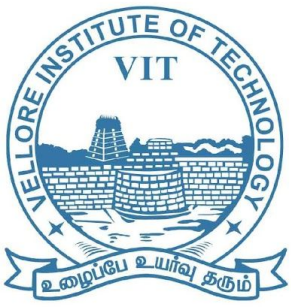



VIT®

Vellore Institute of Technology

(Deemed to be University under section 3 of UGC Act, 1956)

# ARTIFICIAL INTELLIGENCE

**Course Code**: ITE2010
**Slot**: C1+TC1

# Project Report

(Face Recognition)

Under the guidance of
**Prof. Chandrashekhar**

SUBMITTED BY
Vinit Shahdeo - 15BIT0335

**FACE DETECTION AND RECOGNITION SYSTEM**


**Abstract**

Face (facial) recognition identifies the humans by the unique characteristics of their Faces. Face recognition technology is the least intrusive and fastest bio-metric technology. With increasing security needs and with advancement in technology extracting information has become much simpler.

This project aims on building an algorithm and application based on face detection and recognition. The most basic purpose is to identify the face and retrieving information stored in database.
It involves two main steps. First to detect the face of a person and identify the distinguishing factors in image n storing them and Second step to compare it with the existing images and returning the data related to that image.
The algorithms used for face detection is Viola-Jones object detection and for face recognition simple correlation algorithm is used.


## INTRODUCTION

### Biometrics

Biometrics is used in the process of authentication of a person by verifying or identifying that a user requesting a network resource is who he, she, or it claims to be, and vice versa.
The property that is used is a human trait associated with a person itself like structure of data with the incoming data we can verify the identity of a particular person.
There are so many types of the biometric system like detection and recognition, iris recognition etc., these traits are used for human identification in surveillance system, criminal identification, face details etc

# Objective

The face is our primary focus of attention in social life playing an important role in conveying identity and emotions. We can recognize a large no. of faces learned throughout our lifespan and identify faces at a glance even after years of separation.

There are very large variations but our aim is more obvious in visual stimulus due to changing condition, aging and distractions such as beard, glasses or changes in hairstyle.

The Computational models which are of face recognition are interesting because they can contribute not only to theoretical knowledge but also to practical applications.
Computers could be applied to a wide variety of tasks including criminal identification, security system, image and film processing, identity verification, tagging purposes and human-computer interaction because of their ability to detect and recognize the faces.
It is very difficult to develop a conceptual model of face detection and recognition because of the complexity of the faces, multidimensional and meaningful visual stimuli.

In website and all the other applications , face detection and recognition is widely used .

The tagging feature which is done automatically adds a new dimension to sharing pictures among the people who are in the picture and also gives the idea to other people about who the person is in the image. In our project, we have studied and implemented a pretty simple but very effective face detection algorithm Viola- Jones algorithm.
Our aim in this project is to first detect face and compare it with the faces in our database and then recognize the person.

**LITERATURE SURVEY – PAPER 1**

FACE DETECTION (VIOLA–JONES OBJECT DETECTION ALGORITHM)

The Viola–Jones object detection framework is the first object detection framework to provide competitive object detection rates in real-time proposed in 2001 by Paul Viola and Michael Jones. The variety of object classes can be detected by training the algorithm. It was motivated primarily by the problem of face detection. This algorithm has four stages:

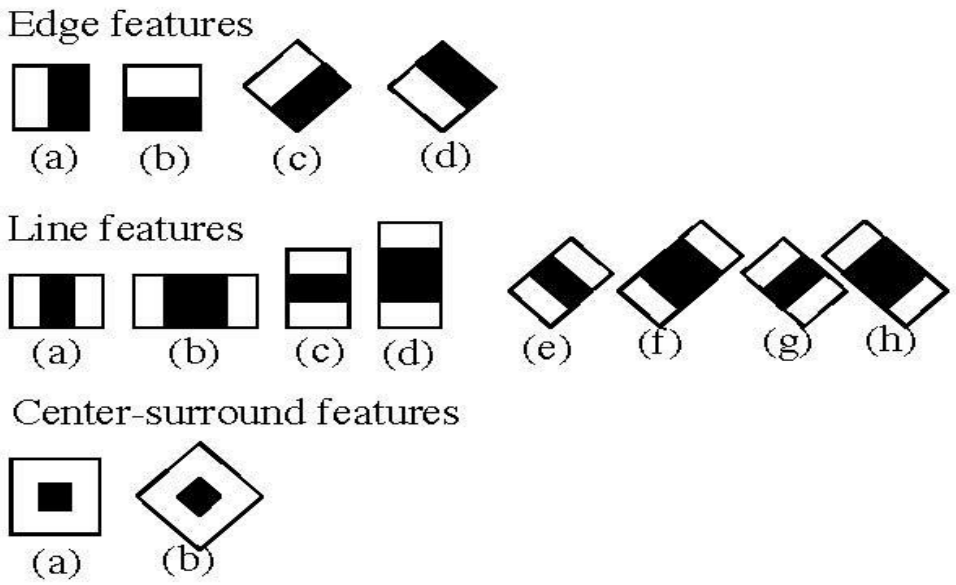
 Haar Feature
 Integral Image
 Adaboost Training
 Cascading Classifier

HAAR LIKE FEATURES

- There are different types of haar feature which are applied on the image to see whether the feature exists in the image.

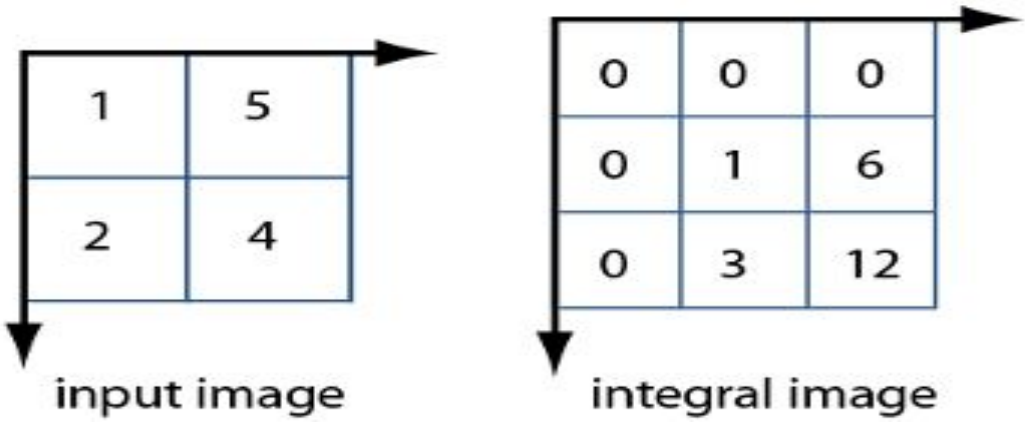
- The 24X24 window is applied all over the image ,in which for each operation, we are subtracting the sum of pixels in white region with ,sum of pixels in black region which output an

integer value, that determines the validation of the corresponding feature.




