VIRTUAL MOUSE USING OPENCV

BALA HARSHITHAA B[1], LEELAVATHI R[2]

[1]*Student, Department of Information Technology, Bannari Amman Institute of technology, Erode, Tamil Nadu, India*

[2]*Assistant Professor, Department of Information Technology, Bannari Amman Institute of technology, Erode, Tamil Nadu, India*

---***---

Abstract - *Human-computer interactions highly rely on hand gesture recognition. As we can see, there are a lot of new technological developments taking place. For example, biometric authentication, which is widely used in smartphones, and hand gesture recognition, which is a contemporary method of Human Computer Interaction. As an alternative to the current methods for controlling cursor motions, which include physically pressing buttons or modifying the angles of an actual computer mouse, this research proposes a method for the use of a real-time camera in Human Computer Interaction. It can perform all the activities that a standard computer mouse can, and it also controls many mouse events with the help of a camera and computer vision technology to detect finger movement gestures on the computer's window and allows users to control the entire system with just their fingers. This system includes a User-Interface through which the user can enable the tracking of the hand movements and generates coloured coordinates to detect the hand. The real-time photos that the virtual mouse colour recognition application is continuously gathering will undergo a variety of filters and conversions. The application will employ a photo processing strategy once the system finishes to extract the focused coloured coordinates from the frames that are transformed. It is made more approachable by using finger detecting techniques for rapid camera access. This technology decreases the need for a real mouse, saving time and effort.*

***Key Words*: Open CV, Hand-gestures, Virtual mouse, Image processing.**

1. INTRODUCTION

A mouse is referred to as a pointing device in computer nomenclature, using two-dimensional movements in reference to a floor. This movement is converted into pointer motions on a display, making it possible to control the GUI of a computer platform. In the modern era, there have been several distinct types of mouse, like mechanical mouse, which moves by using a sturdy rubber ball that rotates when the mouse is moved. After that, an LED sensor was developed to replace the rubber ball in the mouse with a hard exterior that could detect movement on the table and transfer data to a computer for processing. The laser mouse was subsequently launched in 2004 to solve the shortcomings of the optical mouse, which include challenges in tracking highly reflective surfaces and to improve motion accuracy with the tiniest hand motion. Despite its accuracy, the mouse nevertheless has limitations in terms of both its physical capabilities and its technical capabilities.

For instance, a laptop mouse is a consumable piece of hardware since it eventually needs to be replaced either because the mouse buttons have worn out and now make worthless clicks or because the laptop no longer recognises the mouse as a whole. Interactions between people and computers are crucial. Despite the limitations of laptop technology, it is expanding.

Given the development of a touch-screen mobile device, the market is starting to demand the adoption of the same technology across all platforms, including desktop machines. Despite the fact that desktop PCs with contact displays already exist, the price may be too high. An alternative to touch screens would be a digital human-computer interface that uses a webcam or other photo-taking tools instead of a physical mouse or keyboard. The movements a person makes to engage with a webcam can be translated into pointer motion by a software programme that frequently uses the webcam, much like a real mouse.

In this system that is developed, the camera takes pictures, stores them in a temporary location, analyses the frames, and then detects the pre-programmed hand gesture actions.

This type of virtual mouse system is typically developed using the Python programming language, and computer vision projects also employ the OpenCV package. The system makes use of well-known Python libraries like MediaPipe for the real-time tracking of hands and finger movements. The PyAutoGUI package is also used to track the finger as it moves across the screen to perform actions like left- and right-clicking, dragging, and scrolling. This also makes use of Numpy for calculations, Pycaw for audio related functionalities like removing background noises, TKinter for UI, Screen Control for controlling the screen brightness.

2. LITERATURE SURVEY

The following literature helped in developing the proposed work with greater performance and implementing it in a way by overcoming the disadvantages of the existing systems.

[1] By using an IR camera and an IR pen that serves as a virtual marker, Ashish Mehtar and Ramanath Nayak created the **Virtual Mouse** on March 12, 2015. It also makes use of hardware. They suggested a product that strives to be a virtual marker and has the added benefit of incorporating capabilities from a mouse. The existing Virtual Marker is altered to serve as both a mouse pointer and a marker, giving it access to all of the mouse's features. In their article, they suggested a hardware implementation of a virtual mouse that, by making the existing "Virtual Marker" very responsive in real time, improves performance.

