Hand gesture and voice-controlled mouse for physically challenged using computer vision

Aarti Morajkar*[1] , Atheena Mariyam James [2] , Minoli Bagwe[3] , Aleena Sara James[4] , Aruna Pavate[1]

[1]*University of Mumbai, St Francis Institute of Engineering, Information Technology, India, morajkaraarti41@gmail.com, arunaapavate@gmail.com, atheenajames96@gmail.com, minolibagwe@gmail.com, aleenajames998@gmail.com*

Keywords
HCI
Gesture
AI
Media pipe
Virtual Mouse

Abstract
A Human-Computer Interface (HCI) is presented in this paper to allow users to control the mouse cursor with hand gestures and voice commands. The system uses computer vision EfficientNet B4 architecture with no code ml to identify different hand gestures and map them to corresponding cursor movements. The objective is to create a more efficient and intuitive way of interacting with the system. The primary purpose is to provide a reliable and cost-effective alternative to existing mouse control systems, allowing users to control the mouse cursor with hand gestures and voice commands. The system is designed to be both intuitive and user-friendly, with a simple setup process. The highly configurable system allows users to customize how it works to suit their needs best. The system's performance is evaluated through several experiments, which demonstrate that the hand gesture-based mouse control system can accurately and reliably move the mouse cursor. Overall, this system can potentially improve the quality of life and increase the independence of individuals with physical disabilities.

Introduction

Artificial intelligence is putting intelligence to make machines intelligent and capable of performing logical tasks designed by humans. Computer vision is part of AI that uses image samples to train machines. Computer vision provides different solutions such as disease prediction [1-2], landmine detection [3], designing adversarial samples to make machine learning models more robust [4-5], Lip reading recognition [6], and many more.

AI has a massive impact on people with disabilities to improve their lifestyles, providing the same access and services regardless of their disabilities. Gesture recognition is a technology that interprets hand gestures as commands using images. The voice assistant interface enables the hands-free operation of digital devices. This work aims to develop a new Human-Computer Interaction System that utilizes natural and intuitive hand gestures and voice commands rather than external mechanical devices such as a mouse. The proposed research introduces a novel system that utilizes hand gestures and voice commands to facilitate computer mouse movements for users. Voice assistants are hands-free and require minimal effort, allowing fast response times. This system benefits teachers, clinicians, and other users who can benefit from the hands-free operation and physically challenged people.

Many HCI systems capture human biological information as input, such as bioelectricity and speech signals, resulting in richer HCI modes. These new interactive methods made the HCI process more user-friendly and convenient. The field of human-computer interaction improved in terms of branching and interaction quality. Many researchers concentrated on using multimodality, intelligent adaptive interfaces rather than command/action-oriented ones, and active rather than passive interfaces instead of conventional interfaces [7].

This research aims to develop a cutting-edge Human-Computer Interaction System that simplifies the usage of natural and intuitive hand gestures and voice commands rather than relying on an external mechanical device like a mouse. Our proposed system utilizes hand gestures and voice assistant technology to enable users to efficiently

control computer mouse movements, with the benefits of hands-free, effortless operation and speedy response times. This system has potential applications in various fields, such as education, healthcare, and defense, to enhance user experience and accessibility. Specifically, this system can benefit individuals with physical disabilities, in-car systems, and military operations. The objectives of the proposed system are:

1. To replace direct mouse clicks and points with gestures to control computers and other devices to simplify completing tasks.
2. To offer a cost-effective alternative to existing mouse control systems by eliminating the need for costly hardware such as additional sensors and special controllers using a deep learning model.

The remaining work is organized as follows: section II discusses the related work, section III describes the methodology, section IV concludes the work.

Literature Review

In recent years, a growing interest has been in developing new human-computer interaction (HCI) systems that replace traditional input devices such as the mouse with more natural and intuitive alternatives. One such alternative is hand gesture-based mouse control, which allows users to control cursor movements and perform mouse functions using hand gestures. In this paper, we present a review of the current state of the art in hand gesture-based mouse control, including recent developments in gesture recognition algorithms, sensing technologies, and applications of this technology in various fields.

