Virtual Keyboard-Mouse in Real-Time using Hand Gesture and Voice Assistant

Dr.Shimaa[1] Othman, Dr.Hany Maher[1], Yosr Mansour[1]

[1] Higher Institute of Computers and Information Technology, Computer Depart., El. Shorouk Academy, Cairo, Egypt

Email:dr.shimaa.osman@sha.edu.eg
Email:dr.hany.maher@sha.edu.eg
Email:yosr.mansour@sha.edu.eg

Abstract

The development of computer applications has made our daily tasks more convenient and efficient. One of the latest trends in this area is the integration of voice assistants and virtual input devices. This approach is particularly useful for individuals with mobility impairments or for situations where manual interaction with a computer is not possible. The application of computer vision and AI technologies has made this possible. Computer vision enables machines to interpret and understand visual data, such as the movements of the human body, while AI provides the decision-making abilities required to execute commands based on the input received. The integration of voice assistants and virtual input devices into computer applications has the potential to revolutionize the way we interact with technology. The combination of computer vision and AI technologies is driving this innovation, and it will be interesting to see how it continues to evolve in the future. This desktop application offers a comprehensive solution for users seeking a more accessible and efficient computing experience. By integrating a voice assistant, virtual mouse, and virtual keyboard, it enables individuals with physical disabilities or those preferring alternative input methods to fully utilize their computers, improving productivity and enhancing their overall digital experience. We use Python, Media Pipe, and OpenCV. The Media-Pipe library provides features that boost the model's efficacy and is especially useful for AI projects. The user will be able to control the computer cursor using different hand gestures, type on the virtual keyboard while holding colored caps or tapes and left-click and drag items.

1. INTRODUCTION

Introducing our cutting-edge desktop application that integrates hand gesture and voice assistant technology with three exciting features: voice assistant, virtual mouse, and virtual keyboard. With our intuitive hand gesture controls, you can easily navigate your desktop and interact with our voice assistant feature hands-free. Our virtual mouse offers a convenient alternative to physical mouse usage, while our customizable virtual keyboard ensures a seamless typing experience. Try our innovative application today and revolutionize the way you interact with your desktop.

In recent years, the subject of human-computer interaction has seen amazing advancements aimed at improving the user experience and accessibility. One of the most interesting developments is the creation of a virtual keyboard-mouse system that works in real time with hand gestures and voice instructions. This study investigates the design, implementation, and prospective applications of such a system, which incorporates cutting-edge computer vision, machine learning, and natural language processing technologies. By interpreting hand gestures collected by a camera and processing spoken commands through a voice assistant, this technology provides a touch-free and intuitive interface for interacting with digital devices. The combination of these modalities creates a more ergonomic and accessible solution, which is especially advantageous for individuals with physical limitations.

The purpose of this research is to offer a full overview of the Virtual Keyboard-Mouse system's technological components, user interface design, and performance evaluation. It also analyses the technology's larger implications for the future of human-computer interaction, highlighting its potential to transform how we interact with digital surroundings.

2. LITERATURE REVIEW

The integration of artificial intelligence (AI) into the field of medical imaging has ignited a wave of transformative innovation and research [1, 2]. conducts a comprehensive review of existing literature, illuminating the pivotal role of AI in revolutionizing medical image analysis and its wide-ranging applications.

2.1 Hand gesture

Hand gestures are performed by making specific hand movements that are recognized by a Hand gesture recognition is typically done using computer vision techniques and machine learning algorithms. First, the hand region is detected and tracked in real-time video using computer vision techniques like background subtraction, skin color segmentation, or a combination of both.

Many practical applications, such as virtual mice, remote controllers, sign-language recognition, and immersive gaming technologies, use fingertip detection. Therefore, one of the primary objectives of vision-based technology over the past few decades, particularly with conventional red-green-blue (RGB) cameras, has been virtual mouse control via fingertip recognition from pictures [1].

Numerous RGB-D sensor types, such the Kinect V2, or VicoVR [2], among others, can provide body tracking. Among them, the low-cost, CPU-free Kinect V2 is becoming increasingly popular these days. Convolutional neural network (CNN)-based RGB-D image-based systems have lately demonstrated exceptional performance in HCI [3,4]. However, for model rendering and assessment, these systems need high-performance GPUs and a bigger dataset.

Utilizing fingertip sensing and RGB-D pictures, a novel virtual mouse technique. Without using a mouse, gloves, or markers, the user interacted with the computer by moving their fingertip in front of a camera. The method showed off not just extremely precise gesture estimations but also useful applications[1].

