**Implementing Optical Port Data Communication in Energy Meters using DLMS/COSEM**

Prof. P. G. Chilveri1, Prof A. B. Chandgude2, Nikhil Ekvachane3, Soham Tembe4, Sumeet Upasani5

***Abstract*: - Energy meters are used for commercial purpose to record consumption of energy by consumer and correspondingly billing the consumer. Indian metering industry is a heterogeneous one with multiple communication protocols so even though there are many communication technologies used today for meter reading applications they lack the properties of interoperability and homogeneity. The presented solution describes a microcontroller-based device, which works on an Open Protocol, that provides real time local data display and record, for several electrical parameters, acquired from a power and energy meter device through optical communication. The open protocol used here is IEC 62056, which helps to overcome most of the challenges regarding data acquisition, uniformity and homogeneity among various energy meter manufacturers in India. This project focuses on DLMS COSEM Metering application protocol and provides guidelines on the implementation of DLMS COSEM in clients (data collection units) as well as in servers (Meters). The device is designed to be connected in a home, or residence, which makes it very useful as an interface for Optical Port Data Communication.**

***Keywords: Energy meter, DLMS/COSEM, Optical communication, IS15959***

# INTRODUCTION

As now education system is shifting to virtual platform, there might be possibilities of any misconduct happening. In an educational programs, exams are a critical component. The detection and malpractice prevention in exam is important to maintain academic integrity. When exams are administered in a conventional and proctored classroom environment, the students are monitored by human proctor throughout the exam. We introduce an AI based auto proctored system for continuous online supervision, in case if there is unavailability of halls and during lockdown if the exam needs to be conducted. This system takes the access of the candidate’s system web-camera and microphone. By tracking the eye movement and any audio disturbance the candidate is warned by the system. Candidate is monitored continuously, and based on proctored report, if candidate is guilty or not is decided. Main goal of this system is to maintain academic integrity of the exams and prevent the students from misconduct.

# OBJECTIVES

* To study the movements of a participant's eye in the course of various proctored activities.
* To avoid any misconduct happening by detecting users voice, active window and gaze.
* To take access of the candidate system by enabling web-camera and microphone, for proper functioning of exam.
* To maintain academic integrity in e-learning.

# REVIEW OF LITERATURE

[1] A Real-TimeMouse Pointer Control Implementation Based on Eye Gaze Tracking is discussed by aryam Sohail, M.Nafees Geelani in 2012. This review focuses on the testing the potential of eye gaze as a possible means of interaction application using built-in camera available in computing devices. A low cost real time solution for eye gaze tracking is developed with the help of opencv library. Author has used qt framework for GUI which has several advantages like formation of gui using cross platform GUI library. Conclusion was discussed and suggestions are given on where more study is needed.

[2] Zhou Zhang, EI Sayed Aziz, Sven Esche, Constantin Chassapis in their research named A Virtual Proctor with Biometric Authentication for Facilitating Distance Education in 2011. These authors focused on efficient and reliable proctoring for tests. In this paper, author took two stage approach (facial detection and facial recognition) for designing the virtual proctor is introduced. The hardware is available and affordable. A common webcam, instead of special bio- metrics data readers, can meet the hardware requirements for real-time facial recognition and tracking. The results proved that the proposed method has a higher reliability than the existing facial recognition methods and with certain drawbacks like lighting conditions.

[3] Sebastian Hoffner proposed the system named Gaze Tracking Using Common Webcams in 2018. This author effectively uses gaze tracking using a calibration free, feature-based approach on laptops using built-in webcams. To try the model, the free and open source software library Gaze is implemented and evaluated. It is shown that Gaze reaches very good eye tracking capabilities and manages To be easily usable and extendable. while drawbacks in webcam gaze tracking like, most web- cams are limited to small frame rates and it is extremely difficult to find the real specification like sensor sizes and focal lengths for many webcams warrants for further studies in the field.

[4] Automated Online Exam Proctoring is discussed by Yousef Atoum, Liping Chen, Alex X. Liu, Stephen D. H. Hsu in 2015. For this review, the author focused on a multimedia analytics system for online exam proctoring, which aims to maintain academic integrity in e-learning. Advantage proposed buy author is that the system is affordable and convenient to use from the

text taker’s perspective, since it only requires to have two inexpensive cameras and a microphone. With the collected database of 24 test takers representing real- world behaviors in online exam, author demonstrated the capabilities of the system. Based on results author warrants further research on this important behavior recognition problem and its educational application.

