PORTABLE ATTENDANCE MONITORING SYSTEM USING FACE RECOGNITION

Atchut Vardhan K, Assistant Professor, IT Department, MVGR College of Engineering, Vizianagaram-535005

Srikanth V.M.K, Student, IT Department, MVGR College of Engineering, Vizianagaram-535005

Abstract

Automation has become the order of hour.80% of the things have been automated in every sector. So, the demand for automation is increasing at a tremendous rate. In the recent past, face-recognition and detection has evolved rapidly. Using a Face recognition system is way better than other biometric systems because, it involves a moderate computation. This paper is about monitoring attendance using face recognition, which is done using Open Computer Vision (Open CV) programmed with Python. Detection and Recognition of face leads to the generation of attendance reports and the reports are published.

Keywords: Face-recognition; open CV ;Attendance-monitoring; Haar-cascade; Ada-boost; Eigenfaces; Principle Component Analysis(PCA); Fisherface; Linear Discriminant Analysis(LDA); Local Binary Patter Histogram(LBPH);

1. Introduction

The face recognition technology began in early 1977 with the first automated system developed by Kanade using a feature vector of the human face. In 1983 Sirovich and Kirby introduced Principle Component Analysis (PCA) for feature extraction. By using PCA [1][10] in 1991 Turk and Pentland developed a method called Eigen-face [2][3][11]. And later in 1994, a pattern called Local Binary Pattern Histogram (LBPH) [4][5][12] was introduced. In 1996 Linear Discriminant Analysis for Dimensional Reduction was developed as Fisher-face [6][11][14] which was developed using Linear Discriminant Analysis (LDA) [7][13]. A face detection technique was developed by Viola and Jones called as HAAR Cascades [8][9] and Ada Boost. In this project, we use HAAR Cascades algorithm and any among Eigen-face, Fisher-face, and Local Binary Pattern Histogram is used.

In this paper starting with the face detection and recognition, how these techniques are used to monitor the attendance is explained. First, we need to detect a face and then recognize the face. Recognition leads to a unique identification. For the detection and identification, there are algorithms as mentioned before.
In this, we use HAAR Cascades for face detection and Eigen-face for face identification.

2. Face Detection Algorithm

For face detection, we use HAAR Cascades [8][9] classification. This Cascade is a machine learning process where a huge amount of data is first trained with images with faces and images without faces. Initially, it calculates the pixels of the area. For example, consider the figure

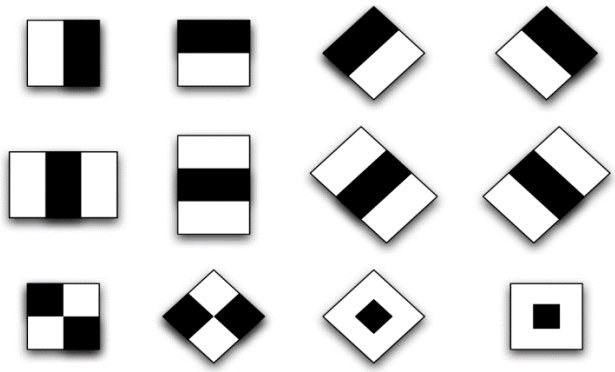


Fig.1: The HAAR features are the difference between the sums of pixels of the dark region and sum of pixels of the light region.

Now consider the below picture

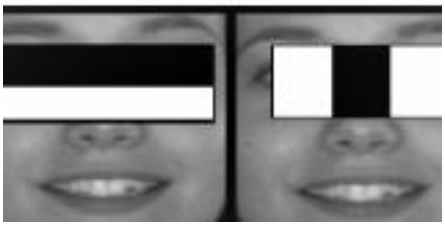


Fig. 2 (a) (b)

