

LBPH-based Enhanced Real-Time Face Recognition

Farah Deebea¹, Aftab Ahmed⁴

School of Information and Software Engineering
University of Electronic Science and Technology of China
Chengdu, Sichuan, China

Fayaz Ali Dharejo³

Computer Network Information Center
Chinese Academy of Sciences, University of Chinese
Academy of Sciences, Beijing, China

Hira Memon²

Department of Computer System Engineering
Quaid e Awam University of Engineering Science and
Technology
Nawabshah, Pakistan

Abddul Ghaffar⁵

Key Laboratory of Instrumentation Science & Dynamic
Measurement, Ministry of Education
North University of China
Taiyuan, China

Abstract—Facial recognition has always gone through a consistent research area due to its non-modelling nature and its diverse applications. As a result, day-to-day activities are increasingly being carried out electronically rather than in pencil and paper. Today, computer vision is a comprehensive field that deals with a high level of programming by feeding the input images/videos to automatically perform tasks such as detection, recognition and classification. Even with deep learning techniques, they are better than the normal human visual system. In this article, we developed a facial recognition system based on the Local Binary Pattern Histogram (LBPH) method to treat the real-time recognition of the human face in the low and high-level images. We aspire to maximize the variation that is relevant to facial expression and open edges so to sort of encode edges in a very cheap way. These highly successful features are called the Local Binary Pattern Histogram (LBPH).

Keywords—Face recognition; feature extraction; Local Binary Pattern Histogram (LBPH)

I. INTRODUCTION

Among other biometric methods, face recognition is also one of the ways to identify any individual subject. Face recognition identifies anyone by comparing the physical characteristics of the item. There are two face recognition modes, still images and live video. The first step in face recognition is face detection. Therefore, to perform facial recognition, the system must position the face earlier in the input image or video stream. This step is called face acquisition or detection. In this section, the main face recognition methods are described. Such as neural networks, geometric feature matching, feature-based methods and graph matching [1]. The following techniques were studied in the context of facial illustrations. The geometric feature matching method depends on the calculation of a set of geometric features of the facial image. The general specification is defined by a vector showing the position and size of the main facial features, for example, the eyebrows, nose, mouth and facial contours. T. Kanade [2] completed an innovative study on automatic face recognition by implementing conventional features. His system achieved higher results with 75% accuracy on a data set of 20 subjects, and each subject took two photos, one for the model and one for the test. Later, I.J.

Cox et al. [3] proposed a hybrid distance method for automatic face recognition. He scored 95% peak recognition accuracy on a dataset of 685 subjects. In the hybrid distance method, each face image shows 30 manual induced distances. B.S. Manjunath et al. [4] proposed the fragmentation process for the detection of feature units of every single facial image, which decreased the storing capacity of the database, and it generated 35-45 feature units per facial image. Face detection based on geometrical approaches based on the geometrical proportion between regular features by employing some statistical models. These models calculate the distances between features; they could be more beneficial for identifying expected matches in a large dataset. Geometric Feature-based algorithms have some advantages over other methods like rotation independently, faster with execution time, scaling [5].

Like geometrical feature matching and feature-based face recognition approach, graph matching face recognition approach is one of them [6]. Introduced dynamic link architecture to falsification invariable object identification, in this approach researcher utilize elastic graph matching approach for calculating the nearest saved graph. This approach called Dynamic link structure is an addition to standard ANN. Sparse graphs represented memorize objects. The vertices of sparse graphs are tagged through a multi-resolution statement in the context of a local power spectrum, and their borders are tagged with geometric distance vectors.

Recognizing the object from multimedia (Video, image) is called object recognition, can be identifying by employing any efficient method like elastic graph matching. Elastic graph matching process by matching the cost function randomly modified at every node. The better testing outcomes were obtained on the dataset of 87 subjects and a group of office objects containing various expressions with an alternation of 15 degrees. The matching procedure takes more computation time; it takes 25 secs to match with 87 saved objects on the symmetric device by 23 carriers. After that L. Wiskott [6] modified this approach and compared individual front view faces of 112 images. Probe pictures were deformed because of the rotation in depth and variation in facial appearance. Functional outcomes were achieved of facial images on big

rotated angles. Resulting obtained 86.5% recognition percentage on testing of 111 face images at 15 degrees' rotation, and 66.4% recognition percentage on testing of 110 face images at 30 degrees' rotation of a 112 neutral frontal views [7]. Generally, dynamic link structure is dominant on various facial recognition approaches in the context of rotation stability; though, the matching procedure is extensive in terms of computation.