[5] Mortiz Kassner, William Patera in their research named an open source platform for mobile eye tracking and gaze based interaction in 2012, It makes use of high resolution cameras, open source software framework and graphical user interface to playback and visualize video and gaze data. Results proved that pupil can provide gaze estimation accuracy of 0.6 degree with latency of processing time of 0.045 seconds. The limitations occur to this system are parallox error and tracking robustness in IR rich environment. Further advances require accessible, affordable and extensible eye tracking tools.

[6] Probin Sharma and Shubham Joshi proposed A system to detect engagement level of students in 2020. It makes use of laptop inbulit webcam to grab real time information about student which determines the concentration level. To determine the concentration indexes, python and keras for facial emotion analysis, haarcascade algorithm for eye tracking and CNN are used. By performing testing with fifteen students, the results showed that the system correctly identifies engagement period of student. This system wasn't able to find concentration index values they were fluctuating because the student wore glasses. For further improvement they make use of other capturing cameras besides laptop built in web-camera.

[7] Dhaval Pimplaskar, Dr. M. S. Nagmode, Atul Borka in their research named Real Time Eye Blinking Detection and Tracking Using Opencv in 2013. It focused on method to determine eye position and direction based on initial centroid analysis technique. This was validated by tracking eye position within high and low closure condition in real time. It runs 25-30 frames per second for implementation. This technique was implemented in C++ using OpenCV library in windows environment using single camera. The major benefits of this is to capture images without any instruction. For this review the connected component technique and centroid method is used to track and blinking of eyes by using opencv platform.

[8] Aniket M. Taksande and Prof. R. P. Patil proposed an implementation of Face and Eye Tracking System Using High Level Synthesis in 2016, For this review, This system is developed by implementing an algorithm proposed by Viola-Jones for face and eye detection and tracking. By monitoring various features of face and eye, attention of person can be estimated. Also given system uses Vivado High Level Synthesis (HLS) for implementing Viola-Jones algorithm. Xilinx Zynq-7000 SoC series I is used as hardware platform, processing platform. OpenCV libraries which gives optimum real time image processing, is used to obtain required application. This system is running on the ARM Cortex-A9 dual core processor of the Zynq-7000 SoC and with Vivado HLS along with OpenCV libraries, and the results show that the system performs satisfactorily when a face is not inclined. However, if face is inclined at certain angle, system stops detection and tracking of face and eyes. This system will not gives accurate results when the light intensity in a room is not sufficient to capture images also if person is wearing any spectacles will resist eye detection and tracking.

[9] Attention Based Glaucoma Detection: A Large-scale Database and CNN Mode is discussed by Liu Li, Mai Xu, Xiaofei Wang, Lai Jiang, Hanruo Liu in 2019, In this paper, the authors proposed a new deep learning method, named AG-CNN, for automatic glaucoma detection and pathological area localization upon fundus images. The AG-CNN model is composed of the subnets of attention prediction, pathological area localization and glaucoma classification. This model implemented in python using deep learning. The experiment results showed that the predicted attention maps significantly improve the performance of glaucoma detection and pathological area localization in the AG-CNN method, far better than other state-of-the-art methods.

[10] Advancing NLP with Cognitive language processing signals is discussed by Nora Hollenstein, Maria Barrett, Marius Troendle, Francesco Bigiolli, Nicolas Langer, ce Zhang in 2019. For this review, the authors explored that when we read, our brain processes language and generates cognitive processing data such as gaze patterns and brain activity. These signals can be recorded while reading. Cognitive language processing data such as eye-tracking features have shown improvements on single NLP tasks. These authors analyze whether using such human features can show consistent improvement

[11] Hemalatha Vadlamudi proposed an Evaluation of Object Tracking system using Opencv in python in 2020. This object tracking system aims to improve performance of object detection and tracking by contributing originally to two components ie. motion segmentation and object tracking. 2.In this Object tracking system, using YOLO based GMM model and OpenCV to improve the accuracy of the Object in video and also It can easily track the moving object by using point tracking and kernel tracking methods. It not only detect the object but also track the object in the videos. 3.Some of the main advantages of this system is To identify the targeted object in moving sequence, To analyze YOLO based algorithm with GMM model to get good accuracy for feature extraction and classification, To analyze the motion of the object in a video using OpenCV, To analyze SSD and Mobile Nets algorithm for tracking the objects.

