Introduction

Maintaining security in an organization specifically in the workplaces like examination hall in the education sector, lockers room in banking places, server room or control room of an organization, is the dominant thing and keeping the environment free of wrongdoing and also one of them is the data-center.

A data center is basically a physical facility that organizations use to house their critical applications and data. It holds proprietary and sensitive information such as intellectual property, customer data, and financial records, and therefore needs to be both physically and digitally secure. Traditional ways are failing to tackle modern challenges of efficiency, security. There is an immense need for reform and leading technology to protect several areas from unauthorized users.

Authentication may be defined as "providing the right person with the right privileges the right access at the right time." In general, there are three approaches to authentication. In order of least secure and least convenient to most secure and most convenient, they are Something you have - card, token, key. Something you know- PIN, password. Something you are- a biometric.

The most common person authentication methods are passwords, PINs, smart keys, smart cards, but have limitations and vulnerabilities. For instance, passwords and PINs can be easily forgotten, hard to remember, or stolen. Smart keys can be easily lost or duplicated. Smart cards with magnetic strips can be forged. But a person's biometrics or biological traits cannot be stolen, forgotten, or misplaced. As a result, they provide a much more secure and reliable way to authenticate an individual when compared to the traditional methods.

Biometrics is the measurement and statistical analysis of people's unique physical and behavioral characteristics. The term *biometrics* is derived from the Greek word *bio*, meaning *life*, and *metric*, meaning *to measure*. Every human being possesses certain unique features in terms of both physical and behavioral characteristics that are different from everybody else on the planet. Based on that several types of biometric exist like Fingerprint, voice recognition, facial recognition, iris recognition, retina scan, etc. Some confusion still exists about which types of biometrics organizations should use. Based on the Organization's needs we can use one of them.

The proposed solution implies resolving complications by facial recognition. Facial recognition is an easy task for humans. Experiments have shown [1], that even one to three-day-old babies can distinguish between known faces but the same process for the computer to do the same task is difficult. How computer read the images? Basically, the computer interpreter the image as a combination of pixels. Each pixel contains a different number of channels. Pixels are organized in an ordered rectangular array. The size of an image is determined by the dimensions of this pixel array called Digital Image. When Processing the images there may be several problems that exist with images like blurred images, illumination, etc. Here we need Digital image processing techniques to process the images.

Digital image processing focuses on two major tasks–Improvement of pictorial information and Processing of image data for storage, transmission and representation for autonomous machine perception [2].

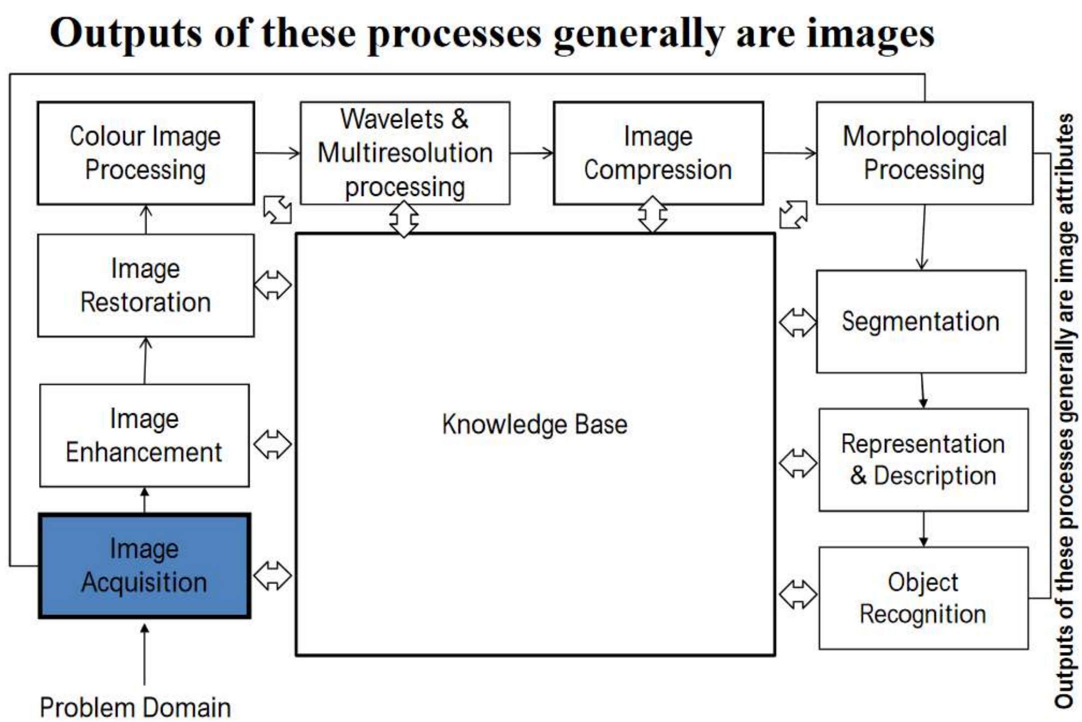


Figure -1: Fundamental steps in digital image processing [2]

Consider Figure 1 that shows the fundamental steps in digital image processing. It starts with Image Acquisition, in that the image is captured by a sensor (e.g., Camera), and digitized if the output of the camera or sensor is not already in digital form, using an analog-to-digital convertor. Image Enhancement is the process of manipulating an image so that the result is more suitable than the original. The idea behind enhancement techniques is to bring out hidden details, or simply to highlight certain features of interest in an image. Image Restoration, for Improving the appearance of an image. Color Image Processing uses the color of the image to extract features of interest in an image. Wavelets Are the foundation of representing images in various degrees of resolution. It is used for image data compression. Compression Techniques for reducing the storage required to save an image or the bandwidth required to transmit it. Morphological Processing for extracting image components that are useful in the representation and description of shape. In this step, there would be a transition from processes that output images, to processes, that output image attributes. Image Segmentation procedures partition an image into its constituent parts or objects. The more accurate the segmentation, the more likely recognition is to succeed. Representation Make a decision whether the data should be represented as a boundary or as a complete region. It almost always follows the output of a segmentation stage. The description also called, feature selection, deals with extracting attributes that result in some information of interest. Recognition is the process that assigns a label to an object based on the information provided by its description. Knowledge about a problem domain is coded into an image processing system in the form of a knowledge database [2].