## INTERNAL IMAGE

- This method reduces the cost of calculating sum of pixels while validating a feature.
- We are doing the same by taking only the corner pixel values. For a given input image, or getting integral image, we sum up all pixels that are falling to the left and top region of the pixel.
- The advantage of the integral image is, it reduces the computation by 4 times.

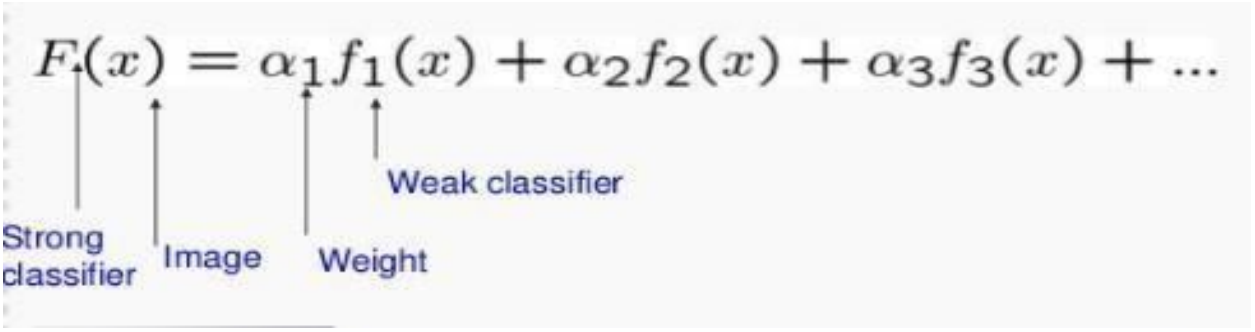


## ADABOOST TRAINING

- Adaboost decides some set of features and gives weight to each feature, and the linear combination of all these features ,is used to decide whether it is a face or not.

- Weak classifiers are some features which at least perform better than random guessing, if we give hundred face images, it will be able to detect more than fifty as face images.

- The output of the weak classifier is 1, if the feature is recognized else 0.The sum of product of relevant weak classifiers and there weights gives the strong classifier, whose value must be more than threshold for validation as a face image, which is less for a non-facial image.

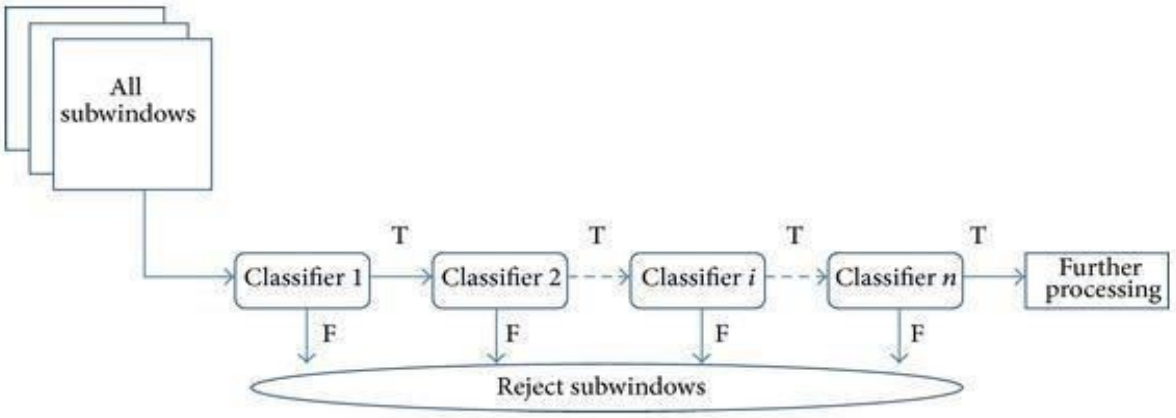



## CASCADING CLASSIFIER

- If we have an input image say 640X480, we have to perform 2500 feature validations for each 24X24 window, and take the linear combination of all those classifiers for validation.

- In cascading,we divide these 2500 features into set of stages in which the number of features increases in ascending order.

- The advantage is we can reject any non-face image with less time.




# LITERATURE SURVEY – PAPER 2

DETECTION AND LOCALIZATION TECHNIQUES

Face detection and localization from images is a key problem and a necessary first step in face recognition systems, with the purpose of localizing and extracting the face region from the background. It also has several applications in areas such a content-based image retrieval, video coding, video conferencing, crowd surveillance, and intelligent human-computer interfaces.

- Eigenface-Based Method
- Spatial Matching Detector Method
- Neural Networks Method
- Fuzzy Theory Based Method
- Other Methods

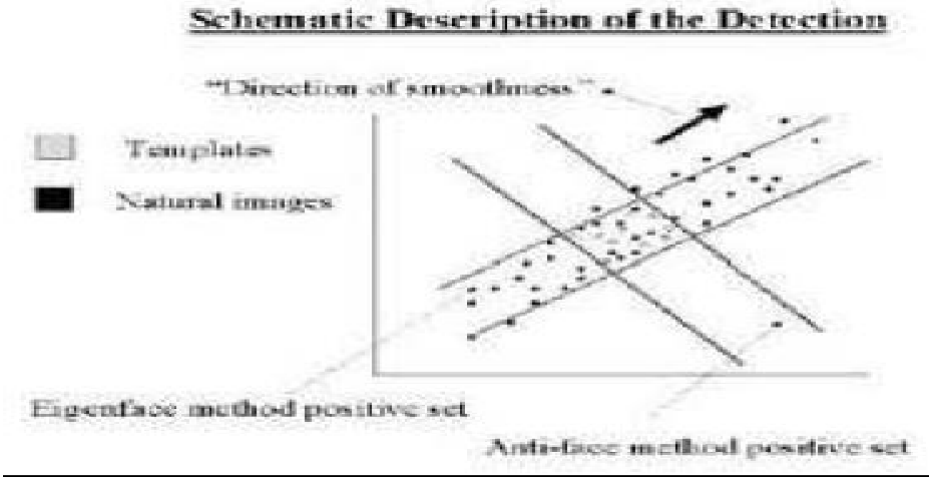
## Eigen face-Based Method

This method approximates the multi-template T by a low-dimensional linear subspace F, usually called the face space.
Concept of t is used, members of the images were called as t , if their distance from F is smaller than a certain threshold. The images which pass this test are projected on F and these projections are compared to those in the training set.
The system decomposes an entire input image into sub-band images which contain the discriminant features. Multiple sliding windows within different sub-bands are aligned to the same spatial location.

