



---

## **IMPLEMENTATION OF FACE DETECTION AND RECOGNITION USING OPEN-CV WITH DEEP LEARNING TECHNIQUE**

**Mr. Wagh Shrikant A.<sup>1</sup>, Prof. Dhakane V. N.<sup>2</sup>**

<sup>1</sup>PG Student Department of CE, S.N.D.COE and RC, Yeola

<sup>2</sup>Professor Department of CE, S.N.D.COE and RC, Yeola

Email: [shri.snjb@gmail.com](mailto:shri.snjb@gmail.com), [vikasdhakane@gmail.com](mailto:vikasdhakane@gmail.com)

---

### **ABSTRACT**

This proposed system gives an ideal way of detecting and recognizing human face using Open-CV, and python which is part of deep learning. This report contains the ways in which deep learning an important part of computer science field can be used to determine the face using several libraries in Open- CV along with python. This report will contain a proposed system which will help in the detecting the human face in real time. This implementation can be used at various platforms in machines and smartphones, and several software applications. Face Recognition is the most popular and trending technology in the present era. It is an effective way to provide vision to a machine for better interaction with humans. The way of living will be reflected if machines can read our faces. The face recognition system will move the world in a new dimension. It will be beneficial in many ways to find the identity and security. In this paper, a face recognition system is proposed for advanced applications such as access and security, payments, criminal identifications etc. The process of identification will be based on face recognition which is further divided into three steps: detection of face, extractions of the features and classification, and real time recognition. Detection of face is recognized as the essential step of our system. It is used to extract a face in a frame, which is based on the Viola- Jones object detection algorithm that uses AdaBoost classifier with Haar and LBP features. Local Binary Patterns (LBP) is utilized to extract the unique features of the face like eyes, nose, and mouth in the feature extraction phase. The facial image is correlated with the images available in the database for the classification. The system is implemented in Python using Open-CV library.

We are also using Histogram of Oriented Gradients (HOG) is a feature descriptor used in computer vision and image processing applications for the purpose of the object detection. It is a technique that counts events of gradient orientation in a specific portion of an image or region of interest.

**Keywords** - Python, Open-CV, Deep Learning, Face detection, HOG, Local Binary Patterns (LBP) etc.

---

### **1. INTRODUCTION**

Face recognition is the technique in which the identity of a human being can be identified using ones individual face. Such kind of proposed system gives an ideal way of detecting and recognizing human face using Open-CV, and python which is part of deep learning. This report contains the ways in which deep learning an important part of computer science field can be used to determine the face using several libraries in Open- CV along with python. This report will contain a proposed system which will help in the detecting the human face in real time. This implementation can be used at various platforms in machines and smartphones, and several software applications. Face Recognition is the most popular and trending technology in the present era. It is an effective way to provide vision to a machine for better interaction with humans. The way of living will be reflected if machines can read our faces. The face recognition system will move the world in a new dimension. It will be beneficial in many ways to find the identity and security. In this paper, a face recognition system is proposed for advanced applications such as access and security, payments, criminal identifications etc. The process of identification will be based on face recognition which is further divided into three steps: detection of face, extractions of the features and classification, and real time recognition. Detection of face is recognized as the essential step of our system. It is used to extract a face in a frame, which is based on the Viola- Jones object detection algorithm that uses AdaBoost classifier with Haar and LBP features. Local Binary Patterns (LBP) is utilized to extract the unique features of the face like eyes, nose, and mouth in the feature extraction phase. The facial image is correlated with the images available in the database for the classification. The system is implemented in Python using Open-CV library. We are also using Histogram of Oriented Gradients (HOG) is a feature descriptor used in computer vision and image processing applications for the purpose of the object detection. It is a technique that counts events of gradient orientation in a specific portion of an image or region of interest.

