*May - June 2020*
*ISSN: 0193-4120 Page No. 5018-5022*

# Finger Gesture Mouse Using OpenCV

**[1]Abhishek Panda, [2]Yerriswamy T, [3]Aisha Aslam, [4]Anusha S, [5]Amal Mirwaiz**

[1,3,4,5]UG Student, [2]Assisstant Professor
[1,2,3,4,5]School of Computing and Information Technology, Reva University,
Bengaluru, Karnataka, India
[1]pupun123abhishek5@gmail.com, [2]yerriswamy.t@reva.edu.in, [3]aisha.aslam1505@gmail.com
[4]anushasahadev11@gmail.com, [5]amalmrwz@gmail.com

***Article Info***
***Volume 83***
***Page Number: 5018-5022***
***Publication Issue:***
***May - June 2020***

***Article History***
***Article Received:****19 November 2019*
***Revised:*** *27 January 2020*
***Accepted:*** *24 February 2020*
***Publication:*** *16 May 2020*

**Abstract**
In this paper the main idea behind finger gesture mouse is to make the work of human much easier and better with Human Computer Interaction, thus reducing of time computing and to increase the user comfort during the use of system. It is possible with python OpenCV, PyAutoGUI and NumPy packages. We have developed a PC application which is used to follow the color finger gesture and then execute the functions accordingly. This is an alternative of physical mouse. This can be used at all the places where reaching to a physical mouse is a bit difficult and inconvenient.

***Keywords:*** *Gesture Mouse, OpenvCV, Machine Learning, Image Processing, PyAutoGUI.*

## 1. Introduction

As the technologies are getting advance day by day the devices are enhanced into a compact size. Some devices have gone wireless and some of them gone latent. This project proposes a system that could make some of the devices go latent within the future that's the future of HCI (Human-Computer Interaction). The reason for this task is to build up a virtual mouse utilizing finger motion

The universally accepted language, hand gesture is most expressive and efficient. It is expressive and efficient for the dumb and deaf to understand or used in conferences, during lectures or while watching movie. This experimental project the finger gesture mouse uses webcam fixed to the laptop or desktop for high definition recording which captures the shots of images formed using different colours on their finger tips or finger caps like color-green, color-red, color-blue. We split the project into different phases such as image processing, extraction of valid image, recognizing the image, executing the command provided by the image.

The principle motivation behind picture handling is catching various pictures utilizing various hues at that point extracting the valid image. Subtracting the commotion and foundation taking the handled picture for perceiving the picture and continuing with their capacity determined for this following signal.

By this, we are aiming to make a mouse that is at large finger gesture software for our laptops and desktops with a fixed webcam. This project performs regular operations as moving cursor, right click, left click, scroll which could be used during conferences or while watching movie. If a high-resolution webcam is used, then it could be used by lectures or in meeting where it could be operated from afar. All the operations that a mouse can do this finger gesture mouse can perform the same functions.

## 2. Related Work

"Virtual mouse implementation using open CV"[1] Proposed a model made using Open CV and Anaconda was proposed. Webcam is used to record movements made by hand. Recognition of these movements is performed using the methodology of machine vision. Open CV video capture is used to capture the live video data. Using color variance methods, the coordinates that are related to the observed color may be used to control the cursor. This model's principal weakness is that it requires various identifications, various inputs / gestures.
"Design and Development of Hand Gesture Based Virtual Mouse"[2] Proposed an approach using hand movements, a suggested layout mouse control, icon search, and other functions are introduced. Webcam is used for video entry, and the built-in OpenCV functions are used to detect midpoints. This model's biggest downside is that it is not as efficient and precise as other gesture-based UI solutions.

*May - June 2020*
*ISSN: 0193-4120 Page No. 5018-5022*

"Hand Mouse Interface using Virtual Monitor Concept for Natural Interaction"[3] The paradigm being proposed specifies that the virtual display enables the hand mouse to navigate a virtual space. The method was fairly simple to execute and demonstrated excellent accuracy and insight. Kinect camera is used to record and remove physical features from images. This model's downside is that it relies solely on the Kinect sensor. This is not as easy to use the entire body for mouse manipulation as using only the paws.

