

# Automated Face Detection, Recognition and Gender Estimation Applied to Person Identification

<sup>1</sup>Md Hannan, <sup>2</sup>Md Rafiqul Islam, <sup>3</sup>Md Ashraful Haque,  
<sup>4</sup>Md Safaet Hossain, <sup>5</sup>Anwaar Ulhaq and <sup>6</sup>Jeyson Jaman Sawan

<sup>1,2,4,6</sup>Department of Computer Science and Engineering, City University, Bangladesh

<sup>3</sup>Department of Electrical and Electronics Engineering, Daffodil Int. University, Bangladesh

<sup>5</sup>School of Computing and Mathematics, Charles Sturt University, Australia

## Article history

Received: 08-02-2019

Revised: 22-02-2019

Accepted: 19-03-2019

Corresponding author:

Md Rafiqul Islam

Department of Computer  
Science and Engineering,

City University, Bangladesh

Email: rafiqulislam.cse24@gmail.com

**Abstract:** Face detection, recognition and gender estimation are one of the most significant research areas in computer vision, not only because of the challenging nature of faces as an object but also due to the countless applications that require the application of face detection, tracking and recognition. Although many significant types of research on face detection, recognition and gender estimation problems have done in the last few years separately, there is no particular research on face detection, recognition and gender estimation together from a real-time video for person identification. So, we feel that these types of significant research are still needed to work. The main contributions of our paper are divided into three parts, namely face detection, recognition and gender estimation for person identification. In our research work, we use Local Binary pattern Histogram (LBPH) method and Convolution Neural Network (CNN) to extract the facial features of face images whose computational complexity is very low. By calculating the Local Binary Patterns Histogram (LBPH) neighborhood pixels and Convolution levels, we extract effective facial feature to realize face recognition and gender estimation. We show the experimental results using these methods to recognize face and gender for person identification. CNN increase the calculating speed of testing real-time video and also improve the recognition rate. By using LPBH, we get 63% accuracy on average where CNN gives 99.88% training accuracy for face recognition-1 and 96.88% accurate for gender estimation-1 and 100% training accuracy for face recognition-2 and 93.38% training accuracy for gender estimation-2. However, Convolution Neural Networks (CNN) learns fast and predict efficiently.

**Keywords:** Face Detection, Face Recognition, Local Binary Pattern Histogram (LBPH), Convolution Neural Network (CNN), Gender Estimation

## Introduction

Face detection, recognition and gender estimation have significant values in multiple sectors in real life application and study as well. In real life, Face detection, recognition and gender estimation are used in multiple sectors like security, conference, for surveillance, to identify a particular human, to recognize the persons face etc. But they have great differences between the image and video. Compared with the image, they have broad application in the video. Based on the real-time video, in the application of automatic person identifications, face detection, recognition and gender estimation have a complex problem because an image of a person's face

can contain noise, light issues, the angle of the face, wrinkles, low resolution and the persons face details can be large. Even though the condition of an image can affect the results. For face detection, recognition and gender estimation, the whole system needs to be improved in many sections to give a better result. The system must extract the features of the image accurately will help the application to improve the rate of the accuracy of detection, tracking and recognition to identify a person.

In the past decade, a number of research works have been done for automatic face detection, recognition (Zhao *et al.*, 2003; Hjelmas and Low, 2001; Yang *et al.*, 2002) and gender estimation separately. But separate detections, recognition and gender estimation does not

provide sufficient information when no appearance information is available. Nobody has been completed any of the research work for face detection, recognition and gender estimation jointly. It can provide better results by sharing information when combining it into a single framework. In addition, a number of methods have been applied for face detection, recognition and gender estimation separately. Most of the researchers follow and use the boosted cascade framework because it has advanced features. The features of the boosted cascade framework help to compute and construct the classifier which performs more accurately. However, this rate of accuracy is required a number of cascade stages that will help to reduce the similar achievement in the detection and recognition accuracy. Hence, we can reduce few cascade stages to not to achieve similar accuracy in face recognition. We can apply more advanced feature and a more advanced algorithm that will detection, recognize and estimate the gender the person more accurately to identify the person.

To solve texture classification problem (Ojala *et al.*, 1996), Local Binary Patterns (LBP) was actually developed (He *et al.*, 2011), LBP was introduced in the field of facial recognition for the first time by Ahonen *et al.* (2006). The algorithm has been used to detect the face and recognize the face. LBP is combined with the histogram strategy. Arashloo (2015) had suggested multi-scale LBP that will describe the face variations and the classification is done based on Linear Discriminant Analysis (LDA). To recognize the face, this method uses a static image and thus the accuracy rate of recognition is higher; however, the main drawback is that it isn't used in a video. Kalal *et al.* (2010) have proposed a Tracking Learning Detection (TLD), the method can track an object for a long time and they have combined this method for faces to recognize, this module track the faces based on TLD. But the complexity of computation is very high for this model, the model takes more time to track the face because of computational power. Hence the LBP is a well-defined method for face recognition and to the image analysis. In their works, the Local Binary Pattern Histograms (LBPH) are used to extract the facial features from facial areas and describe the whole face based on that extracted features. However, LBPH uses

more time to compute all the bins. All the bins are necessary to recognize the face of the person.

In this paper, our main objectives are to design and build a complete system to detect, recognize and estimate gender for person identification. This complete system is divided into two parts; first to create the face database where each individual faces is captured and stored. Then, to test the video (Chrysos *et al.*, 2015; Vasisht and Chen, 2017; Alqahtani *et al.*, 2018) scene where people can be reliably detected, recognized, estimated and identified. We propose a new method to describe the facial feature by LBPH and CNN. We calculate the neighborhood pixels to express face through the regression of local binary features (Chung *et al.*, 2017). We compute the LBP operator of these neighborhood pixels. According to these operators, we use a histogram to express the whole face. By comparing the LBPH between the test video and train database, we obtain the results of face detection, recognition and gender estimation for person identification.

In the light of the brief discussion of face detection, recognition and gender estimation for person identification, the following five connected components addressed in this paper are:

(a) Face detection and tracking which give probabilities of peoples present and searches faces in the video scenes, (b) a model to capture the extracted information of each individual detected face which will store in the model, (c) a database that is used in recognition task and the (d) model uses an algorithm to recognize the face to identify individual person. The final mode is then used to (e) track the face of the people and recognize their identities. Figure 1 shows the overview of the whole LBPH recognition system.

Even though some existing work have discussed face detection, tracking and recognition separately, none of these existing works have considered face detection, tracking and recognition combined. Moreover, even though some authors applied different approaches for different purposes, none of them have applied Local Binary Patterns Histograms for face detection, tracking and recognition together to identify people.

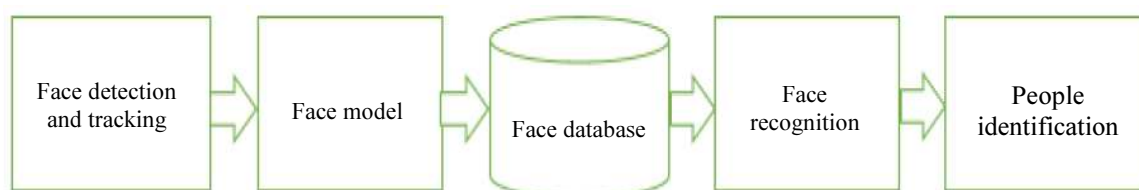


Fig. 1: The complete system overview of our paper

After all, from the above-mentioned references, the following shortcomings are observed:

- There are few individual studies that have applied Haar Classifier, LBP classifier, SVM separately. In addition, there are no well-known studies that have combined all these techniques together at the same dataset to investigate the variations in technique-based findings
- There is no significant study that has applied the above-mentioned LBPH techniques for face recognition with personal identification

To address the above-listed shortcomings, in our present work, we address all above discrepancies by introducing a framework to detect face and recognize the person with identification. So, to the best of our knowledge, our work is the first attempt to apply LBPH for face detection, tracking and recognition to identify a person. With the present work, we specify the different features of face for detection, tracking and recognition. The major contributions of this paper are listed as follows:

- We apply Haar-Cascade and LBP classifiers to extract the features for face detection and recognition and CNN class of deep learning to extract the key features of each face for face recognition and estimation of gender
- To detect and recognize the faces, we identify through our results that CNN classifier performs better
- We apply Local Binary Patterns Histograms (LBPH) algorithm and convolution neural networks algorithm to recognize the peoples for identification in real time
- We apply CNN for gender estimation for the detected faces of peoples in real time
- Our work also shows the importance of face recognition for person identification

Franc and Cech (2017) have proposed the idea for learning a CNN from weakly annotated images. Their contribution was to connect the annotation with the right/correct face. They have used EM-CNN algorithm for prediction of attribute labels from un-annotated facial images and a CNN for age and gender estimation. They have achieved 41% a single face of images, 46% multiple detections of images and 13% no detection. Age and gender prediction was measured by Mean Absolute Error (MAE) (Franc and Cech, 2017). Yang *et al.* (2015) have presented the idea of a deep convolutional network (DCN) to detect the face by calculating the facial parts that are responded by their spatial structure and arrangement. In this paper, they have achieved to detect faces under acute

occlusion and unconstrained pose variation. Their accuracy rate proposed Faceness-Net was 90.99% compared to 85.67% of Cascade-CNN (Yang *et al.*, 2015). Abrevaya *et al.* (2018) have proposed the idea of a framework that is a combination of CNN encoder that allows their approach to apparently take benefit of redundant training data that showed the same factor, with a multi-linear-model-based decoder that was able to correctly reconstruct the training data and decouples shape. The framework learns a generative 3D face model with the advantages of both CNN robustness to corrupted and incomplete data and the multi-linear model capacity to effectively model and decouples shape variation. Their method was capable of learning from more magnitude data (Abrevaya *et al.*, 2018).

