IARJSET

ISSN (O) 2393-8021, ISSN (P) 2394-1588

International Advanced Research Journal in Science, Engineering and Technology

Impact Factor 8.066 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 11, Issue 5, May 2024

DOI: 10.17148/IARJSET.2024.11516

Mouse Cursor Control Using EyeMovements

Mithali M Jahagirdar[1], Kunchapu Meghana[2], Kavitha H C[3], Mujasum B[4], Shreelakshmi C M[5]

UG Student, Department of CSE, GSSSIETW, Mysuru, India[1-4]

Assistant Professor, Department of CSE, GSSSIETW, Mysuru, India[5]

Abstract: Recent advancements in technology have played a pivotal role in enhancing various aspects of human life. Recognizing the valuable contributions of individuals with disabilities to society, it is crucial to facilitate their effective engagement through appropriate platforms. A prototype has been devised to operate household appliances by tracking eye movements for cursor manipulation. Through capturing eye movements via a camera and pinpointing the center position of the pupil, a range of commands for a virtual keyboard are generated. These commands enable interaction with the virtual keyboard through a motor driver, allowing the prototype to maneuver in different directions based on the chosen command. The contemporary computing industry is dedicated to advancing hands-free computing to aid quadriplegics. This article proposes a Human-Computer Interaction (HCI) system specifically designed for individuals facing amputations or hand mobility impairments. This system employs eye-based interfaces to translate various eye movements like blinking, staring, and squinting into actions of the mouse cursor. The implementation of this system necessitates the utilization of Python, OpenCV, NumPy, and other software tools for face recognition utilizing a standard webcam. Techniques such as the HOG feature, linear classifiers, and the sliding window method can construct a face detector devoid of additional equipment or sensors, ensuring a hands-free experience. The paper introduces an innovative algorithm for managing the movement of a computer screen cursor through iris movements. By accurately detecting the iris position within the eye and mapping it to specific locations on the computer screen, the algorithm empowers physically challenged individuals to control cursor movements in all directions. Moreover, the algorithm facilitates actions such as opening and closing folders, files, or applications through a clicking mechanism.

Keywords: Technology advancements, Technology advancements, Eye tracking, Hands-free computing.

I. INTRODUCTION

In the realm of technology today, the prevalent method of navigating screens via computer mice or finger gestures has gained widespread use. This technology tracks and interprets mouse or finger movements to control cursor motion. However, individuals referred to as "amputees," who lack arm function, are unable to benefit from this technology. Therefore, if facial movements, particularly eye and facial expressions, could be monitored and gaze direction detected, it would offer them cursor control freedom. The creation of an 'eye-tracking mouse' would prove highly advantageous for amputees. Despite the simplification of cognitive interactions with computers through Graphic User Interfaces (GUIs), the physical demands remain significant. GUI usage necessitates fine motor skills for manipulating input devices like mice or trackballs, essential for positioning the cursor and executing commands effectively. Unfortunately, statistics reveal that a significant number of individuals in the United States, ranging from 250,000 to 400,000, suffer from spinal cord injuries or dysfunction, many lacking the motor skills required to operate input devices efficiently. Consequently, their interaction with GUI-based computers is severely restricted. Similarly, professionals like surgeons, whose hands are occupied with critical tasks, encounter similar limitations. Acknowledging these challenges, our team has focused on developing a hands-free cursor control system, eliminating the necessity for coordinated hand movements. Eye tracking technology has emerged as a valuable asset in the medical field, revolutionizing both research and clinical applications by offering profound insights into human behavior, cognition, and visual perception. Through precise measurement and analysis of eye movements, eye tracking empowers researchers and medical practitioners to unveil intricate patterns of visual attention, gaze fixation, and cognitive processes.

The approach outlined in this paper stands out due to its departure from conventional methods that rely on electrodes, infrared technology, or other light sources for eye tracking. Instead, the sole hardware requirement is a standard PC or laptop equipped with a webcam, rendering it practical and accessible. Through capturing successive snapshots of the user via the webcam, the program efficiently processes these frames individually at a high speed and compares the iris movement in each frame relative to the initial frame. The frames undergo several processing stages before the eyes can be accurately tracked. Upon obtaining the processed image, the program calculates the iris movement, prompting the cursor on the screen to correspondingly relocate.