"Hand Gestures Mouse Cursor Control"[4] Proposed model code portion of the system's source code was written in C language using the OpenCV software. The device had been checked on the hand pad for different degrees of daytime illuminance. The authors perform 20 hand movements for increasing perceived degree of illuminance. Web camera (A4Tech PK-635 M), catches the picture of the human hand put on a hand pad and it senses movement and produces a order for it for the moment of the cursor. The downside is the technology cannot be utilized every minute of any day as daylight changes over a few seconds from a luminance value to another depending on the conditions.

## 3. Methodology

### A. System Design and Analysis

Fig. 1 shows the basic idea of what the application is doing with the video which is provided by the OpenCV package through the frame. Later the video tracks the centroid and moves the mouse accordingly as per the functions defined in the program.

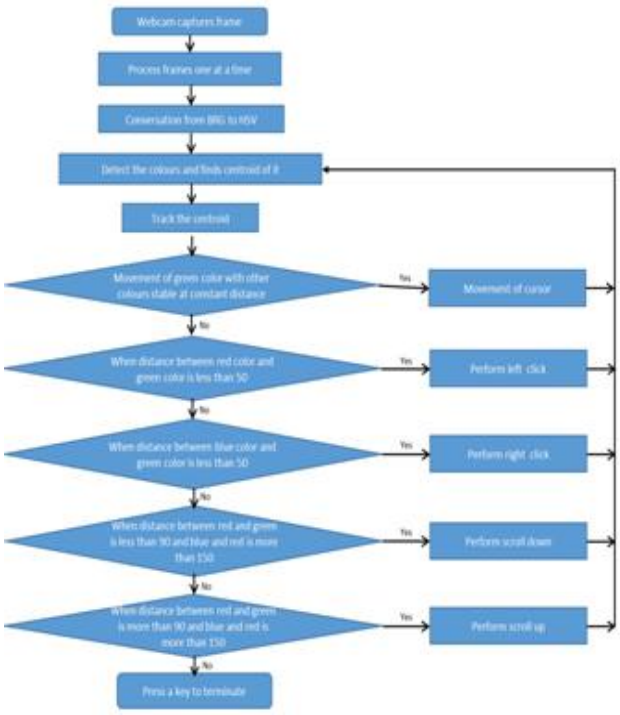


Figure 1: System design flow chart

### B. Software used

A computer vision-based mouse has been programmed to make the work of a person easier with the help of OpenCV, NumPy, PyAutoGUI which are the packages used in python. Basically, we have used Anaconda as our platform on which we are building our program.

#### 1) Anaconda

Anaconda is an open-source free software which is used for programming Python and R programs for specific computing like data science, machine learning applications, large scale data processing, predictive analytics, etc. Anaconda consists of desktop GUI application called Anaconda Navigator and conda prompt which is a command prompt for Anaconda. The default applications that are available in Anaconda are Jupyter Lab, Jupyter Notebook, QtConsole, Spyder, Glue viz, Orange, RStudio, and Visual Studio. Anaconda also has a cloud called Anaconda Cloud. This is fundamentally a mix of all the product in one asylum to create an application. We have used Anaconda to run our python code which uses all the packages of python for development. And even high-speed processing which is provided by Anaconda for the development of application.

#### 2) Spyder

Spyder is an astonishing coherent condition written in Python, for Python, and organized by and for researchers, specialists and information examiners. It offers the combination of all the advanced editing, analysis, debugging, and profiling functionality of a development tool in a single package. Spyder is present in anaconda navigator like a host application. This helped us to execute and run the undertakings that we had written in python with procuring all the gatherings required for the application to run.

