the user. Upon successful verification, a message will be displayed " Login Success"	PASS
TC06			
ITERATION	EXPECTED RESULT	ACTUAL RESULT	STATUS
6	On Clicking verify images, Images will be verified against the authenticated user	Camera will be opened and If the images are not read previously, it will return "Unknown"	FAIL
TC07			
ITERATION	EXPECTED RESULT	ACTUAL RESULT	STATUS
7	On clicking Quit, The framework stops	The framework stops	PASS



CHAPTER 6: RESULTS

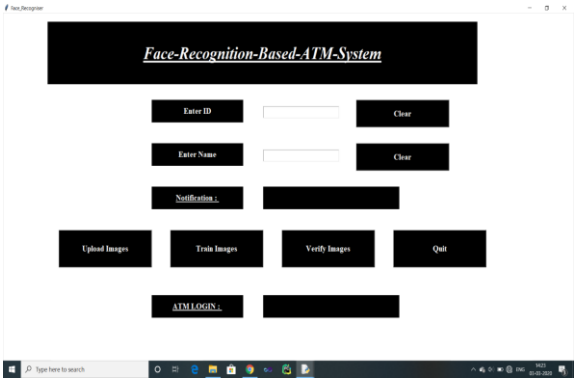


Fig.6.1.1:Frame Work

International Journal of Research
Available at https://journals.pen2print.org/index.php/ijr/
e-ISSN: 2348-6848
p-ISSN: 2348-795X
Volume 07 Issue 06
June 2020

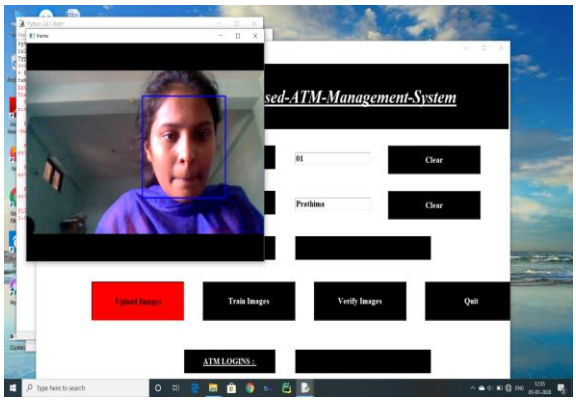


Fig.6.1.2:Uploading Images

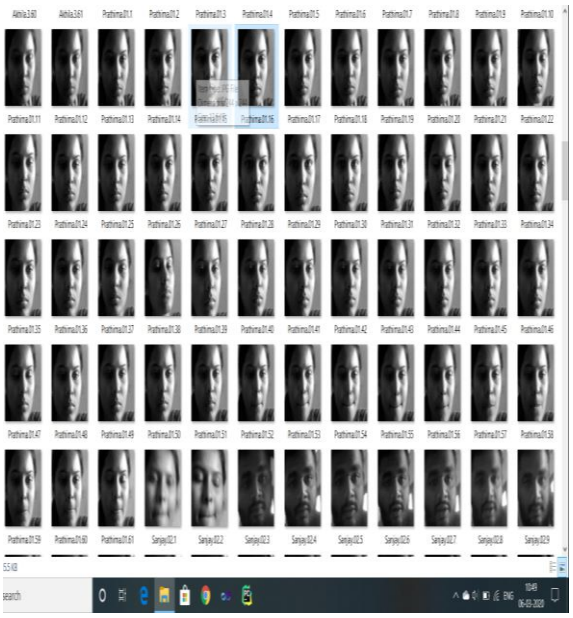


Fig.6.1.3: Upload Details

Fig.6.1.4:Training Details

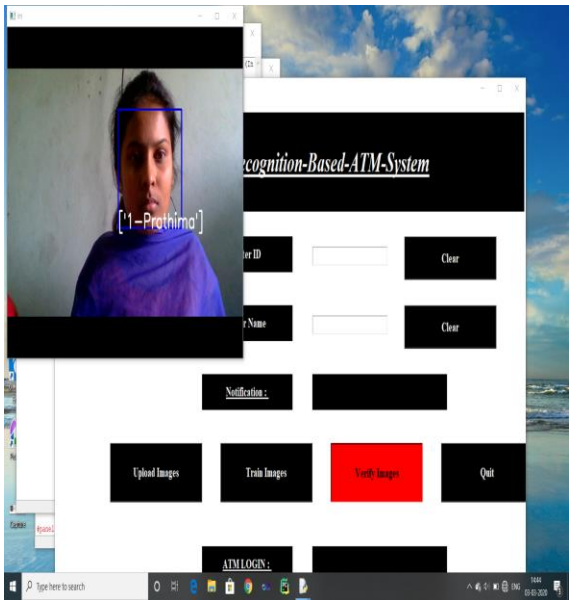


Fig.6.1.12: Verification of the User

Fig.6.1.13: Trained Images

Fig.6.1.14: Trained Images

Fig.6.1.15: Trained Images

International Journal of Research
Available at https://journals.pen2print.org/index.php/ijr/
e-ISSN: 2348-6848
p-ISSN: 2348-795X
Volume 07 Issue 06
June 2020