IARJSET ISSN (O) 2393-8021, ISSN (P) 2394-1588

International Advanced Research Journal in Science, Engineering and Technology

Impact Factor 8.066 ⵋ Peer-reviewed & Refereed journal ⵋ Vol. 11, Issue 5, May 2024

DOI: 10.17148/IARJSET.2024.11516

II. PROPOSED SYSTEM

This project revolves around the precise prediction of facial landmarks on a given face, offering a multitude of potential applications. These landmarks enable tasks ranging from detecting eye blinks in videos to predicting the subject's emotions, showcasing the vast and captivating possibilities of facial landmark analysis.

Utilizing Dlib's prebuilt model, which not only swiftly detects faces but also accurately predicts 68 2D facial landmarks, proves highly advantageous.

Since the project hinges on facial feature detection to control the cursor, the initial phase entails accessing and activating the webcam. Subsequently, the program proceeds to extract individual frames from the video feed. These frames then undergo a series of processes to detect features, which are then utilized to manipulate the cursor. This iterative process repeats for each frame within a loop.

The outlined system entails the following functionalities:

- Mouth opening
- Right eye winking
- Left eye winking
- Eye squinting
- Head movements (Pitch and Yaw)

The advantages offered by this system are:

- Facilitating hands-free manipulation of the mouse cursor.
- Offering support to individuals with disabilities.
- Governing the mouse pointer through eye movements and facial expressions.
- Emulating diverse mouse actions, including left-click, right-click, and double-click, utilizing eye and facial cues.

III. METHODOLOGY

In the context of mouse clicking, an eye blink serves as the action trigger. When the user decides to initiate a selected action, such as clicking, the eye blink triggers the mouse pointer to execute the click event. The user positions themselves in front of their personal computer or laptop screen, with a small video camera placed above the display to monitor their eye movements.

The computer continuously analyzes the video feed of the user's eyes to determine where their gaze is focused, without requiring any attachments to the user's head or body. To select a key, the user gazes at it for a specific duration, and to press a key, the user simply blinks their eye. Notably, there is no need for a calibration procedure in this system, as it relies solely on eye data without requiring any external hardware.

In optimizing the eye-tracking system, we employ a multi-step approach to enhance efficiency. Initially, a rapid face detection algorithm identifies the face within the webcam's image frame, defining the area of interest. Subsequent eye detection algorithms efficiently isolate and track the eyes, focusing on movement in one eye for faster processing.

Utilizing simplified intensity-based methods that capitalize on the iris's lower intensity, swift iris detection is achieved. By referencing the corners of the eye, precise tracking of iris movement captures changes in the user's focus. This movement is promptly translated into cursor position on a graphical user interface, facilitating seamless interaction. These optimizations ensure real-time performance and accurate gaze mapping, enhancing the system's usability and efficiency for practical applications.

IARJSET
ISSN (O) 2393-8021, ISSN (P) 2394-1588
International Advanced Research Journal in Science, Engineering and Technology
Impact Factor 8.066 ⫶ Peer-reviewed & Refereed journal ⫶ Vol. 11, Issue 5, May 2024
DOI: 10.17148/IARJSET.2024.11516