Kabid et al. proposed [7] to create a novel mouse cursor control system that employs a webcam and a color-detecting technique. The system records every frame the webcam captures until the project is completed by implementing an infinite loop. Color-caught frames from the webcam captured frames are used to detect the color pixels on the fingertips. The distance between two detected colors is calculated using the OpenCV function. For clicking events, the proposed system uses close gestures. However, the system's efficiency could be improved due to the difficulties and complexity associated with background interference.

Rokhsana et al. [8] proposed a real-time vision-based gesture-controlled mouse system. It employs color-based image segmentation for detecting hands, and contour extraction is performed to obtain the boundary information of the desired regions. The system uses a MATLAB function for moving operations, which calculates the centroid of the hand region. This approach is not limited to only controlling a mouse; it can control other devices such as televisions, robots in dangerous nuclear reactors, and other industrial setups. The system's sensitivity to surrounding noise and brightness can also be increased.

Kollipara et al. [9] proposed a system that utilizes libraries such as OpenCV, NumPy, and sub-packages. The model is built using computer vision techniques, and the detection and movement of the mouse are based on color fluctuations. The color detection model can be designed to identify a particular color from a colored image, which can improve the system's accuracy.

Reddy et al. [10] proposed a model for recognizing motions, detecting fingers, and controlling mouse operations. The OpenCV library is used for image processing, and the PyAutogui module is used for mouse control. The algorithm's implementation involves two different approaches for mouse control: one using color caps and the other recognizing gestures made with bare hands. It involves integrating the video and processing the photos through backdrop removal. Background subtraction helps by ignoring steady items and only considering foreground objects. Fingertip detection includes finger guessing, circle recognition, and color identification. Gesture recognition involves identifying the skin tone, detecting contours, forming convex hulls, and inferring the gesture.

Sugnik et al. [11] proposed a technology that uses hand gesture recognition and image processing to create a virtual mouse and keyboard. The mouse operates using a convex hull technique, where gestures are detected or recorded and used to map the mouse's functionalities. The keyboard function uses a hand position system that records the user's hand position in a video. However, the Convex Hull algorithm may encounter issues and lose accuracy if there is external noise or flaws within the webcam's operational range.

Shibly et al. [12] aims to develop a hand gesture-based virtual mouse system that allows users to control a computer cursor using hand gestures instead of a traditional physical mouse. The methodology used in this study involved designing and developing a prototype system that captures and processes the hand gestures of the user using a camera and a machine learning algorithm. The work describes the various components of the system, including the hardware and software used and the algorithms utilized for hand gesture recognition.

Sharma et al. [13] used video processing techniques to track the position of the user's hand and translate its movements into corresponding movements of the computer cursor. To achieve this, the authors used a computer vision algorithm called skin color segmentation to detect the user's hand from the video stream. The authors applied a motion estimation algorithm based on the Lucas-Kanade method to track the movement of the hand. The authors also used a machine learning algorithm called K-Nearest Neighbour (KNN) to recognize hand gestures. This algorithm classifies hand gestures based on the fingers' and palm coordinates. The authors trained the algorithm using a dataset of hand gesture images and achieved a recognition rate of 95%.

Mishra et al. [14] used a deep convolutional neural network (CNN) called YOLOv3 to detect and localize the fingertips in the video frames. The authors used a custom-built data collection system that captured egocentric video of a user's hand performing various gestures. The annotated frames with fingertip locations used this annotated data to train and evaluate the YOLOv3 model. The proposed system showed promising results in terms of accuracy and efficiency. It could be applied to various applications involving hand gesture recognition, such as virtual or augmented reality interfaces.