#### 3) OpenCV

OpenCV (Open Source Computer Vision Library) is a is an open source computer vision and machine learning software library. It was worked for PC vision application and to quicken the utilization of machine in items. The library has in excess of 2500 calculations which are utilized to distinguish and perceive faces, identify objects, classify human actions in videos, track camera movements, track moving objects, etc.

Along these lines, the OpenCV library assumes a significant job in our application. It distinguishes the picture from the camera and afterward utilizes a similar picture for development of mouse.

#### 4) PyAutoGUI

PyAutoGUI helps us to control the mouse and keyboard through python scripts. It urges us to speak with the mouse through the python code.

*May - June 2020*
*ISSN: 0193-4120 Page No. 5018-5022*

The application imports all the packs required for the application to work.

### C. Software Implementation

Many default limits are portrayed in the start of the program so it to be default all through the program. The cut-off focuses are later called during the execution of the program.

The default esteems are set before composing the program for the capacities to utilize later when no worth is given by client or when no info is taken from the client. We define the state change which changes the state of performance in the application like

- when the mouse is on
- when the mouse is off
- when to show centroid
- when not to show centroids
- we can even go to recalibration mode and calibrate all the colours once again.

Next, we define a mask function in which we perform morphosis which contains erode and dilate functions present in the cv2.inRange function of cv2 package. As a result, the frame filters out other color by undergoing morphosis which results in masked frame. Later we have a draw_centroid function which find the centroid of the color and returns the center of all the color in the frame.

The Calibrate_color function which is used to filter the required colored objects from the background and convert BRG to HSV color code. Here we call the mask function and even the Erode and dilated function. This encourages us in clearing the foundation and evacuating the commotion by taking just the shading required for the working of the mouse. Function to convert BRG image HSV :

hsv = cv2.cvtColor(frame, cv2.COLOR_BGR2HSV)

Choose_action is a capacity which takes the situation of the shading as info which are additionally utilized for development of the mouse. The partition can help us in setting assorted request to different components of the mouse. It consists of moving, dragging, left click, right click, scrolling up and scrolling down hence sending out the allocated function as return statement.

Perform_action take the input from choose_action and Moves the cursor accordingly which involves all the functions of the mouse in it. This function is the main part of the program which uses PyAutoGUI package to move the mouse. PyAutoGUI uses the mouse action which are present in the package and then matches with the input taken from the choose_action function to perform the allocated action. Code to start webcam for video capturing:

cap = cv2.VideoCapture(0)

We use the webcam which present on the laptop, that's why we use 0 as the input to the cv2.VideoCappture() function, else if we had other camera input devices we can use other numbers for representing them. The calibration work is called while composing the fundamental capacity. This calibrates all the color and decides the range at which it lies. After that the cv2 functions creates the frame through which it takes input and Moves the mouse. The cv2 work is the bundle which is available in OpenCV. We additionally utilize a flip work in light of the fact that the picture is rearranged and can't move in the straight organization until and except if it is swapped and made straight. Toward the finish of the program we crush all the windows which we have made previously with cv2.destroyAllWindows() function. We also use cam.release() closing the web-camera which was called by cv2_function. Code to flip the frame which is taking video:

frame = cv2.flip( frame_inverse, 1)

## 4. Experimental Results

The function of the mouse which are defined from choose-action are based on the distance between the color.

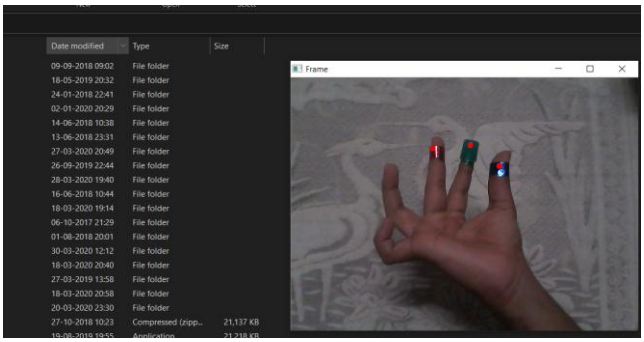

Figure 2: Fingers at proper distance to operate free cursor moment.

