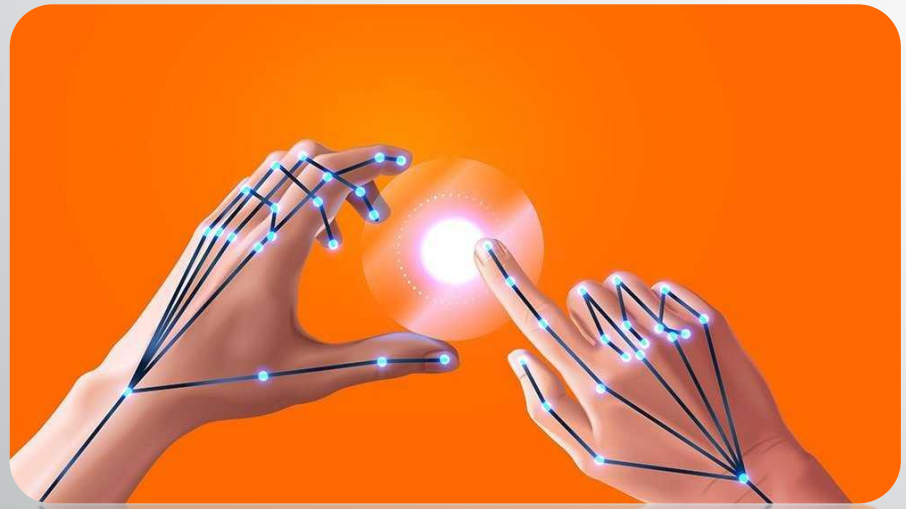
MACHINE LEARNING

VIRTUAL MOUSE

USING HAND AND EYE MOVEMENTS



UZUMAKI CLAN

99210041598-N.Lohith Reddy

99210041474-MD.Imran

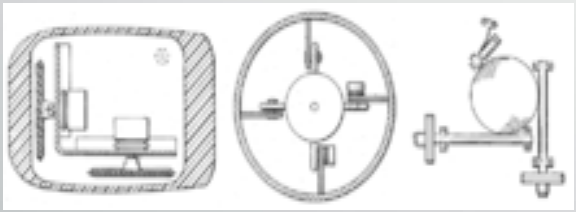
99210041796-N.Subash

9921004936-N.Vinay Kumar Reddy

99210041251-N.B.Adarsh

CONTENTS

History



History

- The trackball, a related pointing device, was invented in 1946 by Ralph Benjamin as part of a post-World War II-era fire-control radar plotting system called the Comprehensive Display System (CDS).
- Another early trackball was built by Kenyon Taylor, a British electrical engineer working in collaboration with Tom Cranston and Fred Longstaff. Taylor was part of the original Ferranti Canada, working on the Royal Canadian Navy's DATAR (Digital Automated Tracking and Resolving) system in 1952.
- Engelbart's first "mouse“ in 1963
- On 2 October 1968, three years after Engelbart's prototype but more than two months before his public demo, a mouse device named Rollkugelsteuerung (German for "rolling ball control") was shown in a sales brochure by the German company AEG-Telefunken as an optional input device for the SIG 100 vector graphics terminal, part of the system around their process computer TR 86 and the TR 440 main frame.
- First mice on personal computers and workstations -The Xerox Alto was one of the first computers designed for individual use in 1973 and is regarded as the first modern computer to use a mouse.

Abstract

- Human hands and eyes have a wealth of information that may be accessed and applied in a variety of ways In this study, a brand-new technique for using a live camera to control mouse movement is introduced.
- We illustrate how it can perform all activities that current mouse devices can by using a camera and computer vision technology, which is the basis behind our approach to managing mouse duties.
- In addition to focusing on a human computer interaction application based on eye and hand tracking, this research aims to propose a low-cost real-time system for tracking mouse eye and hand motions.
- Additionally, these motions may be recorded and utilized as control signals to let users interact with interfaces without the need of a mouse or keyboard.
- Algorithms for computer vision and image processing can be used to accomplish this. This model is used to operate the system using hand gestures or eye gestures in the graphical user interface.

