iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 6 **Issue:** IV **Month of publication:** April 2018

DOI: http://doi.org/10.22214/ijraset.2018.4552

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 6.887
Volume 6 Issue IV, April 2018- Available at www.ijraset.com

Eye Ball Movement based Cursor using Raspberry PI

T.H. Feiroz khan[1], T. Gopikrishna[2], N. Dayakar reddy[3], T. Sai Dheeraj[4], Shaik Mohammed Shahid[5]
[2, 3, 4, 5]*B. Tech (IV) Year Student , Department of Computer Science and Engineering SRM Institute of Science and Technology, Chennai, India*
[1,]*Assistant Professor, Department of Computer Science and Technology SRM Institute of Science and Technology, Chennai, India*

Abstract: ***Some peoples cannot able to operate computers because of an illness. The idea of eye controls of great use to not only the future of natural input but more importantly the handicapped and disabled. Moreover, implementing a controlling system in it enables them to operate computer without the help of another person. It is more helpful to handicapped peoples. Those are need to operate computers without hand this one is most useful those can operate cursor by movement of eye. In this paper Camera is capturing the image of eye movement. First detect pupil center position of eye. Then the different variation on pupil position gets different movement of cursor. What all these applications have in common is that the use of personal computers is mostly based on the input method via keyboard and mouse. While this is not a problem for a healthy individual, this may be an insurmountable bound for people with limited freedom of movement of their limbs. In these cases it would be preferable to use input methods which are based on more abilities of the region such as eye movements. To enable such substitute input methods a system was made which follows a low-price approach to control a mouse cursor on a computer system. The eye tracker is based on images recorded by a mutated webcam to acquire the eye movements. These eye movements are then graphed to a computer screen to position a mouse cursor accordingly. The movement of mouse by automatically adjusting the position where of eyesight. Camera is used to capture the image of eye movement.***

I. INTRODUCTION

Nowadays personal computer systems are carrying a huge part in our everyday lives as they are used in areas such as work, education and enjoyment. What all these applications have in common is that the use of personal computers is mostly based on the input method via keyboard and mouse. While this is not a problem for a healthy individual, this may be an insurmountable bound for people with limited freedom of movement of their limbs. In these cases, it would be preferable to use input methods which are based on more abilities of the region such as eye movements. To enable such substitute input methods a system was made which follows a low-price approach to control a mouse cursor on a computer system. The eye tracker is based on images recorded by a mutated webcam to acquire the eye movements.

These eye movements are then graphed to a computer screen to position a mouse cursor accordingly. The movement of mouse by automatically adjusting the position where of eyesight. Camera is used to capture the image of eye movement. Recently there has been a growing interest in developing natural interaction between human and computer. Several studies for human-computer interaction in universal computing are introduced. The vision-based interface technique extracts motion information without any high cost equipment from an input video image. Thus, vision-based approach is taken into account an effective technique to develop human computer interface systems. For vision-based human computer interaction, eye tracking is a hot issue.

Eye tracking research is distinguished by the emergency of interactive applications. However, to develop a vision-based multimodal human computer interface system, an eye tracking and their recognition is done. Real time eye input has been used most frequently for disabled users, who can use only their eyes for input. There are different reasons for which people need an artificial of locomotion such as a virtual keyboard. The number of people, who need to move around with the help of some article means, because of an illness. Moreover implementing a controlling system in it enables them to move without the help of another person is very helpful. The idea of eye controls of great use to not only the future of natural input but more importantly the handicapped and disabled. Camera is capturing the image of eye movement. First detect pupil center position of eye. Then the different variation on pupil position get different command set for

II. LITERATURE SURVEY

In the paper, there were many techniques followed Mat lab based control, Sensors based control cursor, Eye movement controlled wheel chair by using Raspberry pi is existing one that controls the wheel chair by monitoring eye movement.

A. Tracking Students' Cognitive Processes During Program Debugging—An Eye-Movement Approach

This study explores students' cognitive processes while debugging programs by using an eye tracker. Students' eye movements during debugging were recorded by an eye tracker to investigate whether and how high- and low-performance students act differently during debugging. Thirty-eight computer science undergraduates were asked to debug two C programs. The path of students' gaze while following program codes was subjected to sequential analysis to reveal significant sequences of areas examined. These significant gaze path sequences were then compared to those of students with different debugging performances. The results show that, when debugging, high-performance students traced programs in a more logical manner, whereas low-performance students tended to stick to a line-by-line sequence and were unable to quickly derive the program's higher-level logic. Low-performance students also often jumped directly to certain suspected statements to find bugs, without following the program's logic. They also often needed to trace back to prior statements to recall information and spent more time on manual computation. Based on the research results, adaptive instructional strategies and materials can be developed for students of different performance levels, to improve associated cognitive activities during debugging, which can foster learning during debugging and programming.

