International Journal of Research Publication and Reviews

Journal homepage: www.ijrpr.com ISSN 2582-7421

Virtual Cursor Control System Using Artificial Intelligence

[1]BARATH.A, [2]Dr.C.PRIYA

[1]PG Student, Department of Computer Applications, Dr. M.G.R Educational and Research Institute
Chennai, TamilNadu, India, abarath2205@gmail.com
[2]Professor, Faculty of Computer Applications, Dr. M.G.R Educational and Research Institute
Chennai, TamilNadu, India, priya.mca@drmgrdu.ac.in

ABSTRACT—

This project advocates for an approach to Human-Computer Interaction (HCI) that involves controlling cursor movement through a real-time camera, offering an alternative to traditional methods such as manual button input or physical mouse repositioning. By utilizing computer vision technology, the system can effectively manage various mouse events, mirroring the capabilities of a physical mouse. The virtual mouse color recognition program continuously captures real-time images, subjecting them to a series of filters and conversions. Upon completion of this process, the program employs image processing techniques to determine the coordinates of the targeted color position within the converted frame. Subsequently, it compares the colors present in the frames with a predefined list of color combinations, each corresponding to different mouse functions. Upon identifying a matching color combination, the program executes the associated mouse function, translating it into an action on the user's machine.

Keywords— Human Computer Interaction (HCI) , Vitual Cursor, Artificial Intelligence , Open CV , Cursor Control

Introduction

Computer technology has significantly advanced in the last ten years and has become an essential part of our daily lives. While the mouse is the primary accessory for Human Computer Interaction (HCI), it may not always be suitable for certain real-life scenarios, such as Human Robot Interaction. This project focuses on exploring and creating a Computer Control (CC) system that utilizes hand gestures. Additionally, the webcam can be effectively utilized in various HCI applications, including sign language databases and motion controllers. The field of HCI has seen remarkable progress in gaming technologies over the years, with devices like the Microsoft Kinect and Nintendo Wii offering more natural and interactive ways to play video games. Motion controls are considered the future of gaming and have significantly boosted video game sales, as seen with the success of the Nintendo Wii. HCI through hand gestures is highly intuitive and effective for one-on-one interactions with computers, providing a Natural User Interface (NUI). Extensive research has been conducted on innovative devices and methods for cursor control using hand gestures, with applications extending beyond HCI to sign language recognition, making hand gesture recognition even more valuable. Numerous studies have explored alternative methods to the traditional computer mouse for HCI, emphasizing the need for natural and intuitive techniques that can serve as viable replacements.

SYSTEM ANALYSIS

Regarding the Application

The AI virtual mouse system serves various purposes. It can minimize the need for a physical mouse, especially in situations where using one is not feasible. By eliminating the need for devices, it enhances human-computer interaction. The proposed model boasts an impressive accuracy rate of 99%, surpassing other virtual mouse models. It has a wide range of applications. Given the current COVID-19 situation, using devices by touch is risky due to potential virus transmission. Therefore, the AI virtual mouse offers a safe alternative to control PC mouse functions without physical contact. This system can also operate robots and automation systems without the need for additional devices. Additionally, it enables the creation of 2D and 3D images through hand gestures. The AI virtual mouse is ideal for playing virtual and augmented reality games without traditional mouse devices. Individuals with hand-related issues can utilize this system to manage computer mouse functions effectively. In the realm of robotics, the proposed HCI system can effectively control robots. Moreover, in fields like design and architecture, this system can be utilized for virtual prototyping.

B. Hand Gesture Recognition

Hand gestures are categorized into two types: static and dynamic.

