

Hand Gesture Recognition To Control PC

Bhupathi Shwejan Raj¹, Bolisetty Sujith², Janagama Vamshi Krishna³, Raja Pavan Karthik⁴, Suravarapu Ankith⁵

^{1,2,3,4,5} Department of Computer Science Engineering
Amrita Vishwa Vidyapeetham

amenu4aie20017@am.students.amrita.edu; amenu4aie20018@am.students.amrita.edu;
amenu4aie20036@am.students.amrita.edu; amenu4aie20060@am.students.amrita.edu
;amenu4aie20070@am.students.amrita.edu

Abstract

Since the development of computer technology, the technique for constructing a process of human-computer interaction has evolved. In HCI (Human-Computer Interaction) technology, the mouse is a fantastic invention. Even if wireless or Bluetooth mouse technology has yet to be established, it is not device-free. A Bluetooth mouse necessitates the usage of a battery and a connected dongle. The presence of additional electronics in a mouse makes it more difficult to use. This limitation is not met by the proposed mouse technology. Using computer vision and hand motions, this research presents a virtual mouse system. Specifically, we use the openCV to recognize gestures and make it attainable to identify relatively complex gestures using only one cheap monocular camera. This system takes frames using a webcam or built-in camera, analyses them to make them track-able, and then identifies various user motions and performs the mouse function. Sing their hand in different gestures. In this paper we are going to primarily do basic tasks like moving the mouse around, left click, volume operations, brightness operation using hand in different gestures using python programming language.

Keywords - Gesture recognition, openCV, human-computer interaction, virtual mouse system, python.

Introduction

In day-to-day life, hand gesture recognition is one of the systems that can detect the gesture of hand in a real time video. The gesture of hand is classified within a certain area of interest. Designing a system of Hand gesture recognition is one of the goals of achieving the objectives of this project. The task of recognizing hand gestures is one of the main and important issues in computer vision. With the latest advances in information and media technology, human computer interaction (HCI) systems that involve hand processing tasks such as hand detection and hand gesture recognition. The first step in any hand processing system is to detect and locate the hand in the real-time video from the webcam. The detection of the hand is challenging because of variation in pose, orientation, location and scale. Also, different intensity of light in the room adds to the variability. In the process of detection of the hand , hand gesture recognition generally involves multiple levels such as image acquisition, pre-processing, feature extraction and

gesture recognition. Image acquisition involves capturing images in the video frame by frame using a webcam. The captured images go through the image preprocessing process which involves colour filtering, smoothing and threshold. Feature extraction is a method that involves extracting features of the hand image such as hand contours, while gesture recognition is a method to recognize hand gestures by extracting the features.

In this study, designing hand gesture recognition is one of the complicated jobs that involves two major problems. Firstly is the detection of the hand. User hand is detected by using a webcam in real-time video. The problem would be the unstable brightness, noise, poor resolution and contrast. The detected hands in the video are recognized to identify the gestures. At this stage, the process involves are the segmentation and edge detection. With various information of image like colour, hand posture and shape based (shape of hand) in a real time would affect the recognition of gestures. Another problem is to create a sign that is suitable to be used one hand at a time. The extraction of the hand needs to be followed to determine each number and sign used. The extraction of the hand involves contour and convexity defects.

This study comprises on how to implement a complete system that can detect, recognize and interpret the hand by using Python and OpenCV in any intensity of light, pose or orientation of hand. In order to accomplish this, a real-time gesture based system is developed.

Literature review

In a [1] research paper they discussed how they implemented mouse navigation, Selection of icons and its functions and tasks like left, right, double click and scrolling, and they built a system which was based on image comparison and motion detection technology, and they did mouse indicator movements and selection of the icon. And they did an analysis on the results. They got accuracy when they measured using a plain background and with complex background for all four mouse events: Left click, Double Click, Right click and scrolling up and Down. And accuracy for plain background was highest for Right click(95%) and was lowest for scrolling up and down(78%) and the accuracy for complex background was highest for Right click (79%) and lowest was for Left click and scrolling up and down(40%). By the accuracy they have been concluded that the system was not efficient in complex and rough background but was faster for plain background than others, it was achieved by using a high definition camera.

In this article [2] they discussed mouse operation, media player controlling and static gesture based computer operation they performed successfully in MATLAB. And In mouse operation they have to perform each mouse click event separately that is left click, right click and scrolling. In static gesture based computer operating, they have used black background to increase accuracy. They have tested static gestures without black background and got correct recognition, but the condition is that the background of the database image and the background of the input image during testing time should be the same. Accuracy was taken as measure for result,

the number of input images were 20 for all three gestures, the accuracy for the first and the second gesture was 95% i.e. the images recognized were 19 out of 20 and the accuracy for the third gesture was 100%. It recognized all 20 images correctly.