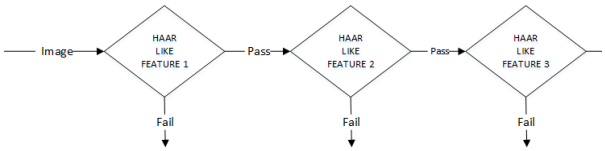
The eyes region in the image 2(a) is darkened and the below region is lightened. So the sum of pixels is calculated. The lightened region is considered to be a feature used for detection. So the nose part is considered to be a feature used for detection. From image 2(b) the eyes region is lightened and the nose bridge is darkened. So the sum of pixels is calculated and the eyes part is used as a feature for

detection. In this way, many features are trained by using this method, of a face to the machine. Now by finding all the pixel numbers of the faces, the detection is done.

Consider a 24*24 window where we get 160000 features. It is difficult to calculate all those features. But suppose if we take only 4 features of any image and train the cascade the computation is better. But if a feature appears where the cascade is not trained then the difficulty in detection increases. So this problem is achieved by "Ada Boost"[8][9].

The Ada Boost[8][9] classifier is trained on each and every feature of the image in the training data. For each feature, there will be a threshold value where it classifies the images with face and images without a face. All features cannot detect exactly. So by considering features with a minimum threshold rate, the detection will be easy. This classification process is done several times, and all the classification weights with the minimum threshold are noted. Now, the classifier with a lower rate is again combined with the feature having a minimum threshold and the process is continued until required accuracy is obtained. According to a survey 200 features of an image gives 95% accuracy in detection.

This is a time consuming process. So the other way is first to recognize the part which is not a face. By eliminating that region number of features will be reduced. A concept called Cascade where the features are
grouped into different stages is introduced. If there are 200 features instead of applying all features, a classifier is first trained on some features in the first stage. If it passes, then remaining features are tested in the second stages on the same classifier and this process continues. But due to any one feature, if the first stage fails, discard it. Remaining features are not considered. So we take another classifier, and in this way the classification is done.



HAAR Cascade of classification

Fig. 3

3. Face Recognition

For the recognition of a face, algorithms like
1) Eigen face [2][3]
2) Fisher face [6][11][14] can be used.

3.1. Eigen Face

The recognizer should be trained with images in multiple ways with multiple images of a person. For a face to be recognized, all features are not required. Some features like eyes, nose, ears etc. are key features to any face. So, the pictures having individual features are known as Eigen Faces. These features are not trained in a normal way. They are turned into gray-scale images by highlighting a single feature in the face.

For example, the Eigen faces are shown as

Fig. 4 Eigen Faces

From the above figure, all features may not be clear but some features show clearer. So based on these features the recognition is done. This feature extraction in Eigen Face is done by PCA (Principle Component Analysis).
So how does PCA[1][10] works? Initially Arithmetic mean is calculated for all Eigen Faces of called as mean. Each Eigen Face will have a weight. All the weights are subtracted from the mean and stored in a vector. How the Eigenvector is calculated is not mentioned in this paper. The computation of PCA[1][10] is a mathematical concept which is not discussed in this paper.

PCA Algorithm

Input: -
K: A set of data of Eigen Faces.
Process: -
Step 1: Arbitrarily choses k Eigen faces from the training data.
Step 2: Repeat
Step 3: The PCA computes the variance of the Eigen Faces.
Step 4: If the maximum variance is obtained, store those features and if due to any one Eigen Face the variance is minimum, then those set of features are discarded.
Step 5: PCA again computes by considering the discarded features with the features with maximum variance.
Step 6: Until, the process reaches saturation point when the discarded features computed with features with high variance, the variance will be minimum.
Output: -
A set of k-Eigen faces with high variance.