Systems can be used in photos, videos, or in real time machines. The objective of this article is to provide a simpler and easy method in machine technology. With the help of such a technology one can easily detect the face by the help of dataset in similar matching appearance of a person. The method in which with the help of python and Open-CV in deep learning is the most efficient way to detect the face of the person. This method is useful in many fields such as the military, for security, schools, colleges and universities, airlines, banking, online web applications, gaming etc. this system uses powerful python algorithm through which the detection and recognition of face is very easy and efficient. The most useful area in which face recognition is important is the biometric that is used for authentication process which makes the work easier. Face recognition is one of the widely used

technologies or systems in which it has the potential to perform tasks such as to have records provided in by the dataset in many areas such as the school and colleges attendance systems, it can also be helpful in catching the thieves or the terrorist, can be helpful in the security of common people and the much needed security areas in the country. Face recognition can be used by the government to verify the voters list, find missing persons, find the population or census, immigration process, also provide security over internet scams protection commerce and highly used in the medicine and healthcare range. This brings in a very high demand or a real time face recognition system for several uses for the people and government. Providing such excellent systems there would be ease in several activities.

---

## 2. REVIEW OF LITERATURE

1. Malaya Khan, Sudeshna, Rani Astya, Face Detection and Recognition Using OpenCV, 2019 International Conference on Computing, Communication, and Intelligent Systems (ICCCIS)-IEEE\_2019. This proposes the PCA (Principal Component Analysis) facial recognition system. The key component analysis (PCA) is a statistical method under the broad heading of factor analysis. The aim of the PCA is to reduce the large amount of data storage to the size of the feature space that is required to represent the data economically.
2. Vijay Kumar Sharma, Designing of Face Recognition System, School of Electronics & Communication Engineering, Shri Mata Vaishno Devi University, Katra, India, 182320, 978-1-5386-8113-8/19/\$31.00 ©2019 IEEE., The process of identification will be based on face recognition which is further divided into three steps: detection of face, extractions of the features and classification, and real time recognition. Detection of face is recognized as the essential step of our system.
3. Tejashree Dhawle<sup>1</sup>, Urvashi Ukey<sup>2</sup>, Rakshandha Choudante<sup>3</sup>, Face Detection and Recognition using OpenCV and Python, International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056 Volume: 07 Issue: 10 | Oct 2020., Face recognition systems are currently associated with many top technological companies and industries making the work of face recognition easier. The use of python programming and Open-CV makes it an easier and handy tool or system which can be made by anyone according to their requirement. The proposed system discussed in this project will be helpful for many as it is user friendly and cost\_ efficient system. Hence by the use of python and Open- CV the face recognition system can be designed for various Purposes.
4. J. Manikandan, S. Lakshmi Prathyusha, P. Sai Kumar, Y. Jaya Chandra, M. Umaditya Hanuman, "Face Detection and Recognition using Open CV Based on Fisher Faces Algorithm", International Journal of Recent Technology and Engineering (IJRTE) ISSN: 2277-3878, Volume-8 Issue-5, January 2020. , It's a method of characteristic someone supported facial expression. This method is enforced in 2 stages. They're the training stage and therefore the testing stage. This study primarily consists of 3 elements, specifically face detection from the image, feature extraction and storing many reminder images, and recognition. Face finding rule is employed to detect the face from the given image.
5. Face Detection in Real Time Based on HOG. N. J. Wang, S. C. Chang and P. J. Chou. Taipei, Taiwan: IEEE, DOI:10.1109/ISPACS.2012.6473506, 2012. International Symposium on Intelligent Signal Processing and Communications Systems. pp. 333-337. ISBN: 978-1-4673- 5081-5., In the field of face recognition and establishment of face database, face detection is a crucial step. Most of the face detection proposed as now are focused on software algorithms to b improve the detection rate and decrease the false alarm. However, the more complex algorithm, the more computation time is required. It hinders our real-time applications. In this paper, we proposed a real-time multi-face detection system based on hardware design to enhance the processing time.

---

## 3. SYSTEM ARCHITECTURE

### 3.1 System Overview

To create a complete project on Face Recognition, we must work on 3 very distinct phases:

- Face Detection and Data Gathering
- Train the Recognizer
- Face Recognition

The below block diagram resumes those phases:

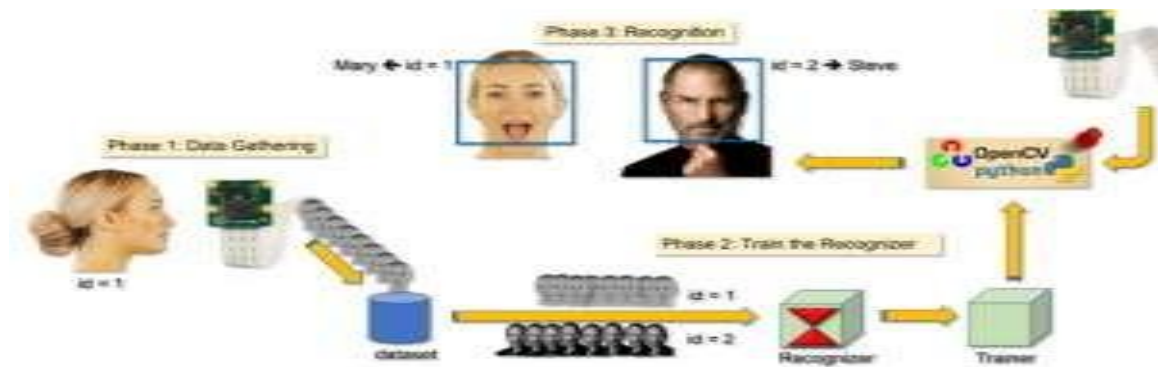


Figure 1: System Overview

In order to create this system first we will have to make the datasets. When the image quality becomes favorable different procedures will take place in the face recognition system the tasks are performed using the python queries "python encode\_faces.py". The input will be taken from the dataset which will be received in the "encodings.py". There will be precision formatting in the system wherein face embedding for each face will occur. Secondly a file "recognize\_faces\_images.py" will contain all the required methods and the techniques for the process of identification of the face of the person from the given image of the dataset.

The given file will be executed by the python command "python recognize\_faces\_image.py-encodings". We can resize or turn the image for proximity with the goal for getting the desired output. The present classifier along with Open-CV libraries will enhance the outcome or results in the face recognition system.

In this paper, the process of identification based on face recognition is split into major three stages: detection of face, data set generation and trainer, face recognition.

### 3.1.1. Face Detection

It is the first and very important stage of our system. Face detection is a method to extract the desired faces from a frame. It used the Viola-Jones object detection algorithm [2] which uses AdaBoost classifier with Haar and LBP features. There are some similarities in human faces. These predict-abilities can be matched using Haar Features. Haar like features are defined as the difference in the acquired intensity of the two rectangular regions. The most common features of human faces are the eye and nose area and depending on the darkness and brightness of the same and compared with the cheeks. Haar-Like features are evaluated. It is the collection of numbers of the face images taken from the different sources. It provides the numbers of features and their utilization by using the Adaboost algorithm. There are various types of the methods available to classify the detection of the face like color technique, Adaboost face detector, feature invariant technique etc.

### 3.1.2 Face Recognition:

The process of face recognition is different from the detection of the face. In the process of detection of face, the system recognizes the face with an image and locates its position where as in face recognition the system recognizes specific people. Face recognition basically compares the input face within an image and visually search with all facial images in data set, find and identify specific faces within large image collections.

Numbers of face recognition algorithms are available to recognize the face like Eigen faces, Fisher faces, LBP Histogram etc. are available where all these techniques have different type of approach to extract the image. In this system, LBPH method is used to perform face recognition. The Eigen faces and Fisher faces determine the most dominant features of the face in the training set and more time-consuming methods as compared to the LBPH, however LBPH figure out all the faces in training set individually. LBP is a texture recognition algorithm that is also used to represent faces [3, 10], which is one the easiest face recognition algorithms. Due to its computational simplicity and robustness to monotonic gray scale transformations.

LBP is used to extract the exclusive structures of the face like eyes, nose, and mouth in the feature extraction phase. LBP is associated with the histogram of oriented gradients (HOG), to improve the performance on same data sets [5, 11]. It is available in the Open-CV library, which are implementing in Python. Face recognition technology has many advantages such as (hygienic) just look no touching, convenient to use, it doesn't require the use of keys or RFID cards or passwords, (highly usability) everyone can use this technology as people's faces are stable it won't affect by weather like dry, wet or work conditions, accurate, secure etc.

### 3.1.3 Data Set and Trainer

In this process, face dataset of the user is created, in which 41 images of each user are taken and the attributes used are user ID and username. Which are trained into the dataset or database. The input data is synthesized and checked the available data in datasets [12]. If there are any similarities in them or they are matched to the ones in dataset, then they are recognized successfully.