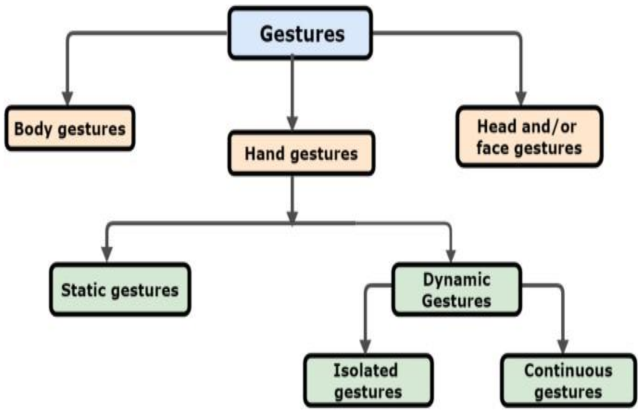

Fig. 1 Classification of Hand Gesture

The acquisition of hand gesture recognition data can be achieved through two methods:
1.camera vision
2.sensors

The initial method involves the utilization of a range of sensors or equipment that are physically connected to the user's hand or arm in order to capture the position, movement, and paths of the hand and fingers.

1. The glove-based method is capable of detecting the position, acceleration, bending of the hand & fingers, and degree of freedom of the hand. Examples of sensors used in this method include flex sensors, gyroscope, and accelerometer.

2. EMG (Electromyography) involves measuring the electrical pulses through muscles of humans to interpret the captured signal and detect finger movements.

3. Wi-Fi and radar-based methods utilize radio waves, broad-beam radar, or spectrogram to detect any changes in the strength of the signal. Vision-based approaches involve acquiring images/videos of a person's hand gestures using various cameras, such as Single cameras, Stereo-camera, and multiple camera-based systems. Additionally, light coding techniques like the projection of light can be used to obtain a 3D view of an object using sensors like PrimeSense, Microsoft Kinect, Creative Senz-3D, or Leap Motion Sensors. Invasive methods, such as body markers like colored hands, wrist bands, and marked fingers, can also be utilized.

C.System Architecture

In the proposed virtual cursor control system, I utilize the MediaPipe and OpenCV libraries to track hand gesture movements using a camera for image acquisition and hand detection and gesture recognition processing. MediaPipe, an open-source tool developed by Google, is employed for this purpose.

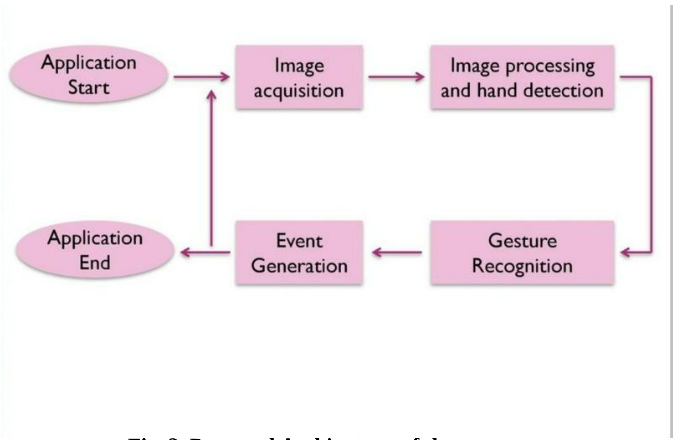

Fig. 2 Proposed Architecture of the system

The system captures images of different hand movements through cameras, which are then recognized by an AI algorithm. The recognition model is initially trained using a collection of hand gesture images to understand and identify various gestures. Once recognized, these gestures are converted into corresponding mouse movements, allowing them to be performed on the computer screen. There are various techniques, classifiers, and algorithms available to improve gesture recognition, and it is crucial to select the most effective approach to enhance human-computer interactions.
The User Interface Layer is the forefront of this system, allowing users to engage with interfaces that are customized to their preferences, whether they prefer graphical, web-based, or voice-enabled interfaces for accessibility purposes. Below this layer is the Sensor Interface layer, which connects with various input devices and sensors such as mice, touchpads, gesture recognition cameras, eye-tracking devices, and microphones for voice commands. The raw input data from these sensors is then processed in the Data Processing and Feature Extraction layer, which utilizes machine learning algorithms to analyze user behavior and extract relevant features.

