Hand Gesture Recognition System for Human-Computer Interaction

S. Sakthivel, A. Anbarasi
Department of Electronics and Communication Engineering,
Anna University, Chennai, India

ABSTRACT

The aim of this project is to moving mouse cursor on the screen without using any hardware which is used very often now-a-days i.e. mouse. We use the newly born technology named Sixth Sense Technology for this purpose. We implement computer mouse movement through finger by image grabbing using Sixth Sense Technology which gets processed in MATLAB. By applying vision technology and controlling the mouse by natural hand gestures and colour based sensing, we simulated the mouse system. This mouse system can control all mouse tasks, such as clicking (right and left), double clicking, dragging and dropping.

KEYWORDS: *Camera, Control computer, MATlab, Simulink.*

I. INTRODUCTION

Nowadays 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.

“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. Another approach was developed by Chu-Feng Lien. 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 time.

In this study, a colour 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.Video cameras and computer vision techniques may be used to capture many details of human shape and movement. The shape of the hand may be analyzed over time to manipulate an onscreen object in a way analogous to the hand’s manipulation of paper on a desk. Such an approach may lead to a faster, more natural, and more fluid style of interaction for certain tasks.

“Hand Gesture Recognition System for Human - Computer Interaction” is an attempt in ubiquitous computing. Here, we will be using three colour tapes on our fingers. One of the tapes will be used for controlling cursor movement while the other coloured tapes will be used for click, drag and drop events of the mouse. Thus, the system will provide a new experience for users in interacting with the computer.

II. TERMINOLOGY

2.1. Binary Image

A binary image is an image that has only two possible values for each pixel. It is also familiar as bi-level or two levels. The names black and white, B &W. Monochrome or monochromatic are used for this concept but this can also be used for grayscale image or one that have only one sample per pixel. In digital image processing, binary images often result from certain operations like “segmentations”, “thresholding” etc. It is formed by thresholding a gray scale or colour image. The interpretation of the pixel’s binary values differs from device to device, 0 is interpreted as black and 1 is interpreted as white.

2.2. Gray scale Image

A “gray scale or greyscale” digital image is an image in which the value of every pixel is a single sample . In a gray scale image, red green and blue all have equal intensities in RGB. Thus, a single intensity value is used for every pixel. Image of this sort are typically composed of shades of gray varying from black at the weakest intensity to white at the strongest & with many shades of gray in between. Gray scale image are typically stored with & bits per sampled pixel which allows 256, (2^8=256) intensities. i.e. for 8 bit grey scale image, there 256 are possible intensities.

2.3. Conversion of colour image to gray scale image

The conversion of a colour image to gray scale image is possible by first obtaining value of its red, green and blue (RGB) primaries. Then it is necessary to add 30% red value + 59% green value + 11% blue. These percentages are chosen irrespective of the scales (0.0 - 1.0, 0 to 255, 0% to 100% etc).
The resultant level is the desired gray value. If we note the percentages it is clear that the percentage is higher for green and lower for blue. It is chosen in such a way due to the different relative sensitivity of normal human eye.

2.4. Luminance

Luminance describes the amount of light that passes through or is emitted from a particular area and it falls within a given solid. It is the photometric measure of the density of luminous intensity is a given direction. Luminance is the “emission” or “reflection” from flat surfaces. When looking at a surface from particular angle of view, the amount of luminous power perceived by the eye will be indicated by the luminance. Thus the brightness of the surface is indicated by the luminance.
Luminance is defined by

Lv = d2F / dAd;Cos< (4.1)

Where,
Lv – Luminance (cd/m2)
F - Luminous Flux or Luminous Power (lm)
< - Angle between the Surface Normal and the Specified Direction
A - Area of the Surface (m2)
; - Solid Angle (Sr)

Relative luminance follows the photometric definitions of luminance, but with the values normalized to 1 or 100 for a reference white for4 colour space such as XYZ, the litter Y refers to relative luminance for RGB colour spaces however, relative luminance can be calculated from liner RGB components :

Y = 0.2126R + 0.7152G + 0.722 B (4.2)

Here, the co-efficient are all positive having a large green co-efficient and small blue co-efficient. The three forms the middle row of the RGB to XYZ colour transformation matrix.

2.5. Segmentation

Segmentation is the process of portioning a digital image in to multiple images i.e. breaking the image in to sets pixels by doing. So it is more meaningful and easier to analyze the picture. Image segmentation is used to locate objects and boundaries lines, curves etc) in images. A set of regions collectively cover the entire image when the image is segmented. Each of the pixels in the region is similar with respect to three characteristics colour, intensity or texture. However, the adjacent regions are significantly different with respect to the three characteristics. For our research, we have used the “region-growing method”.

2.6. Thresholding

Thresholding is the simplest method of image segmentation. It is a useful process to be able to separate out the regions of the image that corresponds to “objects” in which we are interested to work out from the regions of that image that correspond to background thresholding often provides an easy and convenient way for performing this segmentation on the basis of different intensities or colours in the foreground and background regions of an image. By thresholding it is possible to be able to see the areas of an image that consists of pixels whose value lie within a specified range, or band of intensities.

2.7. Adaptive Thresholding

Adaptive thresholding takes a grayscale or colour image as input and in the simplest implementation. It out puts a binary image representing the segmentation. It is used to separate desirable foreground image objects from background based on the difference in pixel intensities of each region. It does so by selecting an individual threshold for each pixel base on the range of intensity values in its local neighbourhood.