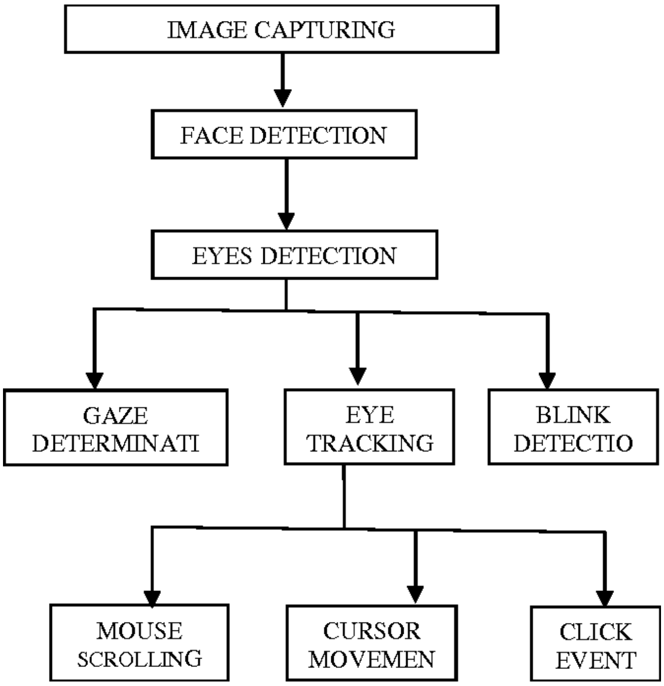

Figure 1: Flow Diagram

Pupil Tracking: Pupil tracking, a gaze detection technique often integrated with other methods, recognizes that the eye serves as more than a tool for swiftly moving cursors. Eye movement input surpasses the speed of other input factors. Typically, before engaging any mechanical pointing system, users direct their gaze towards their intended destination. Eye movement input is derived from the individual's pupil. For instance, if a person focuses on a central mouse pointer, that point becomes the input location, initiating gaze tracking. The cursor then follows the direction of the person's eye movement, halting when the eye returns to its initial position. Various detection methods involve multiple stages, including facial and eye positioning from different perspectives.

Regression Approach: This approach minimizes disparities between anticipated and actual eye positions. The characteristic-based method involves facial feature extraction for face identification. Initially, it categorizes facial and non-facial regions, aiming to surpass our limited understanding of facial structures. This method comprises several phases and encompasses images of numerous faces.

Horizontal Pupil Movement: Circular artifacts enable horizontal eye pupil movement. When the pupil shifts leftward, the mouse pointer also moves left, and likewise for rightward movement.

Vertical Pupil Movement: Vertical eye pupil movement utilizes pupil scaling. Eyes are slightly half-closed when gazing downward, a phenomenon exploited to guide the mouse pointer from top to bottom.

IV. SYSTEM ARCHITECTURE

Within the system architecture, we've integrated face detection, face tracking, eye detection, and real-time interpretation of eye blinks to facilitate a non-intrusive human-computer interface. The traditional mouse-based computer interaction is substituted with human eye movements. This method aids individuals with paralysis or physical disabilities, particularly those without the use of hands, to compute effectively and with ease.

Initially, the camera captures an image and utilizes OpenCV code for face and eye detection, determining the center position of the human face. This center position serves as a reference for the user to control the cursor by moving left and right. The system's workflow involves face recognition, followed by eye detection, and then interpreting eye blinks to select phrases on the keyboard displayed on the screen.

