**ABSTRACT:**

E- Health sector has evolved as an integration system of order entry systems, an administrative system, and departmental subsystems within a hospital. In this upcoming years technology has expanded its limit and now with IOT devices it became handy for the sectoral use. Some uses can be explained as implementing controlling system to those who cannot operate computers on their own, operating wheelchair without the hands, etc.

Our project is a designed for patients especially one with disabilities. The concept of this is to apply eye movement to control appliances along with voice control.

Pupil detection algorithm and eye blink detection algorithm is used.

This project is based on Artificial Intelligence augmented with IOT. Many algorithms have been used here along with image processing. The project includes registration of patients, storing their details into the system, and also computerized billing in the pharmacy, and labs. The main intention of introducing this system is to reduce the pain of the patients and to allow them easy access.

It provides many functionalities taking into consideration of Covid-19 spread. Looking for this spread of disease it is observed that patients suffer from lot ability to move. This project is basically a track outline of technology and health care. It lists out all the essential requirements for the client so as to provide expected quality. Different modules are designed for different clients using it.

**Objective:**

With the advent of technology and the move toward Internet of Things (IoT), a new technique has been developed by

combining an eye moment and voice embedded with IoT to create a platform for the use of patients.

**PROJECT DESCRIPTION:**

In this project algorithms such as pupil detection, eye blink detection are used. This works on machine learning, IOT and most of the part on Artificial Intelligence.

In this work, the eye blinks are detected without additional image processing, determining if the eyes are open or closed according to whether or not the pupil position is obtained, as proposed in . Then, the time while the eyes are closed indicates the blink type and, in case of forced blink, the action to perform. Although the accuracy of this method is highly depending on the pupil detection success, it does not require any additional processing time and this is a key factor for the low-performance proposed system.

the no blocking procedure used to detect forced eye blinks by means of open-close-open eye sequences and perform actions depending on its duration. The variable *run* is used to know if the eye is closed. Then the local system time, , updated every millisecond, is used to calculate the elapsed time without blocking the main thread (pupil detection and mouse events). The taction*n* indicates the time that the eye must remain closed to execute action , notice that the taction*n* time must to be quite larger than a natural blink.

The  indicates the detection hysteresis time and finally, to validate a forced blink, a minimum elapsed time (topen) is required between a close-open eye sequence.

When the eye is closed, the eyelashes fill a significant area of the image which could generate false pupil detections in specific users. Large eyelashes also are the main disturbing source when the user looks down. The threshold constraints defined in the valley pupil location algorithm avoid these false positives. shows a forced eye blink image and its pupil location result, where the yellow crosses are the absolute minimums for each row. In this case there is not a set of rows with a minimum valley that satisfies the fixed thresholds; hence, the algorithm result is successful.

An optimized algorithm to locate the pupil centroid in the acquired low-resolution sensor images ( pixels) is detailed. Although it has similar detection performances than state of the art algorithms, it can be easily integrated in a low-cost microcontroller without the help of any external memory. Moreover, a procedure to detect forced eye blinks is implemented in order to perform different mouse actions depending on the time that the eye remains closed (no pupil detection) rejecting natural blinks.

**EXISTING SYSTEM**

In the existing system, they are giving remote control type of IOT controlling device, but this system is not much help for all the paralyzed peoples because of handicapped people unable to move their hands freely. So, we go for Brain-computer interface based on the eye blink it moves that direction accordingly. It also has some limitation.

## ****PROPOSED SYSTEM****

In order to overcome the limitation in the existing system, we go for the alternative method is Eyeball based device control. In this method, the camera is focused on the eye by using OpenCV we need to find out the centroid of the eyeball. By tacking the eyeball movement, we can control the device accordingly. It very useful for Elderly and disables people who need guidance to live. In addition to this we have enabled voice feature too which makes the working efficiently.

**LANGUAGES AND TECHNOLOGY USED:**

* *Python programming:*
* *Artifical Intelligence:*
* *Machine Learning:*
* *IOT:*

**HOW IT WORKS:**

There are many steps to be followed while working of this device.

1st Step: Image Taking

At this stage, the connection with the camera is estab-

lished using the IP Camera

The ﬁlm captured by the object (camera) is

then received. A frame of it is separated and sent to the pre-

processor for preparation.

2nd Step: Image Preprocessing

In this step, the image obtained from the previous step

-which is in RGB space- is taken and sent to the processing

stage after applying the noise ﬁlter

3rd Step: Eye Identiﬁcation

At this stage, the system is designed to ﬁnd the exact co-

ordinates of the face in the image according to its apparent

features. In this algorithm, the regions of both eyes in the

image are searched by Six Segmented technique. First, a

rectangle containing six sub-sections is selected in the im-

age. Then, color contrast is measured in each section. The algorithm moves the rectangle throughout the image. Fi-

nally, the rectangle that has the speciﬁc characteristics rep-

resents the of eyes and eyebrows region. This technique

not only oﬀers the fastest face and eye recognition tech-

niques, but also responds well to the rotation of the head.

So, this method is not sensitive to the face rotation up to 20

degrees and the change of distance from the camera. After

identifying the rectangle containing both eyes in the im-

age, the ﬁrst three halves of the image containing the two-

eye image are separated from the image, and then the im-

age is broken up into two left and right eye sent separately

to the next step to identify the pupil.

4th Step: Pupil Detection

At this stage, the incoming image -which can be the

right or left eye- is identiﬁed by extracting the iris medium

and ﬁnally the pupil’s exact location using a combination

of two methods for removing the eyelids

The obtained point relative to the origin of

the image has a coordinate, which the algorithm sends it

as the output of this step to the statistical weighted aver-

aging step.

5th Step: Statistical Weighted Averaging

At this stage, the output of processing each frame pro-

duces the coordinate matrix of a trace as four numbers (xr,

yr, xl, yl). After removing all zero points resulting from non-

recognized points on the face, each value is rounded to

two decimal places. Then, two values with the highest fre-

quency of weighted average are selected.

These steps are repeated for each vari-

able, their values are calculated and will be sent to the cali-

bration or tracking step (each applicable) along with non-

zero coordinates.

6th Step: Tracked Data Usage

At this stage, depending on the mode of the device (re-

ceive command or calibration mode) input data is directed

to one of the following paths.

Calibration

In this case, the obtained points can be cited if the quo-

rum is reached after the 5th step (statistical weighted aver-

aging) and can be recorded as a valid data of that point for

the person. After this, the algorithm announces the next

point for calibration. This cycle is repeated for all eleven

points and a database is created for the person to compare

and receive the command

Receive Command

In this step, the coordinates obtained from the previ-

ous step are compared with the coordinates of the refer-

ence points in the individual calibration ﬁle for each eye,

and the distance from all points of the individual proﬁle

is obtained. Then, the closest reference point for the coor-

dinates obtained is selected for each eye. If the responses

of both eyes are identical, that point is selected as the ﬁ-

nal output. Otherwise, the algorithm compares the diﬀer-

ence in the coordinates obtained with the selected point by each eye, and the ﬁnal response is the point with smaller

distance from the reference point. Finally, the result is sent

for the command’s conﬁrmation step

Command Conﬁrmation

When the required command is selected, the patient

can conﬁrm the command by looking to the “conﬁrm”

point at the middle of the selection pane. This step is nec-

essary to prevent the implementation of an incorrect com-

mand. When conﬁrmed, the corresponding command will be sent for execution. Otherwise, the command is canceled

and the program returns to the beginning waiting for the

next command