Intelligent Face Recognition System

By

Ms Aalisha Sheth (11BIT002)

Ms Biyanta Shah (11BIT028)

Ms Aayushi Shah (11BIT029)

NIRMA UNIVERSITY
INSTITUTE OF TECHNOLOGY

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

AHMEDABAD - 382481

MAY 2015

ABSTRACT

This report provides a general overview of face recognition. Before undergoing face recognition, it is important for the system to detect the face and hence we first have reviewed face detection and then proceeded to face recognition. Face detection is the process of finding the face in the entire image and detecting the various facial features. In face recognition the system recognizes the face through various parameters like Eigen faces, Fisher faces, binary values etc. The faces in the training database are compared to the one in the test database to find whether the face exists in the database or not. We have used a few face detection and recognition algorithms along with their flaws and benefits and we have compared them with each other to find out the most favorable based on certain conditions. In the end we conclude by proposing a system, which we believe, provides better results.

CONTENTS

BIBLIOGRAPHY

LIST OF FIGURES

1. INTRODUCTION

1.1 Project Summary

1.1.1 Project Title

INTELLIGENT FACE RECOGNITION SYSTEM

1.1.2 Project Definition

The project aims at developing a system which would help in extracting facial features of the face, store the features and use the same for determining whether a test face has been matched or not.

1.1.3 Project Description

The system mainly involves 2 steps:

- Face Detection
- Face Recognition

Face Detection Methods are used in the system to capture the frontal face from the input frame and the same are used for creating the training database.

The training databases are further trained using different face recognition algorithms. When a test image is brought to check, the facial features of the test are matched against all the trained ones and accordingly we get to know whether a match of the test image in the trained set exists or not.

1.1.4 Objective

The objective of our project is to implement a face recognition system that accurately recognizes the face in the image and matches it with those in the dataset.

1.1.5 Scope

After implementing the basic algorithms for face recognition while recognizing faces from the database itself, we would like to push ourselves to achieve live face detection and recognition.

1.1.6 Hardware and Software Employed

HARDWARE :

Processor: Intel(R) Core(TM) i7-4500U CPU @ 1.80GHz 2.40GHz

Installed Memory(RAM) : 8.00GB

System Type: 64-bit OS,x64 based processor

SOFTWARE :

Matlab 2014a

2. LITERATURE REVIEW

Face recognition has been used in large range of applications such as access control, identity authentication and surveillance. The main problem this technology is facing is which parameters to use to detect the face and its features. Automatic detection of the facial features from the images is of grave importance in the field of security and face remodelling. Detection of various features like mouth, nose, eyes, lips in different lighting conditions and at different angles is a daunting task.

The key to face recognition begins with face detection. Face detection is the first step to be taken. In face detection, the face is detected from the image along with its various facial features. Three of the most common face detection algorithms are Viola Jones, Kanade-Lucas-Tomasi (KLT) and CAM-Shift algorithm. While Viola-Jones works on still images, the other two algorithms sought to detect the face in the entire video.

Viola Jones uses haar cascaded features to detect a frontal face and its features. The haar features are pre-calculated and used for every image where a face needs to be detected. The facial haar features calculated are similar for all human faces. For **Kanade-Lucas-Tomasi** algorithm eigen-values of the facial features are calculated and they are used throughout the video. The eigen-features are calculated when the face comes up in the first frame of the video. These features are then traced in the entire video and corresponding points are obtained. In this manner the entire face is detected during the duration of the video. **CAM-Shift** algorithm stands for Continuously Adaptive Mean shift algorithm [1]. It uses the skin tone or skin texture features or background color for face detection. CAM-Shift uses features, which remain constant for the length of the video. The texture or tone is extracted using hue channel extractor and with the help of a histogram the pixel values are calculated. Then the object is tracked throughout and the pixel values are matched with the created histogram thus enabling us to detect the face throughout the video. The underlying condition for the three algorithms is that it needs a frontal face at fist to successfully carry on the detection further.

Once the faces are detected, we need to recognize them. Recognition can be done in two ways. They can be recognized by a live camera database or by a test-train database, which is stored beforehand in the system. The three algorithms used here for face recognition are Principal Component Analysis (PCA), Linear Discriminant Analysis (LDA) and Linear Binary Pattern (LBP). These algorithms were written with improvement to the preceding one. PCA uses eigenfaces, LDA uses fisherfaces while LDA uses binary pattern. Each more superior than the previous one.

Principal Component Analysis as the name suggests picks out the principal components or features in the face and recognizes the face in the database. The covariance matrix is created and Eigen vectors and eigen values are extracted by mathematical operations. These Eigen values are then compared with the rest of the database and the match is found. LDA uses fisherfaces, here the interclass variance is maximized whereas intraclass is minimized. The images in the database are divided into classes for more efficient recognition unlike PCA. LDA is thus supervised classification whereas PCA is unsupervised classification. LBP uses the binary values of the pixels. The matrix is formed and the pixel values are recorded. These values are then converted to binary and a histogram is formed. Such histograms are formed all over. The histograms of this train database are compared with the images in the test database with respective histograms and the image is located in the database. Such a similar technique is used for live detection and recognition.

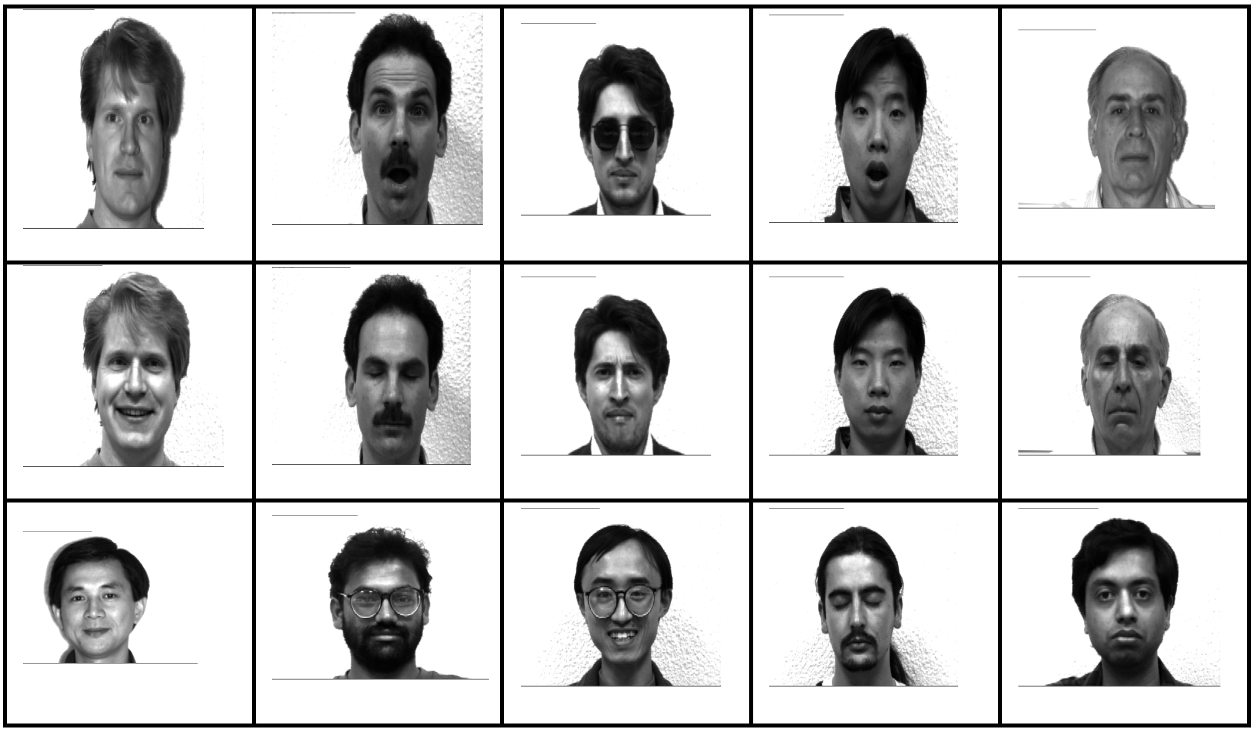
At the end, this project includes the comparison and analysis of the different face recognition and detection algorithms based on different parameters, which suggests which is better for certain circumstances.