B. Driver Distraction Detection Using Semi-Supervised Machine Learning

Real-time driver distraction detection is the core to many distraction countermeasures and fundamental for constructing a driver-centred driver assistance system. While data driven methods demonstrate promising detection performance, a particular challenge is how to reduce the considerable cost for collecting labelled data. This paper explored semi-supervised methods for driver distraction detection in real driving conditions to alleviate the cost of labelling training data. Laplacian support vector machine and semi-supervised extreme learning machine were evaluated using eye and head movements to classify two driver states: attentive and cognitively distracted. With the additional unlabelled data, the semi-supervised learning methods improved the detection performance (*G*-mean) by 0.0245, on average, over all subjects, as compared with the traditional supervised methods. As unlabelled training data can be collected from drivers' naturalistic driving records with little extra resource, semi-supervised methods, which utilize both labelled and unlabelled data, can enhance the efficiency of model development in terms of time and cost.

C. A Device Controlled Using eye Movement

Research carried out in this technical paper suggests an application of the widely implemented eye tracking techniques. Traditionally, HCI uses mouse, keyboard as an input device but this paper presents hand free interface between computer and human. Here providing a novel idea to control computer mouse cursor using human eyes movement. It controls mouse moving by automatically affecting the position where eyesight focused. The project mainly comprises of three sections namely Image Capture, Image Processing, and Cursor Control. After capturing image from webcam, the shape of pupil is recognized using Hough Transform and the center coordinate help to determine the exact point on the screen where the user is looking also coordinate will instruct the computer mouse to move specific location. This system is very helpful for solving the HMI problems of the disabled so that it can provide a way for them to communicate with the outside world.

III. PROPOSED SYSTEM

This system is totally independent system, and all the module will work independent each other. The individual components used are Raspberry pi, camera, SD card and monitor. The Raspberry Pi gets the first half of its name from a long-standing tradition of using fruit to name new computing systems— from classic microcomputers like the Acorn, Apricot and Tangerine to more recognizably modern brands including Apple and BlackBerry—but the second half comes courtesy of the Python programming language. For capturing the image normal web camera is used in our system. Moreover, High resolution HD web camera can be used but it increases the image memory size in MB. So that system cannot read the image and process efficiently as per requirements, and it will also increase the processing time**.** This system is coming under real time data acquisition, data processing and controlling cursor. To detect the exact eye pupil location is very challenging. A new image processing technique used for eye pupil center detection and tracking, which works based on open computer vision (OpenCV) library. Most of coding part done with the help of OpenCV library. There is a several algorithms like Haar cascade, Hough transform, edge detection are available for different application. To connect the raspberry pi board to desktop Putty software is used. Python language is used for codding, which is user friendly and helpful to resolve the error efficiently. OpenCV 3.0.0 library with python is used in this system.

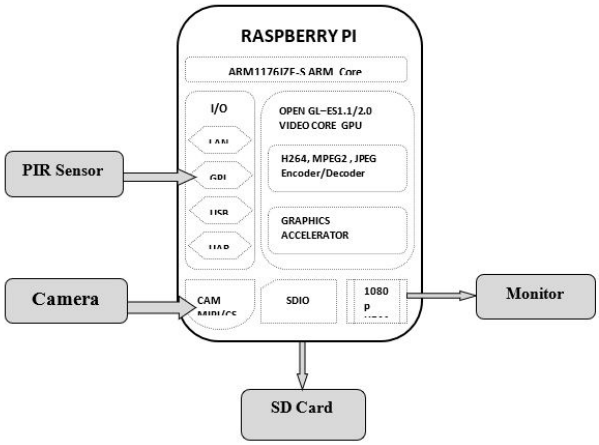
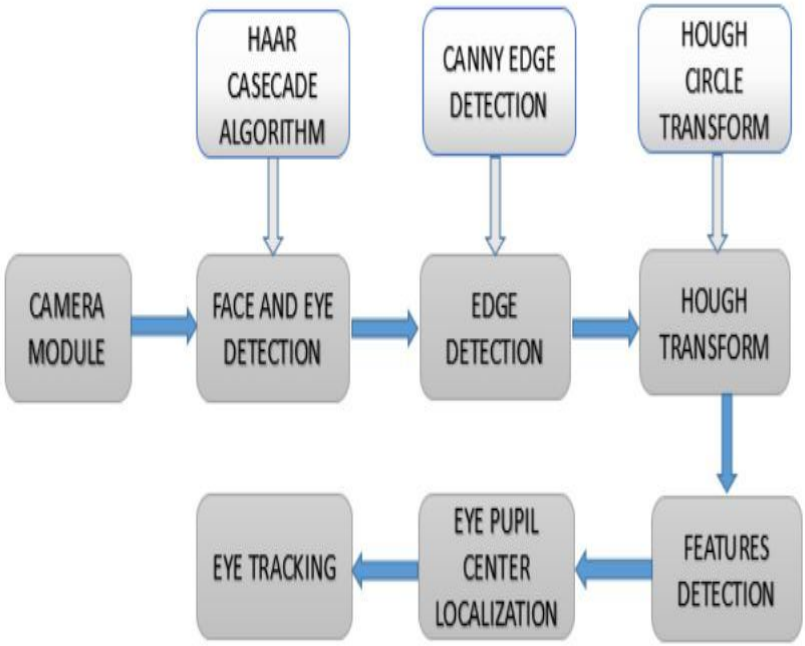