The neural network is so much simplified face recognition approach because of its non-linear architecture in the net system. Therefore, the features extraction phase is more effective than the linear technique, it selects a dimensionality reducing linear projection that increases the scatter of all expected models [8]. ORL database contains 40 objects, with 400 images of each object, recognition accuracy was 96.2% obtained on this dataset. It takes 4 hours of training time and less than 0.5 seconds for classification and provides limited invariance to transformation, variation, scale, and distortion. Though, the number of the individual is proportional to computing time. Increase in the number of people, the computation time also increases. Generally, neural networks approach gets difficulties when the number of people increases. Furthermore, a neural networks approach is not appropriate for only a single model image recognition experiment, since various model pictures per individual subject are required to train the system at "optimum" parameters setting.

SIFT is the most well-known and widely adopted technique for feature extraction. This technique jointly uses the difference of Gaussians (DOG) and image pyramid concepts. By employing this technique, image is processed at different scales by the Gaussian filter. This technique gives excellent achievement if there are any illumination or viewpoint variations presents, it is also invariant to the rotation as well as a scaling factor [10]. Under the SIFT feature matching technique, each feature of the test images is compared to the dataset images. Euclidean distance gives best-extracted feature vector.

SIFT algorithm has four primary phases for the feature matching process named as Scale-Space Extreme Detection, KeyPoint Localization, Orientation Assignment and Key Point Descriptor" [10]. SIFT algorithm is straightforward and gives better results, but one drawback is of its computational complexity and time.

SURF is a features indicator in an input image. SURF detects the local features of the face image. The SURF features detector is more efficient and robust than the SIFT features detector. Comparing to the SIFT, SURF provides excellent results. SURF indicators determine the interest points in the face image. For the detection of interest points, SURF uses decimal number estimation of the determinant of Hessian blob indicator that could be calculated by three decimal numbers actions by utilizing a pre-calculated integral picture [9].

CNN is a very excellent and mathematical tool used for complex computation for multimedia object images videos, for many tasks like recognition, segmentation, etc. CNN is also handy for the study of 2D variability shapes. In CNN

Local features with some share, weights are combined [11] which is also used for sub-sampling purposes such as for shifting the level, scale-invariance, and deformation.

Gabor wavelet frequency and alignment demonstrations based on human graphical method, it is more suitable for representing texture learning. Extracting the feature from particular positions are also supported by Gabor wavelet, also perform image analysis at different scale and orientations. Frequency and rotations variations are also handled in these techniques [12]. By using the Gaussian envelope, the Gabor wavelet is an accentuated [13].

The principal component analysis is a traditional algorithm, broadly employed in machine vision and pattern recognition technology as well as used in feature extraction purpose [14]. In this algorithm, it is stated that "any face image can be reconstructed nearly as a weighted sum of a small set of images which define a facial base (Eigen images), and an average image of the face." In 1991, Eigenfaces technique was suggested by Turk and Pentland [14]. Eigenfaces were also proposed for facial images recognition. Meanwhile, PCA became best effective algorithm for facial recognition. The PCA was frequently used for features extraction and dimension reduction.

So far, the literature is well defined, the rest of the work contains an overview of the facial recognition system, Section III deals with the methodology and Section IV shows the result and the discussion, and finally conclusion and the future meaning is given.

II. OVERVIEW OF THE FACE RECOGNITION SYSTEM

Face detection and identification process is a machine learning technique, by learning and extracting the physical characteristics of the human. Matching these features with the tested images can identify the person or deny those people to recognize. There are several challenging and varying parameters in face detection and identification like illumination, different poses, change expressions, low-quality input images, etc.