3. FACE DETECTION

3.1 DATASET

Yale Dataset is used for the face detection methods to be carried in our project. It consists of faces for 15 different people, each having 11 different expressions, giving in all 165 images for testing of the method.

A sequence of images using the Face Database provided by the Audio Visual Technologies Group is also made and used for the algorithms, which take video as an input. Face Detection methods were also applied on random videos in order to find the advantages and disadvantages of the methods use in detection.



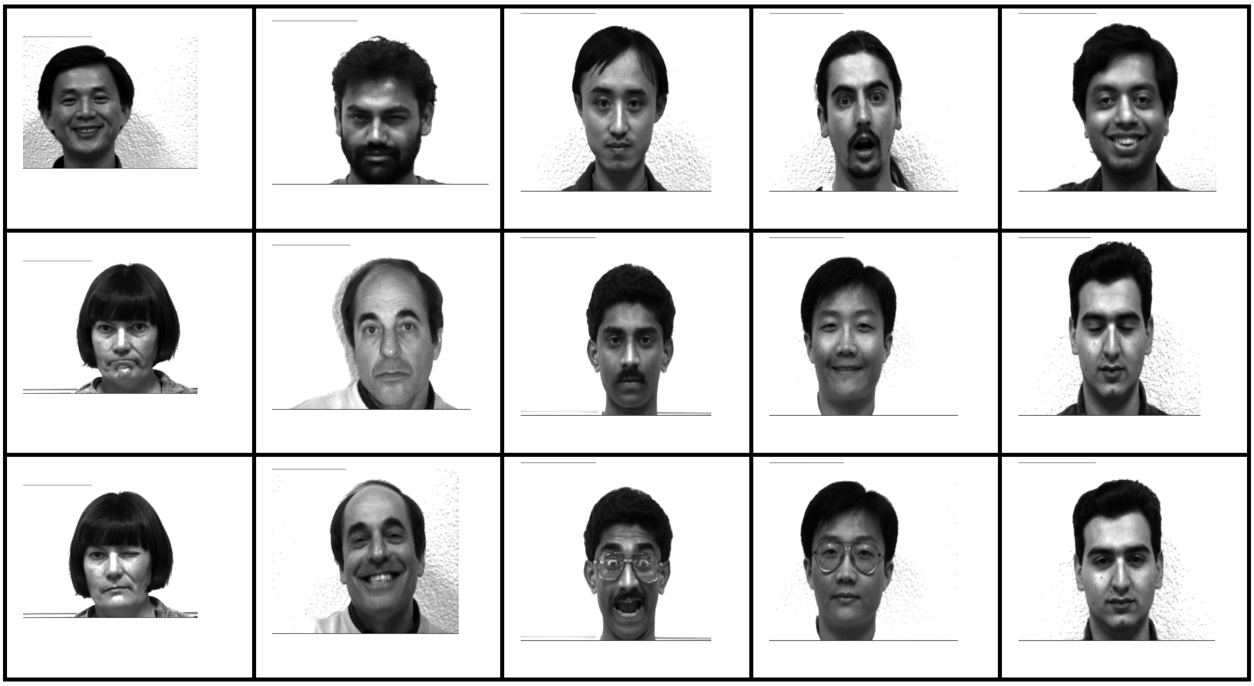


Fig 3.1: Yale Database [2]

3.2 Viola Jones

Viola Jones Face Detector method was the basic Face Detection Algorithm used to identify the frontal upright faces. The characteristics features of this method is its accurate detection rate and its goal to extract the faces apart from the non-faces in the image, thereby giving a low false rate [3].

METHOD :-

The Viola Jones Face Detector follows a sequence of 4 steps. Initially the input image is converted into an integral image. This conversion helps in reducing the processing time and does the same number of calculations on the input whatever is the size.

Making each pixel point equal to sum of all pixels above and to left of the corresponding pixel value forms integral Images [4].

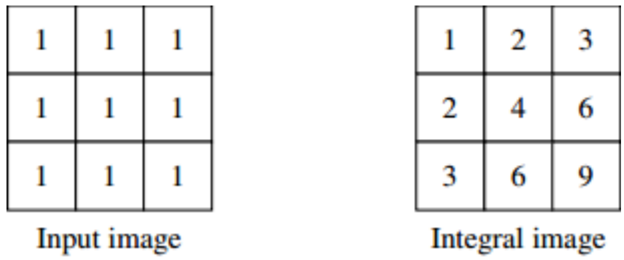


Figure 3.2: The Integral Image

Using such concept helps in calculating the sum of pixels within arbitrary sized rectangles in constant time.

Viola Jones Detector uses and scans a sub window for detecting of the faces and the analysis of the window is done using features containing two to three rectangles called the HAAR Feature

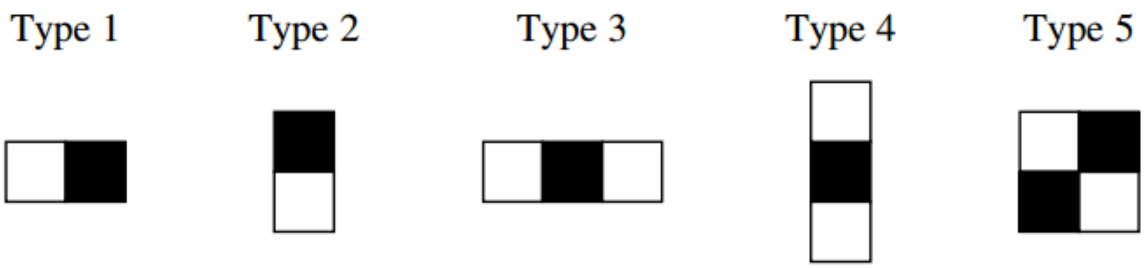


Figure 3.3: Haar Features

A single value for each Haar Feature is obtained by calculating the difference between the sum of the white rectangles from sum of the black. Taking in consideration all the possible sizes and positions of the features, a total of approximately 160000 features could be constructed which resulted in high computational efficiency. Hence, it was necessary to reduce the large number of repetitive features and improve the results by using a learning algorithm AdaBoost to extract the best features and train the classifiers that use them [4].

AdaBoost Learning Algorithm:

It is a machine learning algorithm which forms a strong classifier by the linear combination of weak classifier. If the value of the feature is more than the threshold value, the scanned sub-window is detected as a face or else a non face. As only a few of the all the features satisfy the criteria, the best features are selected.

Problem - Since to determine a weak classifier, every time the each feature is to be evaluated for the training images in order to find the best one, resulting in more time consumption.

Solution - The best feature is selected by the weighted error it produces, which is a function of the weights belonging to the examples. Here the weight of the correctly classified example is decreased and on the other hand for the misclassified it is kept constant.

Paul and Viola in order to increase the accuracy to a better level introduced the concept of the **Cascaded Classifier**.

The detector when tested on an input images with more than one faces determines that large amount of the evaluated sub windows would be non faces. Resulting, to discard the non faces instead of finding the faces.

Cascaded Classifier consists of stages each having a strong classifier. At each stage, it is determined that whether the sub window is a face or not.

If not a face it is immediately discarded and if possible chances of it being is a face, it is passed on to the next stage [4].

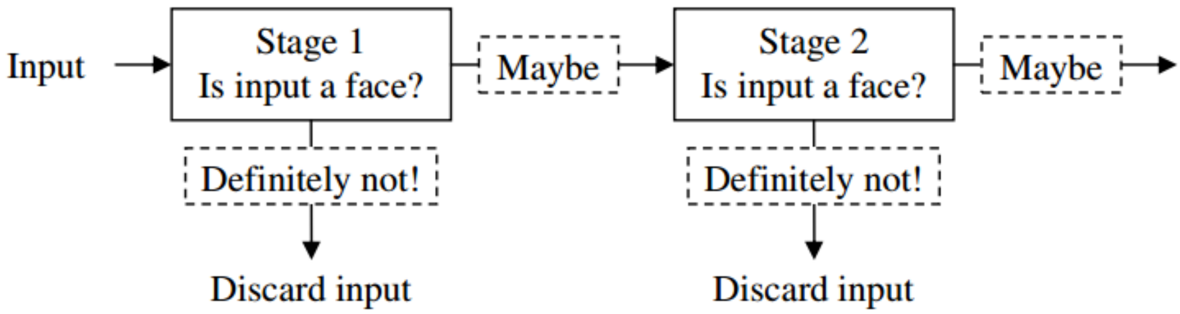


Fig 3.4: Cascade Classifier [4]

The basic disadvantage of the Viola Jones Detector is that the entire face must point towards the camera, that is it has to be a frontal face for detection and not tilted [5].