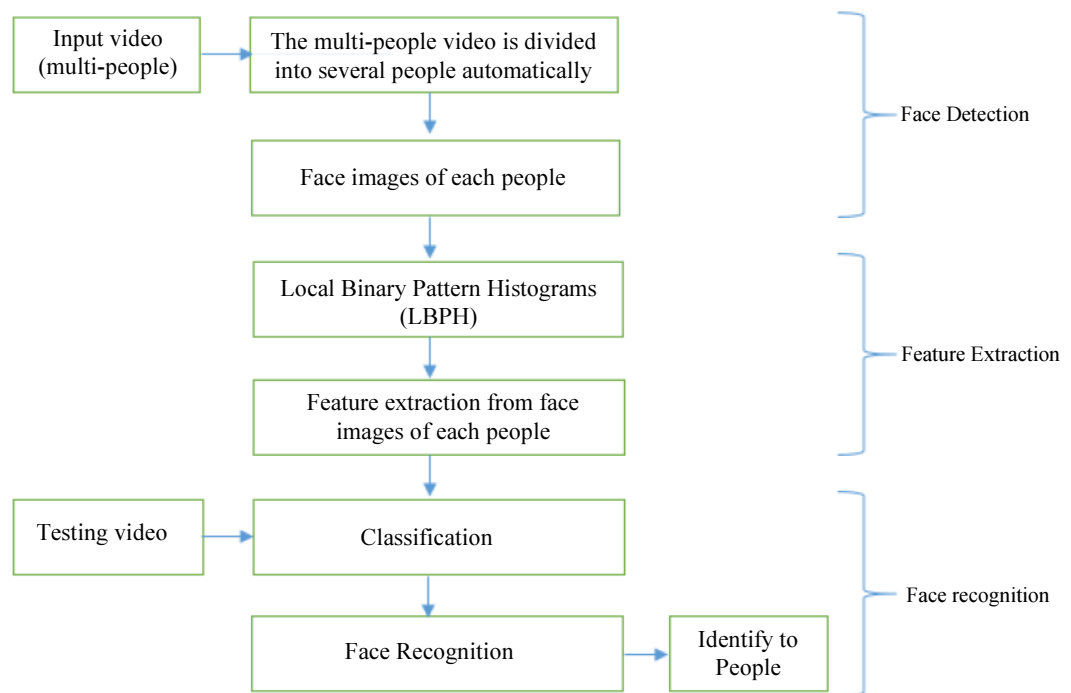
The rest of the paper is organized in two parts of each section ( for both LBPH and CNN) as follows: Section 2 presents the methods regarding the issue of this work where section 2.1 present the work on face detection, afterward, section 2.2 covers about feature extraction and section 2.3 present the deeper insight into face recognition. Section 3 explains the results of the works. Section 4 includes discussion. Conclusion and suggestions for future work are presented in Section 5.

## Methods

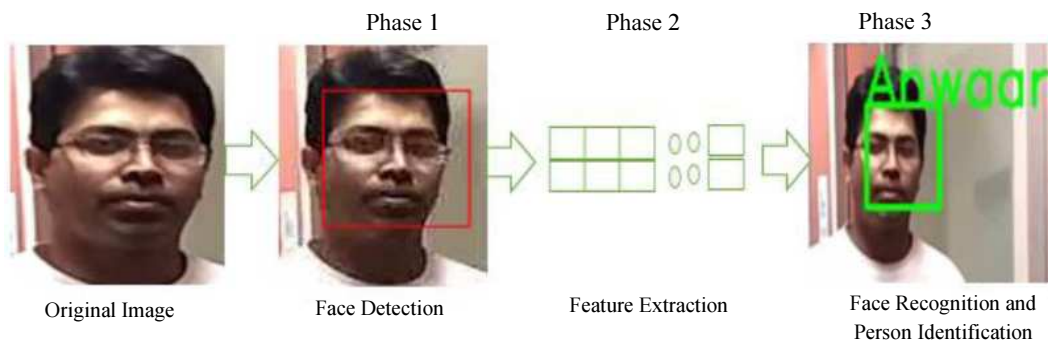
### *LBPH for Face Detection and Recognition*

In this section, we present our framework and focus on several factors for face detection and recognition for person identification in the video scene (Fig. 2). The processed used in our proposed system can be explained using the following flowchart.

As we shown in the above flow chart input to our program is a video (Wolf, 2018). The program works in three different phases namely face detection, feature extraction and the face recognition stage for person identification. In the first phase, our program tries to detect the faces in the input videos. When the face is detected, the program draws a rectangle and tracks the faces in the video (Yaseen *et al.*, 2018; Rani and Prasath, 2017; Singh and Sahran, 2018). Detection of faces in a given video is independent in a video. Once the face is detected by the program, the next phase is to extract the features correctly to recognize the person. We have trained our model for a certain number of people by extracting facial features from some videos. Now with the help of the trained model, in the last phases, our program tries to recognize individuals in the video. After face recognition, whether it is successful or not it goes to the next step i.e., person identification. The following (Fig. 3) shows each step of our works.



**Fig. 2:** Flowchart of the whole system



**Fig. 3:** Steps of Face Recognition System by LBPH

### Face Detection

In computer vision literature, face detection and recognition is always one of the most significant studied topics. The main purpose of face detection and recognition of a face from an arbitrary image that contains a number of challenging issues. To build the system there are a lot of challenges considering the variations in pose, lighting, occlusions and rotation of the image, scaling factor and expression of the face. In the last few decades, hundreds of methods have been proposed to and recognize the face. In this section, we present two face detection and recognition methods namely haar cascade classifier and Local Binary Pattern (LBP) classifier, both methods are analyzed and evaluated to choose the one method that suits our face

recognition goal. Obviously, the target is to achieve the best accuracy and computational efficiency as well. In addition, on the last part of this section, we discussed the most important similarities and differences of them and suggested one method that is eventually better for our system.

**Haar Cascade Classifier:** Haar cascade is explained by two things haar-like features and cascade classifiers. Haar-like feature considers adjacent rectangular regions at a specific location in a detection window, sums up the pixel intensities in each of the regions and then calculates the difference between these sums. Due to its intuitive similarity with haar-wavelets, these features get their name as haar-features. In the face detection phase, the window is moved over the input image and for each-

subsection, Haar-like features are calculated. The difference is then compared with a learned threshold value to detect the presence of an object. But for accurately detecting an object, a large number of Haar-like features are required. To implement this algorithm, we need a large number of positive and negative samples where positive samples indicate the presence of an object and negative samples indicate the absence of the same object to train the classifier. For this, we need to apply each and every feature on all the training images and for each feature calculate the best threshold value which will classify the images as positive or negative. We select the features with the minimum error rate and this process is continued until the required accuracy or the required number of features are found. These features cannot detect the object on their own but when used together they form a strong classifier. Let's consider a scenario in which the task is to detect a face. The major part of the image is a non-face region. For this situation, the concept of the cascade of classifiers was introduced. Instead of applying all the features on an image at once, a group the features into different stages applied them one-by-one. So if a region fails in the first stage itself, the classifier doesn't need to apply the remaining classifiers. We apply the second stage if it passes the first stage and so on. Thus, the region which passes all these stages contains the object or in this case a face. Figure 4 shows the procedure of face detection.

**LBP Classifier:** The basic theme on which Local Binary Pattern works is to achieve a local structure of an image by comparing each pixel with its neighborhood. A

pixel is taken as the center and it is compared with all its neighborhood pixels. It is assigned to 1 if a pixel value of neighborhood is greater than the pixel value of the center otherwise it is assigned 0. In this way shifting the center each time we end up with a binary matrix of an image containing relative values. For example if we are using 3×3 neighborhood like the matrix shown in example (Fig. 5), we have 8 neighbors and after comparison each time we will end up with 8 digits binary number corresponding to each center, for example (Fig. 5) shown below comparing the value at center to neighborhood values results in binary matrix with values 11011001. In this way, for 8 digits there are 28 combinations known as Local Binary Patterns.

A more formal description of the LBP operator can be given as:

$$LBP(x_c, y_c) = \sum_{p=0}^{p-1} 2^p s(i_p, i_c) \tag{1}$$

where, in Equation (2),  $(x_c, y_c)$  as the central pixel with intensity  $i_c$ ; and  $i_p$  being the intensity of the neighbor pixel and  $s$  is the sign function defined as:

$$s(x) = \begin{cases} 1 & \text{if } x \geq 0 \\ 0 & \text{else} \end{cases} \tag{2}$$

The value of  $i_c = 7$  in example (1) and after the application of Equation (1) and Equation (2) on the given matrix, we got binary value = 11011001 corresponding to center.

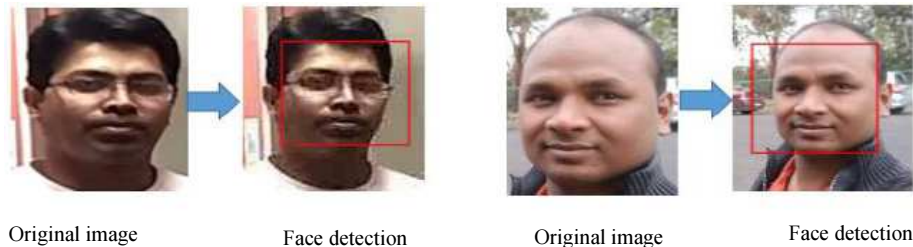


Fig. 4: Detection of the Faces

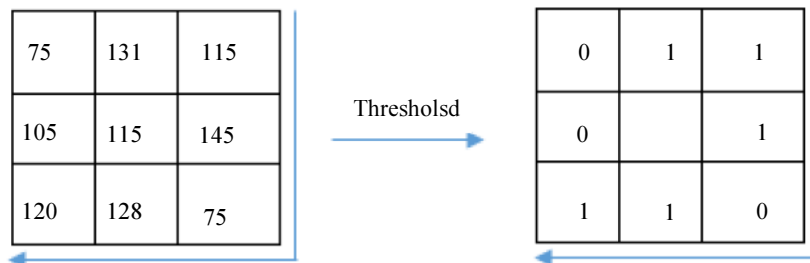


Fig. 5: The matrix of the Threshold of 3×3 of LBPH



### Feature Extraction Using LBPH Algorithm

The main provocation of face recognition to identify a person and to identify the person's face. Proper feature from that person's face needs to be extracted which is unique for that individual person, but that is robust for the appearance of facial variability. Correct features of the face should also be extracted fast to make this process fast. LBP method is used to extract the right feature for each person which will find the similarities between faces in the video. Even though there are many of the LBP based method developed to extract features, we are using Histogram Oriented Gradient (HOG).