There are several different perspectives about face detection and recognition system; some of the projects only focus on images with high resolution; some of them focus on low resolutions. Recently researchers focus on the different frontal view of images, from different angles, different lighting illuminations, etc.

Traditionally, Face recognition system follows four primary phases, listed follows; also the basic face recognition diagram is shown in Fig. 1.

- a) Face Detection
- b) Preprocessing
- c) Feature Extraction
- d) Feature Matching

A. Dataset Preparation

We create our own dataset; the dataset contains a total 1000 images, 333 face images of each person with 60×60 resolution of each image. It is created based on face detection.

Make different facial expressions and postures to a scene and detect faces. The saved pictures are stored in the same folder to form the generated face dataset. At this stage, the dataset is preprocessed for the feature extraction process. The dataset images have been converted into grayscale images for features extraction, and then normalized those images for good recognition results. For features detection, Haar modules have been used to detect these local features in a given an input image. Here, the input image refers to the digital image captured by the camera. After detecting features, the classifier will classify the input image as a face image as shown in Fig. 2.

In this project, face detection algorithms are developed based on Local Binary Patterns Histogram (LBPH). The LBPH-based algorithm, the first step is to extract the image pattern with the LBPH algorithm. Then, two thresholds are set to calculate the probability of face in the image pattern. After that, the sliding window applied to identify the faces in given images and recognize those faces. From Fig. 3 we can understand well.

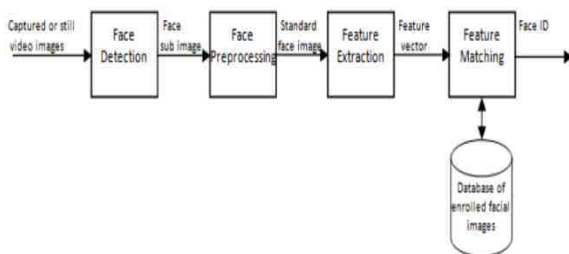


Fig. 1. Basic Phases in Face Recognition.

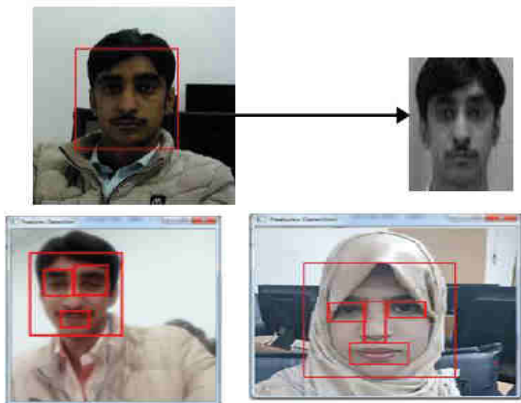


Fig. 2. Image Preprocessing and Feature Selection.

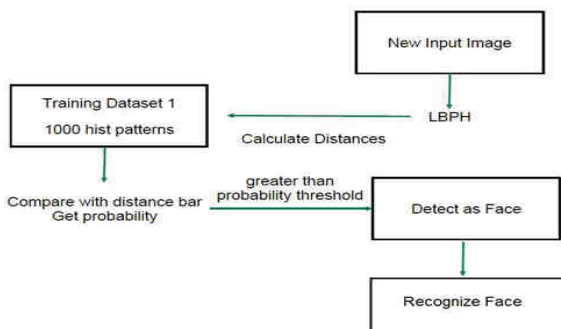


Fig. 3. Face Detection and Recognition Methodology.

III. METHODOLOGY

A. Local Binary Patterns Histogram

The goal of face detection is to detect [15] and locate faces in the image, to extract human face to use in other areas. Nowadays, there are many different algorithms to accomplish face detection or recognition, such as Fisher faces, Eigenfaces, Scale-invariant Feature Transform (SIFT) and Speed-Up Robust Features (SURF). In this section, LBPH-based face detection algorithm is introduced. LBPH algorithm is the combination of Local Binary Patterns (LBP) and Histograms of Oriented Gradients (HOG) descriptor. LBP is an easy but powerful way to extract and label the pixels of an image. Using the LBPH, we can easily represent face images with just a straightforward vector.