3.3 Kanade-Lucas-Tomasi

KLT face detection algorithm was the next proposed method which is used to detect the face using the feature points called the Eigen Values. It is applied on a video sequence and involves basic 3 steps.

Initially the first job is to detect the first frontal face from the video frame, which is done using the Viola Jones Face Detector. The idea behind using the Viola Jones Detector only once is to reduce the computational expense and not apply to all the frames. Also if applied on all frames it may fail to detect tilted faces and the output might not be what is required [6].

KLT algorithm extracts a set of the feature points from all the video frames. As the first face is detected, feature points are tracked and those points are used further in the video to track down the face. It computes displacement of features between the simultaneous video frames when the image brightness constancy constraint is satisfied and motion of the image [7].

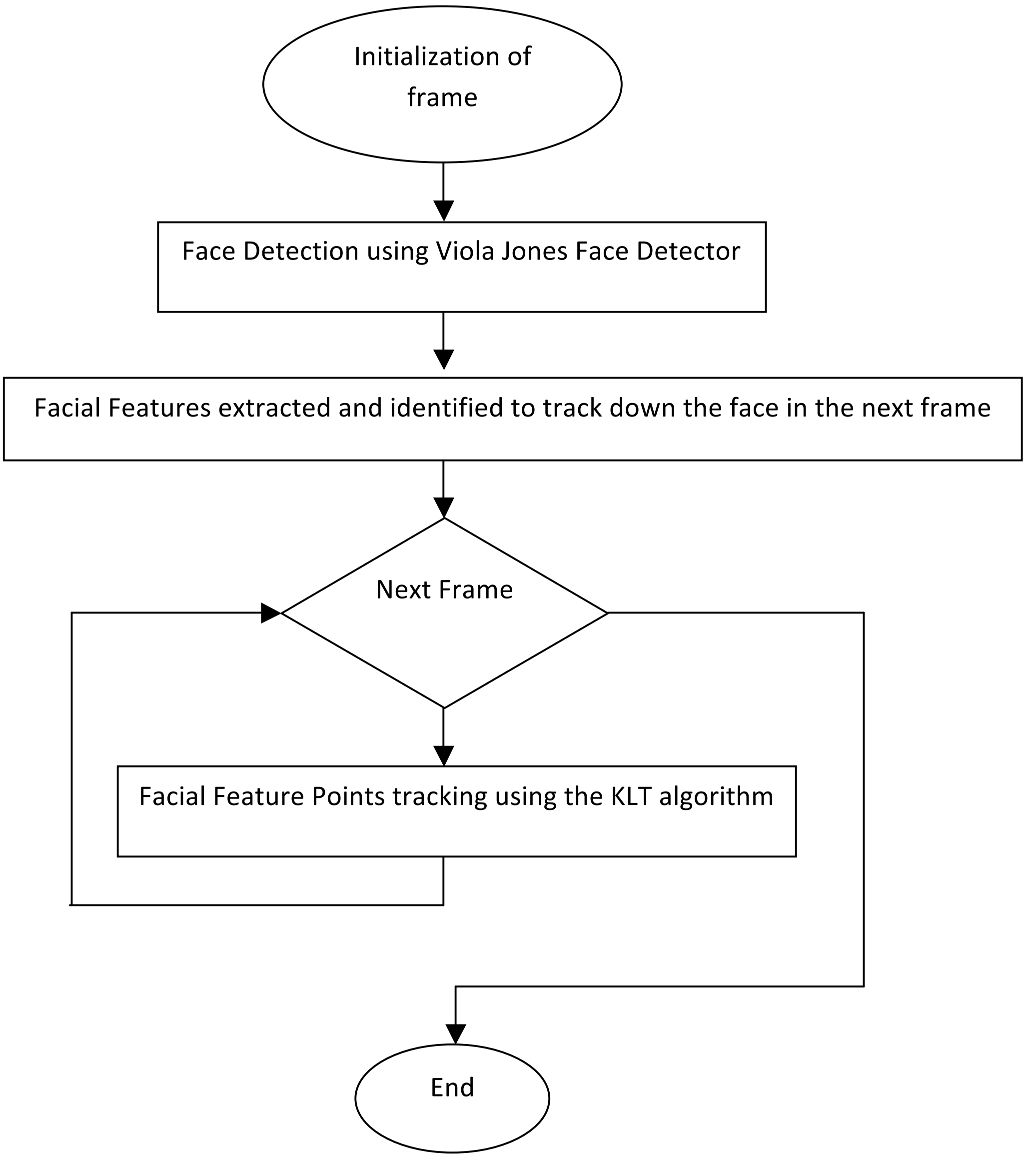


Figure 3.5: Flowchart for KLT

PROBLEM - Here initially as the Viola Jones Face Detector is used, if the face is tilted or rotated to any side, it fails to detect the face in the video frame. Also if the face is not upright towards the camera, no feature points would be tracked down and hence face detection would fail.

Hence for this algorithm to work correctly it is necessary that the person's face is frontal in the initial video frame for the face detection process.

3.4 Cam Shift

Cam Shift is one more approach for detecting faces from video frames just like KLT. The algorithm is basically based on the concept of Mean Shift but also has certain advantages, which weren't there in the Mean Shift.

The Mean Shift worked only on static probability distributions but not on dynamic, but Cam Shift solved this problem by readjusting the size of the search window for the next frame. Even if there is a fast motion, it is correctly able to track down the frame from the source.

METHOD:

Cam Shift algorithm works by tracking the hue of an object, in this case using the skin tone of the face in the source. Here any of the features can be used such as shape, texture or color, but the skin tone is used because a good contrast is obtained between the face and the background and remains the same even if the face is tilted or rotated.

Once a feature is selected, histogram of pixel values is constructed to track down an object. Using this histogram the object in the successive frames are detected and tracked [8].

In CAM-shift algorithm, a back projection image is obtained from the color histogram model of each image frame, the size of searching window is adjusted adaptively on the object’s size, and the center position of window is computed iteratively [8]. By extending the searching window’s size adaptively, the issue of the object losing instantaneously from the current window for motion acceleration is solved.

PROBLEM- The first problem that might occur is that only Hue Component of the HSV color space is being used. So if the face is not of a single shade and more similar shades occur, chances of tracking down the face is less [9].

The second one is that Viola Jones Face Detector is used initially, so if the face is not detected initially the entire process of tracking down the face goes wrong and required output is not obtained. Hence necessary that the person is facing upright to the camera in the initial frame itself.

SOLUTION- For solving the HSV color space problem, the histograms can be converted from 1D to 2D by considering the Hue as well as the Saturation component of the HSV color space [9]. This will enhance the tracking of the features in the successive frames even if it's not a single shade.

4. FACE RECOGNITION

4.1 DATASET

Yale Dataset is used for the face detection methods to be carried in our project. It consists of faces for 15 different people, each having 11 different expressions, giving in all 165 images for testing of the method.

Along with the Yale Database, the AT&T database also called the ORL Database [10] was used for implementing the Face Recognition Algorithms. ORL set contains faces of 40 different people and each person has 10 different facial expressions in all giving a database of 400 images.

Two sets were made for the implementation called the Training Set and the Testing Set. The training set contains images, which are trained and used in matching whenever a test image is brought. The test set includes faces, which are fed as an input to the system and matched against the training set images.

Figure 4.1: Sample ORL Database

A dataset of images was also created containing the faces of a student and was also used in testing the methods.