2.2 Gesture Recognition Techniques

Gesture recognition forms the backbone of virtual keyboards and virtual mice operated by hand gestures. Researchers have explored diverse methodologies, including computer vision and deep learning algorithms, to enhance accuracy. Noteworthy studies include Lin et al.'s work, employing a combination of CNNs and LSTM networks for real-time hand gesture recognition [6]. Similarly, Zhang et al. proposed a method integrating depth information for precise gesture recognition [5].

2.3 Virtual Mouse using Hand Gestures

Virtual mouse systems, such as the one explored in the study by [7], represent a significant leap in HCI. By utilizing hand gestures, users can control the computer cursor intuitively. These systems commonly employ depth-sensing cameras and machine learning algorithms. The research underscores the precision and responsiveness achieved by mapping hand gestures to cursor movements, laying the foundation for seamless interaction.

Constructed a virtual mouse system using color detection [11].They used webcam for detecting mouse cursor movement and click events using OpenCV built-in functions. A mouse driver, written in java, is required as well. This system fails to perform well in rough background.

In the present work[8], a method for hand gesture identification is presented, and computer vision techniques are used to construct a virtual mouse and keyboard with hand gesture recognition. To realistically control the computer, full keyboard functions, mouse pointer movement, and click events are incorporated. All the inputs taken into consideration have their recognition and response rates computed and shown in the findings. Comparing the accuracy of the proposed strategy to other cutting-edge algorithms reveals that it performs better with a 95% accuracy.

detects the coordinates on the hand and recognizes the form and motion when the user holds their hand in front of the camera. This is accomplished through the use of the blazing palm detector model and the hand landmark model.

A multidimensional approach to HCI may be seen in the convergence of hand gestures and voice commands as shown in [9]. Users may interact with computers virtually by fusing several modalities, which improves conventional user interfaces. The study emphasizes the connection between movements and voice, demonstrating how subtle gestures combined with context-aware voice instructions allow for complex and flexible interactions.

2.4 Virtual Keyboard using Hand Gestures

A Virtual Keyboard Using Shadow Analysis [12]. This system detects keyboard, hands shadow, fingertips using colour segmentation and sobel technique. Ambient lighting conditions are required for this system. This system can analyze 3 frames per second.

A finger recognition and gesture based augmented keyboard system [13]. The system was developed using OpenCV libraries and Python. Palm detection is used for typing on the augmented keyboard. Virtual Keyboard performs based on the movement of the finger.

3. SYSTEM IMPLEMENTATION

The aim of this paper is to implement a computer application that uses alternative methods of keyboard and mouse cursor control for rehabilitation purposes, allowing stroke patients to recover from Therefore, we propose a new keyboard and mouse cursor control system based on visual and color recognition techniques that utilize hand gestures recorded by a webcam. A system, which incorporates cutting-edge computer vision, machine learning, and natural language processing technologies. By interpreting hand gestures collected by a camera and processing spoken commands through a voice assistant

Modules partitions

First, as we declared in the architecture of our project in Figure 1, we'll be using Pyqt5 to design and activate the interface of the application to make it usable on Desktop systems using Python programming language.

The figure represents a system architecture for an integrated application that combines a voice assistant, virtual keyboard, virtual mouse, and camera functionality, all within a Windows application environment. The diagram outlines how these components interact with each other, leveraging deep learning models and cloud-based platforms for enhanced functionality and training.

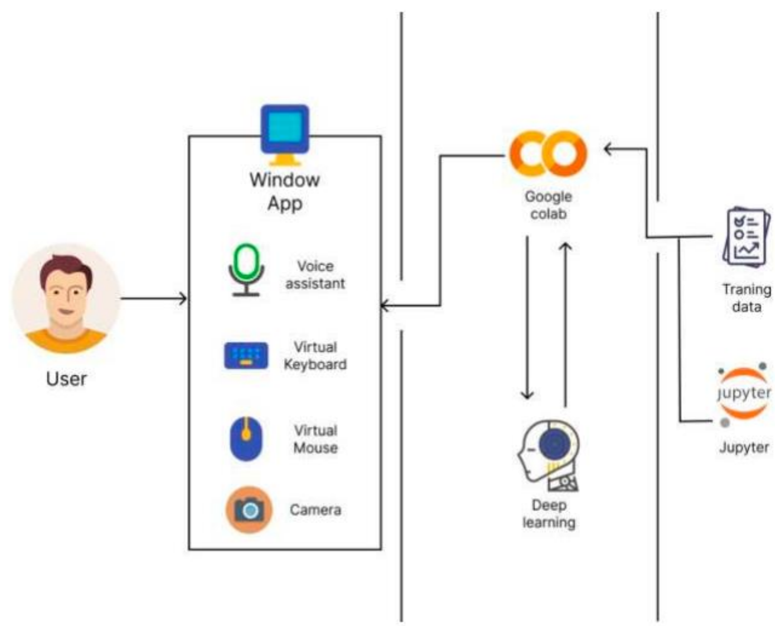

Figure 1 project Architecture

The application implementation consists of two main modules Hand Gesture and Voice Assistant