III. COLOUR RECOGNITION

3.1. Colour

To understand the basics of camera vision it is necessary to know how a computer Sees an image. A computer can only work with numbers, so when an image is imported on a computer, the computer sees it as a lot of numbers. For each pixel in the image, the computer uses a code and all those codes together form the total image. For image processing, it is required to understand the way a computer encodes an image. There is more than one way to encode a picture by using different colour spaces. In this chapter different colour spaces are described. Once these codes are analyzed, they can be used for colour definition and when the colours are defined, pixel labelling is used to recognize colours in the picture.

3.2. Colour Space

A colour space describes the range of colours that a camera can see. It is a collection of codes for every colour. Each pixel in an image has a colour that is described in the colour space, so this colour space can be used for pixel labelling. There are different ways to describe all colours, so there are also different colour spaces. In this section, 3 of most well known colour spaces are described. The size of a colour space depends on the number of tones of a main colour.

3.3. RGB-Space

In the RGB colour space, each colour is described as a combination of three main colours, namely Red, Green and Blue. This colour space can be visualized as a 3D matrix with the main colours set out on the axis. The values for the main colours vary from 0 to 1. Each colour is coded with three values, a value for red, blue and green. In this colour space, an imported image on a computer is thus transformed into 3 matrices with values per pixel for the representing main colour.

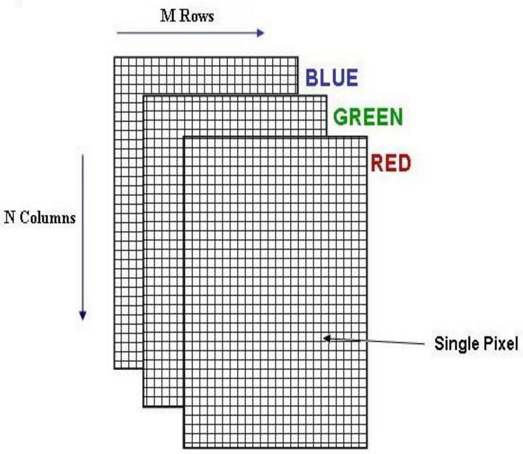


Fig 1 RGB Colour space model

3.4. HSV-Space

Another colour space is the HSV space. This is a cylindrical space and can be modelled as a cone turned upside down. The vertical position defines the brightness (V), the angular position defines the hue (H) and the radial position represents the saturation (S). For every value of V, a circle can be drawn as a cross section of the cone.

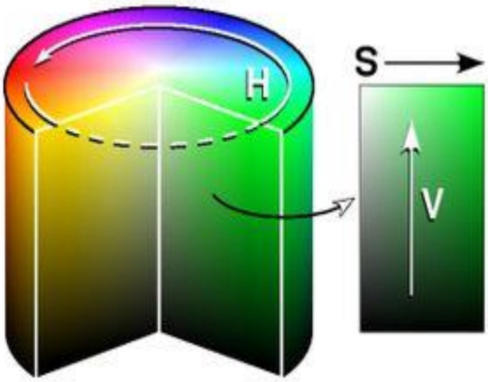


Fig 2 HSV colour space model

The saturation always ranges from 0 to 1 and it specifies the relative position from the vertical axis to the side of the circle. Zero saturation indicates no hue, just gray scale. The hue is what is normally thought of as colour and varies from 0 to 1. Brightness ranges also from 0 to 1, where V=0 indicates black. Just like in the RGB space, a colour in this space is also coded with 3 values. Therefore an image is also transformed into three matrices when imported on a computer.

3.5. LAB Space

The last colour space that will be discussed is the LAB space. This is also a colour space with three parameters. L defines Lightness, A denotes the red/green value and B the yellow / blue value.

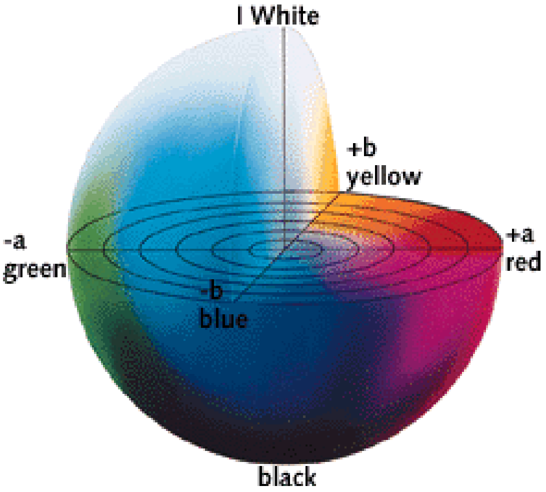


Fig 3 LAB Space model

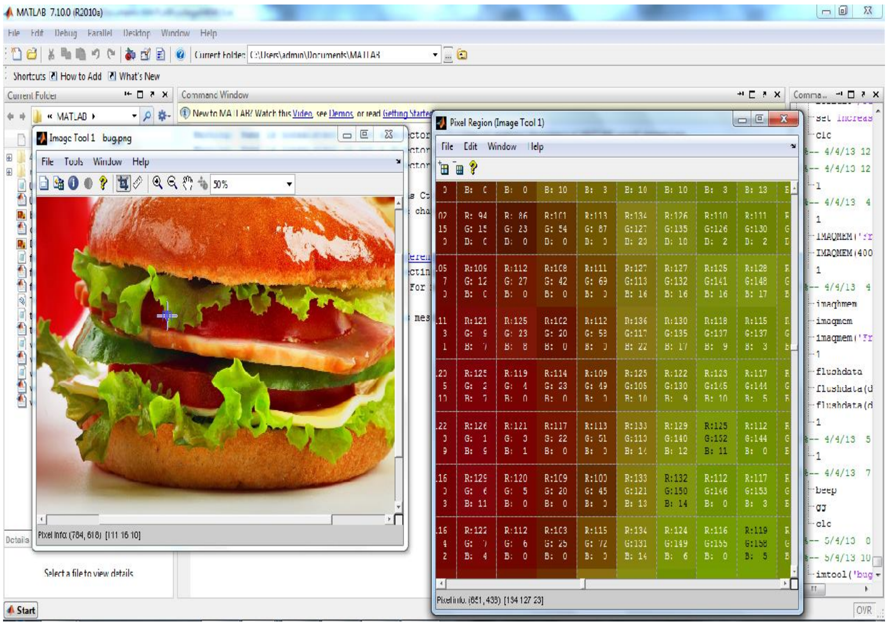
This colour space is not as easy to visualize as the other two spaces, because L is a value between 0 and 100 and for every value of L, another picture with a specific colour range can be drawn. In these pictures the A value is on the horizontal and B the vertical axis.