System Flow

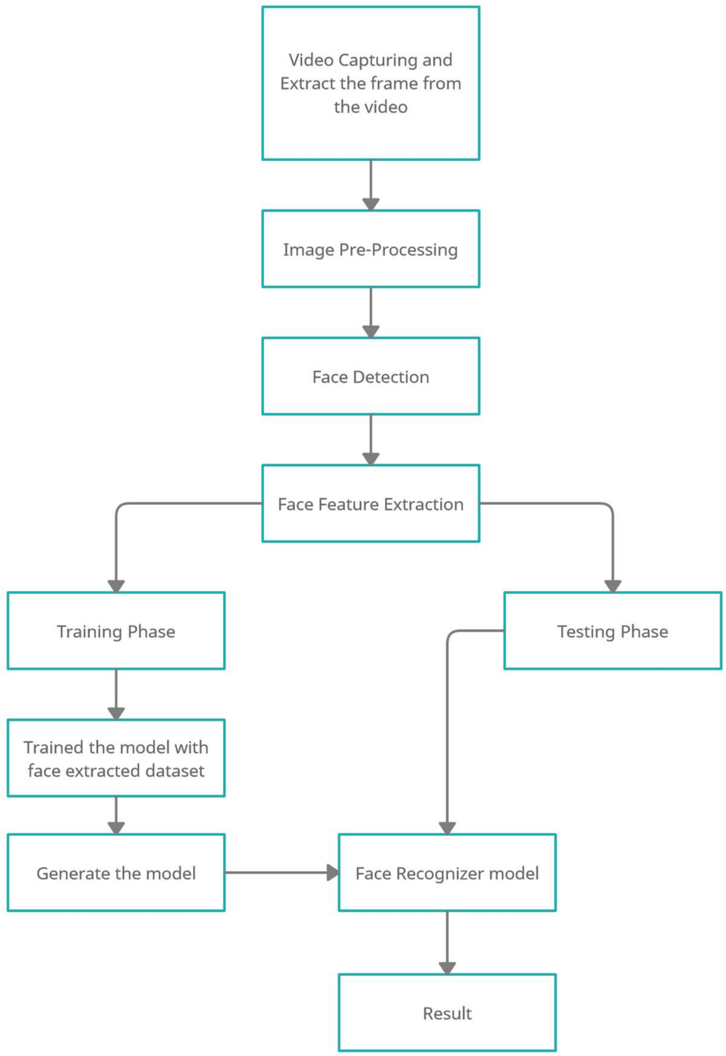


Figure-2: System Work flow of face feature extraction & face recognition

Consider the above diagram wherein the Face detection training phase, the color FERET standard dataset was used for the positive images. Color FERET dataset consists of 1208 different person images, each person having two frontal-face images with the three different image dimensions that respectively 512*768, 256*384, and 128*192. We trained our model with 1000 positive images of 512*768 dimension and 2000 negative images of the same dimension as consider for positive images. To train, the model viola-jones face detection algorithm is used.

Testing consists of video-capturing and Extract frame from the video phase wherein images capture from camera and then used in the pre-processing stage, the image converts into the RGB to greyscale, and in the last phase, the image check within the trained model, and then results are displayed that whether the face is detected or not.

Consider figure-2 wherein face detection phase output was the detection of face. Now the output of that phase is input for the face feature extraction phase. That is further divided into two tasks that are training phase and testing phase.

In training phase, the model is trained against the face dataset that was extracted by the face detection phase. Based on the literature reviews there are two major methods to train the model that is template matching and feature matching. Template matching method [3] that involves the direct comparison of pixel intensity values taken from facial images. For the Feature matching method, there are several methods available. Most popular are Eigenface method and Fisher's Face method. In the Eigenface method [4], that calculated the eigenvectors and eigenvalues of the covariance matrix of facial images and only the principal components were preserved and compared. The Fisher’s face method [5] used both principal component analysis and linear discriminant analysis to produce a subspace projection matrix

Literature Review

Face Feature Extraction A Complete Review [6]

In this paper, the researcher focuses on the general feature extraction framework for robust face recognition. They collect about 300 papers regarding face feature extraction. They believe that a general framework for face feature extraction consists of four major components: filtering, encoding, spatial pooling, and holistic representation. Then, they provide a brief review of deep learning networks, which can be seen as a hierarchical extension of the traditional framework. They provide a detailed performance comparison of various features on the LFW and FERET face database.

The filtering phase is to generate robust features for a given face image, it is beneficial to convolve the image with a specific filter, either using a pre-defined pattern or using a discriminative filter learned from the training dataset. In the Encoding phase, the output of the encoding phase could be a histogram or a feature vector. Spatial pooling can be seen as a way to further compress the coding vector according to the spatial layout of the image to form a final holistic feature. There are two classical pooling methods: average pooling that preserves the average response, and max pooling that preserve the maximum response. Block division can be seen as a part of feature pooling, as features extracted from different blocks are aggregated. This could be quite beneficial to face recognition, as the frontal structure of a face includes several distinct areas (eyes, nose, mouth, etc.). In Holistic Representation, they discourse about template matching methods and Feature Extraction methods. Eigenface method, Fisher's face method, and Principal Component Analysis methods are used for the Feature Extraction. [7-8]

After that, they discuss various filtering techniques including Gabor, LBP, Difference of Gaussians. Gabor wavelets are widely used in the image processing field in that they capture local structure corresponding to spatial frequency, spatial localization as well as orientation selectivity. In LBP, pixels of an image are labeled by thresholding the neighborhood of the pixel locally compared to the pixel itself to a binary number. Specifically, its feature vector is built by comparing the pixel with each of its neighboring pixels, and it interpolates values bilinearly at non-integer coordinates.

