Journal of Engineering Sciences

Vol 13, Issue 06, June/2022

ISSN NO:0377-9254

# Real Time Face Detection and Face Recognition using OpenCV and Python

*Pothuraju Chandrakala*
*MTech (CSE)*
*Department of CSE*
*Aditya College of Engineering & Technology*

*Dr. B.Srinivas*
*Assistant Professor*
*Department of CSE*
*Aditya College of Engineering & Technology*

*Dr. M.Anil Kumar*
*Professor*
*Department of CSE*
*Aditya College of Engineering & Technology*

**ABSTRACT:**

A facial recognition system is a type of computer program that uses a combination of facial recognition and computer vision to identify a person based on their appearance. Face recognition is commonly used in the fields of identity recognition and police investigations. It can be used to identify individuals based on their facial expression. This paper presents a method that uses OpenCV and Python to perform facial recognition. Deep learning is a major component of computer science that can be used to perform facial recognition. This paper presents a method that uses OpenCV and Python to perform facial recognition. The paper proposes a method that can be used to detect the human face in real time. This technology can be used in various applications, such as smartphones and machines and real time applications.

**Keywords: facial recognition, computer vision, deep learning.**

## I. INTRODUCTION

One of the most challenging tasks in the field of artificial intelligence is computer vision. This process involves connecting the various parts of a computer software to the visual representations we see around us.Through computer vision, software can learn about the visual representations in its environment. For instance, by determining the size and color of a fruit, its shape and appearance the type of fruit and its quality can be classified. The goal of computer vision is to provide a more accurate and useful information to humans. It can also help us identify and understand the various elements of the visual representations around us[1][2]. The term "openCV" refers to an open-source framework that enables the development of computer vision applications. It includes various software components and plugins that can be used to integrate these applications[3]. The openCV framework has been released several times. It is free and open-source, and it can be used for various tasks such as numerical productivity. The first version of openCV was released in C, though it has since been upgraded to version 2.0, which uses a C++ implementation. This version also supports new features. The openCV framework can be downloaded from the website opencv.org. It can be used with the latest version of the software, Photos must be in Grayscale or BGR format to be saved or

displayed in the framework. A type of computer vision application known as face detection can be used to visualize and identify facial features in real-time videos or photos. It can also perform object detection by analyzing the semantic artifacts of various classes in digital images. Due to the advancements in technology, face recognition has become more important. It can be used in various fields such as defense, marketing, and photography[4][5].

The field of recognition has caught the attention of researchers due to its simplicity by using open CV and python. This is because, with the use of Python, it can be easily implemented. Face recognition is widely used in various applications, such as public protection and social networking. It can also be used in man-machine contact. It can also be used in various applications, such as voter registration and attendance control[[6], [7]. This paper aims to provide an overview of the main roles of openCV in face recognition and how it can be used in various applications. It also states the various modules that can be used in the development of OpenCV. The paper also reviews the literature on the various applications of openCV in face recognition. It aims to provide an overview of the main roles of openCV in face recognition and how it can be used in various applications. The paper is divided into three sections: face detection, face recognition, openCV library and openCV algorithm. In section 2, it talks about face detection. In section 3, it covers face recognition. In section 4 covers about OpenCV library and openCV Algorithm. The section5 of this paper talks about the various modules of openCV. In section 6, the paper introduces the use of Python for OpenCV. In section 7, the paper assessment of the literature reviews and comparision table is discussed. Section 8 concludes the paper.

### 1.1 **Face Detection**

Due to its applications in human interaction and computer security, face detection has gained widespread attention. A subset of image processing, face detection is used to extract valuable information from images. It is mainly used to improve the compression and quality of images. With the help of facial recognition technology, it can identify individuals in a picture without disturbing the background noise. It can then classify the image into two groups based on its facial features. The goal of face detection is to analyze the image to identify the faces in it, and remove the background. There are two types of mistakes that face detection makes: false positive and false negative. A false positive mistake occurs when a face is mistakenly identified in a picture without any other faces. A false negative mistake occurs when the algorithm cannot identify the objects in the picture due to the number of faces already identified by humans. The detection rate of the algorithm

should be high to ensure that it can identify the faces in the image correctly[1][8].