[12] Ahmad Aljaafreh, Murad Alaqtash, Naeem Al-Oudat, Jafar Abukhait, and Ma’en Saleh in their research named A Low- cost Webcam- based Eye Tracker and Saccade Measurement System in 2020. For This Survey, A consistent algorithm is

developed to suit the quality of the webcam using open-source software (Python) to record the time series of the eye location. Likewise, several algorithms are proposed to extract highlevel eye movement saccadic measurements from the raw gaze outputs. Pygame, Dlib and OpenCV software packages are used for face, eye, and pupil detection. The time and location of both pupils and marker are saved in a file for processing. Experimental results demonstrated that the proposed strategy is a quick, simple and efficient technique for eye tracking and saccade measurement. Authors suggests that in future machine learning techniques can be utilized to recognize and classify normal and pathological gaze patterns.

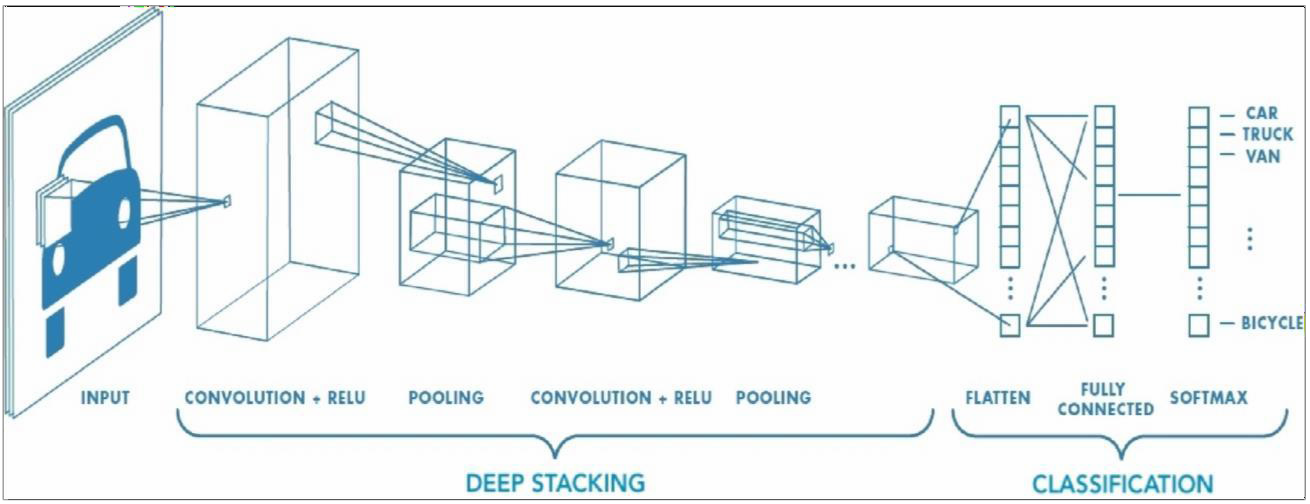
[13] Ildar Rakhmatulin and Andrew T. Duchowsk proposed A Deep Neural Networks for Low Cost Eye Tracking in 2020. This paper focuses on detailed analysis of modern techniques that can be used to track eye gaze. A dual coordinate system is given for controlling the computer with the help of a gaze. The first set of coordinates–the position of the face relative to the computer, is implemented by detecting color from the infrared led via opencv library. The second set of co-ordinates giving gaze position is obtained via YOLO package. They have used haar cascade and dlib to monitor gaze position and deep learning for gaze detection and datasets. Author suggest it is necessary to focus on standardization of research in the field of eye movement, which will allow more competent comparison of various research and give a more complete picture of the situation in the field of development of these technologies and thereby more intelligently present to researchers their developments.

[14] Optimal visual search based on a model of target detectability in natural images is discussed by Shima Rashid, Krista Ehinger, Andrew Turpin, Lars Kulik in 2020. The author focuses to develop a model of a target detectability in natural images to estimate whether target is present or not. It detects known target in natural background based on biological aspects of human visual system. They had made used of automated prediction algorithm of a pretrained deep neural network. Results showed that human observers performed worse than ideal observers. It is shown that deeper CNNs, are not better at producing human like representations, so development of sophisticated models of visual attention and fixation control.