## 3.2 SYSTEM WORKING FLOW MODULES

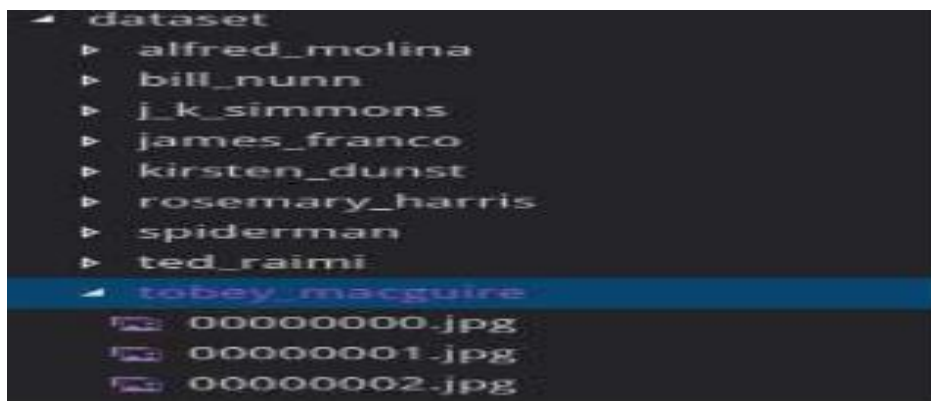
**Face Detection:** Look at the picture and find a face in it. **Data Gathering:** Extract unique characteristics of anybody's face that it can use to differentiate him from another person, like eyes, mouth, nose, etc. **Data Comparison:** Despite variations in light or expression, it will compare those unique features to all the features of all the people you know.

**Face Recognition:** It will determine "Hey, that's my boy Kirill!"

### 1.3 PROCESS

#### 1.3.1 Dataset

The dataset has the following structure, containing actor images collected using the Bing Image API (getData.py).



**Figure 2: Dataset Structure**

As I'm using Spiderman 2, I've collected several pictures of the actors in the movie. Per actor I have ~15 images. More will do better, but this number seems to work fine.

### 3.3.2 TRAINING

For every image in the dataset, we first get a square enclosing the face in the image, then generate a 128d vector for that face, which is dumped to the 'encodings. Pickle' file. We can either use CNN (slower, more accurate) or HOG (faster, less accurate) for the face detection process. Here I've used the face recognition library, which gives me both the options.

### 3.3.3 FACE RECOGNITION

Consider an image, be it a still from the movie, or a frame of a video clip. First, we identify the faces in the image using the same method as above (CNN or HOG), generate an encoding for it(128d vector), and then compare it with our collected encodings. The actors with the most matched encodings are the actor in the image.

---

## 4. METHODOLOGY & ALGORITHM

### 4.1 Methodology

The concept of OpenCV was put forth by Gary Bradski which had the ability to perform on multi-level framework. OpenCV has a number of significant abilities as well as utilities which appears from the outset. The OpenCV helps in recognizing the frontal face of the person and also creates XML documents for several areas such as the parts of the body.

Deep learning evolved lately in the process of the recognition systems. Hence deep learning along with the face recognition together work as the deep metric learning systems. In short deep learning in face detection and recognition will broadly work on two areas the first one being accepting the solidary input image or any other relevant picture and the second being giving the best outputs or the results of the image of the picture. We would be using dlib facial recognition framework that would be the easy way to organize the face evaluation. The two main significant libraries used in the system are dlib and face recognition.

Python being a very powerful programming languages and one of the programming languages that are being used all over the world has proven to give best results in the face recognition and detection systems. Together face recognition and detection becomes very easy and fruitful with the help of the python programming language and Open-CV.

The project was made in the following way

### 1. Making the face detection module:

- In order to setup the environment variables we had to link the directories and paths which were been accessed during the execution of the program so that the libraries we call later can work together smoothly.
- Next, we integrate Open-CV with python
- We rebuilt it to include face sub module. We used a rebuilt version of Open-CV with face sub module because it included the CNN & LBPH classifier which was to be used for face detection.
- We install the latest version of NumPy and include it
- Finally we get the classifier ready.