### 1.2 Face Recognition

With facial recognition, you can easily identify and authenticate yourself using the most visible part of your body. This technology is the fastest and most accurate way to do so. According to worldwide data, most people are not aware that facial recognition is being performed on them. This procedure is considered to be one of the least invasive methods of facial recognition. The process of facial recognition involves analyzing the various features of a person's face. It is commonly regarded as a great tool for identifying potential threats such as terrorists and scam artists. However, widespread use of this technology has yet to be seen. Biometric face recognition is expected to replace fingerprint as the most common method of authentication and user identification in the near future.

### 1.3 OpenCV Library

OpenCV is a massive open-source library that can be used for various kinds of image processing and machine learning. It can be used in various languages such as Java and Python. It can be used to analyze videos and photographs for various kinds of objects, such as human handwriting. It can be integrated with other libraries, such as the highly-performing numpy. This allows users to get a good performance out of their work. OpenCV is written in C++, and its main interface is a C++ implementation. However, it has a slightly less robust language training. The latest algorithms and technologies are available in the C++ GUI. Also, the various languages that are used to implement OpenCV, such as Java, Python, and C++, are supported by the platform[3]. Wrappers have been created in various programming languages to promote wider acceptance. The latest version of OpenCV, which is 3.4, is available in JavaScript plugins. These can be used on web platforms. Originally, the OpenCV project was focused on supporting CPU-intensive applications. In 1999, it was released as an open-source library. It has been widely used for implementing various face detection and recognition techniques[11].

### 1.4 Haar Cascade

The Haar Cascade method is a machine-learning technique that can be used to detect objects. It learns from a series of images that have positive and negative characteristics. The Haar cascade method can be used to see objects in different frames. In the first image, the Haar cascade classifier shows the view of its operation.

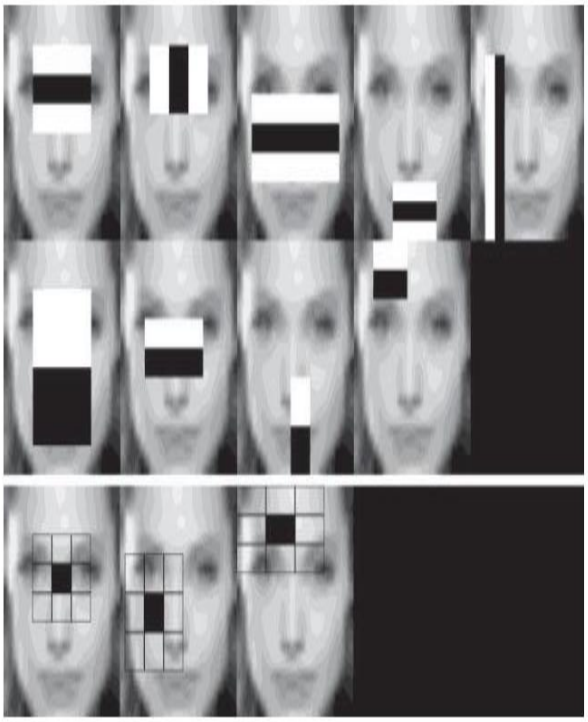

Fig1: View of Haar Cascade Classifier

### 1.5 Local Binary Pattern

A local binary pattern (LBP) is a simple technique that labels images by region and number. It can be used to treat the resulting image as a binary number. Due to its computational simplicity and discriminative power, the LBP has become a popular technique for creating various image labels. The LBP is regarded as a unifying solution to the various structural and statistical models used in texture analysis. One of its key features is its ability to handle monotonic changes induced by illumination variations. This is very important in real-world applications. Its computational simplicity allows it to perform complex analysis in real time[13]. The image below shows the various facial expressions that are labeled using local binary patterns.

