Face Detection and Recognition
Papers and Abstracts

Devanshi Piprottar, Shivani Shah, Suhani Ladani, Jagat Panchal, Parth Thakkar

[1]Institute of Engineering and Technology - Ahmedabad University
Ahmedabad, India

[2]Algorithm and Optimization for Big Data

[3]Information and Communication Technology
Guide: Dr. Ratnik Gandhi

{email}@iet.ahduni.edu.in

Abstract. *Be it the new snap features everyday, or attendance management, securities in phone, applications, face detection and recognition have been imbued in multiple in elds and applications. There are multiple algorithms for face detection and recognition. KLT, Viola Jones have been widely used for face detection. LDA, PCA, SVM are some of the face recognition algorithms. Better security, assurance is required with advancing technology to ensure that our data is safe; we make use of the fact that every human being has different fingerprints and iris to ensure personal restricted access.*

Index terms:Linear Discriminant Analysis, KNN, PCA, labeling, viola Jones, camshift

1. Introduction

Face Recognition is a natural Human Computer Interaction (HCI) technology where human life is superior by artificial intelligence. Humans are considered as having special face recognition system in their brain. Researchers are trying to understand this human face recognition system in order to perceive human knowledge to design an automated face recognition system such that they have equal ability as human to recognize the faces with endless memory and higher speed. Face Recognition is beneficial to areas like biometrics, security, law administration and access management. Face Recognition is being used in identifying or tracking the suspect, reconstruction of faces from remains in forensic medicine. It can also be useful for person identification in passports, driving license, etc. or for automatic identity verification on borders.

Face Recognition becomes one of the most biometrics authentication techniques from the past few years. It is an interesting and successful application of Pattern recognition and Image analysis. The system has basically two main tasks: verification and identification. Face verification means a 1:1 match that compares a input face images(which has to be identified) against a database face images. Face identification means a 1:N problem that compares a query face image against all image templates in a face database. Machine recognition of faces is gradually becoming very important due to its wide range of commercial and law enforcement applications, which include forensic identification, access control, border surveillance and human interactions and availability of low cost recording devices. Various biometric features can be used for the purpose of human recognition like fingerprint, palm print, hand geometry, iris, face, speech, gaits, signature etc.

There are a number of algorithms that can be used for recognition, the drawback of one algorithm is improved in the next. OpenCV-python is a platform for the demo of the face recognition model. The recognition is performed on a detected face using Linear Discriminate Analysis (Fisher face) and KNN for the classification. Algorithm speed has been improved by a number of approaches that have been discussed here. The complete process and algorithm followed has been described in detail and the difficulties, challenges along with the solutions have been elaborated.

2. Literature Survey

There exist several methods for extracting the most useful features from (preprocessed) face images to perform face recognition. One of these feature extraction methods is the Local Binary Pattern (LBP) method. With LBP it is possible to describe the texture and shape of a digital image. This is done by dividing an image into several small regions from which the features are extracted.

These features consist of binary patterns that describe the surroundings of pixels in the regions. The obtained features from the regions are concatenated into a single feature histogram, which forms a representation of the image. Images can then be compared by measuring the similarity (distance) between their histograms. Because of the way the texture and shape of images is described, the method seems to be quite robust against face images with different facial expressions, different lightening conditions, image rotation and aging of persons.
Keywords: Local Binary Patterns, texture, feature histogram, binary patterns

The focus on face recognition with the help of Independent Component Analysis. ICA reduces dependencies of both second order and higher order in the input. ICA also reconfigured as a generalization of PCA. Comparative analysis of space dimensionality sensitivity of ICA and discriminant performance of ICA when combined together with Bayes classifier's MAP criteria and Fisher's Linear Discriminant has been carried out in this paper. From the above two analysis it has been found out that ICA performs better in whitened and compressed space. Its performance is better than the fisher faces and eigenfaces method. But the performance of ICA deteriorates when combined with Fisher's Linear Discriminant.
Keywords: ICA, PCA, LDA, Bayes Classifier

Improving Classification with Class-Independent Quality Measures: Q-stack in Face Verification
A clear distinction between the single- and multiple classifiers scenarios has been made through existing approaches to classification with signal quality measures. This paper presents a uniform approach to sharply categorize based on the concept of stacking. Q-stack improves classification in uni and multimodal system by using the quality measures that are class independent and baseline classifier scores. Along with the

application of Q-stack on the task of biometric identity verification using face images and associated quality measures, discussion of Q-stack as a generalized framework in any single, multiple, and multimodal classifier ensemble. For reducing the error rates below those of baseline classifiers in single- and multi-classifier scenarios, some techniques are proposed.
Keywords: Evidence vectors, Classifier Stacking, statistical dependencies

3. Related works

The detection and recognition could be considered a type of classification problem.

