

---

## Virtual Mouse Implementation using Color Pointer Detection

Suresh D S<sup>1</sup>, Bhavana I V<sup>2</sup>

<sup>1</sup>Professor & Head, Department of ECE, Channabasaveshwara Institute of Technology,  
Gubbi, Tumkur, Karnataka, India.

director@cittumkur.org

<sup>2</sup>PG Scholar, Department of ECE, Channabasaveshwara Institute of Technology,  
Gubbi, Tumkur, Karnataka, India.

ivar.bhavana6@gmail.com

---

**Abstract:** *Many modules have been developed to help the physical world interact with the digital world. Here we present a novel approach for Human Computer Interaction (HCI) where, we control cursor movement using a real-time camera and color pointers. Our method is to use a camera and computer vision technology, such as image segmentation, background subtraction and color tracking, to control mouse tasks (left clicking, right clicking, double-clicking and scrolling actions) and we show how it can perform everything as current mouse devices can. A color pointer has been used for the object recognition and tracking, so as to implement the module without any physical contact with the system. Click events of the mouse have been achieved by detecting the number of pointers on the images.*

*The application has been created on MATLAB environment with operating system as windows 7. This method mainly focuses on the use of a Web Camera to develop a virtual human computer interaction device in a cost effective manner.*

**Keywords:** *Human Computer Interaction, Background Subtraction, Color Detection, Web Camera, Computer Vision.*

---

### 1. INTRODUCTION

As the computer technology is grow up, the importance of human computer interaction is rapidly increasing. Most of the mobile devices and laptops are using touch screen technology. But this technology is still not cheap enough to be used on desktop systems. Creating a virtual human computer interactive module such as mouse or keyboard, can be an alternative way for the touch screen. The motivation is to create an object tracking application to interact with the computer, and develop a virtual human computer interaction device.

A virtual mouse is software that allows users to give mouse inputs to a system without using an actual mouse. To the extreme it can also be called as hardware because it uses an ordinary web camera. A virtual mouse can usually be operated with multiple input devices, which may include an actual mouse or a computer keyboard. Virtual mouse which uses web camera works with the help of different image processing techniques.

A color pointer has been used for the object recognition and tracking. Left and the right click events of the mouse have been achieved by detecting the number of pointers on the images. The hand movements of a user are mapped into mouse inputs. A web camera is set to take images continuously. The user must have a particular colour in his hand so that when the web camera takes image it must be visible in the acquired image. This color is detected from the image pixel and the pixel position is mapped into mouse

input. Depending upon the size of the image taken by camera, various scaling techniques are used because the pixel position in the image will not have a correspondence with screen resolution.

In this paper, the mouse cursor movement and click events are controlled using a camera based on color detection technique. Here real time video has been captured using a Web- Camera. The user wears colored tapes to provide information to the system. Individual frames of the video are separately processed. The processing techniques involve an image subtraction algorithm to detect colors. Once the colors are detected, the system performs various operations to track the cursor and performs control actions. No additional hardware is required by the system other than the standard webcam which is provided in every laptop computer.

### 2. OBJECTIVE

The great challenge lies in developing an economically feasible and hardware independent system so that humans can interact without having any physical connection with the computer.

The main objective is to develop an object tracking application to interact with the computer, and a virtual human computer interaction device, where in no physical contact with the device is required and also to provide an easier human-machine interaction routine.

### 3. DIGITAL IMAGE PROCESSING

Digital images are electronic snapshots taken of a scene or scanned from documents, such as photographs, manuscripts, printed texts, and artwork. The digital image is sampled and mapped as a grid of dots or picture elements (pixels). Each pixel is assigned a tonal value (black, white, shades of gray or color), which is represented in binary code (zeros and ones). The binary digits ("bits") for each pixel are stored in a sequence by a computer and often reduced to a mathematical representation (compressed). The bits are then interpreted and read by the computer to produce an analog version for display or printing.

#### 3.1 Image Processing

Image processing is any form of signal processing for which the input is an image, such as photographs or frames of video; the output of image processing can be either an image or a set of characteristics or parameters related to the image. Most image-processing techniques involve treating the image as a two-dimensional signal and applying standard signal processing techniques to it. Image processing usually refers to digital image processing, but optical and analog image processing are also possible. Here image processing is in the sense, splitting each pixel of the image into RGB components.