### 2. Facial Recognition and Classification

- We start off by capturing the video from the web cam of our computer
- The algorithm to be used and the data structures to be implemented were decided. The codes for detection were rewritten.
- Identify the faces in it. After this process face detection is complete
- Make a training data set out of the captured faces
- Next we use CNN Classifier and LBPH classifier to produce a yml file
- Lastly the recognition of faces is performed.

## 4.2 Following two important methodologies (model) used to implement this system

### 4.2.1 THE HISTOGRAM OF ORIENTED GRADIENTS (HOG):

The Histogram of Oriented Gradients (HOG) is a feature descriptor used in computer vision and image processing applications for the purpose of the object detection. It is a technique that counts events of gradient orientation in a specific portion of an image or region of interest. In 2005, Dalal and Triggs published a research paper named Histograms of Oriented Gradients for Human Detection. After the release of this paper, HOG is used in a lot of object detection applications.

**Below are the essential steps we take on HOG feature extraction:**

#### Resizing the Image:

As mentioned previously, if you have a wide image, then crop the image to the specific part in which you want to apply HOG feature extraction, and then resize it to the appropriate shape.

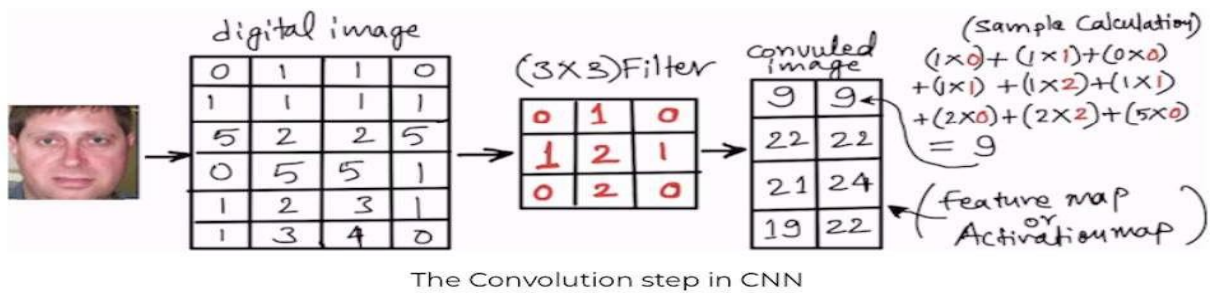
#### Calculating Gradients:

Now after resizing, we need to calculate the gradient in the x and y direction. The gradient is simply the small changes in the x and y directions, we need to convolve two simple filters on the image.

**The hog () function takes 6 parameters as input:**

- **Image:** The target image you want to apply HOG feature extraction.
- **Orientations:** Number of bins in the histogram we want to create, the original research paper used 9 bins so we will pass 9 as orientations.
- **Pixels\_per\_cell:** Determines the size of the cell, as we mentioned earlier, it is 8x8.
- **Cells\_per\_block:** Number of cells per block, will be 2x2 as mentioned previously.
- **Visualize:** A Boolean whether to return the image of the HOG, we set it to True so we can show the image.
- **Multichannel:** We set it to True to tell the function that the last dimension is considered as a color channel, instead of spatial.

### 4.2.2 CNN CLASSIFIER (Face Recognition using DL And CNN in Python)



The Convolution step in CNN

Figure 3: CNN Steps

- Convolutional Neural Networks (CNN) changed the way we used to learn images. It made it very very easy! CNN mimics the way humans see images, by focusing on one portion of the image at a time and scanning the whole image.
- CNN boils down every image as a vector of numbers, which can be learned by the fully connected dense layers of ANN. More information about CNN can be found here.

#### Data Collection and Preparation:

This research paper uses a two pedestrian videos, the first is obtained online from YouTube4 which is provided by BriefCam5 and the other video footage of the Oxford Town Centre. The first dataset contains a CCTV footage of pedestrians in an area walking on the pavement and the other dataset contains a video of people walking in a busy downtown centre in Oxford, England. The Oxford Town Centre dataset has been utilized multiple times in multiple research projects. These are open datasets and can be used for various developments and research projects in the area of object detection, facial recognition, and many such other projects.