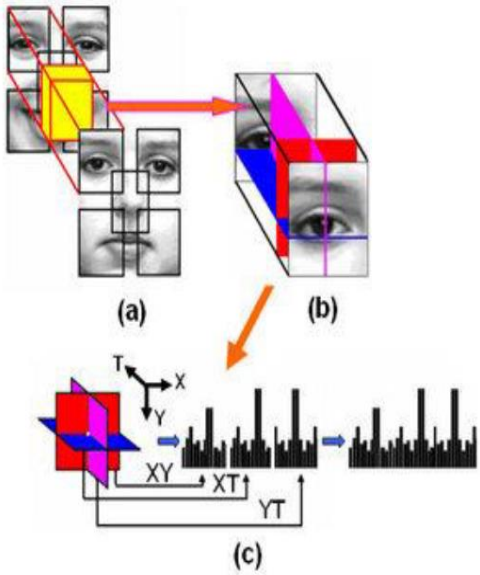


Fig 2: Description of Facial Expression with Local Binary Pattern

### 1.6 Eigen Faces

In order to minimize the dimensionality of the images, the LBP is used to perform a Principal Component Analysis (PCA) on the distribution of facial images. The goal of this process is to find the best possible vectors for each facial expression. The eigenvalue distribution is used to calculate the principal component amount of a facial image. This process is performed by taking into account the covariance matrix's eigenvalue and m eigenvector. After selecting the appropriate eigenvector, the LBP then generates the appropriate principal variable[1][14].

### 1.7 Fisher Faces

Face recognition has been a promising tool for researchers to identify people accurately. The Fisher Face model combines the PCA and FLD calculation models to perform a face recognition computation. The combination of these two models simplifies the operation of the FLD by reducing the input data. The FLD model is used to produce a distribution matrix that can be used to classify and identify people. The face recognition process involves four steps: face identification, classification, estimation, and calculation.

### 1.8 LBPH

The LBP texture operator is a widely used technique in various applications, such as facial recognition. It takes into account the various outcomes of a given binary number by comparing the values of the center and the neighboring pixels. This technique was first identified in 1994. Since then, the LBP algorithm has evolved into a more accurate and efficient method for classification of textures. It was also discovered that combining the technique with the data collected by the directed gradient descriptor can increase its accuracy. The LBP algorithm can also perform various statistical functions, such as the monotonic grey-scale computation, to improve its performance in real-time applications.

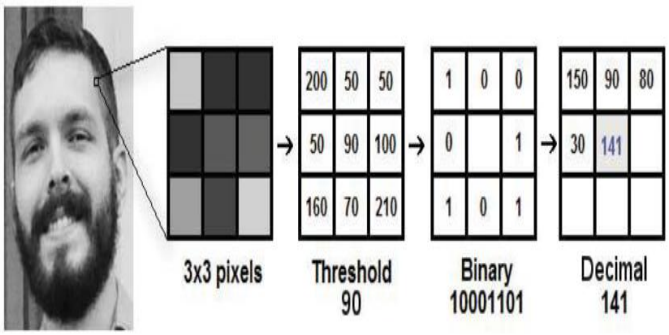


Fig3: LBPH Algorithm for Face Recognition

### 1.9 YOLO

The Yolo system is a real-time object detection system which only works once. It takes advantage of a single neural network to process an image[16]. The network then breaks the image into sections and estimates the probabilities of each box. It then weights the various boxes with the help of the estimated probabilities[17]. The testing phase helps the system identify objects in the picture. It also predicts the likelihood of them forming videos or images[18][19].