The core of the system is the Artificial Intelligence Engine, where advanced algorithms like supervised or reinforcement learning continuously learn from user interactions to optimize cursor control. The outputs from the AI engine are translated into cursor commands by the Cursor Control Module, which handles movement, clicking, and other actions while also incorporating user feedback for adaptive behavior.

Throughout this entire process, the system collects feedback through various mechanisms, refining its algorithms and improving performance over time. Finally, the Integration Layer ensures seamless communication between components and provides APIs for integration with other software applications or assistive technologies. This comprehensive architecture allows the Virtual Cursor Control System to deliver intuitive, adaptive, and personalized cursor control that caters to the diverse needs and input modalities of users.

D. Literature Review

There are various methods available to control the cursor
round of evaluations, they used settings with through hand gestures, but the most common one is wearing a DataGlove. However, this approach reduces both the user's and the system's performance. The complexity of the system is a major issue in this method.

Dung-Hua Liou and Chen-Chiung Hsieh published a paper that implemented adaptive skin color models and a motion history image-based hand moving direction detection technique. The project achieved an average accuracy of 94.1% with a processing time of 3.81 milliseconds per frame. However, the paper struggles to recognize more complex hand gestures in a working environment.

Chang-Yi Kao and Chin-Shyurng Fahn focused on vision-based hand gesture identification for the HCI interface in their paper. The experimental findings showed a face tracking rate of over 97% under typical circumstances and over 94% when the face is partially obstructed. The system's execution efficiency is excellent, and it has inspired plans to market the robot soon. However, high configuration computers are necessary for accurate results.

Hand gestures have the potential to provide a more natural interface to computer vision systems, as they can convey thoughts and actions. However, this system faced limitations in complex environments and was only reliable under proper lighting conditions.

Ashwini M. Patil et al. published a paper on a machine-user interface that utilizes multimedia techniques and basic computer vision for hand gesture recognition. They encountered a significant limitation in the gesture comparison algorithms, as hand segmentation and skin pixel extraction from stored frames were required.

This project employed a camera to capture hand motions using color detection techniques. The primary component of this technique is the utilization of a web camera. They faced certain limitations, such as the requirement for a light operating system background and the absence of objects with vibrant colors. Computers with a specific high configuration perform well in this context.

Yimin Zhou et al. reported a study that presented a high-level hand feature extraction approach for real-time gesture detection. The system they developed demonstrates good accuracy in extracting both flexional and extensional fingers. However, this method can only be used on computers with high configurations.

Pooja Kumari et al. described an experiment where several color bands were utilized for various tasks. The number of colors served as the key to control mouse actions. However, the system relied on a multitude of colors instead of different gestures to performfunctions.

Aashni Haria et al. proposed a paper based on a background extraction and contours detection system. They conducted two sets of assessments to determine the accuracy of their method. In the first uniformly simple backdrops, while the second assessment involved backdrops with discrepancies. Each gesture was performed ten times in each setting, and the average number of successful identifications was calculated as the accuracy, resulting in 85% and 80%. However, the system operates at a significantly slow pace.

E. Data Flow Diagram

The data flow diagram (DFD) is a crucial modeling tool that illustrates the flow of information within a system. It represents the system components, including the system process, the data involved in the process, external entities interacting with the system, and the information flows within the system. DFD shows how information moves through the system and how it is altered by various processes.

External Entities: User: The individual engaging with the virtual mouse system. The user contributes input to the system, such as gestures and camera input. Computer System: This external entity encompasses the hardware and software components of the computer system that the virtual mouse interacts with. This includes the operating system, applications, and any peripherals connected to the computer.

Processes: Input Processing: This process is responsible for receiving input from the user and converting it into a format that the virtual mouse system can comprehend. Depending on the input method (e.g., camera, gestures), this process may involve various algorithms for recognizing and interpreting user actions.

Cursor Control: This process manages the logic for controlling the movement of the virtual mouse cursor based on the input received from the user. It calculates the new position of the cursor and updates its coordinates accordingly.