Histogram Oriented Gradient (HOG): It is one of the feature descriptor used in computer vision and image processing. Histogram of oriented gradients descriptor can be described by the distribution of intensity gradients

or edge directions in an image. The image is divided into a small connected region called cells. A cell can contain several pixels and for each of the pixel, a histogram of a gradient is made. The descriptor contains histogram of the gradient of each and every pixel. For better accuracy the HOG is contrast-normalized, this is done by calculating intensity over a larger area several cells are known as block, then this value is used to normalize all cells within that block. The normalized result gives better performance on variation in illumination and intensity. HOG descriptors have some advantages over other descriptors such as it is invariant to geometric and photometric transformations except object orientation. It is particularly suited for human detection in images. Figure 6 shows the feature extraction of the face and the calculated values of the local binary pattern.

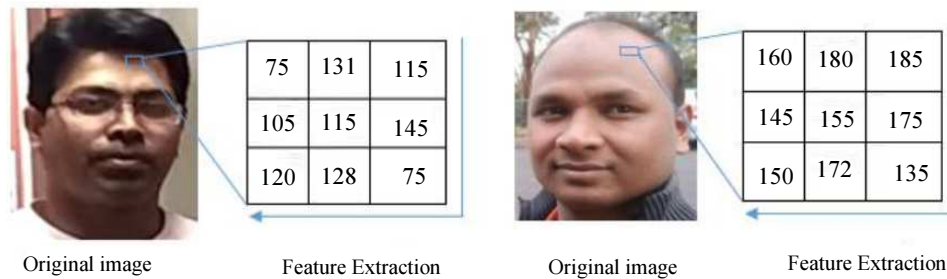


Fig. 6: Feature extraction from faces

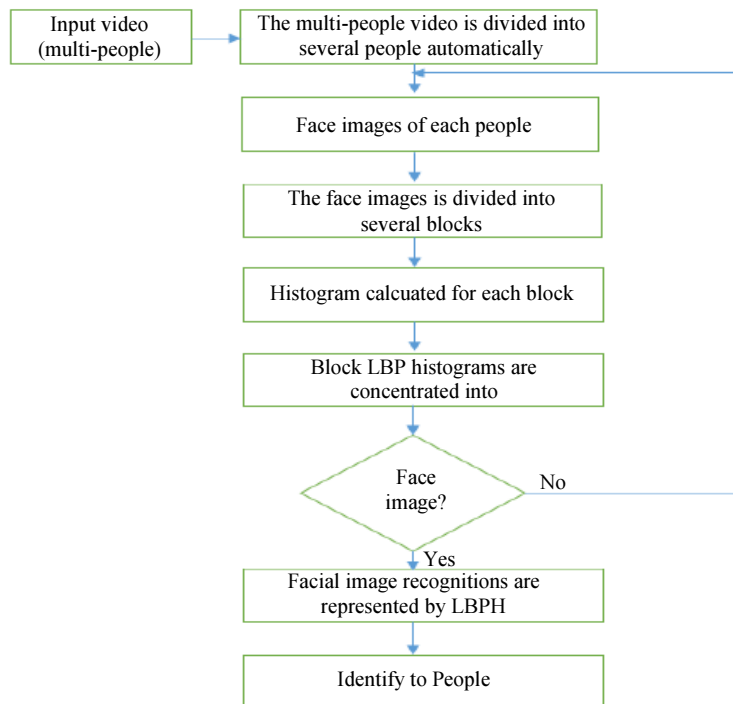
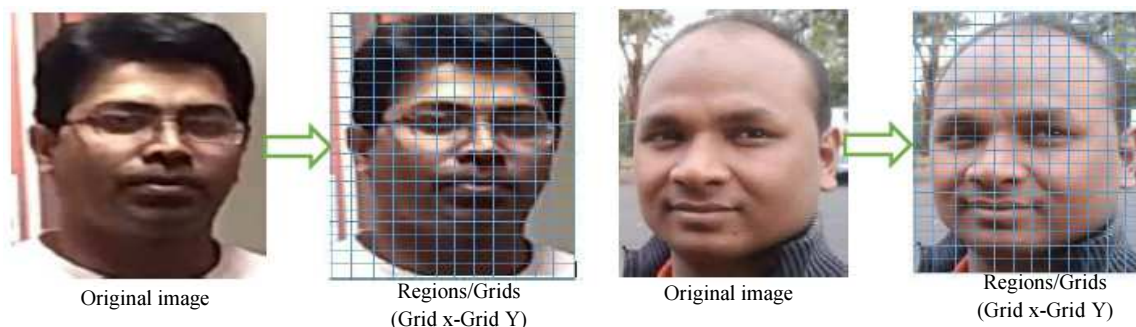


Fig. 7: Flowchart of the LBPH



**Fig. 8:** Region of Interest of Images example

### Face Recognition and Person Identification

To identify a person, face recognition is one of the most significant applications of computer vision. Although many techniques have been applied in this area during the last few years, in this paper, for doing this significant task we propose Local Binary Patterns Histograms (LBPH). Jiang and Miller (2017) have proposed an algorithm mainly focused on face recognition, whose aim is to efficiently summarize the structures of images by calculating local binary operator. The algorithm has popularity because it is quite simple in computation and it has discriminative power (Zafeiriou *et al.*, 2015). In recent years, it has played an increasing interest in many areas of image processing and computer vision and has shown its effectiveness in a number of applications, in particular for facial image analysis, including tasks as diverse as face detection, faces recognition, facial expression analysis, demographic classification, etc.

**LBPH Algorithm:** In our paper, we have proposed the Local Binary Patterns (LBP) methodology to implement face detection and recognition for person identification. Fig. 7 represents the flowchart of the LBPH based face recognition system. To identify a person, first the system search for the region of interest which is shown in Fig. 8. The system works on local facial features that summarize the face image and extract the special structure from the image using the LBP operator (Chung *et al.*, 2017). LBP operates with a set of nine pixels making a pixel as center pixel and then comparing the eight other pixel value with the center pixel value and then compute a binary number of the comparison.

**Input:** Training face image dataset

**Output:** Features are extracted from each face and compare with the center pixel then detect the face and recognition with an unknown face:

1. Initialize  $var = 0$
2. For each face image  $I$  in the dataset

3. Initialize a pattern of histogram,  $H=0$
4. For each center pixel  $t_c \in I$
5. Compute the pattern label of  $t_c$ ,  $LBP(1)$
6. Increase the corresponding bin by 1
7. END FOR
8. Find the highest LBP feature for each face image and combined into a single vector
9. Compare with the test face image
10. If it matches it most similar face in the database then successfully recognized

### Flowchart of the LBPH of Our System

#### LBPH Operation

Concerning the functionality, the region of interest in an image is split into an  $N \times M$ . The region of interest in the image (in our case face) size is the same for width height, resulting in  $m \times m$ .

The local binary operator is applied to utilize the region of interest. This operator applied to each image, the operator takes nine pixels and compares a pixel (the center pixel) to its eight neighbor pixels. The comparison returns 1 when it finds a higher pixel value than the center pixel value otherwise it returns 0 to other neighbor pixels. This procedure is applied to all 8 neighbors. The LBP derived 8 value which is binary values. The comparison is then formed an 8-bit binary number, by merging the resulting values of the comparison together. The formed 8-bit binary number can translate into a decimal value, which is called the value of pixel LBP, the values range in between 0-255. The operation is shown in (Fig. 9). This operation is performed for every pixel for each region of interest.

#### Extracting the Histograms

The histogram is created for the current region of the interest after calculation of the number of LBP values in the appearance of all region of interest. The histogram for each region of interest consists of 256 bins in this process. This following equation describes the process:

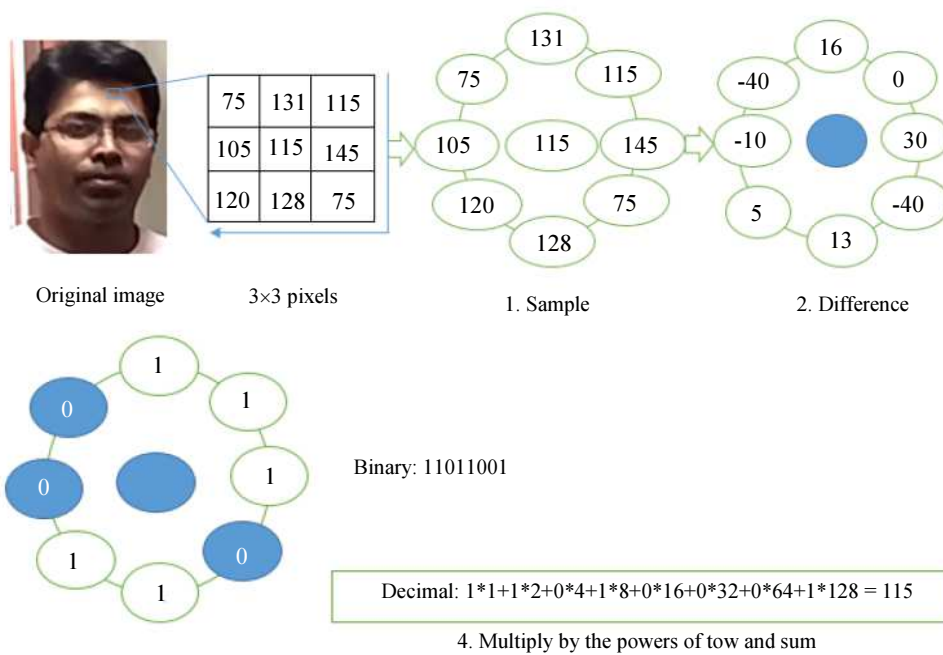


Fig. 9: Operations of Local binary patterns

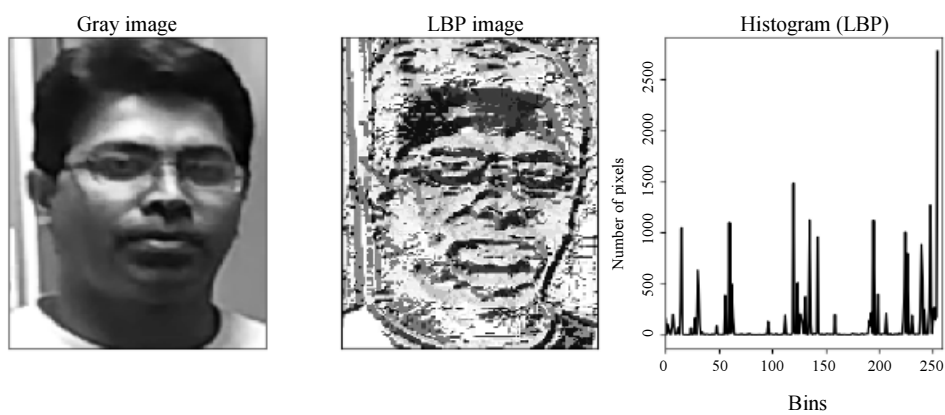


Fig. 10: Face image-Labels histogram example 1

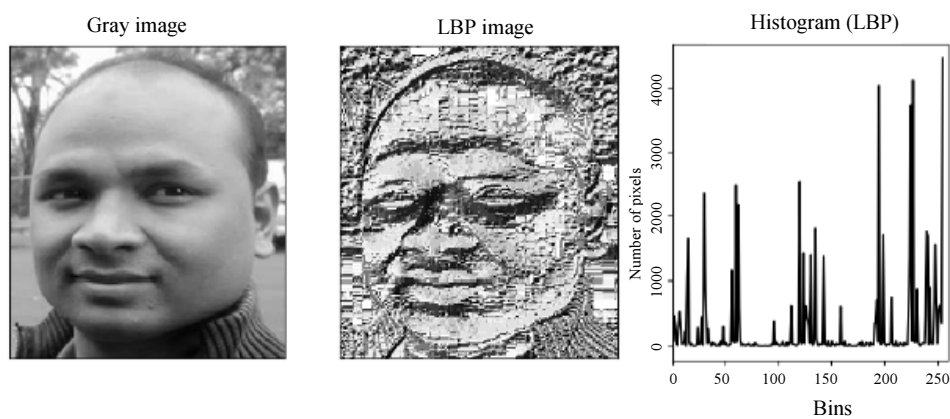


Fig. 11: Face image-Labels histogram example 2



$$H_i = \sum_{x,y} I \{LBP(J(x,y)) = i\}, 0, \dots, 255 \quad (3)$$

where,  $H_i$  is the value of  $I$  as  $H$  is the bin of value,  $J(x, y)$  represents the  $(x, i)$  pixel of an image whereas  $I$  represent the conditional operator, The process return 1 as true if the statement is true or 0 otherwise if the statement is false. After calculating the histograms for all the region of interest. The histograms for each region of interest section then formed into a single histogram. This single histogram contains  $256 * m * m$  bins which are defined the image as a feature vector.

A Data Base (DB) of faces which contains the feature vectors for each corresponding face images is required to make the face recognition proceed. Then the different feature vectors are calculated for a distance of face images. The image is matched to another image in the database if the distance of the image is closest to the distance that corresponds to feature vectors of the face in the database. When the distance is closest it the best match for the image. The following equation shows the process:

$$(M, Index) = \min(dst[db\ size]) \quad (4)$$

where,  $dst$ , calculated distances array and  $\min$  represents the function that returns minimum value ( $M$ ) of the array and ( $Index$ ) is the values of the index. To exclude faces from the image a threshold is applied that that only correspond to the match image in the database and do not correspond to other faces in the database:

$$Output = \begin{cases} "Index\ is\ a\ match", & \text{if } M < Threshold \\ "No\ match" & \text{Other wise} \end{cases} \quad (5)$$

As shown above, The LBP is a very popular model for its simplicity and model is multiple and modified in a different way, implementations over time (Singh and Sahran, 2018; Wolf, 2018). The model has accelerated the implementations efficiently as it is utilized in a different way such as GPUs and SIMD instructions, this is reported in the past (Juránek *et al.*, 2010). LBPH known to the authors are (Kryjak *et al.*, 2012) as the only FPGA implementations and concerned both face detection. This proposed system has an impact on face recognition revolution and this system introduced a novel LBPH face recognition implementation.

### CNN for Face Detection, Recognition and Gender Estimation

The term Convolution Neural Network (CNN) is a class of deep neural networks that is applied to classify

or analyze visual imagery. Researchers find a various way to use CNN as CNN's use variation of multilayer perceptions that is designed to minimal pre-processing requirements. The convolution neural network is a type of artificial neural network that can be used to recognition tasks, classification tasks or other data analysis tasks as CNN's multilayer is able to pick out or detects patterns, images and make sense of them. CNN's can detect (Li *et al.*, 2015) patterns, faces are so useful to classify image analysis. The CNN has hidden layers that are called convolution layers that is the basis of CNN. It also has other non-convolution layers as well. The convolution neural networks actually inspired by biological processes more preciously from the connectivity patterns between neurons of the human brain. CNN is a deeply complex hierarchical structure as it is inspired by the structure of neurons and connections of the brain that plays a big role in remembering and labeling objects. CNN follows the more complex and deep neural networks (Parkhi *et al.*, 2015; Sun *et al.*, 2015) but relatively little pre-processing.

The CNN consists of an input and an output layer between these two layers there is more complex and that includes the multiple convolution layers, pooling layers, fully connected layers and normalization layers which are called hidden layer. CNN's have two components (1) hidden layers or feature extractor that is designed to take classify the 2D structure of an input image and (2) the classification part which is fully connected layers. The convolution layer as it receives an input from the previous layer then transforms the inputs that are specifically designed to process pixel data and then the layer outputs the transformed input to the next convolution layer. These operations of the convolution layers called convolution operation. To use a convolution layer, the number of filters of the layers should be specified. The filters are able to detect patterns (like edges, circles, triangles, noise) and deeper the convolution layer that can detect more sophisticated patterns (like objects, eyes, ears, etc.) and if the convolution layer is deeper than it can detect dogs, cats, human faces, etc. It depends on how deep the convolution layer is! A convolution neural network learns from a dataset that includes lots if not millions of pictures then the model can generalize the inputs (images, video, live feed, real-time videos, etc.) and make predictions for images that are not shown to the model before.

The CNN's have a different architecture as it is complex (Farfade *et al.*, 2015) and act likes human neuron and compared to other neural networks CNN are a bit different. In CNN's, the layers are organized in three dimensions and the layers measures width, height and depth of the image from an input. And the output of the neurons in one layer does not connect to all the

neurons in the next layer rather than that the layer connects a small region of the output neurons to the next neurons. In every neuron of a neural network assigns an output value that comes from the previous layer. The output value computes by applying some function to the input values. The functions are specified by a vector of weights and a bias. Researchers can also assign random weights to each neuron by assigning random value. The vectors and bias (called a filter) that learns and progresses by adjustments of the weights of each neuron and the CNN's.

Our model takes images with 3 channels and each image size is 200 by 200. The model is pretty straight forward CNN model until the bottleneck layer. We have used the same model for two different tasks and the model works fine with two models. The bottleneck layer

of the model is output 1D tensors. The bottleneck layer's output is used as an input to a dense layer than we have fed that to another layer with soft max activation for recognition, Gender classification tasks. Another important part is loss function, for recognition, gender classification tasks we have used categorical loss function. But for recognition task, we have used ADAM as an optimizer, for gender, we have used SGD optimizer.

Figure 12, we have discussed the total system of our program. The Convolution Neural Networks is a class of deep learning and CNN is more than machine learning algorithm and the CNN classifier learns pretty accurately and can classify as well. The Deep learning algorithm is like artificial neural networks.