DOI: 10.17148/IARJSET.2024.11516

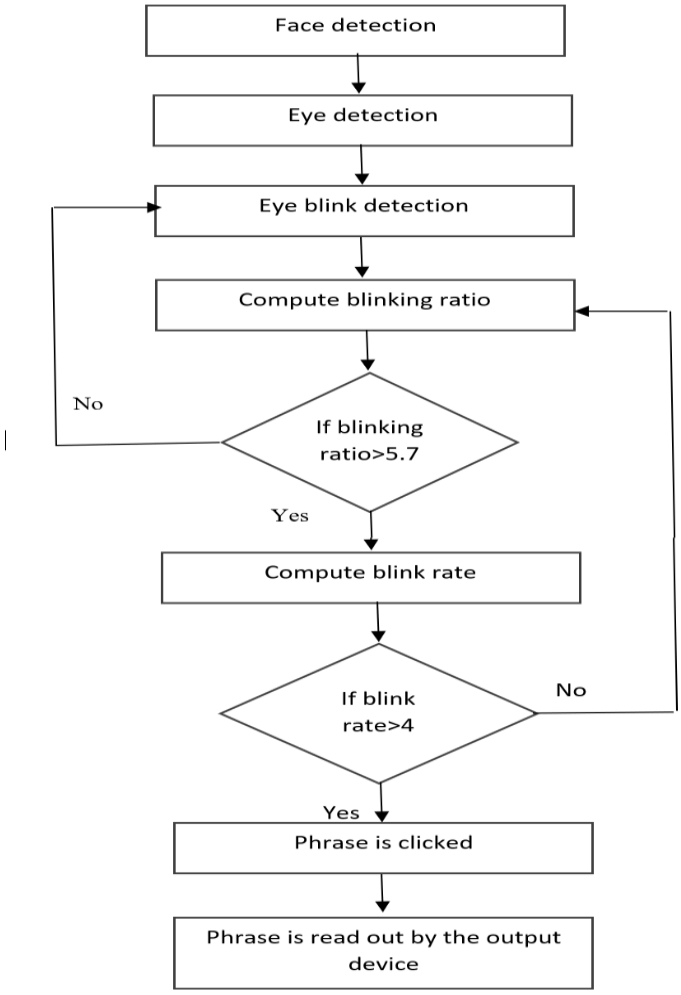

Figure 2: System Architecture

The system establishes a protocol where a blink ratio exceeding 5.7 is computed as a blink, while a ratio below 5.7 is disregarded. Based on the blink rate, if it surpasses 4, the pointer clicks the designated phrase, and audio assistance is provided. Conversely, if the rate falls below 4, further computation and verification are conducted for output. This user-friendly operation, particularly the blink detection feature, enables individuals with physical disabilities to easily engage with the system. Facial landmark features are employed for face detection with the assistance of dlib.

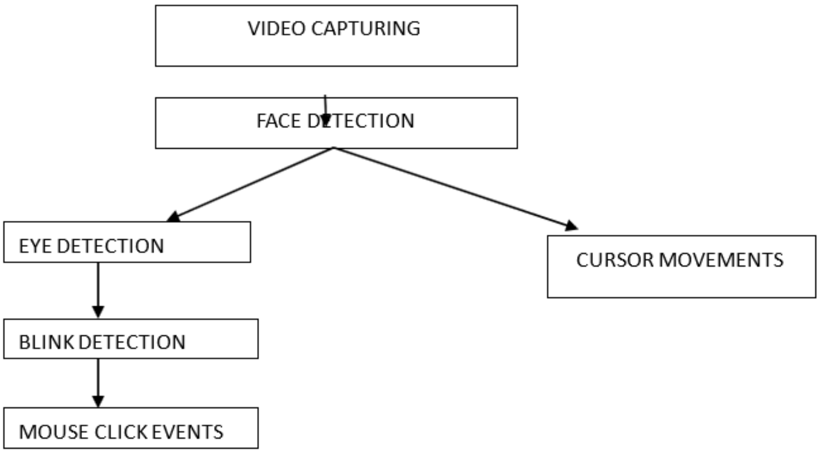

Figure 3: Low Level Design (LLD)

IARJSET

ISSN (O) 2393-8021, ISSN (P) 2394-1588

International Advanced Research Journal in Science, Engineering and Technology

Impact Factor 8.066 ⁘ Peer-reviewed & Refereed journal ⁘ Vol. 11, Issue 5, May 2024

DOI: 10.17148/IARJSET.2024.11516

Low-level design (LLD) is a component-level design process that follows a step-by- step refinement process. This process can be used for designing data structures, required software architecture, source code and ultimately, performance algorithms.

In this paper, we have designed a system fig 4.6 which can be easily controlled by the paralyzed people. This system provides the speaking power without using mouth. Users can able to speak what they want through their eye blink. The user interface is very easy to use for all age groups from children to eldery person. The constructed system takes live video taken using webcam as input. From the input, the system will detect face and eye using facial landmark structure. The system will be built on several parts as deting face, eyes, eye blinks, virtual interface on screen, select the phrase button and finally read the phrase using eye blinking with the help of a speaker. The system architecture is as follows:

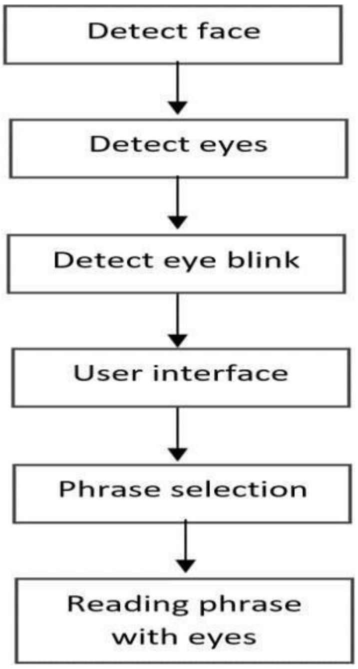

Figure 4: Architecture for paralyzed people

Face Detection: Facial recognition technology involves identifying human faces in images or videos, commonly implemented using tools such as OpenCV and dlib, employing diverse detection methods. In this context, the detector utilizes a combination of classic Histogram of Oriented Gradients (HOG) features and a linear classifier. Within dlib, a facial landmarks detector integrated to identify facial features such as eyes, ears, nose, and more.

