Hand Gesture Mouse Control

Indian Institute Information Technology, Allahabad

Arpit Bhatt(IIT2018003), Divyanshi Raisinghani(IIT2018022), Suneet Kumar Makkar(IIT2018090), Kartic Choubey(IIT2018181), Chinmay Sethi(IIT2018182)

Abstract - This project is a mouse simulation system which performs all the functions performed by your mouse corresponding to your hand movements and gestures. Simply speaking, a camera captures your video and depending on your hand gestures, you can move the cursor and perform left click, right click, drag, select and scroll up and down. The predefined gestures make use of only three fingers marked by different colors.

Keywords - colour detection, contour, gestures, mouse simulation

I. Introduction

The project "Hand Gesture Mouse Control" is developed aiming to better the process of human-computer interaction. It aims to provide the user a better understanding of the system and to let them use alternate ways of interacting with the computer for a task. The task here is to control the mouse even from a distance just by using hand gestures. It uses a program in python and various libraries such as PyAutoGUI, Numpy and image processing module OpenCV to read a video feed which identifies the users' fingers represented by three different colors and track their movements. It retrieves necessary data and implements it to the mouse interface of the computer according to predefined notions. The project can be useful for various professional and non-professional presentations. It can also be used at home by users for recreational purposes like while watching movies or playing games.

II. Motivation

The project's primary aim is to improve the scope of human and computer interaction by developing an effective alternative way of controlling the mouse pointer and its various functions such as left click, right click, scroll up, scroll down and selection. It helps user interact with the computer from a considerable distance without any issue and efficiently without actually touching the mouse. It also decreases the hardware requirement for the interaction by eliminating the necessity of a mouse. All the user needs is a web camera (which is mostly present in all laptops these days) which can record real-time videos. The main objectives of the project are as follows: -

a. Obtain input video feed

b. Retrieve useful data from the image to be used as input

c. Filter the image and identify different colors.

d. Track the movement of colors in the video frame.

e. Implement it to the mouse interface of the computer according to predefined notions for mouse pointer control.

III. BACKGROUND

Vision based computer mouse control using hand gestures

By Sandeep Thakur, Rajesh Mehra, Buddhi Prakash

This paper delineates a vision based interface for regulating a computer mouse via 2D hand gestures. The evolution of Human Computer Interaction (HCI) has diverted the interest of researchers towards natural interaction techniques in recent years. Numerous applications of real time hand gesture based recognition in the real world have been deployed where we interact with computers. Hand gestures rely upon camera based color detection technique. This method mainly focuses on the use of a Web Camera to develop a virtual HCI device in a cost effective manner. This paper proposes a vision based system to control various mouse activities such as left and right clicking using hand gestures to make the interaction more efficient and reliable.

Mouse Cursor Control System Based on Hand Gesture

By Horatiu-Stefan Grif

The apparition on the market of the low-cost webcams with, at least, satisfactory qualities open up new directions regarding the implementation of human computer interaction (HCI) interfaces. The paper presents a HCI interface for mouse cursor control. The purpose of the implemented solution is to control the mouse cursor by user hand gestures captured through a webcam. For improving the gesture recognition based on the fluctuation of illuminance levels the finger strips color detection was used. The results reveal the good behavior of the system in low light condition.

IV. METHODOLOGY

The vision based cursor control using the hand gesture system will be developed in Python language, using the OpenCV library. The system could control the movement of a Cursor by tracking the users' hand. Cursor functions were performed by using different hand gestures. The system has the potential of being a viable replacement for the computer mouse.

a. **Obtain input video feed**

It decreases the hardware requirement for the interaction by eliminating the necessity of a mouse. All the user needs is a web camera (which is mostly present in all laptops these days) which can record real-time videos. It helps users interact with the computer from a considerable distance without any issue and efficiently without actually touching the mouse.

b. **Retrieve useful data from the image to be used as input**

The first thing that we do is convert the captured video into HSV format. Now the user gets to calibrate the color ranges for three of his fingers individually.The user has an option to use the default settings as well. Depending on the calibrations, only the three fingertips are extracted from the video, one by one.

c. **Filter the image and identify different colors.**

Finding contours in the mask relevant to that color range. Discarding contours of irrelevant areas using area filters. Finding the largest contour amongst the remaining ones and applying a method of moments to find its center.

d. **Track the movement of colors in the video frame.**

The thumb, with yellow color is responsible for the position of the cursor. Now the three centers are sent for deciding what action needs to be performed depending on their relative positions

e. **Implement it to the mouse interface of the computer according to predefined notions for mouse pointer control.**

Depending upon its output, the program will carry out either of the following using the PyAutoGUI library: free cursor movement, left click, right click, drag/select, scroll up, scroll down.

