2023 IEEE 8th International Conference for Convergence in Technology (I2CT)
Pune, India. Apr 7-9, 2023

Face Gesture Based Virtual Mouse Using Mediapipe

Akula Kumar Raja
Department of Electronics and Communication Engineering
Amrita Vishwa Vidyapeetham
Amritapuri, India
akulakumarraja23@gmail.com

Chidakash Sugandhi
Department of Electronics and Communication Engineering
Amrita Vishwa Vidyapeetham
Amritapuri, India
chidakash133@gmail.com

Gorantla Nymish
Department of Electronics and Communication Engineering
Amrita Vishwa Vidyapeetham
Amritapuri, India
gorantlanymish@gmail.com

Nama Sai Havish
Department of Electronics and Communication Engineering
Amrita Vishwa Vidyapeetham
Amritapuri, India
saihavishnama8@gmail.com

Manazhy Rashmi
Department of Electronics and Communication Engineering
Amrita Vishwa Vidyapeetham
Amritapuri, India
manazhyrashmi@am.amrita.edu

2023 IEEE 8th International Conference for Convergence in Technology (I2CT) | 979-8-3503-3401-2/23/$31.00 ©2023 IEEE | DOI: 10.1109/I2CT57861.2023.10126453

***Abstract*—A disabled person's life is always dependent on someone else who needs aid with mobility or any other task. Individuals with disabilities may face challenges when using computers. The most common way of interacting with computers is using a mouse and keyboard. It is difficult for people with physical disabilities to use them. Facial movements are one of the best possible actions by physically disabled individuals, By recognizing and responding to these movements, it is possible for them to operate the computer using only their facial expressions. Face recognition is a contemporary approach to interaction between humans and computers (HCI) i.e., The proposed system can easily control the computers by using face gesture recognition. It can be a viable replacement for traditional HCI tools in the future. This research outlines the techniques utilized in the design, implementation, and evaluation of the experiments conducted and presents the results obtained.**

***Index Terms*—Human-computer interaction, facial movements, facial expression recognition, Mediapipe, Deep Learning, and Face recognition.**

I. Introduction

A computer can play a critical role in the daily lives of every individual even for people with disabilities. It can assist them in various ways, including communication, education, and employment, as well as providing access to information and resources. Specialized technologies, such as screen readers, speech recognition software, and alternative input devices, can make a computer more usable for those with various impairments. The internet also allows for greater autonomy and socialization opportunities for disabled individuals. In summary, Overall, the use of computers and assistive technologies can greatly enhance the independence, productivity, and inclusion of disabled individuals in all areas of life.

In this research, a new virtual mouse controller was developed to control cursor movements and perform mouse operations using a web camera. The proposed system aims to overcome the limitations of all the previous cursor control systems. The proposed system is designed in such a way that it can be accessed by everyone. It can easily control the cursor movements on the computer screen even for disabled persons.

A. PROBLEM STATEMENT

The challenge of enabling individuals who have lost the use of their arms to continue using a computer for their day-to-day activities are the focus of this research. The objective of this project is to create a virtual Human-Computer Interaction (HCI) system where a human can interact with a computer just by using his face. The goal of our project is to provide an alternative method of interaction for individuals with arm impairments, to improve their quality of life by enabling them to carry out their day-to-day activities on computers. The proposed solution is a virtual HCI system that utilizes facial expression recognition and tracking technology to interpret and respond to the user's commands. Facial expression recognition [12] is a prevalent subject in the field of deep learning. Applications in this field often target devices with limited computational power. The system will be evaluated based on its effectiveness and ease of use for individuals with arm impairments.

B. PROPOSED SYSTEM

The proposed system uses the Opencv, autopy, Mediapipe, and Pyutogui [19] libraries to identify the facial landmarks and determine the direction of the face [15]. OpenCV [18] is a software library that contains a collection of image-processing algorithms specifically designed for object detection. It is written in the Python programming language and can be utilized to develop real-time computer vision applications [8, 9]. This library is commonly used in image and video analysis, particularly for tasks like facial recognition and identifying objects in pictures or videos. It can be used to build a wide range of applications that require image processing capabilities. OpenCV was used to capture data from a live video feed via a webcam. This data was then used to extract individual frames. These frames are converted from BGR To RGB and then sent to a MediaPipe for the identification of landmarks. This proposed model uses the MediaPipe which contains 468 landmarks provides more accurate and precise control of mouse operations compared to previous approaches. With the help of these landmarks, we calculate the Mouth aspect ratio, Eye aspect ratio, left-eye aspect ratio, right-eye aspect ratio, and right-click and left-click aspect ratios. These are used to trigger mouse functions under certain conditions.