[15] Achudan Ts and Gobinath N proposed A Real Time Eye Gaze Detection using Machine Learning Techniques in 2020. The authors focusses on to determine the drivers vigilance using computer vision system in real time. The face detection, Eye openess detection, closeness detection is done with the help of opencv algorithm. This system works under realistic and variable lightning conditions in real time. The results showed that driver can be alerted through this. This system is compatible with many operating systems.

# METHODOLOGY

As in any other neural network, the input of a CNN, in this case an image, is passed through a series of filters in order to obtain a labelled output that can then be classified. The specificity of a CNN lies in its filtering layers, which include at least one convolution layer. These allow it to process more complex pictures than a regular neural network.



# Fig. 1 Block Diagram of Image Capture and Image Process

**Convolution layers**

Since pictures can be very big, it is inefficient to have every pixel as an input. In fact, you wouldn’t just need every pixel, but each of the individual RGB values of each pixel to be an input. Pre- processing is therefore required, through a series of layers which

appear at the beginning of the neural network to reduce it into a smaller input, before applying a traditional neural network to an image. The most important of these layers are the convolutional layers. These convolutional layers are, as the name suggests, made out of a basic building block called a convolution. A convolution is applied to small regions of an image, sampling the values of pixels in this region, and converting it into a single pixel. It is applied to each region of pixels in the image, to produce a new image. The idea is that pixels in the new image incorporate information about the surrounding pixels, thus reflecting how well a feature is represented in that area.

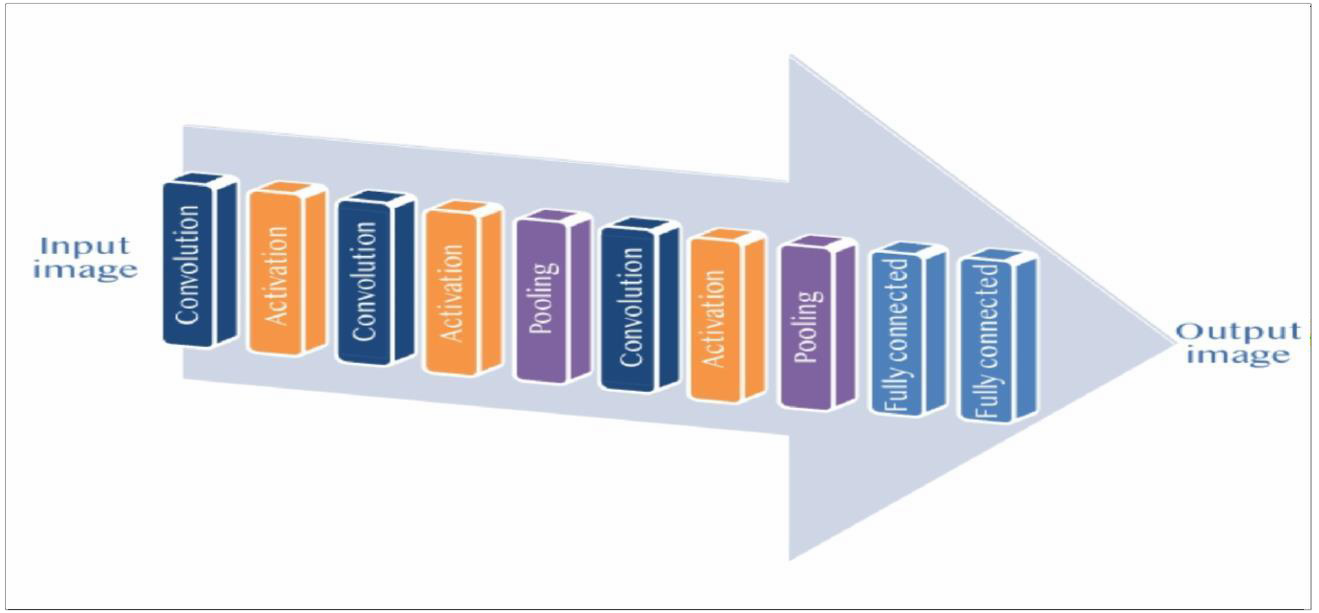
Convolutions have two important attributes, size and stride. The convolution below is applied to the image is of size 3x3 and stride

1. It is of size 3x3 because it operates on 3x3 grids of pixels. It has stride 1, since the gap between regions that the convolution is applied to is 1. In other words, the convolution advances by 1 each item. The size and stride of the convolution determines the size of the output image. For instance, 3x3 convolution with stride 1, converts a 5x5 image to a 3x3 image.