The procedure for recognition is explained in the below flow-graph

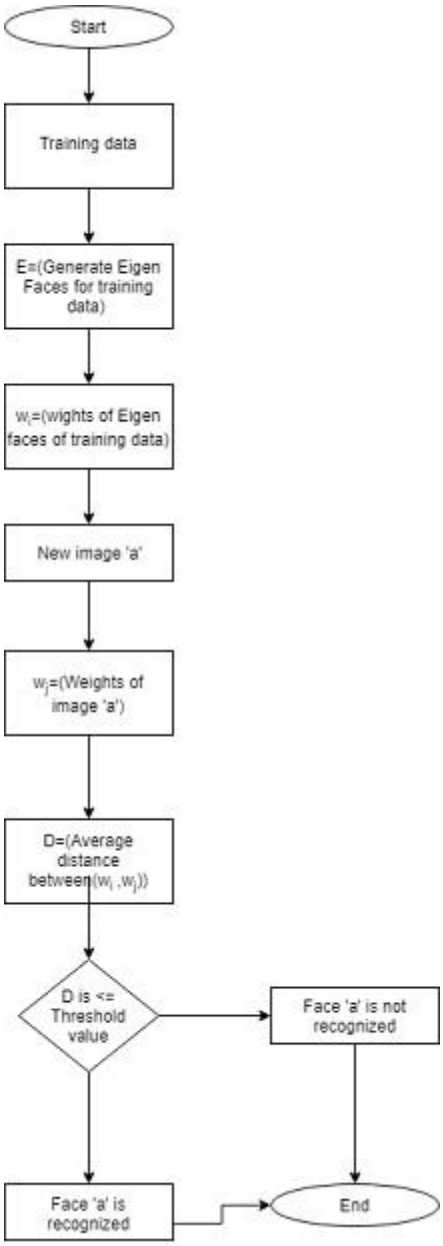


Fig. 5

• Initially, the originals faces are considered and stored.
• Now all the faces are converted into Eigen Face.
• After converting the face, the weights of each face is calculated and stored. This is considered to be as w_i.
• Now a new image is taken.
• Weights of the new image are calculated and stored. This is considered to be as w_j.
• Now the comparison is done by considering Euclidean Distance between the weights (w_i-w_j). Now an average is calculated on all the distances D. If the value D reaches the threshold, then it recognizes the face else it will not recognize the face.

The process is great. But there is a problem in this algorithm. In different illumination conditions, this will not recognize the face perfectly. Consider the below image,

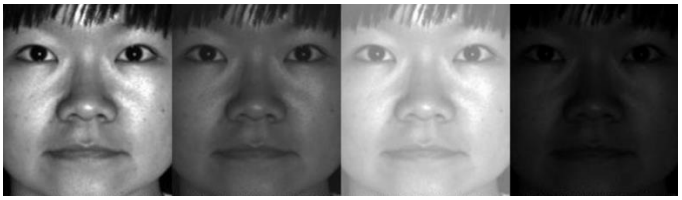


Fig. 6

Now the recognizer cannot recognize the face. So another algorithm called Fisher Face covers this drawback. The Fisher face is discussed in the next section.

3.2. Fisher Face

In this algorithm, the light is also considered as a feature. The Fisher Face [6][11][14] is also based on Eigen Faces. But these fisher faces are computed by using LDA (Linear Discriminant Analysis) [7][13], for pattern recognition. The difference between PCA and LDA is the PCA uses maximum variation and LDA uses the difference between the means of each different class. LDA maximizes the ratio between class scatter and with-in class scatter. For this reason, the lighting condition will not affect the face recognition. The Fisher Face is done more on classification. But in reality, considering light as a feature will increase the complexity and also not a useful feature for face recognition. The implementation of LDA is not discussed in this paper since it is a mathematical concept. So in order to overcome this problem a concept called Local Binary Pattern Histogram (LBPH) [4][5][12] which is not discussed in more detailed. So in this project, we use Eigen Face algorithm for face recognition.