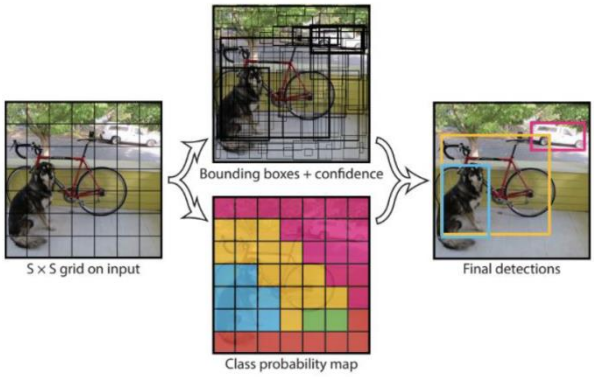


**Fig 4: The YOLO Process**

### 1.10 Faster R-CNN

In 2015, Ross Girshick proposed a framework that uses a combination of deep learning and computational methods to perform object detection. It is regarded as one of the most popular platforms for developing such systems. The R-CNN framework was able to greatly improve the efficiency of object detection by implementing a region proposal network (RPN). This network is a completely unsupervised system that trains side-by-side with each other to predict the boundaries of objects. This paper aims to introduce the R-CNN framework's architecture and features, as well as the concepts of suppression and anchor boxes. RPN is a critical component of the framework, and it remains one of the most widely used entity detection frameworks.

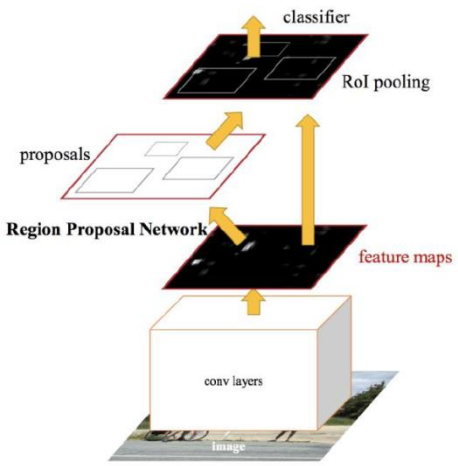


Fig5: Faster R CNN Process

### 1.11 Single Shot Detectors (SSDs)

The feed-forward single shot detector (SSD) is a method that generates a border-box array consisting of class-based entity instances in a set of boxes. It then generates a final detection by preserving the original image classification design[16][20].

## 2. Modules of OpenCV

### 2.1 Main Modules

Main modules are the main components of OpenCV and are included in the package versions by necessity. They are called core modules since they include essential functions, including image recognition, filtering, and transformation [8], [21].

### 2.2 Extra Modules

Extra modules are not included in the OpenCV release by default. Additional machine vision functionalities, such as text detection, are applied to these modules [21].

The main modules are described below:

- Core: Have most of OpenCV's core features.
- Imgproc: Picture processing tools such as transforms, manipulations, and filtering are used.
- Imgcodecs: Image reading/writing features are included.
- Videoio: Video reading/writing features are included.
- Highgui: Used for creating GUI to display output.
- Video: Motion detection and monitoring.

 calib3d: This package includes calibration and 3D reconstruction functions for estimating translation between multiple pictures.

 features2d: Item identification and categorization algorithms utilizing keypoint detection and descriptor extraction algorithms, which are included in this library.

 Objdetect: used for detecting object.

 DNN: Classify and detecting objects, between other objects.

 Ml: Used for regression and classification, and it encompasses the vast majority of machine learning.

 Flann: Supports optimized algorithms for searching for high-dimensional attributes in massive data sets using nearest neighbor search. Fast Library for Approximate Nearest Neighbors (FLANN) is an acronym for Quick Library for Estimated Nearest Neighbors (FLANN)[22]. Photo: Removes noise and creates HD images, among other roles for photography-related computer vision.

 Stitching: Used for stitching image.

 Shape: Deal with issues such as shape transformation, pairing, and distance.

 Superres: Deal with enhancement and resolution algorithms.

 Videostab: Have algorithms for stabilizing video.

 Viz: create 3D display window for widgets.