Fig.1 Block Diagram

IV. PROPOSED METHODOLOGY

The principle of this system is eye pupil detection and eye tracking based on computer vision technology. A new algorithm introduced for detecting the eye pupil location by Image processing. In this technique several stages used to find out the movement of eye, such as Face detection and Eye detection, color conversion, Edge detection, Hough Transformed, motion detection and object tracking. During initial stage the system acquired the captured Images by USB Web camera. The first direction is to detect the user Face accurately. If there is multiple faces are presented it will display the individuals and also showing the run time error. A system indicates and represents the face of user in a specific area of image. After that system will performed the several operations of image processing to track the Eye pupil. The figure 3 represents the complete methodology of proposed implementation. Here it will give the step by step information of the system working



ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 6.887
Volume 6 Issue IV, April 2018- Available at www.ijraset.com

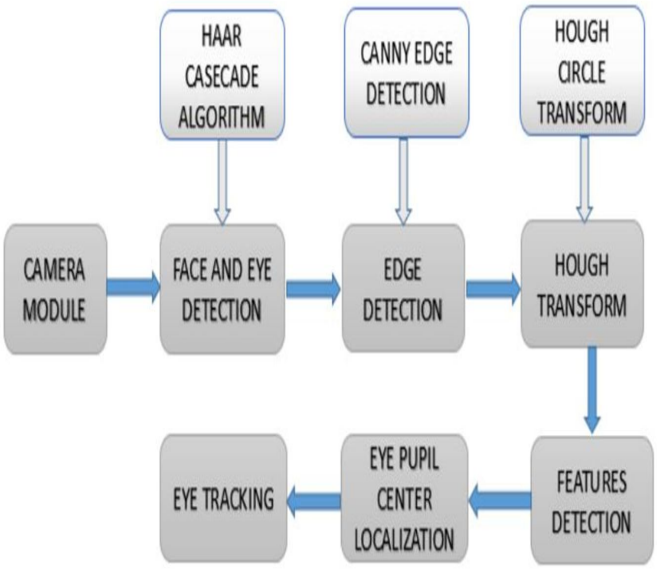


Fig.2 system process design

First camera module will start to capture the images. For the face detection Haar cascade algorithm is used. After detection of proper face, it will be trying to detect the eye inside the face region of interest. And again, Haar cascade algorithm is used like as face detection to detect eye. It will draw the rectangular box over the Eye. Now, the main target is to detects the eye pupil and define its center points. There is several image processing operations performed in system, such as blur Image, color conversion, thresholding, filtering, edge detection and Hough transform is used. For circle detection Hough transform method is used. By using the USB webcam allowed to capture the images on raspberry. And Image Processing based all OpenCV library are installed in raspberry pi memory. There it will process and working without any processing delay

The system will crop the eye region of interest initially and it will detect all possible circle presented on that particular area. Than it will successfully detects the eye ball. After that corner detection method, we applied for eyes region of interest, and find out the corners. Where average of both two point defined its Center point. Now we measure distance between the Center point and eye circle Center point using coordinates system logic. According to the eye pupil movements, distance will be varied. A minimum distance indicates the eye pupil presented in left, and maximum values indicates the eye moved on right. And if there is no movements of the eye, than it concludes eye is in the middle position. Than the commands applied for all operation, when eye movement is left, the cursor moves left side will run. And when the eye moved is right the cursor on the screen is move right. If eye is blinked once, the cursor selects the content displayed on the screen . A system started with capturing images continuously by camera. And captured images processed in Raspbian system. USB camera is used to capture the image at high pixel rates. In idle condition the eye will be consider open.