Feature selection using principal component analysis [9]

In this paper, researchers used principal component analysis for feature selection. PCA has been widely used in a variety of fields such as image processing, pattern recognition, data compression, data mining, machine learning, and computer vision. The principal components analysis methodology has been applied to face recognition, image denoising, and machine learning, etc.

The proposed method indeed deals with the feature selection issue from a viewpoint of numerical analysis. It exploits the eigenvector to evaluate the contribution, to the feature extraction result, of each feature component. The feature extraction result of an arbitrary sample is a linear combination of all the feature components of this sample and the entries of the eigenvector are the coefficients. That is, in the linear combination, the coefficient of i^{th} feature component of the sample is indeed the i^{th} entry of the eigenvector. As a result, if the i^{th} entry of the eigenvector has a very small absolute value, the i^{th} feature component of all the samples will statistically have a little effect on the feature extraction result.

For testing, they used 3 standard datasets ORL Dataset, Feret, and AR Datasets. Rates of classification errors of AR is 0.5047, Feret is 0.4583, ORL is 0.05. The rates of classification errors on the original samples obtained using the nearest neighbor classifier. The feature components selected by their method can obtain a lower classification error rate than the original samples. For example, while the classification error rate on the original samples of the AR database is 0.5047, scheme 1 of their method obtains a classification error rate of 0.3192 when it selects 600 feature components from all the feature components of the original samples of the AR database. we also see that scheme 1 of our method outperforms scheme 2 of our method.

Human Face Feature Extraction and Recognition Base on SIFT [10]

In this paper, researchers used SIFT (scale-invariant feature transform) method. SIFT was firstly presented by David Lowe in 2004 and is successful in video image matching. This method mainly includes four steps: extracting feature in multi-scale; locating the key-point; calculating the direction feature of the key-point and generating the descriptor of the key-point. This method has many advantages, such as scale invariance, rotation invariance, affine invariance, etc., and has strong robustness for the occlusion problem. Now, this method has been successfully used in object recognition and panorama stitching, etc. The image features extracted by SIFT have such advantages as scale invariability, rotation invariance, affine invariance, which could reduce mismatching because of occlusion, confusion, noise, and keep a high matching rate.

Firstly, extreme value is detected in the different scale of the image, and it is the key-point; secondly, the position of key-point is located by the filter of the key-points; thirdly, the gradient direction of the key-point is determined by the key-point neighborhood; fourthly, the feature descriptor is calculated by the feature of the key-point.

In this paper, the database of the test faces is the ORAL. This face DB is composed of forth people, and everyone has ten face photos that have different lightness, position, and expression. Those photos of the same people have apparent differences from each other. The size of the photo is 92 ×112. To check the performance of this algorithm, the number of training photos is from 3 to 7. The TN is the training number; the CT is the correct rate of face recognition; the ET is the misclassification rate; the RT is the refusal rate.

TN	*3*	*4*	*5*	*6*	*7*
CT(%)	92.1	94.6	96.3	96.8	97.1
ET(%)	6.3	4.5	2.4	1.8	1.3
RT(%)	0.06	0.9	1.3	1.4	1.6

Figure -3: The Result of Face Recognition

Face Recognition: Features versus Templates [3]

This paper focuses on two traditional classes of techniques applied to the recognition of digital images of frontal views of faces under roughly constant illumination. The first technique is based on the computation of a set of geometrical features from the picture of a face. This was the first approach toward an automated recognition of faces. The second class of techniques is based on template matching. The paper has focused on a comparative analysis of these two different approaches to face recognition.

The two approaches are Geometric, Feature-Based Matching and Template matching Geometric, Feature-Based Matching: A face can be recognized even when the details of the individual features (such as eyes, nose, and mouth) are no longer resolved. The remaining information is, in a sense, purely geometrical and represents what is left at a very coarse resolution. The idea is to extract relative position and other parameters of distinctive features such as eyes, mouth, nose, and chin. Goldstein and Kaya showed that a computer program provided with face features extracted manually could perform recognition with apparently satisfactory performance. Template Matching: In the simplest version of template matching, the image, which is represented as a bidimensional array of intensity values, is compared using a suitable metric (typically the euclidean distance) with a single template representing the whole face. There are, of course, several, more sophisticated ways of performing template matching. For instance, the array of grey levels may be suitably preprocessed before matching. Several full templates per each face may be used to account for the recognition from different viewpoints. Still another important variation is to use, even for a single viewpoint, multiple templates.

The results obtained by using this approach (about 90% correct recognition using geometrical features and perfect recognition using template matching) favors the implementation of this template-matching approach.

Facial feature extraction for face recognition: a review [11]

This paper provides an up-to-date review of major human facia recognition research. The overall process which is conveyed in the paper focuses on face detection, feature extraction, and face recognition. The importance of facial features for face recognition cannot be overstated. Many face recognition systems need facial features in addition to the holistic face, as suggested by studies in psychology. Several Techniques of facial feature extraction are covered in this paper are Color segmentation techniques and Appearance-based approaches.

The Color segmentation technique makes use of skin color to isolate the face. Any non-skin color region within the face is viewed as a candidate for eyes and/or mouth. The performance of such techniques on facial image databases is rather limited, due to the diversity of ethnic backgrounds.

The concept of "feature" in these approaches differs from simple facial features such as eyes and mouth. Any extracted characteristic from the image is referred to as a feature. Methods such as principal component analysis (PCA), independent component analysis, and Gabor-wavelets [45] are used to extract the feature vector. These approaches are commonly used for face recognition rather than personal identification.

In this paper, they have reviewed the works done in facial feature extraction, concerning resource discovery. They have also focused on and evaluated different techniques and ideas, used to reduce bandwidth consumption during this process.