3.1. Binary Classification

In order to preserve efficiency and to reduce the false positive rate, classification problem is divided into cascades of classifiers. An input image is passed from one classifier to the next classifier as long as each classifier classify window as a detected face. Every classifier has threshold value of high detection rate. Every classifier in the cascade is trained with the negative set including false positive scenario of previous stage classifier. Face detection is a binary classification since the algorithm extracts a frame and finds whether there is a frame in the face. 0: No face detected, and 1: face detected. In current haar cascade classifiers have haar features which can easily discard non-face section. The haar features is similar to edge detection haar features applied to an image. Haar features contains the white and black region segments which signifies varying intensity and represented by -1 and +1 respectively similar to convolution matrix in edge detection. Different features of varying size are applied to the image the single value output is obtained by summing up pixel values under black and white region and subtracting them. The result with maximum value is chosen as promising feature. Haar features are designed so as they incorporate all the possible combinations for detecting face e.g. eye region or a bridge of nose etc.

3.2. Multi class Classification

k Nearest Neighbors In pattern recognition, the k-Nearest Neighbors algorithm (or k-NN for short) is a non-parametric method used for classification and regression. In both cases, the input consists of the k closest training examples in the feature space. The output depends on whether k-NN is used for classification or regression:

- In k-NN classification, the output is a class membership. An object is classified by a majority vote of its neighbors, with the object being assigned to the class most common among its k nearest neighbors (k is a positive integer, typically small). If k = 1, then the object is simply assigned to the class of that single nearest neighbor.
- In k-NN regression, the output is the property value for the object. This value is the average of the values of its k nearest neighbors.

k-NN is a type of instance-based learning, or lazy learning, where the function is only approximated locally and all computation is deferred until classification. The k-NN algorithm is among the simplest of all machine learning algorithms. Both for classification and regression, it can be useful to assign weight to the contributions of the neighbors, so that the nearer neighbors contribute more to the average than the more distant ones. The neighbors are taken from a set of objects for which the class (for k-NN classification) or the object property value (for k-NN regression) is known. This can be thought of as the training set for the algorithm, though no explicit training step is required.

The training examples are vectors in a multidimensional feature space, each with a class label. The training phase of the algorithm consists only of storing the feature vectors and class labels of the training samples. In the classification phase, k is a user-defined constant, and an unlabeled vector (a query or test point) is classified by assigning the label which is most frequent among the k training samples nearest to that query point.

4. Our Approach

1. Face Detection using Viola-Jones
   - I **Haar Feature:** Viola Jones detector uses haar like features to detect the presence of such facial feature in an image. They developed these features using the idea of haar wavelet. To compensate the effect of different lighting conditions, all the images are mean and variance normalized. Then the scalar product between the images and the haar feature is used to detect the face.
   - II **Integral Image:** In an integral image the value of pixel (x,y) is the sum of pixels above and to left of (x,y). Calculation of sum of pixels becomes easier, how large may be the number of pixels, to an operation involving just four pixels.
   - III **Adaboost algorithm:** The basic idea is to apply each and every feature on all the equally weighted video frames and find best thresholds for each feature which will classify the faces to positive and negative. It has the cases errors or misclassifications. The features selected based on minimum error rate, having best classification of the face and non-face images. After each classification, misclassified images are assigned with higher weight. Until the required accuracy is not achieved these processes get repeated.
   - IV **Cascading:** To avoid higher computation cost of strong classifiers, Cascade classifiers used to group all the features into several stages containing several features.