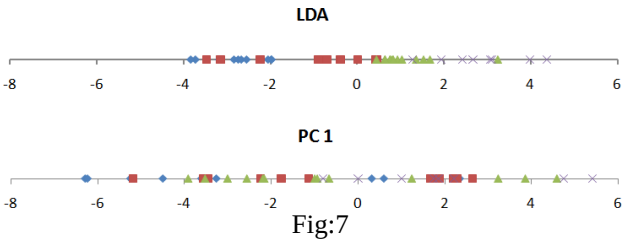


Fig:7

3.3. Working of LDA?

LDA is similar to PCA but it focuses on maximizing the separability among known categories. PCA [1][10] is used on entire data but LDA is for an individual in training set. For example,

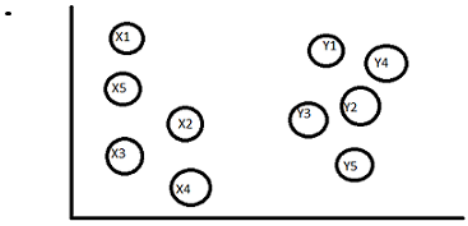


Fig: 8.

X1 through X5 belong to face 1 and Y1 through Y5 belong to face 2.
There are many techniques to represent or just categorize by projecting onto X and Y axes. But, LDA uses a new axis

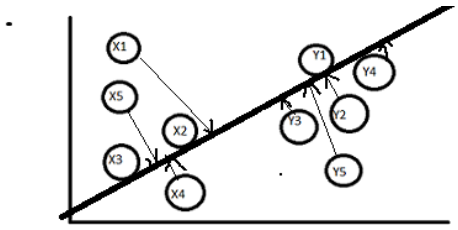


Fig. 9

Now all the data points are projected onto the new axis.

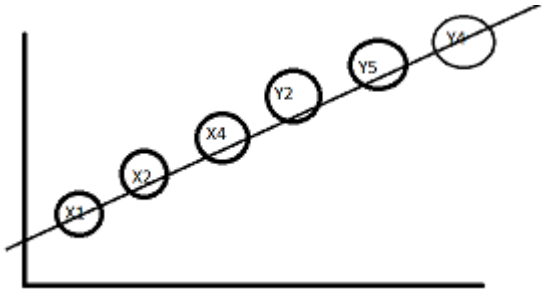


Fig. 10

It maximizes the distance between two means and minimizes the variation which is called as scatter and represented by s^2 within each category. This process is same for 2-dimensions. But, for 3-D data points there has to be a central value in each category and a decentralized distance from the central point.

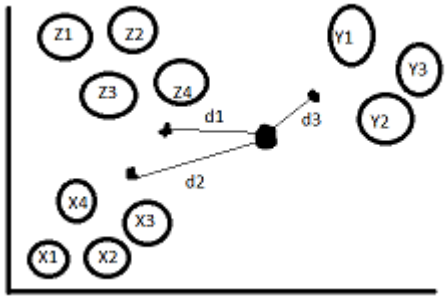


Fig. 11

The maximum distance between each category is the central point while the scatter of each category is $\frac{d1^2+d2^2+d3^2}{S1^2+S2^2+S3^2}$. In this way LDA is plotted and separation of data is done. LDA is faster and efficient than PCA.

4. Attendance monitoring

The following process explains how the face detection and recognition is done for implementing attendance monitoring system. In this process, the HAAR Cascades [8][9] algorithm is used for face detection and Eigen Face [2][3] algorithm is use for face recognition.