3.6. Pixel labelling

The colour matrix is now a matrix of size 256x256x256 with all zeros, except for every colour that is associated with the ball, the value on these coordinates is one. This matrix is used for pixel labelling. An image consists of a lot of pixels, depending on the resolution of the camera. Each pixel has its own colour and by means of the colour matrix this colour can be recognized. The process, in which each pixel is given a value depending on its colour, is called pixel labelling.

3.7. Pixel labeling using MATLAB

The standard colour space in Matlab is the RGB-space; an imported image automatically has its RGB-values. Every pixel in the image has its own red, green and blue value and these values define a coordinate in the colour matrix. The pixel will receive the value of the colour matrix at this coordinate. When the value is one, the colour of this pixel can be associated with the coloured tape, and in the new image the pixel will have the value one


MATLAB 7.10.0 (R2010a)
File Edit Debug Parallel Desktop Window Help
Current Folder: C:\Users\admin\Documents\MATLAB
Shortcuts How to Add What's New
Current Folder
Command Window
MATLAB
Image Tool 1 - bug.png
File Tools Window Help
50%
Pixel Region (Image Tool 1)
File Edit Window Help
Details
Select a file to view details
Start
OVR
Comma...

New to MATLAB? Watch this Video, see Demos, or read Getting Started

Pixel info: (784, 618) [111 16 10]

0	B: 0	B: 0	B: 10	B: 3	B: 10	B: 10	B: 3	B: 13
02 15 0	R: 94 G: 15 B: 0	R: 86 G: 23 B: 0	R:101 G: 54 B: 0	R:113 G: 87 B: 0	R:134 G:127 B: 23	R:126 G:135 B: 10	R:110 G:126 B: 2	R:111 G:130 B: 2
05 7 0	R:109 G: 12 B: 0	R:112 G: 27 B: 0	R:108 G: 42 B: 0	R:111 G: 69 B: 0	R:127 G:113 B: 16	R:127 G:132 B: 16	R:125 G:141 B: 16	R:128 G:148 B: 17
11 3 1	R:121 G: 9 B: 7	R:125 G: 23 B: 8	R:102 G: 20 B: 0	R:112 G: 58 B: 0	R:136 G:117 B: 22	R:130 G:135 B: 17	R:118 G:137 B: 9	R:115 G:137 B: 3
20 5 10	R:125 G: 2 B: 7	R:119 G: 4 B: 0	R:114 G: 23 B: 0	R:109 G: 49 B: 0	R:125 G:105 B: 10	R:122 G:130 B: 9	R:123 G:145 B: 10	R:117 G:144 B: 5
22 0 9	R:126 G: 1 B: 9	R:121 G: 0 B: 1	R:117 G: 22 B: 0	R:113 G: 51 B: 0	R:133 G:110 B: 14	R:129 G:140 B: 12	R:125 G:152 B: 11	R:112 G:144 B: 0
16 0 3	R:129 G: 6 B: 11	R:120 G: 5 B: 0	R:109 G: 20 B: 0	R:100 G: 45 B: 0	R:133 G:121 B: 13	R:132 G:150 B: 14	R:112 G:146 B: 0	R:117 G:153 B: 3
16 4 2	R:122 G: 7 B: 4	R:112 G: 6 B: 0	R:103 G: 25 B: 0	R:115 G: 72 B: 0	R:134 G:131 B: 14	R:124 G:149 B: 6	R:116 G:155 B: 0	R:119 G:158 B: 5

Pixel info: (651, 433) [134 127 23]

```
set increas
clc
%-- 4/4/13 12
%-- 4/4/13 12
1
%-- 4/4/13 4
1
IMAQMEM('fr
IMAQMEM(400
1
%-- 4/4/13 4
imaqhmem
imaqmem
imaqmem('fr
1
flushdata
flushdata(d
flushdata(d
1
%-- 4/4/13 5
1
%-- 4/4/13 7
beep
gg
clc
%-- 5/4/13 0
%-- 5/4/13 10
imtool('bug
```

Fig 4 Pixel labelling of an image

IV. METHODOLOGY

Most image-processing techniques involve treating the image as a two-dimensional signal and applying standard signal-processing techniques to it. The video captured by the camera is being processed by the MATLAB program that helps in motion detection and color recognition. The results of this processing can be used in numerous security applications such as intrusion detection, Spy robots, fire alarm, person finder etc.

4.1. Bringing data into the MATlab workspace

The toolbox stores acquired data in a memory buffer, a disk file, or both, depending on the value of the video input object Logging Mode property. To work with this data, you must bring it into the MATLAB workspace. To bring multiple frames into the workspace, use the get data function. Once the data is in the MATLAB workspace, you can manipulate it as you would any other data.