Features are selected from multiple sub-bands to calculate the likelihood ratios. Face locations are reported where the likelihood ratios exceed a fixed threshold.

## Spatial Matching Detector Method

Support Vector Machine is used for the embracement of this approach, various template matching methods, other discriminable Kernel Cost Function methods.
Only images which passed the threshold test imposed by the first detector were examined by the second detector, etc.
The performance of the algorithm was compared with well known eigen face and support vector machine but it came to be a little faster than that.
The algorithm's "positive set" (the images it classifies as members of the multi-template), is orthogonal to the direction around which random images cluster, hence, there are relatively few false alarms.




## Neural Networks Method

Neural networks have been applied, with considerable success, to the problem of frontal face detection. The system arbitrated between multiple networks to improve performance over a single network.
Hybrid neural method is proposed to locate human eyes. The new neural network model proposed, the Constrained Generative Model, performed an accurate estimation of the face set, using a small set of counter-examples.

The use of three layers of weights allows evaluating the distance between an input image and the set of face image.

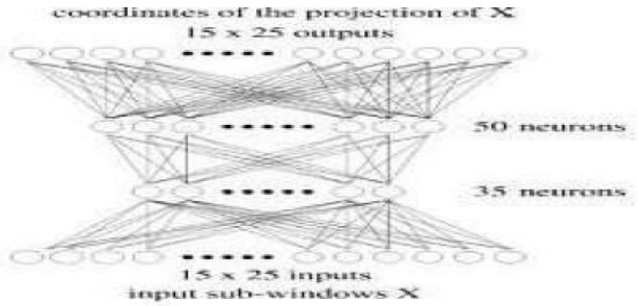



Fig. 13. Neural network layers

## Fuzzy Theory Based Method

This approach detects faces in color images based on the fuzzy theory.
In this paper, Wu et al. made two fuzzy models to describe the skin and hair color, in which used a perceptually uniform color space to describe the color information to increase the accuracy and stableness.
The models are used to detect the colour regions of the skin and other parts, then comparing them with the pre-built head-shape models by using a fuzzy theory based pattern-matching method to detect face candidates.

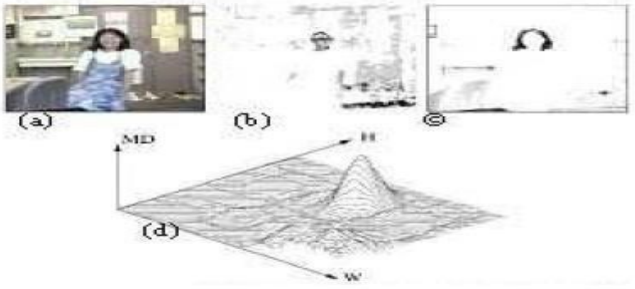



Fig. 14. An MMD obtained by comparing the skin-color similarity map and the hair-color similarity map with the head-shape models

## OTHER METHODS

This method integrates the above multiple methods and various techniques, i.e. Bayesian, neural networks, fuzzy logic, and others.
To extract facial expression information for automatic recognition Three methods were developed.
The first one is facial feature point tracking using a coarse-to-fine pyramid method. The second one is dense flow tracking together with principal component analysis (PCA). The third one is high

gradient component (i.e. Furrow) analysis in spatio-temporal domain.

## RECOGNITION TECHNIQUES

The last step in the process is the face recognition. More than 20 years have been used in the research for this recognition. It has become a new major research area in the last few years because of a number of potential applications ranging from security access control, personal identification to human computer communication. A number of face methods have been proposed.

Statistical approach
Feature Matching Approach
Neural Networks Approach
Fuzzy Theory Approach
Other Approaches

## STATISTICAL APPROACH

In this approach quantitative description of faces is characteristic, elementary numerical description — features — are used. The set of all possible patterns forms the pattern or feature space.
The classes form clusters in the feature space, which can be separated by discrimination hyper-surface. The approach chiefly embraces geometrical parameterization method, eigneface method, Fisherface method, evolutionary pursuit algorithm, etc.
The use of geometrical parameterization, i.e., distances and angles between points such as eye corners, mouth extremities, nostrils, and chin top is one method of characterizing the face. A simple distance measure is used to check for similarity between an image of the test set and the image in the reference set.

## FEATURE MATCHING APPROACH

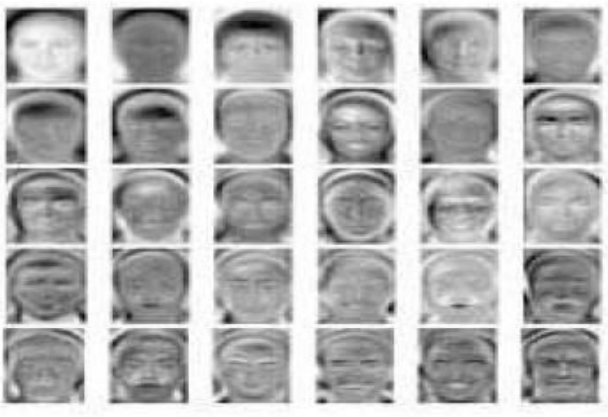
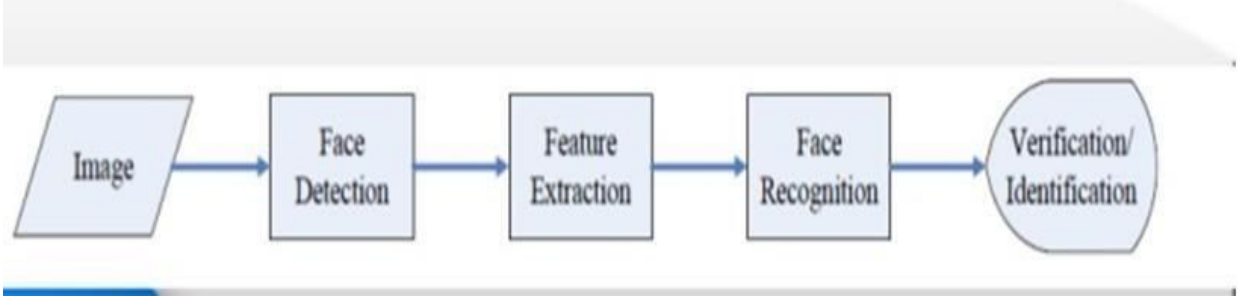


