Design of Human Computer Interaction using Image Recognition and Gesture Movements

Swarali Narvekar
EXTC Department
Vivekanand Education
Society's Institute of
Technology, Mumbai University

Manali Godse
Biomedical Department
DJ Sanghvi College of
Engineering
Vile Parle (W), Mumbai
University

ABSTRACT

This paper deals with the implementation of an invisible computer mouse that enables interaction with computer without attaching a hardware mouse. The methodology used is based on the SixthSense Technology where the user is able to move the cursor by the movement of fingers. The basis of the approach is a fast segmentation process to obtain the moving hand from the whole image against different backgrounds and lighting conditions. The movements and arrangements of the hand fingers are grasped as gestures that act as an interaction instruction for the projected application interfaces.

General Terms

Motion Detection and Gesture Movement

Keywords

Colour markers, Open CV and SixthSense

1. INTRODUCTION

'SixthSense' Technology augments the physical world around us with digital information and enables the use of natural hand gestures to interact with that information. By using a camera, 'SixthSense' sees what user sees and visually augments any surfaces or objects the user is interacting with. It projects information onto surfaces, walls, and physical objects, and enables user to connect with the projected information through natural hand gestures, arm movements, or interaction with the object itself.

2. RELATED WORK

2.1 Mouse Movement using Colour detection

Vision-Based Human-Computer Interaction through Real-Time Hand Tracking and Gesture recognition vision-based interaction is an appealing option used for replacing primitive human computer interaction (HCI) via an invisible mouse. The functions of vlcj to capture the images given by the user and store the captured images in a temporary storage buffer [3]. They have examined pixel (skipping consecutive four pixels) of the image captured to find the red colour. The pointer reaches the position of the midpoint by examining x and y coordinates of the current area detected. The midpoint of another pixel containing the red colour is found out. The current pixel is compared with previous pixel to move the cursor. If the compared two pixels are the same, then the pointer starts moving. The mouse functions are performed depending upon the distance detected between the new computed pixels. Another approach works in a continuous manner where camera takes the live video, sending it to the laptop, and MATLAB installed in laptop processes the input and recognizes the colours at the finger tips of the user [1]. Cursor movement is enabled by bridging the crevice between the two different worlds i.e. Physical and Digital world by using vision based technique. The incoming live video from camera is converted into frames of images. These images that are obtained from the slicing of video are then processed for colour recognition process. The output of the colour recognition process are the images that contains only those colours of which colour caps are present at the finger tips of the user. For this purpose, RGB values of the colour caps are set prior in the code so that no other colour will be detected in the image after colour recognition except the caps colours. The output is only those colours which are present at the fingertips of the user. The colour is then associated with the mouse cursor in code so that whenever the colour moves in the output image from one position to another, the mouse cursor gets attached at the same position where the colour is now displayed.

Mouse Simulation is accomplished using Two coloured Tapes [5]. Here, tapes of two different colours are applied on the user's hand fingers. One of the tapes is used for controlling cursor movement while the relative distance between the two coloured tapes is used for click events of the mouse. The first step in the designed system is to separate out the potential hand pixels from the non-hand pixels. After background subtraction, the process of skin segmentation is done. The skin segmentation process outputs an image which is ready for detection of color tapes in the finger. For this an algorithm based on HSV color space is used which is very effective to select a certain color out of an image. The index finger with yellow colour tape is used as a cursor controller to control mouse cursor movement. The system records the distance (say D) between the yellow and red tapes in the index finger and the thumb respectively. Here, the index and thumb must be apart as much as possible so as to get maximum distance .This distance is regarded as the threshold distance for the event. Now, as the thumb moves towards the index finger, the distance between the finger tips or in other words, the distance between yellow and red tapes is decreases. In the second step, when the thumb is close to the index finger the system records the reduced distance. The mouse cursor event takes place depending upon the change in the distance between the two coloured tapes.

