Touchless Tech: Activating Applications Without Traditional Input Devices

Prakash Bethapudi[1], Appari Suma Poornima[2], Bonthalakoti Mounika Venkata Sai[3]
Ambati Devi Priya[4], Bourothu Venkat[5]
[1] *Professor, Andhra University, Visakhapatnam, Andhra Pradesh, India*
[2,3,4,5] *B.Tech IV Year, Andhra University, Visakhapatnam, Andhra Pradesh, India*

***Abstract*—This project introduces an accessibility system designed to control a laptop without requiring a traditional mouse or keyboard. Instead, the system leverages hand gestures and face tracking to facilitate intuitive interactions. Hand gestures allow seamless operations such as opening and closing applications like Chrome and Notepad, as well as capturing screenshots. Face tracking is used to adjust volume and brightness through up, down, left, and right movements. This system aims to provide an alternative input mechanism for users with physical disabilities or those looking for a touch-free computing experience. The motivation behind this project stems from the increasing need for inclusive computing solutions that cater to individuals with limited mobility. Traditional accessibility tools such as voice commands or adaptive hardware solutions can be expensive or ineffective in certain situations. By integrating computer vision and artificial intelligence, this system provides a cost- effective and easy-touse alternative that can function with a standard webcam. The implementation focuses on real-time processing, ensuring minimal latency and high accuracy for gesture recognition and face tracking. Additionally, this system can be beneficial not only for individuals with disabilities but also for users who prefer a hands-free approach to computing, improving efficiency and reducing strain from prolonged use of physical input devices. With advancements in AI and machine learning, gesture- based controls are becoming increasingly viable, making this project a step forward in the development of accessible and intuitive human-computer interactions.**

***Index Terms*—Disabilities, Face Tracking, Frame Extraction, Gesture Recognition, Hand Gestures, Human Computer Interaction, Mobility, Motion, Reducing Strain**.

I. INTRODUCTION

The continuous evolution of Human-Computer Interaction (HCI) has fostered the development of more natural, intuitive, and inclusive interfaces. Traditional input methods—such as keyboards and mice—pose significant accessibility challenges for individuals with physical impairments, limiting their ability to fully engage with technology. To address this, recent advancements in computer vision and artificial intelligence have led to alternative, touchless control systems that promote inclusive computing environments.

This paper presents a gesture and voice-controlled virtual mouse system designed to replace physical peripherals with a vision-based interface. Using a standard webcam, the system captures real-time video input and processes it with frameworks such as OpenCV and MediaPipe to detect hand gestures and map them to standard cursor and keyboard operations. Additionally, voice recognition capabilities extend system interaction by allowing users to issue verbal commands for commonly used functions.

By eliminating the need for external devices or complex hardware setups, the proposed solution offers a costeffective and scalable alternative for controlling computing systems. It empowers users—particularly those with mobility limitations—to perform essential laptop functions such as cursor navigation, clicking, and typing through simple gestures and speech inputs. Real-time image processing techniques including filtering, thresholding, and feature extraction ensure the system operates efficiently under varied lighting and environmental conditions.

This gesture-based virtual control system not only enhances accessibility but also broadens the scope of HCI applications in fields like education, industrial automation, gaming, and assistive technology.

Through this work, we contribute to the growing body of research in inclusive and immersive human-computer interaction by offering a functional prototype that bridges the gap between physical ability and digital access.

II. LITERATURE REVIEW

[1] Amardip Ghodichor, Binitha Chirakattu[2015] has proposed Virtual Mouse using Hand Gesture and Color Detection In today's technological world, many technologies are continually evolving. One such intriguing concept is the Human-Machine Interface. On a wired mouse, for example, there is no option to increase the limit. To utilise a wireless mouse, the computer must have Bluetooth hardware and a Bluetooth dongle installed. The suggested technology, which relies on gesture recognition instead, will have no such limitations. The main technologies used in this study were object detection, image processing, and colour identification using "Sixth sense technology." Sixth sense technology is a group of wearable devices that act as a gesture-based interface between the physical and digital worlds. The idea is to move the mouse pointer on the screen using only finger motions rather than needing any hardware, such as a mouse. This is known as gesture recognition.