**2.3 OpenCV based on Python**

Guido van Rossum created the Python programming language to make it easy to understand. He wanted to reduce the number of characters in the code to make it easier for the programmer to communicate. Compared to other programming languages, such as C/C++, Python is slower. However, it can be used to expand the capabilities of C/C++. Also, Python is very flexible. The C++ function that adds a function to the output of a program helps in the implementation of C++ code. This can be used to create various modules in Python. One of the main advantages of using Python is that it is as fast as C++. Also, it is very easy to program in. Through the use of the openCV function, you can also create various applications that are easy to implement. The implementation of Python is built on top of the C++ code. Also, the contributions of the other libraries, such as the numpy, help in the development of high-level statistical applications. In addition to the C++ code, the implementation of Python also takes advantage of the various features of the openCV function. It can convert various OpenCV structures into the appropriate Numpy arrays. This is very good because it makes it easier to implement various functions in the same package. Also, the SciPy library can be used with this to create computer vision projects. With the use of openCV, you can quickly build out complex computer vision projects.

Assessment of Literature Reviews for Face Detection and Face Recognition

**Table 1 - Comparison table**

| Ref. | Aim | classifier | accuracy | Result/purpose |
|---|---|---|---|---|
| [21] 2018 | SVM | OpenCV more accurate than dlib | 83% head detect | The OpenCV library is more productive, has improved facial recognition and detection accuracy |
| [22] 2018 | Head Detection and Tracking | Haar-like<br>CMT<br>Cascade | 68% tracking | The proposed system successfully detected the head of a human using OpenCV libraries, specifically using Haar-likeattribute detection. |
| [9] 2018 | comparative of classifying the face using different classifiers | MLP<br>Extra Tree<br>(Random Forest)<br>KNN<br>RadialSVM<br>Gaussian NB<br>LinearSVM<br>(Logistic Regression) | 99.1%<br>86.4%<br>95.9%<br>99.3%<br>98.4%<br>98.8%<br>99.2%<br>99.4% | The results show that Logistic Regression outperforms the other algorithms for face classification in terms of speed andaccuracy. |
| [7] 2018 | real-time recognitionFacial emotion | Haar<br>SVM | 93.7% | The findings suggest that with today's computing power, user-independent, completely automated real-time coding of facial expressions in a continuous video stream is a goal that can be achieved. |
| [4] 2018 | Face Detection under Different Lighting | Haar | 80% | The experiment demonstrated that the picture processing system has facial recognition in various lighting conditions. |
| [18] 2019 | Designing ofFace Recognition System | Haar-like<br>LBPH | 80% | The system is tested by more than 150 people and has a reliability of approximately 80%. It is measured with multiple cameras in various settings, and lighting conditions, and the findings are about the same. A Logitech C90 USB webcam is used |

| | | | | |
|---|---|---|---|---|
| | | | | here. |
| [12] 2019 | Student Monitoring System for School Bus | Haar-Cascades LBPH Eigenfaces | 85% | The system watches the bus and detects the students and their movements, acknowledges the faces of the students, and their count is also tracked and alerts the audience if necessary. |
| [13] 2019 | Face Recognition in Parallel Computer | Keras VGG-Face | module performance depends on the number of processors | The results collected were rather similar to what was predicted. The efficiency gained by simply running the program on a machine with more computing capacity and cores than a simple laptop was important. |
| [14] 2019 | Recognition of the gender, age, and face of the person | cascade LBP LBPH | successfully work but affected by mobile type, face coverage, expressions of face, | Gender, face, and age recognition was achieved |

## 3. Conclusion

Computer Vision is a field of artificial intelligence that focuses on the processing of images and videos. It involves developing computers that can extract the most important details from the images. The OpenCV library, which is a C++-based open-source project, provides various functions for developing computer vision applications. Some of these include object detection, facial recognition, and medical diagnosis. In this paper, we discuss the various functions of OpenCV in face recognition and object detection. We show how the library can be used to develop effective face recognition and object detection algorithms. We then introduce the OpenCV modules and their applications, and we discuss the various literature reviews that have been published about the library. We then conclude by talking about the human face that can be recognized using the OpenCV platform.

