Virtual mouse with RGB colored tapes

Article *in* International Journal of Applied Engineering Research · January 2016

CITATIONS
0

READS
529

3 authors, including:

Kanchana Venkatasubbaiah
Amrita Vishwa Vidyapeetham
4 PUBLICATIONS **96** CITATIONS

SEE PROFILE

International Journal of Applied Engineering Research ISSN 0973-4562 Volume 11, Number 8 (2016) pp 5767-5771

Virtual Mouse with RGB Colored Tapes

Upasana V
PG Student, Department of Computer Science,
Amrita Vishwa Vidyapeetham University, Mysuru Campus,
Bogadi 2nd Stage, Mysuru, Karnataka,
Amrita University, India.
E-mail: upasana134@gmail.com

Monisha Joseph
PG Student, Department of Computer Science,
Amrita Vishwa Vidyapeetham University, Mysuru Campus,
Bogadi 2nd Stage, Mysuru, Karnataka,
Amrita University, India.
E-mail: monishajoseph.17@gmail.com

Kanchana V
Assistant Professor, Department of Computer Science,
Amrita Vishwa Vidyapeetham University, Mysuru Campus,
Bogadi 2nd Stage, Mysuru, Karnataka,
Amrita University, India.
E-mail: kanchana.subbaiah@gmail.com

Abstract
A recent development in growing computer environments has resulted in a push for the development of more intuitive forms of human-computer interaction method [HCI]. The aim of this research is to investigate a new approach for controlling the mouse movements using a real-time camera. Instead of changing or adding mouse parts, a method where access to the mouse/cursor by hand gestures with colored tapes can be enabled. It tries to use a camera and computer vision technologies such as gesture recognition and image segmentation to control mouse tasks with colored tapes and shows how it can perform all the mouse functions a current mouse device can. In the first phase, hand gestures are acquired using a camera based on colour detection technique using segmentation and image subtraction algorithm.In the second phase, RGB colored tapes are used to control different functions of the mouse and also the combination of these 3 colors by considering the area of each object/colored tape using the blob analysis and the bounding box algorithm. The user must wear the red, green and blue tapes to the fingers such that it is easy to make the movements for each tape and also the combination of the colored tapes to acquire the desired output of the cursor movement in the system.

Keywords: Color Detection, Video Processing, Virtualized Control, HCI, Vision technology, Cored tapes, Image segmentation, Gesture recognition

Introduction

Due to the recent advancement in computer technology the conventional hardware-device such as mouse and keyboard are becoming less user friendly. Touch technology is one of the widely used techniques for Human Computer Interaction (HCI) in recent times, replacing mouse and keyboard. Computers are becoming part of our daily life and humans are dependent on computers to get the work done in one way or the other, hence other intelligent machine which can be used along with the computer are being developed which helps in friendly Human Computer Interaction. The conventionally used interaction devices (mouse & keyboard) are unnatural and cumbersome to use at times (by disabled people). Gesture recognition is one of the recent techniques which is natural and intuitive means of communication and mostly occur from hands or face of human beings. Gesture Recognition can be useful under such conditions and provides intuitive interaction. Computing environments presently are strongly tied to the availability of a high resolution pointing device with a single, discrete two dimensional cursor.

Our project is an attempt in ubiquitous computing. Thus, the system will provide a new experience for users in interacting with the computer. Here we also make use of HCI, Ubiquitous Computing, Background Subtraction, bounding box technique and integration of Java in MATLAB.

Related Work

A lot of study is being carried out in the fields of Human Computer Interaction (HCI) and Robotics. Developers have tried to control cursor movements using different video devices for HCI. But all of them used different methods to make mouse cursor move. Approach by Hojoon Park made cursor movement through index finger and for clicking events, [1] it used the angle between index finger and thumb. But the angle and the distance was difficult to be maintained in a constant value. Likewise, Erdem et al controlled the mouse action by using fingertip tracking [2]. A click was implemented by defining a screen such that a click was performed when the hand of the user was passed over the region. The drawback of this was each user's skin tone varied