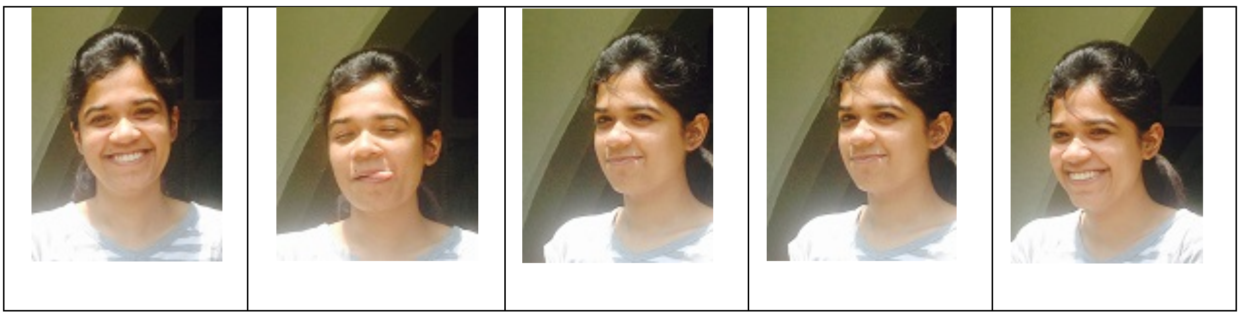


Figure 4.2: Sample Database

4.2 Principal Component Analysis

Face Recognition should be able to recognize the face in different illuminations, views and expressions. The main aim of **Principal Component Analysis** is to find the patterns in the data and to minimize the dimensionality of the database without the loss of information. So the basic outcome would be to represent the entire dataset onto a smaller dataset without any loss of data. To reduce the dimensionality we need to maximize the variance between the components. The basic concept would be to not focus on the common features of the face but collect the information of the face as a whole in varying conditions. Mathematically it means to calculate the eigen-values of the face.

PCA uses eigen-faces for the process of recognition. Eigen-faces are extracted from eigen features, which are obtained from faces of the people in the database. These are extracted from the covariance matrices created for each face. Each face is then represented in terms of eigen-faces. The number of faces in the database will be equal to the number of eigen-faces. By finding the eigen-vectors and eigen-values from the training dataset of images using mathematical operations on covariance matrices we get the eigen-faces. But since we need to reduce the dimensionality of the dataset, we do not take into account all of the eigen-faces. We choose the largest eigen-values and their corresponding eigen-faces, which would give us the maximum information helping us recognize the faces from the database. These values are known as the principal components. Following are the steps towards recognition:

1. To determine the mean face for the dataset of images.
2. Calculate the vector differences corresponding to each image and the mean image.
3. Using those vectors, calculate the covariance matrix for the corresponding dataset. Using the covariance matrix, eigen-values and eigen-vectors are found.
4. When a face is to be recognized from the test dataset, calculate the weights based on the test image and project it onto each of the eigen-faces.
5. The Euclidean distance between the weights of both the images is calculated which provides a measure of similarity between both the images. Using this value, the face is recognized from the train dataset.

Mathematical Steps:

Find the mean image for all the images in the training dataset. Let the images in the training dataset be T_1, T_2, T_3,…., T_M where M are the total images in the training dataset. The average of these images is calculated as:

$$\Psi = 1/M \ (\Sigma \quad T_n) \qquad (1)$$

Every image will differ from the mean image by a certain vector. This vector is denoted by ϕ.

$$\phi_n = T_n - \Psi \qquad (2)$$

These differences are used to calculate a covariance matrix. Covariance is the measure of how the datasets correlate between them. It is calculated as follows:

$$C = 1/M \left(\sum_{n=1}^{M} \phi_n * \phi_n^T \right) \qquad (3)$$

Here ϕ_n is the variance of each image and ϕ_n^T is the transpose of variance. The covariance matrix C is basically the product of the original matrix and its transpose.

From this covariance matrix, we find the eigen-vectors and eigen-values through the Jacobi method. Once eigen-vectors V_n are calculated, along with the variance of each image, it forms the eigen-faces.

Figure 4.3 The eigen-faces calculated for an image [11].

For the input test image the weight , W_n is calculated as follows:

$$W_n = V_n \text{ (Input test image} - \Psi) \qquad (4)$$

Where Ψ is mean image and V_n is eigen-vector.

PROBLEM- In PCA we look for maximum variance, and the variance is not only maximized between classes but also within the classes, thus functioning less efficiently than the normal correlation algorithm. We can improve the efficiency by removing the three most significant principal components but that would lead to loss of information that actually helps in differentiating faces.

4.3 Linear Discriminant Analysis

Linear discriminant analysis is another method that is used for dimensionality reduction. It has applications in statistics and pattern recognition as well. It finds a linear combination of features that separates or classifies the objects in two or more classes. If there are more than 2 classes, it may also be called Multiple Discriminant Analysis. Computation costs are notable reduced by dimensionality reduction. Also, over fitting can be avoided by it.

In LDA, the objects are projected onto a lower dimensional space while being separated into groups. This makes it easy to recognize which group a particular object belongs to. LDA is also known as supervised learning as the class label plays an important role here. It finds its application in face recognition as characterizing the faces into groups, makes it easy to recognize and match the face of a person to its equivalent in the dataset

In LDA, we find the axes that maximize the separation between two classes. The procedure for LDA includes finding eigenvectors, their corresponding eigenvalues, the within-class scatter matrix and the between-class scatter matrix. If all the eigenvalues have a similar magnitude, it means our data is projected on a good feature subspace. Otherwise, we keep the eigenvectors with higher eigenvalues as they represent more information about the distribution of data than those with lower eigenvalues. Eigenvectors with eigenvalues equal to or close to 0 are generally dropped.

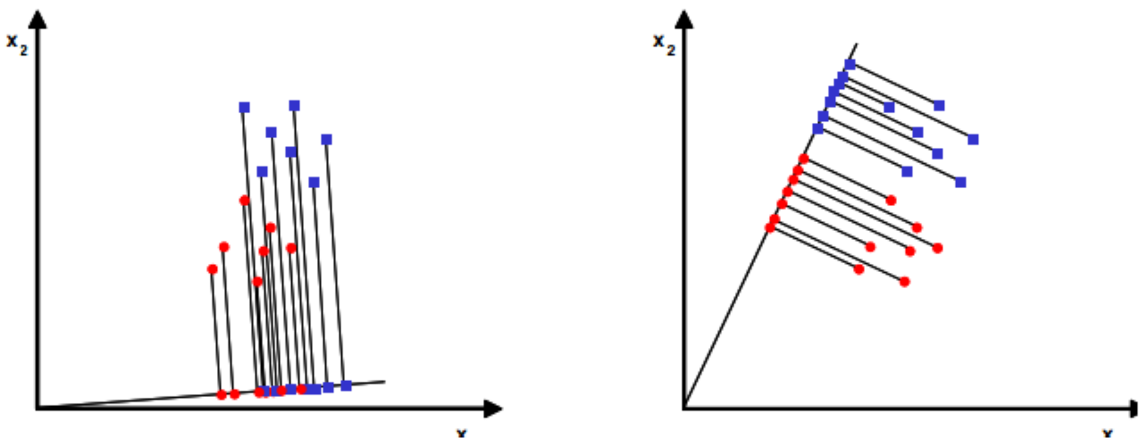


Figure 4.4: Scalar Projection of the Classes[13]

Mathematical Steps:

Assume a set of d-dimensional samples(x1,x2,.. xn), N1 belonging to class w1 and N2 belonging to class w2. We wish to obtain a scalar y by projecting the samples x on to a line.

$$y = W^T x \tag{5}$$

Compute the mean vectors for each class:

$$\mu i = 1/ Ni \Sigma_{x £ wi} x \tag{6}$$

The within-class scatter matrix is calculated as follows:

$$S_w = \Sigma_{i=1} S_i \tag{7}$$

$$\text{where } S_i = \Sigma_{x £ wi} (x - \mu i)(x - \mu i)^T \tag{8}$$

The between-class scatter matrix is calculated as follows:

$$S_B = = \Sigma_{i=1} N_i(\mu - \mu i)(\mu - \mu i)^T \tag{9}$$

Hence, to maximize the between class scatter and minimize the within class scatter we use the projection: $y = S_B S_W^{-1}.x$

Using this projection, the objects are projected onto a lower dimensional subspace where they are characterized into groups.

PROBLEM: It produces at most W-1 feature projections, where W is number of classes. It is a parametric method and it will fail if the information about the distribution is not in the mean but in the variance of the data.

4.4 Local Binary Pattern

LBP(Local Binary Pattern) is applied for facial expression analysis, background modeling and face recognition. This method gives a better performance and better result than traditional method like principle component analysis (PCA) used for face recognition. This is because LBP can deal with problems such, as illumination since it does is computation by taking the background into consideration.