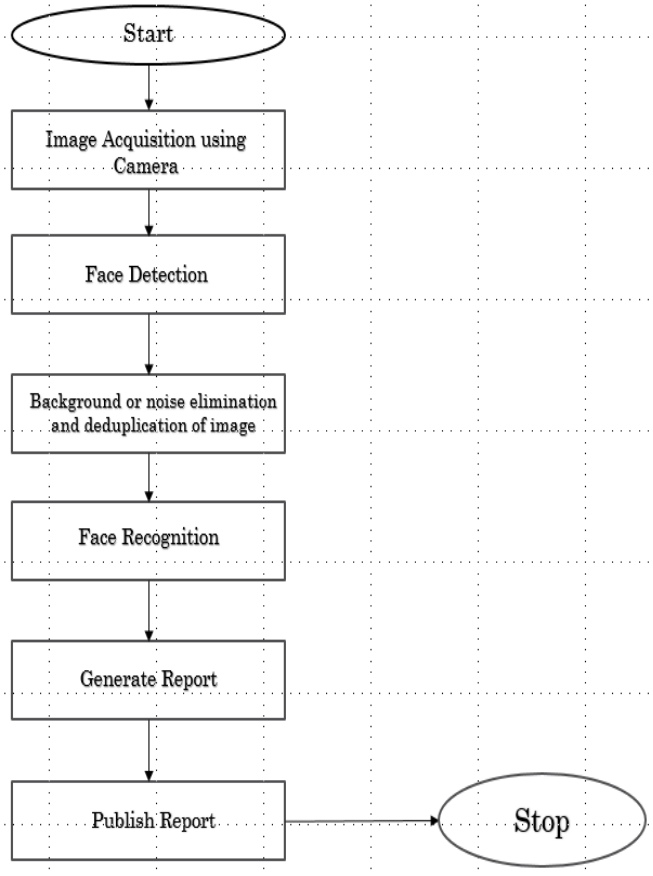


Fig. 12

Image Acquisition
Initially, the camera will take a photo of the entire class and stores in a file.

Face detection
Now after taking a photo it should detect that there are faces in the photo. HAAR Cascades algorithm is used for detection.

Eliminating errors or de-duplication of images
In this, all the blind spots present in the picture are and all unwanted features are also removed. And if any face is detected twice then that should also be removed.

Face recognition
Once after deleting all blind spots and other duplications, it should recognize the faces either by their names or the id of a person. Eigen Face algorithm is used for recognition.

Generate Report
Once after recognizing the face, a report is generated as the member is present in a file to that particular id of a person. This process is done for all faces.

Publish Report
After a final report is generated, that file should be sent to all the people who are authorized. Finally, the attendance will be done.

5. Hardware Configuration

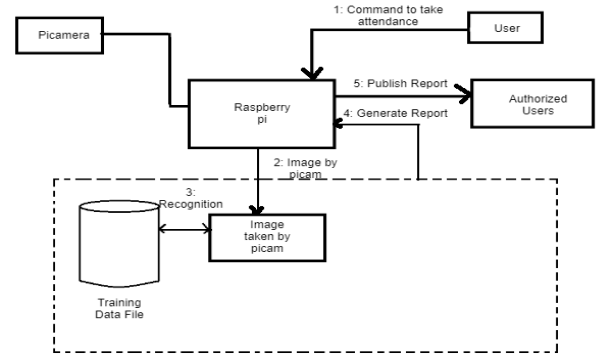
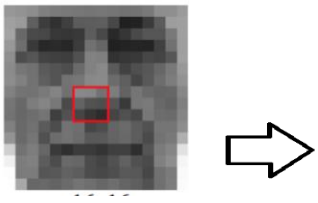


Fig. 13. Hardware configuration and setup

6. Future Scope

The Eigen Face [2][3][11] and Fisher Face [6] algorithms give results, but there some drawbacks in those algorithms. In the Eigen face under different illumination conditions the algorithm may not give better results. In order to overcome this drawback, the Fisher Face was introduced. But this takes light as a feature which is not required in reality. So to overcome all these problems an algorithm called Local Binary Pattern Histogram (LBPH) [4][5][12] is introduced. This algorithm gives better results and the computations are done at a faster rate. The LBPH takes the pixel values and a result is generated called as a histogram which is a value stored. When a new face is encountered then the histogram value is calculated for that photo. Now that histogram value is compared to the values stored in the file. If the match is present it is recognized. Else it is not recognized. So how does the LBPH calculates the histogram value? If we take a picture, the middle value is considered and all the surrounding values are subtracted from the center pixel value. If the subtracted value is less than or equal to the center value it is denoted as 1 and if it is a negative value or greater than the center value it is denoted as 0. And consider all values in clockwise direction. For example, consider



140	128	105
95	82	70
60	43	54

1	1	1
1		0
0	0	0

11100001

225

In this way, every image in the training set has a histogram value. So there will be a file where all the histograms are stored. So this will be the efficient algorithm for the face recognition.

And also a single camera may not give all members faces. So for that, we should arrange the cameras in such a way that all members in the room are covered. In different orientations, the cameras need to be fixed. By doing this the black spots can be eliminated very faster. And by placing cameras a face can be recognized many times but the code should be written in such a way that once the attendance is given to that id if it recognizes any number of time it is considered to be as only 1 time.

References

[1] Jonathon Shlens, “A Tutorial on Principle Component Analysis”, report, December 10, 2005: Version 2.

[2] Davies, Ellis, Shepherd (eds), “Perceiving and Remembering faces”, Academic Press, London, 1981.

[3] Matthew A. Turk, Alex P. Pentland, “Face Recognition Using Eigen faces”, report.

[4] Md. Abdur Rahim, Md. Najmul Hossain, Tanzillah Wahid, Md. Shafiul Azam, “Face Recognition using Local Binary Patterns(LBP)”, GLOBAL JOURNAL OF COMPUTER SCIENCE AND TECHNOLOGY GRAPHICS AND VISION, Volume 13 Issue 4 Version 1.0 Year 2013.

[5] M. Grudin, “On internal representation in face recognition systems”, Pattern Recognition, 33(7):1161 1177, 2000.

[6] Peter N. Belhumeur, Joao P. Hespanha, David J. Kriegman, “Eigenfaces vs. Fisherfaces: Recognition Using Class Specific Linear Projection”, IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE, VOL. 19, NO. 7, July 1997.

[7] Alaa Tharwat, Tarek Gaber, Abdelhameed Ibrahim, Aboul Ella Hassanien, “Linear Discriminant Analysis: A detailed tutorial”, report, AI Communications 00 (20xx) 1-22, DOI 10.3233/AIC-170729, IOS Press.

[8] Vandna Singh, Dr.Vinod Shokeen, Bhupendra Singh, “Face Detection by HAAR Cascade Classifier with simple and complex background images using openCV implementation”, International Journal of Advanced Technology in Engineering and Science, Volume No.01, Issue No. 12, December 2013.

[9] Lahiru Dinalankara, “Face Detection and Face Recognition Using Open Computer Vision Classifiers”, report, August 4, 2014.

[10] Chin-Shu Chang, Teh-Lu Liao, Po-Yun Hsu, Kuo-Kuang Chen, "Human face recognition system using modified PCA algorithm and ARM platform", Computer Communication Control and Automation (3CA) 2010 International Symposium on, vol. 2, pp. 294-297, 2010.

[11] Jun Zhang, Yong Yan, M. Lades, "Face recognition: eigenface elastic matching and neural nets", Proceedings of the IEEE, vol. 85, no. 9, pp. 1423-1435, 1997.

[12] W. Ge, W. Quan, C. Han, "Face description and identification using histogram sequence of local binary pattern", Proc. 7th Int. Conf. Adv. Comput. Intell. (ICACI), pp. 415-420, Mar. 2015.

[13] H. M. Moon, D. Choi, P. Kim, S. B. Pan, "LDA-based face recognition using multiple distance training face images with low user cooperation", Proc. IEEE Int. Conf. Consum. Electron. (ICCE), pp. 7-8, Jan. 2015.

[14] P. N. Belhumeur, J. P. Hespanha, D. J. Kriegman, "Eigenfaces vs. Fisherfaces: Recognition using class specific linear projection", IEEE Trans. Pattern Anal. Mach. Intell., vol. 19, no. 7, pp. 711-720, Jul. 1997.