Fig. 21. The 30 eigenfaces derived by the Eigenface method

**Major Objectives:-**

Save Images to Database
Detect faces
Match detected faces to the database
Recognize the faces
Provides accurate information about them




# FACE RECOGINITION

In this project after detecting the face of a person, the image of the detected face is correlated with the face in data base. Correlation is any of a broad class of statistical relationships involving dependence, though in common usage it most often refers to the extent to which two variables have a linear relationship with each other. Image in database and detected face are in grayscale and are called 2D-image. The gray scale images can be represented in the form of MxN matrix and can be correlated by using:

$$r = \frac{\sum_m \sum_n (A_{mn} - \bar{A})(B_{mn} - \bar{B})}{\sqrt{(\sum_m \sum_n (A_{mn} - \bar{A})^2)(\sum_m \sum_n (B_{mn} - \bar{B})^2)}}$$

Where $A$ =*meanofmatrixA* and $B$ =*meanofmatrixB*

# ALGORITHM OF FACE DETECTION

 The image of a person is captured in camera.

 Using Face Detection algorithm image of face is extracted from the image captured from camera.

 Image of face is then resized such that its size is equal to the size of image in database.

 Let A=resized image of face captured from camera and B=image in database correlating two images.

$$r = \frac{\sum_m \sum_n (A_{mn} - \bar{A})(B_{mn} - \bar{B})}{\sqrt{(\sum_m \sum_n (A_{mn} - \bar{A})^2)(\sum_m \sum_n (B_{mn} - \bar{B})^2)}}$$

# PROPOSED ARCHITECTURE

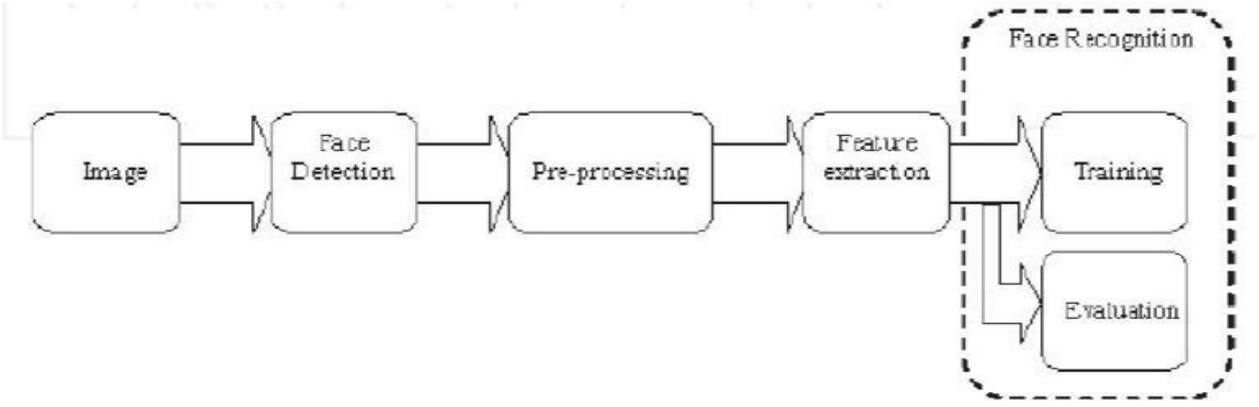
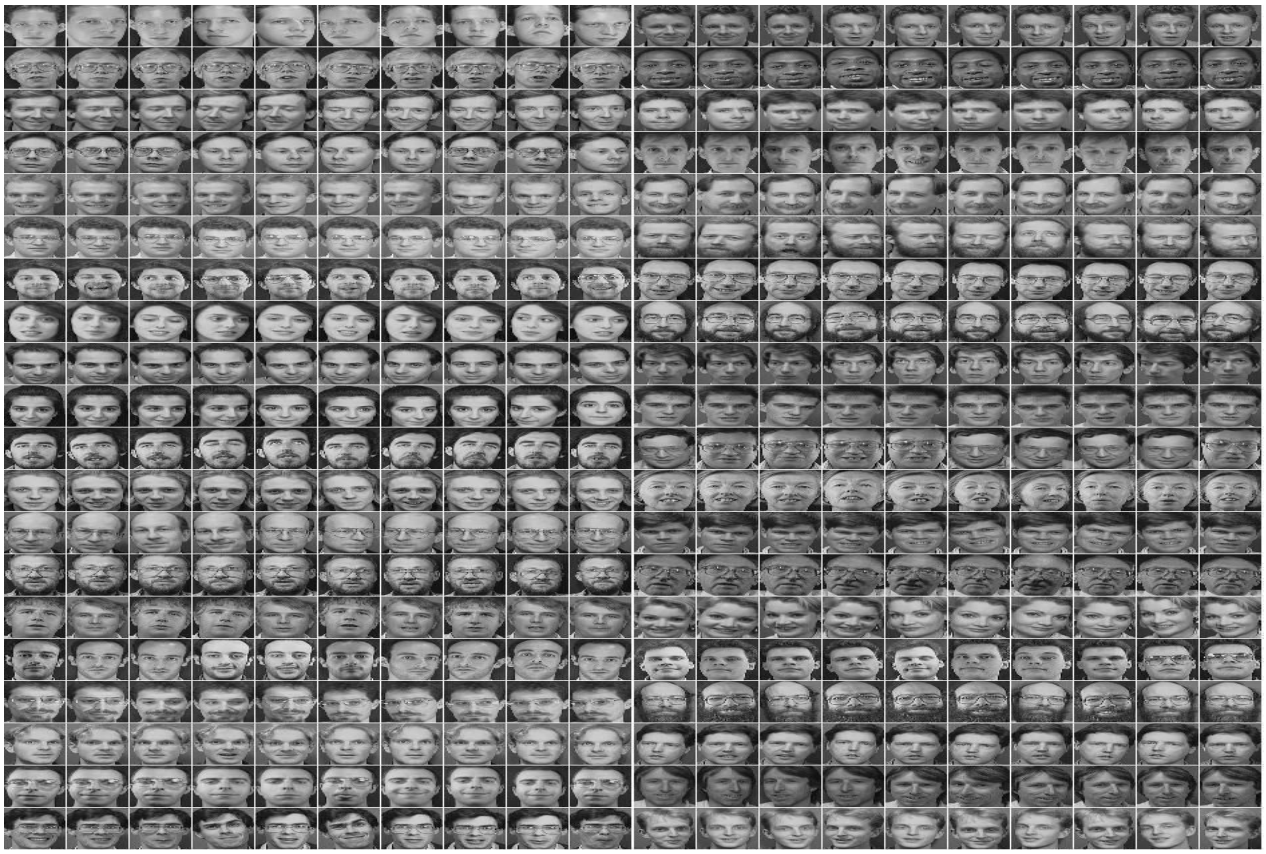