LBP uses a local neighborhood. To define a central pixel LBP operator uses its 8 neighbors. If the neighborhood pixel is greater than the central pixel, the algorithm assigns a value of 1 to it. Otherwise, 0. Then going in an anticlockwise direction, it concatenates the values of the neighboring pixels to form a binary string. The decimal equivalent of that binary string is assigned to the central pixel. Similarly, all the pixels are assigned new values.

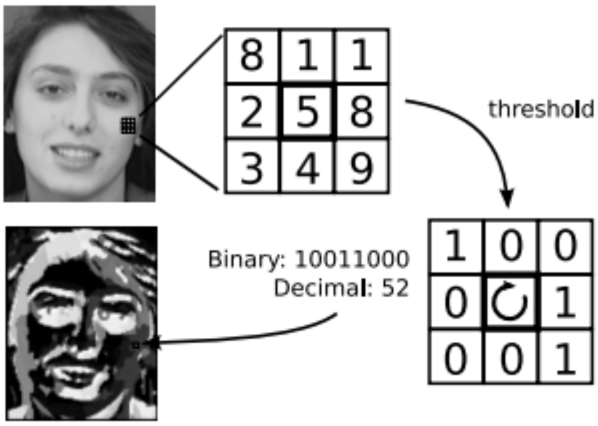


Figure 4.5: Assigning values to pixels using LBP

After assigning values to the pixels, a histogram for each image is created. Histogram is a graph that shows frequency of an element in the data set. X axis in Histogram represents the intensity/value of a pixel. Ex. For gray scale image, each pixel is represented by value between 0-255. Now, if we take histogram of an image it will have all unique pixel values on X axis and its corresponding frequencies on Y axis. Mostly similar images but slightly shifted in any direction will show high Euclidean distance if compared directly. But if we take histogram of the same and then compare histogram of both the images, then Euclidean distance between both will be much lower than previous case. Histogram has an advantage that it can solve the shifting problem that may occur during image capture.

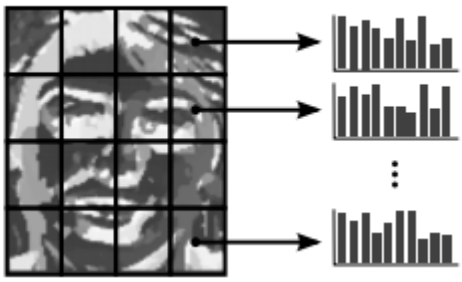


Figure 4.6: Histograms of an image

Histogram for the test image is created and is compared to the histograms of the images in the database. If for any comparison, the Euclidean distance falls below a certain threshold, then those images are said to be matching or belonging to the same person.

5. IMPLEMENTATION

IMPLEMENTATION OF FACE DETECTION ALGORITHMS

Face Detection algorithms like the Viola Jones, Kanade-Lucas-Tomasi and Cam Shift were implemented on various Datasets and video sequences. Following were the outputs of the implementation of Face Detection Algorithms:

1) VIOLA JONES IMPLEMENTATION

As stated earlier, Viola Jones Detector detects the frontal face from the input image and the 'cascadeobjectdetector()' function from the vision package of MATLAB is used to detect the face from the image. The function 'cascadeObjectDetector()' uses the basic Haar Features and Cascading to detect the face .The same function when provided with different arguments such as Nose, Mouth, Eyes etc extracts the facial parts too of the input image along with the detected face.

The following images are the results of face detection using Viola-Jones.

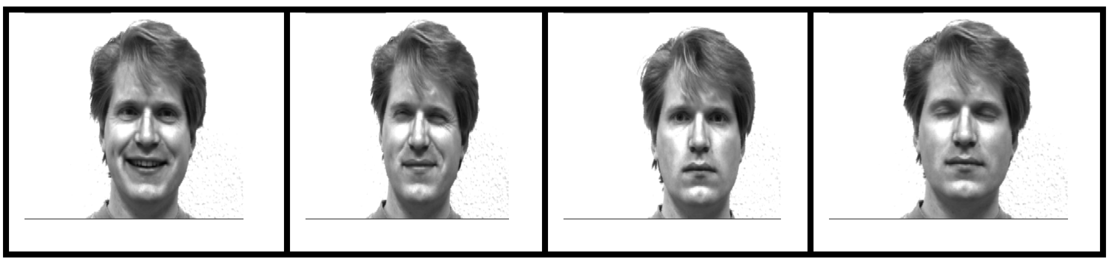


Figure 5.1: Sample Input images

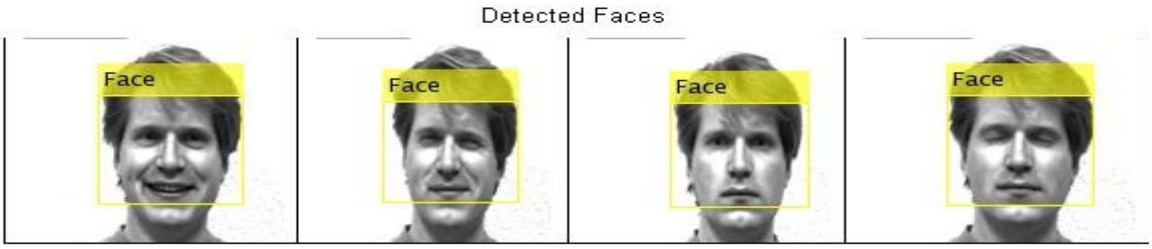


Figure 5.2: Frontal Faces Detected

Figure 5.3: Facial Organs Detected

The above illustrations indicate detection of face and the facial parts from an image which has a single person facing upright to the camera.

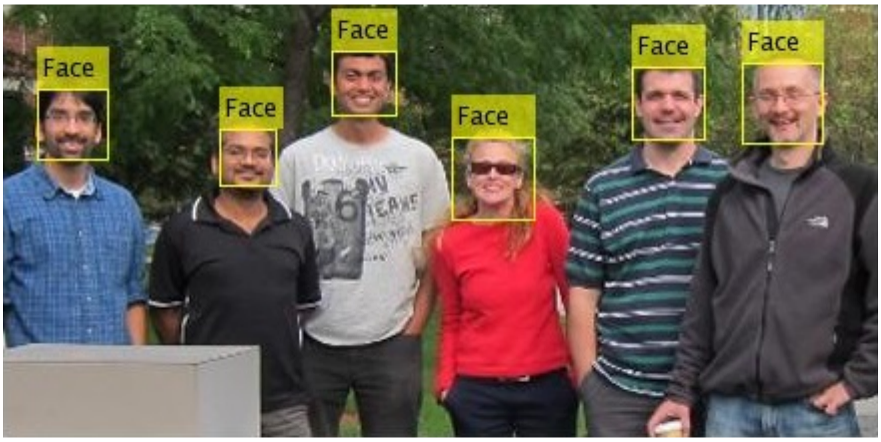


Figure 5.4: Detection of multiple faces

2) KANADE-LUCAS-TOMASI IMPLEMENTATION

KLT algorithm initially uses the 'cascadeObjectDetector()' function used by the Viola Jones Detector to indicate the face. Once the face is detected, the feature points are tracked down in that frame and the same points are used to detect the face whenever the face is tilted or moved sideways.

The feature points are obtained using the function called the 'detectMinEigenFeatures()'. Once the features in the current frame are obtained, the same values are used to find the corresponding point in the next frame of the video.

KLT here is applied on a video made from sequence of Images from the Face Database provided by the Audio Visual Technologies Group. The first face in the video must be frontal for rest of faces to be detected.