Ojala et al. first introduced local Binary Patterns (LBP), and it is designed to be a texture analysis for the gray-scale image [16] [17]. To detect faces in an RGB (colored) photo, we have to convert the image into a grayscale image at first. For each pixel $P_{i,j}$, it is a vector that contains three values, which is to represent the degree of red, blue and green. We convert the RGB image into gray-scale image by:

$$G_{i,j} = (0.2989, 0.5870, 0.1140)^T \cdot P_{i,j} \quad (1)$$

Where $G_{i,j}$ represented the corresponding pixel in the gray-scale image. The LBP operator is going to compare the center value with its P neighbor values on the circle with radius R and assigns neighbor value as 1 if center value is bigger than the neighbor, assigns 0 on the contrary. In this case, we set $P = 8$ and $R = 1$ which means that we consider a 3×3 check. The LBP operator labels the center pixel by thresholding the 8 neighbors. For each center pixel, the LBP operator outputs an 8 bits' binary number and we convert it into a decimal between 0 and 255 as a result.

We utilize the following notation to describe the LBP operator:

$$LBP_{P,R}(x,y) = \sum_{i=1}^8 \sin(G_i - G_c)2^i \quad (2)$$

Where $LBP_{P,R}(x,y)$ is the results of the LBP operator, (x,y) is the coordinate of center the pixel in the 3×3 check. G_c is the center value in gray-scale image, G_i is the neighbor value in gray-scale image.

The LBP operator reduces the influence of illumination and returns the texture by considering every pixel in an image that excludes the boundary pixels. The figure shows the demo of the LBP operator for one pixel.

B. Extracting Histograms with LBP

Using the LBP result, we can generate a histogram for this image and formed a data vector to describe the patterns of the original image. The Histogram is about the frequency of the occurrences of LBP result for each pixel. From the last part, after doing the LBP operator, the value of each pixel is between 0 and 255. Thus, the histogram contains only 256 positions.

However, we are not considering the whole image directly. We divide the image into several image pieces. We calculate

the values of 256 positions for each image piece given Nx and Ny, representing the number of cells in the horizontal direction and vertical direction respectively.

Then, we concatenate all histograms of all image pieces to form a more significant histogram. For instance, if we set Nx = 4 and Ny = 4, we will have $4 \times 4 \times 256 = 4.096$ positions in the total histogram. Finally, we can use this total histogram to represent the image by just one data vector.

C. LBPH Algorithm

The Local Binary Patterns Histograms (LBPH), was introduced in the year 2006 [18]. LBPH algorithm was commonly used for facial recognition. This algorithm is based on the local-binary-operator [19], broadly implemented in face recognition, due to its discriminating strength and calculation easiness [20].

Face recognition is performed by employing the Local Binary Pattern Algorithm. The LP operator is applied for local binary features by considering the Local Binary patterns [21] which helps to shorten the local special features of the face image. The LBP is the binary ratio of pixels intensities within the center pixel. And it's around eight pixels. The mathematical description is described in the below equation.

$$LBP(p_c - q_c) = \sum_{m=0}^7 S(t_m - t_c)^{2^t} \quad (3)$$

Center pixel is shown by t_c and (p_c, q_c) represents the surrounded eight pixels, it is very though very useful to determine the face feature. In face matrix feature extracted from the image to compare the values with center pixel values to finally generate binary code.

D. Feature Vectors

Images divided into the region to show the faces efficiently then these images have been subdivided into A^2 regions, i-e $8^2 = 64$ regions. Histogram of each image is composed by each potential label, where each bin in histogram tells the information about the pattern to get the feature vectors from histograms. The each regional histogram V $(V-1) + 3$ bins: $V(V_1)$.