[2] Rhitivij Parasher,Preksha Pareek [2016] event triggering using hand gesture using open cv has proposed Gesture recognition and pattern recognition are advancing at an exponential rate in the passing years. The study of hand gesture recognition has proven exciting. Hand gesture recognition technology allows humans to interact with computers in a fresh, natural, and imaginative way. Gesture recognition might be beneficial in circumstances such as human-computer interaction, sign language, and game play. In human-computer interaction and robotic technologies, hand gesture recognition is frequently employed. The mouse and keyboard are now utilised as human-computer interfaces, but hand and eye actions will soon replace them. We may use this app to count the number of fingers on our hand. To trigger on an event, we can utilise predefined gesture situations. Hand gesture recognition might provide a low-cost interface for engaging with virtual objects in the future.

[3] Ahemad Siddique, Abhishek Kommera, DivyaVarma [2016] has proposed Simulation of Mouse using Image Processing Via Convex Hull Method. This research shows how to control the cursor's position without using any electrical equipment. Various hand gestures will be used to perform actions such as clicking and dragging things. The recommended system will just require a camera as an input device. The proposed system will necessitate the usage of OpenCV, Python, and other technologies. The output of the camera will be displayed on the system's screen so that the user may fine-tune it. The Python needs that will be used to develop this system are NumPy, math, wx, and mouse.

[4] Abhilash S S1, Lisho Thomas2, Naveen Wilson3, Chaithanya C4[2018], has proposed virtual mouse using hand gesture. This paper introduces a revolutionary webcam vision-based cursor control system that recognises hand motions collected by a camera using a colour detection method. While holding colour caps or tapes, the user may move the computer cursor around with their hand, and left click and drag will be done using various hand movements. It may also transmit files between two systems that are connected by a network. The proposed system consists of a sensor and a low- resolution camera that can track the user's hand carrying colour caps in two dimensions. The system will be built with Python and OpenCV. Hand gestures are the most natural and straightforward manner of communication. The output of the camera will be shown on the monitor. Color detection will be utilised to determine the shape and placement of the gesture. The file transmission method is implemented using Python server programming.

[5] Abdul Khaliq and A. Shahid Khan [2015], has proposed Virtual Mouse Implementation using Color Pointer Detection, International Journal of Electrical Electronics &Computer Science Engineering, Hand gestures were employed to control mouse cursor movement and click occurrences in a Human Computer Interaction (HCI) approach. A camera was used to capture hand motions using a colour detection algorithm. The goal of this technology is to create a low-cost virtual human computer interaction device using a Web Camera.

[6] Erdem, E. Yardimci, Y. Atalay, V. Cetin, A. E[2002], has proposed Computer is vision-based mouse They describe a computer vision-based mouse that use a camera to steer and operate a computer's or computerised system's pointer. The user just moves the mouse-shaped passive device placed on a surface inside the camera's viewing area to move the pointer on the computer screen. Computer vision algorithms are widely used to
analyze video captured by a camera, and the computer moves the pointer in response to mouse movements. The
computer vision mouse has zones that correspond to clickable buttons. To click a button, the user just places his or her finger over one of these locations.

[7] Hojoon Park [2008] has proposed "A Method for Controlling the Mouse Movement using a Real Time Camera This experiment shows how a real-time camera may be used to control mouse movement. Changing mouse components, such as adding more buttons or altering the tracking ball's location, was a common remedy in the past. Rather, we recommend that the hardware design be changed. We show how to utilise a camera and computer vision methods like image segmentation and gesture recognition to control mouse functionalities (left and right clicking, double-clicking, and scrolling). This article will show you how to construct a mouse control system from scratch.

[8] ShanyJophin, Sheethal M.S, Priya Philip,TM Bhruguram[2018] has proposed Gesture Based Interface Using Image and Motion Comparison With the use of a real time camera This study proposes a ground-breaking method to mouse movement and function execution. In this instance, we advise that the hardware design be changed. The bulk of earlier methods relied on changing mouse component properties like the tracking ball's location and adding additional buttons. We utilise a camera, coloured material, image comparison technology, and motion detection technology to control mouse movement and fulfil its operations.

III. METHODOLOGY

The development of the system for activating applications without using a mouse or keyboard follows a structured and iterative methodology to ensure functionality, reliability, and user accessibility.

1. Research & Requirement Analysis

The project begins with in-depth research on existing gesture and accessibility-based systems. Various methods, tools, and user needs are analyzed to gather clear and practical system requirements. This phase helps in understanding the expectations and setting achievable goals for the system.