IMAGE: Firstly, the camera captures the face of the person trying to log in. This would be our input image.
FACE DETECTION: The input image is detected using the face detection algorithm described. In this stage the image of the face is extracted from the captured image and is resized so as to match the size of the image in the database. Viola Jones algorithm described above is used for this stage.

PRE-PROCESSING: In this stage the image is improved so as to supress some unwilling distortions and enhance some image features that are important for further processing. Various image enhancement and restoration techniques are used in this particular stage.In this stage RGB image is converted to grayscale image using cascade object detector.
FEATURE EXTRACTION:In this stage the grayscale image obtained in previous step is compared with the Haar features for identifying the required features.
FACE RECOGNITION: The processed image is now compared with the datasets stored in the system. If the extracted features match with any of the stored images, the login is successful. The statistical correlation formula is used for this.

# EXPERIMENTAL RESULTS DATABASE USED

Our Database of Faces contains a set of face images taken between April 1992 and April 1994 at the lab. The database was used in the context of a face recognition project. There are ten different images of each of 40 distinct subjects. For some subjects, the images were taken at different times, varying the lighting, facial expressions (open / closed eyes, smiling / not smiling) and facial details (glasses / no glasses). All the images were taken against a dark homogeneous background with the subjects in an upright, frontal position (with tolerance for some side movement). The files are in PGM format. The size of each image is 92x112 pixels, with 256 grey levels per pixel. The images are organized in 40 directories (one for each subject), which have names of the form sX, where X indicates the subject number (between 1 and 40). In each of these directories, there are ten different images of that subject, which have names of the form Y.pgm, where Y is the image number for that subject (between 1 and 10).



The database can be retrieved from

http://www.cl.cam.ac.uk/Research/DTG/attarchive:pub/data/att_faces.tar.Z
As a 4.5Mbyte compressed tar file or from

http://www.cl.cam.ac.uk/Research/DTG/attarchive:pub/data/att_faces.zip
As a ZIP file of similar size.

# COMPARATIVE STUDY OF PROPOSED SYSTEM WITH RELATED EXISTING SYSTEM

| EXISTING SYSTEM | PROPOSED SYSTEM |
|---|---|
| The companies that use the traditional attendance system suffer from time fraud. | Using facial biometric systems in a company would eliminate the time fraud problem. |
| Any unauthorized person may access the company's authorized section using other person's identity. | Anyone that is not in the system will not be given access. |
| One person has to be always available for monitoring the attendance system. | The facial system is automated. The company doesn't need to worry about the monitoring. |

## CODE:

### Find faces in pictures

```
import face_recognition
image = face_recognition.load_image_file("your_file.jpg")
face_locations = face_recognition.face_locations(image)
```

### Find and manipulate facial features in pictures

```
import face_recognition
image = face_recognition.load_image_file("your_file.jpg")
face_landmarks_list = face_recognition.face_landmarks(image)
```

### Find and manipulate facial features in pictures

```
import face_recognition
image = face_recognition.load_image_file("your_file.jpg")
face_landmarks_list = face_recognition.face_landmarks(image)
```

### Identify faces in pictures

```
import face_recognition
known_image = face_recognition.load_image_file("biden.jpg")
unknown_image = face_recognition.load_image_file("unknown.jpg")

biden_encoding = face_recognition.face_encodings(known_image)[0]
unknown_encoding = face_recognition.face_encodings(unknown_image)[0]

results = face_recognition.compare_faces([biden_encoding], unknown_encoding)
```