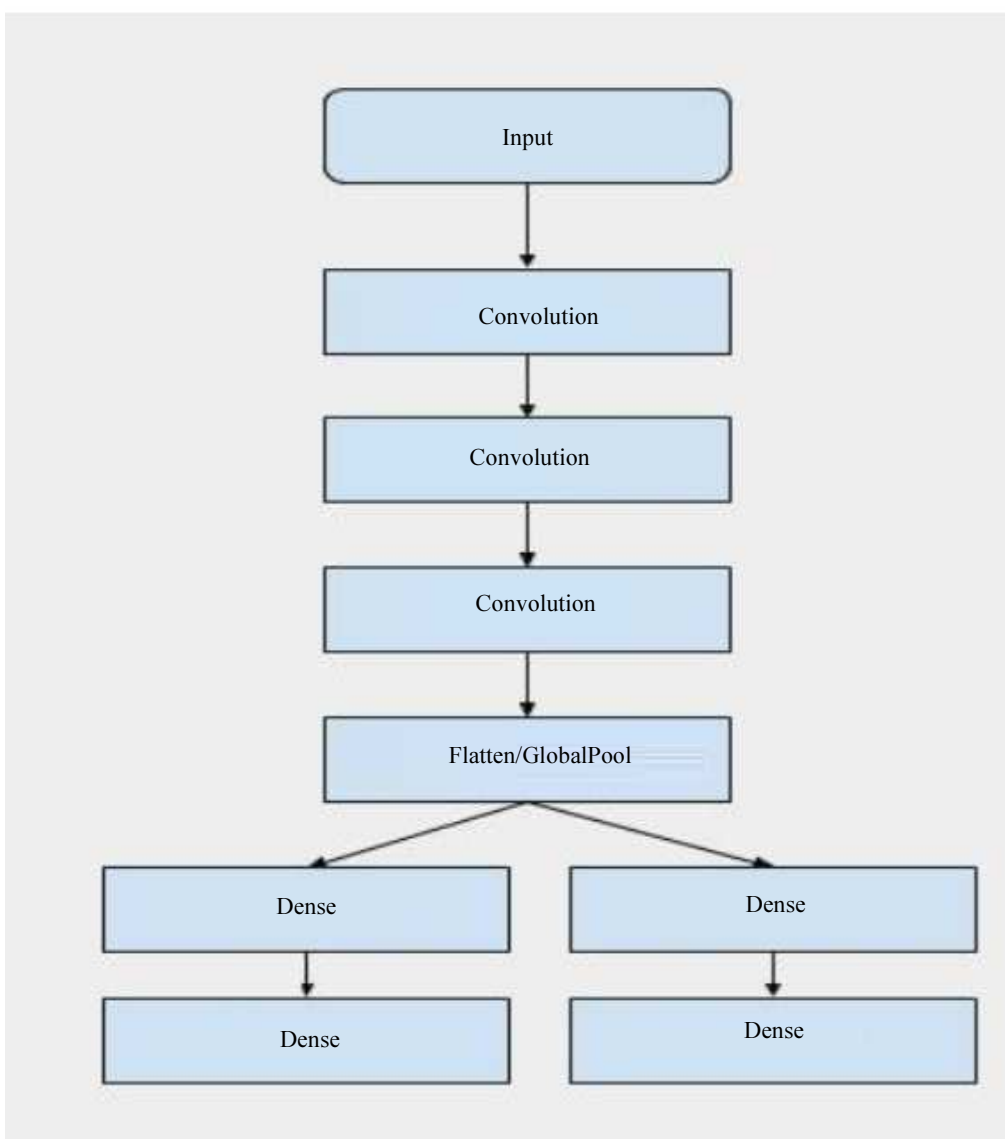


Fig. 12: CNN model of our system

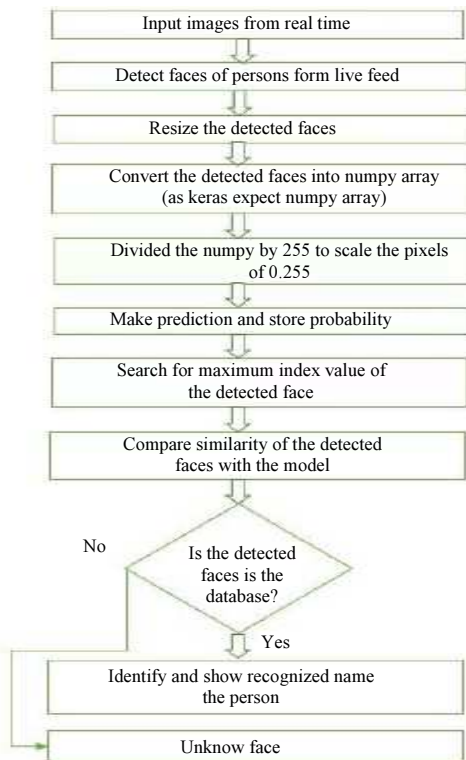


Fig. 13: Flowchart of the Face Recognition System using CNN

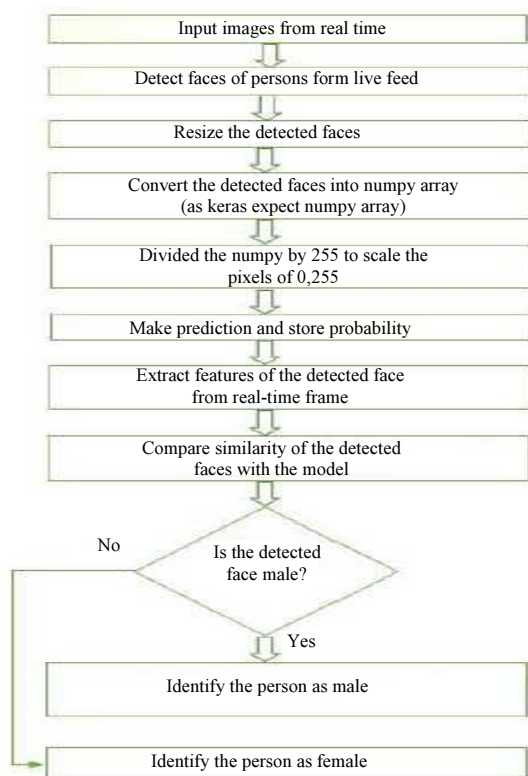


Fig. 14: Flowchart of Gender Estimation using CNN

**Algorithm: GR.**

**Input:** mixed labeled supervised data for gender  $D_G$  and Unlabeled data for recognition  $D_R$ .

**Output:** Predict the accuracy of the data/Test labeled data  $T_{DL}$  by using machine learning model  $M^*$ .

**Method:-**

1. Label and divide the mix labeled data for gender  $D_G$  categorization.
2. Crop and resize the unlabeled data and Label the data for recognition  $D_R$  category.
3. Divide the labeled data  $D_G$  and  $D_R$  in train, test and valid dataset.
4. Create two model  $M_G$  for gender and  $M_R$  for recognition using  $D_G$  and  $D_R$  according.
5. **For** each model of  $M^*$  **do**
6.     Pre-process the data before feeding into the network.
7.     **For** gender ("male", "female") **do**
8.         Predict the detected faces in real time and categorize either male or female;
9.     **End For**
10.    **for** recognition  $(n_1, n_2, \dots, n_N)$  **do**
11.        Predict the detected faces and show the name of the person;
12.    **End For**
13. **Return** predicted the accuracy of the face in real time.

In the below section, we present our framework and focus on several factors for face detection, recognition for person identification and gender estimation in real time. The processed used in our proposed system can be explained using the following flowchart.

Face Detection and recognition: Face detection has been one of the most studied topics in computer vision literature. The goal of face detection is given an arbitrary image containing an unknown number of faces (possible none) to localize the face and determine its size. This is a challenging task considering the variations in pose, lighting, facial expression, rotation, scale and occlusions. Hundreds of methods have been proposed in the last few years.

Figure 13, we present the face detection methods namely convolution neural networks that were analyzed and evaluated in order to choose what suits our goals best. Obviously, we target for the best possible accuracy but the computational efficiency is of great importance. In addition, on the last part of this section, we discussed the most important similarities and differences of them and suggested one method that is eventually better for our system.

Gender Estimation: Automatic gender classification of faces has become relevant to an increasing amount of applications, as the field of Artificial Intelligence is taking place side-by-side with the human.

Gender (Levi and Hassner, 2015) classification is a binary classification problem similar to face recognition system. To classify the detected face as male or female first, detect the face from a real-time frame and crop the faces from the frame and extract features and label the faces as male or female. To classify the gender estimation, we have used the CNN model to predict the faces as shown in Fig. 14.

### Experimental Result Analysis

The results of the experiments (for both LBPH and CNN) are presented in this section. The OpenCV and Python were used as the experimental platform. The process of experimental result analysis section can be split into two main phases. These are datasets and result. This implementation is used to test the performance of the LBP-method on different kind of face images. Several parameters, like the LBP operator (P and R), non-weighted or weighted regions and the dividing of the regions, are varied to see the influence of these parameters on the performance.

#### Datasets

##### Dataset for LBPH

In our experiment, we have created two different databases on the 3 videos which we used. Table 1 shows the dataset of the face recognition for LBPH-based

method and also shows the accuracy result of LBPH-based method. These databases consisted of total 2345 images has been taken for 4 different peoples. The average feature vector for each person was extracted then the system was loaded by the extracted feature. Figure 15 shows some sample images of the dataset that is used by our model.

We evaluated the proposed algorithm on the 3 videos of the second sequence. The First, second, third videos are 3.25, 0.15 and 0.14 min long respectively, shot at 11 fps, totaling 2345 frames per video encoded. The size of the frames is 300×200 pixels. A sample frame from this sequence, together with the detections generated by the LBPH method is shown in Fig. 9. The running speeds of the 3 videos are shown in Fig. 10 and 11.

##### Dataset for CNN

In our experiment, we have created two different databases for gender estimation and face recognition. These databases consisted of total 8000 (UTKFace for gender estimation-1), 14706 (CITYFace+UTKFace gender estimation-2) 3167(CITYFace Recognition-1), 3397(CITYFace Recognition-2) images of different peoples. The average feature vector for each person was extracted then the system was loaded by the extracted feature. Figure 16 shows the dataset for Face recognition using CNN. Table 2 in section 3.2.2 shows how the datasets are divided for training, testing and for validation.



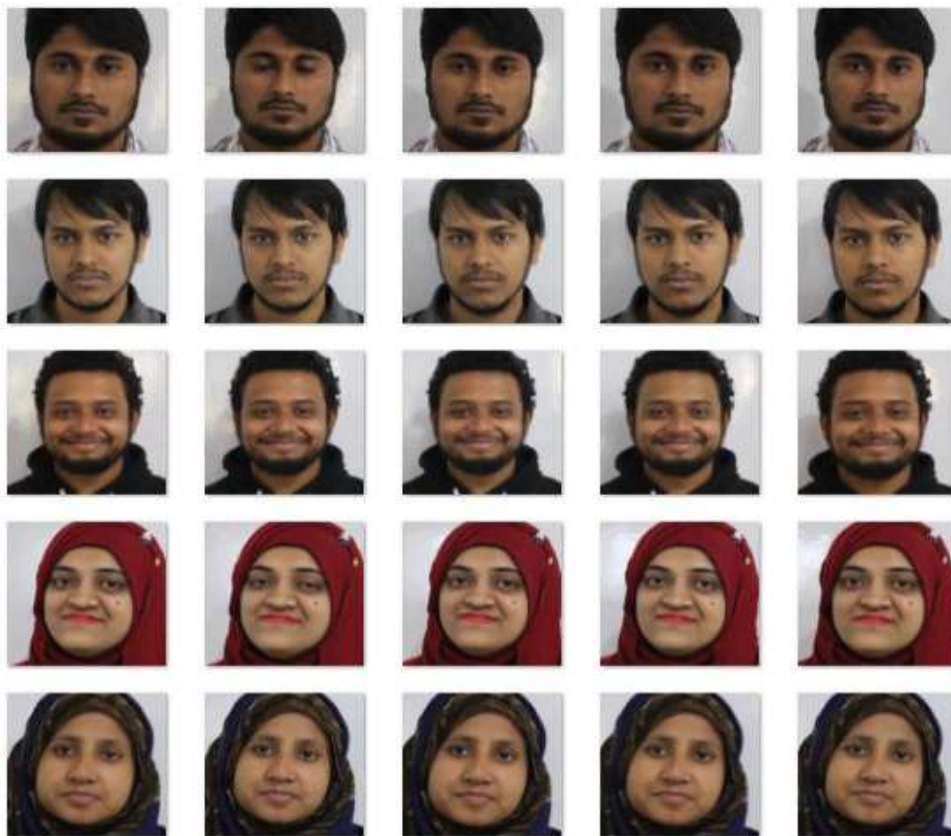
Fig. 15: Face recognition training databases for LBPH