The reviewed work has highlighted several issues and challenges related to hand gesture-based mouse control systems. For instance, one study [10] identified the problem of the model's sensitivity to specific color detection, leading to detection errors. Another study [11] reported limitations in detecting hand movements in a pre-defined zone and the lack of advanced mouse functionalities. Additionally, the system's accuracy is affected by various lighting conditions, further reducing the effectiveness of color and shape-based algorithms. To address these challenges, the proposed approach by the authors provides solutions that improve accuracy and efficiency and provide more advanced mouse functions for users.

Methodology

Gesture-controlled virtual mouse implementation using deep learning involves creating a pipeline to detect hand gestures and map them to mouse actions. The following are the steps:

1. Collection and processing of the Data: - This is the initial stage of gathering information on the hand motions to operate the virtual mouse. The collected images are transformed into tensors of nodes. The data is preprocessed to remove pertinent details like hand position and orientation before being captured using a depth sensor or camera.
2. Gesture Recognition Model training: - The model is trained using examples of labeled hand movements. A machine learning model recognized the hand motions.
3. Model: A Convoluted Neural Network (CNN) using the Efficientnet4 model is used for gesture recognition. The EfficientnetB4 trained on a custom dataset to accommodate customized gestures.
4. Run the model and Map Gestures to Mouse Actions: - After building the pipeline, it is executed on a device to detect hand gestures in real time. The detected gestures were mapped to mouse actions, such as clicking, scrolling, or moving the cursor.
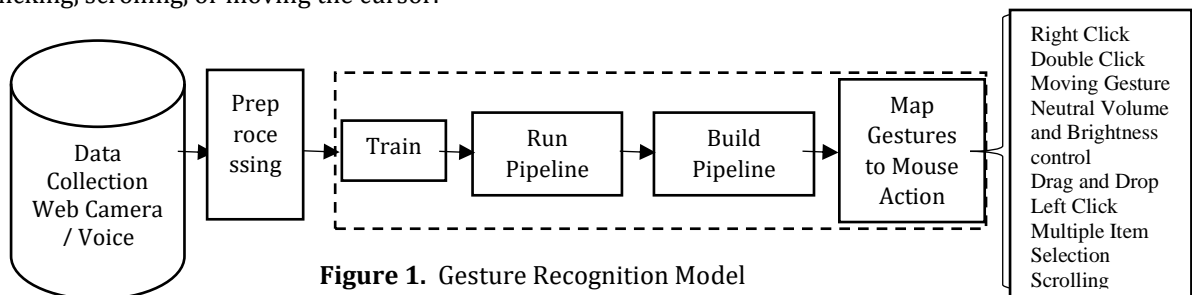

Figure 1. Gesture Recognition Model

A voice assistant can be added to a gesture-controlled virtual mouse implementation using MediaPipe. To do this, a voice recognition module can be included in the pipeline to detect and recognize voice commands from the user. The recognized voice commands can then be mapped to mouse actions or other actions, such as opening a file or launching an application. Figure 1 illustrates the implementation of a gesture-controlled virtual mouse with a voice assistant using MediaPipe. The hand gestures are captured using a depth sensor or a camera and preprocessed to extract relevant features such as hand position and orientation. The gesture recognition model is trained to recognize hand gestures from the collected data. It received the hand gesture as input and output as recognized gestures. The mouse action mapping module mapped the recognized hand gestures to mouse actions such as clicking, scrolling, or moving the cursor. It receives recognized hand gestures and the tracked hand position and orientation as input and outputs the mapped mouse actions. The virtual mouse module simulated the mouse's actions on the computer by accepting mouse actions as input. The voice recognition module detects and recognizes voice commands from the user. The implementation involved capturing and preprocessing hand gestures, recognizing the hand gestures using a machine learning model, tracking the hand in the video stream, mapping the recognized gestures and voice commands to the mouse and other actions, and executing the mapped actions on the computer.