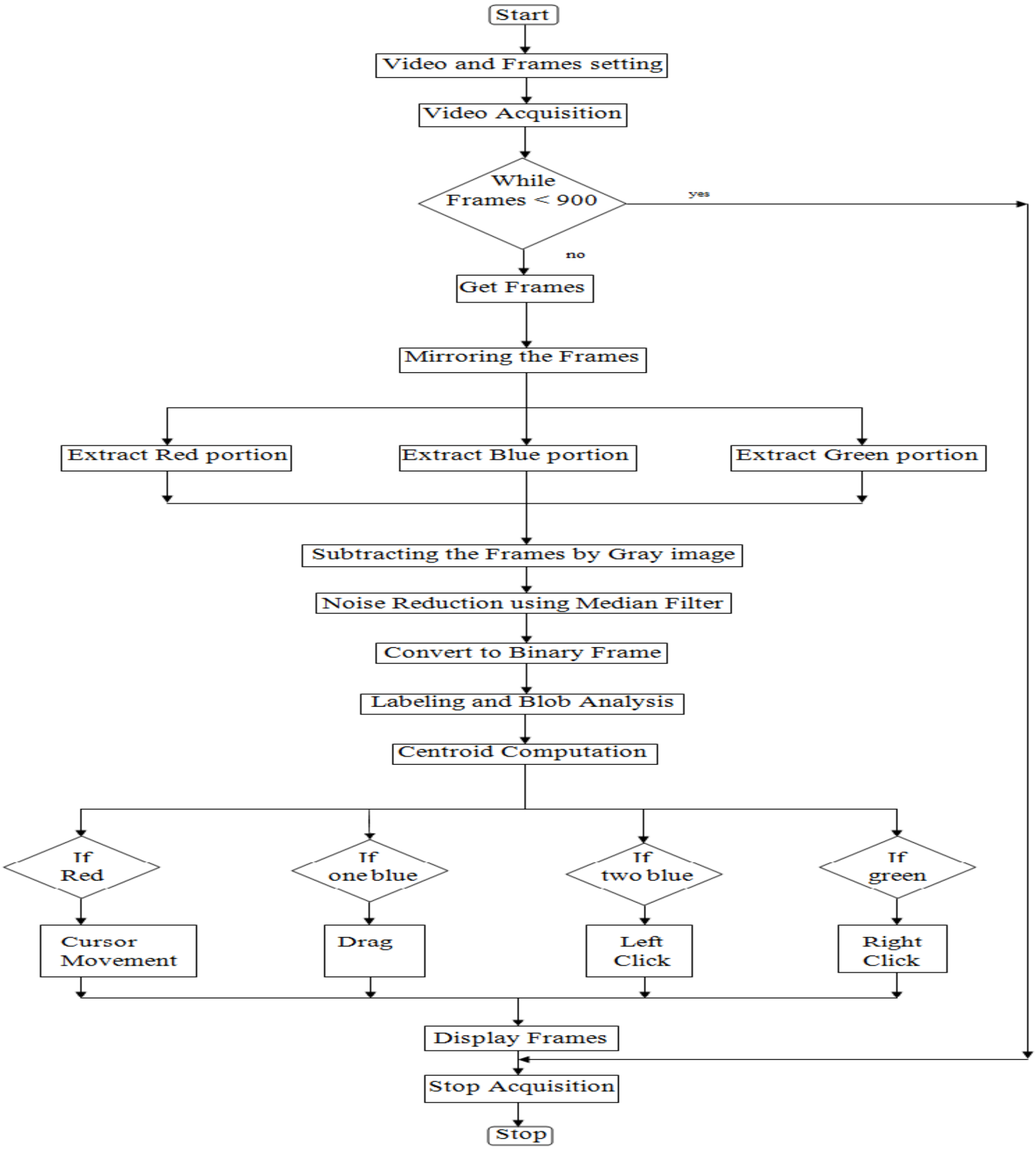


Fig 5 Flow chart of algorithm

4.2. Image acquisition

The first stage of any vision system is the image acquisition stage. After the image has been obtained, various methods of processing can be applied to the image to perform the many different vision tasks required today.

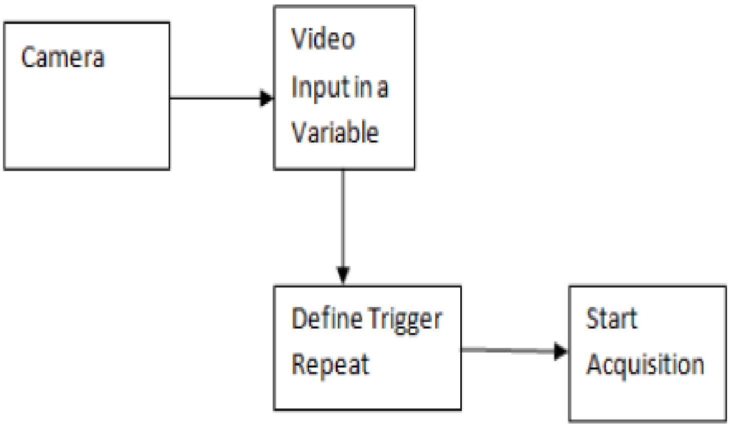


Fig 5 Flow chart of Image Acquisition

4.3. Motion detection

Motion detection in consequent images is nothing but the detection of the moving object in the Scene. In video surveillance, motion detection refers to the capability of the surveillance system to detect motion and capture the events. Motion detection is usually a software-based monitoring algorithm which will signal the surveillance camera to begin capturing the event when it detects motions. This is also called activity detection.