Eye Detection: After detecting the face, eye region is detected with the help of facial landmark features. Using the face landmarks datasets, we can point out 68 landmarks on the face each landmark is assigned with an index. Using these indices, the desired region of the face is detected. Point index for two eyes:

- left eye: - (37, 38, 39, 40, 41, 42)
- right eye: - (43, 44, 45, 46, 47, 48)

After extracting eye region, it is processed for detecting eye blinks. The eye region detection is done at the initial stage of the system.

Eye Blink Detection: By precisely delineating the eye region with two lines – one horizontal and the other vertical – we can identify blinks. Blinking entails the temporary closure of the eyes, accompanied by eyelid movement, constituting a swift and natural process.
Determining an eye blink involves observing specific indicators:

- The absence of the eyeball and closed eyelids indicate a blink.
- When the upper and lower eyelids are connected, it signifies a blink.

If these conditions persist for a duration of approximately 0.3 to 0.4 seconds, it is considered a blink; anything longer suggests closed eyes.

IARJSET

ISSN (O) 2393-8021, ISSN (P) 2394-1588

International Advanced Research Journal in Science, Engineering and Technology

Impact Factor 8.066 ⁒ Peer-reviewed & Refereed journal ⁒ Vol. 11, Issue 5, May 2024

DOI: 10.17148/IARJSET.2024.11516

When the eye is open, both vertical and horizontal lines are nearly identical, whereas in a closed eye, the vertical line diminishes significantly or disappears entirely. Using the horizontal line as a reference point, a ratio is calculated relative to the vertical line. Establishing a threshold value, if the ratio exceeds this value, the eye is considered closed; otherwise, it is deemed open.

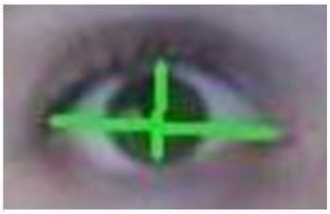

Figure 5: Opened eye and Closed eye

Controlling the mouse pointer: Pyautogui is a GUI automation module. This module is used to control the mouse and keyboard events programmatically. In our system, scrolling of the mouse pointer across the phrases is achieved through moveTo() and moveRel() methods by giving the relevant x, y location of the screen as arguments.

V. FUTURE SCOPE

The system design empowers individuals with paralysis to engage with computers devoid of physical keyboards or mice. Leveraging object detection and image processing via OpenCV, it accomplishes accurate keyboard and mouse functionalities, offering advantages to those with limb disabilities. Although cost-efficient, its performance under low light conditions raises apprehensions, indicating potential enhancements with the integration of higher resolution cameras. This figure depicts the calculation of MAR and EAR after the detection of face.

VI. RESULTS

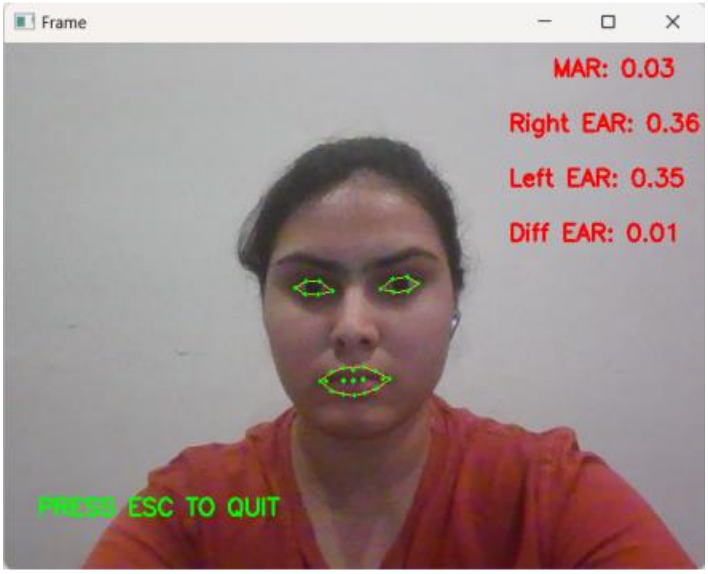

Figure 6: Initial Calculation of EAR and MAR

This figure depicts the initial face detection of face and then the detection of eye and mouth with the help of EAR and MAR. Haar-Cascade algorithm is used for face detection and eyes tracking which is followed by EAR to detect drowsiness.