# Pooling layers

In addition, there is another kind of layer called a max pooling or down-sampling layer. These are interspersed every so often between the convolution-activation pairs. They do not necessarily have to occur after every convolution-activation pairs. The convolutional layers aren’t supposed to reduce the size of the image significantly. Instead, they make sure that each pixel reflects its neighbours. This makes it possible to perform downscaling, through pooling, without losing important information. A widespread method to do so consists in max pooling, in other words using the maximum value from a cluster of neurons at a previous layer. Indeed, max-pooling layers have a size and a width. Unlike convolution layers, they are applied to the 2dimensional depth slices of the image, so the resulting image is of the same depth, just of a smaller width and height.

# Deep stacking towards a fully connected layer



**Fig. 2 Deep stacking towards a fully connected layer**

The different types of layers described are combined through a repeated stacking process illustrated below, known as deep stacking. Convolutional layers followed by activation functions, as well as interspersed down-sampling layers, essentially replace width and height with depth until the image is relatively small.

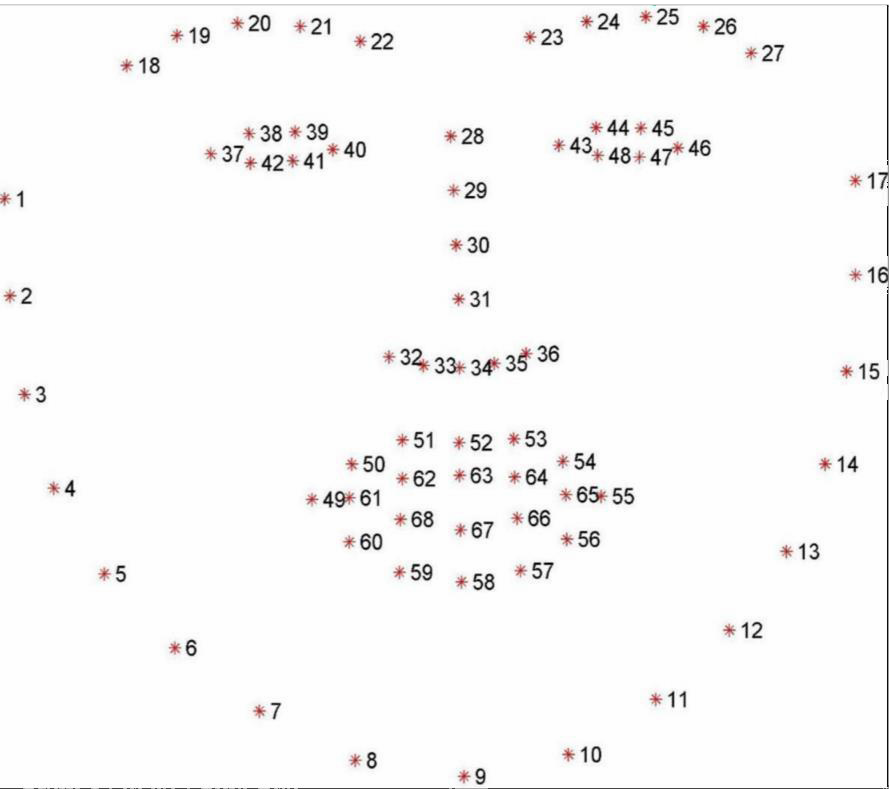
# Fully connected layer

After several convolutional and max pooling layers, the high-level reasoning in the neural network is done via fully connected layers. Neurons in a fully connected layer have connections to all activations in the previous layer, as seen in regular (nonconvolutional) artificial neural networks. Their activations can thus be computed as an affine transformation, with matrix multiplication followed by a bias offset (vector addition of a learned or fixed bias term).

# EXPERIMENTATION AND RESULTS

The first thing to do is to find eyes before we can move on to image processing and to find the eyes we need to find a face. The facial keypoint detector takes a rectangular object of the dlib module as input which is simply the coordinates of a face. To find faces we can use the inbuilt frontal face detector of dlib.

Following is the image for the points we are going to detect on face.