Figure 5.5: Sequence of images used in video [12]

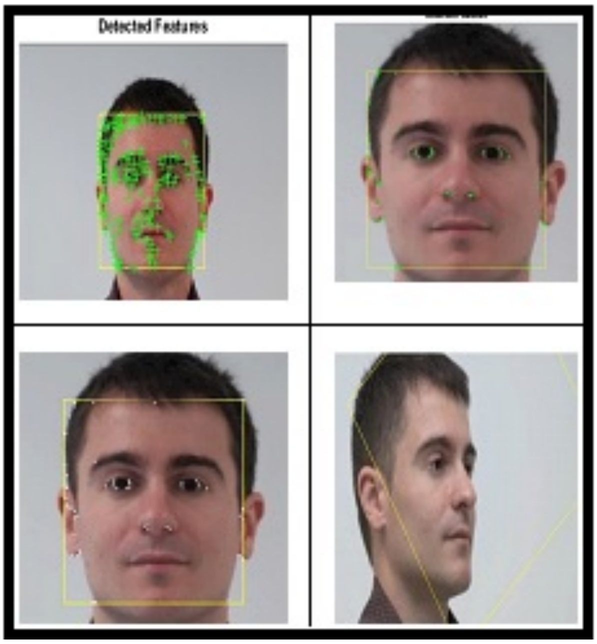


Figure 5.6: Results of Kanade-Lucas-Tomasi

3) Cam Shift Implementation

Cam Shift algorithm also works on an video input and uses the Hue Channel from the HSV color space. It uses a single feature to track down the face in the rest of the video. Features like texture, color etc anything can be used. Here skin tone is used as the chances of the skin color being constant is more compared to the other features.

With the parameter as skin tone selected, we use the 'vision.HistogramBasedTracker' for tracking. This tracker uses the Cam Shift algorithm which uses the histogram values generated to track down the face. In our case, the hue channel pixels are obtained from the nose of the detected region which are used to initialize the histogram.

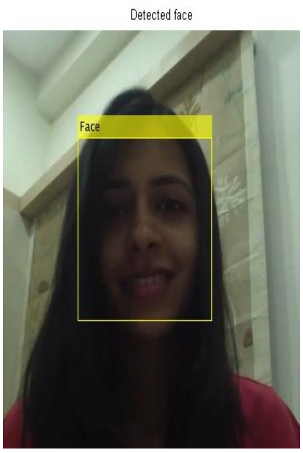


Figure 5.7: The first frame frame where face is detected

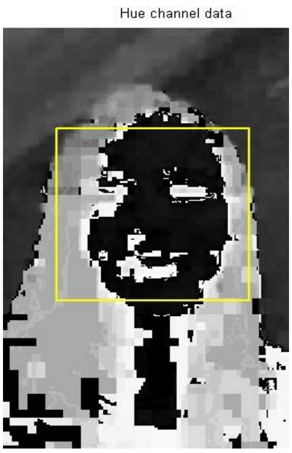


Figure 5.8: For skin tone, the converted to HSV color

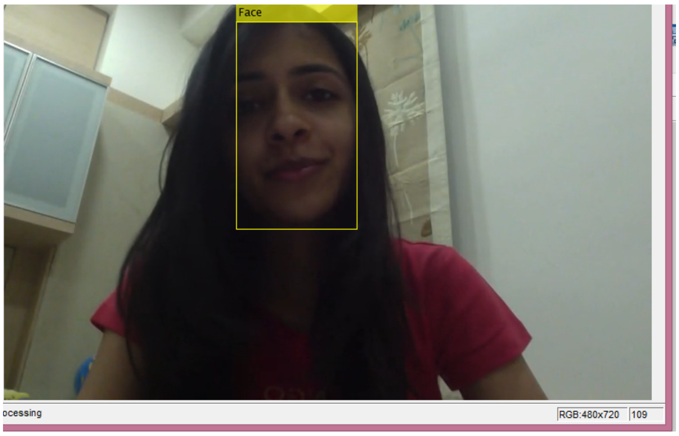


Figure 5.9: The face being detected in all the video frames it appears in

IMPLEMENTATION OF FACE RECOGNITION ALGORITHMS

Face Recognition algorithms like the Principal Component Analysis, Linear Dscriminant Analysis and Local Binary Pattern were implemented on 3 different Datasets discussed above. Following were the outputs of the implementation of Face Detection Algorithms:

1) PRINCIPAL COMPONENT ANALYSIS IMPLEMENTATION:

PCA is the most conventional method used for face recognition. PCA uses eigen-faces for the process of recognition which are extracted from eigen features obtained from faces of the people in the database we used. Each face is then represented in terms of eigen-faces. The number of faces in the database will be equal to the number of eigen-faces. Initially in our implementation we load the database we are going to use, calculate the mean face for the dataset and determine the vector differences corresponding to each image and the mean image. Covariance matrix is generated for the dataset and using this matrix the eigen-values and eigen vectors are found.

Finally after the training, when a test image is brought, the minimum distance is calculated between the training and test image and in this way the face is recognized.

Figure 5.10: Eigen vectors for ORL Dataset

Figure 5.11: Eigen Vectors for Yale dataset

In our implementation, the user needs to specify the test image from the dataset, which the user wants to test. After that, the training database is loaded, the eigen-values for all are calculated and the minimum distance is calculated to find the most exact match of the test image from the training ones.

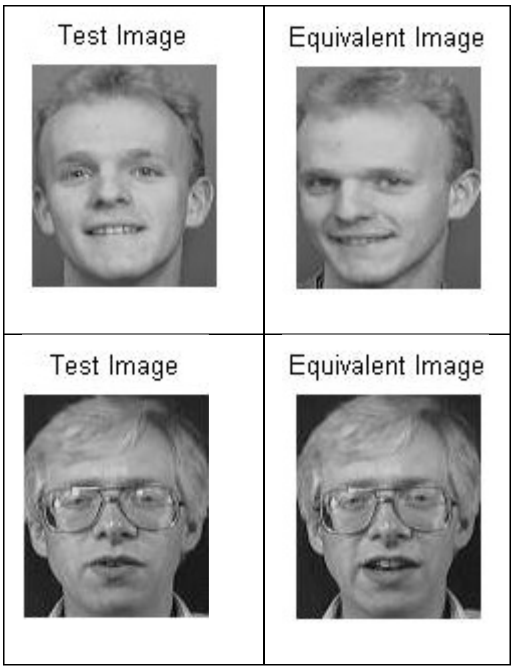


Figure 5.12: PCA Results for the ORL Dataset

2) LINEAR DISCRIMINANT ANALYSIS IMPLEMENTATION:

LDA was the next purposed method after PCA for the recognition of the faces. It also tried to eliminate some of the disadvantages of PCA and so came up with the concept of Fisher Faces.

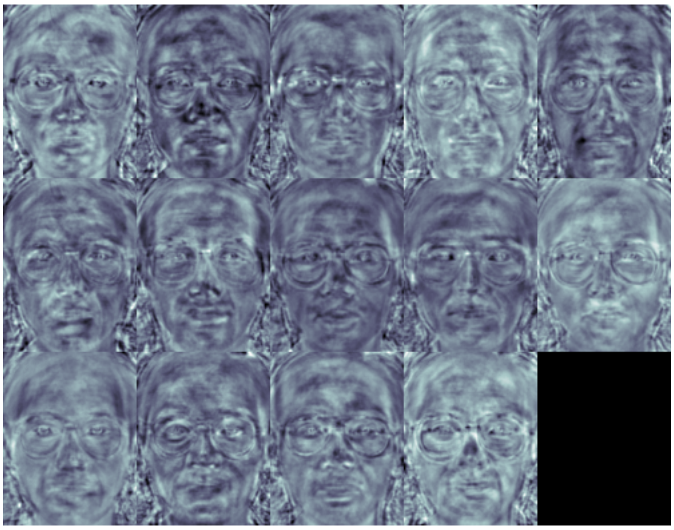


Figure 5.13: Fisher Faces for the few faces of Yale Dataset

3) LINEAR BINARY PATTERN IMPLEMENTATION:

In LBP, converting it to a gray scale the image is first normalized. Then the pixels of the image are reassigned values based on their surrounding pixels.

Using these values, the image is compared to the other images in the database and a match is found depending on whether the match exists in the database or not.

