Ocular Mouse

Dr.P.Muralidhar V.Anvesh Reddy K.Sreeja Reddy P.Gourinath

Abstract

The world of computers may constantly improve in this period of contemporary applied sciences. Because it meets the demands of quadriplegics, Hands-free Computing has gained popularity recently. It demonstrates a human-computer interaction approach that is extremely beneficial to people with hand impairments. Our proposed system functions like a conventional computer mouse by translating eye motions such as winking and squinting in the direction of cursor activities. For implementing this system, an elementary webcam and software specifications such as Python, Numpy, OpenCV, and some more programmes needed for facial recognition are used. A linear classifier, a histogram of oriented gradients (HOG) feature, and a deep learning technique are used to construct the face detector. This system is finger-free and does not require any extra hardware or sensors.

Keywords

Hands-free Computing,Webcam,Python,OpenCV,Deep learning, Linear Classifier,Histogram of Oriented Gradients

1. Introduction

Moving a finger or the conventional computer mouse around the screen to move the cursor is a prevalent technological practise nowadays. The technology tracks any mouse or finger movement and translates it into cursor movement. Amputees, or those without functional arms, won't be able to use the mouse with the current technology. As a result, the ocular movement is frequently mapped to the mouse, allowing the amputee to move the cursor as they choose if their eyeball movement is detected and the direction in which their focus is observed can be identified. A person with amputation might benefit greatly from an "eye tracking mouse." The attention tracking mouse isn't widely accessible right now, and only a few companies have created and released this technology.We're working on a mouse that uses facial landmarks and eye tracking to let the user control the pointer with his face.We attempt to determine the user's direction of "look" and move the pointer in that direction as his face attempts to project. Clicking is accomplished by winking the right or left eye.

1.1 Related work

Technology progress has resulted in the creation of several human-computer interaction (HCI) systems, including those that use eye and facial motions to operate the cursor. With a suggested algorithm for a multimodal HCI system that effectively controls all mouse actions, including clicks and scrolling, one proposed system allows the user to control the mouse cursor using facial movements. Additionally, the system allows for the addition of functions to address real-world scenarios, and other computer - generated biases can be managed through the use of HCI techniques [1]. The Eye Aspect Ratio (EAR) approach, which may be extended to include all mouse activities, is used by another system that uses the Raspberry Pi and OpenCV to control the cursor by eyeball movement [2].A third system also uses Dlib and the Haar Cascade algorithm to control the cursor depending on the user's facial expressions and eye movements. This system may be extended with the inclusion of new features like clicking events and interfaces that respond to blinking and eye movement [3]. Another facial expression-based solution enables people with physical disabilities to utilise the mouse cursor independently by recording and sending data from the subject's mood using a normal web camera and the Viola & Jones algorithm [4,10]. The mouse cursor is controlled in a system based on eye movement employing limbus tracking, pupil monitoring, electrooculography, and saccade, which has certain restrictions on head movement and squinting [5]. Another technique improves cursor control utilising an eye mouse and an object similar to a HaarHowever, it lacks scrolling and red-eye effect detection [6]. Other detectors include Hough man Circle Detection, OpenCV, and separate methods for detecting the iris. The centre of the pupil is used for the same reason by a system that makes use of the Raspberry Pi and specialised hardware to regulate eye movement and coordinate with cursor function. These systems provide a number of advantages for HCI, such as allowing people with physical disabilities to use computers on their own, offering fresh ways to connect with computers, and furthering HCI through the usage of IoT [7]. The method for identifying eye blinks and winks in real- time utilising video and MATLAB software is described in this study. The method includes localization of the eyes and face pair,optical flow categorization, and analysis. The technology presented in this paper has been confirmed and tested in a variety of settings and has demonstrated remarkable accuracy in identifying blinks and winks and carrying out mouse-analogous operations [8]. This study suggests an algorithm that controls mouse movements in a computer by detecting iris movements. Physically challenged people can benefit from being able to communicate and use computers. Cursor movement is controlled by iris movement, whereas mouse clicks and scrolling are controlled by eye blinking [9]. The use of spectral histograms and support vector machines to detect faces is described.

2. Proposed work

The primary goal of our suggested technique in this system is to provide a way for directing the computer mouse cursor through face motions rather than physical manipulation. This method eliminates the need for wearable devices or sensors because the system may be used with a laptop's built-in camera or a conventional webcam to work. The user's options include opening their lips to activate programmes, squinting their eyes to activate and deactivate scroll mode, and winking one eye to click events.

