FINGER CONTROLLED MOUSE POINTER

15IT376L - MINOR PROJECT II

Submitted by

Ansh Vinod Motwani (RA1611008010661)

Ayush Roy (RA1611008010641)

Saumya Awasthi (RA1611008010414)

for the assessment of 3rd year Minor Project

in the

DEPARTMENT OF INFORMATION TECHNOLOGY

SRM INSTITUTE OF SCIENCE AND TECHNOLOGY

KATTANKULATHUR

APRIL 2019

SRM INSTITUTE OF SCIENCE AND TECHNOLOGY

KATTANKULATHUR

BONAFIDE CERTIFICATE

Certified that this Minor project report **"FINGER CONTROLLED MOUSE POINTER"** is the bonafide work of **"Ansh Vinod Motwani", "Ayush Roy", "Saumya Awasthi"** who carried out the project work under my supervision at SRM Institute of Science and Technology, IT Department, Kattankulathur.

SIGNATURE	**SIGNATURE**
Ms. Monica Catherine	Dr. G. Vadivu
GUIDE	**HEAD OF THE DEPARTMENT** **Information Technology**

DECLARATION

We, **Ansh Vinod Motwani (RA1611008010661), Ayush Roy (RA1611008010641), Saumya Awasthi (RA1611008010414)**, studying in III year B. Tech Information Technology program at, SRM Institute of Science and Technology, Kattankulathur, Chennai, hereby declare that this project is our original work and we have not copied/ duplicated any material from sources like internet or from print media, excepting some vital company information/ statistics and data that is provided by the company itself.

Signature of the Students

Date:

Place:

ACKNOWLEDGEMENT

The success and the final outcome of this project required guidance and assistance from different sources and we feel extremely fortunate to have got this all along the completion of our project. Whatever we have done is largely due to such guidance and assistance and we would not forget to thank them.

We express our sincere thanks to the Head of the Department, Department of Information Technology, **Dr. G. Vadivu**, for all the help and infrastructure provided to us to complete this project successfully and her valuable guidance.
We owe our profound gratitude to our project guide **Ms. Monica Catherine**, who took keen interest in our project work and guided us all along, till the completion of our project work by providing all the necessary information for developing a good system.

We are thankful to and fortunate enough to get constant encouragement, support and guidance from all the Teaching staff of the Department of Information Technology which helped us in successfully completing our minor project work. Also, we would like to extend our sincere regards to all the non-teaching staff of the department of Information Technology for their timely support.

Ansh Vinod Motwani

Ayush Roy

Saumya Awasthi

ABSTRACT

Our project deals with controlling the mouse pointer and carrying out the functions of a trackpad with the help of hand gestures. Moving our hand would allow us to change tabs, double click an item, drag objects and carry out other simple tasks. In existing research, an external camera is used for better image capture and detection but that makes is uneasy to use in everyday life. Using an external camera also requires extra battery usage and has an added hardware cost. Since our software will be using the computer webcam, there would not be any extra cost. Our software can make everyday life very easy. If the computer is kept at a distance, the person would not have to get up but instead can interact with the computer by just gesturing with their hands. It could greatly help partially paralysed or bedridden people who cannot walk or get up to use their computer kept at a distance. There would no longer be any need to keep laptops on the bed while using it as it is harmful to keep the laptop on bed. This problem faces lots of issues as mostly those solutions cater to only a few gestures and for the remaining gestures there is a need to use the mousepad. But we plan on making a complete replacement for a mouse which will not require any mouse input. The one thing which would be unique about our project would be its precision. We will design it in such a way that the user can easily point to buttons and select anything on the screen without having trouble in moving the pointer to a very small distance. Our pointer will also be having mouse assist which will be helping the user to effortlessly maneuver the pointer. We aim to create a robust software which could actually be used in everyday life and does not remain as just a project. It would be as user friendly as possible and we will make it in a way that people can actually use it.

TABLE OF CONTENTS

Chapter No.	Title	Page No.
	Abstract	**iv**
	Table of Contents	**v**
	List of Figures	**vi**
	Abbreviations	**vii**
1	Introduction ……………………………………...........	1
2	Goals of the System …………………………….........	2
3	Requirement Analysis …………………………...........	3
4	Literature …………………………………………...........	4
5	Project Description ………………………………........	5
6	Implementation ………………………………….........	15
7	Problems and Drawbacks ……………………….........	18
8	Applications ………………………………………..........	19
9	Conclusion ……………………………………….........	20
10	Further work …………………………………….........	21
11	References ………………………………………..........	22

LIST OF FIGURES

1.1 List of steps ………………………………………. 7

2.1 Real time image captured ………………………… 8

2.2 Real time image captured is flipped ……………… 9

2.3 Flipped image in HSV format …………………..… 10

2.4 Red colour filtered image ………………………… 11

2.5 Green colour filtered image ………………………. 11

2.6 Blue colour filtered image ………………………... 11

2.7 Noise removed image with contour ………………. 12

2.8 Region and its centroid marked …………………... 13

3.1 Frame life cycle ………………………………….. 15

4.1 Detection of all three contours at once …………… 16

5.1 Implementation flowchart ………………………… 45

ABBREVIATIONS

HCI	Human Computer Interaction
OpenCV	Open Source Computer Vision Library
GUI	Graphical User Interface
FPS	Frames per second
IDE	Integrated Development Environment
HSV	Hue Saturation Value

Chapter 1 - INTRODUCTION

The computer world has advanced a lot in the past few years. From huge PC’s to small smartphones, innovation has become very important to us. Everyone gets bored of any technology within a few years. Earlier we needed to press buttons. Now it all works with just a touch. Human Computer Interaction has changed the way we interact with computers. It is the study of new methods and technologies needed to ease the human life. It is also meant to reduce the cost of the already present technology. In our project, we plan to create a software which is not only cost effective but easy to use. Using our software, people can carry out multiple mouse functions using hand gestures and without having to actually touch the mouse pad or the screen. The user need not be present extremely close to the system in order to carry out this process. It can be done from a distance. This reduces a lot of human effort.

The greatest challenge is develop an economically feasible application, which is also hard- ware independent. This is to ensure a person can easily make use of this application without having much difficulty in setting up the physical computer setup. Basically the main objective is to develop an application that can help a person to interact with the computer, and a virtual human computer interaction system, where no physical contact with the device is required and also to provide an easier HCI routine.

Chapter 2 - GOALS OF THE SYSTEM

The goals of this system are:

1. Capability to control the mouse without physically touching the system.
2. A system that ensures Vision based machine.
3. Capability of controlling mouse functions using hand gestures.
4. To eliminate the limitations of stationary hand.
5. To provide real time gesture based tracking system.

Chapter 3 - REQUIREMENT ANALYSIS

Software Requirements

- OpenCV
- Python 3
- Anaconda
- Spyder IDE

Hardware Requirements

- Microsoft Windows 7/8/10
- 2 GB RAM minimum, 8 GB RAM recommended
- 4 GB of available disk space minimum, 16 GB Recommended
- 1280 x 800 minimum screen resolution
- Integrated Webcam (1.3 Megapixels)

Chapter 4 - LITERATURE

OpenCV

OpenCV is an Open Source Computer Vision Library. It contains various functions required for computer vision. It encourages the use of machine perception and gives a platform for computer vision applications.

This library has more than 2000 optimized programs. These programs vary from each other which shows that the library is extensive. The library can be used to detect faces, track camera movement, track the moving object be it a mouse pointer or an image.

The OpenCV app has been downloaded more than 18 million times. It is extensively used by all sorts of users, from big multinationals to a regular college going student. The app provides an excellent learning opportunities and an area to execute and perform various small or big projects.

The library was originally developed by Intel and then further managed and developed by Willow Garage and then Itseez.

The library is cross platform and is free to use, which is an added bonus. It is licensed under open-source BSD license. It is most popular library that is free to use. This free to use factor helps OpenCV gain popularity among college students. It supports all languages C, C++, Python etc.

It has a vast community channel where various users can connect on a platform and get help and gain knowledge from each other. For a platform to be successful, community plays a huge role.

OpenCV is optimized to outperform other open libraries. It is very efficient and easy to use and get used to. It provides huge advantage to process real time programs. Its designs also allow the Library to full advantage of the multi core processor that is being used by the host PC and even hardware acceleration.

Image Processing

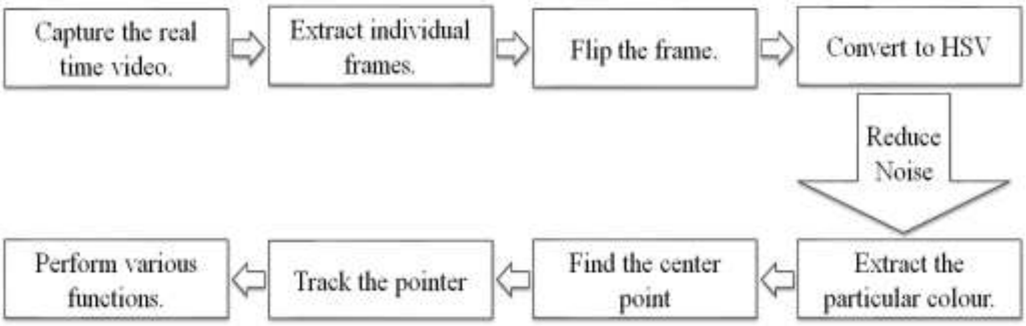
Digital image processing is the use of algorithms defined by libraries like OpenCV to perform image processing on digital images or the video. An image is analysed as it is clicked. Then the user is given tools to discard that image or process it any way that they want. The image acquisition is done using a web-camera in order to lower the budget of our project. For the system to work we need a sensor to detect the movement of the user. Since there are three colours that are being used red, green and blue. The webcam of the computer is used to as a sensor. The webcam capture the real time video at a variable frame rate and resolution which can be easily modified by the user. The quality of the video depends on the hardware capability of the system on which the project takes place. The frame rate and resolution can be modified in the system but it all depends on the budget of the project. A better hardware can provide better results and better frames but it won't affect the result as much as we have to isolate individual colours and not process the actual video. Each colour is individually isolated. In order to reduce the overall budget of the process we perform the following:-

- We use the inbuilt web camera instead of using an external web cam to capture the digital image as it would work just as good at a reduced cost.
- The video obtained from the webcam is divided into frames. Each video has a frames per second value. It can go very high but an ideal frame rate is 30. The video is split into individual frames.
- We then further process each frame individually so that we can isolated our three colours and then we can add functionality to the same.
- We have to use a particular algorithm to obtain this result
- We have to use a mathematical formula to decide the colour ranges for each individual colours. As these ranges will determine the end result.

Chapter 5 - PROJECT DESCRIPTION

This is a step by step description of each and every step carried out to successfully complete the project. It has broadly been defined into 10 steps. They are as follows:

1. Capture the real time video.
2. Extract the individual frames.
3. Flip the frames.
4. Convert to HSV form.
5. Extract the particular colours.
6. Merge the output of each colour extraction.
7. Reduce the noise.
8. Find the region and its centroid.
9. Track pointer according to the centroid movement.
10. Perform different actions with the different colours extracted.


Capture the real time video.
Extract individual frames.
Flip the frame.
Convert to HSV
Reduce Noise
Extract the particular colour.
Find the center point
Track the pointer
Perform various functions.

1.1 List of Steps

Capturing the real time video

For the system to sense the movement of hand we need a sensor. We used the web camera in our work for that sensing. The web camera captures the real time video that is it gives us the real time news feed. The captured image is at a particular resolution and at a particular frame rate. We can change the frame rate and resolution according to our needs.

The web camera captures the real time video. This is divided into different frames. These frames are then processed further.

2.1 Real time image captured

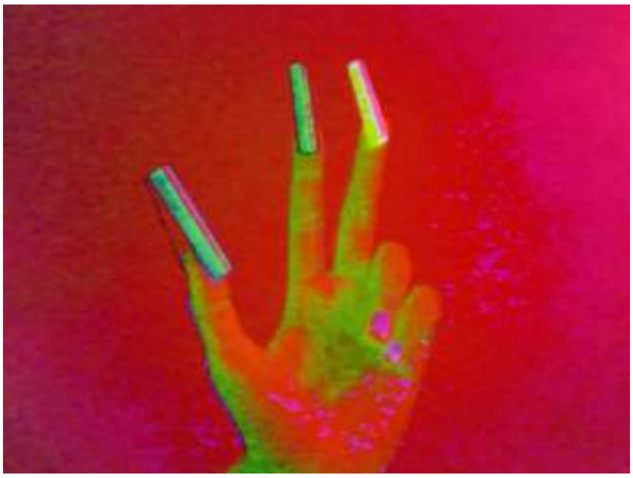
Flipping the image

When the camera captures the real time feed and converts it into the frames then the frame obtained is inverted. It is like the image we get when we stand in front of the mirror. So if we move our coloured strip to right it will in turn move to left and so will our mouse pointer. This will create a lot of confusion in operating the mouse pointer. Hence we need to flip each frame. This is done by vertically inverting the frame.

2.2 Real time captured image is flipped

Changing to HSV format

The frame then is converted to HSV format. The frame that we obtained is in BGR format so we convert it into HSV format. HSV format basically views in the hue, saturation and value channels. HSV separates *luma*, or the image intensity, from *chroma* or the colour information. Using the Hue component makes the algorithm less sensitive to lighting variations.

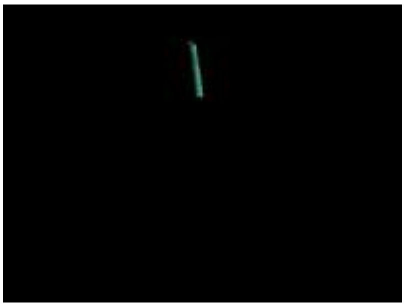


2.3 Flipped image in HSV format

Extract the colours and the merge the output

From the obtained frames the different colours that is red, green and blue are extracted. The colour range for each colour is defined in the HSV format. All the colour lying in that particular colour is extracted. After this all the three frames having separate colours is merged into a single frame and hence we obtain the image with only our desired colour range.

2.4 Red colour filtered image

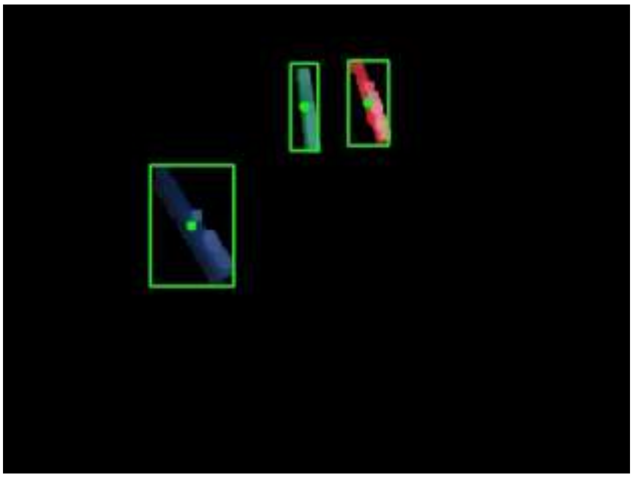


2.5 Green colour filtered image

2.6 Blue colour filtered image

Reduce the noise

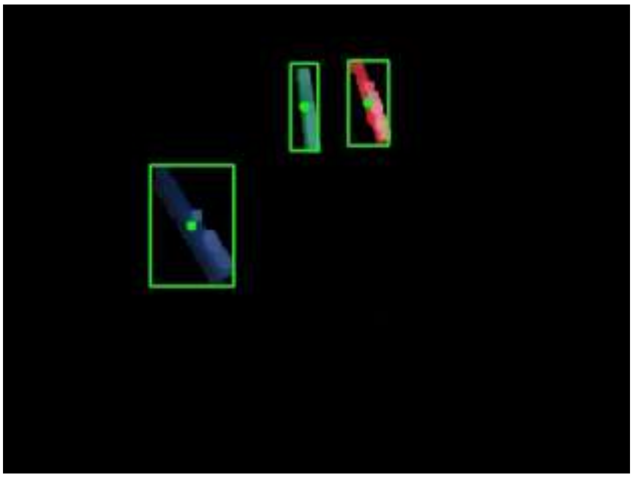
The frame that we have contains a lot of noise. This noise needs to be reduces to obtain a clearer image. For this we first do erosion and then we do dilation. In erosion we see the specified number of pixels if its majority portion is of a certain colour then all the others are made into that colour. In dilation the coloured image is dilated upto the specified number. After this we remove the false positives and false negatives from the image.



2.7 Noise removed image with contour

Find the region and its centroid

The frame has a lot of images of that colour so we find the image with greatest area of that colour and make a contour around that. Then we make a rectangle surrounding that contour. The centroid of the rectangle surrounding the contoured area is taken out.



2.8 Region and its centroid marked

Tracking and performing different actions

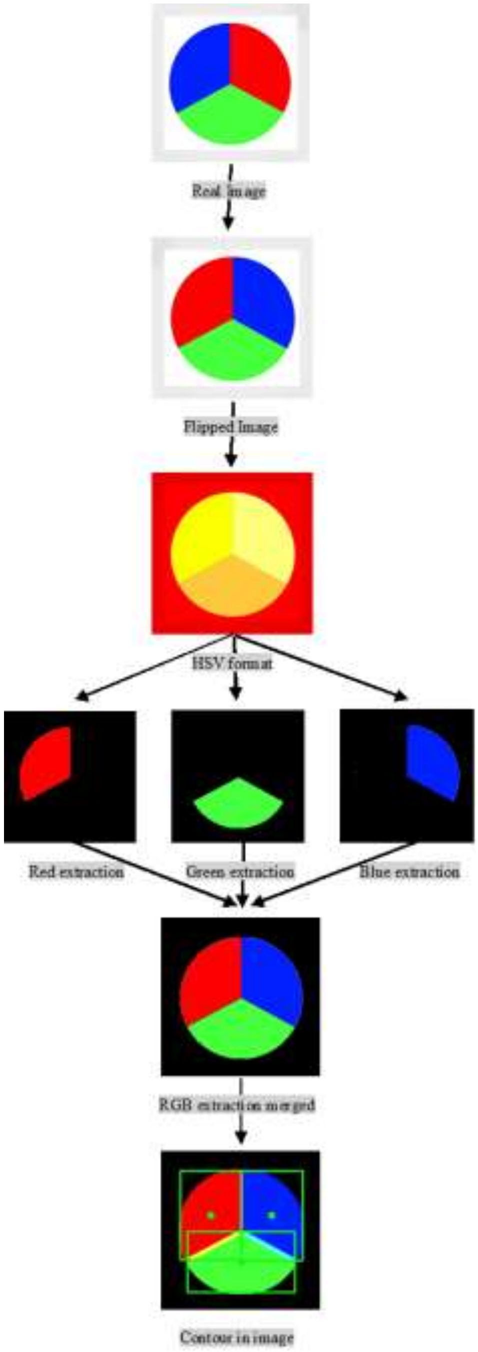
The tracking and different actions performed are done by the use of the inbuilt library in python named pyautogui. This library has many functions for left click, right click or just moving the pointer and many more.

The clicking actions are performed as follows:

- The movement of the mouse pointer will be controlled by green colour.
- If the blue colour shows, then the system will detect a right click.
- Similarly, all the three colours together will detect a right click.

Image Life Cycle

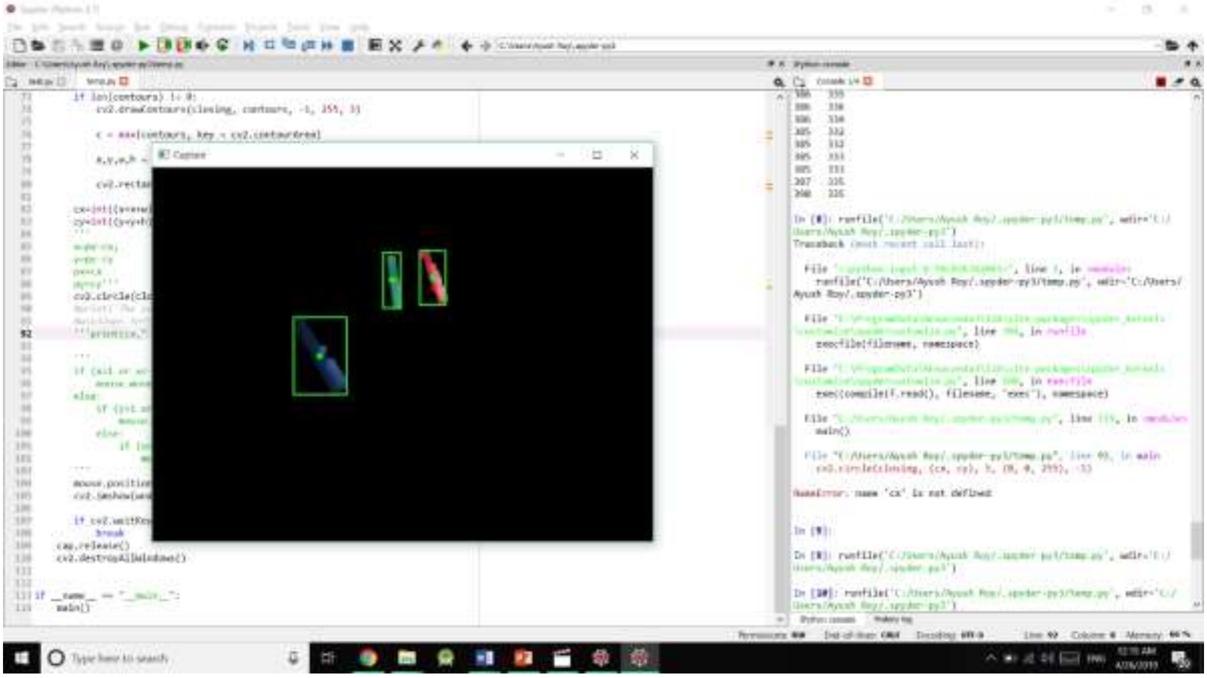
A single frame undergoes the following process from the time it is captured till the time any action is performed.



3.1 Frame life cycle

Chapter 6 - IMPLEMENTATION

For the implementation of our project, we run the project directly from the Spyder IDE. By simply pressing the “PLAY” button or pressing “F5” on the keyboard will begin the application. Once the project runs, a new Spyder window will open. On the window, you would be able to see the contour of all the fingers with colour strips on a black background. The movement of the mouse pointer will be controlled by the movement of the index finger with green colour on it. If the thumb with blue colour shows, then the system will detect a right click. Similarly, if the middle finger with red colour on it shows while the other two fingers are showing, then a left click will be detected. When just the index finger and middle finger show, then the system will detect a scroll action and on moving the fingers up or down, it will scroll in that direction.



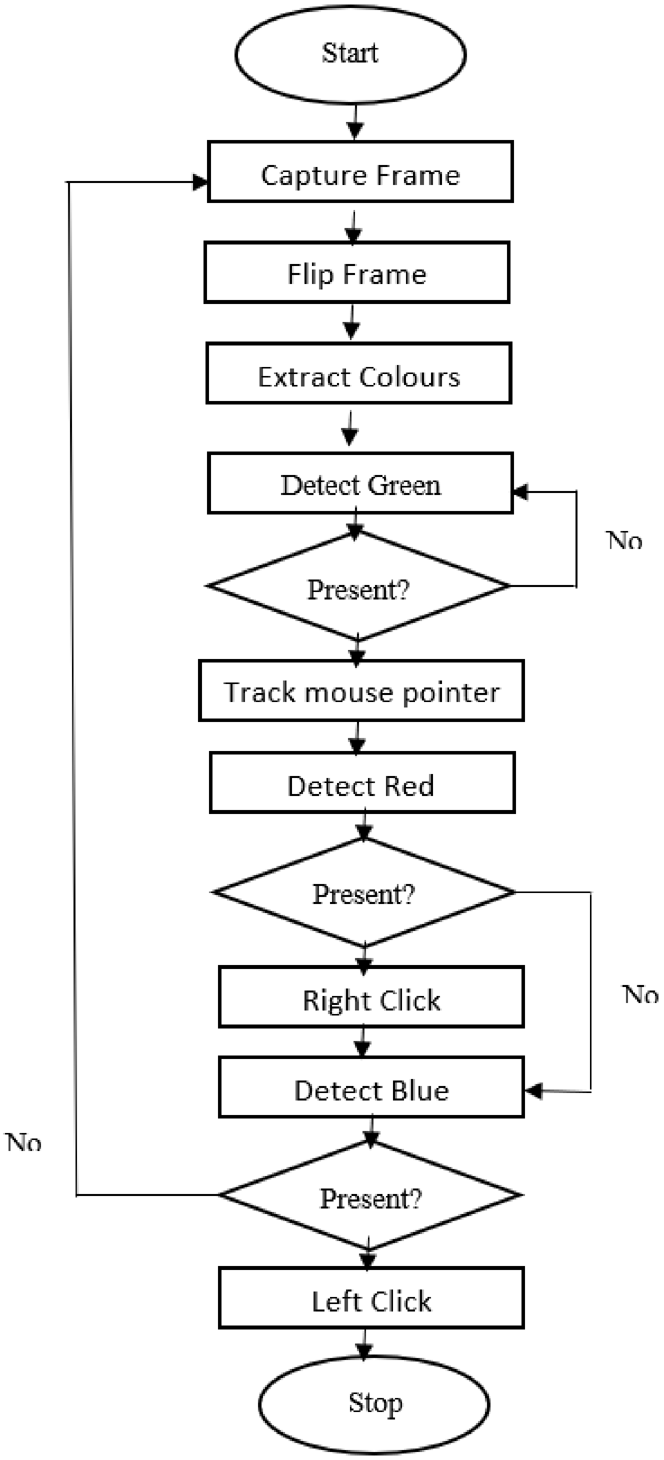
```
        if len(contours) != 0:
            cv2.drawContours(closing, contours, -1, 255, 3)

            c = max(contours, key = cv2.contourArea)

            x,y,w,h =

            cv2.rectan
...
    cap.release()
    cv2.destroyAllWindows()


if __name__ == "__main__":
    main()
```

```
In [8]: runfile('C:/Users/Ayush Roy/.spyder-py3/temp.py', wdir='C:/Users/Ayush Roy/.spyder-py3')
Traceback (most recent call last):

  File "[illegible]", line 1, in <module>
    runfile('C:/Users/Ayush Roy/.spyder-py3/temp.py', wdir='C:/Users/Ayush Roy/.spyder-py3')

  File "[illegible]", line [illegible], in runfile
    execfile(filename, namespace)

  File "[illegible]", line [illegible], in execfile
    exec(compile(f.read(), filename, 'exec'), namespace)

  File "[illegible]", line [illegible], in <module>
    main()

  File "C:/Users/Ayush Roy/.spyder-py3/temp.py", line [illegible], in main
    cv2.circle(closing, (cx, cy), 5, (0, 0, 255), -1)

NameError: name 'cx' is not defined


In [9]:
```

4.1 *Detection of all three contours at once*

Mouse Control Flow Chart

The following is the implementation flow chart for the mouse pointer.



5.1 Implementation flow chart

PyAutoGUI

This library is used to provide a Python module for GUI control. We use this library for defining all the major mouse movement functions. Some of those functions are as follows.

Mouse Movement : The moveTo() function is used to move the mouse to any specific location on the screen.

pyautogui.moveTo(100,100)

Mouse Clicks : The click() function defines any type of single or multiple click on the screen.

pyautogui.click(100,150,1,0,’left’)

Mouse Scrolling : The mouse scrolling up and scrolling down can be defined by the scroll() function.

pyautogui.scroll(10)

The various other mouse functions like double click or dragging can be made with a combination of the above stated functions.

Chapter 7 - PROBLEMS AND DRAWBACKS

The principal challenge was to accurately distinguish the hand with a webcam. We required a PC Vision library for this reason. Many are accessible however we chosen to proceed with OpenCV as it is the most famous and has been ported to numerous dialects and is bolstered on many working frameworks from Android to Windows. It has a decent library accumulation of standard picture preparing capacities. At that point we needed to first setup OpenCV on our IDE(Spyder IDE). That was an exceptionally dreary assignment. We additionally needed to get familiar with some fundamental utilization of OpenCV. For which we alluded to numerous instructional exercises on the web. In the wake of learning OpenCV, We needed to find out about the skin identification systems and picture handling methods like Foundation Subtraction, Picture Smoothening, Commotion Evacuation and Decrease. Presently, in the wake of distinguishing the hand effectively and mapping the signals, we needed to figure out how to utilize the Windows Programming interface all together tune the product with the Metro UI. For learning it, we fabricated some fundamental applications dependent on it. Along these lines, to put it plainly, there was a precarious expectation to absorb information. As, top of the line cameras and sensors are in all respects expensive we chose to run with a straightforward webcam. Along these lines, we chose to upgrade our product and its usefulness all together the disadvantage of utilizing a straightforward webcam. Additionally, for testing our undertaking we had an essential necessity of having a white foundation with no obvious part aside from our palm.

Chapter 8 - APPLICATIONS

There are various areas where our project can be used can be used to replace traditional devices, some of them are:

1. Without physically touching, PC control can be utilized to track gestures both accurately and powerfully. This helps the advancement of moderate vision based UIs that can be utilized in International Journal of Scientific and Research Publications, a wide range of instructive or on the other hand recreational applications or even in controlling PC programs.
2. For the most part, this framework is valuable for impaired individuals to carry out numerous PC exercises. It gives them a chance to collaborate with the PC society.
3. It can be used for programming, intuitive TV, computer game, commercial and other such action. It is utilized for perusing procedures.
4. On execution, the application influences the cursor to move with the assistance of the hand which is an automation of the work area.

Chapter 9 - CONCLUSION

The finger controlled mouse pointer was assessed and evaluated. The vision based cursor control utilizing hand motion framework was created in the Python language, utilizing the OpenCV library. The framework had the capacity to control the development of a Cursor by following the clients hand. Cursor capacities were performed by utilizing diverse hand signals. The framework has the capability of being a suitable trade for the PC mouse, anyway because of the imperatives experienced; it can't totally supplant the PC mouse. The significant imperative of the framework is that it must be worked in a sufficiently bright room. This is the principle motivation behind why the framework can't totally supplant the PC mouse, since it is extremely regular for PCs to be utilized in open air situations with poor lighting condition. The exactness of the hand motion acknowledgment could have been improved, if the Template Matching hand signal acknowledgment technique was utilized with an AI classifier. This would have taken much longer to actualize, yet the exactness of the signal acknowledgment could have been improved. It was hard to control the cursor for exact cursor developments, since the cursor was truly precarious. The majority of the activities which were proposed to be performed utilizing different motions were finished with palatable outcomes.

Chapter 10 - FURTHER WORK

We would improve the execution of our project particularly the gesture controlling. Moreover, we additionally need to reduce the reaction time of the project for cursor development so that it can totally be utilized to supplant our regular mouse. We are also intending to structure an equipment execution for the equivalent inorder to improve exactness and increase the usefulness to different spaces, for example, a gaming controller or as a universally useful PC controller. Other advanced execution incorporate the hand gesture acknowledgment stage to utilize the Template Matching strategy to recognize the hand motions. This strategy requires the utilization of Machine Learning and an AI classifier, which requires a significantly long effort to develop. However, it would have allowed the usage of more hand gestures which thus would allow the utilization of more mouse capacities. Another usage of this technology would be to utilize the PC to prepare the hearing impaired and visually challenged people.

Chapter 11 - REFERENCES

- https://www.opencv.org
- https://ieeexplore.ieee.org/document/7307391
- https://www.sciencedirect.com/science/article/pii/S2351978918304438
- https://www.irjet.net/archives/V3/i10/IRJET-V3I10233.pdf
- https://pyautogui.readthedocs.io/en/latest
- https://www.stackoverflow.com
- https://forum.openframeworks.cc/t/finding-centroid-of-image-with-opencv/2275
- https://www.pyimagesearch.com/2016/02/01/opencv-center-of-contour/