3.1 Hand Gesture Module:

Creating a virtual keyboard using hand gestures involves using computer vision techniques to detect and recognize hand gestures in real-time. Using MediaPipe is an open-source framework developed by Google that provides a comprehensive solution for building real-time multimedia processing pipelines.

a. Capture Video Stream, Process Frames and Recognize Hand Gestures

In this step, you capture frames from the webcam, convert them to RGB (as MediaPipe requires RGB input), and process the frames to detect hands using the MediaPipe Hands model.

b. BlazePalm Detector

- We train a palm detector instead of a hand detector to localize hand landmarks in real-time from a single image or video frame.
- The model is based on a two-stage approach (hand region and hand landmarks)

c. Hand Landmark Model

- After running palm detection over the whole image, our subsequent hand landmark model performs precise landmark localization of 21.
- The model has three outputs.
    - 21 hand landmarks
    - A hand flag indicating the probability of hand presence in the input image.
    - A binary classification of handedness, e.g., left, or right hand.

d. Recognize Gestures

In this step, you analyze the positions of hand landmarks to recognize specific gestures. Implement the logic for recognizing gestures like thumbs up, peace sign, etc.

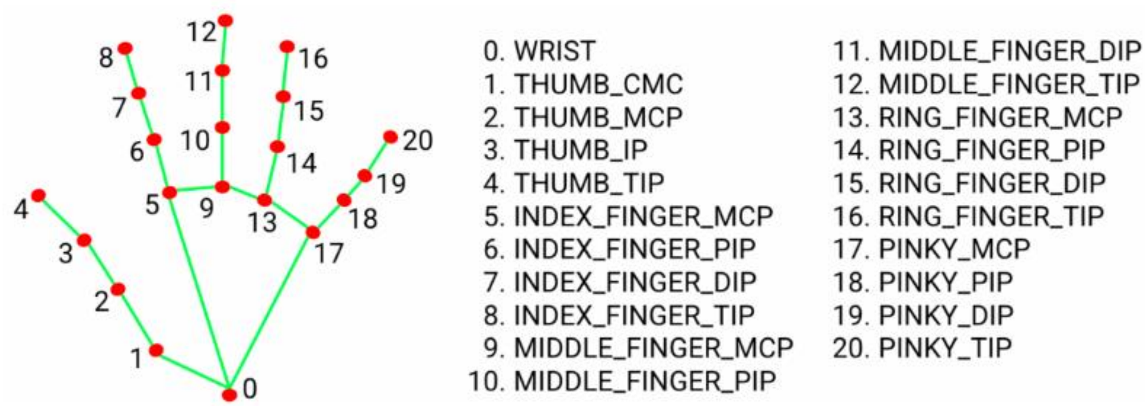

Figure 2 hand landmark

Following the successful development of our hand gesture recognition system, we implemented this technology into four specialized sub-modules to enhance user interaction and control:

3.1.2 Virtual Keyboard algorithm

System must capture images via camera. The system provides the user with typing by hand gestures such as the hardware keyboard. Here I used point number 12 to define an indicator and point No. 8 and 7 in the action process.

- The system must allow the user to "click" on the key.
- The system must display the correct output for the key entered by the user.
- Functions provided by the system to the user:
    I. Letters from A to Z
    II. Numbers from 0 to 9

III. Enter button.
IV. Delete button.
V. Spacebar
VI. Capitalization

3.1.3 Virtual mouse algorithm

- Use the camera to detect hand gestures through the 21 points on the hand.
- Know if the finger is closed (0) or open (1).
- By using points 8, 12, and 20, I can control the mouse.
- And control the action of the mouse by closing and opening points 8 and 12.
- The system provides control of the mouse by hand gestures and meets the needs of the user, such as the hardware mouse.
- Functions provided by the system to the user:

i. Click on the left mouse button.
ii. Click on the right mouse button.
iii. Move the mouse pointer right, left, down and up That’s like a hardware mouse)

3.1.4 Media control player

- Use the camera to detect hand gestures through the 21 points on the hand.
- Stop Video by points 4, 8, 12, 16 and 20.
- Video Plus Five Sec by points 8
- Video Minus Five Sec by points 8, 12
- Increase video volume by points 8, 12 and 16
- decrease video volume by points 8, 12, 16 and 20

3.1.5 Scroll

- Uses gestures to facilitate scrolling through documents and web pages, providing a seamless browsing experience.
- Using virtual mouse can be scroll in pc by points 8, 12 and 16.