When distance between the blue_color and green_color is less than 50 computer units then a right click function is performed stopping every other function.

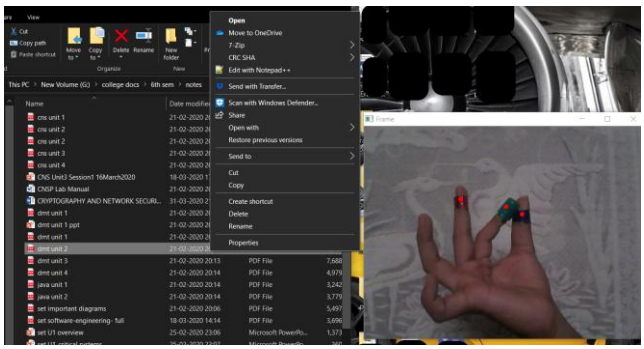

Figure 3: Fingers performing right click.

When the distance between red_color and the green_color is less than 50 computer units then a left click function is performed.

*May - June 2020*
*ISSN: 0193-4120 Page No. 5018-5022*

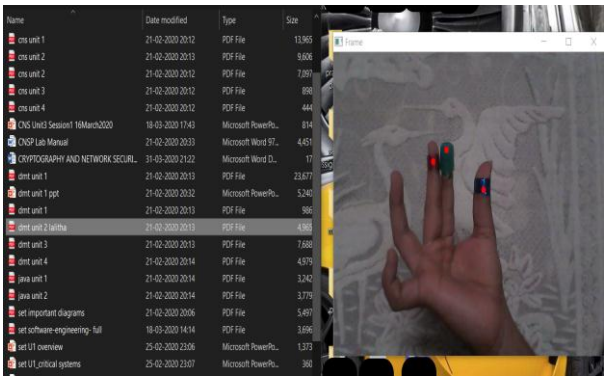

Figure 4: Fingers performing left click.

When the distance between green_color and red_color is less 90 computer units and the distance between blue_color and red_color is more than 150 computer units then scroll down function is performed .

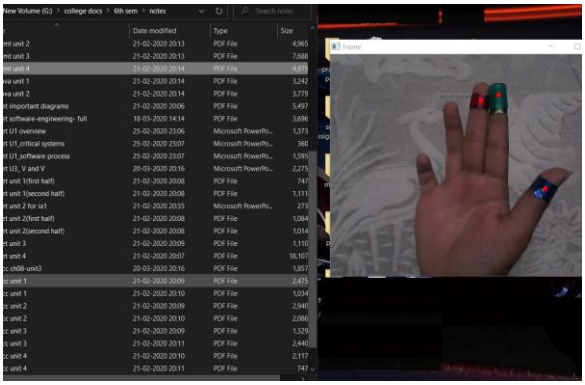

Figure 5: Fingers performing scroll down action.

When the distance between green_color and red_color is more than 90 computer units and the distance between blue_color and red_color is 150 computer units then scroll up function is performed.

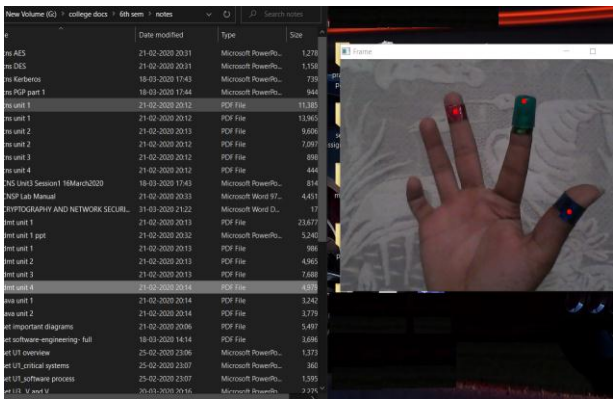

Figure 6: Fingers performing scroll up action.