**Table 1:** Dataset and Accuracy of the 3 videos using LBPH

Video	Divided Into the Number of frames	Number of face images stored in database	From the input video, number of face images compared with database	Recognized Image	Unrecognized Image	Recognition and person Identification Accuracy (%)
1	2245	2245	2245	1936	309	71
2	52	52	52	43	9	63
3	48	48	48	40	8	63

**Table 2:** Divided datasets for face recognition and gender estimation using CNN

Dataset	Total number of images in dataset	Training set	Validation set	Testing set
UTKFace (gender)	8000	6400	1600	2000
UTKFace +CITYFace (gender)	14706	9411	2353	2942
CityFace (recognition-1 )	3167	2550	617	345
CityFace (recognition-2 )	3397	2453	434	510



**Fig. 16:** Dataset for face recognition using CNN

## Results

In this section, the experimental results of the implementation are presented. We have already discussed the program works in the methodology sections. In this section, we have provided the details about the results obtained by us while using this program against some of the test cases. We have explained the

output of each and every test case using the screenshots of the output provided by our program. While making this project, we faced a lot of challenges and we have tried to minimize it as much as possible of these problems. We evaluated the proposed algorithm on the real-time video frame (using LBPH and CNN). We can see the input facial images used to create databases for face recognition are given above.



### The Result of LPBH Algorithm

In this section, the experimental results of the implementation are presented. We have already discussed the program works in the methodology sections. In this section, we have provided the details about the results obtained by us while using this program against some of the test cases. We have explained the output of each and every test case using the screenshots of the output provided by our program. While making this project, we faced a lot of challenges and we have tried to minimize it as much as possible of these problems. As mentioned above, we used our own created database. The database consisted of total 2345 images that are 4 different people. The average feature vector for each person was extracted then the system was loaded by the extracted feature. To identify a person, we performed our experiments face detection and

recognition based on the above mentioned LBPH algorithm. In the proposed algorithm, different types of face images have been recognized. Based on the algorithm, the face image of unknown identity is compared with face images of known individuals from a large database. During testing, we recognize the face for which there exists a weighted combination of basis images that is the closest to the test face image. Figure 17 we can see the input facial images used to create databases for face recognition are given.

As mentioned before, we have considered a constraint that the input video must be shot in a homogenous domain. And also that the system will recognize only those faces for which it is initially trained. So for database 1, we have the trained the machine for two peoples. One of them is labeled as ANWAAR and the other one is labeled as TINA.

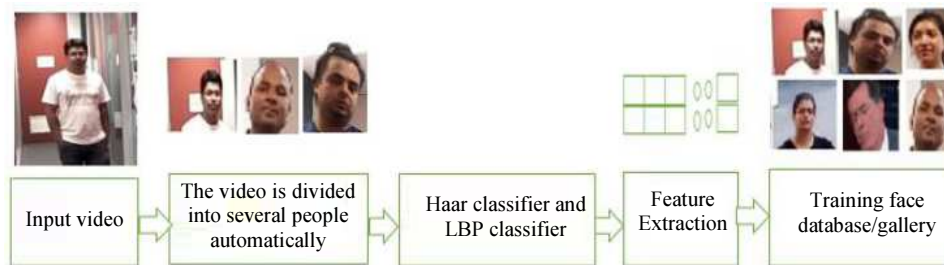


Fig. 17: Face recognition-training face database block diagram using LBPH

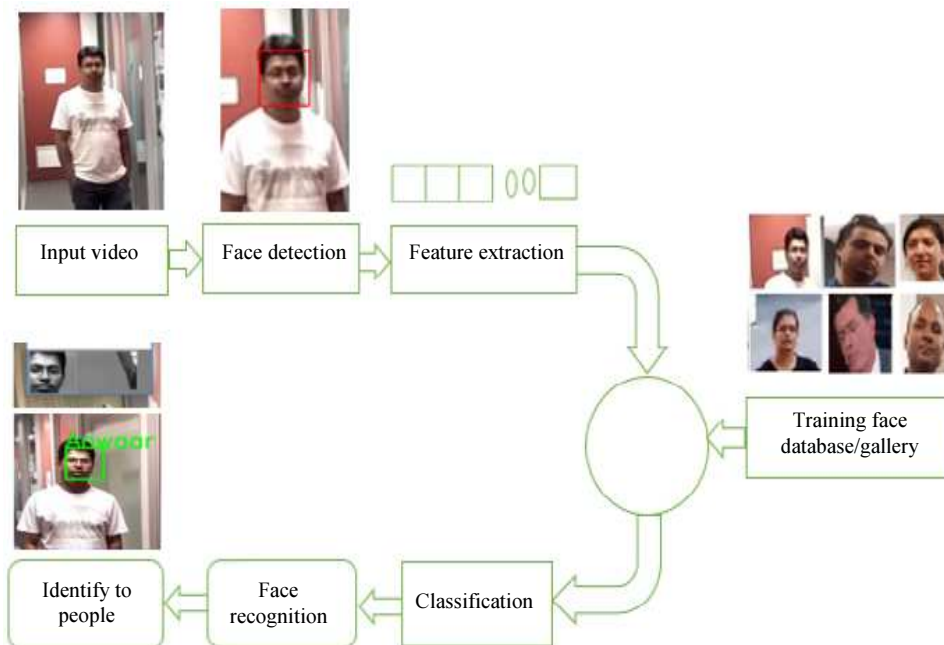


Fig. 18: Face recognition-testing video for person identification block diagram

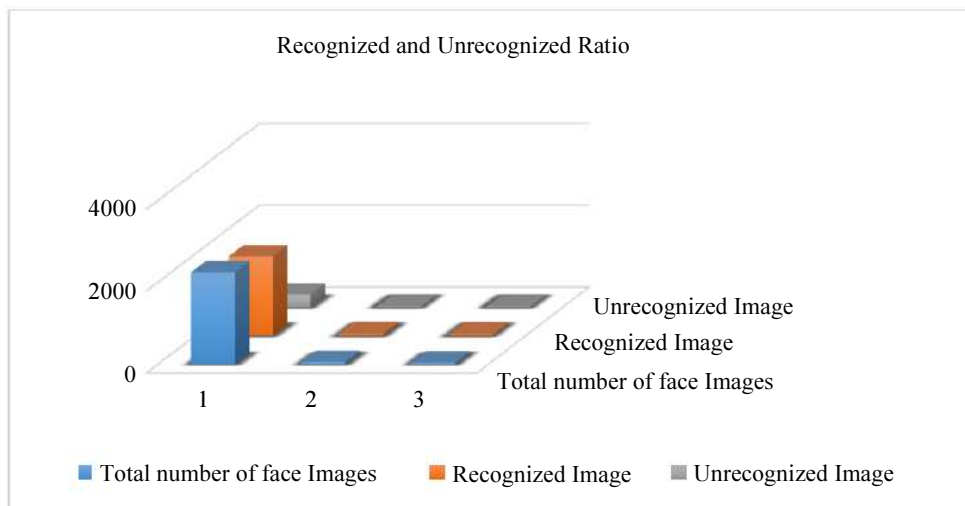
Similarly, database 2. We have provided the output for three test videos where each of the test videos represents different scenarios. We have tried to represent the different stages of face detection, tracking and recognition in the results provided by the program. The following Fig. 18 shows the different stages of the output provided by the program when the test cases were given as input videos.

And also in Fig. 18, we can see the facial images that are stored in the database which compared with the input facial images. If the input face images are found or the more similarities face images are matched in the database then we say the face image is successfully recognized. Based on the algorithm the input face images are compared with database facial images for identification. The face recognition with identification results are shown.

The screenshots of the output provided by the program for the first video are shown in Fig. 19a and b. For the first video, only one person will be seen. We have trained the machine to recognize him and label him as ANWAAR. Similar to video 1 in the second video too, only one person will be seen. We have trained the machine to recognize her and label her as TINA. Initially, the frame is empty. When the person enters the frame, the program correctly detects the presence of a person and draws a rectangle around them. The program has correctly recognized the presence of person 1 and 2 as seen in Fig. 19a and b and it continues to track their face and correctly label it as ANWAAR and TINA. Figure 20 shows the result of the number of recognized and unrecognized images.



**Fig. 19:** Output of video 1 and 2: Single person video; Face is detected, recognized and identified



**Fig. 20:** Recognized and unrecognized ratio of input video 1, 2 and 3

### Result of CNN

The result of using Convolution Neural Networks is presented in this section. We have used CNN classifier (a class of deep learning) to recognize a face and to estimate the gender of the detected face from a real-time video frame. The whole system of Face Recognition using CNN is shown in Fig. 21.

As mentioned before, we have considered the input is from the real-time video frame. The system will recognize only those faces for which it is initially trained. For the database, we have the trained the machine for five persons. Each of them is labeled as 0 to 4 and 0 labeled images mean ABDUL AHAD, 1 labeled images mean -JUBAIR AHMED so on. We have tried to represent the different stages of face detection and recognition in the results that are gained by the model.

Figure 22, we can see the facial images that are stored in the database which compared with the input facial images. If the input face images are found or the more similarities face images are matched in the database then we say the face image is successfully recognized. Based on the algorithm the input face images are compared with database facial images for identification. The face recognition with identification results are shown.