     In given approach, Face Detection used to detect face form the input of recognition problem.
2. **Resize the images and add those into the database:** Images that are in Face Database folder are read one by one and resize into the window size of by ignoring less significant components. Append those images into database. Along with that unique number of labels are counted and stored as string.
   Resizing of images is required for
   – Dimensionality Reduction in width x height
   – Uniquely identifiable size
   – Noise reduction
3. **Train the database:** Training the database is done separately at the starting of recognition and then we are just loading the trained results every time in order to have faster computations. By applying knn nearest neighbors (k=1) and creating fisher faces and save those results for further recognition.

4. **Face Recognition using LDA:** The linear pattern recognition algorithm is developed using LDA and other techniques commonly in use. LDA is a generalized form of fisher's linear discriminant. LDA is a representation method of subspace derivation. It is a method of linear transform derived from training data. This is a process of approximate vectors by projecting images in a low dimensional subspace. LDA performs dimensionality reduction by preserving as much of the class discriminatory information as possible and ignoring some of the less significant components. It may lead to loss in information but as far as eigenvalues are small it doesn't create much difference. The scatter within-classes but also the scatter between-classes are taken into consideration. It is more capable of distinguishing image variation due to identity from variation due to other sources such as illumination and expression.

   LDA is related based on PCA and feature analysis for finding out the best linear combination of features to clarify the data. LDA is to overcome PCA limitation of intra class variability. Since the traditional LDA classification performance is degraded because their criteria for separation are not related straight to their accuracy in classification in output space. The linear model has no effect of noises and will mostly not over fit, so even if the face patterns distribution is complex and non-convex mostly, the cost effective solutions to the Face Recognition can still be provided by integration of linear methods with other strategies. Multi-class LDA algorithms are available to deal with two or more classes. Bayesian methods consent defining intra-class variations, so it illustrations a well performance than LDA. LDA is proper to pattern classification if the numbers of training samples of each class are large.

   There are two different scatter matrix:
   - Within Class Scatter Matrix

   $$S_w = \sum_{i=1}^{c} \sum_{j=1}^{n_i} (Y_j - M_i)(Y_j - M_i)^T$$

   - Between Class Scatter Matrix

   $$S_b = \sum_{i=1}^{c} (M_i - M)(M_i - M)^T$$

   Projection Matrix: $y = U^T x$
   LDA transformation achieves well by maximizing the between-class scatter while minimizing the within-class scatter.
   scatter matrices of the projected data: $\tilde{S}_b$, $\tilde{S}_w$,
   Product of eigenvalues: max$\frac{|\tilde{S}_b|}{|\tilde{S}_w|}$ =max$\frac{|U^T S_b U|}{|U^T S_w U|}$
   $S^{-1} S_b = U \Lambda U^T$
   Now, take these assumptions into account:
   - Square images with Width=Height=N
   - M is the number of images in the database
   - P is the number of persons in the database

We consider database of faces in form of vectors where for every face f_i, we have vectors:

$f_i = [i_1, i_2, \ldots i_{N^2}]$; where i = a, b, c, d,... Here, f_i is a column vector. Average of all the faces is computed such that $\bar{m} = 1/M.F_i$ Here F_i is the matrix formed by appending all the column vectors f_i. Average face of each person is calculated Similar to what is done in PCA, we subtract the average faces from the training faces. Now, for the of intra-class separation, we build scatter matrices S_1, S_2, S_3, S_4

Such that:
$S1 = (a_m a_m^T + b_m b_m^T), S2 = (c_m c_m^T + d_m d_m^T)$
$S1 = (e_m e_m^T + f_m f_m^T), S2 = (g_m g_m^T + h_m h_m^T)$
Where i_m is the vector formed after subtraction.
Thus, within-class scatter matrix $S_w = S_1, S_2, S_3, S_4$

With the interclass and intraclass we seek to maximize the function such that inter class separation is as large as possible while intra class separation is as small as possible. Thus, the overall algorithm would first require the application of PCA first as mentioned above and then LDA application. To classify the face, project the faces onto the LDA-space and then run a nearest-neighbor classifier.

LDA face recognition algorithm is used to recognize every detected face by predicting the label and assign it to face region frame.