and it was difficult to detect the varying colored tones. One more approach developed by Chu-Feng Lien, controlled the cursor movements and click events by making use of only the finger tips [3]. The method for clicking was based on image density, and the user had to hold the mouse cursor on the required spot for a certain period of time. Unfortunately, it was a difficult task when the duration the hand had to be held in the same position was longer. Paul et al used another method to click where the motion of the thumb was used to mark a clicking event [5]. The mouse pointer was made to move with the movement of the hand with a special sign for each event where the different signs had to be remembered. S Malik proposed and developed a real-time system [4] which could track the 3D position and 2D orientation of the index and the thumb fingers of each hand without making use of any special markers or gloves. The system could also be used for single pointing and pinching gestures. In robotics Asanterabi Malima et al. introduced a finger counting system [6] that controlled the behavior of robots. A research of the existing systems for on-screen choice selection shows that most of the people are still limited to the usage of devices like mouse, joystick, touch screen, touchpad and trackball. These devices need contact of hand with the system. Our proposed approach tries to avoid contact with the hardware

Research Method

Proposed system:

The technology is helpful for the disabled people who face a lot of problems in communicating. This will help them use their voluntary movements, such as fingers and color tapes to control cursor actions and communicate through a customized, educational software. People who suffer from severe disabilities can also benefit from this and take part in recreational activities and use internet or play games. This system uses an externally attached camera or inbuilt camera to capture and detect the user's hand movements and the colors used. The proposed algorithm tracks the motion accurately to control and move the cursor easily.

Image Processing

Image processing is a method of signal processing in which the input is images, photographs or frames of video; the output after processing can be either an image or parameters related to the image or a set of characteristics. Most of the image-processing techniques involve handling the image as a 2D signal and applying standard signal processing techniques to that. Image processing generally refers to digital image processing, but analog and optical image processing are also possible. In this, processing is in the sense that we are dividing each pixel of each image into different RGB components.

Stage one of this project is to acquire the video input from the PC. Features of the integrated camera and also the screen size are obtained and the information obtained is fed into the program. The images or frames are obtained through the camera. The video output is obtained from the video preview function and its format can be specified. In our project we have obtained RGB output with both the maximum width and length of the screen. The screen length is set to 100 for the proper resolution. Video processing is done by considering each frame as an image using different image processing tools in the MATLAB. Once the processing is done, we free the memory by deleting the video. The step by step process done on the image to get the desired result is explained in the next section of the paper.

Capturing Of Video

The web camera captures the video at a fixed frame rate and also resolution which is actually determined by the hardware of the web camera. The resolution and frame rate can be changed if required. The video is divided into multiple image frames. This depends on the FPS (Frames per second) of the camera.

Flipping Of Images

Whenever the camera captures an image, it is an inverted one. It means that if we move the pointer/tape towards the left, image of the pointer moves towards the right and vice-versa. It is alike to the image obtained if we stand in front of a mirror (Right is detected as left and left is detected as right). To avoid this, we must vertically flip the image. Image captured is an RGB image but flipping actions cannot be performed on it directly. So the individual color channels of the image are divided and then flipped individually. Once the flipping of red, blue and green colored channels completes, they are concatenated and then an RGB image which is flipped will be obtained.

Conversion Of Flipped Image Into Gray Scale Image

A grayscale image that is M pixels tall and N pixels wide is represented as a matrix of double data type of size M × N. Element values like (frame image (m, n)) denote the pixel grayscale intensities in [0,1] with 0=black and 1=white. The computational complexity is reduced in a gray scale image when compared to a colored image. Hence, the flipped image is converted into gray scale image. Later, all the necessary operations are performed after converting the image into gray scale.

Color Detection

```
redThresh = 0.24; %Threshold for red detection
greenThresh = 0.05; % Threshold for green detection
blueThresh = 0.15; % Threshold for blue detection
```

Detection of colors is done with MATLAB function *Z=imsubtract(x, y).*

Where (x,y) are the real numeric arrays of same size and class. This function subtracts each element in array Y from corresponding element in array X and outputs the difference in the corresponding element of the output array Z. This imsubtract will detect the RGB colours.