IARJSET ISSN (O) 2393-8021, ISSN (P) 2394-1588

International Advanced Research Journal in Science, Engineering and Technology
Impact Factor 8.066 ⵋ Peer-reviewed & Refereed journal ⵋ Vol. 11, Issue 5, May 2024
DOI: 10.17148/IARJSET.2024.11516

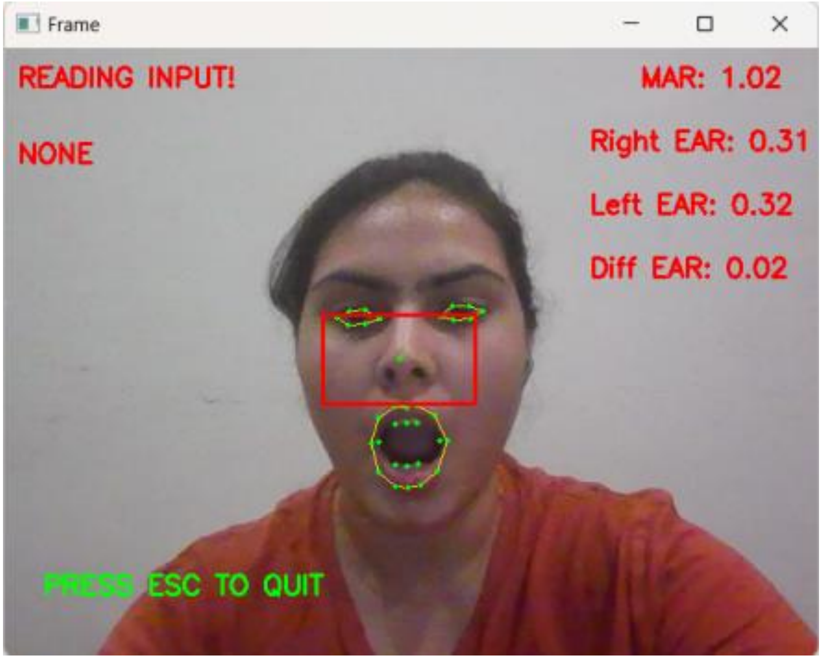

Figure 7: Activation of Mouse

The Figure 6 shows the activation of mouse with the help of mouth. Opening of mouth leads to activation of mouth and this is calculated with the help of MAR.