### Automatically locate the facial features of a person in an image

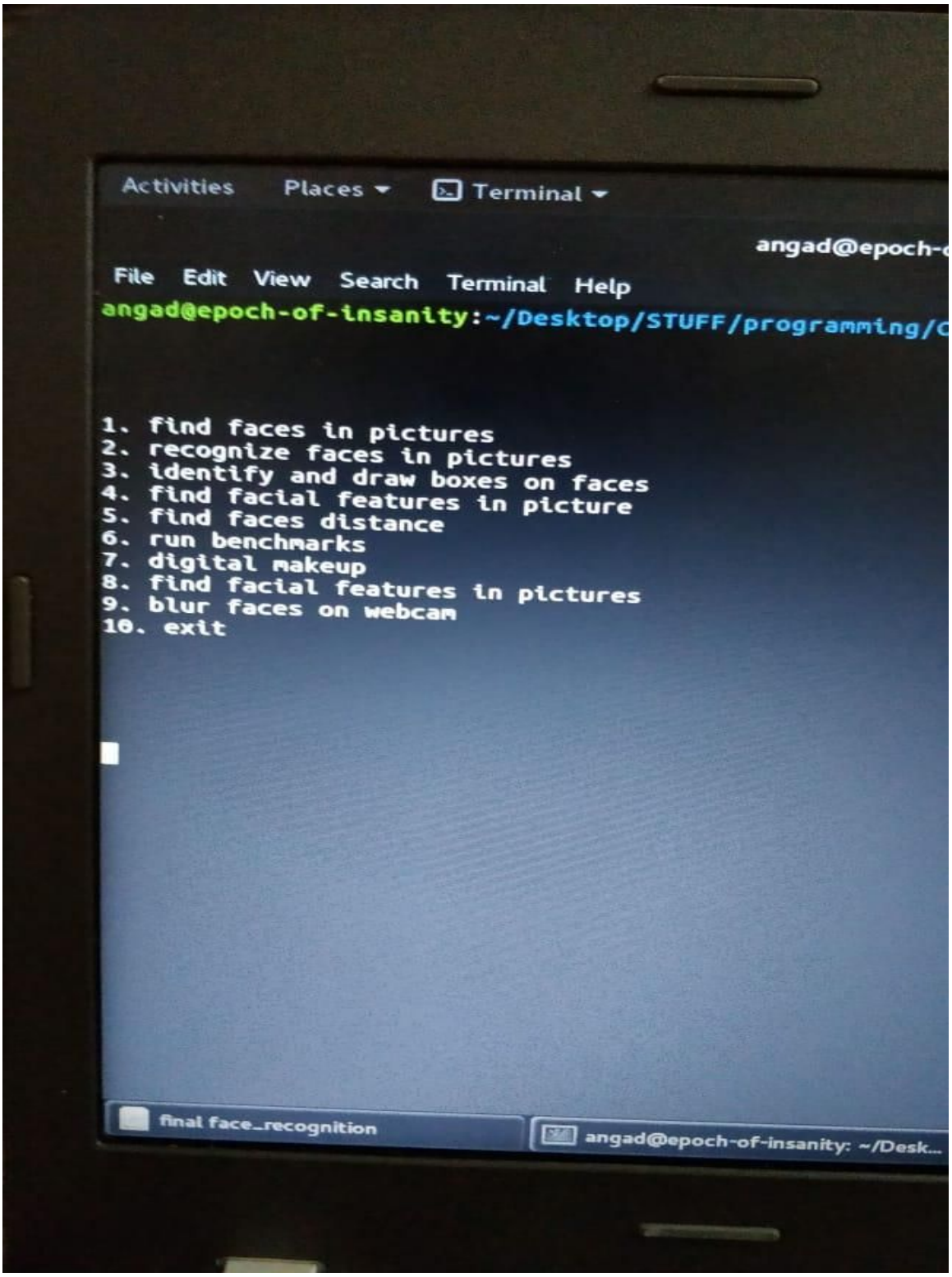
```
import face_recognition

image = face_recognition.load_image_file("my_picture.jpg")
face_landmarks_list = face_recognition.face_landmarks(image)

# face_landmarks_list is now an array with the locations of each facial feature in each face.
# face_landmarks_list[0]['left_eye'] would be the location and outline of the first person's
```

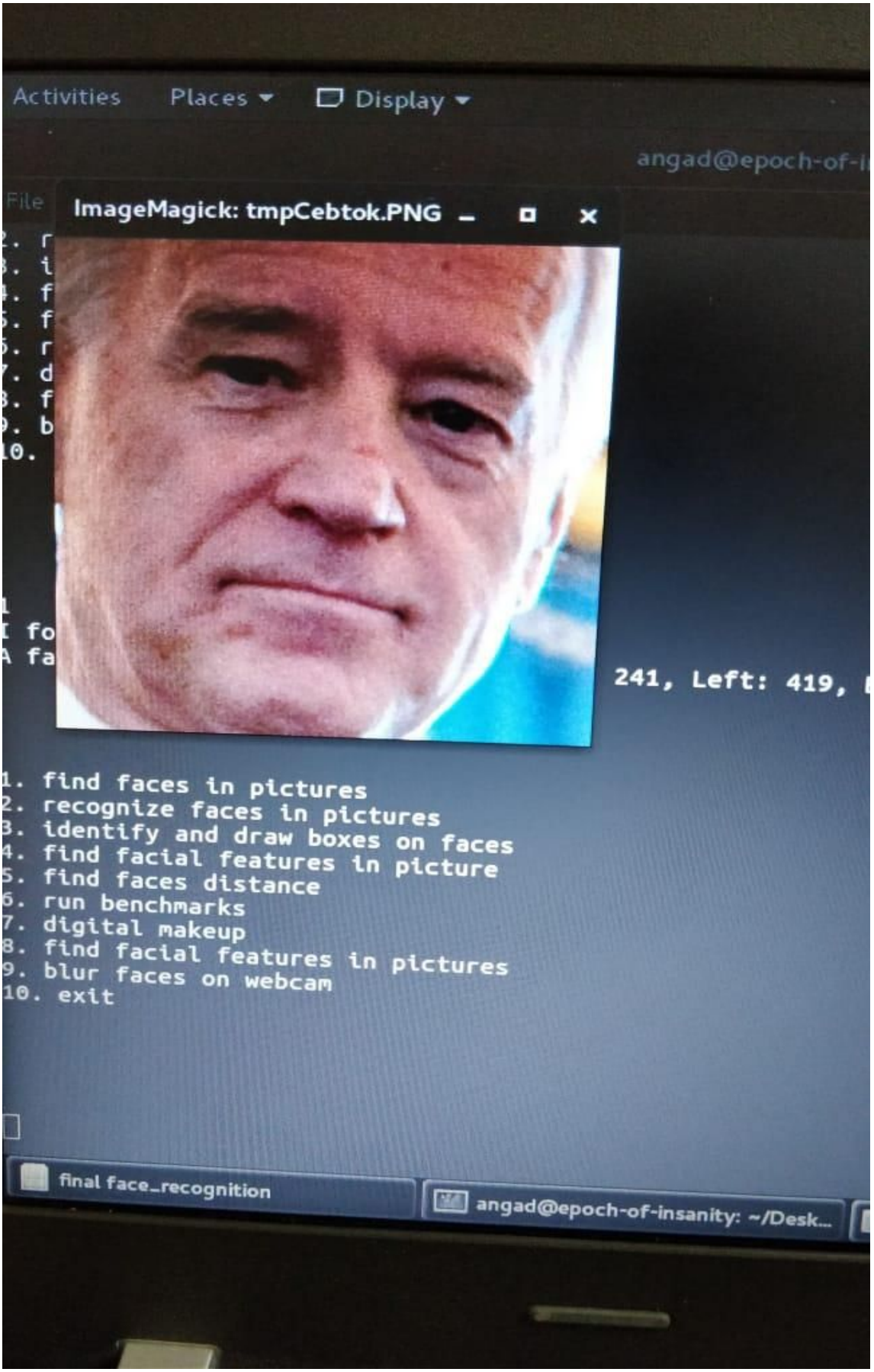
left eye.