And when none of these conditions applied the cursor is free to move but at the point when all the three shading are near one another or having separation not exactly 60 computer units a drag function is performed.

## 5. Applications

Finger signal can be utilized rather than optical mouse and stylus during:

- Courses and Introductions
- Gatherings
- Intuitive Classes

This can be used for hands-free navigation in smart home entertainment systems.
This can be actualized in the gaming business too.

## 6. Conclusion

Application was executed in Python utilizing the OpenCV library. The program viably engaged the user to control the cursor and do distinctive cursor operations using various hand gestures. Concealing strips were similarly utilized for the structure to see flags easily. Regardless of the way that the structure demonstrated constraint of being a potential option for the PC mice, it besides had a couple of wasteful edges about it. the PC mice, it furthermore had a few inefficient angles about it. Exceptionally, the light conditions of the room factored into the performance of the system. Since a customary mouse can be used in circumstances with poor lighting conditions without impacting the efficiency, this system of HCI still won't be preferred over a PC mouse for a basic system build. The execution could be also improved with the usage of further sophisticated hardware. For instance, a high-definition camera could be utilized for image capture and smart gloves. But those advancements could be much more expensive, hence defeating the purpose. Despite all this, almost all the cursor functions were successfully performed using the suggested system without any significant obstacles. Right now, is still really suitable in different circumstances as a dynamically invaluable kind of Human PC Association.

## 7. Future Enhancement

Innovation advancement in the current world is significant for financial development and the improvement of human prospects. The project's potential development involves rendering the fingertip detector module invariant to changes in lighting and 3D position panel prediction that can be used to improve 3D object reality. In the future, to detect contact events on the simulated screen, we can take advantage of the other visual features (e.g. character form, symbol feature) in the human-computer interface. We would likewise have the option to gather land data from a virtual guide utilizing a virtual mouse. In the present e-learning, the PC marker is changed to do instead of a marker and go about as a mouse pointer while giving it the upside of each of the a mouse's highlights. After through frame capture, the current implementation performs a series of image processing operations on the whole file. Although the tracker's speed is pretty good, we may be able to substantially increase the tracking rate by then monitoring local hand features (e.g., fingertips).

*May - June 2020*
*ISSN: 0193-4120 Page No. 5018-5022*

**8. Acknowledgment**

We express our sincere gratitude towards Mr. Yerriswamy T for his support and guidance throughout the project on the topicFinger Gesture Mouse Using OpenCV.

**References**

[1] Kollipara Sai Varun,I. Puneeth and Dr. T. Prem Jacob,"Virtual mouse implementation using Open CV",Sathyabama University, Chennai, India,Third International Conference on Trends in Electronics and Informatics (ICOEI 2019), CFP19J32-ART; ISBN: 978-1-5386-9439-8, 2019

[2] Kabid Hassan Shibly, Samrat Kumar Dey, Md. Aminul Islam and Shahriar Iftekhar Showrav,"Design and development of hand gesture based virtual mouse",Dhaka International University, Dhaka, Bangladesh,1st International Conference on Advances in Science, Engineering and Robotics Technology 2019 (ICASERT 2019), ICASERT. 2019.8934612, October 2019

[3] Changhyun Jeon, Oh-Jin Kuon, Dongil Shin and Dongkyoo Shin, "Hand-mouse interface using virtual monitor concept for natural interaction",Sejong University, Seoul 05006, South Korea, 10.1109/ACCESS.2017.2768405, Dec 2017

[4] Marian-Avram VINCZE and Horaţiu-Ştefan GRIF, "Hand gestures mouse cursor control", University of Tîrgu Mureş Nicolae Iorga Street, no.1, 540088, Tîrgu Mureş, Romania, Scientific Bulletin of the „Petru Maior" University of Tîrgu Mureş Vol. 11 (XXVIII) no. 1, 2014, ISSN 2285-438X, 2014.