2.2 Mouse Simulation Using Skin Detection

A prototype consisting of a virtual screen and colour cameras has been developed for colour recognition. The screen is assumed to be a flat surface with four rectangular coordinates sent to the system to map it to the workstation screen. The cameras are mounted on top of the screen in order to cover the maximum view. After the finger comes in contact with the screen, the coordinates are transferred into the system to override the mouse handling operations. The mechanism operates in three stages. Firstly, a camera captures a real time video of moving hand in front of the screen. The three stages are then performed on the captured frame. In the first stage different implementation models are employed. The colour space model is used to extract the skin pixels; motion detection is used to extract the identified object [4]. The second stage comprises of calculating the probability of the hand identified by convoluting the outputs of different models in stage I to obtain the finger tip. In the third stage, the finger coordinates detected are transferred on the mouse server with the help of sockets [2]. These coordinates are then used to over ride the system mouse movements.The first stage is the Skin Pixel identification. To identify skin pixels, a colour space model is used. Various colour space models such as RGB, HSV, LUV, YCbCr, etc. have been investigated. The YCbCr component values of Human Skin colour are calculated. The pixels of the input image are classified into skin colour and non-skin colour clusters by using a thresholding technique that exploits the information of a skin colour distribution map in the YCbCr colour space. Then the object is detection is achieved by building a representation of the scene called the background model or the reference frame. Deviations from the model for each incoming frame are found. Any significant change in an image region from the background model signifies a moving object. Difference in the reference image and the current image is used to find the object. Morphological operations such as Erode, Dilate are performed on the image to enhance the object identification. The edges of the extracted object are identified by canny edge detection algorithm. After processing the edges, different contours are identified. Assuming that the object represents the largest contour, it is extracted in order to detect the finger tip location using structural analysis.The finger tip is located by applying a convex hull algorithm. Each feature either crest or trough belonging to the hand is identified as a defect. The finger tip is then identified from among these defects. The finger tip location gives the position of the coordinates. The coordinates then retrieved are transferred for further processing in the third stage to simulate the mouse movements. The coordinates found through stage II are converted to xml format, so as to transport the coordinates in universal accepted format. These coordinates are fed as an input to the mouse simulator which maps the xml points to mouse coordinates by overriding the mouse handling events. The overridden functions then simulate the mouse movements as per their definitions.

2.3 Hand Gesture Recognition Method for Mouse Operations

Hand gestures are used for natural and intuitive human-computer interaction. The method used combines existing techniques of skin colour based ROI segmentation and Viola-Jones Haar-like feature based object detection, to optimize hand gesture recognition for mouse operation. The mouse operation has two parts, movement of cursor and clicking using the right or left mouse button. In this paper, colour is used as a robust feature to first define a Region of Interest (ROI). Then within this ROI, hand postures are detected by using Haar-like features and AdaBoost learning algorithm. The AdaBoost learning algorithm significantly speeds up the performance and constructs an accurate cascaded classifier by combining a sequence of weak classifiers. Once gestures are recognized, then they are assigned to different mouse events, such as right click, left click, undo, etc [7]. First, input image is converted from RGB to HSV colour space. Next, ROI module segments the hand from background by using skin colour. The minimum and maximum skin ranges are Hue = 0 - 20, Saturation = 30 - 150, and Value = 80 - 255. The skin colour pixels, which have the values between these ranges, are selected. A rectangle is then drawn around the detected skin colour area. This sub image is then fed to the Haar-like feature classifier. Next, the Haar-like feature classifier detects and recognizes the subsequent gestures to be used for computer mouse operations. To meet the requirement of real-time performance, the classifiers are trained with Haar-like features, which can be computed efficiently by using the integral image technique. A boosting algorithm which can efficiently get rid of false images and detect target images with selected Haar-like features stage by stage.The mouse cursor is controlled by the approximate centre of the detected gesture. To do this, first a bounding box is drawn around the detected gesture. Then a circle is drawn which best fits the bounding box. The centre of the circle is approximately the centre of the palm. The movement of centre of the palm is used as a reference for the motion of the cursor. Open Palm gesture is used for mouse movement. To perform the mouse operations like left-button click, right button click, etc., five different hand gestures representing these operations are "palm", "fist", "two fingers", "three fingers", and "little finger" . Four gestures are enough to perform the basic mouse operations. The fifth gesture is used to undo the last command. Undo operation is useful in case of any errors. Five gestures have been carefully selected in order to obtain a high recognition rate.