To perform the specific area with the help of the LBP operator from the edges of the image if not exist this means some section of the border is not related. For the image $(C \times D)$, the feature vector is designed with the help of calculating the LBP code for all pixels (P_c, Q_c) with $x \in \{U + 1, \dots, C - U\}$ and $q_c \in \{U + 1, \dots, D - U\}$. If an image is divided into $a \times a$ regions, then the histogram for region (a_p, a_q) , with $a_p \in \{1, \dots, a\}$ and $a_q \in \{1, \dots, a\}$, Mathematically,

$$J_k(A_p - A_q) = \sum p, q \{LBP_{v,u(p,q)=M(K),k=1,\dots,V(V-1)+3}\} \quad (4)$$

$$P \in \left\{U + 1, \dots, \dots, \frac{C}{A}\right\} A_p = 1$$

$$\{(A_p - 1)(\frac{C}{A})\} 1, \dots, \dots, C - U \quad A_p = A$$

$$\{(A_p - 1)(\frac{C}{A})\} 1, \dots, \dots, A_p(\frac{C}{A}) \text{ else}$$

$$\in \left\{U + 1, \dots, \dots, \frac{D}{A}\right\} A_p = 1$$

$$\{(A_q - 1)(\frac{D}{A})\} 1, \dots, \dots, D - U \quad A_q = A$$

$$\{(A_q - 1)(\frac{D}{A})\} 1, \dots, \dots, A_q(\frac{D}{A}) \text{ else}$$

In which M is the label of binary k and $B(Z) = \{1, Z \text{ is True } 0, Z \text{ is False}\}$

From the feature vector, we can get three districts levels of the locality of the face; these labels combine information of the little background level and architecture histogram which provide the knowledge about the face.

E. Comparing the Feature Vectors

To measure the feature of images, the sample (H) and a model (I) are used as so that the difference values between feature vectors can be measured. Here with the help of histograms can measure the difference between two images. - Histogram Intersection

$$F(H, I) = \sum_{v=1}^{j^2} \left(\sum_{e=1}^{Q(Q-1)+3} \min(H_{e,v}, I_{e,v}) \right) \quad (5)$$

Log-likelihood statistics

$$M(H, I) = \sum_{v=1}^{j^2} \left(- \sum_{e=1}^{Q(Q-1)+3} H_{e,v} \log I_{e,v} \right) \quad (6)$$

Chi-square statistics

$$Y^2(H, I) = \sum_{v=1}^{j^2} \left(\sum_{e=1}^{Q(Q-1)+3} \frac{(H_{e,v} - I_{e,v})^2}{H_{e,v} + I_{e,v}} \right) \quad (7)$$

The χ^2 with eight of face rank the similarity of images by the computing the histograms. The deeper the value of the χ^2 , the greater is the similarity.

IV. RESULTS AND DISCUSSIONS

The proposed primary face recognition application has been implemented in Python, Open CV image processing library and LBPH algorithm over HD camera. In this application, the algorithm applied for face recognition is distributed in three dissimilar and independent sections.

1) Face image acquisition Module, 2) Dataset training module, 3) Face recognition module. In the image acquisition module, the user needs to run "Dataset Creator. Py" file from Python IDLE shell and enter the subject ID (see Fig. 4). This module will open the "Image Acquisition" window, which will detect and capture the face images.

Application opens the externally connected camera with PC, apply the Haar classifier to detect the face and capture face images (see Fig. 5). The algorithm permits to store face images in a folder with the subject ID and sample number.

After the preprocessing, all the 2000 captured images of 4 subjects will be stored in the same folder. Each image will be assigned subject ID and sample number. Sample number is the number of images per face image. So the sample number will be different, while the subject ID will be the same for a single face image (see Fig. 6).

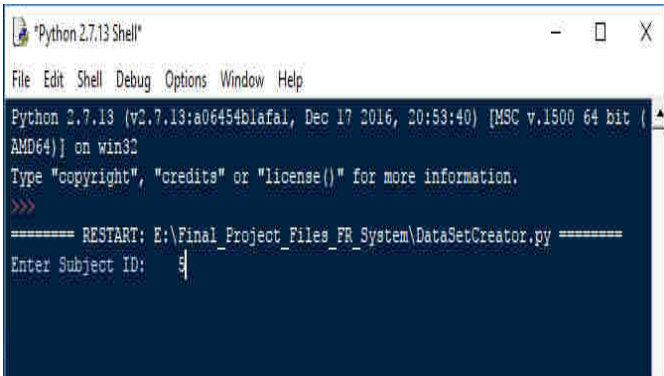


Fig. 4. Image Acquisition.