Additionally, the user has the ability to direct the cursor in the desired direction by simply changing the angle and motion of their face.

2.1 Facial Landmarks Prediction Model

To recognise and process certain facial features, such as the mouth, right eye, left eye, nose, and jaw, the system uses facial recognition. The system can recognise and react to many facial motions, such as the opening of the lips, squinting of the eyes, and winking, thanks to these expected markings.

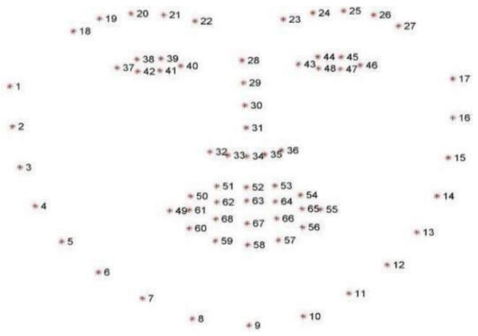


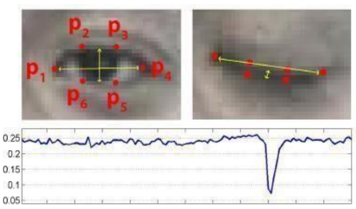
Fig 2.1 The pre-existing facial landmark detector contained in the dlib library is employed to approximate the placement of 68 coordinate pairs (x1, y1) that correspond to various facial features on the visage.

Specific features that allow us to detect blinks through Eye-Aspect-Ratio and yawns through Mouth-Aspect-Ratio can be produced by making expected facial expressions. The PyAutoGUI module is used to programme these operations in order to control the mouse cursor.

2.2 Eye Aspect Ratio [EAR]:

Eye blinks and winks can be detected using the eye aspect ratio (EAR). When the eye is closed, the EAR value falls below a set threshold, and when the eye is open, the EAR value rises over the threshold.

If EAR=THRESHOLD, then

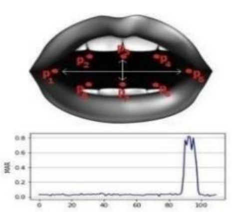


$$\text{EAR} = \frac{\|p_2 - p_6\| + \|p_3 - p_5\|}{2\|p_1 - p_4\|},$$

2.3 Mouth Aspects Ratio [MAR]:

The mouth aspect ratio (MAR) value increases when the mouth isopen and decreases when it is closed, allowing the system to detect when someone's mouth is opening and shutting. By integrating MAR and eye aspect ratio (EAR) data, the suggested approach makes use of numerous modalities.

If MAR=THRESHOLD, then



$$\textbf{MAR} = \frac{\|p_2 - p_8\| + \|p_3 - p_7\| + \|p_4 - p_6\|}{2\,\|p_1 - p_5\|}$$

'CLOSE' is the STATUS code.

3. Implementation

The initial stage of implementation is specifying the relevant threshold values and initialising the frame and frame length counters, which control mouse activity. The next stage is to getthe HOG face sensor's readings of the corners of the face and the markers for the mouth, both eyes, and nose. The face in the image frame is then converted to grayscale format, and the coordinates(x1, y1) for the facial markers are saved in a NumPy array. The Mouth Aspect Ratio and Eye Aspect Ratio are then calculated. Finally, a bounding box is drawn precisely from the nose's centre, and within that bounding box, the required operations canbe carried out.

3.1 Procedure

The programme in this project employs facial landmark detection to carry out mouse motions including left- and right-clicking, scrolling, and pointer movement. It employs the PyAutoGUI, Dlib, and OpenCV libraries.To determine the eye aspect ratio, mouth aspect ratio, and direction between two places, the programme first provides a few assistance functions. Then, it initialises some variables, including the mouse action thresholds and the action counters.After that, the programme configures Dlib's face detector and predictor and gets the indexes for the facial landmarks representing the left and right eyes, nose, and mouth.The software grabs a video frame from the camera, modifies its dimensions, and transforms it into a grayscale format as part of the main set of instructions.Then it goes on to identify any facial features visible in the grayscale image and pinpoint their precise spatial location. It specifically returns the positions of the landmarks for the identified facial region's left and right eyes, mouth, and nose.In order to initiate mouse motions like left and right clicks and scrolling, the programme computes the aspect ratios for the eyes and lips. When the mouth is open, it also moves the cursor using the direction function. The camera frame is then shown, and the programme then waits for a key push to end.