Introduction

- The PC mouse is one of the wondrous developments of people in the field of Human-Computer Interaction (HCI) innovation.
- The requirement for simple and effective input methods grows as computer devices importance in our everyday lives continues to grow.
- Input technologies like touchscreens and voice recognition have become increasingly sophisticated and practical in recent years, replacing the conventional mouse and keyboard arrangement, which has been the norm for decades.
- But even these cutting-edge input techniques have their limitations, especially when it comes to circumstances in which making physical touch with a device is either impractical or undesirable.
- Here is where the idea of a virtual mouse that can be controlled with hand and eye movements is useful. Users can interact with computing devices in a more natural way by combining hand and eye motions.

Introduction

- There's always room for advancement in the realm of computers, thanks to new technology in our digital era.
- Handsfree computing is popular nowadays because it caters to the needs of quadriplegics (people suffering from paralysis of all four limbs).
- We want to demonstrate a Human-Computer Interaction (HCI) system that is essential for amputees and others who have trouble using their hands.
- The system is a mouse-like eye-based interface that converts eye movements like blinking, staring, and squinting into mouse cursor actions.

Literature Survey

Sr.No	Name of Research Paper	Author Name	Technology Used	Advantages / Disadvantages
1	Virtual Mouse Control Using Colored Fingertips and Hand Gesture Recognition 11-12 Sep 2020	Vantukal Reddy , Thumma Dhyanchand , Galla Vamsi Krishna,Satish Maheshwaram	Hand Gesture Recognition, Image processing,Neural Network Algorithm,Python, opencv,pyautogui	**Advantage** : used neural network for hand gesture recognition. **Disadvantage** : Limitation of This Mouse they used Colored Fingertips
2	Virtual Mouse Implementation Using Opencv , 2019	Kollipara Sai Varun. I Puneeth, Dr.T.Prem Jacob	Opencv,IP,Deep Learning, Pyautogui,numpy,Anaconda	**Advantage** : used python for hand gesture recognition. **Disadvantage** : In this they used color variation techniques ,it's very complex to use
3	Hand Gesture - Virtual Mouse For Human Computer Interaction , 2018	Sherin Mohammed, V H preetha	Matlab s/w ,Two cameras,	**Advantage** :Get 90% Correct Detection for both views, under enough Light Condition **Disadvantage** : used two cameras and its cost is more

Literature Survey

4	Virtual Mouse, 12th march 2015	Ashish Mhetar.B K Srioop,Kavya AGS,Ramanath Nayak,Ravikumar Javali,Suma K V	IR Camera, USB-HID,IR Pen , Teensy(ARM M4)	**Advantage :** its work as a virtual marker , its effective to use while teaching . **Disadvantage :** They used a Hardware devices, and the Cost of this is More,.
5	Virtual Mouse Using Object Tracking 10th july 2020	Monali Shetty,Christina Daniel,Manthan Bhatkar,Ofrin Lopes	HSV Technique,Python, Opencv,Ip	**Advantage :**Accuracy of this System is in Plain Background :95% **Disadvantage :** Non-Plain Background :40%
6	Design and Development of Hand Gesture based Virtual Mouse 19 Dec 2019	Kabid Shibly,Samrat Dey,Aminul islam, Shahriar Showrav	HCI Technology	**Advantage :** use of latest technology. **Disadvantage :** In this System barrier is the lightning Condition That's Why System Still Cant enough to replace the

Problem Statement

“To design a virtual mouse which detects hand gestures and performs operations only using simple Hand and Eye gestures.”

Proposed System

“In proposed system users don’t have to color their fingers with a specific color and are not required to use any sensors.”

Methodology

We use two different algorithms to perform the mouse actions and we combine both the algorithms using good graphical user interface, so that user can choose the way how he can control the mouse.

Algorithm

- Import the necessary libraries: cv2, mediapipe, pyautogui.
- Open the camera and initialize the hand detector.
- Define a variable to store the index finger position and set it to (0,0).
- Start an infinite loop to capture frames from the camera.
- Read the frames from the camera and flip them horizontally using cv2.flip().
- Convert the color space of the frame from BGR to RGB using cv2.cvtColor().
- Use the hand detector to detect hands in the frame.
- Get the landmarks of the detected hands and draw them on the frame using cv2.circle() and mp.solutions.drawing_utils.draw_landmarks().
- Loop over the landmarks and check if the current landmark is the index finger or the thumb.
- If the landmark is the index finger, draw a yellow circle around it using cv2.circle() and calculate the index finger coordinates relative to the screen dimensions using the following formula: index_x = screen_width/frame_width * x, index_y = screen_height/frame_height * y, where x and y are the pixel coordinates of the index finger landmark.
- If the landmark is the thumb, draw a yellow circle around it using cv2.circle(), calculate the thumb coordinates relative to the screen dimensions using the same formula as in step 10, and perform the following actions based on the distance between the thumb and the index finger:
- If the distance is less than 50 pixels, simulate a left click using pyautogui.click().
- If the distance is less than 200 pixels, move the cursor to the index finger position using pyautogui.moveTo().
- If the distance is less than 30 pixels, simulate a right click using pyautogui.click(button='right').
- If the distance is less than 15 pixels, simulate a left click using pyautogui.click(button='left') and scroll up using pyautogui.scroll(10).