Face Feature Extraction Methodology

LBPH (Local Binary Pattern Histogram) Algorithm

Local Binary Pattern (LBP) is a simple yet very efficient texture operator which labels the pixels of an image by thresholding the neighborhood of each pixel and considers the result as a binary number. It was first described in 1994 (LBP) and has since been found to be a powerful feature for texture classification. It has further been determined that when LBP is combined with histograms of oriented gradients (HOG) descriptor. Using the LBP combined with histograms the images can be represent as with a simple data vector. The basic idea for developing the LBP operator was that two-dimensional surface textures can be described by two complementary measures: local spatial patterns and gray scale contrast. The original LBP operator (Ojala et al. 1996) forms labels for the image pixels by thresholding the 3 x 3 neighborhood of each pixel with the center value and considering the result as a binary number. The histogram of these $2^8 = 256$ different labels can then be used as a texture descriptor.

The LBPH uses 4 parameters: **Radius**: the radius is used to build the circular local binary pattern and represents the radius around the central pixel. It is usually set to 1. **Neighbors**: the number of sample points to build the circular local binary pattern. Keep in mind: the more sample points you include, the higher the computational cost. It is usually set to 8. **Grid X**: the number of cells in the horizontal direction. The more cells, the finer the grid, the higher the dimensionality of the resulting feature vector. It is usually set to 8. **Grid Y**: the number of cells in the vertical direction. The more cells, the finer the grid, the higher the dimensionality of the resulting feature vector. It is usually set to 8.

The first computational step of the LBPH is to create an intermediate image that describes the original image in a better way, by highlighting the facial characteristics. To do so, the algorithm uses a concept of a sliding window, based on the parameter's **radius** and **neighbors**.

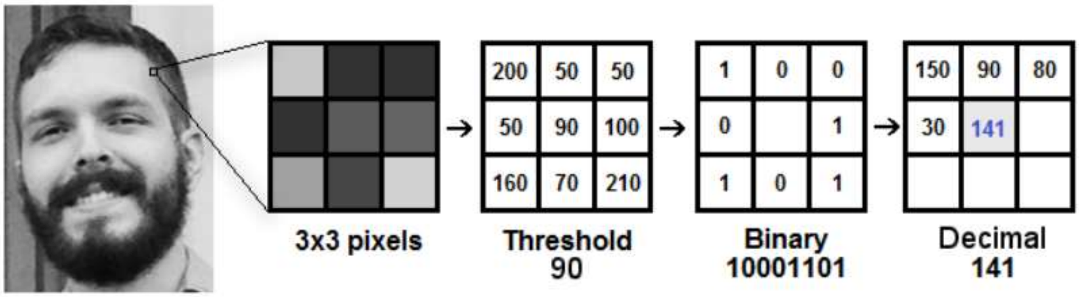


Figure - 4: LBP Calculation

Here the image is facial image in grayscale. The part of the image is obtained as a window of 3x3 pixels. It can also be represented as a 3x3 matrix containing the intensity of each pixel (0~255). Then, take the central value of the matrix to be used as the threshold. This value will be used to define the new values from the 8 neighbors. For each neighbor of the central value (threshold), we set a new binary value. In the algorithm 1 for values equal or higher than the threshold and 0 for values lower than the threshold. Now, the matrix will contain only binary values (ignoring the central value). The image is concatenate each binary value from each position from the matrix line by line into a new binary value (e.g., 10001101). Then, it converts into the binary value to a decimal value and set it to the central value of the matrix, which is actually a pixel from the original image. At the end of this procedure (LBP procedure), a new image which represents better the characteristics of the original image.

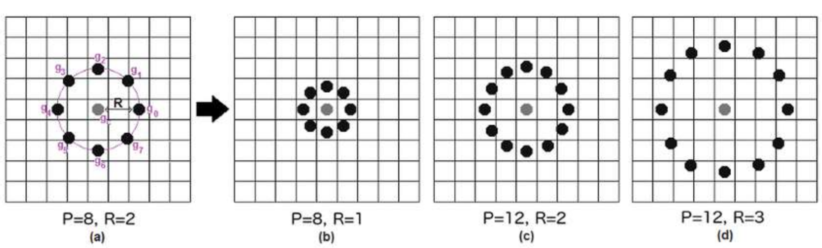


Figure -5: Circular LBP

It can be done by using **bilinear interpolation**. If some data point is between the pixels, it uses the values from the 4 nearest pixels (2x2) to estimate the value of the new data point. using the image generated in the last step, we can use the **Grid X** and **Grid Y** parameters to divide the image into multiple grids.

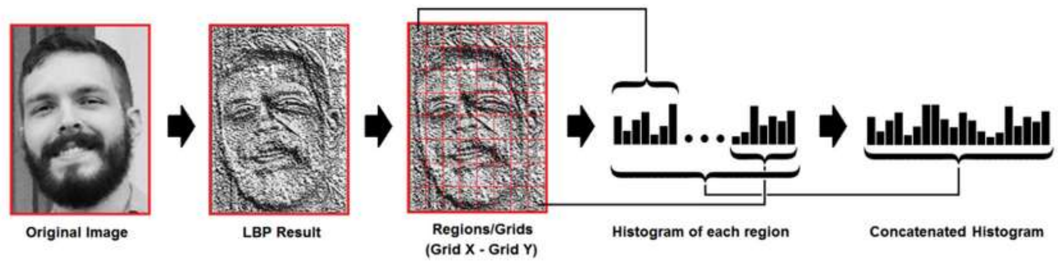


Figure -6: Calculating the Histogram from the Gradients

Algorithmic Description of LBPH method

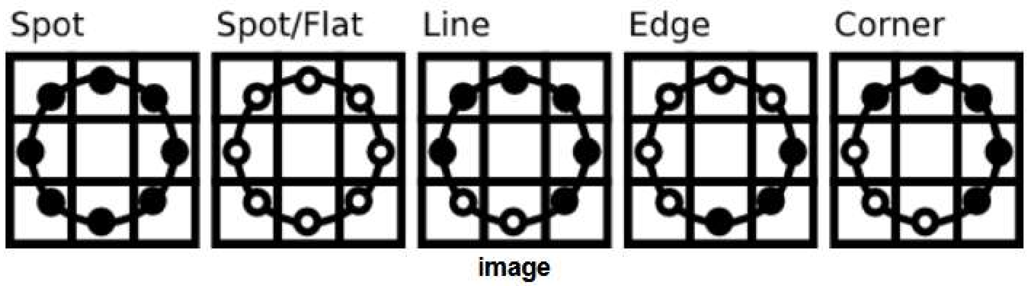
A more formal description of the LBP operator can be given as:

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

, with (*xc*, *yc*) as central pixel with intensity *ic*; and *in* being the intensity of the neighbor pixel. *s* is the sign function defined as:

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

This description enables you to capture very fine-grained details in images. In fact, the authors were able to compete with state-of-the-art results for texture classification. Soon after the operator was published it was noted, that a fixed neighborhood fails to encode details differing in scale. So, the operator was extended to use a variable neighborhood in. The idea is to align an arbitrary number of neighbors on a circle with a variable radius, which enables to capture the following neighborhoods:



For a given Point (*xc*, *yc*) the position of the neighbor (*xp*, *yp*), *p*∈*P* can be calculated by:

$$x_p = \quad x_c + R\cos(\frac{2\pi p}{P})$$
$$y_p = \quad y_c - R\sin(\frac{2\pi p}{P})$$

Where *R* is the radius of the circle and *P* is the number of sample points.

The operator is an extension to the original LBP codes, so it's sometimes called *Extended LBP* (also referred to as *Circular LBP*). If points coordinate on the circle doesn't correspond to image coordinates, the point gets interpolated. Computer science has a bunch of clever interpolation schemes, the OpenCV implementation does a bilinear interpolation:

$$f(x,y) \approx \begin{bmatrix} 1-x & x \end{bmatrix} \begin{bmatrix} f(0,0) & f(0,1) \\ f(1,0) & f(1,1) \end{bmatrix} \begin{bmatrix} 1-y \\ y \end{bmatrix}.$$

Implementation Environment and results

With the help of the python modules namely OpenCV, NumPy, tkinter we have implemented the methodology in Microsoft windows 10 environment. The capturing and verifying of the images in this implementation are done with 4megapixel camera available within the laptop.