3.2 Voice Assistant Module

- Allows access to the computer's mic when the user is talking.
- The user's voice is converted into text and processing is done, and then the required commands are executed.
- The system must provide the user with the ability to carry out the command requested by the user and talk to the computer by voice.
- Functions provided by the system:

a. Internet searches
b. Search inside the device for what the user wants, whether it is an audio file, video or text.
c. Date >today-yesterday -last week -last year

d. Time
e. Weather
f. Open google &YouTube.
g. Write notes.
h. Weather state
i. Send e-mails.

4. RESULTS

In this paper, we comprehensively evaluated the performance and accuracy of various interactive models, including a voice assistant, a virtual Keyboard, and a virtual mouse model. Our testing approach involved assessing a total of 15 voice assistant functions, each tested five times to ensure consistency. Similarly, we evaluated the virtual Keyboard input model's ability to recognize and process commands for typing letters and numbers, along with control functions, with each function undergoing ten trials. Additionally, we assessed the virtual mouse model's performance in executing actions, also using ten trials for each function. Overall, our tests aimed to identify strengths and areas for improvement in these models, ensuring their reliability and effectiveness in practical applications.

4.1Voice assistant model test

Name of function	input	output	task	Num of trails	Right output	percentage
open_google	Voice	No	Open the google search at the browser	5	5	100 %
open_youtube	Voice	No	Open the YouTube at the browser	5	5	100 %
speak	text	Voice	Turn the text into voice	5	4	80 %
today	Voice	Date (text)	Give you the date of the current day	5	5	100 %
yesterday	Voice	Date (text)	Give you the date of the previous day	5	5	100 %
last_year / last_week	Voice	Date (text)	Give you the date of the previous year /week from the current day	5	5	100 %
time	Voice	time (text)	Give you the time now	5	5	100 %
Weather	Voice	text	Give you the temperature degree and the state of the weather now	5	4	80 %

search	Voice	no	Search for the input text on google engine and display the results it in the browser	5	5	100 %
rec_audio	Voice	text	Turn the voice into text	5	4	80 %
Make_note	Voice		It takes a voice command and turn them into txt file	5	4	80 %
E_mail	Voice and text (Voice receiver)	No	It takes a voice commands and turn them into email and send it to specific user(receiver)	5	4	80 %
file	email Voice (file that you want to find)	List of matched files	It searches in the whole hard desk to find the matched files	5	5	100 %
virtual_mouse	Voice	No	It opens the virtual mouse program	5	5	100 %
virtual_keyboard	Voice	No	It opens the virtual keyboard program	5	5	100 %

Table 1 Voice assistant test

4.2Virtual Keyboard model test

Name of function	input	output	task	Num of trails	Right output	percentag e
Capital Letters	Video in real-time	A-Z	Write Sentences	10	10	100%
Small-Letters	Video in real-time	a-z	Write Sentences	10	10	100%
Numbers	Video in real-time	0-9	Write Numbers	10	8	80%
Backspace	Video in real-time	Backspace	Delete last character	10	9	90%
Caps lock	Video in real-time	Caps lock	Convert the letters	10	9	90%

Enter	Video in real-time	Enter	Press Enter Buttons	10	10	100%
Space	Video in real-time	Space	Make Space	10	9	90%

Table 2 Virtual Keyboard test

4.3Virtual Mouse model test

Name of function	Input	output	task	Num of trails	Right output	percentage
Right-Click	Video in real-time	Right-Click	Make Right click	10	10	100%
Left-Click	Video in real-time	Left-Click	Make Left Click	10	10	100%
Double-Click	Video in real-time	Double-Click	Make Double-Click	10	8	80%
Scroll-UP	Video in real-time	Scroll-Up	Make Scroll Up In Files & PDF	10	9	90%
Scroll-down	Video in real-time	Scroll-Down	Make Scroll Down in Files & PDF	10	9	90%
Stop-Pointer	Video in real-time	NO	Give you the date of the per	10	10	100%
Move-Pointer	Video in real-time	Mouse Move	Make the mouse move	10	9	90%

Table 3 Virtual Mouse test

5. CONCLUSIONS

The desktop application with voice assistant, virtual mouse, and virtual keyboard features opens up a world of possibilities for users. With the voice assistant, users can control their desktop through voice commands, executing tasks and commands effortlessly. The virtual mouse feature provides an alternative input method, enabling users to navigate and control the cursor without a physical mouse, enhancing accessibility and convenience. The virtual keyboard feature offers an on-screen typing solution, allowing users to input text without a physical keyboard. Together, these features empower users to interact with their desktop in new and convenient ways. They can perform various tasks, browse the web, write documents, control applications, and more, all with improved accessibility and ease of use. The application caters to different user needs, including those with physical limitations or those who prefer alternative input methods. Ultimately, this powerful combination of features promotes productivity, efficiency, and an enhanced user experience.