The screenshots of the output provided by the program for the first system and. It shows in Fig. 22 that our programs classify the detected faces correctly in a busy background. We have trained the model with 5 class of each person and after 10 epochs, our CITYFace Recognition-1 model is 99.88% accurate and it is classifying the detected faces and giving the faces names as shown in the above picture. Meanwhile, our CITYFace Recognition-2 is 100 % accurate when we trained our model with our previous trained weight of CITYFace Recognition-1. After loading the weight, we have achieved 100% accuracy only after 6 epochs.



Fig. 22: Face Recognized from real-time video Frame

The screenshots of the output provided by the program for the second system and. It shows in Fig. 23 that our programs classify the detected faces correctly in a busy background and classify the images either male or female. We have trained the model with 2 class of 0 means "male" and 1 means "female" and after 16 epochs, our gender model name UTKFace is 96.88% accurate and it is classifying the detected faces and giving the faces names as shown in the above picture. Meanwhile, our second model with more images of UTKFace+CITYFace dataset which is trained after loading our previous model weight. Our second gender estimation model is giving 93.38% accuracy only after 6 epochs. Table 3 shows the accuracy for training, testing and validation using CNN class of deep learning.

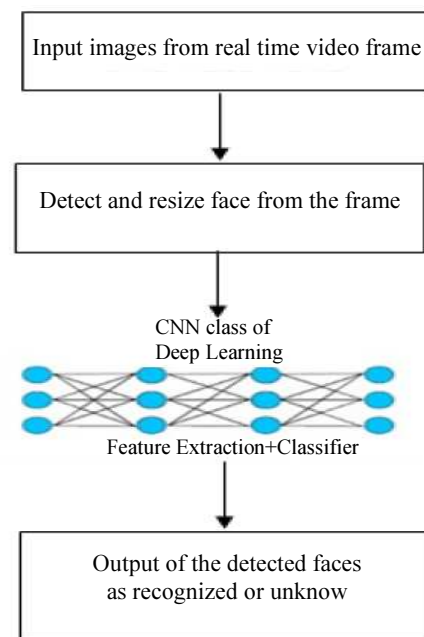


Fig. 21: Block diagram of Face recognition using CNN

**Table 3:** Accuracy of face recognition and gender estimation using CNN

Dataset	Training Accuracy	Validation Accuracy	Testing Accuracy
UTKFace (gender estimation-1)	96.88 %	87.38 %	93.90 %
UTKFace + CITYFace) (gender estimation-2	93.38 %	86.81 %	83.96 %
CITYFace (recognition-1)	99.88 %	86.47 %	64.37 %
CITYFace (recognition-2)	100%	100%	100%

## Discussion

This section is expected to give a review of our paper, issues found during the research, the results gained, and also to figure out the future directions. The brief discussion of face analysis area infers a great deal of conceivable or potential applications for researchers. In the field of banking, surveillance, and multimedia equipment, this application is becoming essential. For this reason, the group of face analysis has adequate work keeping in mind the end goal to totally understand the face discovery and acknowledgment challenges, most vital of which was mentioned in the methodology section. Note that every application identified with face analysis has different requirements in the investigation procedure. This is the principle motivation behind why all algorithms and methodologies for face analysis are application dependent and standardization or generalization is very difficult, at least for the moment.

The actual contributions of face detection, tracking and recognition applications are useful to empower the face analysis research group to keep assembling more strong frameworks by connecting diverse methodologies and joining them. Hence, the key point of this research is to contribute by investigating the Local Binary Patterns Histogram, inspired by the following reasons. First, it can be applied to face detection, tracking and recognition for person identification; accordingly, its implementations empower the likelihood to explore in research fields with a low computation complexity. And then, to show its power to posture and illumination changes. Following the structures of our research paper are: before achieving the experimental outcomes, methodology section is a key part to consider to prevail with such a sort of learning algorithms where input samples are required to train the framework. Section 3 presents the experimental result analysis. Firstly it introduces the images databases used. Instead of using existing databases, we have made our own particular database adjusted to the prerequisites of this research. In the case of face detection experiments, two different methodologies have been utilized to discover which one of them is the best choice and besides to exhibit how essential the training samples for the last framework performance are. The experimental results are first focused in a face detection algorithm (see section 2.1) used as a face detection stage. Then the feature extraction (see section 2.2). The next step in the experimental section concerns the face recognition system (see section 2.3), where before implementing the

face detection method proposed in the methodology. Obviously, the detection and recognition ratios results obtained are sufficient enough to perceive how a simple texture operator can be used as a face descriptor.

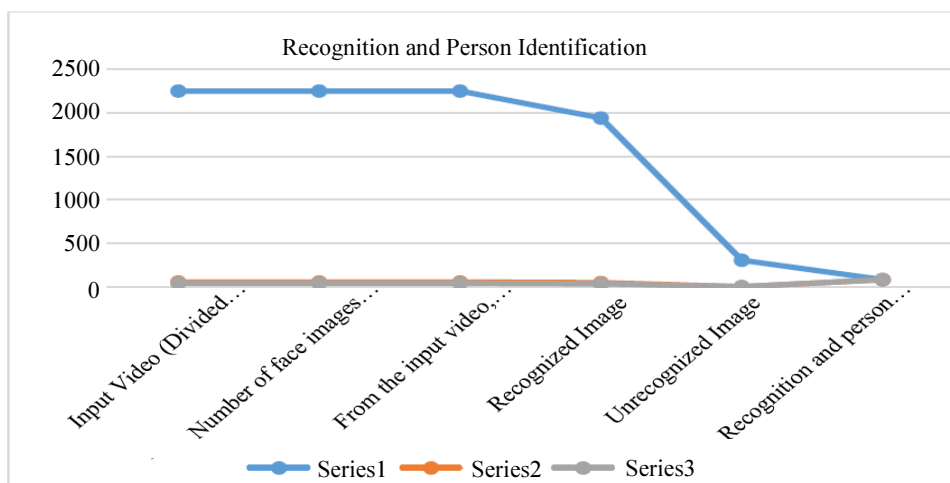
We mentioned before, the first part of the methodology is dedicated to the face detection method (see section 2.1). The second section of the methodology is feature extractions (see section 2.2). And then the last part consists of implementing the face recognition (Jiang and Miller, 2017; Zafeiriou *et al.*, 2015) (Chung *et al.*, 2017) using the LBPH method. In the face recognition case, the idea of using LBPH is more important, since LBP enhanced featured vector for the face recognition task consists of an excessive number of elements.

It should be noted that when testing LBP in face detection task (2.1), non-negligible values are obtained when using only faces samples for training the system. For future works, it would be interesting to continue exploring the use of only face class since it is not easy to find the best non-faces set that best describes all non-face texture possibilities. In order to increase face detection and recognition ratios, rotated face samples should also be added to the training databases. Moreover, also for future improvements, the face recognition systems implemented could be improved by solving the problem of having an external individual, not present in training samples that try to identify itself. In the way in which it is done now, the individual will be identified with the individual Name with minimum distance, but it should be rejected by the system. Therefore, some kind of maximum distance threshold should be used to avoid this problem.

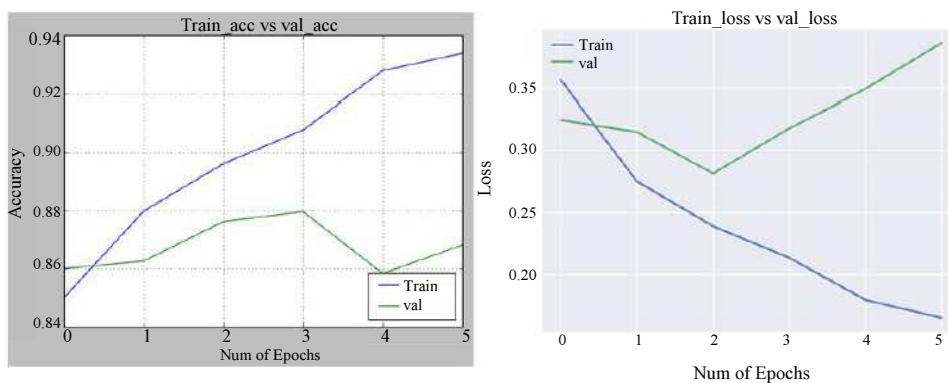
As already mentioned in experiments and results section, all the above analyses have been tried to use python because of the likelihood of using OpenCV libraries which contain complete and open source image processing and machine learning algorithm. Figure 24 illustrates the graph of face recognition used by LBPH algorithm. In addition, Fig. 25 represents the training and validation accuracy ratio per epochs for gender estimation (For dataset 2) using CNN and Fig. 26 represents the training and validation accuracy ratio per epochs for face recognition (for dataset 2) using CNN. Also, another inspiration is the likelihood of making a Windows dialog based application with a friendly GUI to test the entire framework with static images and video mode.



**Fig. 23:** Output of gender estimation face detected faces in real-time



**Fig. 24:** Graph for face recognition using LBPH



**Fig. 25:** Graph for training and validation accuracy and loss of gender estimation-2 using CNN

The fundamental expectation of the created FaceProcessingTool application is to assess experimental results and it could be enhanced, for instance, by speeding LBPH calculation utilizing multi-threading, e.g. each down-examined image

handled in a new thread. For future work, the LBPH strategy is presently accessible in python and OpenCV so we will use deep learning to take advantage of powerful image processing toolboxes to evaluate other method combinations.