2. Technology Selection

After defining the requirements, the next step involves selecting the appropriate technologies. OpenCV is chosen for its powerful image processing capabilities, while MediaPipe is selected for its efficient hand and facial landmark tracking, enabling accurate gesture recognition in real time.

3. System Design

In this phase, the system's architecture is planned. Components such as gesture detection, facial tracking, machine learning model integration, and application control modules are designed. The design ensures modularity, so each part can be developed and tested independently.

4. Implementation

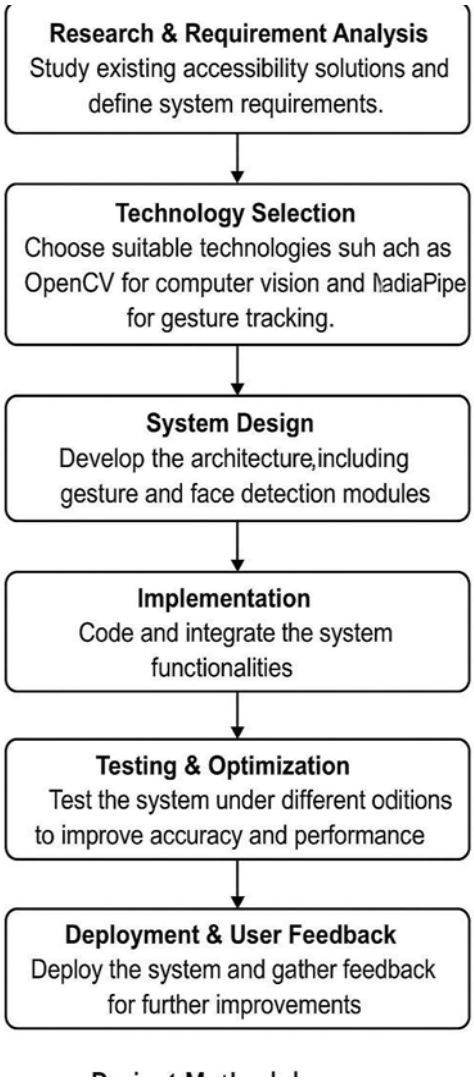
The system is implemented by writing code for the designed modules and integrating them into a cohesive unit. Key functions like capturing webcam input, detecting gestures, and triggering application actions are developed using Python and relevant libraries.

5. Testing & Optimization

The system is rigorously tested under various conditions—such as different lighting, backgrounds, and hand positions—to evaluate its performance. Based on test results, optimizations are made to enhance the system's speed, recognition accuracy, and reliability.

6. Deployment & User Feedback

Finally, the system is deployed in a real- world environment where users interact with it hands-free. Feedback is collected from users to identify areas of improvement, usability issues, and suggestions, which guide future updates and feature enhancements.in easily marking all the landmarks of a face such as eyes, ear, nose.



3) *Mouse pointer controlling*

After marking all the landmarks, these landmarks are then used to control the movement of the mouse. Facial landmarks are used to control the movement of the mouse. As the face moves so does the mouse.

4) *Controlling the mouse click events*

Landmarks assigned to the eyes help in click events of the mouse. Eye Aspect Ratio (EAR) is the simplest feature takes the most advantage for detecting the eye blinks and winks. As the EAR as reduces below the threshold for a certain period the click event is triggered. The movement of the user's face for the last frames and performs a click operation using function if the movement is within a certain threshold. Right and left clicks are executed according to the movement is within a certain threshold.

B. Virtual Keyboard:

1)*Hand tracking module*

A Hand Detector module is made using Mediapipe and OpenCV for tracking the movement of the hand. Mediapipe helps in detecting and tracking of the hand. It involves two stages palm detection to detect hand and provide a cropped image and then hand landmark identification to provide 21 landmarks of the hand such as wrist, middle thumb, upper thumb, middle index finger.

IV.PROPOSED SYSTEM

A. Virtual Mouse:

1) *Face and eye detection*

Haar-Cascade is used for detecting the human face and eyes of the user. Haar-Cascade distinguishes between the facial and non-facial areas of the user by using number of steps and images of faces .

2) *Face and eye landmarking technique*

After extracting features from the face of the user Mediapipe framework is used to landmark all the structural features of the face. Mediapipe helps

2) *Video capturing*

Images are captured from the webcam input and the Mediapipe is able to detect the landmarks of the hand and provide a live output displaying all the landmarks of the hands.