This dataset is a very unique dataset, in the sense that it uses video footage straight out of a public CCTV camera that on the contrary was assigned for public safety reasons. In this video, it shows that the pedestrians are walking or acting in a normal and unrehearsed manner. These pedestrians are just normal people walking on the road or pathway minding their own business.

Although these people do not know about the research projects, they were aware of being under supervision cameras and it is with their consent that this dataset has been created hence not violating any ethical issues.

#### Data Pre-processing:

This research first downloads the video footages that are available online. The video footage contains a fixed camera that detects individuals in a region of interest (ROI) and measures their distances in real-time without recording any sort of data. Moreover, this research proposes a novel approach towards detecting people and whether or not they are violating any social distancing regulations.

While detecting the interpersonal distances between the individuals present in the video, with the help of facial detection, their faces are detected to verify whether the individuals are wearing a mask or not.

#### Object Detection and Tracking:

Although there has been a confusion between the two terms Image Classification and Object Detection, often meaning them to be the same, they are completely different. Image Classification performs identification of objects in images while Object Detection performs identification of the objects including its location in the images. Both of these terms are widely popular in Computer Vision tasks (Merkulova, Shavetov, Borisov and Gromov, 2019). They can be used in every field possible such as healthcare, defence, sports, and various other industries.

The next question that arises is whether Object Detection and Tracking are the same terms or not. Yes, Object Detection and Tracking are two very similar terms in the way they are functioned. They are basically designed to perform the same functionality but with a little difference. Object Detection is used to detect objects present in an image or in multiple images where an object is stationary while Object Tracking performs detection on videos, that is, it keeps a track of the following object detected while it is moving (Porikli and Yilmaz, 2012). A video is a combination of fast-moving frames and thus identification of the objects and their location from every frame is performed by Object Tracking.

#### Face Recognition with Open-CV:

OpenCV (Open-Source Computer Vision) is a popular computer vision library started by Intel in 1999. The cross-platform library sets its focus on real-time image processing and includes patent-free implementations of the latest computer vision algorithms. In 2008 Willow Garage took over support

and OpenCV 2.3.1 now comes with a programming interface to C, C++, Python and Android. OpenCV is released under a BSD license so it is used in academic projects and commercial products alike.

OpenCV 2.4 now comes with the very new Face Recognizer class for face recognition, so you can start experimenting with face recognition right away. This document is the guide I've wished for, when I was working myself into face recognition. It shows you how to perform face recognition with Face Recognizer in OpenCV (with full source code listings) and gives you an introduction into the algorithms behind. I'll also show how to create the visualizations you can find in many publications, because a lot of people asked for.

---

## 5. ALGORITHMS

The currently available algorithms are:

- Eigenfaces (see EigenFaceRecognizer::create)
- Fisherfaces (see FisherFaceRecognizer::create)
- Local Binary Patterns Histograms (see LBPHFaceRecognizer::create)

You don't need to copy and paste the source code examples from this page, because they are available in the src folder coming with this documentation. If you have built Open-CV with the samples turned on, chances are good you have them compiled already! Although it might be interesting for very advanced users, I've decided to leave the implementation details out as I am afraid, they confuse new users. All code in this document is released under the BSD license, so feel free to use it for your projects.

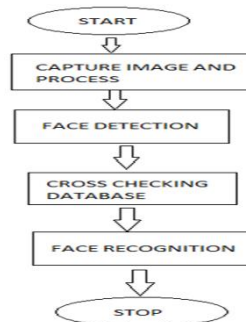
---

## 6. RESULT AND ANALYSIS

The process of identification of a person is divided into the following steps:

- Face detection
- Feature extraction
- Classification

There are several facial extraction processes, but LBP is used here. The whole picture is divided into numbers of pieces and they have the different values of pixels. These pixel values are compared with the threshold level and provides the decisions. The shape, location, dimensions etc. of the pictures affected the analysis. The pieces of the picture are putted into a single histogram and basis on the nature of histogram, the decision is taken. When an input image is given, then its histogram is created. Which is compared with the histograms of the images present in the dataset. The image which is matched best is recognized. The flow diagram of the process is shown in figure



The first process to the system is to store the numbers of images in its database. This database is used to identify the related person. If camera captured any face, then it calls to the database and check the measurements available in its database. The system is verified over 150 peoples and have an efficiency about 80 percent. It is tested on different environment and lighting conditions with different cameras and the results is near about same. The camera used here is Logitech C90 USB webcam.