V. IMPLEMENTATION

The implementation of the code consists of several functions.

Starting with taking the video from the webcam and converting into colour space. As the project requires track hand movement, HSV colour format is used.

The project will first ask the user for a setting for calibrating all the three colours for their HSV values,which is done by by function calibrateColor() for all the three colours using three different windows for each.User can set the value so that the corresponding colour the detected properly or can used default value by pressing d.For removing the noises erosion and dilation is done.Then these noise free mask are sent for future calculation.After calibration user enters mouse simulation mode by pressing “p” button.

For performing different actions of the mouse relative position of the centroid of all the three colours are taken in consideration.

Location of each of the three centers involves:

- Finding contours in the mask relevant to that color range.
- Discarding contours of irrelevant areas using area filters.
- Finding the largest contour amongst the remaining ones and applying a method of moments to find its center.

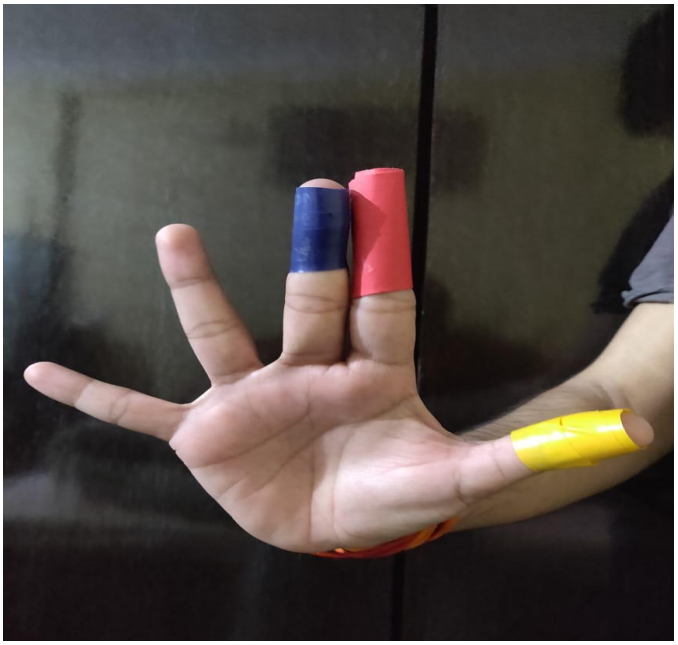
For deciding the position of the cursor the position of the centroid of yellow colour is responsible.While for the other operations of the mouse relative position of all the centroids is responsible.

As there can be noise in the video captured by the camera which can create vibration . On scaling up, these vibrations create a lot of problems with the accuracy of the cursor position. To reduce this in the cursor, we make use of differential position allocation for the cursor. We compare the new center with the previous position of the cursor. If the difference is less than 5 pixels, it is usually due to noise. Thus, the new cursor position is inclined more towards the previous one. However, a larger difference in previous position and new center is considered as the movement given by the user and the new cursor position is set close to the new center.

The code contains chooseAction() function which decides which function of mouse is to be performed.
Function allowed -