II. RELATED WORK

In paper [1], A mouse control system for individuals with physical disabilities has been developed using Kinect technology. This system enables the user to navigate a cursor on a computer screen by moving their head and performing a mouse click by opening or closing their mouth or eye. However, since the production of Kinect has been discontinued. Therefore, instead of utilizing the Kinect control system, Our proposed system is better to use, Our proposed model accurately recognizes mouse clicks regardless of distance, due to the web camera's narrower angle of view and ability to capture brighter images. This results in better landmark detection and improved accuracy. The improvement is attributed to differences between the web camera and Kinect. In [2], the authors developed a system that utilizes the dlib library for recognizing and tracking facial expressions. The system works by receiving live input from the user and analyzing it to identify faces. After recognizing the face, landmarks are detected, and based on these landmarks, certain mouse operations are performed. However, the proposed system has some limitations in cursor movement, as the cursor can only move either vertically or horizontally. Our proposed model addresses this limitation by allowing the cursor to move in any direction. In [3], the authors have designed and implemented a control system for a computer mouse that is intended for physically disabled individuals. The system utilizes a web camera, the dlib library, and OpenCV to track the user's face and facial movements. The user can move the cursor on the screen by turning their face in different directions and perform clicks by opening or closing their mouth or eyes. However, the proposed system only had a one-click operation. In contrast, our proposed model has been developed with additional click operations.

This proposal [4] presents a novel method for a virtual head mouse interface device that utilizes the strength of modern face detection techniques, pattern recognition, and motion tracking algorithms. Instead of determining head rotation, it uses relative movement of the face in the image to control the cursor. The aim is to create a precise and lag-free cursor control experience through enhanced activation, control and clicking procedures. However, there are Potential disadvantages of this model including false positives, leading to unintended actions, and the requirement for additional hardware (infrared illumination or a sticker reflector). Our proposed model aims to overcome all these limitations. The authors of the paper [5] describe a method for using a web camera and a microcontroller to capture facial expressions and control a computer mouse. Four small luminous stickers are placed on the subject's face, and the subject is asked to make various facial expressions. The camera records the movement of the stickers and assigns a unique binary code to each expression. The microcontroller then sends this code to the computer, which uses it to control the movement of the mouse and trigger click actions. This method requires the use of an extra microcontroller and luminous stickers to perform the mouse operations. In contrast, our proposed model does not require these materials, making it cheaper than the previous model.

In [13] The authors present a method for recognizing facial expressions using thermal images and a modified ResNet152 model. The ResNet152 was chosen for its optimization efficiency and accuracy potential through increased depth. The paper aims to train thermal images with the modified ResNet152 to predict emotions. The authors in [7] describe a system for real-time computer vision that utilizes the Python, TensorFlow, and OpenCV libraries. this system is used to perform mouse operations based on hand gestures [14]. They employ a machine learning-based approach and implement it using the SSD MobileNet [16] model, which is trained using transfer learning. This system allows for efficient and accurate real-time computer vision through the use of advanced machine-learning techniques and powerful libraries. The proposed method in paper [6] utilizes OpenCV and other libraries to create a virtual mouse system that tracks finger movements in real time. The system captures video frames, processes them to detect the color of fingertips, and uses the midpoint of two fingers as the cursor. Finger touching is interpreted as a left or double click. However, the system has some limitations, such as the need for color caps on the fingers and difficulty in detecting fingers in complex backgrounds. Our model is much more advanced and cheaper. This [10] article proposes a way to control a computer mouse through the use of RGB-D images and fingertip recognition. The method utilizes a Microsoft Kinect Sensor 2 to determine the hand and palm center, and the K-cosine algorithm is used to identify the fingertips based on hand contour information. The fingertip position is then transformed into RGB images and applied to control the cursor on a computer display. This approach utilizes depth information from the RGB-D images to control a computer mouse, but it is only intended for people who have no disabilities. In contrast, our proposed model can be used even for disabled persons using face gesture recognition. It is an upgraded version of all the previous cursor control systems.