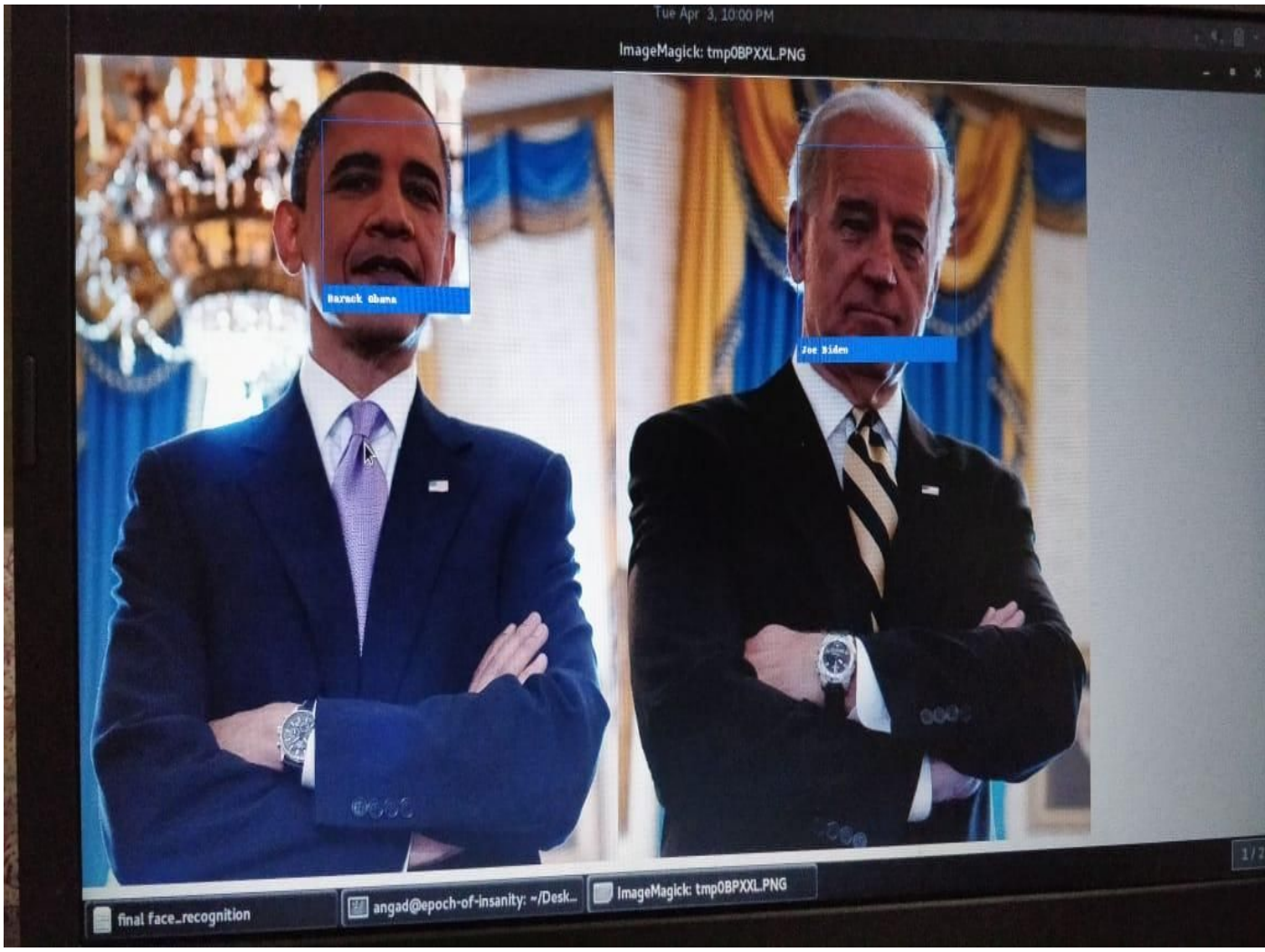
**Recognize faces in images and identify who they are**