#### 3.2 Image Analysis

Image analysis is the extraction of meaningful information from images; mainly from digital images by means of digital image processing techniques. Image analysis tasks can be as simple as reading bar coded tags or as sophisticated as identifying a person from their face. In this project we do various image analyzing techniques. The main thing done is the color detection.

At first we receive an image from the web cam. Then each pixel is retrieved from the image and extracts the red, green and blue values (RGB) from each pixel. Now we can easily detect a particular color since all the colors are combinations of RGB values. Here we just to try to detect only Red Green and Blue colors. This is done by traversing through the image, retrieving each pixel, extracting RGB values and then comparing the color values to the RGB values.

Digital image processing is the use of computer algorithms to perform image processing on digital images. As a subcategory or field of digital signal processing, digital image processing has many advantages over analog image processing. It allows a much wider range of algorithms to be applied to the input data and can avoid problems such as the build-up of noise and signal distortion during processing. Since images are defined over two dimensions digital image processing may be modeled in the form of multidimensional system.

### 4. SYSTEM ANALYSIS

System analysis or study is an important phase of any system development process. The system is studied to the minute detail and analysed. The system analyst dwelled deep into the working of the present system. The system was viewed as a whole and the input of the system are identified. During analysis

phase for each problem identified many alternative solutions were evaluated and selected the most feasible one. A feasibility analysis was performed to evaluate possible solutions to recommend the most feasible one.

#### 4.1 Purpose

The purpose of the project is to create a virtual mouse that works with the help of a web camera. In this project a camera continuously takes images of hand movements of a user, which is then mapped into mouse inputs. This means that we can give inputs to computer without having any physical connection with the computer and without having any hardware movements.

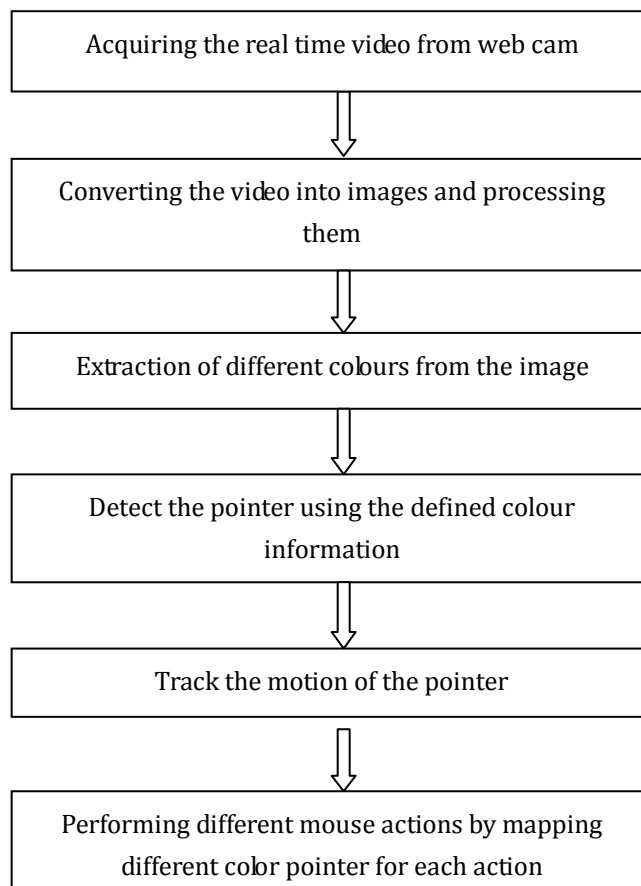
#### 4.2 Problem Definition

To develop a software solution to a problem, the first step is to understand the problem. The problem here is to develop a way so that humans can interact with a computer without having any physical connection with the computer. Many ideas were put forward but they all required physical movement of hardware. Another idea put forward was to use the principle of photoelectric effect. But for that a special hardware is needed and it is not economically feasible. So the final decision is to develop a virtual mouse which uses simple and cheap image processing techniques.

### 5. METHODOLOGY

The implementation has been divided into various steps and each step has been explained below. The system flow explains the overview of the steps involved in the implementation of virtual mouse.

