Web Camera Based Mouse Controlling Using RGB Colours

R. Lulu1, Vanshri Kurhade2, Neha Pande3, Rohini Gaidhane4 , Nikita Khangar5, Priya Chawhan6,

Ashwini Khaparde7

Dept. of Electronics and telecommunication Engg.

SRMCEW Nagpur

1,Asst. Professor, 2-7,Schlor

***ABSTRACT**:- The paper deals with the enhancement of human interaction with the digital world. Miniaturization of computing devices allows us to be in continuous touch with the digital world. Restriction of information on traditional platforms like paper, digital screen etc is overcome with the help of this technology. Dependency on traditional hardware input devices like keyboard, mouse etc will be reduced considerably, thereby allowing portability. It makes use of hand movements/gestures to feed input to a computer or any other digital device.The Virtual Keyboard is an adaptation of the 6th sense technology developed by Pranav Mistry,PhD student at MIT media labs.It works on a similar principle of hand movement/gesture control to perform simple functions like Right/Left click, Scroll control etc, which would otherwise have to be performed with the help of a mouse or a keyboard.Image Processing technique is an important basis for the implementation of this technology.The key features of this technology include media player volume control, power point slide control, camera control, scrolling,mouse control, initiation and termination of call. The paper deals with the enhancement of human interaction with the digital worldcomputing devices allows us to be in continuous touch with the digital world.*

The code will be develop in MATLAB. Recognition and pose estimation in this system are user independent and robust as we will be using colour tape(Red,Green and blue) on our finger to perform actions. The software can be used as an intuitive input interface to application that required multi-dimensional control e.g. computer games.

KEYWORDS** :- **Hand gesture, Human Computer Interaction, Colour Detection, Web camera, Background Subtraction, colou strips

I. INTRODUCTION

Since the computer technology continues to grow up, the importance of human computer interaction is enormously increasing. Nowadays most of the mobile devices are using a touch screen technology. However, this technology is still not cheap enough to be used in desktop systems. Creating a virtual human computer interaction device such as mouse or keyboard using a webcam and computer vision techniques can be an alternative way for the touch screen. In this study, finger tracking based a virtual mouse application has been designed and implemented using a regular webcam. The motivation was to create an object tracking application to interact with the computer, and develop a virtual human computer interaction device.

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 user’s 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 straightforward 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

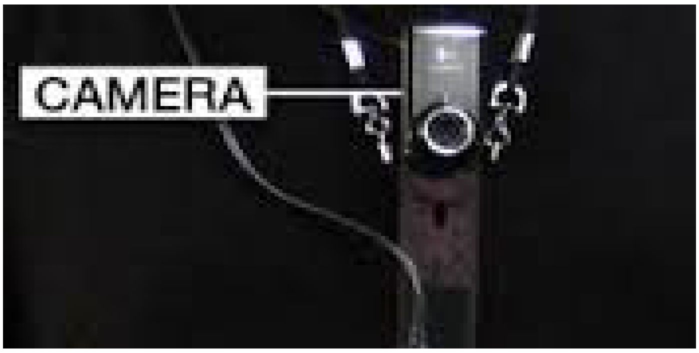
Images are everywhere, from everyday devices like cameras and smart phones to specialized devices for medical imaging, automotive safety, industrial automation, and more. Each of these uses for image processing has unique challenges. MATLAB and Image Processing Toolbox provide a flexible environment to explore design ideas and create unique solutions for Matlab toolbox used in our project are 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. WEBCAMERA



This Webcam is used to capture and recognises 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.

4. COLOUR STICKERS :-

colour stickers which are at the tip of the user's fingers. Marking the user's fingers with red, green, and blue tape helps the webcam recognize gestures. The movements and arrangements of these Markers are interpreted into gestures that act as interaction instructions for the projected application interface.

IV. INTRODUCTION TO THE SYSTEM

In our work, we have tried to control mouse cursor movement and click events using a camera based on colour detection technique. Here real time video has been captured using a Web- Camera. The user wears coloured 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 colours. Once the colours 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.

Following are the steps in our approach:

(i) Capturing real time video using Web-Camera.

(ii) Processing the individual image frame.

(iii) Flipping of each image frame.

(iv) Conversion of each frame to a grey scale image.

(v) Colour detection and extraction of the different colours (RGB) from flipped gray scale image

(vi) Conversion of the detected image into a binary image.

(vii) Finding the region of the image and calculating its centroid.

(viii) Tracking the mouse pointer using the coordinates obtained from the centroid.

(ix) Simulating the left click and the right click events of the mouse by assigning different colour pointers.

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.

2. FLIPPING OF IMAGES

When the camera captures an image, it is inverted. This means that if we move the colour 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 colour channels of the image are separated and then they are flipped individually. After flipping the red, blue and green coloured channels individually, they are concatenated and a flipped RGB image is obtained.