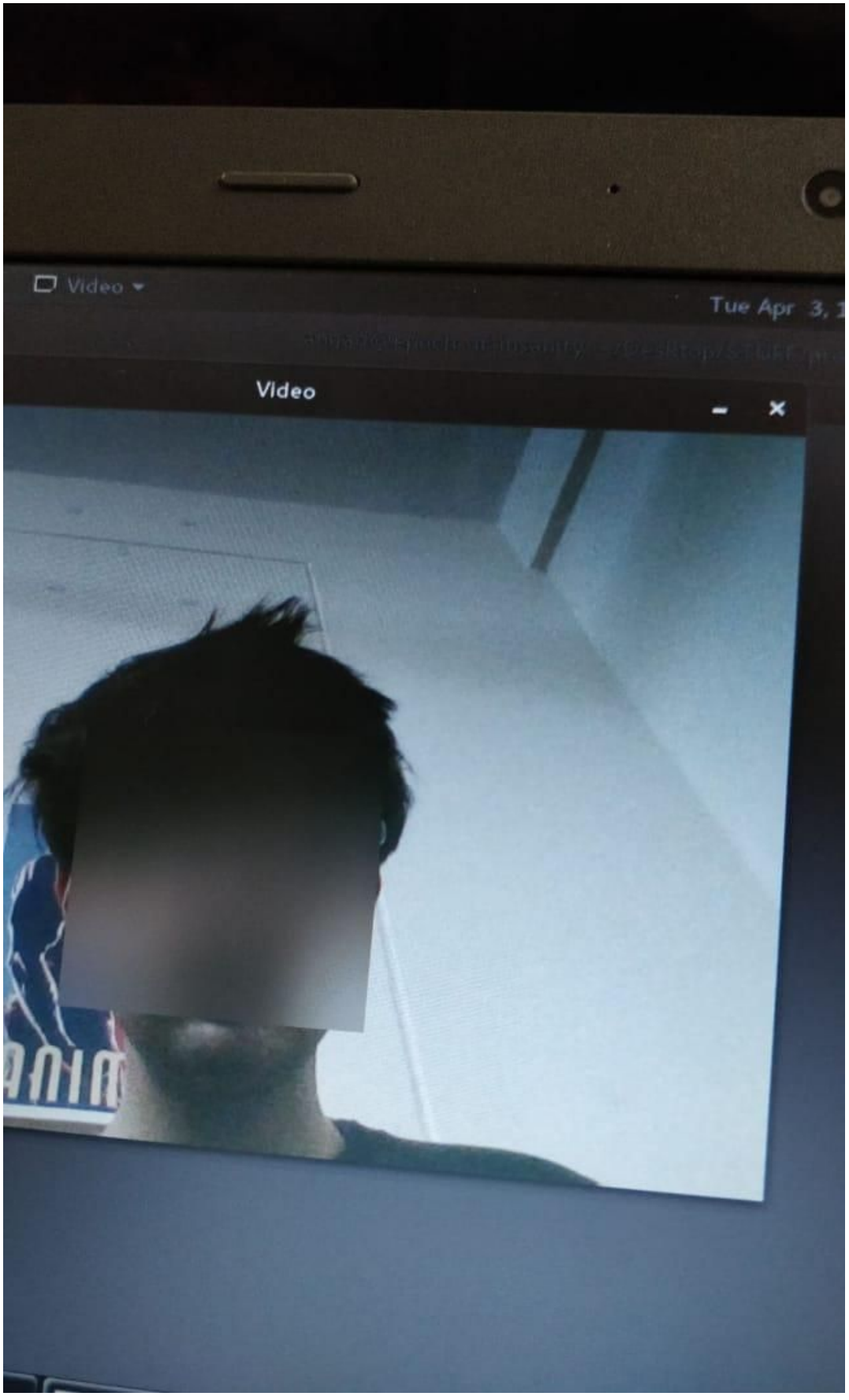
```
import face_recognition

picture_of_me = face_recognition.load_image_file("me.jpg")
my_face_encoding = face_recognition.face_encodings(picture_of_me)[0]

# my_face_encoding now contains a universal 'encoding' of my facial features that can be compared to any other picture of a face!

unknown_picture = face_recognition.load_image_file("unknown.jpg")
unknown_face_encoding = face_recognition.face_encodings(unknown_picture)[0]

# Now we can see the two face encodings are of the same person with `compare_faces`!

results = face_recognition.compare_faces([my_face_encoding], unknown_face_encoding)

if results[0] == True:
    print("It's a picture of me!")
else:
    print("It's not a picture of me!")
```

**Screenshot**


Activities
Places
Terminal
angad@epoch-
File Edit View Search Terminal Help
angad@epoch-of-insanity:~/Desktop/STUFF/programming/
1. find faces in pictures
2. recognize faces in pictures
3. identify and draw boxes on faces
4. find facial features in picture
5. find faces distance
6. run benchmarks
7. digital makeup
8. find facial features in pictures
9. blur faces on webcam
10. exit
final face_recognition
angad@epoch-of-insanity: ~/Desk...


Activities
Places
Display
angad@epoch-of-i
File
ImageMagick: tmpCebtok.PNG
241, Left: 419,
1. find faces in pictures
2. recognize faces in pictures
3. identify and draw boxes on faces
4. find facial features in picture
5. find faces distance
6. run benchmarks
7. digital makeup
8. find facial features in pictures
9. blur faces on webcam
10. exit
final face_recognition
angad@epoch-of-insanity: ~/Desk...


Tue Apr 3, 10:00 PM
ImageMagick: tmp0BPXXL.PNG
Barack Obama
Joe Biden
final face_recognition
angad@epoch-of-insanity: ~/Desk...
ImageMagick: tmp0BPXXL.PNG


Video
Tue Apr 3,
Video

## CONCLUSION

This project has presented an approach for face detection and recognition. We have used Haar features, integral image, Adaboost, Cascading and correlation to successfully detect and recognize the person.

This algorithm can be used in any secured buildings, also this algorithm can be used as a biometric of a person and help security forces to successfully detect criminals and make society safe.

## FUTURE IMPLEMENTATIONS

Face recognition systems used today work very well under constrained conditions, although all systems work much better with frontal mug-shot images and constant lighting. All current face recognition algorithms fail under the vastly varying conditions under which humans need to and are able to identify other people. Next generation person recognition systems will need to recognize people in real-time and in much less constrained situations.

Concept of Soft Computing can be used for automatic face recognition system. Using Soft Computing, neural network can be combined with fuzzy logic to enhance the performance of face recognition.

Another avenue for research would be to implement other feature extraction technique on the same data set. In future, two or more classifiers can be combined to achieve better results.

## REFERENCES


[1] Papageorgiou, C., Oren, M., and Poggio, T. 1998. A general framework for object detection. In International Conference on Computer Vision.

[2] Simard, P.Y., Bottou, L., Haffner, P., and LeCun, Y. (1999). Boxlets: A fast convolution algorithm for signal processing and neural networks. In M. Kearns, S. Solla, and D. Cohn (Eds.), Advances in Neural Information Processing Systems, vol. 11, pp. 571– 577.

[3] Freund, Y. and Schapire, R.E. 1995. A decision-theoretic generalization of on-line learning and an application to boosting. In Computational Learning Theory: Eurocolt 95, Springer-Verlag, pp. 23–37.

[4] Schapire, R.E., Freund, Y., Bartlett, P., and Lee, W.S. 1997. Boosting the margin: A new explanation for the effectiveness of voting methods. In Proceedings of the Fourteenth International Conference on Machine Learning.

[5] John, G., Kohavi, R., and Pfeger, K. 1994. Irrelevant features and the subset selection problem. In Machine Learning Conference Proceedings

[6] Webb, A. 1999. Statistical Pattern Recognition. Oxford University Press: New York.

[7] Quinlan, J. 1986. Induction of decision trees. Machine Learning, 1:81–106.

[8] Roth, D., Yang, M., and Ahuja, N. 2000. A snowbased face detector. In Neural Information Processing 12.