From 43 to 48 : Second Eye From 28 to 36 : Nose

From 48 to 60 : External Part Of Lips From 61 to 68 : Internal Parts of Lips

First we import libraries,

OpenCV-Python is a library of Python bindings designed to solve computer vision problems. NumPy is a Python library used for working with arrays.

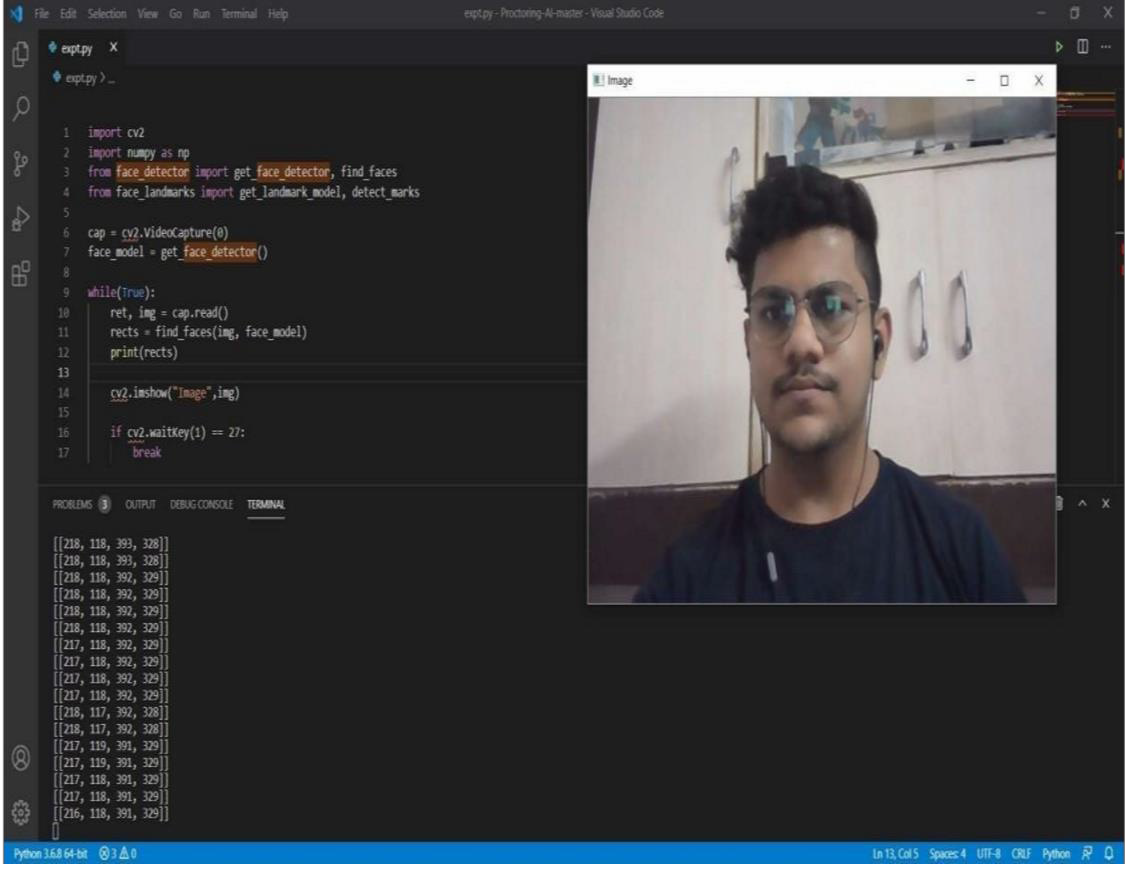
The Dlib library is the most popular library for detecting landmarks in the face. There are two types of detectors in this library. Then we need to take frames from camera, So we load

0 is for first webcam, if having different webcams index can be changed to capture frame from that camera.

Then we run while loop to get frame from camera

Now frame is converted into gray scale format, We detect the faces on the in rect And we get face model. Now we print rects to see what we get,

Here we get co-ordinate of face, and we have two points we have the point at top left of the screen where the face is detected and point at the right bottom and we need exactly these specific points to get a rectangle where our face is same can be done using a grayscle image. Convert to grayscale detector r = dlib.get\_frontal\_face\_detector() rects contains all the faces detected



# RESULTS

As this system uses AI based Proctor the results turn out to be very helpful for the organizations which monitors through this system. As this system makes use of web-camera for gaze estimation, object detection or any other misbehaviour in the frame is suddenly caught by the system. Based on the images captured by web-camera the candidate is being monitored and the warning is being displayed on the users screen. Microphone captures the surrounding as well as the users noise and try to identify it. Use of python with dlib and opencv makes the system easy to build. This system turns out to be user friendly by taking the survey of many test takers and also the overall efficiency of this system makes it better than human proctor.