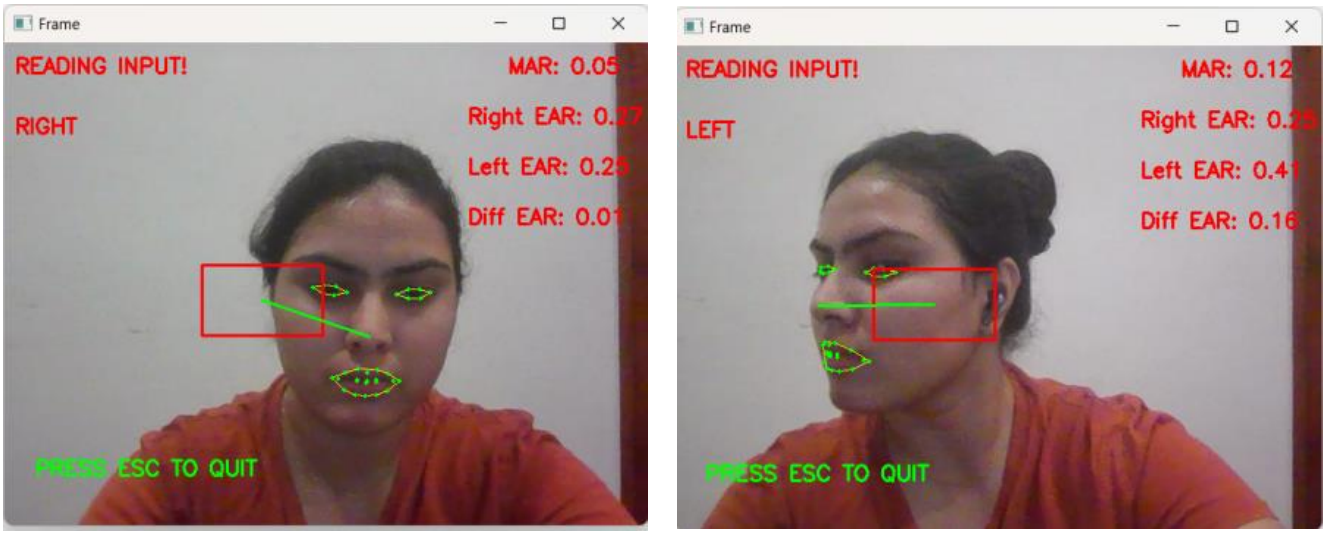

Figure 8: Reading Input

After the mouse is activated, there is a box appearing on the face which represents the movement of the cursor. In this image, the green line shows that the cursor is moving towards right. In the right side of the image, the EAR and MAR is calculated.

IARJSET ISSN (O) 2393-8021, ISSN (P) 2394-1588
International Advanced Research Journal in Science, Engineering and Technology
Impact Factor 8.066 ⁚ Peer-reviewed & Refereed journal ⁚ Vol. 11, Issue 5, May 2024
DOI: 10.17148/IARJSET.2024.11516