The [3] paper discusses the manoeuvre of the mouse pointer and performs various mouse operations such as left click, right click, double click, drag etc. using gestures recognition technique. They have developed a system which recognizes the movement of fingers, they have used colour caps to distinguish it from the background colour. They have calculated the recognition rate of operations and the results show that the recognition rate for right click was best when compared to other operations which was 92% whereas the recognition rates for left, Double, and Drag operations were 87%, 82% and 78%. Not only that, but they have also calculated the Response rate of the cursor movement when distance between user and camera is varied from 0.3 m till 2.4 m. The smoothness in cursor movement decreases when the distance from the webcam is increased.

Methods

Here first we import the required libraries like openCV, numpy, hand tracking module, autopy

OpenCV : OpenCV is a cross-platform library using which we can develop real-time computer vision applications. It mainly focuses on image processing, video capture and analysis including features like face detection and object detection.

Mediapipe : MediaPipe is a Framework for building machine learning pipelines for

processing time-series data like video, audio.

NumPy : NumPy is a Python library used for working with arrays. It also has functions for working in the domain of linear algebra, Fourier transform, and matrices.

Autopy: AutoPy is a simple, cross -platform GUI automation library for Python. It includes functions for controlling the keyboard and mouse, finding colours and bitmaps on-screen, and displaying alerts.

Hand tracking module : This module in which we have written about some functions like hand Detector, find hands, find Position, fingers Up, fingers Distance.

handDetector : Hand detection is the process of finding the joints on the fingers as well as the finger-tips in a given image or the image which is visible in the camera.

findhands : It returns an array of size 5 if it detects a hand in the image or in the camera

findPosition : It returns the position of the finger which is up . It is very useful for doing operations like mouse click, increasing brightness, and volume.

finguresUp : This returns the array of size 5 for 5 fingers. If finger is up it returns 1 and if finger is closed it returns 0

finduresDistance : It finds the distance between the fingers which is helpful while doing the mouse click operation.(In the Mouse click operation if there is no distance between then fingers i.e index and middle finger it performs click operation)

There are basically 8 steps in achieving the goal

Step 1: Find hand Landmarks



Fig - Hand Landmark

We use findhands method provided in handtracking module to find the hand, and findposition method will give the landmark list. Both methods take the frame of the video as input.

Step 2: Get the tip of the fingers.

From the above landmark list, we get the coordinates of the fingertip.landmark list contains the landmark IDs with its x and y coordinate

Step 3: Check which fingers are up.

The method named fingerup in the handtracking module gives a list of length 5 which corresponds to our 5 fingers, if the value of a particular index is 1.

Then The finger is up, if the value is 0 then the finger is down.

Step 4: Checking which condition it satisfies.

- If it detects only the index finger then the cursor activates, we can move it as a mouse.
- If both index and middle fingers are detected, the cursor stops moving, and it is ready for clicking.
- If all fingers are up then the brightness increases.
- If all fingers are closed then the brightness decreases.

- If only the thumb and index are opened then it is ready to control volume.

Step 5: Convert coordinates(640, 480) to (1920, 1080)

- we convert the X coordinate of finger tip from (0,640) to (0,1920)
- similarly we convert Y coordinate of finger tip from (0,480) to (0,1080)
- This is done in order to cover the mouse cursor on the entire screen.

Step 6: Taking actions

- If the action that has to take is moving the cursor, we can use the mouse.move method provided by autopsy library.
- This method takes final cursor coordinates and moves the cursor accordingly.
- If the action is to increase or decrease brightness we will take help of the brightness control library to do the same. And we do the same with volume.

Step 7: when index and middle finger are up :clicking mode

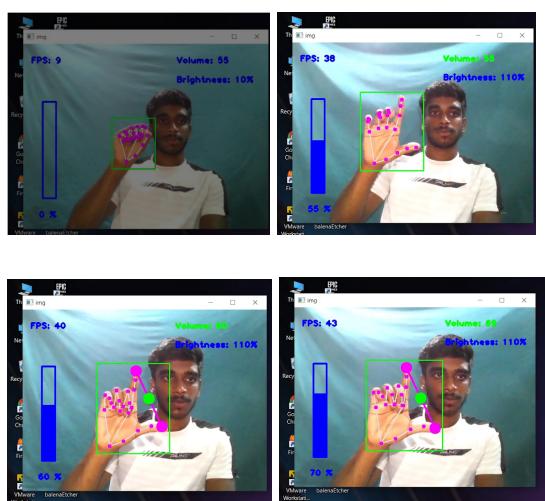
- Now we will find the distance between the tip of the index and middle fingers.
- If the value is less than some threshold value, we consider it as a click.
- We can achieve the mouse click action using the autopsy.mouse.click() method.