Results and Discussion

The hand gesture and voice recognition system incorporate ten gestures: neutral gesture, moving cursor, left click, right-click, double click, scrolling, drag and drop, multiple item selection, volume control, and brightness control. The voice assistant performs launch/stop gesture recognition, and content search on Google, identifies a location, navigates files, displays the current date and time, copies and pastes, sleeps/wakes up, and exit actions. In the proposed system, authors aimed to enhance human-computer interaction using computer vision.
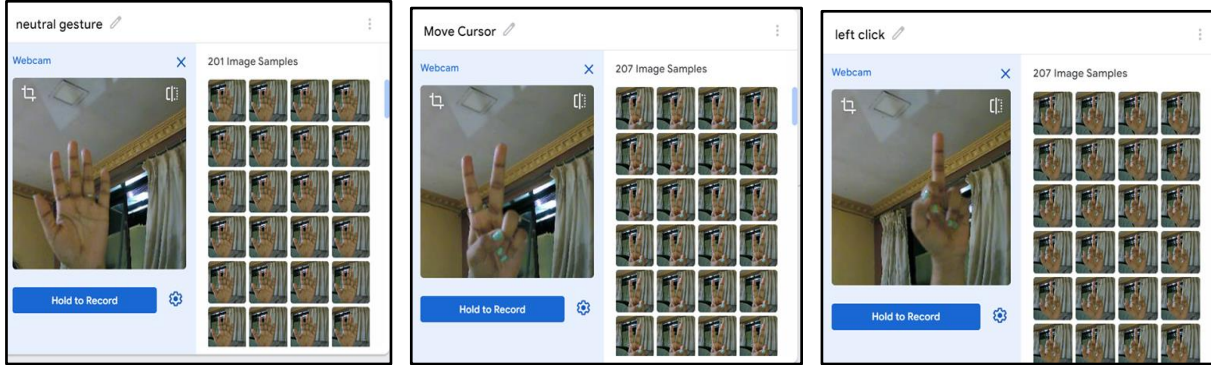

Figure 2. Hand Gestures incorporated by Gesture Recognition System

The webcam is positioned at various distances from the user to monitor hand motions and gestures to detect fingertips as shown in Figure 2. Gesture's ability is assessed under diverse lighting conditions such as bright light settings, low-light configurations, at a much farther distance from the camera, at a closer distance from the camera, with a left hand, right hand, both hands in camera, different backgrounds, and different hands of individuals of varying ages. The Voice Assistant is tested by providing diverse input via the mic and executing various functions such as location, file navigation, current time and date, copy and paste, sleep/wakeup, google search, and start and exit under various conditions.
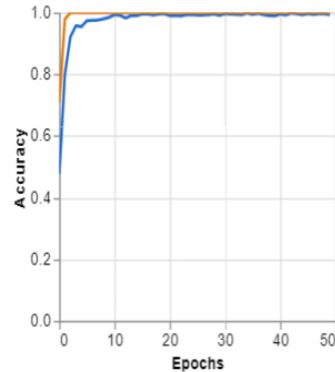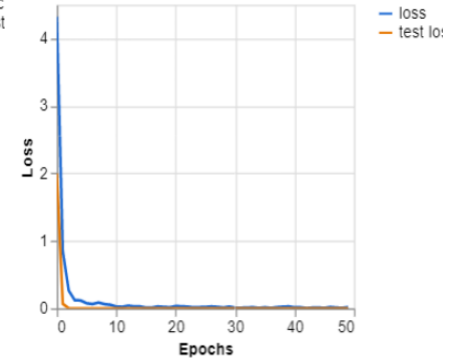

Figure 3. Performance of the model represented by accuracy per class and loss obtained by the model

It is observed that every mouse action gives a few seconds of delay, but apart from that, all the gestures had excellent and high accuracy for all the classes as shown in the Figure 3. The hand gestures are captured using an automated training machine learning model, showing promising results. Using hand gestures to control a mouse can increase productivity and ease of use, particularly for individuals with disabilities or those who find traditional mouse controls difficult. The automated training machine learning model accurately detects and classifies hand gestures, allowing for smooth and precise cursor control. While further research and testing may be necessary to optimize the system's performance, the results thus far suggest that a hand gesture-controlled mouse could become a valuable tool for computer users in the future.