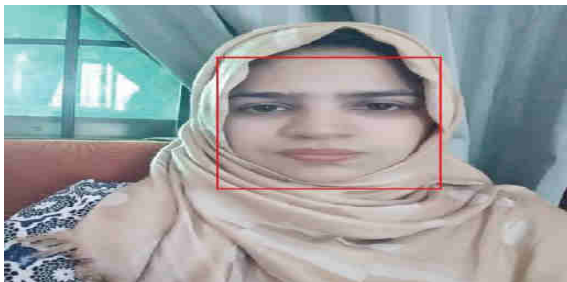


Fig. 5. Face Detection.

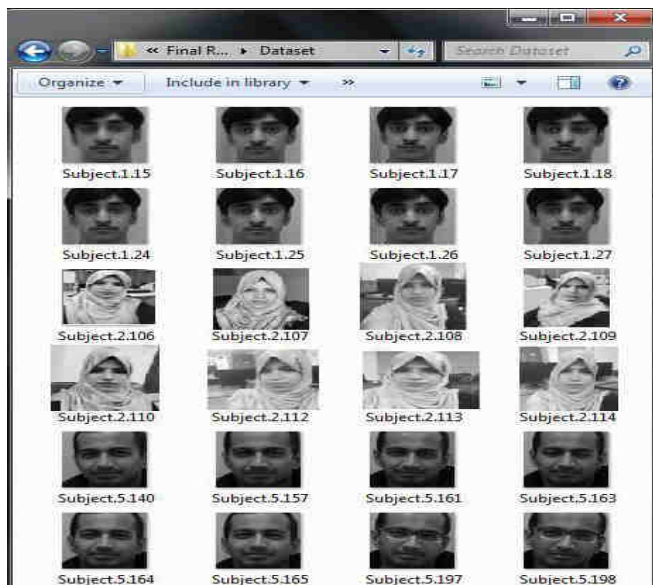


Fig. 6. Subject ID with Sample Number.

A. Experiments

This part states that the experiments executed to match the performance of the facial recognition algorithm applied in this system. It is required to describe the experimental conditions that must be acceptable for the proposed system. So, the proposed experiments must be executed by changing certain factors that are significant in the learning and recognition method of the system. The following test was implemented to examine the efficiency of the operating system—identification percentage based on a threshold. To perform the experiments, besides my dataset images, I also have taken dataset images of

my three lab mates from the school of Information & Software Engineering. The tests were executed, and the efficiency ratios were examined in every situation. Fig. 7 displays the subjects with their corresponding IDs. After the execution of the training of the experiment subjects, facial recognition is executed. The expected results generated by the system are defined below.

B. True Positive

The real positive condition happens when the observed individual's data stored in the dataset folder and the recognized subject matches to the one that is available in the training dataset. Fig. 8 displays the result after applying facial recognition; in this situation, the result is accurate.

C. True Negative

The true adverse condition occurs when the tested subject's data is not stored in the database, and the system could not recognize that subject. Fig. 9 displays the result after applying face recognition; in this condition, the recognition result will be "Unknown," and hence the effect will be considered as correct.

D. True Occlusion

True occlusion condition happens when the subject's data saved in the dataset folder without occlusion condition and the recognized subject matches to the one that is available in the training dataset. Fig. 10 displays the result after applying facial recognition; in this situation, the result is accurate.



Fig. 7. Test Subjects.

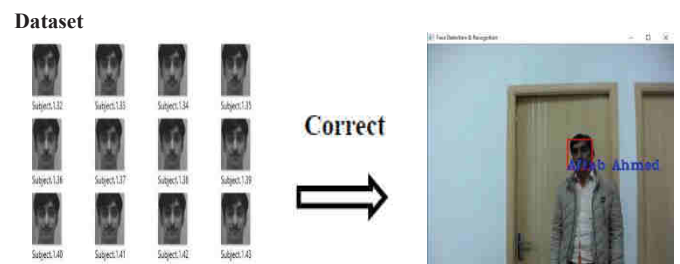


Fig. 8. True Positive.



Fig. 9. True Negative.



Fig. 10. True Occlusions.

E. True Pose Variation

True pose variation occurs when the subject's position is at a different angle concerning the camera, and the system can recognize that subject even if in the dataset, only frontal view face images are stored as shown in Fig. 11.

We evaluate the parameters and different values of our model during the testing experiment which is given in the Table I.

The accuracy of model is shown in Fig. 12 the accuracy is taken against the Training and Validation dataset. We divided our dataset, such as 50% training dataset, 30% testing and rest 20% validation dataset.

F. Setup

To implement the above jobs subsequent Hardware and Software are required to grow the proposed the scheme. We used software tools: Windows/UNIX operating system, Python 2.7/3.6, OpenCV library, Numpy library, Matplotlib library, Pillow library.

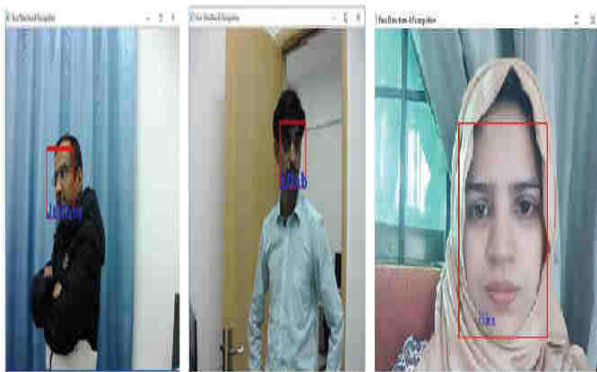


Fig. 11. True Occlusions.

TABLE I. PARAMETERS TO TEST THE EXPERIMENT

Parameter	Value
Training	
Number of Subjects	4
Cam-Subject Distance(cm)	200
Recognition	
Threshold	<250;<500;<750;<100
times	80[5 each subject]

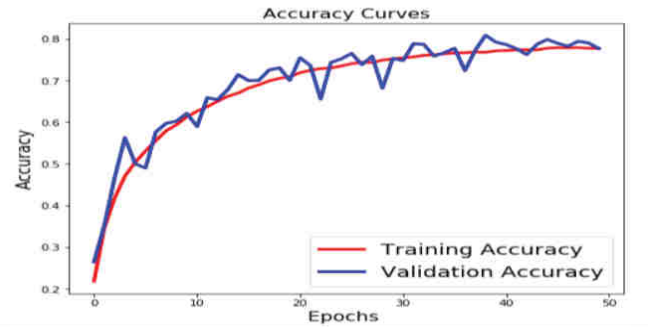


Fig. 12. Training v/s Testing Accuracy Curve.

V. CONCLUSION AND FUTURE WORK

We purposed LBPH for image recognition and face detection in the surveillance camera in a specific area. Having obtained good results from various experimental analyzes of this technique, they also provide valid results for occlusion, pose variation, and illumination. Therefore, the proposed system allows recognition and recognition of faces in a controlled environment. As machine learning is very important nowadays, there are many areas where this work can be expanded. In implementing this project, we have identified some areas for improvement, such as Limitations of distance, the maturity of algorithms and camera qualities, even using DNN techniques. Accuracy can be improved in the future, which is more directly related to our work.

ACKNOWLEDGMENT

We are highly thankful to Dr. SullemanMemon and Dr. Javed Akhtar Unar for their guidance and technical support throughout the project