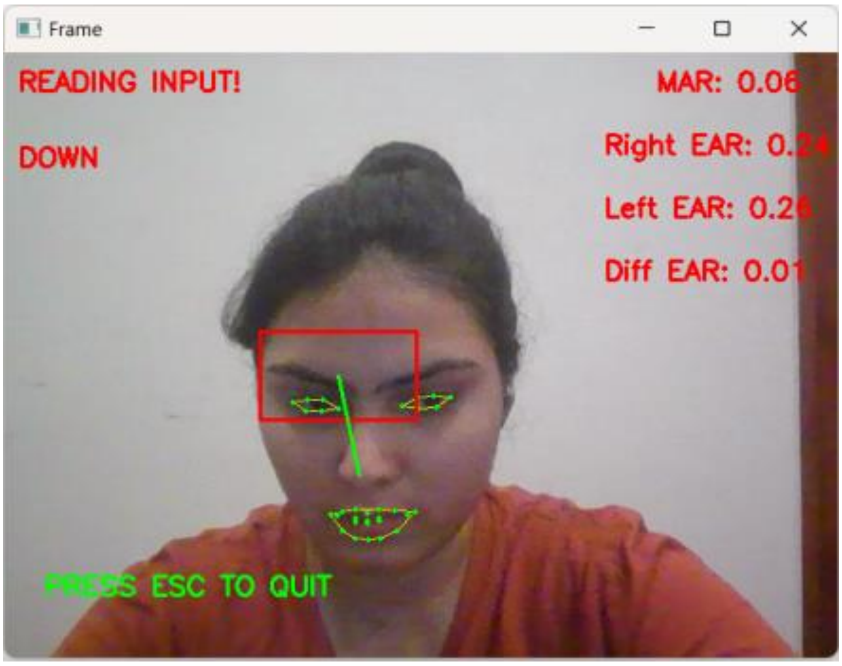

Figure 9: Reading Input downside

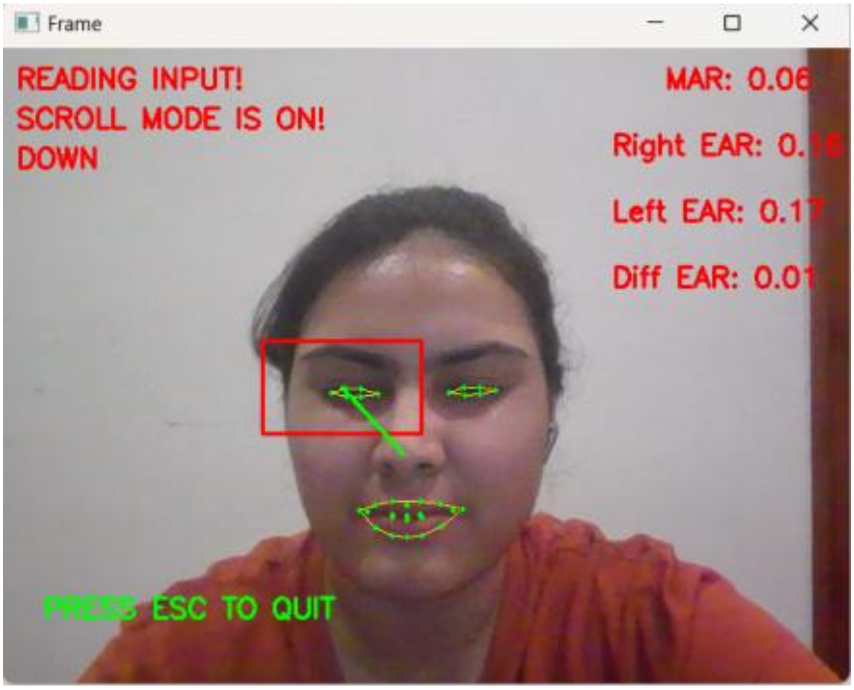

Figure 9: Scroll Mode

By the help pf squinting the eyes, the scroll mode can be turned on by which any document can be scrolled downwards or upwards just by rotating the head down side or upside respectively.

DOI: 10.17148/IARJSET.2024.11516

VII. CONCLUSION

Improvements in the system's speed can be achieved through the utilization of better-trained models. Additionally, enhancing the system's adaptability can involve adjusting the pointer position based on the user's head rotation, offering users control over the cursor's movement speed. Since the range of values is closely tied to aspect ratios, which tend to be minute, future research efforts might concentrate on refining the precision of these ratios. This may entail modifications to the formulas governing aspect ratios to bolster the algorithm's detection accuracy.

Incorporating specific image processing techniques before facial recognition can expedite the face recognition process. A multimodal system holds promise for individuals with hand or arm limitations who struggle with traditional mouse or keyboard usage. Such a system could utilize head movements for cursor navigation while employing speech commands for control inputs.

REFERENCES

[1] B. Rebsamen, C. L. Teo, Q. Zeng, M. Ang. Jr. "Controlling a wheel chair indoors using thought" IEEE Intelligent Systems, 2007, pp. 18-24.
[2] C. A. Chin "Enhanced Hybrid Electromyogram / Eye gaze tracking cursor control system for hands-free computer interaction", Proceedings of the 28th IEEE EMBS Annual International Conference, New York City, USA, Aug 30-Sept 3, 2006, pp. 2296-2299.
[3] J. Kierkels, J. Riani, J. Bergmans, "Using an Eye tracker for Accurate Eye Movement Artifact Correction", IEEE Transactions on Biomedical Engineering, vol. 54, no. 7, July 2007, pp. 1257-1267.
[4] A. E. Kaufman, A. Bandyopadhyay, B. D. Shaviv, "An Eye Tracking Computer User Interface", Research Frontier in Virtual Reality Workshop Proceedings, IEEE Computer Society Press, October 1993, pp. 78-84.
[5] T. Kocejko, "Device which will allow people suffered from Lateral Amyotrophic Sclerosis to communicate with the environment", MSc thesis, January 2008. G. A. Myers, K. R. Sherman, L. Stark, "Eye Monitor", IEEE Computer Magazine, Vol. March 1991, pp. 14-21.
[6] C. Collet, A. Finkel, R. Gherbi, "A Gaze Tracking System in Man-Machine Interaction", Proceedings of IEEE International Conference on Intelligent Engineering Systems, September 1997.
[7] B. Hu, M. Qiu, "A New Method for Human-Computer Interaction by using Eye-Gaze", Proceedings of IEEE International Conference on Systems, Man, and Cybernetics, October 1994.
[8] P. Ballard, G. C. Stockman, "Computer operation via Face Orientation", Pattern Recognition vol. 1. Conference A: Computer Vision and Applications, Proceedings., 11th IAPR International Conference, 1992.
[9] https://www.mathworks.com/matlabcentral/fileexchange/ 247-vfm
[10] Brooks, R.E. (1997) ―Towards a theory of the cognitive processes in computer programming, ‖ Int. J. Man-Mach. Studies, vol. 9, pp. 737–751.
[11] Cheng- Chih Wu, Ting-Yun Hou (2015) ‖Tracking Students' Cognitive Processes During Program Debugging‖—An Eye-Movement Approach, IEEE.