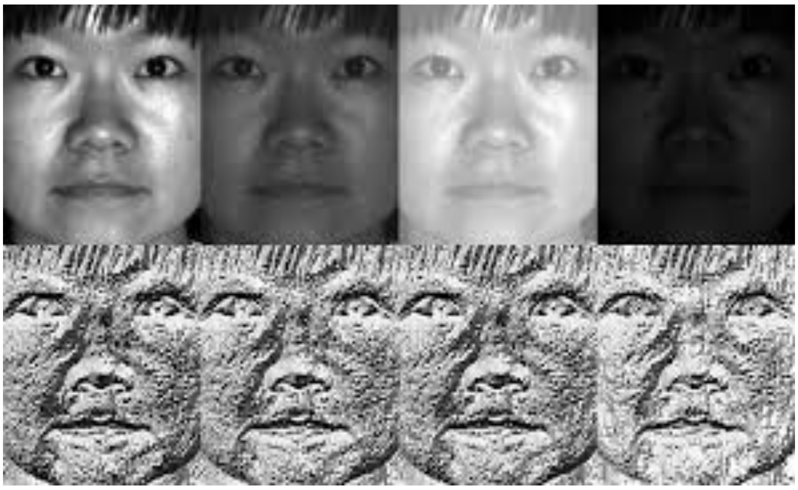


Figure 5.14: Faces Normalized by LBP Algorithm

ANALYSIS AND COMPARISON OF FACE DETECTION ALGORITHMS

The Viola Jones algorithm uses a series of steps for detection of the faces, but the disadvantage here lies that only frontal faces are detected. The tilted and rotated faces are not detected; also if the color tone, size or brightness is not good enough the frontal faces don't get detected. The problem also lies that the facial organs are not always correctly identified using the 'CascadeObjectDetector()' . The below images show the issues discussed above for the Viola Jones Algorithm.

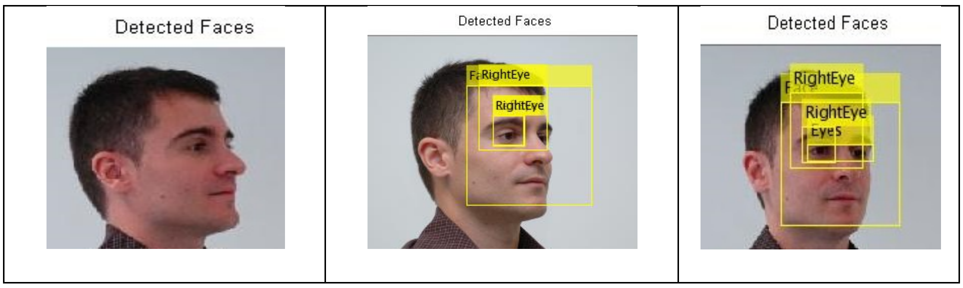


Figure 6.1: Flaws of Viola Jones Algorithm

This Algorithm is best used when the frontal face is clearly visible and recognized, along with all factors appropriate.

Kanade-Lucas-Tomasi (KLT) and CAM-shift algorithms both are used for face detection and feature tracking from the input video files. As discussed in the above sections, both use Viola Jones for face detection. Kanade-Lucas-Tomasi calculates Eigen vectors that help in detecting the face throughout the video whereas CAM-shift uses the skin tone as a medium here and converts the detected face image into the Hue channel data and plots it on the Histogram.

One of the major flaws is that if in the initial video frame, a frontal face is not detected, then both algorithms fail to spot any face in the entire video.

In Kanade-Lucas-Tomasi, if we have a slightly dark image, or an image with less contrast then it will fail to detect Eigen values on the face, which will then not detect tilted or rotated faces ahead in the video stream.

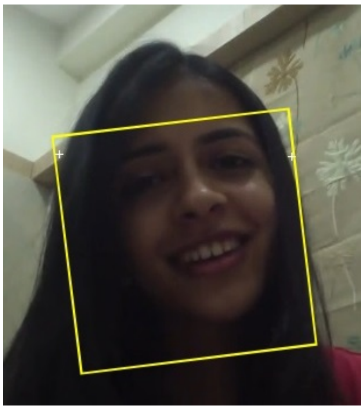


Figure 6.2: Lack of Detected Eigen Values

In CAM-Shift algorithm, the face will not be detected only if there is very high contrast. Once the face is detected, the algorithm converts it into hue channel data, which transforms it into an inverted grayscale image. Thus if there is less contrast the inverted image which will be the same, thus failing to detect the face along the video as it goes ahead.

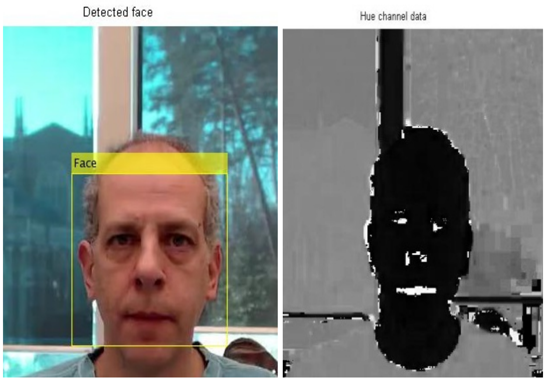


Figure 6.3: Very High Contrast thus no face detected

ANALYSIS AND COMPARISON OF FACE RECOGNITION ALGORITHMS

We have taken into account three face recognition algorithms namely Principal Component Analysis (PCA), Linear Discriminant Analysis (LDA) and Linear Binary Pattern (LBP).

PCA uses eigen-faces to recognize the faces. Eigen-faces are the largest valued eigen-vectors in the covariance matrix . These are found to be analogous to the Least Square (LS) solution. This ensures dimensionality reduction as well as no loss of information. Eigen-faces are calculated on the basis of maximizing the variance. They are calculated using the covariance matrix of the image dataset. While maximizing the variance it fails to acknowledge similarity between other images in the dataset and thus its efficiency decreases. Discarding the three principal components of the image can help removing this error, but removing them causes loss of information vital to the discrimination of faces from others. Thus sometimes PCA might detect another face which is similar to the test image but in reality that might not be the correct match.

LDA uses fisher-faces for face recognition. Fisher-faces recognize that the variation in each class lies in a linear subspace of the image. It distinguishes classes being convex and thus linearly separable. LDA is a more advanced version of PCA. Fisher-faces are insensitive to lighting variations and facial expressions than PCA. PCA had a flaw that it did not differentiate between the similarities of faces, LDA removes that flaw. LDA uses two matrices instead of one:

1. Between-class scatter matrix.
2. Within class scatter matrix.

These two matrices help taking into account the similarities and differences between classes. As LDA helps dividing the images into classes, it is known as supervised classification while PCA is known as unsupervised classification. LDA has one major flaw that it will not work if the information of the facial features is not in the mean but is in the variance of the data. It also just shows W-1 projections of eigen-faces where W is the number of classes.

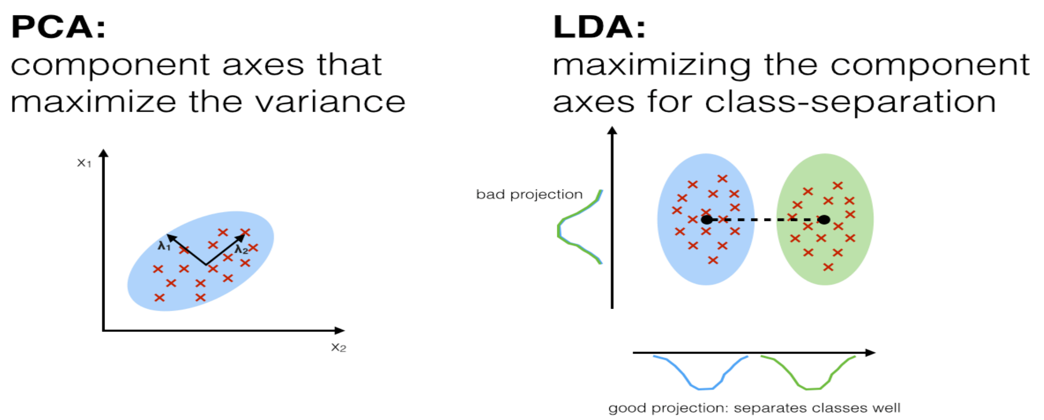


Figure 6.4: Comparison for PCA and LDA

To overcome the flaws we saw in LDA, we go through our next algorithm, which is Linear Binary Pattern (LBP). LBP does not use fisher-faces or eigen-values, it instead uses binary values of the pixels which is more reliable as that does not take into account the background color or illumination. LBP converts the pixel values into 0 or 1. The numbers greater than the central pixel in the grid turn to 1 and the rest to 0. After assigning these pixel values, a histogram for that grid is created and repeated through the entire face in the training dataset thus leaving very low margin for error. By using histograms and the binary values, the color variation problem is eliminated which was a very big problem in PCA and LDA.

Looking through all three algorithms, we can conclude that LBP performs more efficiently than the other two algorithms and thus is more superior for face recognition especially for bad illumination and different viewing points issues.