5. **Face Tracking using Camshift** Camshift algorithm is a modified version of Meanshift Algorithm. Meanshift algorithm works on probability distributions. Color image data characterized as a probability distribution using color histogram; in order to track color faces in video. Due to the dynamic nature of color distribution in the image, mean shift algorithm failed and new algorithm named Camshift is arrived. Dynamic color distribution occur when face in video sequences are being tracked and the face moves so that the size and location of the probability distribution changes in time. The Camshift algorithm adjusts the search window size in the course of its operation and it relies on the zeroth moment information.
Camshift fundamentally increases the gradient of a back projected probability distribution calculated from re-scaled color histograms to discover the nearest peak inside an axis aligned window.

   To find the mean location:

   The back projected probability distribution at position x, y within the window $P(x, y) = h(I(x, y))$ computed from histogram of I.

   Zeroth order image moment, $M_{00} = \sum_x \sum_y P(x, y)$

   first order image moment,
   $M_{10} = \sum_x \sum_y xP(x, y)$; $M_{01} = \sum_x \sum_y yP(x, y)$

second order image moment,
$M_{20} = \sum_x \sum_y x^2 P(x, y)$; $M_{02} = \sum_x \sum_y y^2 P(x, y)$

Mean of the face $x_c = M_{10}/M_{00}; y_c = M_{01}/M_{00}$

Aspect Ratio = $(M_{20}/(x_c)^2)/(M_{02}/(y_c)^2)$

Updating window size using Aspect Ratio.

$w = 2M_{00}$ (Aspect Ratio) ; $h = 2M_{00}$ / (Aspect Ratio)

Updated window size = $s = 2\sqrt{M_{00}/256}$

Camshift has complexity order of (αN^2) where α is some constant, and the image is taken to be NxN. α is typically derived from the moment controls and the average amount of iterations of mean shift up to convergence. Scaling the area of computation to an area around the window size reduce complexity.
Here, Camshift algorithm is implemented with features (e.g., Color transformation, Back projection) and data structures (e.g., Histogram) available in OpenCV. It has inherent functions like (e.g., Meanshift, Camshift) available on general purpose packages. This function class compresses all required data structures and hides all transitional library called inside. Therefore OpenCV has easy and well-ordered programming.
After recognizing every detected face, the human face is tracked till 30 frames with assigned label.

5. Why these approaches?

- Viola-Jones Detector is a Feature based technique. Viola-Jones object detection framework is an object detection approach which provides real time detection of object this technique is extensively used in the field of face detection. Then again, It is computationally expensive. It may also fail to detect the face, when the subject turns or tilts his head.
- LDA is used due to below given reasons: Based on Fisher face projection approach to solve the illumination and expression variation Optimization in low dimension sub space Achieves well by maximizing the between-class scatter while minimizing the within-class scatter Search for to find information for best separation of the classes Effective for large training database
- K nearest neighbors is robust to noisy environment and effective towards large data sets.
- Camshift gives better results in noisy environment without filtering or adaptive smoothing. Noise removed by color model and discarding outlier exception by Camshift. Without any obstruction, it has advantages to deal with uncertain face motion because of natural scaling with the window as the face moves, handle distractions with no outlier apart from window size. Camshift avoid brightness of the pixels and use only hue from HSV algorithm. Hence it has higher accuracy in illumination deviations. Apart from this Camshift has one drawback that it

cannot distinguish the objects having same color. It tracks only the peak of the back-projected probability ignores color composition.

6. Experimental Setup

6.1. Database (Yale)

For face detection we will use the Haar Cascade provided by OpenCV. For face recognition we will the LBPH Face Recognizer. Append all the absolute image paths in a list image_paths. We will not read the image with the .sad extension in the training set. Rather, we will use them to test our accuracy of the training. Define images which will contain faces and Labels will contain the label that is assigned to the image. Read the image and convert to gray scale. Convert the image format into numpy array. Get the label of the image and then detect the face in the image. If face is detected, append the face to images and the label to labels. Return the images list and labels list. Adding faces to training set and perform the training. Recognize the images with the extension .sad into image_paths. On Yale database we are able to achieve almost 100 percent accuracy. There are no False Positive or False Negative cases in this experiment.

6.2. Mugshots

We first made the database with the pictures that had been captured from the different phones and laptop webcam. But the problem with this was that there was access background that disturbed the recognition. Thus we realized that we need a cropped proper database or add an algorithm which extracts the exact face from the picture and then compare it with the video frames.