Conversion Of Gray Scale Image To Binary Image

A binary image is represented by an $M \times N$ logical matrix

where pixel values are 1 (true) or 0 (false).To obtain a binary image from a grayscale image, MATLAB's built in function "*im2bw"* function has been used. It converts the grayscale image (I) to a binary image (*bw).*
Bw=im2bw(I, level)

The output image replaces all the pixels of the input image with a luminance greater than a level with the value 1 (white) and also replaces all the other pixels with the value 0 (black). This range is relative to the signal levels that are convenient for the image's class. Thus, a level value of 0.5 is middle color between black and white, regardless of the class. Here, the threshold 0.15 gave the best result for the large range of illumination change

Figure 1: Detected region's Binary Image

Bounding box
This is used extensively when finding areas of various shapes using coordinate geometry. This method involves first drawing the bounding box, and then subtracting the areas of simple shapes created around the edge of it to find the area of the desired object.

- The top of the rectangle is determined by the y-coordinate of the top-most point-point B
- The bottom of the rectangle is determined by the y-coordinate of the lowest point-point D
- The left side of the rectangle is determined by the x-coordinate of the leftmost point-point A
- The right of the rectangle is determined by the x-coordinate of the rightmost point-point C

The centroid $(\bar{x}, \bar{y})$ of a region bounded by the graphs of the continuous functions M and N such that $M(x) \geq N(x)$ on the interval [a, b], $a \leq M \leq b$, is given by

$\bar{x}=\frac{1}{A}\int_a^b \; x[\, M(x)\text{-}N(x)]dx$

$\bar{y}=\frac{1}{A}\int_a^b \; \frac{M(x)+N(x)}{2}\, M(x)-N(x)\, dx$

Where, A is the area of region(given by $\int_a^b \; M(x)-N(x)dx$)

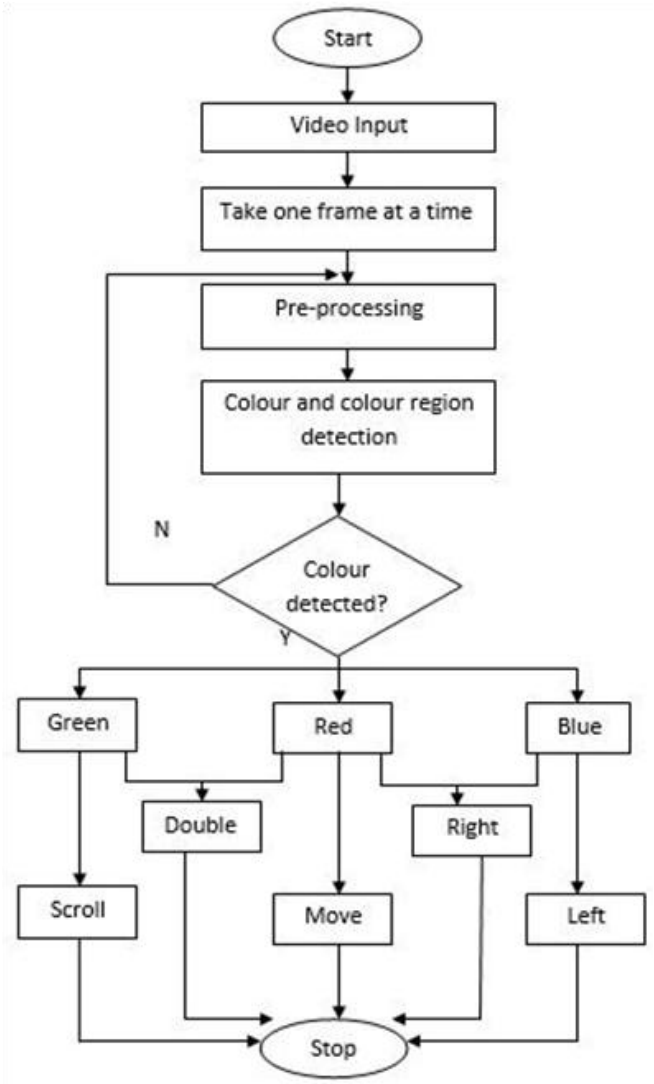
Finding The Centroids
To control the mouse pointer, it is necessary to decide a point whose coordinates can be sent to the cursor. With the help of these coordinates, the system controls the movement of the cursor.To find the centroid of a specific object in an image, it needs to segment the object out of the image and then find the center point.For that, an inbuilt function in MATLAB is used to find the centroid of the detected object/region. The output of function will be in the form of a matrix consisting of the (X, Y) where X (horizontal) and Y (vertical) coordinates of the centroid.