## REFERENCES

[1] Jagtap, A. M., Kangale, V., Unune, K., & Gosavi, P. (2019, February). A Study of LBPH, Eigenface, Fisherface and Haar-like features for Face recognition using OpenCV. In 2019 International Conference on Intelligent Sustainable Systems (ICISS) (pp. 219-224). IEEE

[2] Sigut, J., Castro, M., Arnay, R., & Sigut, M. (2020). OpenCV basics: a mobile application to support the teaching of computer vision concepts. IEEE Transactions on Education, 63(4), 328-335

[3] Adusumalli, H., Kalyani, D., Sri, R. K., Pratapteja, M., & Rao, P. P. (2021, February). Face Mask Detection Using OpenCV. In 2021 Third International Conference on Intelligent Communication Technologies and Virtual Mobile Networks (ICICV) (pp. 1304-1309). IEEE

[4] Mostafa, S. A., Mustapha, A., Gunasekaran, S. S., Ahmad, M. S., Mohammed, M. A., Parwekar, P., & Kadry, S. (2021). An agent architecture for autonomous UAV flight control in object classification and recognition missions. Soft Computing, 1-14

[5] Hoque, M. A., Islam, T., Ahmed, T., & Amin, A. (2020, March). Autonomous face detection system from real-time video streaming for ensuring the intelligence security system. In 2020 6th International Conference on Advanced Computing and Communication Systems (ICACCS) (pp. 261-265). IEEE
[6] Soomro, Z. A., Memon, T. D., Naz, F., & Ali, A. (2020, January). FPGA Based Real-Time Face Authorization System for Electronic Voting System. In 2020 3rd International Conference on Computing, Mathematics and Engineering Technologies (iCoMET) (pp. 1-6). IEEE
[7] Gupta, S. (2018, January). Facial emotion recognition in real-time and static images. In 2018 2nd international conference on inventive systems and control (ICISC) (pp. 553-560). IEEE
[8] Dino, H., Abdulrazzaq, M. B., Zeebaree, S. R., Sallow, A. B., Zebari, R. R., Shukur, H. M., & Haji, L. M. (2020). Facial expression recognition based on hybrid feature extraction techniques with different classifiers. TEST Engineering & Management, 83, 22319-22329
[9] Kashinath, S. A., Mostafa, S. A., Mustapha, A., Mahdin, H., Lim, D., Mahmoud, M. A., ... & Yang, T. J. (2021). Review of Data Fusion Methods for Real-Time and Multi-Sensor Traffic Flow Analysis. IEEE Access
[10] Abdulrazaq, M. B., Mahmood, M. R., Zeebaree, S. R., Abdulwahab, M. H., Zebari, R. R., & Sallow, A. B. (2021, February). An analytical appraisal for supervised classifiers' performance on facial expression recognition based on relief-F feature selection. In Journal of Physics: Conference Series (Vol. 1804, No. 1, p. 012055). IOP Publishing
[11] Sriratana, W., Mukma, S., Tammarugwattana, N., & Sirisantisamrid, K. (2018, July). Application of the OpenCV-Python for Personal Identifier Statement. In 2018 International Conference on Engineering, Applied Sciences, and Technology (ICEAST) (pp. 1-4). IEEE
[12] James, C., & Nettikadan, D. (2019, April). Student monitoring system for school bus using facial recognition. In 2019 3rd International Conference on Trends in Electronics and Informatics (ICOEI) (pp. 659-663). IEEE Hasan et al., Journal of Soft Computing and Data Mining Vol. 2 No. 1 (2021) p. 86-97
[13] Balachandran, B., Saad, K. F., Patel, K., & Mekhiel, N. (2019, December). Parallel Computer for Face Recognition Using Artificial Intelligence. In 2019 14th International Conference on Computer Engineering and Systems (ICCES) (pp. 158-162). IEEE
[14] Salihbašić, A., & Orehovački, T. (2019, May). Development of android application for gender, age and face recognition using opencv. In 2019 42nd International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO) (pp. 1635-1640). IEEE