III. PROPOSED WORK

The proposed system operates by acquiring real-time input from the user through a webcam. This input is used to perform mouse operations on the computer screen. The primary tool utilized in the system is Mediapipe [17], an open-source platform that has the ability to identify facial features through the detection of facial landmarks. It employs machine learning to analyze the video stream from a camera and extract facial features such as eyes, mouth, and nose. This information is then used to detect and track facial movements, such as raising eyebrows or opening the mouth, which in turn can be used to control the movement of the cursor on the computer screen. With the help of Mediapipe, a mouse control system has been developed that allows for the use of facial gestures to perform mouse operations.

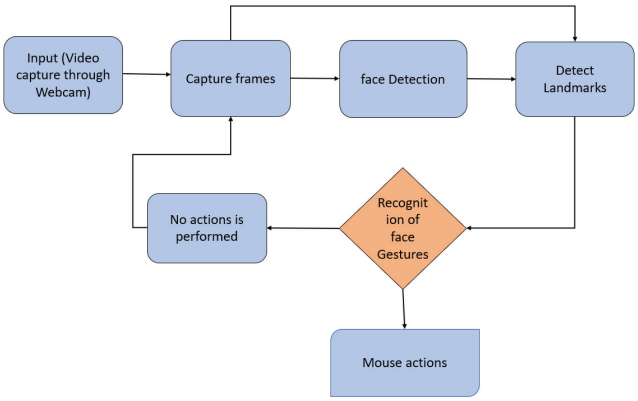

Fig. 1. Block Diagram of our model.

here certain functions are implemented that exactly perform mouse operations. some of them are given below

1) Mouse Cursor Movement
2) Left Click
3) Right Click
4) Double Click
5) Scrolling

A. Mouse Cursor Movement

The mouse cursor is assigned to the landmark of the nose tip. The cursor is moved accordingly to the movement of our nose. The landmark array has been created which contains a total of 468 landmarks detected from the face. The movement of the cursor is triggered by determining the mouth aspect ratio, which is found by computing the euclidean distance between the landmarks of the mouth.

$$Mar = \frac{D1 + D2 + D3}{2 * D4} \quad (1)$$

where D1, D2, D3, and D4 are the euclidean distances between the coordinates of corresponding landmarks as shown in Fig. 2. It can be calculated by the using formula

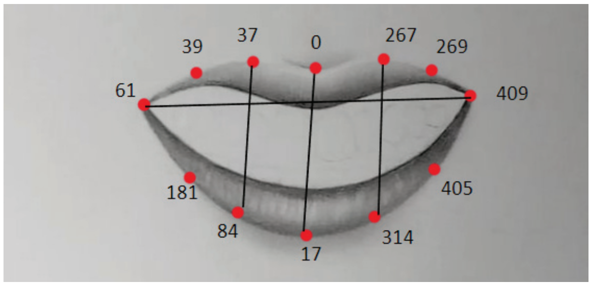

Fig. 2. Mouth landmarks.

given below.

$$D = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2} \quad (2)$$

When the mouth aspect ratio exceeds a certain threshold value, the count is incremented. If the count surpasses a threshold count, the cursor movement function is activated. The threshold value has been determined through experimentation set at 1.1 and the threshold count set to 15 which means if the mouth is open for 15 seconds, the mouse cursor will be activated.

B. Left Click

The system triggers the left click by utilizing the landmarks from the left eye. To perform a left click, the cursor movement must first be disabled, and then the aspect ratio of both the left and right eyes are calculated using an aspect ratio function. The left eye-aspect ratio and right eye-aspect ratio are used to determine if a left click should be performed.

$$EAR = \frac{D1 + D2}{2.0 * D3} \quad (3)$$

The above formula is used to find the aspect ratio of the left eye when the left eye landmarks are passed to the aspect ratio function.