Tracking The Mouse Pointer
After the coordinates have been determined, the mouse driver is accessed and these coordinates are passed on to the cursor. Using these coordinates, the cursorplaces itself in a desired position. Since the object moves continuously, each time new centroid are determined and for each new frame, the cursor will obtain a new position, thus creating the effects of tracking. As the user moves his/her hands across the screen, the mouse cursor moves accordingly. There are no inbuilt functions in MATLAB that can directly access the mouse drivers of the system/computer. But MATLAB supports integration with other languages such as C, C++, and JAVA. As java is a machine independent language, it is preferred over the rest. A java object is created and then it is linked with the computer's mouse drivers.
Depending on the detection of other colors along with red, the system performs the click events of the mouse. Color codes can be modified as required..The java class *"java.awt.Robot"* has all the mouse functions inside it and is used in MATLAB to simulate the mouse events.

Performing Clicking Action
Control actions of mouse are performed by controlling the flags that are associated with the mouse buttons. Therefore, JAVA is used to access these flags. The user has to provide hand gestures as an input in order to create the controlling actions. Since we are using colored pointers, the computation time required is decreased. Besides, the system becomes unaffected to background noise and also low illumination conditions. The detection of the colors green and blue follows the same procedure as above. Clicking action is based on the simultaneous detection of specific two colors as mentioned below.

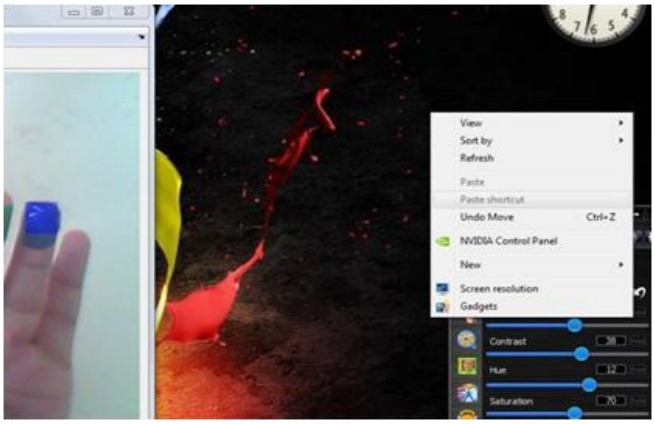
- If Red along with Green colour is detected, double clicking action is performed.
- If Red with Blue is detected, it performs right-click action

GUI interface:
First step is to acquire the real time video. After the image acquisition the pre-processing or image enhancement is done. After this, the segmentation process takes place where the color detection happens. Each color and the centroid position are assigned to a particular function of the mouse. The detailed process description and screenshots is given in the flow diagram.

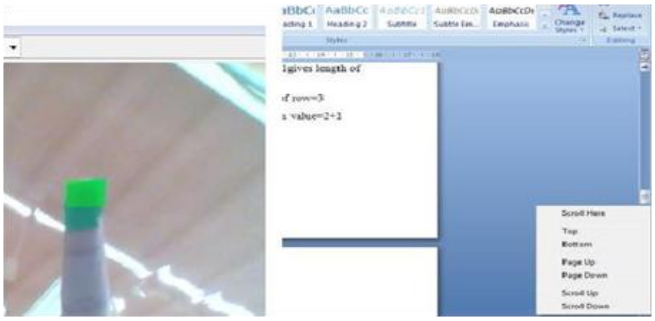


Experimentation And Results

Red Colour Detection: Red colour makes the mouse pointer to move on the screen.



Blue Color Detection: blue colored object produces the left click function.



Green Color Detection: The green color and the colored object are used to produce scroll action of the mouse.

Conclusion

In this paper we have tried to control the mouse pointer using the gesture recognition along with the different color band to perform the mouse action. The presented system can be improved by adding more mouse features with different color band.