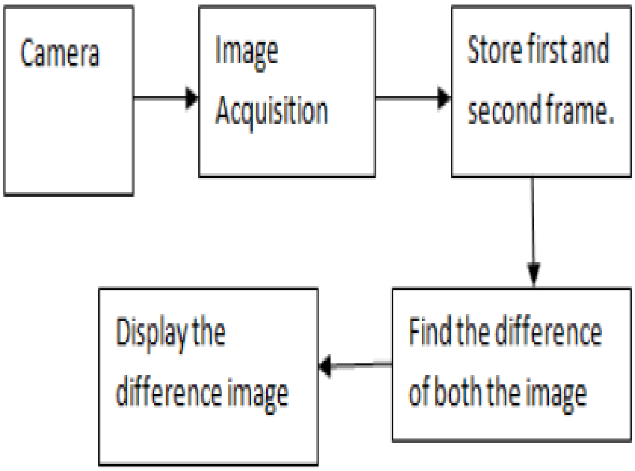

Camera
Image Acquisition
Store first and second frame.
Find the difference of both the image
Display the difference image

Fig 6 Block diagram of motion detection

4.4. Colour recognition

Colour plays an important role in image processing. Each image is composed of an array of M*N pixels with M rows and N columns of pixels. Each pixel contains a certain value for red, green and blue. Varying these values for red, green, blue (RGB) we can get almost any color. Flow diagram of Color Recognition shown in figure

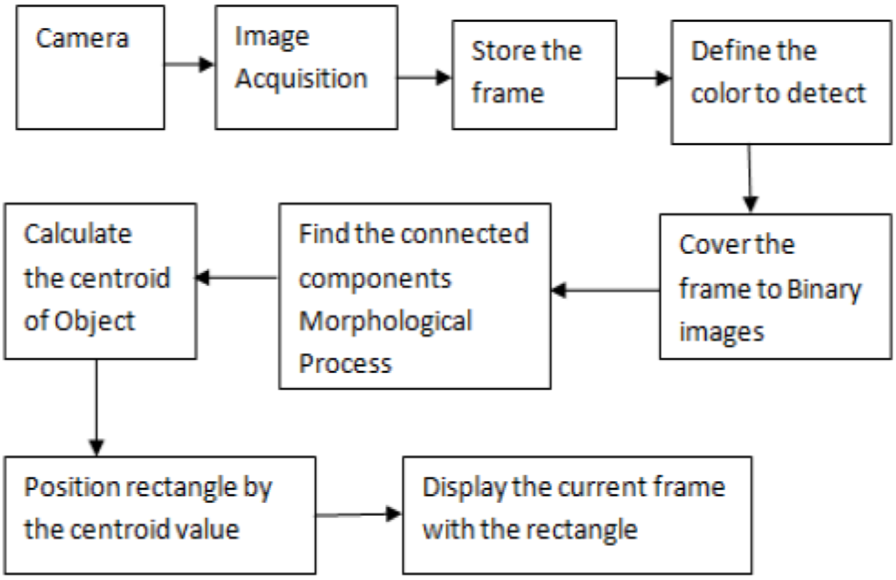

Camera
Image Acquisition
Store the frame
Define the color to detect
Cover the frame to Binary images
Find the connected components Morphological Process
Calculate the centroid of Object
Position rectangle by the centroid value
Display the current frame with the rectangle

Fig 7 Flow diagram of colour recognition

4.5. Mouse cursor movement

We are using the index finger with red colour tape as a cursor controller to control mouse cursor movement. The position mapping the index finger on a camera screen to a desktop screen position. But this method incurs a problem.

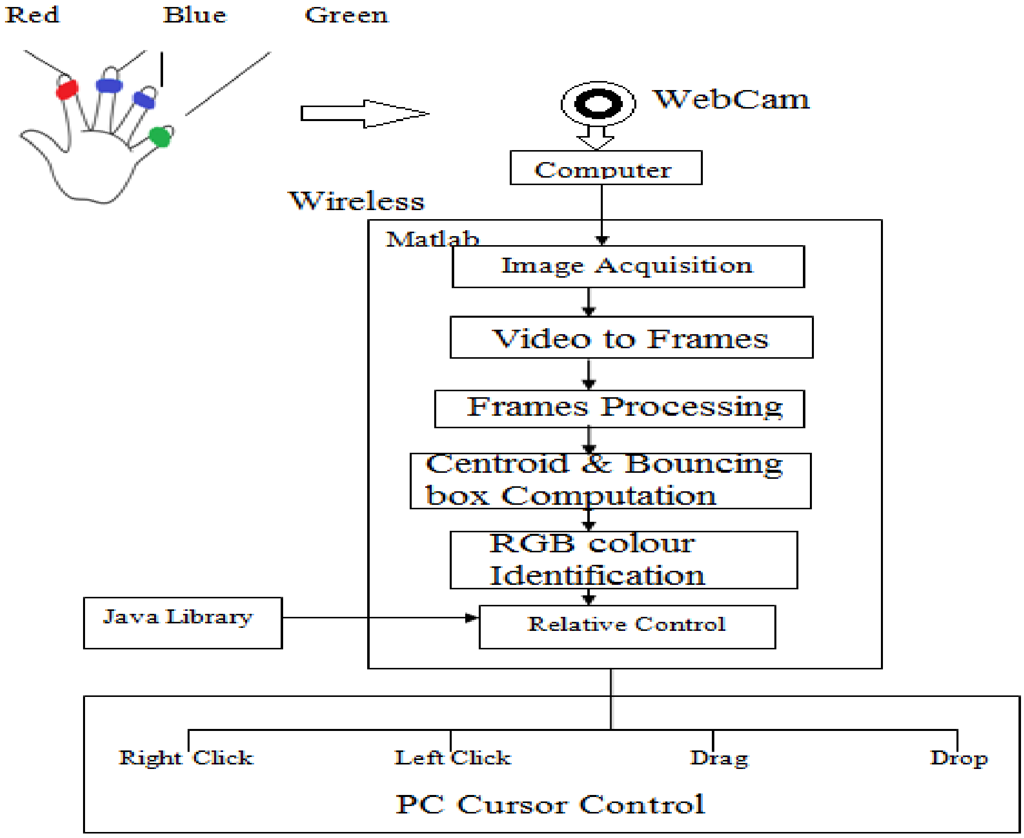


Fig 8 overall process of virtual mouse.