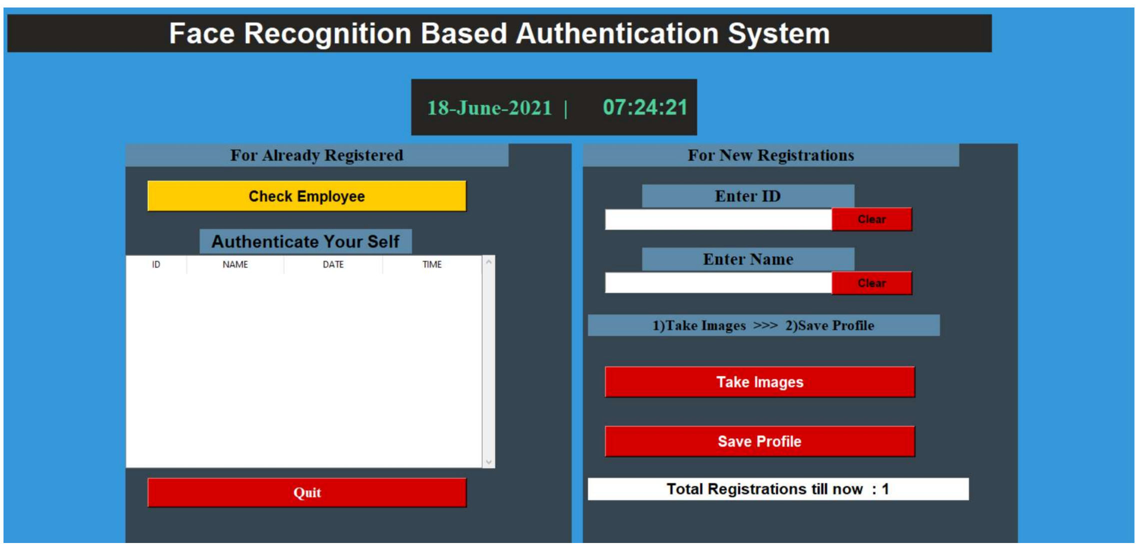


Figure -7: User-interface of the System

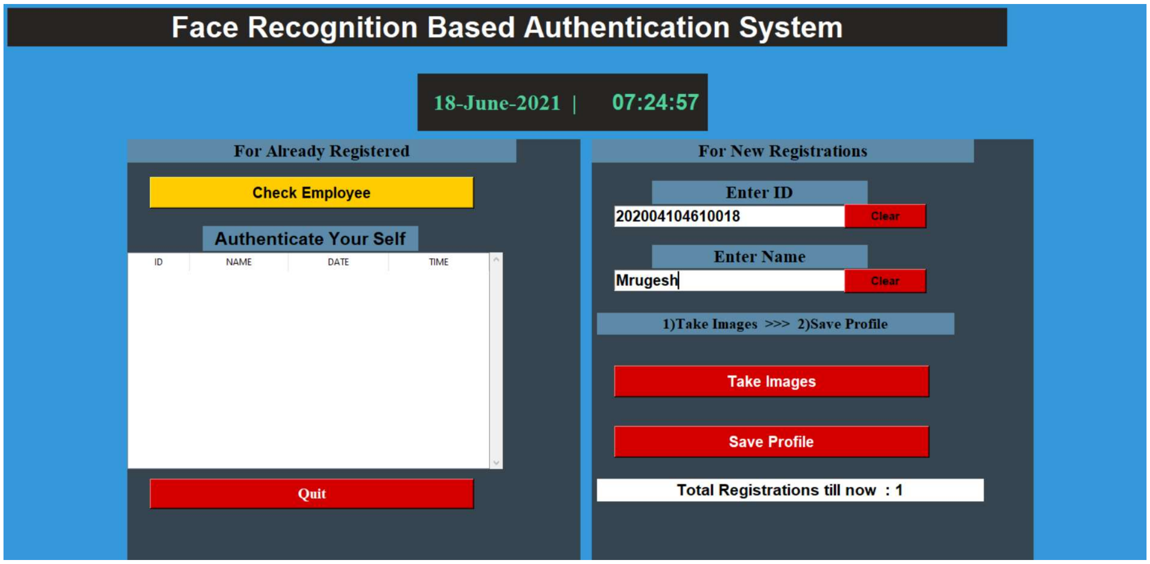


Figure -8: Registration of the new user

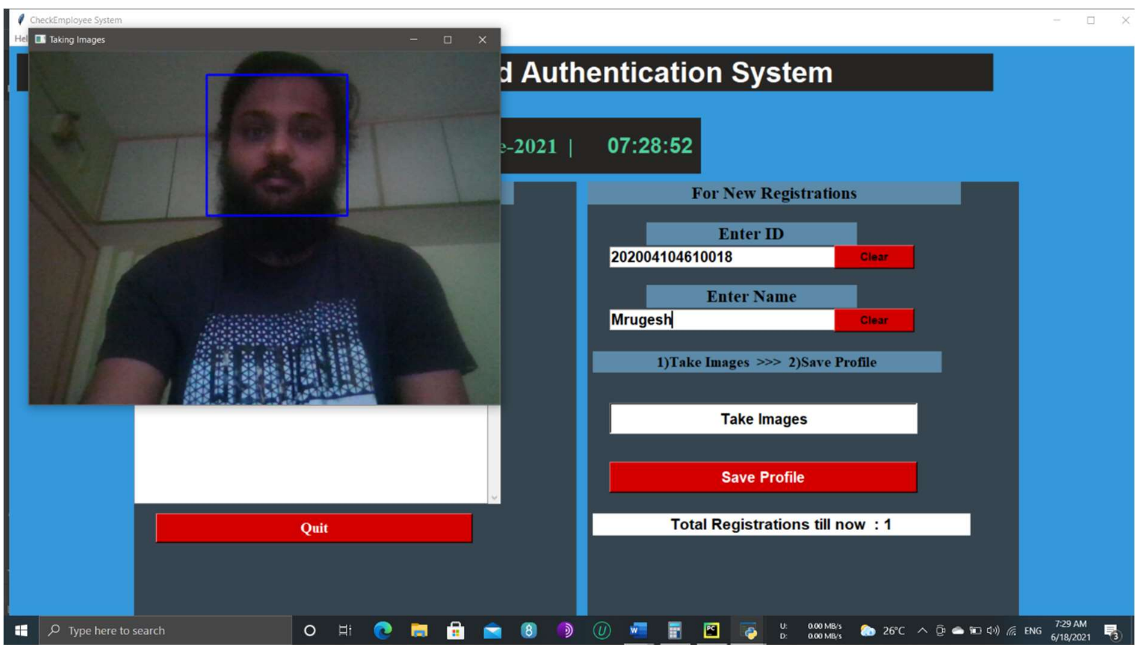


Figure - 9: Capture the image of the user for trained the model.

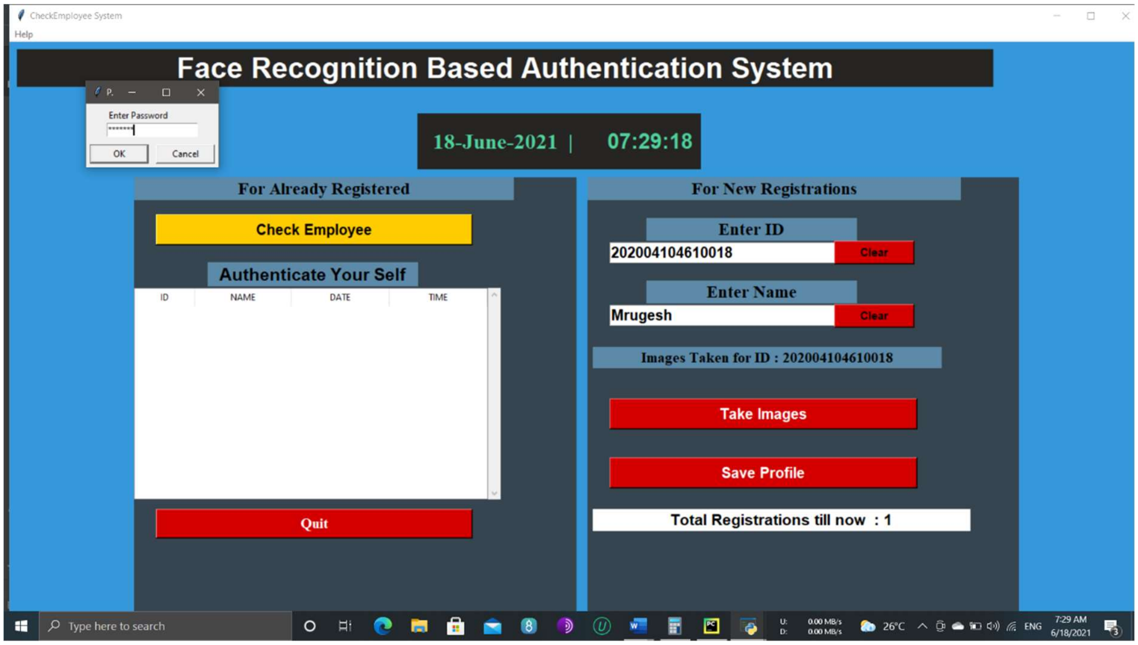


Figure - 10: Save the user information and update the model.