[2] By Sherin Mohammed and VH Preetha in 2018, their **Hand Gesture-Virtual Mouse for Human Computer Interaction** utilised OpenCV and Python and required the deployment of two expensive cameras. Their study focuses on the development of hand gestures in 3-D space for human computer interface systems using two cameras in position. The screen coordinate system is approximated and mapped to the hand gesture of pointing. To fulfil the function of the virtual mouse, we also employ various hand movements. They employ other gestures, such as selecting a folder or an object, in addition to hand movements to point at the screen.

[3] Kollipara Sai Varun, I Puneeth, and Dr. T.Prem Jacob's **Virtual Mouse Implementation Using Opencv** in 2019 uses a webcam or built-in camera to record hand motions and recognise hand tips using computer vision. The machine learning algorithm is utilised by the system's algorithm. Their suggested solution is built on cutting-edge technology that makes use of hand motions. With the use of a camera, hand motions are recorded. After that, hand landmark key points are detected, and depending on the gesture recognised, the mouse cursor executes a variety of actions. Users can operate the mouse in a number of different ways without the need for any hardware or sensors; all you need is your hand and a pair of fingers. Also, it is cost-efficient and user-friendly.

[4] Kabid Shibly, Samrat Dey, Aminul Islam, and Shahriar Showrav used HSI Technology in the **Design And Development Of A Hand Gesture-Based Virtual Mouse** on December 19, 2019. The lightning condition was a barrier. Their suggested mouse system exceeds this restriction. In their study, they suggest integrating computer vision and hand movements to create a virtual mouse system based on HCI. Webcam or built-in camera recordings of gestures that have been subjected to colour segmentation and detection processing. With hands that have coloured caps on the tips, the user will be able to control some computer cursor operations. A user can primarily scroll up or down by using various hand motions, as well as conduct left, right, and double clicks. The system uses a webcam or a built-in camera to take frames, then processes the frames to make them trackable. After that, it identifies various user motions and executes the mouse function.

[5] By using computer vision to operate the mouse cursor, the **Virtual Mouse Using Object Tracking** research by Monali Shetty, Christina Daniel, Manthan Bhatkar, and Ofrin Lopez in 2020 captures hand gestures from a camera using an HSV colour recognition method. Their goal is to develop a system-interacting object tracking application. Their approach, which uses hand gestures recorded from a webcam using an HSV colour detection method, is a computer vision-based mouse cursor control system. Using coloured caps or tapes that the computer's webcam tracks, their technology enables users to move the system pointer and carry out mouse actions like left-, right-, and double-clicks with various hand motions. The system is implemented using real-time computer vision in Python and the OpenCV package. The output is shown on the monitor from the camera.

[6] Vantukal Reddy, Thumma Dhyanchand, Galla Vamsi Krishna, and Satish Maheshwaram's **Virtual Mouse Control Using Colored Fingertips And Hand Gesture Recognition**, published in 2020, uses two techniques for tracking the fingers: One study in the field of human-computer interaction uses a virtual mouse with fingertip detection and hand motion tracking based on image in a live video. They suggest employing hand motion recognition and fingertip identification to control a virtual mouse in their article. Two techniques are used in their investigation to track the fingers: hand gesture detection and the use of coloured caps. It consists of three basic steps: tracking hand gestures, finger detection via colour identification, and cursor implementation. In their investigation, a convex hull is formed around the contour that is detected to produce hand gesture tracking. With the area ratio of the produced hull and contour, hand features are retrieved.