#### 5.1 System Flow



#### 5.2 System Approach

- Capturing real time video using Web-Camera.
- Processing the individual image frame.
- Flipping of each image frame.
- Conversion of each frame to a grey scale image.
- Color detection and extraction of the different colors (RGB) from flipped gray scale image.

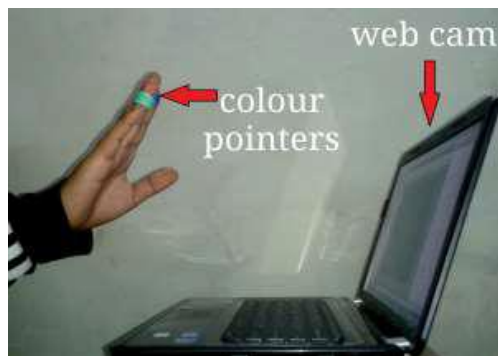
## Virtual Mouse Implementation using Color Pointer Detection

- Conversion of the detected image into a binary image.
- Finding the region of the image and calculating its centroid.
- Tracking the mouse pointer using the coordinates obtained from the centroid.
- Simulating the left click and the right click events of the mouse by assigning different color pointers.

### 5.2.1 Capturing the Real Time Video

For the system to work we need a sensor to detect the hand movements of the user. The webcam of the computer is used as a sensor. The webcam captures the real time video at a fixed frame rate and resolution which is determined by the hardware of the camera. The frame rate and resolution can be changed in the system if required.

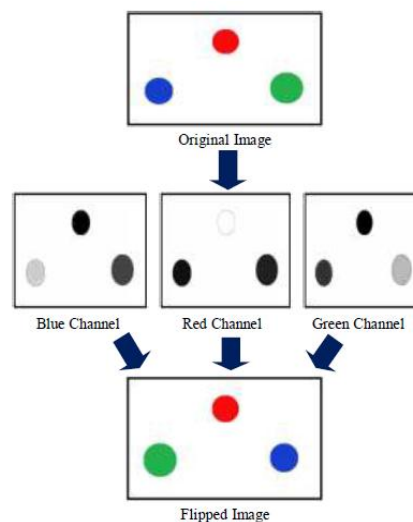
- Computer Webcam is used to capture the Real Time Video
- Video is divided into Image frames based on the FPS (Frames per second) of the camera



**FIGURE 1.** Capturing the Video

### 5.2.2 Flipping of Images

When the camera captures an image, it is inverted. This means that if we move the color pointer towards the left, the image of the pointer moves towards the right and vice-versa. It's similar to an image obtained when we stand in front of a mirror (Left is detected as right and right is detected as left). To avoid this problem we need to vertically flip the image. The image captured is an RGB image and flipping actions cannot be directly performed on it. So the individual color channels of the image are separated and then they are flipped individually. After flipping the red, blue and green colored channels individually, they are concatenated and a flipped RGB image is obtained.



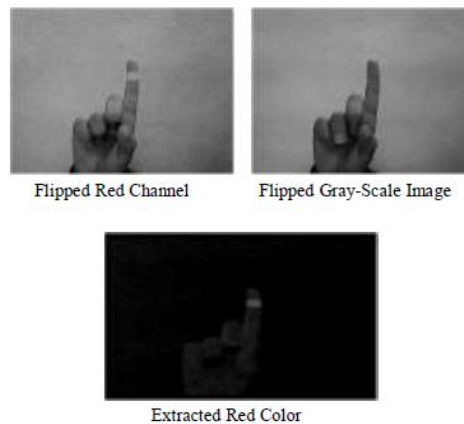
**Figure 2.** Flipped Images

### 5.2.3 Conversion of Flipped Image into Gray scale Image

As compared to a colored image, computational complexity is reduced in a gray scale image. Thus the flipped image is converted into a gray scale image. All the necessary operations were performed after converting the image into gray scale.

### 5.2.4 Colour Detection

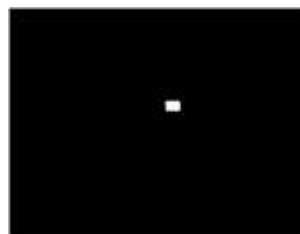
This is the most important step in the whole process. The red, green and blue color object is detected by subtracting the flipped color suppressed channel from the flipped Gray-Scale Image. This creates an image which contains the detected object as a patch of grey surrounded by black space.