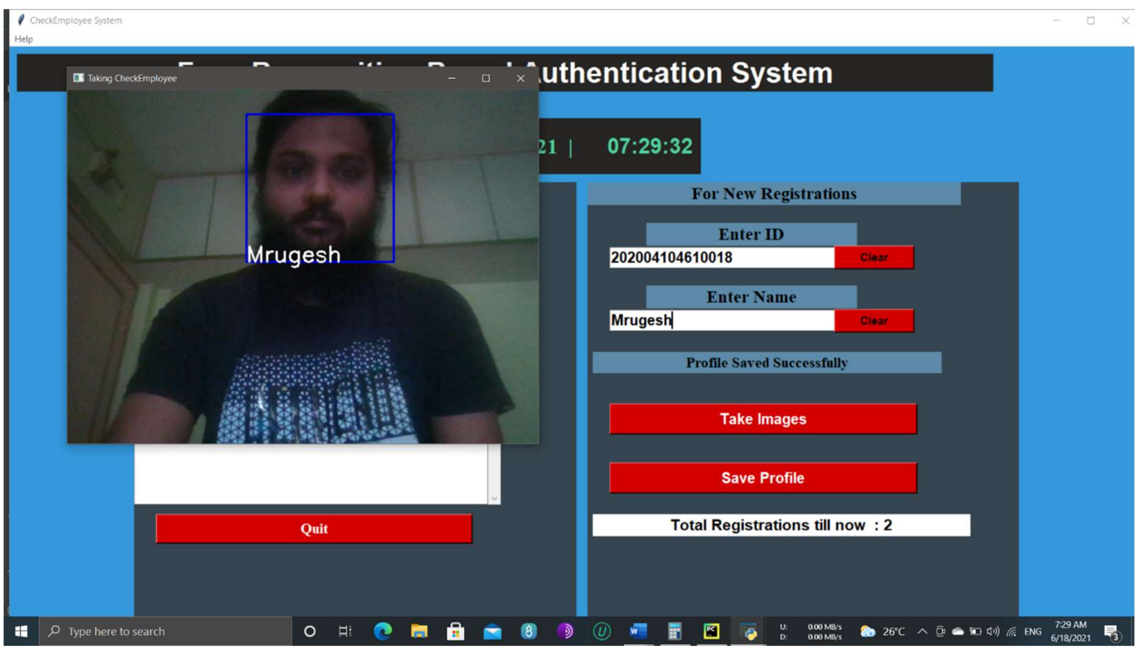


Figure - 11: Check the already register user with frontal face

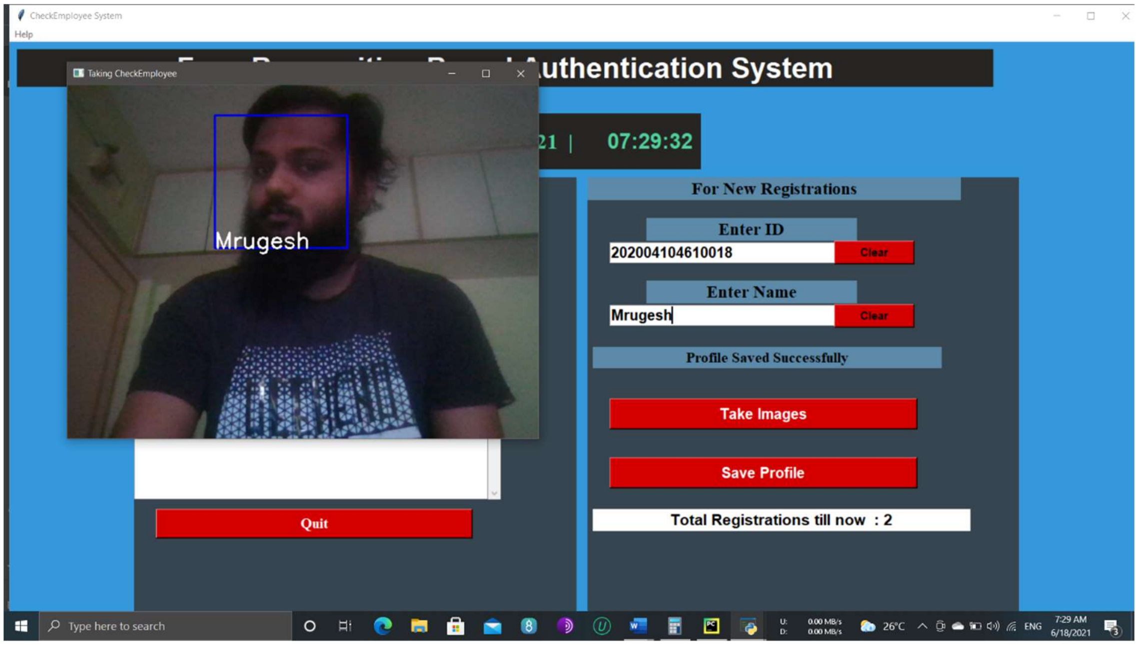


Figure - 12: Check the already register user with 15-degree right side front face.