The low resolution pictures also made a difference in the recognition accuracy. There were some misclassifications or false positives at times. Thus we thought on improving the database. We captured the high resolution images (720px) from the phone and used that for performing recognition.

6.3. Video

Initially we tried the recognition on the faces that were captured in the webcam which is very low resolution video. Considering the low resolution video, the speed had to be good, but when we tried the same on a 480px video the speed was affected significantly. After making a few changes as stated, we were able to get a good speed on a high resolution video. Thus we tried onto high resolution video and have performed the recognition on the same. On low resolution video we are able to get 90 percent accuracy and 98 fps speed and on HD Video we are getting 95 percent accuracy and 65fps speed.

7. Details of every approach

7.1. Relevant Issues

Issues are mainly faced during multiple face recognition. Either there is a swap in the labels, flickering in the labels between the right and the label that belong to the person beside, i.e. interchanging of labels between the faces present on the screen.

The probable reasons for this could be; labeling problem, problem in database, problem with the inter-class distance. After performing the binary classification (that is detection), the labeling for multi-class classification might be a problem. When there are multiple faces, the tracking algorithm might not be able to figure out which label it should take along with it, thus this might be the reason that there is a continuous swapping between the label of multiple faces. There might be a problem with the web cam since it does not ensure a good quality image fetching. In addition, our database is made of the images that we have clicked through the webcam itself. This gives a low resolution images which might not be a good database.

7.2. Identified problems

To debug, it is necessary that we calculate the distance between the two classes(i.e. inter class distance) and even see the distance of the new image fetched from the webcam to that of the two classes. Currently the algorithm make use of Euclidean distance which could be replaced by Mahalanobis distance.

Improvement in the database could be done. We could make a database with high resolution images and record our own good quality video, and perform recognition on that. Whenever we run the program, we are re-training the data set. This is not required. We can train the data set once and store it in a le and re-use it. We have to train the data set only when new data is added to it. Thus, not re-training could save time in processing.

7.3. Debugging

What we figured out is that it was a labeling issue in multi-face recognition. And we even improved our database. While assigning the labels in case of 2 or more faces, it got confused as to which face should be assigned with which label, so at times the labels got swapped. We figured out that it was not a recognition problem since the labels on each of the faces were only of the people present on the screen/video. If it had been a recognition problem then there would have been completely false recognition, i.e. the labels on the faces might have been of those that are not even present on the screen/video.

We made sure that we trained the data set only once, so that reduced our over-load and increased the speed. The added advantage we had from this was that we could now use high resolution pictures and videos for recognition since now speed was not an issue.

8. Conclusion

With the advancing technology there has been advancements in the recognition techniques. A newly emerging trend known as three-dimensional face recognition, claims to achieve improved accuracies. This technique uses 3D sensors to capture information about the shape of a face. This information is then used to identify distinctive features on the surface of a face, such as the contour of the eye sockets, nose, and chin. Another emerging trend uses the visual details of the skin, as captured in standard digital or scanned images. This technique, called skin texture analysis, turns the unique lines, patterns, and spots apparent in a person's skin into a mathematical space.

Thus the recognition has a long way to go. There is a huge scope in accuracy, speed, and most importantly making it online. Here we have made a small model for the face recognition using the tradition techniques of like LDA and LBP.

9. References

1. W. Zhao, R. Chellappa, P.J. Phillips, National Institute of standards and Technology, Face recognition: A literature Survey
2. Chengjun Liu and Harry Wechsler; Comparative Assessment of Independent Component Analysis (ICA) for Face Recognition. International Conference on Audio- and Video-based Biometric Person Authentication
3. "Multi-View Face Detection and Recognition using Haar-like Features", Zhaomin Zhu, Takashi Morimoto, Hidekazu Adachi, Osamu Kiriyama, Tetsushi Koide and Hans Juergen Mattausch, Hiroshima University
4. Riddhi Patel, Shruti B. Yagnik*2, A Literature Survey on Face Recognition Techniques
5. Juwei Lu, Kostantinos N. Plataniotis, and Anastasios N. Venetsanopoulos,Face Recognition Using LDA-Based Algorithms, IEEE TRANSACTIONS ON NEURAL NETWORKS, VOL. 14, NO. 1, JANUARY 2003
6. scholarpedia, K-nearest Neighbor, Leif E. Peterson (2009)
7. Md. Abdur Rahim, Md. Najmul Hossain, Tanzillah Wahid Md. Shafiul Azam, Face Recognition using Local Binary Patterns LBP, Bangladesh, 2013.