Display the frame on the screen using cv2.imshow() and wait for a key event using cv2.waitKey().

Algorithm

- Import the necessary libraries: cv2, mediapipe, pyautogui.
- Open the camera and initialize the hand detector.
- Define a variable to store the index finger position and set it to (0,0).
- Start an infinite loop to capture frames from the camera.
- Read the frames from the camera and flip them horizontally using cv2.flip().
- Convert the color space of the frame from BGR to RGB using cv2.cvtColor().
- Use the hand detector to detect hands in the frame.
- Get the landmarks of the detected hands and draw them on the frame using cv2.circle() and mp.solutions.drawing_utils.draw_landmarks().
- Loop over the landmarks and check if the current landmark is the index finger or the thumb.
- If the landmark is the index finger, draw a yellow circle around it using cv2.circle() and calculate the index finger coordinates relative to the screen dimensions using the following formula: index_x = screen_width/frame_width * x, index_y = screen_height/frame_height * y, where x and y are the pixel coordinates of the index finger landmark.
- If the landmark is the thumb, draw a yellow circle around it using cv2.circle(), calculate the thumb coordinates relative to the screen dimensions using the same formula as in step 10, and perform the following actions based on the distance between the thumb and the index finger:
- If the distance is less than 50 pixels, simulate a left click using pyautogui.click().
- If the distance is less than 200 pixels, move the cursor to the index finger position using pyautogui.moveTo().
- If the distance is less than 30 pixels, simulate a right click using pyautogui.click(button='right').
- If the distance is less than 15 pixels, simulate a left click using pyautogui.click(button='left') and scroll up using pyautogui.scroll(10).

Display the frame on the screen using cv2.imshow() and wait for a key event using cv2.waitKey().

Future Scope

- The future scope for virtual mouse using hand and eye gestures looks promising. With the advancement in technology, the use of virtual mouse using hand and eye gestures has become more popular and accessible.
- One of the major advantages of using virtual mouse technology is that it can help to reduce the physical strain on users. It eliminates the need for a physical mouse, which can be particularly beneficial for those who spend long hours working on a computer. Additionally, virtual mouse technology can be useful for people with disabilities or limited mobility who may find it difficult to use a traditional mouse.
- As technology continues to advance, we can expect to see more sophisticated virtual mouse systems that can interpret a wider range of hand and eye gestures with greater accuracy. This could potentially lead to a more intuitive and efficient way of interacting with computers and other devices.
- Furthermore, the integration of virtual mouse technology with other technologies such as augmented reality (AR) and virtual reality (VR) could open up new possibilities for immersive computing experiences. For example, users may be able to navigate virtual environments using hand and eye gestures rather than traditional controls, providing a more natural and intuitive user interface.
- Overall, the future scope for virtual mouse technology using hand and eye gestures is likely to be significant, and we can expect to see further developments in this area in the coming years.

Conclusion

- In conclusion, the virtual mouse using hand and eye gestures is an innovative technology that has the potential to revolutionize the way we interact with computers and other devices.
- The technology has already made significant progress in recent years, and we can expect to see further developments in the future, with more sophisticated systems that can interpret a wider range of hand and eye gestures with greater accuracy.
- This technology has many potential applications, including helping people with disabilities or limited mobility to interact with computers more easily, reducing physical strain on users, and providing a more natural and intuitive way of navigating virtual environments.
- The integration of virtual mouse technology with other technologies such as AR and VR could also lead to new and exciting possibilities for immersive computing experiences.

- Overall, the virtual mouse using hand and eye gestures is an exciting and promising technology that has the potential to improve the way we interact with technology. As such, it is an area that is worth keeping an eye on for future developments.

Thank you!