Click and Gesture Recognition: This process identifies user actions such as clicking, dragging, scrolling, and other gestures, and translates them into corresponding commands for the virtual mouse system to execute. Output Generation: Once the user input is processed and interpreted, this process generates the appropriate output signals to control the computer interface. This may involve simulating mouse clicks, movements, or other actions.

Data Flows: Input Data: This represents the raw input data received from the user, which could include video frames, audio recordings, or other sensor data depending on the input method. Processed Data: After the input data is processed by the system, this represents the interpreted and transformed data that is used to control the virtual mouse cursor and execute user commands. Output Data: This represents the signals and commands generated by the virtual mouse system to interact with the computer interface. This could include mouse movements, clicks, or other actions.

F. System Implementation

The suggested system utilizes Computer Vision technology to enable hand gestures to function as a virtual mouse. This innovative approach offers an alternative to traditional wired and wireless models.

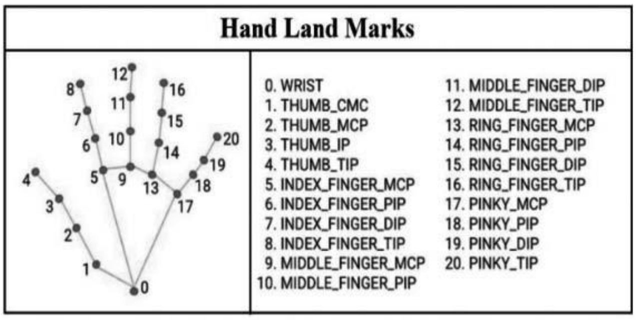

Fig. 4 Hand Gesture Marks

Gesture 1: Neutral Gesture- is indicated by having all five fingers up, resulting in no cursor movement or function being performed.

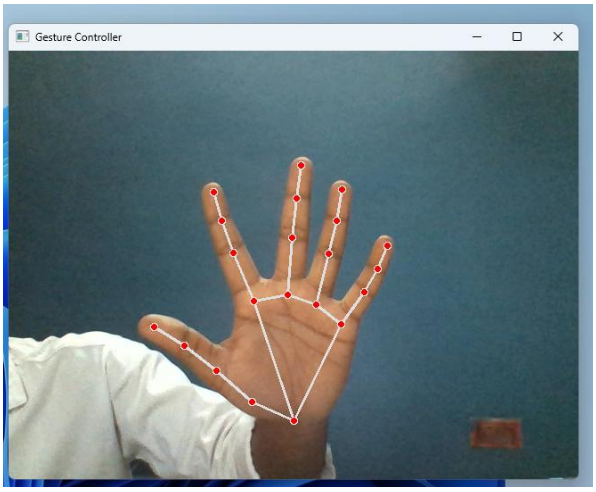

Fig. 1 Neutral Gesture

Gesture 2: To control cursor on computer screen

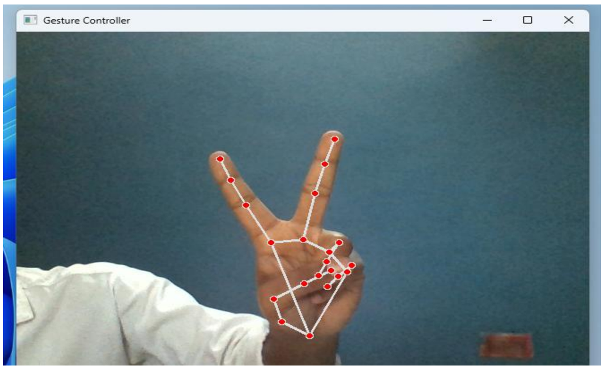

Fig. 3 Control the Cursor

(In this gesture 2 figure shows the control of cursor to move around the computer screen)

Gesture 3: To execute the right-click function (In a physical mouse, the right-click function is used to select or drag components, similarly in a virtual setting to perform a right-click action).