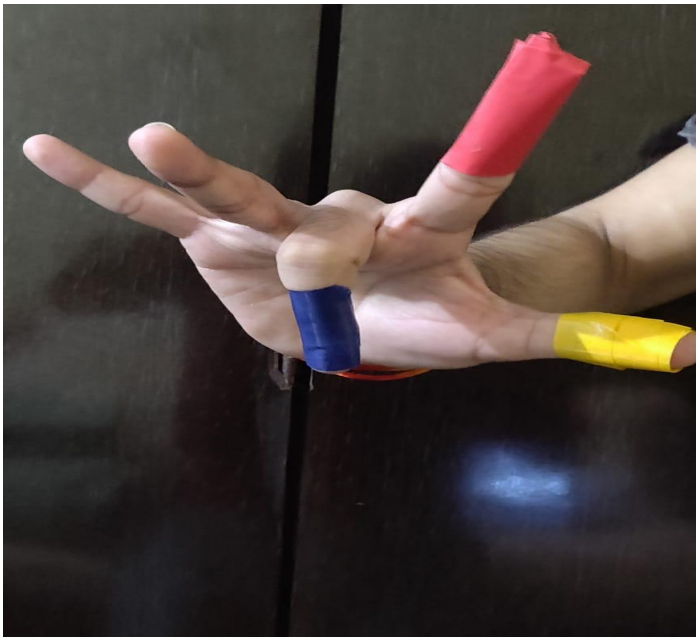
1. Cursor movement

2. Left click

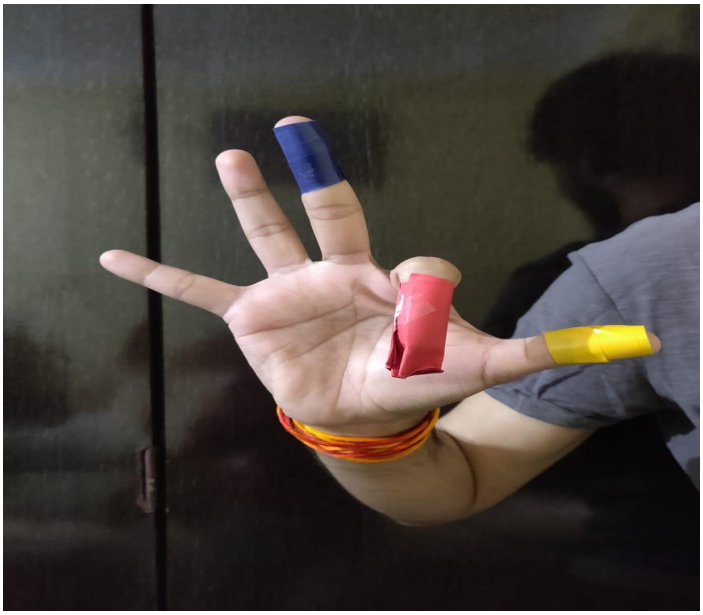


3. Right click

4. Scroll up



5. Scroll down



6. Drag/select

VI. Project specifications

Software Specifications:

1. 64-bit Operating System: Windows 8 or Higher
2. OpenCV 2.4.9 needs to be installed prior to running.
3. Windows Administrator permissions are needed for some parts of the program to function properly.

Hardware Specifications:

1. A Webcam

Environment Specifications:

1. A clear white background
2. There should be no other objects (especially red, blue, yellow coloured) in front of the webcam other than on the fingers.

VII. Conclusions

The vision based cursor control using the hand gesture system was developed in Python language, using the OpenCV library. The system could control the movement of a Cursor by tracking the users' hand. Cursor functions were performed by using different hand gestures. The system has the potential of being a viable replacement for the computer mouse, however due to the constraints encountered; it cannot completely replace the computer mouse. The major constraint of the system is that it must be operated in a well-lit room. This is the main reason why the system cannot completely replace the computer mouse, since it is very common for computers to be used in outdoor environments with poor lighting conditions. The accuracy of the hand gesture recognition could have been improved, if the Template Matching hand gesture recognition method was used with a machine learning classifier. All the operations which were intended to be performed using various gestures were completed with satisfactory results.

VIII. Contribution

1. Arpit Bhatt - Gaussian Control & Counter Extraction.
2. Divyanshi Raisinghani - Skin Detection
3. Suneet Kumar Makkar - Background Subtraction.
4. Kartic Choubey - Morphology
5. Chinmay Sethi - Converting Colorful Image to HSV.

IX. Progress After C2

According to the presentation held in C2, we were asked to improve the accuracy of our project for C3.
We have finally improved over the smoothness of the cursor movement and the clicks have been made more accurate. Earlier, the gesture recognition wasn't clear enough and quite random but now we have improved over the recognition of the hand gestures to the maximum extent so that now the hand gestures are identified correctly and accordingly the apt functions are performed.

References

[1] https://www.researchgate.net/publication/281112512_Hand_Gesture_Recognition_and_Cursor_Control#readp

[2] https://www.researchgate.net/publication/336304431_Design_and_Development_of_Hand_Gesture_Based_Virtual_Mouse

[3] https://www.google.com/url?sa=t&rct=j&q=&esrc=s&source=web&cd=&cad=rja&uact=8&ved=2ahUKEwjKuJ2t7e3rAhVxyDgGHUDuDPwQFjABegQIChAE&url=https%3A%2F%2Fwww.ijecs.in%2Findex.php%2Fijecs%2Farticle%2Fdownload%2F2460%2F2274%2F&usg=AOvVaw3GGWlN3eSh-cv6taTlTZKO

[4] https://github.com/aastavsashasen/SEN-Virtual-Mouse

[5] https://www.researchgate.net/publication/319493604_Controlling_Mouse_Cursor_Based_on_Image_Processing_and_Object_Detection

[6] https://www.geeksforgeeks.org/create-air-canvas-using-python-opencv/amp/#referrer=https://www.google.com