3) *Virtual keyboard*

Virtual Keyboard displaying all the functionalities and keys of a real keyboard is displayed on the same screen as the live hand tracking screen as to trigger click events of the keyboard.

4) *Keyboard functioning*

By creating hand tracking module using OpenCV and Mediapipe we are able to track the movement of the hand on the virtual keyboard. X and Y coordinates each listed

points are used to trigger the keys. Pinching of fingers on a specific key on the hovering

keyboard keys results in input of that key. Euclidean Distance formula is used to calculate the distance between the tip of index finger and tip of the middle finger. The formula is given by (1) :

$$d = \sqrt{(x2 - x1)^2 + (y2 - y1)^2}$$

where (x1, y1) and (x2, y2) are the coordinates for tip of fingers. A detector.findDistance method is called with parameters corresponding to tip of index and middle finger

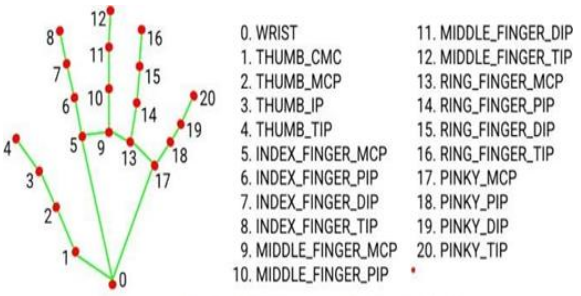


Fig. 1. MediaPipes recognized hand landmark

A. Speech-to-Text

A graphical user interface (GUI) is created for a speech-to text converter application using the tkinterlibrary. A chatbot is created that uses natural language processing techniques. The chatbot takes user input and generates a response based on the input. A Python script recognize speech from a microphone input. The script first initializes the recognizer object from Speech Recognition. Then script enters into an infinite loop that continuously listens for user input from a microphone, adjusts for ambient noise levels, and attempts to recognize the speech using Google's speech recognition API. If successful, the recognized text is converted to lowercase, printed to the console.

System Architecture

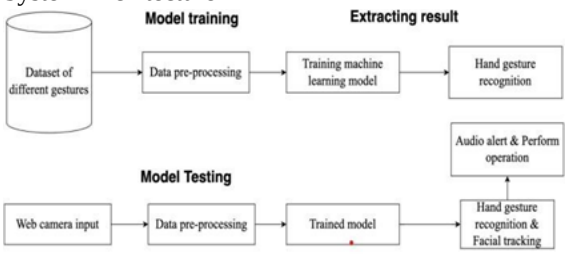


Fig.2. : System Architecture

V. RESULTS AND DISCUSSION

We evaluated the system's ability to accurately detect various hand gestures and trigger their corresponding application- level commands. The following gestures were successfully recognized and mapped to system actions, confirming the effectiveness of our gesture recognition algorithm

Recognized Hand Gestures

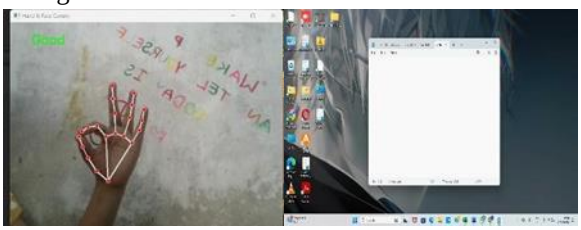


Fig. 3.: Hand Gesture Recognition to Launch Notepad Application

- The image shows a hand gesture being recognized by the system using landmark detection.
- This specific gesture is mapped to open the Notepad application automatically
- .It demonstrates an accessible, hands-free way to control basic laptop functions.

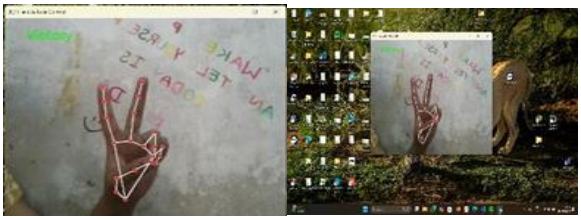


Fig.4.: Hand Gesture Recognition to Snap a Screen Shot

- The image shows a “Victory” hand gesture being recognized by the system using hand landmark tracking.
- This specific Gesture is programmed to take a screen shot on the laptop.
- It provides an easy and contactless way to capture the screen using just hand signs.