---

## 7. ADVANTAGES & DISADVANTAGES

The advantages of the face recognition system include faster processing, automation of the identity, breach of privacy, massive data storage, best results, enhanced security, real time face recognition of students in schools and colleges, employees at corporate offices, smartphone unlock and many more in day to day life.

Few disadvantages in this system include the costing, or the funding, very good cameras of high definition are required, poor image quality may limit the effectiveness of this system, size of the image will matter because it becomes difficult to recognize the face in small images, Face angles can limit the face recognition reliability, massive storage is required for this system to work effectively.

## 8. CONCLUSION & FUTURE SCOPE

Face recognition systems are currently associated with many top technological companies and industries making the work of face recognition easier. The use of python programming and Open-CV makes it an easier and handy tool or system which can be made by anyone according to their requirement. The proposed system discussed in this project will be helpful for many as it is user friendly and cost\_efficient system. Hence by the use of python and Open CV the face recognition system can be designed for various purposes. To improve the calculation speed and efficiency, hog like features are used in face detection and face recognition is done by using LBPH because of its computational simplicity and accuracy. The unique parameters of the face are observed and compared with the data available in the dataset.

## REFERENCES

- [1] Malaya Khan, Sudeshna, Rani Astya, Face Detection and Recognition Using OpenCV, 2019 International Conference on Computing, Communication, and Intelligent Systems (ICCCIS)-IEEE\_2019.
- [2] Vijay Kumar Sharma, Designing of Face Recognition System, School of Electronics & Communication Engineering, Shri Mata Vaishno Devi University, Katra, India, 182320, 978-1-5386-8113-8/19/\$31.00 ©2019 IEEE.
- [3] Tejashree Dhawle<sup>1</sup>, Urvashi Ukey<sup>2</sup>, Rakshandha Choudante<sup>3</sup>, Face Detection and Recognition using OpenCV and Python, International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056 Volume: 07 Issue: 10 | Oct 2020.
- [4] J. Manikandan, S. Lakshmi Prathyusha, P. Sai Kumar, Y. Jaya Chandra, M. Umaditya Hanuman, "Face Detection and Recognition using Open CV Based on Fisher Faces Algorithm", International Journal of Recent Technology and Engineering (IJRTE).
- [5] Face Detection in Real Time Based on HOG. N. J. Wang, S. C. Chang and P. J. Chou. Taipei, Taiwan: IEEE, DOI:10.1109/ISPACS.2012.6473506, 2012. International Symposium on Intelligent Signal Processing and Communications Systems. pp. 333-337. ISBN: 978-1-4673-5081-5.
- [6] Face Detection and Tracking using OpenCV. S.V. Viraktamath, Mukund Katti, Aditya Khatawkar, Pavan Kulkarni. 3, s.l.: SIJ, July-August 2013, The Standard International Journals (The SIJ), Vol. 1, pp. 45-50. ISSN: 2321 – 2403.
- [7] . Dhillson and J. Backhouse, (2001), "Current direction in IS security research: towards socio organization a lperspective", Information Systems Journal, vol. 11, no.2, pp. 127-153, 2001.
- [8] Learning OpenCV –Computer Vision with the OpenCV Library O'Reilly Publication. Learning OpenCV: Computer Vision with OpenCV Library, Kindle Edition. Gary Bradsk<sup>1</sup> and AndrianKehlar.
- [9] M.A. Turk and A.P. Pentland, "Face Recognition Using Eigenfaces", IEEE Conf. on Computer Vision and Pattern Recognition, pp. 586-591, 1991.
- [10] "KyungnamKim" Face Recognition using Principle Component Analysis"
- [11] G B Huang, H Lee, E L. Miller, "Learning hierarchical representation for Face verification with convolution deep belief networks[C]", Proceedings of International Conference on Computer Vision and Pattern Recognition, pp.223-226, 2012.
- [12] Learning OpenCV: Computer Vision with the OpenCV Library 1st Edition, Kindle Edition.
- [13] Paul Viola, Matthew Jones Conference paper- IEEE Computer Society Conference on Computer Vision and Pattern Recognition. "Rapid object Detection using a boosted cascade of simple features."