3. SYSTEM DESIGN

The main purpose is to operate a computer mouse action by recognizing hand gestures using a simple Crystal web camera. The system is classified into two main modules:

1. Hardware Module
2. Software Module

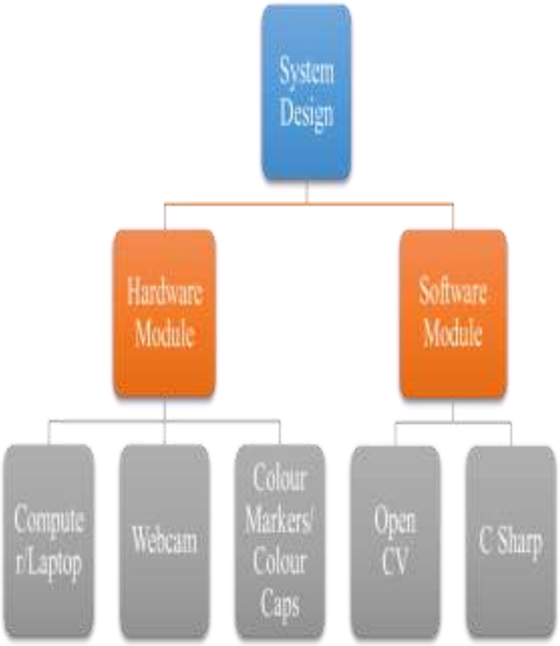

Fig 1: System Design

3.1.1 Computer / Laptop

The system is developed using Aspire 5742 Acer computer with Intel$^{®}$ Core^{TM} I5-480M at 2.67 GHz of CPU and 2G RAM is used. It uses a 64 bit operating system. The Hard Disk used is 500GB.

3.1.2 Webcam

The in-built Crystal Webcam with 1.3 M pixels resolution is used as the main source of input. It is present on top of the display screen. With a 1.3 megapixel CMOS lens, it is capable of up to a 1280 x 1024 resolution. Its Power Line Frequency (Anti Flicker) is adjusted to 60Hz. The Acer Crystal Eye webcam captures videos and images with vivid detail and clarity. The first interaction of the real time world takes place with the digital world via a webcam. A webcam captures and recognizes the object in view and tracks the user's hand gestures. The camera, in a sense, acts as a digital eye, seeing what the user sees. It tracks the movements of the thumbs and index fingers of both of the user's hands. It sends the data to the laptop in which Microsoft Visual Studio 2010 and Open CV have been installed for further processing.

3.1.3 Colour Markers

Colour Markers or Colour Caps are used at the tip of the user's fingers. Marking the user's fingers with different colours helps the webcam to recognize gestures. The movements and arrangements of these markers are interpreted into gestures.

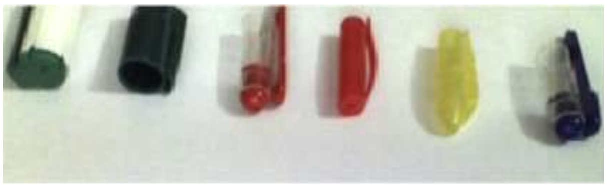

Fig 2: Colour Caps

3.2 Software Module

The algorithm had been developed in Open CV. C Sharp programming language has been used for experimental purpose.

3.2.1 Open CV

OpenCV is an image processing library. It contains a large collection of image processing functions. OpenCV Library Version 2.4.2. has been downloaded from their website. The OpenCV library was officially introduced in 1999 by Intel Research initiative to advance CPU-intensive application. The OpenCV library in the earlier version written in C. However since version 2.0, OpenCV includes both C interface and C++ interface. Starting version 2.2, OpenCV can also be built for Android operating system.

3.2.2 C Sharp

C Sharp is a simple, modern, general-purpose, object-oriented programming language. It is full of features that make development faster and easier, usually at the cost of flexibility and/or runtime performance. The execution of the program takes place at a faster rate as compared to the program in C, C++, JAVA and Matlab. An algorithm is developed in C Sharp for image comparison and motion detection

4. METHODOLOGY

Fig. 3 shows the algorithm used to move mouse cursor on screen using SixthSense technology. Webcam is used to capture the real time video and send the captured information for further processing. The mechanism works in a continuous manner in which the captured real time information is converted into frames of images. The frames of images are obtained continuously one after the other and the latest frame is considered further for processing i.e. for the colour recognition purpose. The camera captures the object in a polar fashion i.e. the right side of the real time image appears as left in the captured image and vice-versa. Therefore the captured image is first and foremost resized and flipped so that the captured image is found to be the same as the real time image. The image captured from the webcam is first and foremost resized for processing. It is resized to an image of 320×240 pixels standard value. The size of the screen is (320,240).