V. IMPLEMENTATION AND SYSTEM DESCRIPTION

This system implementation is working based on real time data acquisition operating system. The low power consumption Raspberry pi board computer is used. Which provides well enough in/out pins, USB ports, UART, PWM, HDMI port and Ethernet adapter port for connecting it through internet via wired or wireless connection. The raspberry pi have a 512 MB RAM and controlled based on ARM architecture. Also, raspberry pi have up to 32 GB external memory capably. Camera is capturing the image of eye movement. USB 2.0 makes USB Cameras ideal for many imaging applications. USB Camera will be interface with raspberry pi. Raspberry pi will be use SD card, then the install Raspbian OS and open CV on raspberry pi. Fist image will be capture by USB Camera. Focus on eye in image and detect the Centre position of pupil by open CV code. Take the center position value of pupil as reference, and then the next the different value of X, Y coordinates will be set for particular command.

A. Raspberry pi installatio

A step is followed to install its own operation system in micro SD card. To boot a Raspbian image file win32 disk imager software used. While putting a bootable memory device on raspberry pi board, then it can access the Raspbian operating system directly without rebooting.

B. System Algorithms

In our system open computer vision (OpenCV) free access library algorithm used for Image processing. The OpenCV library play a very important role, and it gives the knowledge of Image processing. A novel algorithm used for system execution and perform the operation

1) *Creating The Datasets:* The datasets are the images of the human being. This kind creating of datasets trains the database to easily identify the face and then detect the pupils of the eyes.

The creating of datasets is shown in fig no. 3

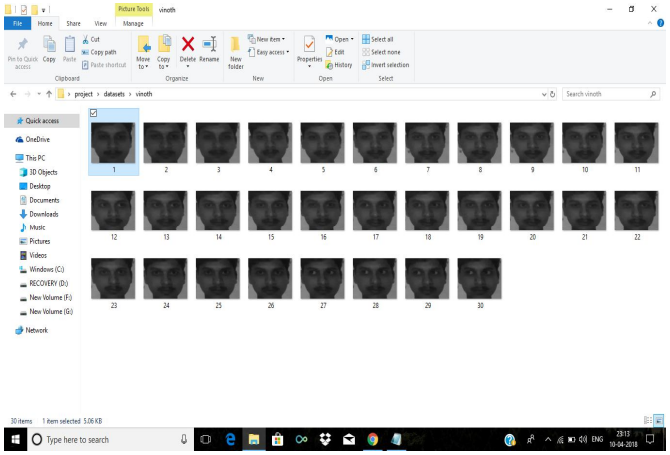


Fig .3 Creating datasets

2) *Run the python IDLE Module:* Move the python code file to the folder which contains the datasets then open the python IDLE file and run the module. This is shown in the fig no.4

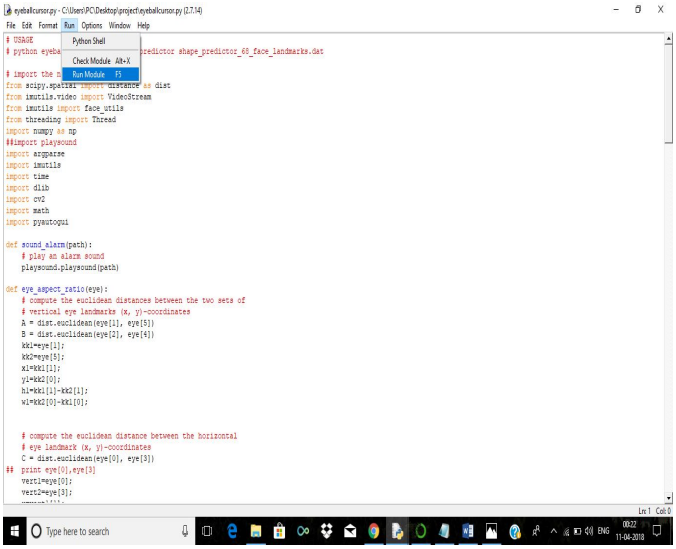


Fig .4 Run the Python IDLE module.