Step8: Display

- At last we display the frame on window with brightness index, volume index, cursor points

Results

We have implemented 4 operations successfully. The four operations include moving the mouse cursor that can be done by raising the index finger and moving in the frame. Left click it can be done by raising the index and middle finger and bringing the two fingers close we can do left click operation. Volume operation we have implemented the increase and decrease of volume it can be achieved by thumb index and little finger to increase and decrease we should move index and thumb fingers simultaneously and to set the volume we need to close the little finger. Brightness operation can be achieved by using all the five fingers if we open the hand i.e. the palm facing the camera the brightness will be increasing and to decrease the brightness we need to close all the fingers then the decrease of brightness operation will be performed.



Conclusion

This paper proposed a hand recognition system that we can use to control PC operations, using python programming language and OpenCV the proposed model has been working successfully. We can see that the model is capable of detecting the hand and working as a mouse. We can hover the hand to move the mouse on screen. The left click action of the mouse can be performed by combining the fingers. And we have performed 2 more operations like increasing and decreasing the brightness that can be achieved by using 5 fingers. Opening all the 5 fingers will increase the brightness and closing the 5 fingers will decrease the brightness and also perform operations of increasing and decreasing the volume. This can be achieved by using the thumb index and little finger movement of the thumb and index up and down will increase and decrease the volume and closing the little finger for setting the volume .

Future Scope

Right now we are able to perform above operations successfully, in our bigger picture we would like to perform more operations like right click of the mouse, scroll the page, Switching tabs on a browser is also on the line facial expression detection are in the line

References

- [1] Kabid Hassan Shibly, Samrat Kumar Dey, and Md. Aminul Islam, “Design and development of hand gesture based virtual mouse,” *Researchgate*. .
- [2] Mayur V. Gore and K. Dr. Anil R, “Human Computer Interaction using hand

gesture recognition - IJERTMayur V. Gore," *jert*. [Online]. Available [Accessed: 22-Oct-2019].

[3] P.Agrawal, K.Gupta. Mouse movement through finger by image grabbing using sixth sense technology. International journal of engineering science and advanced technology, volume-2, Issue-2, pages 245 – 249,2012

[4]. Real Time Hand Gesture Recognition System FOR DYNAMIC APPLICATIONS. Siddharth S. Rautaray¹, Anupam Agrawal² [5]. Gesture recognition –A Review by Miss Kawade Sonam P¹, Prof V.S. Ubale

[5].Z. Xu, K. Wu and P. Hu, "Ultrasonic waves based gesture recognition method for smartphone platform", *Computer. Eng. Appl.*, vol. 54, no. 2, pp. 239-245, Jan. 2018.

[6]. Camillo Lugaressi, Jiuqiang Tang, Hadon Nash, Chris McClanahan, Esha Ubweja, Michael Hays, Fan Zhang, Chuo-Ling Chang, Ming Guang Yong, Juhyun Lee, Wan-Teh Chang, Wei Hua, Manfred Georg and Matthias Grundmann "In MediaPipe: A Framework for Building Perception Pipelines" arXiv:1906.08172v1 [cs.DC] 14 Jun 2019

[7]. Fan Zhang Valentin Bazarevsky Andrey Vakunov Andrei Tkachenka George Sung Chuo-Ling Chang Matthias Grundmann Google Research "MediaPipe Hands: On-device Real-time Hand Tracking 1600 Amphitheatre Pkwy, Mountain View, CA 94043, USA.

[8]. M. Naveen Kumar , A. Vadivel "OpenCV for Computer Vision Applications" Proceedings of National

Conference on Big Data and Cloud Computing (NCBDC'15), March 20, 2015

[9]. Tran, DS., Ho, NH., Yang, HJ. et al.,(2021) "Real-time virtual mouse system using RGB-D images and fingertip detection" " Multimedia Tools and Applications volume 80, pages10473–10490 (2021)

[10] S. S. Abhilash, L. Thomas, N. Wilson & C. Chaithanya, "VIRTUAL MOUSE USING HAND GESTURE," International Research Journal of Engineering and Technology (IRJET), vol. 05 issue: 4, April 2018