# DISCUSSION

The main contribution of this work is to present a comprehensive framework for online exam proctoring. By the reviews of the test takers it is observed that system turns out to be user friendly. But by making the slight changes the system can be improved for further uses. We can make use of more advanced algorithms, such as deep learning based feature representation, typing based continuous authentication , face alignment based posed estimation. We may also make use of additional component such as pen detection. Even after this users may cheat outside the fields of view of cameras. So as to prevent this the system can generate random commands like asking the test takers to look around or under the desk top check the surrounding environment. By displaying a simple icon on the computer screen to check that the web-camera can see it or can play a quick sound clip to check that microphone can hear it this random commands and interventions makes system more robust against cheating behaviour. This system can also allow human to manually inspect the candidate. This manual inspection by using this system will help to verify the true detections as well as it will avoid false detection.

# CONCLUSION

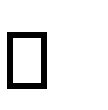
Eye tracking and proctoring using webcam and there can be many ways to solve this problem. In this project computer vision algorithm based solution is implemented. Computer vision algorithms can be successfully used for feature detection. Accuracy for feature detection. Accuracy for feature extraction depends on image quality and lighting conditions. With the development of Computer Vision being backed by tech giant Intel, Future improvements in the system are promising.

# SCOPE FOR FUTURE ENHANCEMET

We introduce an AI based auto proctored system for continuous online supervision, in case if there is unavailability of halls and during lockdown if the exam needs to be conducted, this system helps in it.

With the development of Computer Vision being backed by tech giant Intel, future improvements in the system are promising.

# REFERENCES

[1] I. E. Allen and J. Seaman. Grade change: Tracking online education in the united states, 2013. Babson Survey Research Group and Quahog Research Group, LLC. Retrieved on, 3(5), 2014. 

[2] Y. Atoum, S. Srivastava, and X. Liu. Automatic feeding control for dense aquaculture fish tanks. IEEE Signal Processing Letters, 22(8):1089–1093, 2015.

[3] D. L. Baggio. Enhanced human computer interface through webcam image processing library. Natural User Interface Group Summer of Code Application, pages 1–10, 2008.

[4] S. V. Bailey and S. V. Rice. A web search engine for sound effects. In Audio Eng. Society Convention 119. Audio Eng.

Society, 2005.

[5] C.-C. Chang and C.-J. Lin. Libsvm: A library for support vector machines. ACM T- TIST, 2(3):27, 2011.

[6] J. Chen and X. Liu. Transfer learning with one-class data. Pattern Recognition Letters, 37:32–40, 2014.

[7] C.D. McuMurrough, V. Metsis, J. Rich and F. Makedon. An eye tracking dataset for point of gaze detection. In ETRA,2012.

[8] C.H. Morimoto and M.R.Mimica. Eye gaze tracking techniques for interactive applications.CVIU,2005.

[9] Z.Zhu, Q. Ji. Eye gaze tracking under natural head movements. In CPVR,2005.

[10] X.Zhang,Y.Sugano, M.Fritz and A.Bulling. Appearance based gaze estimation in the wild. In CPVR,2015.

[11] K. Rayner. Eye movements in reading and information processing:20 years of research. Psyochological bulletin,1998.

[12] B.A. smith , Q. Yin, S. K. Feiner, and S. K. Nayar. Gaze locking: Passive eye contact detection for human object interaction

.In UIST, 2013.

[13] K. A. F. Mora, F. Monay, and J.-M. Odobez. Eyediap: A database for development and evaluation of gaze estimation algorithms from rgb and rgb-d cameras. ETRA 2014.

[14] S. Baluja and D. Pomerleau. Non-intrusive gaze tracking using artificial neural networks. Technical report , 1998.

SKNCOE. Pune-41, Dept. of E&TC Engineering

[15] J. Chen and Q. Ji 3d gaze estimation with a single camera without ir illumination. In ICPR , 2008.

[16] A. Duchowski. Eye Tracking Metholodogy: Theory and Practice. Springer Science & Business Mdedia,2007