**Figure 3.** Color Detection

### 5.2.5 Conversion of Gray scale Image into Binary scale Image

The grey region of the image obtained after subtraction needs to be converted to a binary image for finding the region of the detected object. A grayscale image consists of a matrix containing the values of each pixel. The pixel values lay between the ranges 0 to 255 where 0 represents pure black and 255 represents pure white color. We use a threshold value to convert the image to a binary image. This means that all the pixel values lying below threshold value is converted to pure black that is 0 and the rest is converted to white that is Thus the resultant image obtained is a monochromatic image consisting of only black and white colors. The conversion to binary is required because MATLAB can only find the properties of a monochromatic image.

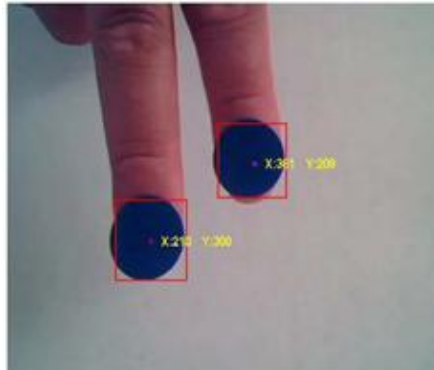


**Figure 4.** Detected region

### 5.2.6 Finding Centroid of an object and plotting Bounding Box

For the user to control the mouse pointer it is necessary to determine a point whose coordinates can be sent to the cursor. With these coordinates, the system can control the cursor movement. An inbuilt function in MATLAB is used to find the centroid of the detected region. The output of function is a matrix consisting of the X (horizontal) and Y (vertical) coordinates of the centroid. These coordinates change with time as the object moves across the screen.

- Centroid of the image is detected and a bounding box is drawn around it.
- Its co-ordinates are located and stored in a variable



**Figure 5.** Bounding box drawn for the detected color pointers.

### 5.2.7 Tracking the Mouse pointer

Once the coordinates has been determined, the mouse driver is accessed and the coordinates are sent to the cursor. With these coordinates, the cursor places itself in the required position. It is assumed that the object moves continuously, each time a new centroid is determined and for each frame the cursor obtains a new position, thus creating an effect of tracking. So as the user moves his hands across the field of view of the camera, the mouse moves proportionally across the screen.

There is no inbuilt function in MATLAB which can directly access the mouse drivers of the computer. But MATLAB code supports integration with other languages like C, C++, and JAVA. Since java is a machine independent language so it is preferred over the others. A java object is created and it is linked with the mouse drivers. Based on the detection of other colors along with red the system performs the clicking events of the mouse. These color codes can be customized based on the requirements.

### 5.2.8 Performing Clicking Actions

The control actions of the mouse are performed by controlling the flags associated with the mouse buttons. JAVA is used to access these flags. The user has to perform hand gestures in order to create the control actions. Due to the use of color pointers, the computation time required is reduced. Furthermore the system becomes resistant to background noise and low illumination conditions. The detection of green and blue colors follows the same procedure discussed above.

Clicking action is based on simultaneous detection of two colors.

- If Red along with single Blue color is detected, Left clicking action is performed.
- If Red along with double Blue color is detected, Right clicking action is performed.

## 5.3 Problems and Drawbacks

Since the system is based on image capture through a webcam, it is dependent on illumination to a certain extent. Furthermore the presence of other colored objects in the background might cause the system to give an erroneous response. Although by configuring the threshold values and other parameters of the system this problem can be reduced but still it is advised that the operating background be light and no bright colored objects be present. The system might run slower on certain computers with low computational capabilities because it involves a lot of complex calculations in a very small amount of time. However a standard pc or laptop has the required computational power for optimum performance of the system. Another fact is that if the resolution of the camera is too high then