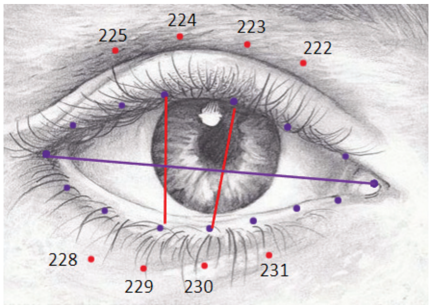

Fig. 3. Left eye landmarks

where D1 and D2 represent vertical lines and D3 represent

horizontal line. These are nothing but distances to the corresponding coordinates of the left-eye landmarks.
Similarly, the right-eye aspect ratio is calculated by passing the right-eye landmarks to the aspect ratio function. The left click-action ratio is also determined by evaluating the landmarks of the eye crease and the lower lid crease of the left eye. Using these landmarks we created a Numpy array of name left-click.

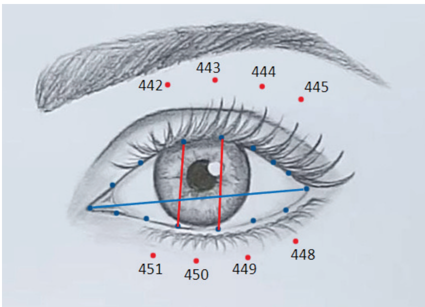

Fig. 4. Right eye landmarks

where D1, D2, and D3 are the euclidean distances between the coordinates of corresponding landmarks. As shown in Fig. 4.

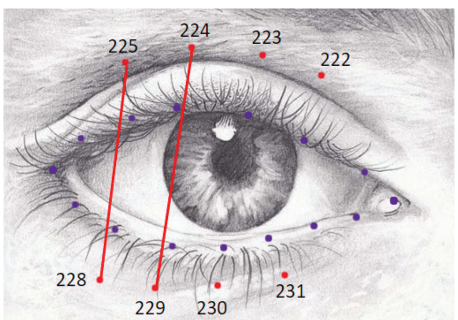

Fig. 5. Landmarks of the eye crease and the lower lid crease of the left eye.

The left-click action ratio is determined by finding the average euclidean distances of the coordinates of corresponding landmarks of the eye crease and the lower lid crease of the left eye as shown in Fig. 5.

```
if Left_EAR > Right_EAR:
    if Left_action < threshold:
        counter += 1
    endif
    if counter > threshold-frame-count:
        # left click action is performed
    endif
endif
```

The left-click function is executed when the $Left_{ear}$ is larger than the $Right_{ear}$ and the left click action ratio is below a certain threshold. This occurs only if the count exceeds the specified threshold frame count. where the threshold value is set to 30.0 by manually performing 'N' no.of experiments. the threshold-frame-count is set to 15 which means if the left eye is closed for 15 seconds, then the Left-click is performed. In paper [11], the left click was originally implemented through the use of eye aspect ratio, but due to its unreliable performance, an alternative method was adopted utilizing landmarks from the eye crease and lower lid crease of the eyes.

C. Right Click

Right-click action is performed similarly to how left-click action is performed. The right-eye aspect ratio is already determined for both left and right clicks. The calculation of the right click-action ratio is performed similarly to the calculation of the left click-action ratio. The right-click landmarks are passed to the function instead of the left-click landmarks to determine this ratio.

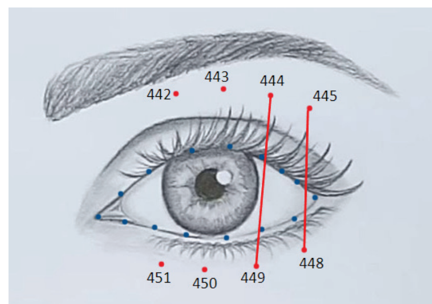

Fig. 6. Landmarks of the eye crease and the lower lid crease of the right eye.

```
if Right_EAR > Left_EAR:
    if right_action < threshold:
        counter += 1
    endif
    if counter > threshold-frame-count:
        # right click action is performed
    endif
endif
```

The above algorithm states that if the $Right_{ear}$ is larger than $Left_{ear}$ and the Right-click action ratio is smaller than the threshold value, then if the count surpasses a threshold-frame-count, the Right-click action is performed. where the threshold value is set to 28.0 by manually performing 'N' no.of experiments. the threshold-frame-count is set to 15 which means if the Right eye is closed for 15 seconds, then the Right-click is performed.