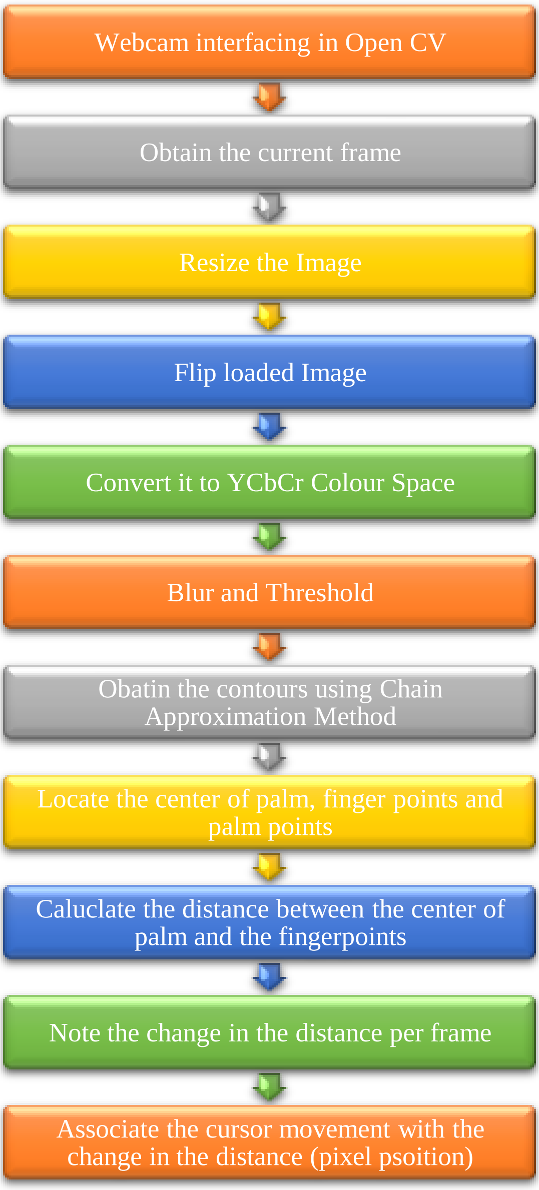
Webcam interfacing in Open CV
Obtain the current frame
Resize the Image
Flip loaded Image
Convert it to YCbCr Colour Space
Blur and Threshold
Obatin the contours using Chain Approximation Method
Locate the center of palm, finger points and palm points
Caluclate the distance between the center of palm and the fingerpoints
Note the change in the distance per frame
Associate the cursor movement with the change in the distance (pixel psoition)

Fig 3: Skin Detection

This is present in the program. But the usual monitor size is 1024×768. Therefore the screen size is caliberated to the monitor size. Image is converted from RGB to YCC colour space. The parameters are set at 0, 131, 80 255, 185 and 135 for the detection of human skin. The Erosion function is used to filter out the smaller noise, but the scope of the necessary information i.e. captured colour marker will also be affected by deformation, so it is necessary to use Dilation function to recover the required information. The sharp contours or edges of the detected object are smoothened by blurring the image by using a Gaussian filter. Threshold being one of the simplest segmentation methods is used to separate out those regions of an image corresponding to the objects which we want to analyze. This separation is based on the variation of intensity between the object pixels and the background pixels. Once the important pixels are separated properly, set them with a determined value to identify them. Here the range of the values in between 20 to 255. The cursor movement is associated with the change in the pixel value from one frame to another. The function threshold binary is used to perform the threshold operation. The palm area i.e. the contour of the palm is found out by using Contour Approximation function in Open CV. The palm area is calculated using convexity defect. The required set of points is created from the convexity defect depth points with the help of which the centre of the palm is detected using minimum enclosing circle. We have then detected the finger points i.e. tip of all the fingers. The palm bounding circle is drawn joining all the finger points. Then the distance between the centre of the palm and the finger point is calculated. This distance keeps on changing from one frame to another as the user moves his/her fingers. In OpenCV one needs to import the dll file i.e. dll file user32.dll to control the mouse pointer. The gestures shown in Fig. 4 have been used extensively to perform the Left click and right click operations respectively.