Since our system is based on image capture through a webcam. And hence, it is dependent on illumination up to a certain extent. Also, the presence of other color objects in the background may cause the system to give faulty response. Although by configuring the threshold and other parameters of system, the problem can be reduced. But still it is better that the operating background be light and no bright colored objects being present. System might run slower on some computers with low computational capabilities as it involves complex calculations in a small amount of time.

Future Enhancements

This project can be developed with various techniques. HSV color space can be used to avoid light intensity fluctuation instead of using the RGB space. The same can be used as a TV remote if a camera is embedded to the TV. Entertainment applications can be developed such as virtual piano, video gaming etc. By using high resolution camera or better sensors,

high end application can be developed. This can also provide services in industries by developing apps for controlling machines.

References

[1] Hojoon Park, A Method for Controlling Mouse Movement using a Real-Time Camera, Department of Computer Science Brown University, Providence, RI, USA,2008

[2] A.Erdem, E.Yardimci, Y.Atalay, V. Cetin, "Computer vision based mouse", A. E. Acoustics, Speech, and Signal Processing, 2002. Proceedings. (ICASS). IEEE International Conference.Volume:4

[3] Chu-Feng Lien, "Portable Vision-Based HCI-A Real-time Hand Mouse System on HandheldDevices".

[4] Shahzad Malik, "Real-time Hand Tracking and Finger Tracking for Interaction", CSC2503F Project Report, December 18, 2003

[5] Robertson P., Laddaga R., Van Kleek M., "Virtual mouse vision based interface", January 2004

[6] Asanterabi Malima, ErolOzgur, and Mujdat Cetin, "A Fast Algorithm for Vision-Based Hand Gesture Recognition for Robot Control", Signal Processing and Communications Applications, 2006 IEEE 14th.

[7]. Abhik Banerjee1, Abhirup Ghosh, "Mouse Control using a Web Camera based on Colour Detection", Department of Electronics & Communication Engineering, Sikkim Manipal Institute of Technology East Sikkim, India.(IJCTT)-volume 9 number 1-Mar 2014.

[8]. ShanyJophin, Sheethal M.S, Priya Philip, T M Bhruguram, "Gesture Based Interface Using Motion and Image Comparison" Dept of Computer.Science AdiShankara Institute of Engineering And Technology, Kalady, (IJAIT) Vol. 2, No.3, June 2012.

[9] A. V. Nadargi, Vaibhav Danve, Varsha Tikale and Ashwini Udmale, "Human Computer Intelligence Interface Based on Virtual Mouse", Department of Computer Engineering, Pune University, India. Vol. 6, Issue, 03, pp.5696-5699, March, 2014.

[10] K S Chidanand Kumar, "A Vision based Application for Virtual Mouse Interface Using Finger-Tip" E&C Department, National Institute of Technology Surathkal, Karnataka, India, Vol. 3 (3), 2012, 4460-4464.

[11] Dikshant Rathod, Kanchan Kadlag, Savita Anarase, PoonamGavade, "Virtual Mouse", International Journal of Engineering and Innovative Technology (IJEIT) Volume 4, Issue 7, January 2015.

[11] Shweta Jain, Shrikant Ujjainkar, "Finger Tracking Virtual Mouse", School of computer science and engineering VIT University Vellore, India. (TIJCSA) Volume 2, No. 03, May 2013 ISSN-2278-1080.

[12] Abishek Easter Raj.A, Anto Daniel.S, "Virtual Mouse based on Color Tracking of Face Makeup", International Journal on Applications in Information and Communication Engineering, Volume 1: Issue5:May 2015

[13] Suresh D, Bhavana, "Virtual Mouse Implementation using Color Pointer Detection" Channabasaveshwara Institute of Technology, International Journal of Research Studies in Science, Engineering and Technology Volume 1, Issue 5, August 2014, PP 23-32

[14] S. Sakthivel, A. Anbarasi, "Hand Gesture Recognition System for Human-Computer Interaction", International Journal of Recent Trends in Electrical & Electronics Engg., Feb. 2014, ISSN: 2231-6612.

[15] Abhik Banerjee, Abhirup Ghosh, Koustuvmoni Bharadwaj, "Mouse Control using a Web Camera based on Colour Detection", International Journal of Computer Trends and Technology (IJCTT)-volume 9 number 1-Mar 2014