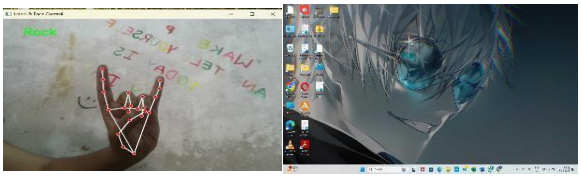


Fig.5.: Hand Gesture Recognition to Close Notepad Application

- The image displays the “Rock” hand gesture, accurately identified by the system using hand landmark detection.
- This gesture is assigned the function of closing the Notepad application on the laptop.
- It enables a quick and touch-free way to exit applications, enhancing accessibility and user convenience.

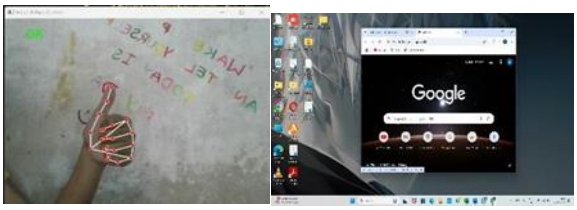


Fig.6. : Hand Gesture Recognition to Open Chrome Application

- The image shows the “OK” hand gesture recognized by the system through precise landmark detection.
- This gesture is programmed to launch Google Chrome on the laptop.
- It provides a smooth, gesture- based method to access the web without needing a keyboard or mouse.

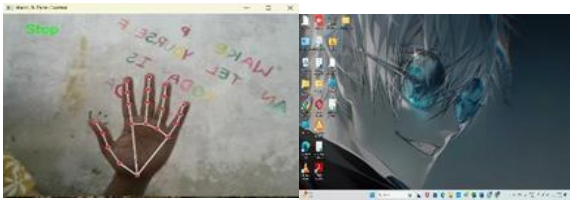


Fig.7.: Hand Gesture Recognition to Close the Chrome

- The image illustrates the “Stop” hand gesture, which has been recognized by the system using hand landmark detection.
- This gesture is programmed to close Google Chrome on the laptop.
- It provides a straightforward and gesture-based approach to terminate the browser, enabling users to interact with the system in a more natural, hands-free manner.

VI. CONCLUSION

Creating a system that would combine an Facial Tracker and hand gesture detection in lieu of the mouse and keyboard. This covers keyboard, mouse cursor movement, and additional keyboard features like printing alphabets. The suggested system may be used to execute some keyboard functions by detecting and interpreting hand gestures and hand tips using a webcam or a built-in camera. The recommended approach can also detect facial and eye movements, which enables it to control mouse operations and provide a realistic user interface. This system requires extremely little care and doesn't need a person to operate it. When typing quickly, this would enhance the results. Additionally, this method may be further improved for usage with a smart television, a night mode and regional language feature may also be added.

REFERENCES

[1] Amardip Ghodichor, Binitha Chirakattu 2015 “Virtual Mouse using Hand Gesture and Color Detection”, Volume 128 – No.11.

[2] Rhitivij Parasher,Preksha Pareek 2016,” Event triggering Using hand gesture using open cv”, volume -02, page 5673- 15676.

[3] AhemadSiddique, Abhishek Kommera, DivyaVarma 2016, “Simulation of Mouse using Image Processing Via Convex Hull Method”, Vol Coimbatore, Tamilnadu, India, 2017” Virtual Mouse Using Hand Gesture Recognition”, Volume 5 Issue VII

[4] Abhilash S S1, Lisho Thomas2, Naveen Wilson3, Chaithanya C4 2018,” VIRTUAL MOUSE USING HAND GESTURE”, Volume: 05 Issue: 04.

[5] Abdul Khaliq and A. Shahid Khan, “Virtual Mouse Implementation Using Color Pointer Detection”, International Journal of Electrical Electronics & Computer Science Enginring, Volume 2, Issue 4, pp. 63-66

[6] Erdem, E. Yardimci, Y. Atalay, V. Cetin, A. E., 2002, “Computer vision-based mouse”, Acoustics, Speech, and Signal Processing, Proceedings (ICASS), IEEE International Conference.

[7] Hojoon Park 2008, “A Method for Controlling the Mouse Movement using a Real Time Camera”, Brown University, Providence, RI,

USA, Department of Computer Science

[8] ShanyJophin, Sheethal M.S, Priya Philip, T M Bhruguram 2012, "Gesture Based Interface Using Motion and Image Comparison", International Journal of Advanced Information Technology (IJAIT) Vol. 2, No.3.