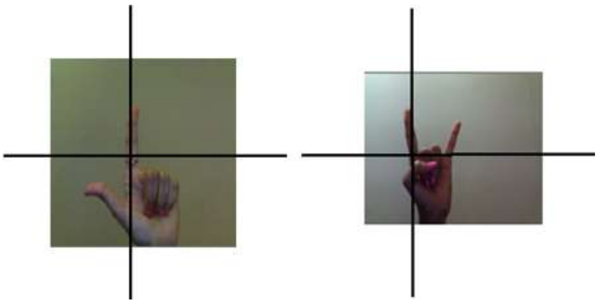

Fig 4: Gestures used for Left click and Right click

5. RESULTS

Each hand gesture was successfully detected and tracked. The above mentioned system can also detect and differentiate between different gestures even if hand is at different angles with respect to horizontal axis successfully to a great extent. Moreover, it can also differentiate gestures when hand is rotated along vertical axis, but only as long as fingers do not overlap. Table 1 shows the detected samples, missed samples, and hit percentage each gesture in real time. In this case the timer count is set equal to 10, 20 and 30 for single click, double click and drag operations.

Table 1 Detection Rate of Different Gestures

Gesture	Total Samples Taken	Samples Detected	Samples Missed	Average Detection Rate
Single Finger	100	100	00	100
Two fingers for left click	100	100	00	100
Two fingers for right click	100	100	00	100

From the above Table 1 it can be seen that all the samples have been detected successfully for single finger which is associated with the mouse scrolling operation as well as two fingers. Out of 100 samples taken, all 100 samples are detected successfully and none are missed. The success detection rate is one hundred percent. The graphical representation of the above tabular format is shown in the below figure 5.

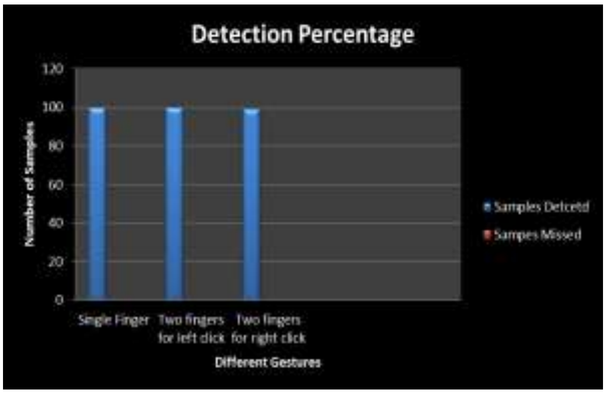

Fig 5: Detection Rate of Different Gestures (Case-I)

Table 2 shows the detected samples, missed samples, and hit percentage each gesture in real time. For this case we have set the timer count equal to 5, 10 and 20 fdor single click, double click and drag operations.

Table 2 Detection Rate of Different Gestures

Gesture	Total Samples Taken	Samples Detected	Samples Missed	Average Detection Rate
Single Finger	100	100	00	100
Two fingers for left click	100	99	01	99
Two fingers for right click	100	99	01	99

From the above Table 2, it can be seen that all the samples have been detected successfully for single finger which is associated with the mouse scrolling operation. So the success detection rate for these is equal to hundred percent. Out of 100 samples taken, 99 samples are detected successfully and one is missed for left click and right click operation. The success detection rate is equal to ninety nine percent. The graphical representation of the above tabular format is shown in the below figure 6.