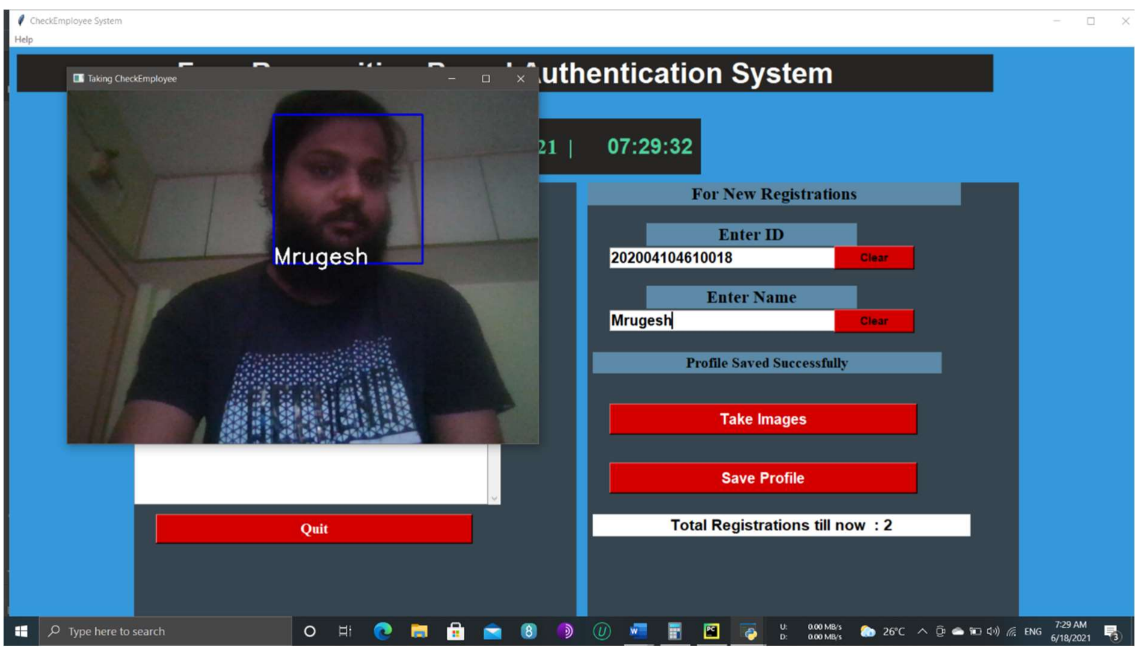


Figure - 13: Check the already register user with 15-degree left side front face.

Experimental Result

The algorithms were tested on PyCharm-version 2020.3.3 with OpenCV library. This testing running on Windows 10 64bit with Intel Core i7 8^{th} Generation, 16GB LPDDR3 RAM and 256 SSD. The settings for the algorithm testing are using 1.1 scale factor with four minimum neighbors' threshold with 24x24 minimum detection scale. Color FERET dataset which is widely used to evaluate the performance of face detection algorithm [11-14]. This set consists of 20000 images with 1004 frontal face images. For training the face detector, all collected 1004 face sample are used. The non-face sub- windows used to train the detector come from 3019 images.

Dataset

Figure - 13: Color FERET dataset

Result Observation

Image	No of Person	Total No of Images	Detected Face	Success Rate
Front Face	Single	995	995	100%
15-degree right side	Single	200	200	100%
15 degree left side	Single	200	200	100%
30-degree right side	Single	200	170	85%
30-degree left side	Single	200	170	85%
45-degree right side	Single	200	10	5%
45 degree left side	Single	200	10	5%
90 degree left side	Single	200	-	0
90-degree right side	Single	200	-	0

The conclusion from the table is for the frontal view of face total 995 images are tested out of which the system has detected all of them. While for the 15-degree left and right results are same and for the image having face rotated 30 degrees right and left the efficiency is approximately 85%. For the 45 degrees both right and left the results are listed out in table and while testing 90 degrees right and left total 200-200 images have been tested and results are specified in table.

References

1. Chiara Turati, Viola Macchi Cassia, Francesca Simion, and Irene Leo. Newborns' face recognition: Role of inner and outer facial features. Child development, 77(2):297–311, 2006.
2. Gonzalez, Rafael C, and Richard E. Woods. Digital Image Processing. , 2002. Print.
3. R. Brunelli and T. Poggio, "Face recognition: features versus templates," in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 15, no. 10, pp. 1042-1052, Oct. 1993, doi: 10.1109/34.254061.
4. L. Sirovich and M. Kirby, ''Low-dimensional procedure for the characterization of human faces,'' J. Opt. Soc. Amer. A, Opt. Image Sci. Vis., vol. 4, no. 3, pp. 519–524, 1987.
5. P. N. Belhumeur, J. P. Hespanha, and D. Kriegman, ''Eigenfaces vs. Fisherfaces: Recognition using class specific linear projection,'' IEEE Trans. Pattern Anal. Mach. Intell., vol. 19, no. 7, pp. 711–720, Jul. 1997
6. H. Wang, J. Hu and W. Deng, "Face Feature Extraction: A Complete Review," in IEEE Access, vol. 6, pp. 6001-6039, 2018, doi: 10.1109/ACCESS.2017.2784842.
7. L. Best-Rowden, H. Han, C. Otto, B. F. Klare, and A. K. Jain,''Unconstrained face recognition: Identifying a person of interest froma media collection,''IEEE Trans. Inf. Forensics Security, vol. 9, no. 12,pp. 2144–2157, Dec. 2014.
8. Y. Huang, Z. Wu, L. Wang, and T. Tan, ''Feature coding in image classifi-cation: A comprehensive study,''IEEE Trans. Pattern Anal. Mach. Intell.,vol. 36, no. 3, pp. 493–506, Mar. 2014.
9. F. Song, Z. Guo and D. Mei, "Feature Selection Using Principal Component Analysis," 2010 International Conference on System Science, Engineering Design and Manufacturing Informatization, 2010, pp. 27-30, doi: 10.1109/ICSEM.2010.14.
10. H. Yanbin, Y. Jianqin and L. Jinping, "Human Face Feature Extraction and Recognition Base on SIFT," 2008 International Symposium on Computer Science and Computational Technology, 2008, pp. 719-722, doi: 10.1109/ISCSCT.2008.249.
11. E. Bagherian and R. W. O. K. Rahmat, "Facial feature extraction for face recognition: a review," 2008 International Symposium on Information Technology, 2008, pp. 1-9, doi: 10.1109/ITSIM.2008.4631649.
12. https://docs.opencv.org/2.4/modules/contrib/doc/facerec/facerec_tutorial.html