1. Results

In order to get high accuracy we have increased our training dataset. There might be a problem with the web cam since it does not ensure a good quality image fetching. In addition, our database was made of the images that we have clicked through the webcam itself. This gives a low resolution images which might not be a good database, reduces accuracy. Therefore we have opted for high resolution images. To increase the speed of face recognition we are train the database separately at the starting of the program and save those results. So, every time re-training of dataset is not required. Due to all the changes we have done, we are able to achieve 96 percent accuracy and approximately 28 fps speed.

Figura 1. We applied our algorithm on low resolution video. Speed is 88 fps. Accuracy that we are getting for face recognition is 85 percent. Because of tracking after every 10th frame detection leads to false positive cases.

We are using LBPH face recognizer on Yale database. Train the data except .sad extension images and taking .sad extension images as recognition input. In above image you can see the name original image and then the recognized class number. The accuracy that has been majored in this experiment is almost 100 percent. LBPH offers robust solution to illumination problem and the presence of localization errors.

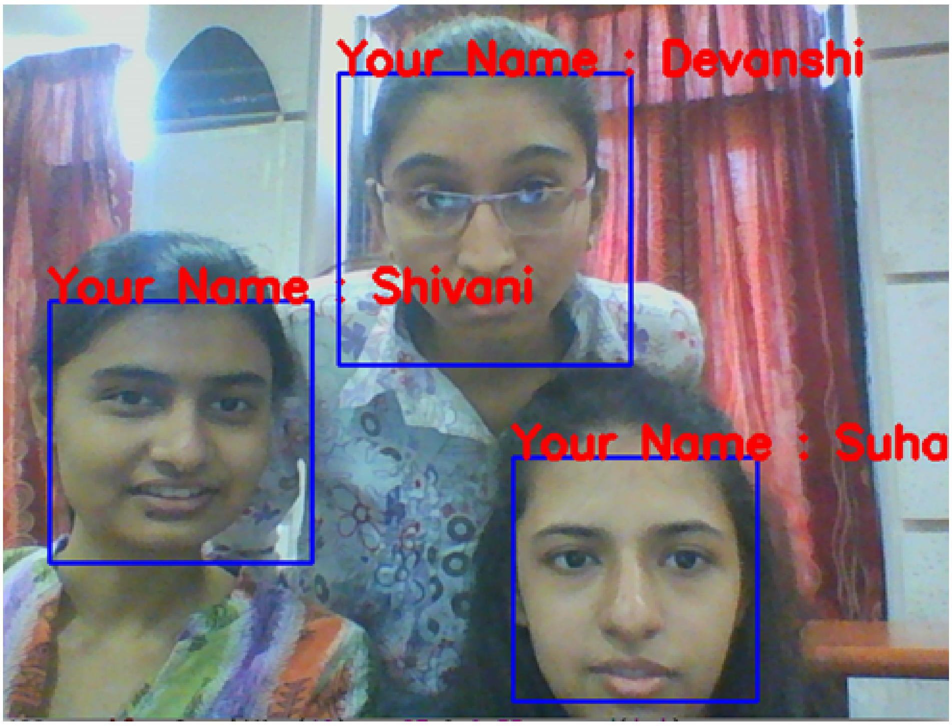

Figura 2. Here, by using given approach we are able to recognize multiple faces correctly. After applying tracking we are able to achieve very high speed. In the current scenario we have multiple variations for illumination effect, side faces, posture variations, etc.