FLOWCHART:-

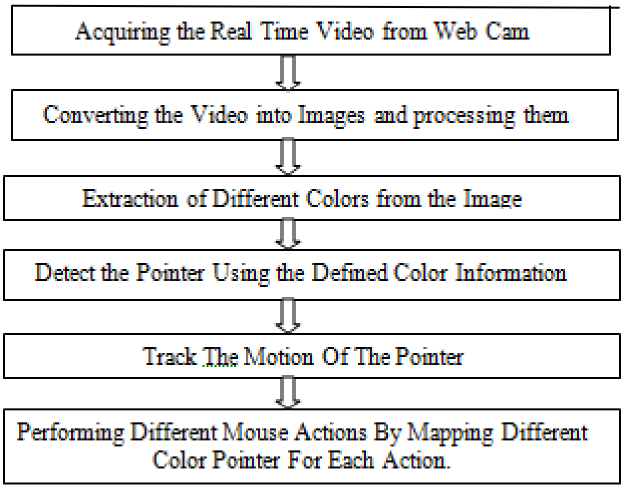

Acquiring the Real Time Video from Web Cam
Converting the Video into Images and processing them
Extraction of Different Colors from the Image
Detect the Pointer Using the Defined Color Information
Track The Motion Of The Pointer
Performing Different Mouse Actions By Mapping Different Color Pointer For Each Action.

Fig. BASIC FLOW CHART

3. CONVERSION OF FLIPPED IMAGE INTO GRAY SCALE IMAGE

As compared to a coloured 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.

4. COLOUR DETECTION

This is the most important step in the whole process. The red, green and blue colour 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.

5. CONVERSION OF GRAY SCALE IMAGE INTO BINARY SCALE IMAGE

To convert the gray scale image to binary scale image MATLAB's built in "*im2bw"* function has been used. Function can be used as;

BW = *im2bw*(I, level)

Where; it converts the grayscale image I to a binary image. The output image BW replaces all pixels in the input image with luminance greater than level with the value 1 (white) and replaces all other pixels with the value 0 (black). Specify level in the range [0,1]. This range is relative to the signal levels possible for the image's class. Therefore, a level value of 0.5 is midway between black and white, regardless of class.

In our study, the threshold 0.15 gave the best result for the large range of illumination change.

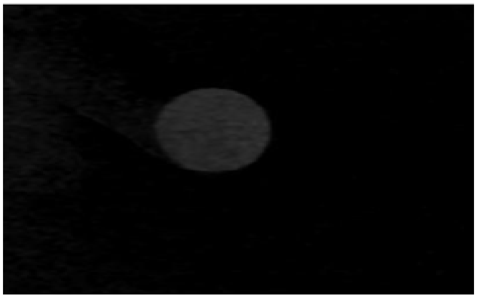


Fig. Gray Image

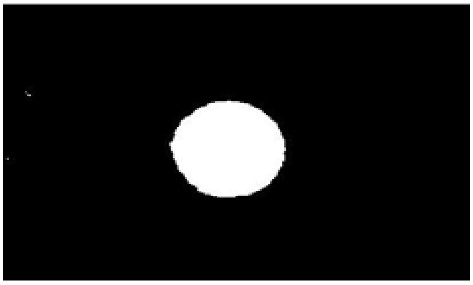


Fig. Binary Image

6. FINDING CENTROID OF AN OBJECT AND PLOTTING

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.

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.

8. PERFORMING CLICKING ACTIONS

Clicking action is based on simultaneous detection of colours.

- If red colour is detected then cursor will move.
- If red along with blue colour is detected, Left clicking action is performed
- If Red along with two stips of Blue colour is detected, Right clicking action is performed

V. PROBLEMS AND DRAWBACK

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.

VI. CONCLUSION

We developed a system to control the mouse cursor using a real-time camera. We implemented all mousetasks such as left and right clicking, double clicking, and scrolling. This system is based on computer visionalgorithms and can do all mouse tasks. This
technology has wide applications in the fields of augmented reality, computer graphics, computer gaming, prosthetics, and biomedical instrumentation. Our motive was to create this technology in the cheapest possible way and also to create it under a standardized operating system.

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] Abhik Banerjee, Abhirup Ghosh, “Mouse Control Using a Web Camera Based On Color Detection” ,International journal of computer trends and technology (UCTT) -volume 9 ,1-mar 2014

[5]Aakansha yeole , S. Chitra Sriraman " Virtual keyboard", International journal of Science, Engineering and Technology Research(IJSETR), Volume 3, issue 3, March 2014

[6] Rachit Puri, “Gesture Recognition Based On Mouse Events”,International journal of computer science (IJCSIT)-vol 5,6-Dec.2013

[7] http://www.mathswork.com/matlabcentral/fileexchange /28757-tracking-red-color-objects-using- matlab

[8] The MATLAB WEBSITE[Online]. Available http://www.mathworks.com/matlabcentral/fileexchange/2 8757-tracking-red-color-objects-using-matlab