**Face Recognition**

**Project name:** Face Recognition.

**Group name:** 11100010

**Group member’s ID:** 52, 64, 66, 88.

**Project Details:**

Face recognition using several classifiers available in the open computer vision library (openCV). Face recognition is a biometric software application capable of uniquely identifying or verifying a person by comparing and analyzing. Face recognition is a identification system and faster than other systems since multiple faces can be analyzed at the same time. The difference between face detection and identification is face detection is to identify a face from an image and locate the face. Face recognition is making the decision “Whose face is it?”, using an image database. In this project both are accomplished using different techniques.

The project is mainly a method for detecting faces in a given image by using **OpenCV-Python**module. The first phase uses picture for detect of our faces and second phase uses camera to for video of our faces which generates a feature set in a location of your PC. It has two simple commands**-**

Face\_ recognition- Recognize faces in a photograph or folder full for photographs.

Face\_detection- Find faces in a photograph or folder full for photographs.

For face recognition, first generate a feature set by taking few image of your face and create a directory with the name of person and save their face image.

The main flow of face recognition is first to locate the face in the picture and the compare the picture with the trained data set. If the there is a match, it gives the recognized label.

**OpenCV:**

A computer program that decides whether an image is a positive image (face image) or negative image (non-face image) is called a **classifier**. A classifier is trained on hundreds of thousands of face and non-face images to learn how to classify a new image correctly. OpenCV provides us with two pre-trained and ready to be used for face detection classifiers:

1. Haar cascade classifier.
2. LBP cascade classifier.

We use here, LBP cascade classifier.

**LBP cascade classifier:**

LBP is a texture descriptor and face is composed of micro texture patterns. So LBP features are extracted to form a feature vector to classify a face from a non-face. Following are the basic steps of LBP Cascade classifier algorithm:

1. **LBP Labelling:** A label as a string of binary numbers is assigned to each pixel of an image.
2. **Feature Vector:** Image is divided into sub-regions and for each sub-region, a histogram of labels is constructed. Then, a feature vector is formed by concatenating the sub-regions histograms into a large histogram.
3. **AdaBoost Learning:** Strong classifier is constructed using gentle AdaBoost to remove redundant information from feature vector.
4. **Cascade of Classifier:** The cascades of classifiers are formed from the features obtained by the gentle AdaBoost algorithm. Sub-regions of the image is evaluated starting from simpler classifier to strong classifier. If on any stage classifier fails, that region will be discarded from further iterations. Only the facial region will pass all the stages of the classifier.s