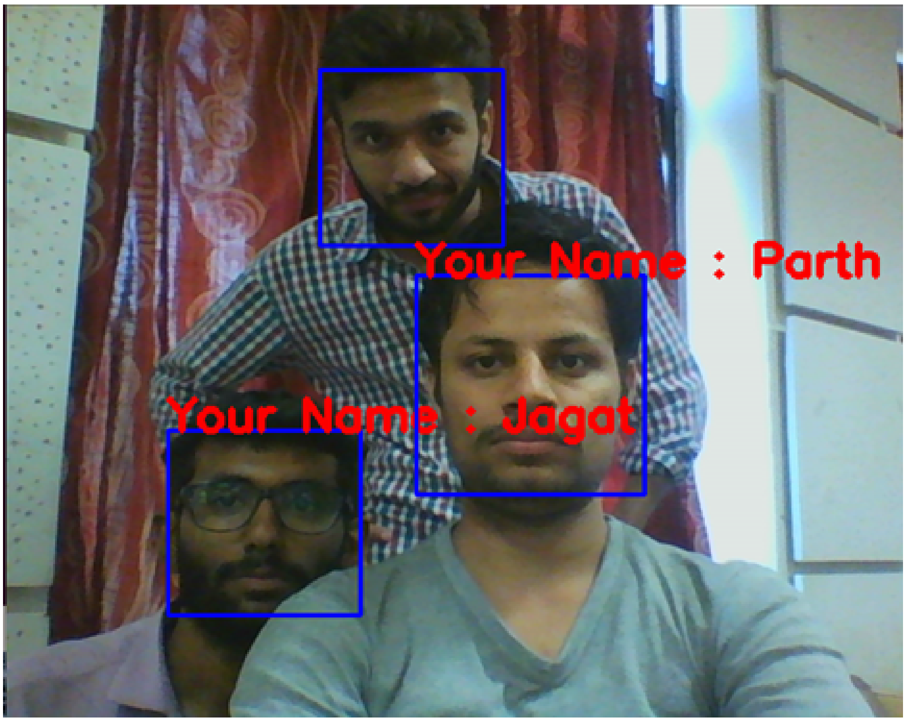

Figura 3. When any new person comes who is not in our database, they remain unrecognized, i.e no labeling

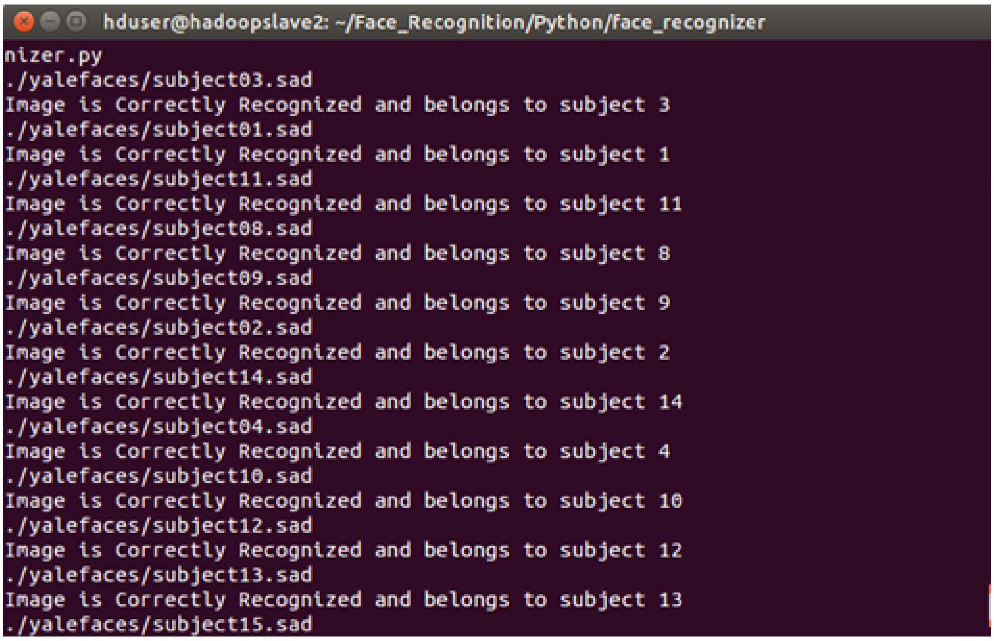

Figura 4. LBPH face recognizer on Yale database

Figura 5. Here, we are attempted to apply our approach onto full HD video. Due to high resolution and large image size we are getting speed of around 20 fps and accuracy of 75