If the resolution of the desktop window is greater than the camera resolution, then the cursor position cannot be accurate because while converting camera resolution to the desktop window resolution, intermediate values are lost. The expected ratio of jumping pixel is up to 4 pixels.

4.6. Mouse interfacing

Using the colour recognition and hand movement, many applications can be developed. It can also be interfaced with computer and can work as a mouse. Similarly a number of mouse interfacing applications can be developed. Many virtual applications can be developed that creates a virtual reality. This works only when the application runs. All the computer applications can be interfaced with hand gesture control and can be controlled directly from hand gesture. It's an example of dynamic gestures, where hand movement (essentially finger in this case) is tracked. Mouse pointer is interfaced to this movement and mouse clicks are initiated through static gestures.

V. CLICK EVENTS

The click events for the mouse are mapped with different hand gestures. The idea focuses on processing between the two colored tapes with different in the fingers. The click events are detailed in the subsequent sub points.

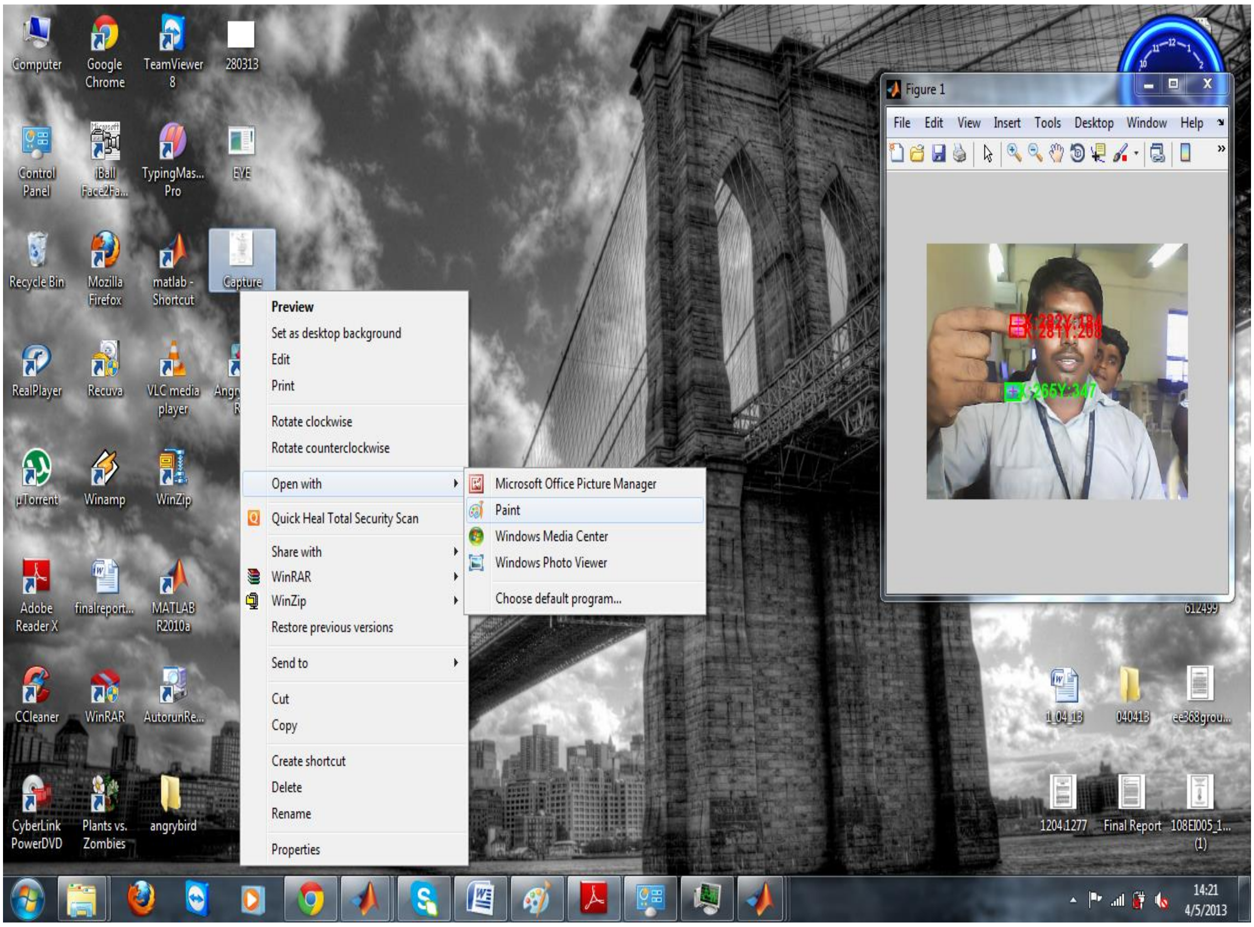


Fig 9 opening a document using virtual mouse

5.1. Dragging operation

The dragging event of the cursor is simulated when the system records the one blue tape in the middle or ring finger in front of the camera pointing at the same location, then the event is recognized as the dragging event of the mouse cursor.