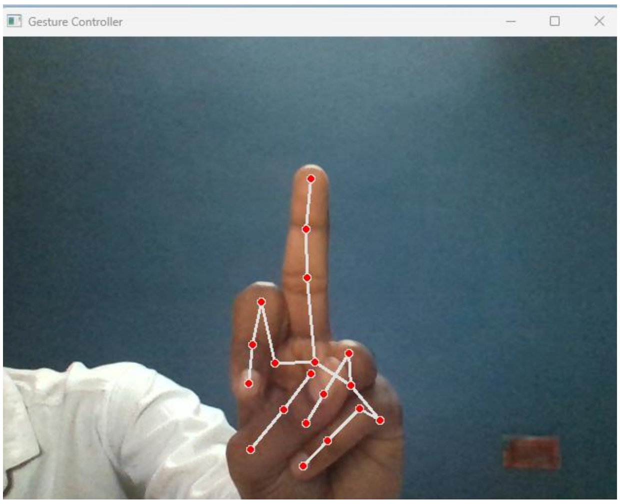

Fig. 3 Right Click

Conclusions

The primary goal of the AI virtual mouse system is to control the functions of the mouse cursor through hand gestures instead of relying on a physical mouse. This can be achieved by utilizing a webcam or built-in camera to detect and process hand gestures and hand tip movements in order to execute specific mouse functions.

Based on the outcomes of the model, it can be concluded that the proposed AI virtual mouse system has performed exceptionally well and exhibits higher accuracy compared to existing models. Furthermore, this model successfully addresses most of the limitations found in current systems. Given its superior accuracy, the AI virtual mouse can be effectively utilized in real-world applications. Additionally, it offers the advantage of reducing the transmission of COVID-19, as it allows for virtual mouse usage through hand gestures, eliminating the need for a traditional physical mouse.

Future scope

The AI virtual mouse under consideration presents certain constraints, including a slight reduction in the accuracy of the right-click function and challenges in performing click-and-drag actions to select text. These limitations are part of the current AI virtual mouse system, but will be addressed in our upcoming research endeavors.

Additionally, there is potential for enhancing the proposed method to incorporate virtual keyboard functionalities alongside mouse operations, opening up new possibilities for Human-Computer Interaction (HCI).

REFERENCES ;

1. international conference on Robotics and emerging allied technologies in engineering (icReate) islamabad, pakistan, april 22-24, 2014.
2. S. Shriram , B. Nagaraj ,J. Jaya , S. Shankar , and p. ajay, "deep learning-Based Real-time ai virtual Mouse System using computer vision to avoid covid-19 Spread", Journal of healthcare engineering volume 2021, article id 8133076, 8 pages https://doi.org/10.1155/2021/8133076 .
3. Bhushan, S.; alshehri, M.; keshta, i.; chakraverti, a.k.; Rajpurohit, J.; abugabah, a. "an experimental analysis of various Machine learning algorithms for hand gesture Recognition". electronics 2022, 11, 968. https:// doi.org/10.3390/ electronics11060968.
4. d.-S. tran, N.-h. ho, h.-J. yang, S.-h. kim, and g. S. lee, "Real-time virtual mouse system using RgB-d images and fingertip detection," Multimedia Tools and Applications, vol. 80, no. 7, pp. 10473–10490, 2021.
5. kavitha R , Janasruthi S u , lokitha S , tharani g, "haNd geStuRe coNtRolled viRtual MouSe uSiNg aRtifical iNtelligeNce", www.ijariie.com, vol-9 issue-2 2023 , iJaRiie-iSSN(o)-2395-4396, pp 307-320.
6. haria, a. Subramanian, N. asokkumar, S. poddar, and J. S. Nayak, "hand gesture recognition for human computer interaction," procedia computer Science, vol. 115, pp. 367– 374, 2017.
7. D.-H. Liou, D. Lee, and C.-C. Hsieh, "A real time hand gesture recognition system using motion history image," in Proceedings of the 2010 2nd International Conference on Signal Processing Systems, IEEE, Dalian, China, July 2010.
8. S. U. Dudhane, "Cursor control system using hand gesture recognition," IJARCCE, vol. 2, no. 5, 2013.