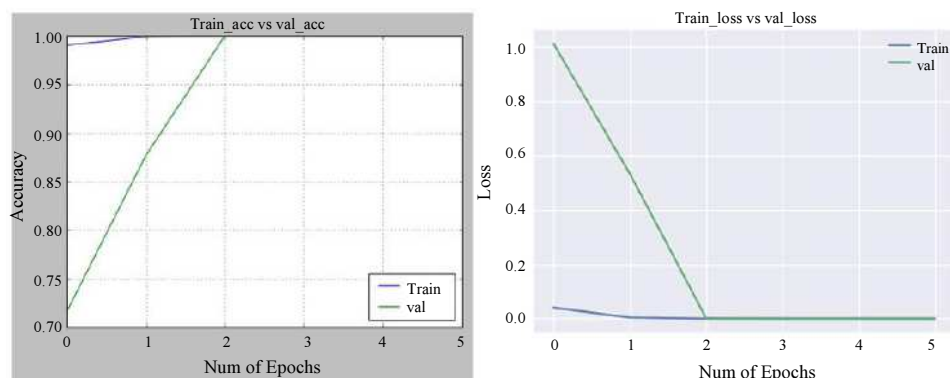


Fig. 26: Graph for training and validation accuracy and loss of Face Recognition-2 using CNN

## Conclusion and Future Work

To identify a person, in this paper we have proposed both proposed Convolution Neural Networks (CNN) and Local Binary Patterns Histograms (LBPH) approach for extracting the features and matching the process for face detection, tracking and recognition. Note that the same methodology had been applied to any other task that builds on localization, such as face tracking, face detection separately and the same CNN methodology had been applied to gender estimation separately. We have first shown that current measures used in face detection, tracking and recognition jointly. We have proposed this method to identify a person specified by the use of particular facial feature extraction. A training model was considered as a collection of training images and a testing model was then obtained by our proposed technique. Experiments were performed on our own created two databases namely A and B. Results have shown that the proposed LBP-based face recognition approach performs better (positive rate of recognition 87%) and is much faster for both databases. Even though our experimental results on both databases show that our method was performing very well for face detection, tracking and recognition jointly to identify people, in the future study we will apply deep learning techniques for doing this research work to improve the positive rate of recognition to identify the person. A training model was considered as a collection of training images and a testing model was then obtained by our proposed technique. Experiments were tested on persons on a real-time video. The gender estimation and recognition tasks performed very well. Future, we will work on age estimation and facial expression recognition using CNN. By the end of the paper, we can say that our CNN model performs better than the LBPH model. CNN is a class of deep neural networks and it is an A.I. (artificial intelligent).

## Acknowledgement

The authors would like to thank the support rendered by City University, Bangladesh to carry out this research

work. They are also grateful to the faculty members of computer science and engineering department, City University Bangladesh for their help in evaluation process.

## Author's Contributions

**Md Hannan:** He implemented this work. He contributed to part of the literature collection and drafted the manuscript in consultation with Md Rafiqul Islam.

**Md Rafiqul Islam:** He devised the main conceptual ideas and supervised this work. He contributed to figure out the whole work as a research paper. Besides, he provided critical feedback and helped to shape the research and manuscript.

**Md Ashraful Haque, Md Safaet Hossain, Anwaar Ulhaq and Jeyson Jaman Sawan:** They are equally contributed to write, edit and revise the manuscript.

## Ethics

This research manuscript is original and the manuscript is not published elsewhere. The corresponding author ensure that all authors have read and approved the manuscript and no ethical issues involved.

## References

- Farfadi, S.S., M.J. Saberian and L.J. Li, 2015. Multi-view face detection using deep convolutional neural networks. Proceedings of the 5th ACM on International Conference on Multimedia Retrieval, Jun. 23-26, IEEE Xplore Press, Shanghai, Chinam, pp: 643-650. DOI: 10.1145/2671188.2749408
- Abrevaya, V.F., S. Wuhner and E. Boyer, 2018. Multilinear autoencoder for 3d face model learning. Proceedings of the IEEE Winter Conference on Applications of Computer Vision, Mar. 12-15, IEEE Xplore Press, Lake Tahoe, NV, USA. DOI: 10.1109/WACV.2018.00007

- Ahonen, T., A. Hadid and M. Pietikainen, 2006. Face description with local binary patterns: Application to face recognition. *IEEE Transact. Pattern Analysis Machine Intelligence*, 28: 2037-2041. DOI: 10.1109/TPAMI.2006.244
- Alqahtani, F., V. Chandran and J. Banks, 2018. 3D face tracking using stereo camera. *Proceedings of the 1st International Conference on Computer Applications and Information Security*, Apr. 4-6, IEEE Xplore Press, Riyadh, Saudi Arabia. DOI: 10.1109/CAIS.2018.8441988
- Arashloo, S.R., 2015. Multiscale binarised statistical image features for symmetric face matching using multiple descriptor fusion based on class-specific LDA. *Pattern Analysis Applicat.*, 20: 113-126.
- Chrysos, G.G., E. Antonakos, S. Zafeiriou and P. Snape, 2015. Offline deformable face tracking in arbitrary videos. *Proceedings of the IEEE International Conference on Computer Vision Workshops*, Dec. 7-13, IEEE Xplore Press, Santiago, Chile, DOI: 10.1109/ICCVW.2015.126
- Chung, H.Y., C.C. Hou and S.J. Liang, 2017. Face detection and posture recognition in a real time tracking system. *Proceedings of the IEEE International Systems Engineering Symposium*, Oct. 11-13, IEEE Xplore Press, Vienna, Austria, pp: 1-6. DOI: 10.1109/SysEng.2017.8088265
- Franc, V. and J. Cech, 2017. Learning CNNs for face recognition from weakly annotated images. *Proceedings of the 12th IEEE International Conference on Automatic Face and Gesture Recognition*, May 30, IEEE Xplore Press, Washington, DC, USA. DOI: 10.1109/FG.2017.115
- He, Y., N. Sang and R. Huang, 2011. Local binary pattern histogram based texton learning for texture classification. *Proceedings of the 18th IEEE International Conference Image Processing*, Sept. 11-14, IEEE Xplore Press, Brussels, Belgium, pp: 841-844. DOI: 10.1109/ICIP.2011.6116688
- Hjelmas, E., and B.K. Low, 2001. Face detection: A survey. *Computer Vision Image Understanding*, 83: 236-274. DOI: 10.1006/CVIU.2001.0921
- Jiang, H. and E.L. Miller, 2017. Face detection with the faster R-CNN. *Computer Vision and Pattern Recognition (cs.CV)*. [arxiv.org/abs/1606.03473](https://arxiv.org/abs/1606.03473).
- Juránek, R., P. Zemčík and A. Herout, 2010. Implementing the local binary patterns with SIMD instructions of CPU. *Winter Seminar Comput. Graphics*.
- Kalal, Z., K. Mikolajczyk and J. Matas, 2010. Tracking-Learning-Detection. *Transact. Pattern Analysis Machine Intelligence*.
- Kryjak, T., M. Komorkiewicz and M. Gorgon, 2012. FPGA implementation of real-time head-shoulder detection using local binary patterns, SVM and foreground object detection. *Proceedings of the Conference on Design and Architectures for Signal and Image Processing*, Oct. 23-25, IEEE Xplore Press, Karlsruhe, Germany, pp: 1-8.
- Levi, G. and T. Hassner, 2015. Age and gender classification using convolutional neural networks. *IEEE Conference on Computer Vision and Pattern Recognition Workshops*, Jun. 7-12, IEEE Xplore Press, Boston, MA, USA. DOI: 10.1109/CVPRW.2015.7301352
- Li, H., Z. Lin, X. Shen, J. Brandt and G. Hua, 2015. A convolutional neural network cascade for face detection. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Jun. 7-12, IEEE Xplore Press, Boston, MA, USA, pp: 5325-5334. DOI: 10.1109/CVPR.2015.7299170
- Ojala, T., M. Pietikäinen and D. Harwood, 1996. A comparative study of texture measures with classification based on featured distributions. *Pattern Recognit.*, 29: 51-59. DOI: 10.1016/0031-3203(95)00067-4
- Parkhi, O.M., A. Vedaldi and A. Zisserman, 2015. Deep Face Recognition. *BMVAi*.
- Rani, P.I. and T.H.E. Prasath, 2017. Detection of human faces in video sequences using normalized gabor lbp histograms. *Int. J. Comput. Math. Sci*.
- Singh, M. and R. Sahran, 2018. A New Approach for Suspect Detection in Video Surveillance. In: *Information and Communication Technology for Sustainable Development*, Mishra, D. Nayak and M. Joshi, A. (Eds.), Springer, Singapore, ISBN-10: 978-981-10-3920-1, pp: 433-441.
- Sun, Y. D. Liang, X. Wang and X. Tang, 2015. Deepid 3: Face recognition with very deep neural networks. *Comput. Vision Pattern Recognit. (cs.CV)*. [arxiv.org/abs/1502.00873](https://arxiv.org/abs/1502.00873)
- Vasisht, S. and M. Chen, 2017. Video-Based People Tracking. *Proceedings of the IEEE 3rd International Conference on Multimedia Big Data (BigMM)*, Apr. 19-21, IEEE Xplore Press, Laguna Hills, CA, USA. DOI: 10.1109/BIGMM.2017.77
- Wolf, M., 2018. *Image and Video Analysis*. In: *Smart Camera Design*. Springer, Champ.
- Yang, M.H., D.J. Kriegman and N. Ahuja, 2002. Detecting faces in images: A survey. *IEEE Trans. PAMI*, 24: 34-58. DOI: 10.1023/A:1014520227213
- Yang, S., P. Luo, C.C. Loy and X. Tang, 2015. From facial parts responses to face detection: A deep learning approach. *Proceedings of the IEEE International Conference on Computer Vision*, Dec. 7-13, IEEE Xplore Press, Santiago, Chile, pp: 3676-3684. DOI: 10.1109/ICCV.2015.419

Yaseen, M.U., A. Anjum, O. Rana and R. Hillet, 2018. Cloud-based scalable object detection and classification in video streams. *Future Generat. Comput. Syst.*, 80: 286-298.

Zafeiriou, S., C. Zhang and Z. Zhang, 2015. A survey on face detection in the wild: Past, present and future. *Comput. Vision Image Understand.*, 138: 1-24.  
DOI: 10.1016/J.CVIU.2015.03.015

Zhao, W., R. Chellappa, P.J. Phillips and A. Rosenfeld, 2003. Face recognition: A literature survey. *Acm. Comput. Surveys*, 35: 399-458.  
DOI: 10.1145/954339.954342