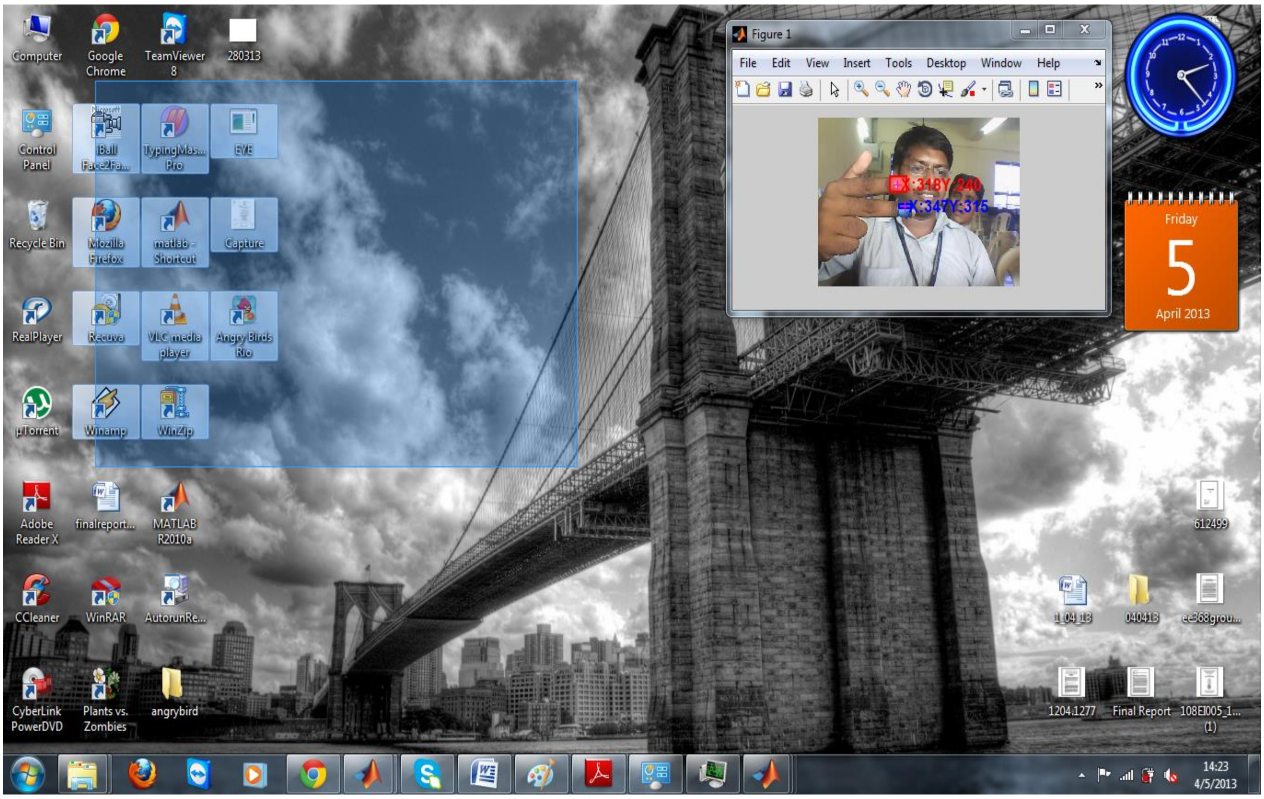
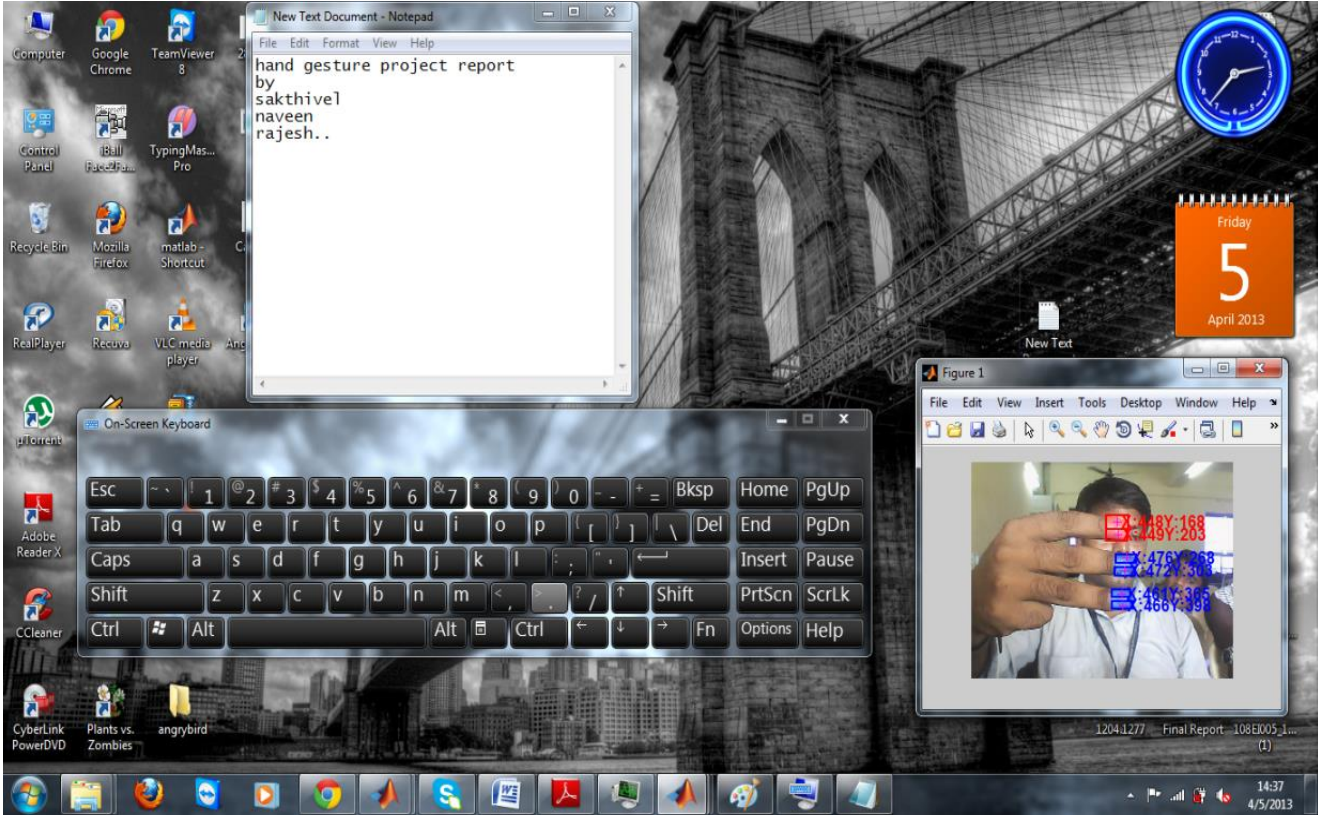