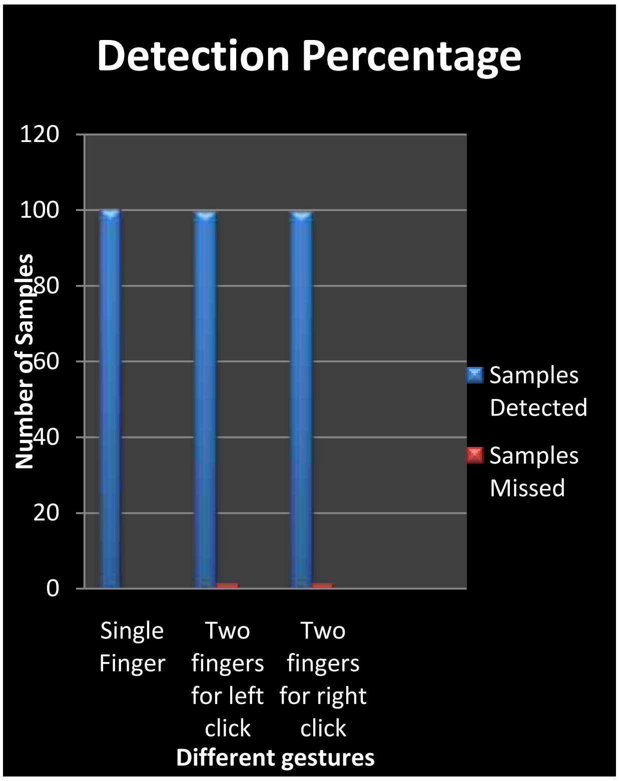

Fig 6: Detection Rate of Different Gestures (Case-II)

The below Table 3 gives the tabular representation of accuracy percentage ..of different cursor movements. For the different parameters, the scroll operation has been performed successfully with hundred percent accuracy for both right scroll and left scroll operations. The accuracy decreases for the single click, double click operations and is the least for left drag operation.

Table 3 Accuracy analysis of Different Mouse Control Operations

Gesture	Accuracy Percentage (%)	
	Left	Right
Scroll	100	100
Single Click	98	96
Double Click	96	94
Drag	96	--

6. CONCLUSION

The system based on image recognition and motion detection technology is used to perform mouse control operations. The mouse operations are associated with the skin detection process instead of colour detection process. The use of colour markers is completely eliminated by associating the cursor with the human skin thereby reducing the hardware components. The system has undergone informal user testing and from the results it can be seen that the combination of timer and gesture recognition is used to perform different mouse control operations such as scrolling, single click, double click and drag operation. There are modifications that can be made to the system like developing the performance of the current system and adding features such as enlarging and shrinking windows, closing window, etc. by using the palm and multiple fingers. The current system is variant to reflection and scale changes and requires proper hand gestures, good illumination technology and powerful camera for the performance of mouse functions. From the results, one can expect that if the algorithms can work in all environments then the system will work more efficiently.

7. REFERENCES

[1] Kunal Gupta, Prateek Agrawal, Mouse Movement Through Finger By Image Grabbing Using Sixth Sense Technology, , [IJESAT] International Journal Of Engineering Science & Advanced Technology Volume-2, Issue-2, 245 – 249 Mar-Apr 2012.

[2] Mridul Birla, Rahul Kulal, Shruti Bardia and Sugandha Rudra TouchX- Mouse Simulation Using Gestures on a Virtual Screen Department of Information Technology, Pune Institute of Computer Technology, Pune, India 2012 IEEE Students' Conference on Electrical, Electronics and Computer Science.

[3] Priya Philip, T M Bhruguram Shany Jophin, Sheethal M.S, Gesture Based Interface Using Motion and Image Comparison, International Journal of Advanced Information Technology (IJAIT) Vol. 2, No.3, June 2012.

[4] Swaminathan P.L. and Vivek Veeriah J. Robust Hand Gesture Recognition Algorithm for Simple Mouse Control International Journal of Computer and Communication Engineering, Vol. 2, No.2, March 2013.

[5] Kamran Niyazi, Swapnil Mahe, Swapnil Vyawahare and Vikram Kumar, Mouse Simulation Using Two Coloured Tapes International Journal of Information Sciences and Techniques (IJIST) Vol.2, No.2, March 2012.

[6] K.Arai & R. Mardiyanto Camera as Mouse and Keyboard for Handicap Person with Troubleshooting Ability, Recovery, and Complete Mouse Events International Journal of Human Computer Interaction (IJHCI), Volume 1 Issue 3.

[7] New Hand Gesture Recognition Method for Mouse Operations Ehsan ul haqI, Syed Jahanzeb Hussain Pirzadcl, Mirza Waqar Bailand Hyunchul Shin4 School o/Electrical and Computer Engineering, Hanyang University,Erica Campus, Korea 978-1-61284-857-0/11/$26.00 ©2011 IEEE.