D. Double Click

When the euclidean distance between one of the landmarks on the eyebrow and the eye is greater than the threshold value, a double-click action is performed as a result of rising eyebrows. where the threshold value is set to 28 by manually performing 'N' no.of experiments.

The landmarks of each eyebrow as shown in figure Fig. 7

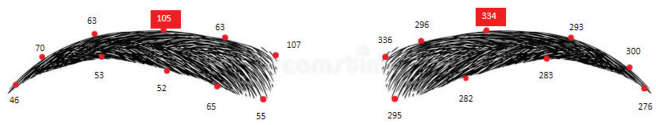

Fig. 7. Landmarks of each eyebrow.

E. Scrolling

For scrolling, the cursor movement must first be activated by opening the mouth for a few seconds. The scrolling function is activated by squeezing the eyes, and deactivated by squeezing the eyes again.

```
if EAR <= threshold:
    counter += 1
    if counter > threshold-frame-count:
        # SCROLL_MODE = not SCROLL_MODE
    endif
endif
```

The scrolling function is activated when the aspect ratio of the eye is less than or equal to a threshold value. where the threshold value is set to 0.2 by manually performing 'N' no.of experiments.

IV. EXPERIMENTAL RESULTS

The comparative analysis among existing systems is given in the table based on various factors or criteria. The proposed system can overcome all the drawbacks of existing systems.

TABLE I
COMPARATIVE ANALYSIS AMONG EXISTING SYSTEMS

S.NO	*Existing Systems*	*Supported Device Functions*	*Drawbacks*
1	Mouse System Using Kinect [1]	Cursor movement, Single left click	Kinect is out of production and the accuracy of mouse clicking recognition is inadequate.
2	Computer Cursor Control Using Face Gestures [2]	Cursor movement, Left click, Right click, Double click, Drag and drop, Scrolling	This system has some limitations in cursor movement, as the cursor can only move either vertically or horizontally.
3	Mouse Cursor Control System using Facial Movements [3]	Cursor movement, Left click	The proposed system only had a one-click operation.
4	Robust absolute virtual head mouse control system [4]	Cursor movement, Left click	This model may have false positives, causing unintended actions, and require additional hardware.
5	Facial Expression Based Computer Cursor Control System [5]	Cursor movement, Left click	The method requires additional hardware, such as an extra micro-controller and luminous stickers, making it more expensive.

Some of the results are captured and shown below.
Once the model detects a face, when the mouth is opened, it activates the cursor movement function based on the mouth

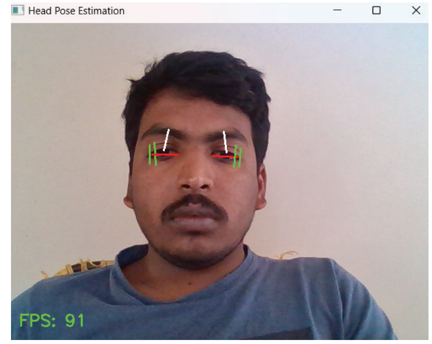

Fig. 8. Face detection.

aspect ratio. that shown in Fig. 9.

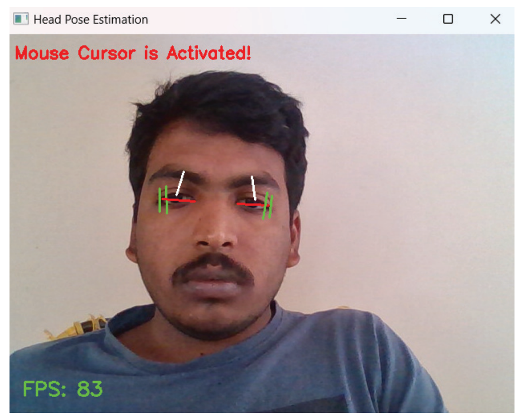

Fig. 9. Mouse Cursor Is Activated.