VI. RESULT AND DISCUSSION

The system acquired the resulted data of image processing and based on the Eye pupil center value signal send to the input mouse which controls the cursor mechanism. The selecting of any content that is visible on the computer screen can be selected by blinking of both the eyes. The detection and movement of the cursor is shown in the fig no.5

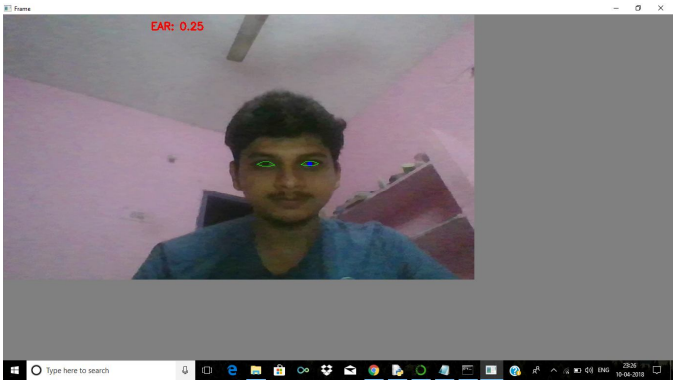


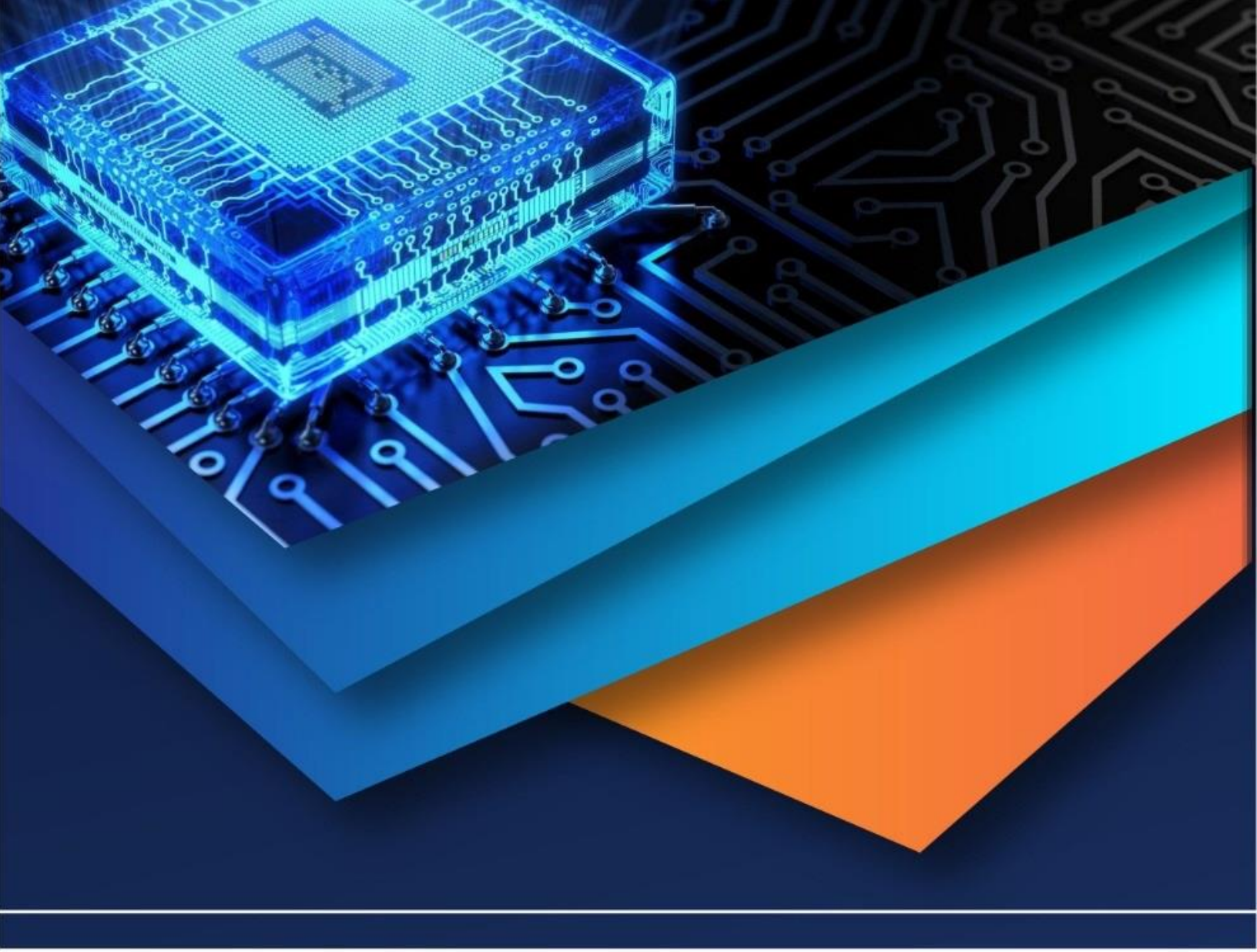
Fig. 5 Detection of eye pupil

VII. CONCLUSION

Traditional research has investigated the differences between debugging experts and novices in an effort to understand how to develop better pedagogical methodologies and tools. However, such research still requires solid and objective evidence to establish the authenticity of its findings. In this research, the experimental results provide objective eye-tracking evidence that confirms the hypotheses made based on the findings of existing research: Most students recognize beacons and pay more attention to these areas when debugging. Low-performance students tend to debug programs aimlessly, whereas high-performance students debug programs in a more logical manner. The major differences between high- and low-performance students in terms of their cognitive processes lie in the former's ability to plan during the debugging process. High-performance students tend to organize the code into chunks, and their comprehension and debugging strategies are based on prior knowledge coupled with their ability to identify problems. Low-performance students, on the other hand, remain fixed on the syntactic details, ultimately failing to build the appropriate mental models for debugging. These conclusions were drawn after examining the differences between high- and low-performance students by applying statistical tests: the -test for spatial data, and sequential analysis for temporal data. Only significant statistical results have been reported in the conclusions, guaranteeing the conclusion validity. Previous research has revealed a relationship between working memory capacity and the cognitive activities related to debugging with regard to mental arithmetic, short-term memory, logical thinking, and problem solving. This study confirms this correlation. However, further study is needed to better understand the relationship, and such studies require firmer evidence. By exploring the differences between