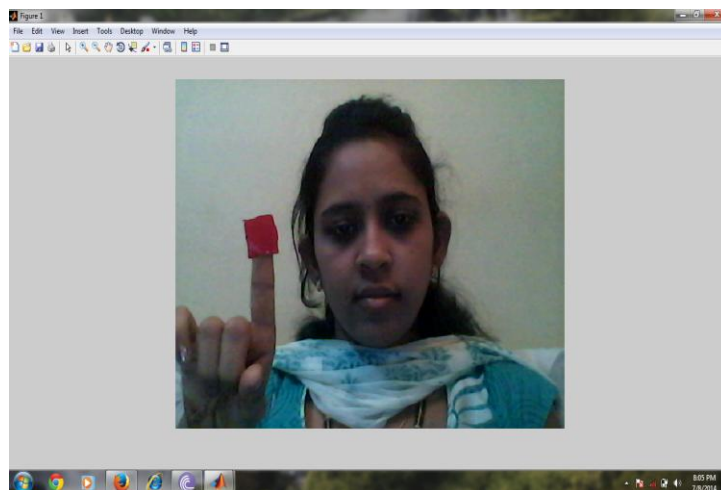
the system might run slow. However this problem can be solved by reducing the resolution of the image by making changes in the system.

## 6. EXPERIMENTAL RESULTS

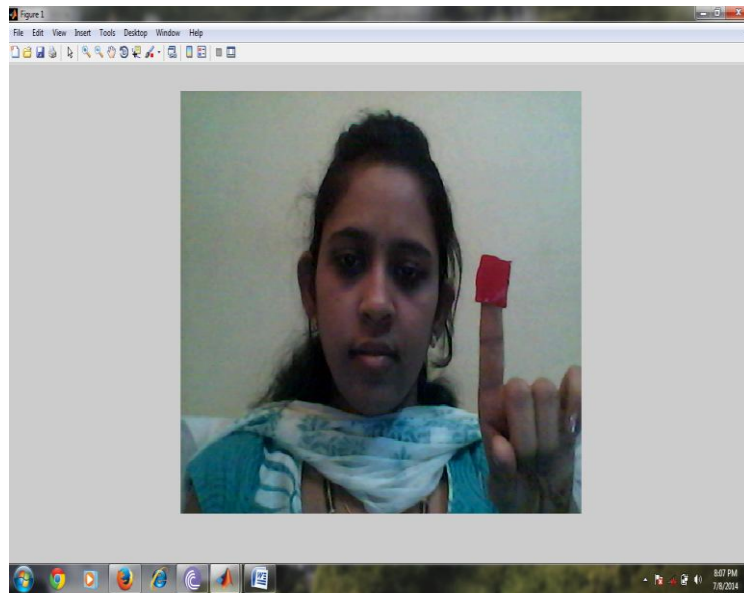
The necessary steps to be followed are as follows:

- Firstly, we need to interface the camera using MATLAB.
- Start the camera to capture the images continuously.
- Capturing the real time video using the started up camera.
- Processing the individual image frame obtained from the video.
- Flipping of each image frame.
- Conversion of each frame to a GRAY-SCALE image.
- Color detection and extraction of different colors.
- Conversion of each frame into BINARY-SCALE image.
- Find the center and draw the bounding box.
- Track mouse pointer using coordinates of centroid.
- Different mouse actions are performed based on the steps below.
  - a. First, we need to detect red color in the image.
  - b. If Red color pointer is detected, we need to track the mouse pointer.
  - c. If a single Blue color pointer is detected, we need to perform Left click.
  - d. If two Blue pointers are detected, we need to perform Right clicking action.
  - e. If three Blue pointers are detected, we need to perform Double clicking action.
  - f. If none of the color has been detected again we need to try tracking different colors from the image.

**Figure 6.** Example Demonstrating Flipping of Images (a) Acquiring single frame and (b) Flipping of the acquired frame.



(a)



(b)