References

[1] Real-time virtual mouse system using RGB-D images and fingertip detection
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).
Abhilash S S, Lisho Thomas, NWCC (2018) Virtual Mouse Using Hand Gesture. International Research Journal of Engineering and Technology (IRJET)
[3]Jiang D, Li G, Sun Y, Kong J, Tao B (2019) Gesture recognition based on skeletonization algorithm and CNN with ASL database. Multimed Tools Appl 78:29953–29970
[4] Ge L, Liang H, Yuan J, Thalmann D (2018) Robust 3D hand pose estimation from single depth images using multi-view CNNs. IEEE Trans Image Process 27:4422–4436
[5] Li Y, Zhang P. Static hand gesture recognition based on hierarchical decision and classification of finger features. Sci Prog. 2022 Jan-Mar;105(1):368504221086362. doi: 10.1177/00368504221086362. PMID: 35296188; PMCID: PMC10358564.
[6] Guo, Lin & Zongxing, Lu & Yao, Ligang. (2021). Human-Machine Interaction Sensing Technology Based on Hand Gesture Recognition: A Review. IEEE Transactions on Human-Machine Systems. 51. 300-309. 10.1109/THMS.2021.3086003.
[7] R. Matlani, R. Dadlani, S. Dumbre, S. Mishra and A. Tewari, "Virtual Mouse using Hand Gestures," *2021 International Conference on Technological Advancements and Innovations (ICTAI)*, Tashkent, Uzbekistan, 2021, pp. 340-345, doi: 10.1109/ICTAI53825.2021.9673251.
[8] K. Aggarwal and A. Arora, "An Approach to Control the PC with Hand Gesture Recognition using Computer Vision Technique," *2022 9th International Conference on Computing for Sustainable Global Development (INDIACom)*, New Delhi, India, 2022, pp. 760-764, doi: 10.23919/INDIACom54597.2022.9763282.
[9] Sharma, M., Akilesh, G., Vishwaa, V. S., Sharon Femi, P., & Kala, A. VIRTUALLY CONTROLLING COMPUTERS USING HAND GESTURE AND VOICE COMMANDS.
[10] Gupta, D., Hossain, E., Hossain, M. S., Hossain, M. S., & Andersson, K. (2021). An interactive computer system with gesture-based mouse and keyboard. In Intelligent Computing and Optimization: Proceedings of the 3rd International Conference on Intelligent Computing and Optimization 2020 (ICO 2020) (pp. 894-906). Springer International Publishing.
[11] Shetty, S., Yadav, S., Upadhyay, R., Bodade, V.: Virtual mouse using colour detection (2016)
[12] Adajania, Y., Gosalia, J., Kanade, A., Mehta, H., Shekokar, N.: Virtual keyboard using shadow analysis. In: 2010 3rd International Conference on Emerging Trends in Engineering and Technology, pp. 163–165. IEEE (2010)
[13] Jagannathan, M., Surya, M., BT, A.M., Poovaraghavan, R.: Finger recognition and gesture based augmented keyboard (2018)

[14] Phursule, R. N., Kakade, G. Y., Koul, A., & Bhasin, S. (2023, August). Virtual Mouse and Gesture Based Keyboard. In 2023 7th International Conference On Computing, Communication, Control And Automation (ICCUBEA) (pp. 1-4). IEEE.
[15] Yadav, K. S., Anish Monsley, K., & Laskar, R. H. (2023). Gesture objects detection and tracking for virtual text entry keyboard interface. Multimedia Tools and Applications, 82(4), 5317-5342.
[16] Godbole, A., Gondke, V., & Devadkar, K. K. (2022, December). Hand Gesture Identification and Voice Command Based Hardware Reduction. In 2022 IEEE Pune Section International Conference (PuneCon) (pp. 1-6). IEEE.
[17] Patel, U., Rupani, S., Saini, V., & Tan, X. (2022, November). Gesture Recognition Using MediaPipe for Online Realtime Gameplay. In 2022 IEEE/WIC/ACM International Joint Conference on Web Intelligence and Intelligent Agent Technology (WI-IAT) (pp. 223-229). IEEE.