In Fig. 10, it is shown that a left-click operation is executed

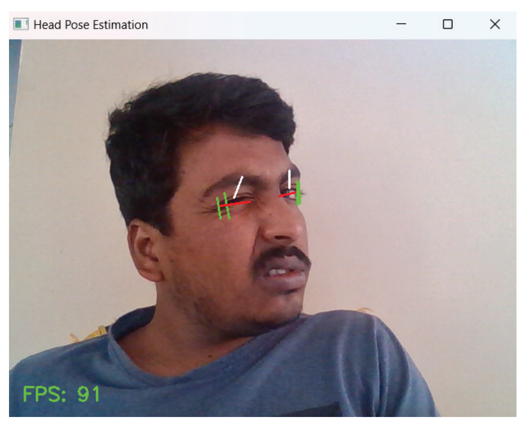

Fig. 10. Left Click.

when a wink of the left eye is detected and the frame rate is nearly 90.

As shown in Fig. 11, the right-click operation is executed whenever the right eye is winked.

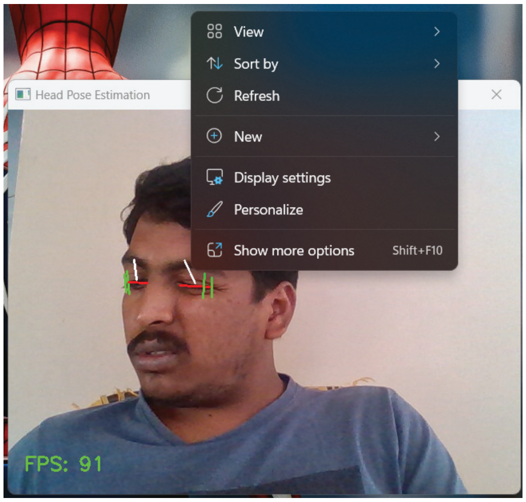

Fig. 11. Right Click.

V. CONCLUSION

A virtual mouse system was proposed for individuals with physical disabilities that utilizes a web camera, mediapipe, and OpenCV. The system allows users to control the movement of the cursor by tracking the position of the nose tip. It also allows for various click actions, such as left-clicking, right-clicking, double-clicking, and scrolling, to be performed based on the aspect ratios of the mouth and eyes. These actions are triggered when specific conditions are met. This system is designed to provide an alternative means of control for those unable to use traditional mice. The proposed system aims to address the limitations of previous cursor control systems by making it accessible to all individuals, including those with disabilities. It is designed to provide easy control over cursor movements on a computer screen.

Future work is to improve the functionality of the virtual mouse. As of now, we implement cursor movement, left-clicking, right-clicking, double-clicking, and scrolling. We want to add other functions like drag and drop etc.

ACKNOWLEDGMENT

We are very thankful to Amrita Vishwa Vidyapeetham for providing guidance. We would like to convey our deep gratitude to our mentor Dr. Manazhy Rashmi for her valuable and constructive suggestions, and enthusiastic encouragement for completing this research work successfully.

REFERENCES

[1] Miyachi, Junpei, and Masayoshi Tabuse. ”Development of Mouse System for Physically Disabled Person by Face Movement Using Kinect.” Proceedings of the 2017 International Conference on Artificial Life and Robotics. 2017.

[2] A. Dongre, R. Pinto, A. Patkar and M. Lopes, ”Computer Cursor Control Using Eye and Face Gestures,” 2020 11th International Conference on Computing, Communication and Networking Technologies (ICCCNT), Kharagpur, India, 2020, pp. 1-6, doi: 10.1109/ICCCNT49239.2020.9225311.

[3] Masayoshi Tabuse, Manase Mizobe, Yasunari Yoshitomi, Taro Asada, Mouse Cursor Control System using Facial Movements, Journal of Advances in Artificial Life Robotics, 2020-2021, Volume 1, Issue 1, Pages 40-45, Released on J-STAGE September 07, 2022, Online ISSN 2435-8061, $https : //doi.org/10.57417/jaalr.1.1_{4}0$

[4] Pallejà, T., Guillamet, A., Tresanchez, M. et al. Implementation of a robust absolute virtual head mouse combining face detection, template matching and optical flow algorithms. Telecommun Syst 52, 1479–1489 (2013). https://doi.org/10.1007/s11235-011-9625-y

[5] M. Vasanthan, M. Murugappan, R. Nagarajan, B. Ilias and J. Letchumikanth, ”Facial expression based computer cursor control system for assisting physically disabled person,” 2012 IEEE International Conference on Communication, Networks and Satellite (ComNetSat), Bali, Indonesia, 2012, pp. 172-176, doi: 10.1109/ComNetSat.2012.6380800.