REFERENCES

- [1] Elham Bagherian, Rahmita Wirza O.K. Rahmat, "Facial feature extraction for face recognition: a review", IEEE 2008 International Symposium on Information Technology.
- [2] T. Kanade, "Picture processing by computer complex and recognition of human faces," technical report, Dept. Information Science, Kyoto Univ., 1973
- [3] I.J. Cox, J. Ghosn, and P.N. Yianios, "Feature Based face recognition using mixture distance," Computer Vision and Pattern Recognition, 1996
- [4] B.S. Manjunath, R. Chellappa, and C. von der Malsburg, "A Feature based approach to face recognition," Proc. IEEE CS Conf. Computer Vision and Pattern Recognition, pp. 373-378,1992
- [5] P. Suja and S. Tripathi, "Analysis of emotion recognition from facial expressions using spatial and transform domain methods," International Journal of Advanced Intelligence Paradigms, vol. 7, pp. 57-73, 2015.
- [6] M. Lades, J.C. Vorbruggen, J. Buhmann, J.Lange, C. Von Der Malsburg, R.P. Wurtz, and M. Konen, "Distortion Invariant object recognition in the dynamic link architecture," IEEE Trans. Computers, vol. 42, pp. 300-311,1993.
- [7] Dharejo FA, Jatoti MA, Hao Z, Tunio MA. PCA based improved face recognition system. Frontiers in Artificial Intelligence and Applications. 2017. <https://doi.org/10.3233/978-1-61499-785-6-429>
- [8] KIRBY, M. AND SIROVICH, L.. Application of the Karhunen-Loeve procedure for the characterization of human faces. IEEE Trans. Patt. Anal. Mach. Intell. 12,1990.
- [9] E. Paul and A S Ajeena Beegom, "Mining images for image annotation using SURF detection technique," IEEE International Conference on Control Communication & Computing India, Trivandrum, 2015, pp.724-728.

- [10] V. Purandare and K. T. Talele, "Efficient heterogeneous face recognition using Scale Invariant Feature Transform," IEEE International Conference on Circuits, Systems, Communication and Information Technology Applications, Mumbai, 2014, pp. 305-310
- [11] C. Yan, C. Lang, T. Wang, X. Du and C. Zhang, 2014, "Age estimation based on convolutional Master Thesis of University of Electronic Science and Technology of China 42 neural network" Pacific Rim Conference on Multimedia, Springer, Malaysia, 2014, pp. 211-220
- [12] J. K. Kamarainen, "Gabor features in image analysis," IEEE International Conference on Image Processing Theory, Tools and Applications, Istanbul, 2012, pp. 13-14.
- [13] S. Murala, A. B. Gonde and R. P. Maheshwari, "Color and Texture Features for Image Indexing and Retrieval," IEEE International Advance Computing Conference, Patiala, 2009, pp. 1411-1416
- [14] M. Kirby and L. Sirovich, "Application of the KL Procedure for the Characterization of Human Faces," IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 12, no. 1, pp. 103-108, Jan. 1990
- [15] Dr C.Sunil Kumar¹, C.N Ravi² and J.Dinesh³. Human Face Recognition and Detection System with Genetic and Ant Colony Optimization Algorithm IOSR Journal of Computer Engineering (IOSR-JCE) e-ISSN: 2278-0661, p-ISSN: 2278-8727, Volume 16, Issue 4, Ver. VII, Jul – Aug. 2014
- [16] A. Ahmed, J. Guo, F. Ali, F. Deebe and A. Ahmed, "LBPH based improved face recognition at low resolution," 2018 International Conference on Artificial Intelligence and Big Data (ICAIBD), Chengdu, 2018, pp. 144-147. doi: 10.1109/ICAIBD.2018.8396183
- [17] C. Darwin. The Expression of the Emotions in Man and Animals. London: John Murray, 1872.
- [18] H. A. Ahonen, T. and M. Pietikinen, "Face description with local binary patterns," IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 28, no. 12, pp. 2037–2041, 2006.
- [19] P. M. Ojala, T. and D. Harwood, "A comparative study of texture measures with classification based on feature distributions," Pattern Recognition, vol. 19, no. 3, pp. 51–59, 1996
- [20] M. Pietikinen, "Local Binary Patterns," vol. 5, no. 3, p. 9775, 2010
- [21] T. Chen, Y. Wotao, S. Z. Xiang, D. Comaniciu, and T. S. Huang, "Total variation models for variable lighting face recognition" IEEE Transactions on Pattern Analysis and Machine Intelligence, 28(9):1519{1524, 2006