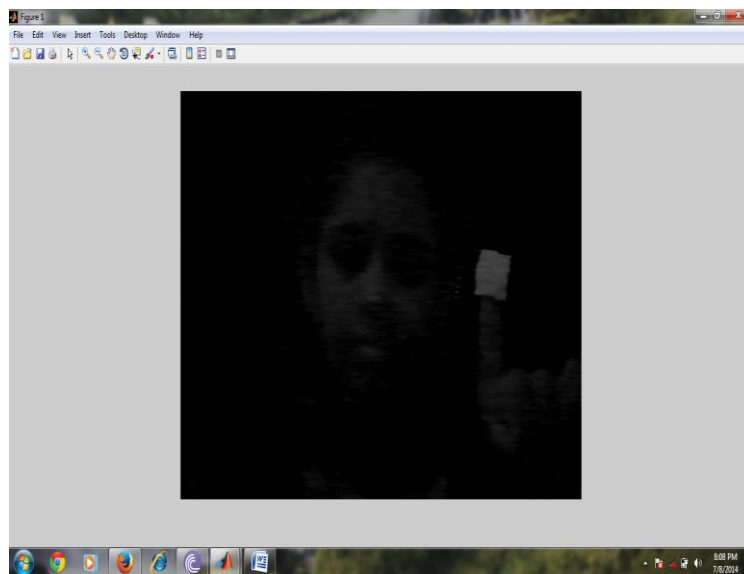


Figure 7. Extraction of Red Colour from the Image using Frame Difference Method

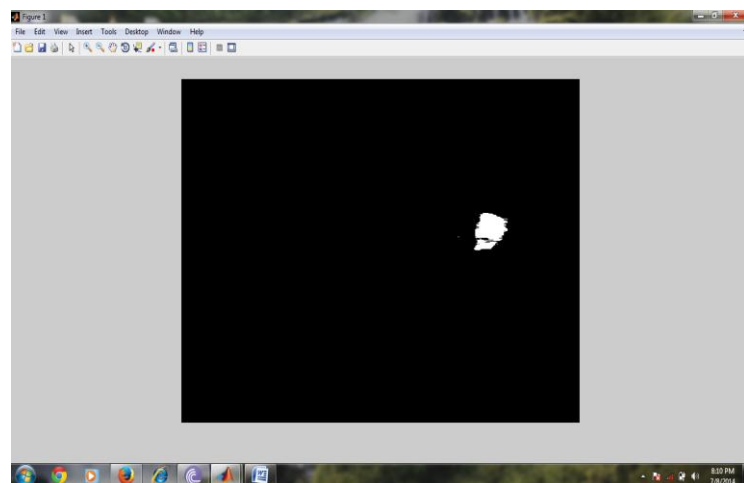
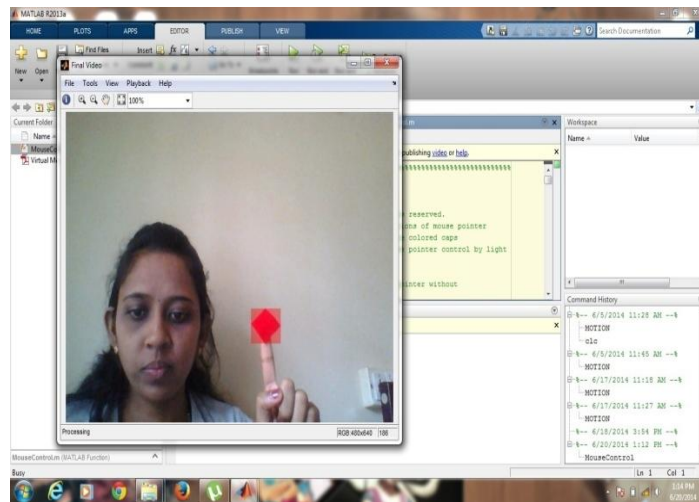
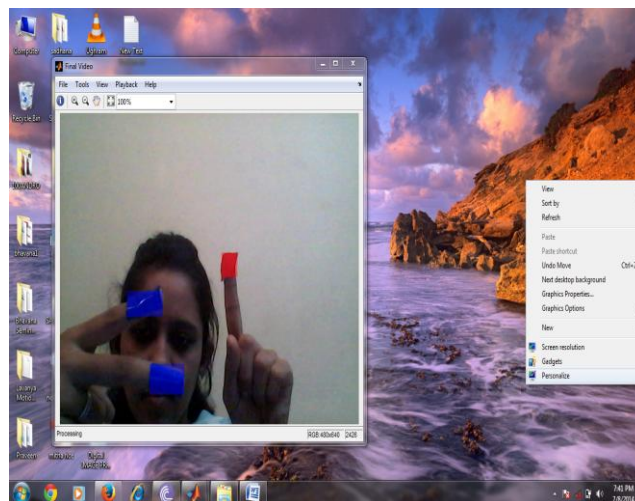


Figure 8. Convert the Image into Binary Image with the Red Objects as White.