high- and low-performance students with regard to their visual attention and gaze paths during debugging, a strong correlation was found between debugging skills and the associated cognitive activities, demonstrating the importance of the instructor's assistance in developing these cognitive abilities while teaching debugging skills. Adaptive instructional strategies and media can be designed based on these research findings. For example, the learning of scaffolding tools might be designed to help students plan debugging
tasks, recall prior information, split the code into meaningful chunks, process computations, and trace the program in a logical manner.

REFERNCES

[1] D. N. Perkins, C. Hancock, R. Hobbs, F. Martin, and R. Simmons, “Conditions of learning in novice programmers,” J. Educ. Comput. Res., vol. 2, pp. 37–55, 1986
[2] J. A. Villalobos and N. A. Calderón, “Developing programming skills by using interactive learning objects,” in Proc. ITiCSE, 2009, pp. 151–155
[3] E. Verdú et al., “A distributed system for learning programming on-line,” Computer. Educ., vol. 58, no. 1, pp. 1–10, 2012
[4] S. Fitzgerald et al., “Debugging: Finding, fixing and flailing, a multi institutional study of novice debuggers,” Comput. Sci. Educ., vol. 18, pp. 93–116, 200
[5] T. Putnam, D. Sleeman, J. A. Baxter, and L. Kuspa, “A summary of misconceptions of high school basic programmers,” J. Educ. Comput. Res., vol. 2, pp. 459–472, 198
[6] S. Xu and V. Rajlich, “Cognitive process during program debugging,” in Proc. 3rd IEEE ICCI, 2004, pp. 176–182
[7] L. Murphy et al., “Debugging: The good, the bad, and the quirky—A qualitative analysis of novices' strategies,” SIGCSE Bull., vol. 40, no. 1, pp. 163–167, 2008
[8] R. E. Brooks, “Towards a theory of the cognitive processes in computer programming,” Int. J. Man-Mach. Studies, vol. 9, pp. 737–751, 1977
[9] E. Soloway, J. Bonar, and K. Ehrlich, “Cognitive strategies and looping constructs: An empirical study,” Commun. ACM, vol. 26, no. 11, pp. 853–860, 1983
[10] V. Renumol, “Classification of cognitive difficulties of students to learn computer programming,” Indian Inst. Technol. India, 2009
[11] V. G. Renumol, D. Janakiram, and S. Jayaprakash, “Identification of cognitive processes of effective and ineffective students during computer programming,” Trans. Comput. Educ., vol. 10, no. 3, pp. 1–21,2010.



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY