

## Virtual Mouse Implementation Using Color Pointer Detection

Abdul Khaliq<sup>1</sup>, A. Shahid Khan<sup>2</sup>

<sup>1</sup>M.Tech Scholar, ECE Dept., AFSET, Faridabad

<sup>2</sup>Assistant Professor, ECE Dept., AFSET, Faridabad  
abdulkhaliq035@gmail.com

**Abstract:** *In this paper we present an approach for Human computer Interaction (HCI), where we have tried to control the mouse cursor movement and click events of the mouse using hand gestures. Hand gestures were acquired using a camera based on color detection technique. 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, Colour Detection, Web Camera, Computer Vision.*

### I. 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 color 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.

### II. RELATED WORK

“Many researchers in the human computer interaction and robotics fields have tried to control mouse movement using video devices. However, all of them used different methods to make a clicking event. One approach, by Erdem et al, used fingertip tracking to control the motion of the mouse. A click of the mouse button was implemented by defining a screen such that a click occurred when a users hand passed over the region [1]. Another approach was developed by Chu-Feng Lien [2]. He used only the finger-tips to control the mouse cursor and click. His clicking method was based on image density, and required the user to hold the mouse cursor on the desired spot for a short period of time. Paul et al, used still another method to click. They

used the motion of the thumb (from a thumbs-up position to a fist) to mark a clicking event thumb. Movement of the hand while making a special hand sign moved the mouse pointer [3].” In this study, 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 image.

### III. INTRODUCTION TO TOOLS

*1. Introduction to MATLAB:* Matlab is a program that allows you to carry out computations in a straight forward manner, removing much of the tedium involved in programming. It is extremely useful for creating simulations of neural networks, as well as for general types of data analysis and visualization.

*2. Image Processing with MATLAB:* Images are everywhere, from everyday devices like cameras and smart phones to specialized devices for medical imaging, automotive safety, industrial automation, and more. Each of these uses for image processing has unique challenges.

MATLAB and Image Processing Toolbox provide a flexible environment to explore design ideas and create unique solutions for MATLAB toolbox used in our project is as follows:

- Image acquisition Toolbox
- Image processing toolbox

a) *Image Acquisition Toolbox*: The Image Acquisition Toolbox is a collection of functions that extend the capability of the MATLAB numeric computing environment. The toolbox supports a wide range of image acquisition operations, including. Acquiring images through many types of image acquisition devices, from professional grade frame grabbers to USB- based Webcam.

b) *Image Processing Toolbox*: Image Processing Toolbox provides a comprehensive set of reference-standard algorithms, functions and apps for image processing, analysis, visualization and algorithm development. You can perform image enhancement, noise reduction, image segmentation, geometric transformations and image registration

3. *Web Camera*: This Webcam is used to capture and recognizes an object in view and tracks the user's hand gestures using computer-vision based techniques. It sends the data to the computer. The camera, in a sense, acts as a digital eye, seeing what the user sees. It also tracks the movements of the thumbs and index fingers of both of the user's hands. The camera recognizes objects around you instantly.



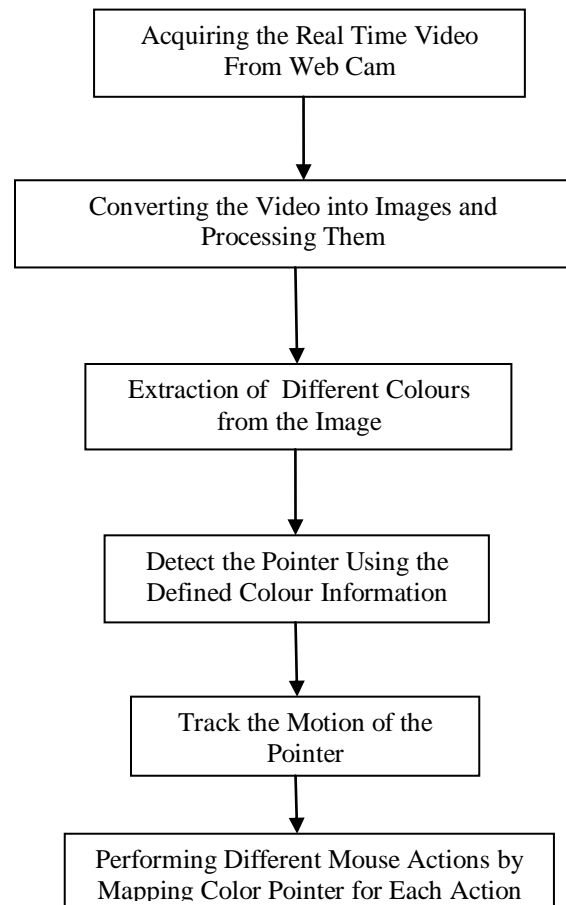
#### IV. INTRODUCTION TO THE SYSTEM

In our work, we have tried to control mouse cursor movement and click events 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, the details of which are provided below. No additional hardware is required by

the system other than the standard webcam which is provided in every laptop computer.

#### 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.
- 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.



*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.

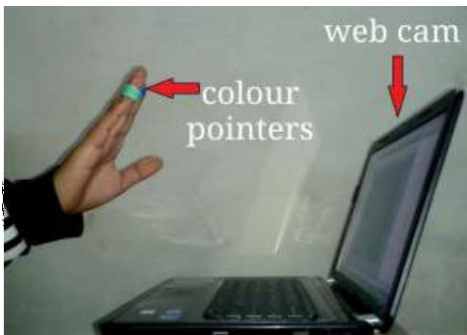


Figure 1 Capturing the Video

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

*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.\

*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.

*Color 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.

*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.

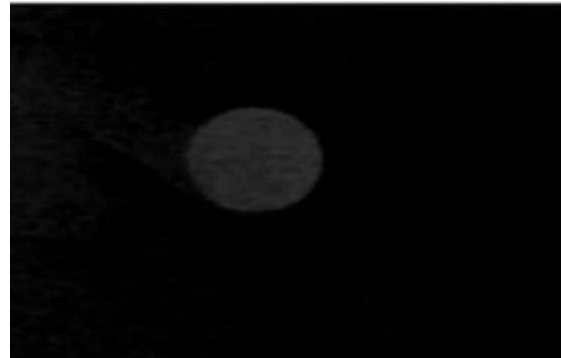


Fig. Gray Image

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.

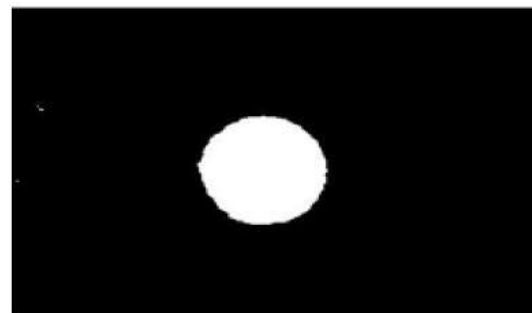


Fig. Binary Image

*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 .

*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.

*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 colour 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.

#### V. 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.

#### VI. CONCLUSION

In this paper, an object tracking based virtual mouse application has been developed and implemented using a webcam. The system has been implemented in MATLAB environment using MATLAB Image Processing Toolbox. This technology has wide applications in the fields of augmented reality, computer graphics, computer gaming, prosthetics, and biomedical instrumentation. Furthermore a similar technology can be applied to create applications like a digital canvas which is gaining popularity among artists. This technology can be used to help patients who don't have control of their limbs. In case of computer graphics and gaming this technology has been applied in

modern gaming consoles to create interactive games where a person's motions are tracked and interpreted as commands. Most of the applications require additional hardware which is often very costly. Our motive was to create this technology in the cheapest possible way and also to create it under a standardized operating system. Various application programs can be written exclusively for this technology to create a wide range of applications with the minimum requirement of resources

#### VII. 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 Mouse Movement 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/12X10AT/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.