[7] Hand recognition and segmentation with a Microsoft Kinect Sensor version 2 depth image is made possible by Dinh-SonTran, HyungJeong Yang, Ngoc-Huynh Ho Soo HyungKim, and Guee Sang Lee's **Real-Time Virtual Mouse System Using Rgb-D Pictures And Fingertip Detection**. By utilising fingertip sensing and RGB-D pictures, they suggest a revolutionary virtual mouse technique. Using detailed skeleton-joint information photos from a Microsoft Kinect Sensor version 2, the hand region of interest and the palm's centre are first retrieved, and they are then translated into a binary image. A border-tracing method is then used to extract and describe the hands' outlines. Based on the coordinates of the hand contour, the K-cosine method is used to determine the location of the fingertip. Lastly, the mouse cursor is controlled via a virtual screen by mapping the

fingertip location to RGB pictures. Using a single CPU and Kinect V2, the device detects fingertips in real-time at 30 frames per second on a desktop computer.

[8] Zhenzhou Wang's **Robust Segmentation Of The Image By Fusing The SDD Clustering Result From A Separate Colour Space** employs SDD clustering as a revolutionary colour segmentation technique that performs better than traditional clustering techniques. By combining the slope difference distribution (SDD) clustering outcomes in several colour spaces, it is suggested that the colour image be segmented. The segmentation method is intended to be two-label segmentation for simplicity, while multiple-label segmentation might be readily generalised from this. The proposed method is contrasted numerically and qualitatively with the most recent colour picture segmentation techniques. Experimental findings confirmed the efficacy of the suggested strategy.

3. PROPOSED SYSTEM

People aspire to live in a world where technology equipment may be controlled and interacted with remotely without the need for any peripheral devices like remote controls, keyboards, or the like. Without a doubt, a virtual mouse will soon take the place of a physical one. It not only provides ease, but it also saves money. The existing approach has the drawback of only accurately detecting colours on plain backgrounds due to its usage of colour detection technology. Additionally, the right click mouse function's accuracy has slightly decreased, and the model has some trouble clicking and dragging to pick text. In terms of daily use, the interactions are also not very user-friendly. While recognising the gesture movements, ambiguity is created when the three various interaction techniques are combined. A physical mouse requires specialised hardware and a surface to operate. Physical people struggle to adjust to unique circumstances, and depending on the environment, they function differently. The mouse only has a small set of capabilities, even in typical operational scenarios. Every wireless mouse has a different lifespan.

The calibration stage and the reputation segment are the two main levels that make up the shade reputation process. The calibration segment's objective is to instruct the system on how to interpret the hue and saturation values for colours that customers have selected. These values and other configurations may be stored by the device, if necessary, in text files that will be used in a later segment of the identification process. The device will start to take frames and look for data for colour as the recognition portion of the process starts at some point in the calibration part, depending on the parameters that were recorded.

4. APPLICATION DESIGN

Upon startup, the software will display a console window; users must make a selection from the primary menu selections because each option affects the program's capabilities differently. The second option enables them to give the customer the ability to pick and change the colours they desire in order to achieve greater accuracy and efficacy at a later stage in the popularity phase. Additionally, the 0.33 option allows the user to modify the program's settings, including the cameras that can be chosen as well as the size of the comment window.

When the default option is chosen, the application will launch several approaches, establish the appropriate variables, and display a number of windows, including one that displays the real time captured frame and others that display the HSV track bars and binary thresholds of individual shades. To increase the popularity of colour accuracy, users are invited to alter the track bars for the HSV that are provided. However, users must have a fundamental understanding of the HSV shade version in order to edit the tune bars effectively; otherwise, incorrect adjustment could cause the entire reputation area to malfunction.

The UI and AI are combined in this application. The user opens the.exe file that contains the application first. When the file is opened, a user interface with the option to TRACK appears. When the user clicks that button, our backend application is run, and the logic inside allows access to the camera and finger tracking actions.

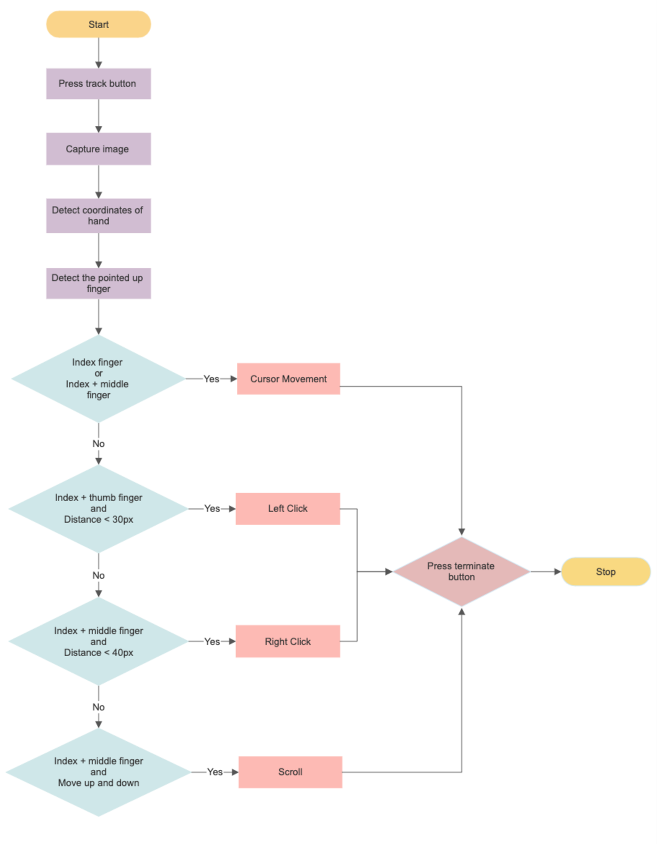

Fig 1: Block Diagram of the Proposed System

5. MODULES USED

5.1. OpenCV

It is a software library for machine learning and computer vision. A standard infrastructure for computer vision applications was created with OpenCV in order to speed up the incorporation of artificial intelligence into products. Since OpenCV is an Apache 2 licensed product, it is simple for businesses to use and alter the code. It also offers Gpu acceleration features for real-time operation. OpenCV is utilised in many different fields, including as mobile robots, facial and gesture recognition systems, and 2D and 3D feature toolkits.

5.2. MediaPipe

A cross-platform framework developed by Google that is mostly employed for the construction of multimodal pipelines in machine learning. Since the MediaPipe framework is built using statistical data, it is most helpful for the cross-platform design process. The MediaPipe framework is multimodal, which means it can be used for motion tracking, object detection, face detection, facemesh, iris scanner position detection, hand detection, hair segmentation, and objection detection. The MediaPipe framework is the finest choice for developers when creating, analysing, and designing systems performance in the form of graphs. It has also been utilised for constructing a variety of cross-platform applications and systems. Here it is used for streaming the video and gesture recognition.

5.3. Numpy

NumPy is an all-purpose toolkit for dealing with arrays. It provides a multidimensional array object with exceptional performance and the ability to interact with these arrays. For scientific computing, it is the essential Python package. The programme is open-source. It has a number of characteristics, including the following crucial ones: Tools for merging C/C++ and Fortran code, a N-dimensional array object, and sophisticated (broadcasting) functions useful Fourier transform, random number, and linear algebra abilities.

5.4. Pycaw

Python Core Audio Windows Library for audio related functionalities like removing background noises.

5.5. TKinter

The typical Python interface for the Tk GUI toolkit is the Tkinter package. The majority of Unix systems, including macOS, as well as Windows machines, support Tkinter. The combination of Python and Tkinter makes it quick and simple to develop GUI apps. An effective object-oriented interface for the Tk GUI toolkit is provided by Tkinter.

5.6. PyAutoGUI

The automation library PyAutogui for Python supports keyboard and mouse control. Alternatively, we could say that it makes it easier for us to automate keyboard and mouse clicks in order to establish interaction with another programme using a Python script. A handful of the features it offers are, In the windows of the other applications, we can move the mouse and click, We can communicate keystrokes with other programmes. As an illustration, filling out a form, entering a search term into a browser, etc. Another option is to take pictures and provide an image. It enables us to find an application window and move, resize, maximise, or reduce it. Display message and alarm boxes.

6. SAMPLE OUTPUTS

Fig 2: Left Click

Fig 3: Right Click

6. RESULT ANALYSIS

The proposed system was successfully implemented and the following results were achieved.