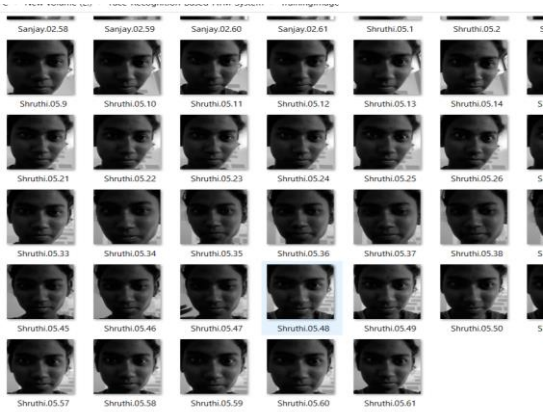


Fig.6.1.16: Trained Images

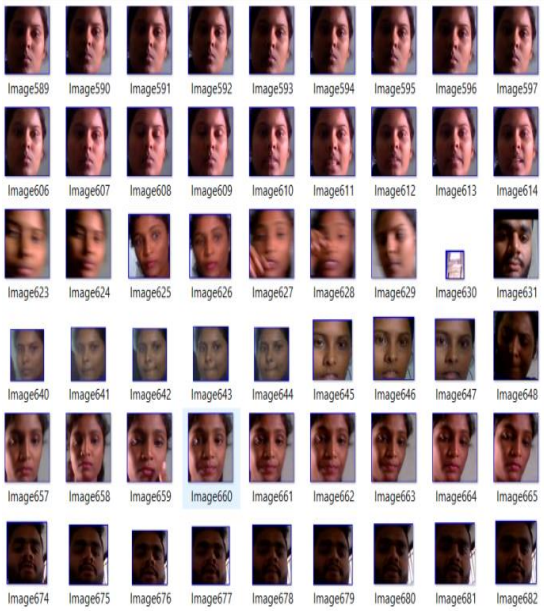


Fig.6.1.17:Images Unknown

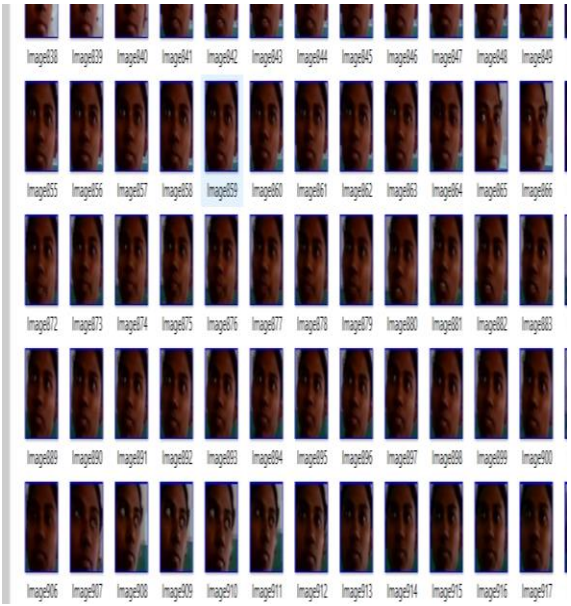


Fig.6.1.18:Images Unknown

CONCLUSION

Face recognition is a large-scale research field when it comes to image processing. While dealing this project, I have studied the substantial analysis of the two aspects such as face detection and recognition techniques in various terms such as runtime and accuracy. The result shows that using LBPH, the multiple faces are detected at a single trainer set and its accuracy is high as compared to Eigenfaces and Fisher faces. There are large number of challenges and issues in face recognition in the real-time applications. Future work would include a comprehensive study to recognize the faces at various angles

FUTURESCOPE

The scope of the project includes that what the future improvements can be done in this system to make it friendly to user. Our project has slight scope in future. It can be amended when requirement emerges as it is versatile in terms of extension. There are some facets which can be further modified such as recognized distance can be extended, Graphics Processing Unit can be used for large amount of database and quick processing, data storage can be made server based.

Bibliography

1. O. M. Parkhi, A.Vedaldi, A. Zisserman Deep Face Recognition British Machine Vision Conference, 2015.
2. H.-W. Ng, S. Winkler. A data-driven approach to cleaning large face datasets. Proc. IEEE International Conference on Image Processing (ICIP), Paris, France, Oct. 27-30, 2014
3. Beyeler, M. (2017). Machine Learning for OpenCV: Intelligent image processing with Python. London: Packt Publishing Ltd.
4. Bruce, P., & Bruce, A. (2017). Practical Statistics for Data Scientists. California: O'Reilly Media.

5. Chollet, F. (2018). Deep Learning with Python. New York: Manning Publications Co.
6. Géron, A. (2017). Hands-On Machine Learning with Scikit-Learn and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems. Brooklyn: O′Reilly.
7. Grus, J. (2015). Data Science from Scratch: First Principles with Python. California: O'Reilly Media. Lutz, M. (2013). Learning Python. California: O'Reilly Media.
8. Matthes, E. (2015). A Hands-On, Project-Based Introduction to Programming. San Francisco: William Pollock.
9. McKinney, W. (2017). Python for Data Analysis: Data Wrangling with Pandas, NumPy, and IPython. California: O'Reilly Media.
10. Phillips, D. (2014). Creating Apps in Kivy: Mobile with Python. California: O'Reilly.
11. Solem, J. E. (2012). Programming Computer Vision with Python: Tools and algorithms for analyzing images. United States: O'Reilly Media.
12. Sweigart, A. (2015). Automate the Boring Stuff with Python: Practical Programming for Total Beginners. San Francisco: William Pollock.
13. *Ulloa, R. (2015). Kivy: Interactive Applications in Python. Birmingham: Packt Publishing Ltd.*