[15] Apoorva, P., Impana, H. C., Siri, S. L., Varshitha, M. R., & Ramesh, B. (2019, March). Automated criminal identification by face recognition using open computer vision classifiers. In 2019 3rd International Conference on Computing Methodologies and Communication (ICCMC) (pp. 775-778). IEEE
[16] Gupta, N., Sharma, P., Deep, V., & Shukla, V. K. (2020, June). Automated attendance system using OpenCV. In 2020 8th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions)(ICRITO) (pp. 1226-1230). IEEE
[17] Srivastava, M., Kumar, A., Dixit, A., & Kumar, A. (2020, February). Real time attendance system using face recognition technique. In 2020 International Conference on Power Electronics & IoT Applications in Renewable Energy and its Control (PARC) (pp. 370-373). IEEE
[18] Sharma, V. K. (2019, May). Designing of Face Recognition System. In 2019 International Conference on Intelligent Computing and Control Systems (ICCS) (pp. 459-461). IEEE
[19] Das, A., Ansari, M. W., & Basak, R. (2020, December). Covid-19 Face Mask Detection Using TensorFlow, Keras and OpenCV. In 2020 IEEE 17th India Council International Conference (INDICON) (pp. 1-5). IEEE
[20] Mehariya, J., Gupta, C., Pai, N., Koul, S., & Gadakh, P. (2020, July). Counting Students using OpenCV and Integration with Firebase for Classroom Allocation. In 2020 International Conference on Electronics and Sustainable Communication Systems (ICESC) (pp. 624-629). IEEE
[21] Boyko, N., Basystiuk, O., & Shakhovska, N. (2018, August). Performance evaluation and comparison of software for face recognition, based on dlib and opencv library. In 2018 IEEE Second International Conference on Data Stream Mining & Processing (DSMP) (pp. 478-482). IEEE
[22] Alcantara, G. K. L., Evangelista, I. D. J., Malinao, J. V. B., Ong, O. B., Rivera, R. S. D., & Ambata, E. L. U. (2018). Head detection and tracking using OpenCV. In 2018 IEEE 10th International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment and Management (HNICEM) (pp. 1-5). IEEE
[23] Patel, R., Patel, M., & Patel, J. (2018, April). Real Time Somnolence Detection System In OpenCV Environment for Drivers. In 2018 Second International Conference on Inventive Communication and Computational Technologies (ICICCT) (pp. 407-410). IEEE
[24] Sharma, S., & Jain, S. (2019, March). A static hand gesture and face recognition system for blind people. In 2019 6th International Conference on Signal Processing and Integrated Networks (SPIN) (pp. 534-539). IEEE.
[25] Zhu, Z., & Cheng, Y. (2020). Application of attitude tracking algorithm for face recognition based on OpenCV in the intelligent door lock. Computer Communications, 154, 390-397
[26] Chandan, G., Jain, A., & Jain, H. (2018, July). Real time object detection and tracking using Deep Learning and OpenCV. In 2018 international conference on

inventive research in computing applications (ICIRCA) (pp. 1305-1308). IEEE
[27] Khan, M., Chakraborty, S., Astya, R., & Khepra, S. (2019, October). Face Detection and Recognition Using OpenCV. In 2019 International Conference on Computing, Communication, and Intelligent Systems (ICCCIS) (pp. 116-119). IEEE
[28] Palekar, R. R., Parab, S. U., Parikh, D. P., & Kamble, V. N. (2017, April). Real time license plate detection using openCV and tesseract. In 2017 international conference on communication and signal processing (ICCSP) (pp. 2111-2115). IEEE