Fig 10 dragging operation using virtual mouse

5.2. Virtual Keyboard

Virtual Keyboard is like an augmented reality. It is designed in Matlab in the same interfacing program. Depending on the finger, application gives input to the keyboard. It checks the frames, based on the finger-colour number or symbol is given as input.


New Text Document - Notepad

```
hand gesture project report
by
sakthivel
naveen
rajesh..
```

VI. APPLICATIONS AND FUTURE ENHANCEMENTS

6.1. Applications

This technology can be used in robotics, gaming and developing systems which could understand human behavior based on their way of interaction.

6.2. Future Enhancements

Hand Gesture recognition can be optimized using developed techniques. Instead of using RGB space, HSV colour space can be used to avoid light intensity fluctuations and thereby giving better results. HAND can be used as a remote control to TV if subject to a camera is embedded. Many entertainment applications can be developed e.g. virtual piano playing, video game, etc. Using high resolution camera or better sensor like a Kinect, high end application can be developed. This can be also utilised in industries machine controlling applications

VII. PERFORMANCE AND CONCLUSION

7.1. Performance

The final result was a complete system that was capable to detect and track objects of at most three and more objects in real time. Although it was not able to track each object separately when there were three objects or more, it could still detect the presence. Experiments also showed that different light settings did not significantly alter the final results. Furthermore, the system was able to ignore background noise very well-mostly came from light reflection. When there were objects that had colour similar to skin colour, both spatial and temporal filtering helped erode these detected regions, therefore reducing the number of false positives.

7.2. Conclusion

The system that we have proposed will completely revolutionize the way people would use the computer system. Presently, the webcam, microphone and mouse are an integral part of the computer system. Our product which uses only webcam would completely eliminate the mouse. Also this would lead to a new era of Human Computer Interaction (HCI) where no physical contact with the device is required.

Our project, Mouse Simulation Using Three Coloured Tapes is an attempt in ubiquitous computing. Here, we will be using two colour tapes on our fingers. One of the tapes will be used for controlling cursor movement while the relative distance between the two coloured tapes will be used for click events of the mouse. Thus, the system will provide a new experience for users in interacting with the computer.

ACKNOWLEDGMENT

The authors like to express their thanks to the management and department of ECE, Rajalakshmi Institute of Technology for their support and encouragement during this work.

REFERENCES

[1] A Low-Cost Hand Gesture Human-Computer Interaction System. Leyuan Liu, *Nong Sang, Saiyong Yang Inst., PR & AI, Huazhong University of Science and Technology, Wuhan, China.E International Conference on Consumer Electronics (ICCE)

[2] Advanced Mouse Pointer Control Using Trajectory-Based Gesture Recognition.Kabeer Manchanda, Benny Bing School of Electrical and Computer Engineering Georgia Institute of Technology, Atlanta, GA 30332

[3] Pointing gesture recognition based on 3D-tracking of face, hands and head orientation Nickel, K. and Stiefelhagen, Proceedings of the 5th international Conference on Multimodal interfaces, Canada, 2003.

[4] Robust Vision-Based Hand Tracking Using Single Camera for Ubiquitous3D Gesture Interaction Sergio Rodriguez, Artzai Picon and Aritz Villodas Infotech Unit, Tecnalia

[5] A gesture-driven computer interface using Kinec Kam Lai, Janusz Konrad, and Prakash Ishwar Department of Electrical and Computer Engineering Boston University Saint Mary’s St., Boston, MA 02215.

[6] Gesture Based Operating System Control Prof. Yuvraj V. Parkale, Electronics and Telecommunication department College of Engineering, Malegaon (Bk), Maharashtra, India. 2012 Second International Conference on Advanced Computing & Communication Technologies.

[7] An Efficient Human-Computer Interaction Framework Using Skin Color Tracking and Gesture Recognition. Nam Vo, Quang Tran, Thang Ba Dinh, Tien Ba Dinh Faculty of Information Technology, University of Science Ho Chi Minh City, Vietnam, M. guyen, International Training Program Center Sai Gon Technology University Ho Chi Minh City, Vietnam.

[8] Developing a Smart Camera for Gesture Recognition in HCI Applications.Yean Choon Ham, School of Electrical Engineering and Telecommunications University of New South Wales, Sydney, Australia.Yu Shi, National ICT Australia, Australian Technology Park, Sydney, Australia.

AUTHORS

S. Sakthivel Born on 11th February 1992 in Pondicherry, Tamil Nadu.He is currently working as a Project Engineer in Wipro. He completed Engineering in the field of Electronics and Communication Engineering in Rajalakshmi Institute of Technology, Chennai. His research interests include image processing and human gesture.

A.Anbaras Born in Pondicherry, Tamil Nadu, on July 6,1992.She completed her Bachelor of Engineering in Information Technology, Pondicherry. Her research interests include Digital image processing

.