7. INTELLIGENT SYSTEM

A system being intelligent is its development to perform tasks that require human intelligence. Till now we have worked on face recognition algorithms where a single input test image is given and the image is matched from the training dataset. In our intelligent system we propose a different lookout.

Our system would detect all the faces found in the picture, not just a singular face but multiple, it would then match these multiple faces in the training dataset and find a match for all of these. The steps for this algorithm would be as follows:

1. Detect all the faces in the image using Viola Jones face detection.
2. Once done, using bbox2points() we can get the co-ordinates of the faces that are detected.
3. These co-ordinates will help us to extract the faces.
4. All the faces are saved. These also act as an input in the training dataset.
5. If the images are already present in the training dataset then it will give a suitable match for the faces.
6. If a new face is given as input then the training dataset will consider the new face as an added entry and will output an unknown match.

7. If this new face appears again, the training dataset recognizes it from the previous entry and then will recognize the face as one of the dataset itself giving an appropriate match.

Thus in addition to detecting and recognizing multiple faces from an image, this algorithm also trains and learns by itself for unknown images.

BIBLIOGRAPHY

[1] Allen, J.G., XU, R.Y.D and Jin J.S. (2004) , “Object Tracking using CAM-Shift algorithm and Multiple Quantized Feature Spaces”, In Proc. 2003 Pan-Sydney Area Workshop on Visual Information Processing (VIP2003), Sydney, Australia. CRPIT, **36**. Piccardi, M., Hintz, T., He, S., Huang, M. L. and Feng, D. D., Eds. ACS. 3-7.

[2]“Yale Face Database”, Retrieved from http://vision.ucsd.edu/content/yale-face-database.

[3] Paul Viola (2001), “The Viola Jones Face Detector”, Retrieved from http://www.cs.ubc.ca/~lowe/425/slides/13-ViolaJones.pdf

[4] Kongens Lyngby (September 2008), “ Implementing the Viola-Jones Face Detection Algorithm”, Retrieved from http://etd.dtu.dk/thesis/223656/ep08_93.pdf

[5]”Viola-Jones Object Detection Framework”, Retrieved from http://en.wikipedia.org/wiki/Viola%E2%80%93Jones_object_detection_framework

[6] Carlo Tomasi and Takeo Kanade (April 1991), “Detection and Tracking of Point Features”, Retrieved from https://www.csie.ntu.edu.tw

[7] Sudipta N Sinha et al. (May 2006) “GPU-Based Video Feature Tracking and Matching”, Retrieved from https://cs.unc.edu.

[8] Adrian Rosebrock (August 8, 2014), "Tutorial: Using Cam Shift to Track Objects in Video", Retrieved from http://www.computervisiononline.com/blog/tutorial-using-camshift-track-objects-video

[9] Issac Gerg, Adam Ickes, Jamie McCulloch (December 7, 2003), "CAMSHIFT Tracking Algorithm", Retrieved from http://www.gergltd.com/cse486/project5/.

[10]"AT&T face database", Retrieved from http://www.cl.cam.ac.uk/research/dtg/attarchive/facedatabase.html.

[11]"Eigen-faces big", Retrieved from http://kixor.net/school/2008spring/comp776/assn3/eigenfaces-big.png. Last Viewed on March 29,2015.

[12] Audio Visual Technological Group, "Face Database", Retrieved from https://gtav.upc.edu/research-areas/face-database.

[13] "LDA theoretical", Retrieved from http://research.cs.tamu.edu/prism/lectures/pr/pr_l10.pdf

PLAIGARISM REPORT

Turnitin Originality Report

Intelligent Face Recognition by Biyanta Aayushi

From Intelligent Face Recognition (Semester-6 Project)

- Processed on 10-May-2015 18:58 IST
- ID: 539954373
- Word Count: 6719

Similarity Index

11%

Similarity by Source

Internet Sources:

8%

Publications:

4%

Student Papers:

3%

sources:

1 2% match (Internet from 02-Apr-2012)

http://dhruvdave.com/wp-content/uploads/2011/04/FinalTheses.pdf

2 1% match (Internet from 05-Jan-2014)

http://www.antiessays.com/free-essays/117606.html

3 1% match (Internet from 12-Jun-2009)

http://research.cs.tamu.edu/prism/lectures/talks/nose00.pdf

4 1% match (Internet from 29-Jan-2007)

http://crpit.com/PublishedPapers.html

5 1% match (publications)

Xiaojie Li. "Algorithm of object hovering detection and tracking based on moving track", 2011 IEEE International Symposium on IT in Medicine and Education, 12/2011

6 < 1% match (Internet from 12-Jun-2009)

http://www.ohiolink.edu/etd/send-pdf.cgi/YOUSSIF,%20ROSHDY%20S..pdf?acc_num=ucin1085714219

7 < 1% match (student papers from 13-Apr-2012)

Submitted to University of the West Indies on 2012-04-13

8 < 1% match (student papers from 17-Sep-2014)

Submitted to Singapore Management University on 2014-09-17

9 < 1% match (Internet from 10-Jul-2014)

http://ijsiet.org/IT2013/2013_142.pdf

10 < 1% match (Internet from 04-Aug-2010)

http://www.dtic.mil/cgi-bin/GetTRDoc?AD=ADA424713&Location=U2&doc=GetTRDoc.pdf

11 < 1% match (Internet from 09-Apr-2010)

http://www.eqb.state.mn.us/pdf/FileRegister/04-83-LWECS-GMW/GMLLCapplication.pdf

12 < 1% match (publications)

P. Marasamy. "Automatic recognition and analysis of human faces and facial expression by LDA using wavelet transform", 2012 International Conference on Computer Communication and Informatics, 01/2012

13 < 1% match (paper)

paper

14 < 1% match (student papers from 24-Apr-2015)

Submitted to Hong Kong Baptist University on 2015-04-24

15 < 1% match (Internet from 26-Jan-2015)

http://buff.starenka.net/in

16 < 1% match (student papers from 21-Jan-2015)

Submitted to Higher Education Commission Pakistan on 2015-01-21

17 < 1% match (Internet from 13-Mar-2015)

http://www.raspberrypi.org/forums/viewtopic.php?f=38&t=85755

18 < 1% match (publications)

Joseph Kee-Yin Ng. "A Selector Method for Providing Mobile Location Estimation Services within a Radio Cellular Network", First International Conference on Availability Reliability and Security (ARES 06), 2006

19 < 1% match (publications)

Ping Xiao. "Regularized Scatter Measure for Linear Feature Extraction", Second International Conference on Innovative Computing Informatio and Control (ICICIC 2007), 09/2007

20 < 1% match (Internet from 12-Sep-2011)

http://en.wikipedia.org/wiki/Viola%E2%80%93Jones_object_detection_framework

21 < 1% match (Internet from 14-Oct-2010)

http://www.scribd.com/doc/33929882/Wireless-sensor-network

22 < 1% match (Internet from 05-Nov-2014)

http://www.ijritcc.org/download/Performance%20Evaluation%20of%20Face%20Recognition%20Algorithms.pdf

23 < 1% match (Internet from 30-Nov-2014)

http://138.100.76.11/visionguided2/sites/files/wfmfolder/phdthesiscarolmartinez_lowres.pdf

24 < 1% match (Internet from 18-Oct-2009)

http://www.logincee.org/file/18368/library

25 < 1% match (Internet from 23-Mar-2015)

http://www.reportworld.co.kr/search/result.html?where=relatekwd&kwd=face+classification+PCA

26 < 1% match (publications)

Alok Sharma. "A Gradient Linear Discriminant Analysis for Small Sample Sized Problem", Neural Processing Letters, 02/2008

27 < 1% match (Internet from 15-May-2014)

http://www.bogotobogo.com/cplusplus/files/opencv2refman.pdf

28 < 1% match (publications)

Young Jae Lee. "Interface of Augmented Reality Game Using Face Tracking and Its Application to Advertising", Communications in Computer and Information Science, 2010

29 < 1% match (publications)

Ruiz-del-Solar, J.. "Illumination compensation and normalization in eigenspace-based face recognition: A comparative study of different pre-processing approaches", Pattern Recognition Letters, 20081015

30 < 1% match (Internet from 26-Jun-2010)

http://www.mightystudents.com/essay/effect.generational.poverty.98941