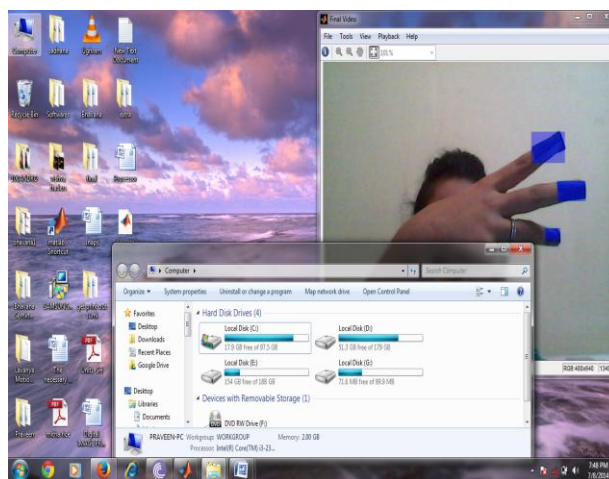




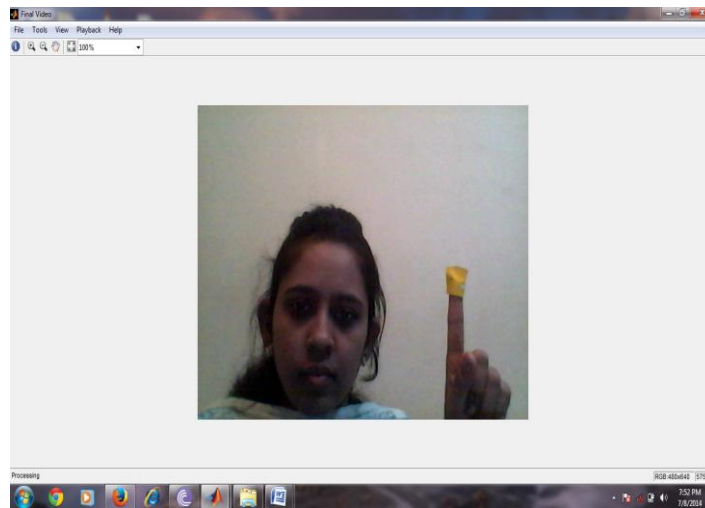
**Figure 9.** A Red color is detected and a Bounding Box is drawn around it. Red pointer movement is mapped with mouse movement.



**Figure 10.** Right Click Event Explanation. A Red along with Two Blue pointers are detected. Bounding box will be drawn for both the detected color pointers. Two Blue pointers indicate Right click which pop up the window as shown in the fig above.



**Figure 11.** Double Clicking Event Explanation. Three Blue Pointers are detected which indicates Double Click. Bounding box will be drawn for all the detected color pointers. Three Blue Pointers indicates right click which pop up the window as shown in the fig above.



**Figure 12.** *No Actions being Performed for the Undefined Colour Pointer. Since we have not defined yellow color, bounding box will be not created. No actions will be performed.*

### REFERENCES

- [1] A. Erdem, E. Yardimci, Y. Atalay, V. Cetin, A. E. “Computer vision based mouse”, Acoustics, Speech, and Signal Processing, 2002. Proceedings. (ICASS). IEEE International Conference
- [2] Chu-Feng Lien, “Portable Vision-Based HCI – A Real-time Hand Mouse System on Handheld Devices”, National Taiwan University, Computer Science and Information Engineering Department
- [3] Hojoon Park, “A Method For Controlling The MouseMovement using a Real Time Camera”, 2008, Brown University, Providence ,RI ,USA, Department of computer science
- [4] <http://www.mathworks.com/matlabcentral/fileexchange/28757-tracking-red-color-objects-using-matlab>
- [5] <http://www.mathworks.com/help/techdoc>
- [6] <http://www.mathworks.com/support/solutions/en/data/1-2X10AT/index.html?solution=1-2X10AT>
- [7] Chris Solomon and Toby Breckon, “Fundamentals Of Digital Image Processing”, Third edition, John Wiley & Sons, Ltd,2011
- [8] Kermit Sigmon, “MATLAB Primer”, Third Edition,1990.