A basic and responsive mouse operating system. Incredibly delicate interactions were performed. The interaction accuracy was scaled-up by decreasing the present sensitivity level. The mouse movements were handled even more precisely in dark spaces. A system which uses the concept of history of images for transforming a real time collection of frames of images and processing them to greatly improve the accuracy of cursor movements and fix issues with malfunctioning of the system due to low video quality and lighting problems. Use of a single camera for hand gesture detection. Elimination of the use of coloured fingertips. It can adapt to different backgrounds.

The accuracy that was achieved for cursor movements was 97.4% and that for the Left click recognition was 96.2%. For the Right click recognition, it was 94.3% and for Scrolling up and down, it was analysed to be 94.7%.

It also proved to be a cost effective and most efficient, user-friendly system that can be implemented very easily.

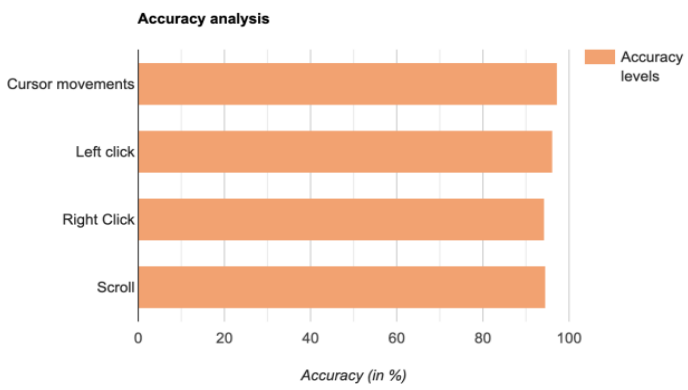

Fig 4: Accuracy analysis of the implemented system

7. CONCLUSION

This model can come to a conclusion by utilising computer vision techniques like open CV, colour variation approaches, and the generation of mouse movement through the usage of specific packages. This will be utilised for mouse movement via coordinates. This can make systems and numerous other applications easier to use. As a result, the open CV offers its customers a variety of models that are easily accessible and will simplify their lives.

As an extension to this work, it would be optimal to study various hardware approaches that would produce more precise hand detections and conduct research into advanced computational materials for image processing. This study not only illustrated the various gesture activities that users might perform, but it also showed the potential for streamlining user interactions with hardware devices and personal computers.

REFERENCES

[1] "Fingertip-writing alphanumeric Character Recognition for Vision-based Human Computer Interaction" by Cheng-Yuan Shih, Chien-Cheng Lee, and Bor-ShennJeng, in International Conference on Broadband, Wireless Computing, Communication and Applications 978-0-7695- 4,2010

[2] Hemanta Saik, Abhik Banerjee, Koustuvmoni Bharadwaj, Abhirup Ghosh, "Mouse Control using a Web Camera based on Colour Detection"in International Journal of Computer Trends and Technology (IJCTT) – volume 9.

[3] Ronak Joshi, Rashmi Adatkar, et.al, (2017). "Virtual Mouse" in Imperial Journal of Interdisciplinary Research (IJIR), Vol-3, Issue-4.

[4] Pritpal Singh, Tanjot Sethi, B.B.V.L. Deepak and Meta Dev Prasad Murthy (2015). "Real-Time Object Detection and Tracking Using Color Feature and Motion" in the International Conference on Communication and Signal Processing.

[5] "Virtual Mouse Implementation using Color Pointer Detection" by D.S. Suresh and I.V. Bhavana.

[6] "A virtual mouse interface with a two layered Bayesian network" by S.W. Lee, D. Kang, S. Huh, M.C. Roh.

[7] D. Lee, D.-H. Liou, and C.-C. Hsieh, "A real time hand gesture recognition system using motion history image," in Proceedings of the 2nd International Conference on Signal Processing Systems, July 2010.

[8] "Cursor system using hand gesture recognition," in IJARCCE by S. U. Dudhane

[9] Izaman Khan and Adnan Ibraheem Rafiqu, "Survey on Various Gesture Recognition Technologies and Techniques" in International Journal of Computer Applications Volume50 – No.7, July 2012

[10] "Contested Collective Intelligence: Rationale, Technologies, and a Human-Machine Annotation" by Ágnes Sándor, Anna De Liddo, et.al, (2012). in Computer Supported Cooperative Work Volume 21, Issue 4–5, pp 417–448.