[6] K. H. Shibly, S. Kumar Dey, M. A. Islam and S. Iftekhar Showrav, ”Design and Development of Hand Gesture Based Virtual Mouse,” 2019 1st International Conference on Advances in Science, Engineering and Robotics Technology (ICASERT), Dhaka, Bangladesh, 2019, pp. 1-5, doi: 10.1109/ICASERT.2019.8934612.

[7] U. Sairam, D. K. Reddy Gowra and S. C. Kopparapu, ”Virtual Mouse using Machine Learning and GUI Automation,” 2022 8th International Conference on Advanced Computing and Communication Systems (ICACCS), Coimbatore, India, 2022, pp. 1112-1117, doi: 10.1109/ICACCS54159.2022.9784972.

[8] S. Shriram, B. Nagaraj, J. Jaya, S. Shankar, P. Ajay, ”Deep Learning-Based Real-Time AI Virtual Mouse System Using Computer Vision to Avoid COVID-19 Spread”, Journal of Healthcare Engineering, vol. 2021, Article ID 8133076, 8 pages, 2021.doi.org/10.1155/2021/8133076.

[9] K. S. Varun, I. Puneeth and T. P. Jacob, ”Virtual Mouse Implementation using Open CV,” 2019 3rd International Conference on Trends in Electronics and Informatics (ICOEI), Tirunelveli, India, 2019, pp. 435-438, doi: 10.1109/ICOEI.2019.8862764.

[10] Tran, DS., Ho, NH., Yang, HJ. et al. Real-time virtual mouse system using RGB-D images and fingertip detection. Multimed Tools Appl 80, 10473–10490 (2021). https://doi.org/10.1007/s11042-020-10156-5

[11] S. R, S. K, A. P. K, A. B. Vishwagna and P. N, ”Mouse Cusor Control Using Facial Movements - An HCI Application,” 2022 International Conference on Sustainable Computing and Data Communication Systems (ICSCDS), Erode, India, 2022, pp. 367-371, doi: 10.1109/ICSCDS53736.2022.9760914.

[12] Devika, R., and J. Divya Udayan. ”Lightweight Deep Learning Facial Expression Recognition Model.” In ICT Systems and Sustainability: Proceedings of ICT4SD 2022, pp. 493-503. Singapore: Springer Nature Singapore, 2022.

[13] Prabhakaran, A.K., Nair, J.J., Sarath, S. (2021). Thermal Facial Expression Recognition Using Modified ResNet152. In: Thampi, S.M., Gelenbe, E., Atiquzzaman, M., Chaudhary, V., Li, KC. (eds) Advances in Computing and Network Communications. Lecture Notes in Electrical Engineering, vol 736. Springer, Singapore. https://doi.org/10.1007/978-981-33-6987-0_32

[14] B. B, D. R, V. T. Selvam, V. V. Kumar and R. R, ”Improved Real-Time Approach to Static Hand Gesture Recognition,” 2021 11th International Conference on Cloud Computing, Data Science Engineering (Confluence), Noida, India, 2021, pp. 416-422, doi: 10.1109/Confluence51648.2021.9377036.

[15] P. M. Manjula, S. Adarsh and K. I. Ramachandran, ”Driver Inattention Monitoring System Based on the Orientation of the Face Using Convolutional Neural Network,” 2020 11th International Conference on Computing, Communication and Networking Technologies (ICCCNT), Kharagpur, India, 2020, pp. 1-7, doi: 10.1109/ICCCNT49239.2020.9225600.

[16] Aadithya, V., Balakumar, S., Bavishprasath, M., Raghul, M., Malathi, P. (2022). Comparative Study Between MobilNet Face-Mask Detector and YOLOv3 Face-Mask Detector. In: Karrupusamy, P., Balas, V.E., Shi, Y. (eds) Sustainable Communication Networks and Application. Lecture Notes on Data Engineering and Communications Technologies, vol 93. Springer, Singapore.https://doi.org/10.1007/978-981-16-6605-6_61

[17] https://google.github.io/mediapipe/solutions/solutions.html

[18] https://opencv.org/

[19] https://pyautogui.readthedocs.io/en/latest/