4.Result

The main goal of this research was to make machines better at interacting with and reacting to human behaviour. The essay concentrated on developing portable, low-cost technology that works with any typical operating system. The suggested method seeks to control the mouse cursor by taking a picture of a person's face and figuring out where their eyes and nose are located. The direction of the head movement will match the direction of the pointer. Basic mouse movements like scrolling, left- and right-clicking, and all-around movement will be controlled by the mechanism.Opening the lips causes the system to activate the cursor, and looking at the head causes it to move. The clicking actions are carried out by giving the eyes winks. The left eye is used to wink when clicking left, while the right eye is used to wink when clicking right. Squinting the eyes may be used to either activate or disable scrolling.

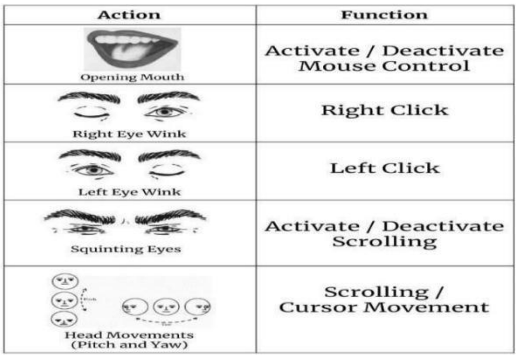


Fig 4.1 A list of proposed tasks and their intended outcomes.

5.Conclusion and Future Scope

With an emphasis on tracking head movement as the main event for cursor movement, our research's goal is to improve the operation of our programme. Faster pointer movements are intended to completely replace conventional mouse activities. Furthermore, we want to include advanced features like dragging, zooming in and out, voice recognition, text extraction for typing, and more. According to this concept, an interface will be created that may be used in practical applications, especially to help physically disabled persons operate a PC or laptop using their facial expressions and movement without the aid of others. Through the use of technology, this paper aims to enable people with disabilities to pursue their goals and lessen their reliance on others.

6. References

[1] Shashidhar R. et al. "Mouse Cursor Control Using Facial Movements -An HCL Application" 2022 International Conference on Sustainable Computing and Data Communication System (ICSCDS). IEEE, 2022.

[2] Pavithra et al., "Cursor Movement Control Using Eyes and Facial Movements for Physically Challenged People." The IRJET, an international journal of engineering and technology research, is published in 2021. p-ISSN: 2395-0072; e-ISSN: 2395-0056.

[3] Satyanarayana B., et al., "Eyeball Movement based Cursor Control Using Raspberry Pi and OpenCV." IJITEE, the International Journal of Innovative Technology and Exploring Engineering, will be published in 2020. ISSN: 2278-3075.

[4] Vasanthan M and colleagues. "Face Expression Based Computer Cursor Control System for Assisting Physically Disabled Person" ResearchGate Publications, COMNESAT 2012, IEEE 2012.

[5] Shrunkhala Satish Wankhede et al., "CONTROLLING MOUSE CURSOR USING EYE MOVEMENT." The 2013 National Conference on Recent Advances in Technology and Management for Integrated Growth (RATMIG) received a special issue of the International Journal of Application or Innovation in Engineering & Management (IJAIEM), ISSN 2319-4847.

[6] Margi Sadashiv Narayan et al., "ENHANCED CURSOR CONTROL USING EYE MOUSE." International Journal of Advances in Electronics and Computer Science, ISSN 2393-2835, December 2016.

[7] Sovit Ranjan Rath et al., "Machine Learning and Deep Learning, Face detection with Dlib using HOG and Linear SVM".

[8] Hari Singh and Jaswinder Singh, "Real-time eye blink and wink detection for object selection in HCI", Springer Nature 2018, a division of Springer International Publishing AG.

[9] "VIRTUAL MOUSE USING EYE TRACKING", Journal of Xi'an University of Architecture & Technology, Volume XII, Issue IV, 2020. K. Akhil Kumar, B. R. Ishwarya, Prasanna D. V., and Mrs. Shyamala.

[10] "Mouse Pointer Control Using Eye Movements", Volume 6, Issue 4, JETIR April 2019, Prof. Bhakti Sonawane, Prathamesh Punde, Saurabh Dubey.