Conclusion

Human-Computer Interaction was a rapidly evolving technological sector. New technological advances were produced every year, and new efforts were taken toward seamless, natural contact between the computer and the user. It has progressed from the traditional keyboard and text-based interface to the more powerful mouse and

touch-based interactions. With this study, we want to move forward to the next phase of virtual touchless interactions. This work developed a system for controlling the mouse cursor with a real-time camera. The technology was based on computer vision techniques such as CNN and could perform all mouse functions. However, due to the wide range of lighting and skin colors, it was impossible to obtain consistent results.

This method improves presentations for physically disabled individuals and enhances reliability. The system provides a comfortable PC and laptop experience for physically challenged persons. Future research involves eye movements to control mouse actions for those who cannot use their hands and introducing more functions to improve system performance.

References

1. Pavate, A., Mistry, J., Palve, R., & Gami, N. (2020). Diabetic Retinopathy Detection-MobileNet Binary Classifier.
2. Pavate, A., & Ansari, N. (2015). Risk Prediction of Disease Complications in Type 2 Diabetes Patients Using Soft Computing Techniques. 2015 Fifth International Conference on Advances in Computing and Communications (ICACC), 371-375.
3. Kumar, A., Pavate, A., Abhishek, K., Thakare, A. R., & Shah, M. (2020). Landmines Detection Using Migration and Selection Algorithm on Ground Penetrating Radar Images. 2020 International Conference on Convergence to Digital World - Quo Vadis (ICCDW), 1-6.
4. Pavate, A., & Bansode, R. S. (2020). Performance Evaluation of Adversarial Examples on Deep Neural Network Architectures.
5. Pavate, A., & Bansode, R. (2023). Design and Analysis of Adversarial Samples in Safety–Critical Environment: Disease Prediction System. In: Gupta, M., Ghatak, S., Gupta, A., Mukherjee, A.L. (eds) Artificial Intelligence on Medical Data. Lecture Notes in Computational Vision and Biomechanics, vol 37. Springer, Singapore.
6. Shi, B., Hsu, W., Lakhotia, K., & Mohamed, A. (2022). Learning Audio-Visual Speech Representation by Masked Multimodal Cluster Prediction. ArXiv, abs/2201.02184.
7. 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. https:doi.org/10.1109/ICASERT.2019.8934612.
8. Titlee, R., Rahman, A. U., Zaman, H. U., & Rahman, H. A. (2017). A novel design of an intangible hand gesture controlled computer mouse using vision based image processing. In 2017 3rd International Conference on Electrical Information and Communication Technology (EICT) (pp. 1-4). Khulna, Bangladesh.
9. Varun, K. S., Puneeth, I., & Jacob, T. P. (2019). Virtual Mouse Implementation using Open CV. In 2019 3rd International Conference on Trends in Electronics and Informatics (ICOEI) (pp. 435-438). Tirunelveli, India.
10. Reddy, V. V., Dhyanchand, T., Krishna, G. V., & Maheshwaram, S. (2020). Virtual Mouse Control Using Colored Finger Tips and Hand Gesture Recognition. In 2020 IEEE-HYDCON, Hyderabad, India (pp. 1-5).
11. Chowdhury, S. R., Pathak, S., & Praveena, M. D. A. (2020). Gesture recognition based virtual mouse and keyboard. In 2020 4th International Conference on Trends in Electronics and Informatics (ICOEI)(48184) (pp. 585-589). Tirunelveli, India.
12. Sharma, Neeta & Gupta, Aviral. (2020). A Real Time Air Mouse Using Video Processing. International Journal of Advanced Science and Technology, 29, 4635 - 4646.
13. Mishra, P., & Sarawadekar, K. (2019, December). Fingertips detection in egocentric video frames using deep neural networks. In *2019 International Conference on Image and Vision Computing New Zealand (IVCNZ)* (pp. 1-6). IEEE.