International Research Journal of Computer Science (IRJCS)
Issue 12, Volume 3 (December 2016)

ISSN: 2393-9842
www.irjcs.com

Sensible Mouse using Sixth Sense Technology

Premangshu Chanda
Dept. Of Computer Science
St. Xavier's College, Kolkata

Subrata Modak
Dept. Of Computer Science
St. Xavier's College, Kolkata

Pallab Kanti Mukherjee
Dept. Of Computer Science
St. Xavier's College, Kolkata

Asoke Nath
Dept. Of Computer Science
St. Xavier's College, Kolkata

Abstract* — *The main aim of Sixth Sense Technology has always been the to reduce the distance between our physical world and digital world. This has been done in numerous ways. The purpose of this paper is to present one such ways of bringing close the two worlds. Numerous techniques for color recognition and detection have been proposed during several past years. A few papers have also been published which compares different approaches of color detection and virtual mouse functions. This paper proposes a method of using mouse pointers and all of its functions through hand gestures and color recognition with cheap and minimal hardware components.

Keywords* — *Gesture, Image Processing, Colored markers, Color detection, Object recognition, Sixth Sense, Real time image processing

I. INTRODUCTION

Humans interact with the physical world around him using the five senses but there is also a sixth sense that humans claim to have and felt it on numerous occasions. A large number of science fiction movies have also portrayed the use of sixth sense. Now people live in an era where it is possible to bring it to use in real life. This paper proposes a technology that brings forward an additional sixth sense. A sixth sense is an ESP (Extra Sensory Perception) which will lead to a more humanized future, connecting both the physical and digital world without or at least, with the use of minimal hardware.

Sixth sense technology has been interpreted differently at different times and has used different methods in doing so. The sixth sense technology employs a simple device which is a wearable device made from very simple objects found around. This device is actually a camera and a computing device such as a cellular phone, computer, or a laptop designed so as to understand and work on gestures, to be more specific hand gestures. Additionally there can be a mini or pocket projector, which will eliminate the need for having any monitor, or display device as it can create its own touch screens from any semi-rough surface for various applications.

The origin of sixth sense technology came into existence with the help of Steve Mann who implemented a wearable computer via one kind of neck-projector called as head-mounted projector that was coupled with a camera system which can be traced back as early as 1990. Later, following his idea, Pranav Mistry, who was a young research scientist at MIT at that point of time came up with new applications of this technology. Thus, it can be said that the sixth sense technology was developed at Media labs in MIT as Pranav Mistry came up with the name 'Sixth Sense Technology" and has since been named Wear Ur World (WUW).

The device has numerous applications such as the drawing with the movement of the index finger which can be done on any surface. The other applications include newspaper, checking time by drawing a wrist watch, mapping, reading, taking pictures, playing virtual pong games etc. The device is not only portable, but also serves as a computer itself without the large hardware and saves a lot of time spent on surfing information.
This sixth sense technology gives one the freedom of interaction of data and information between the digital world and the physical world using hand gestures. This technology also has a wide variety of application in the field of artificial intelligence as well. This technology can increase the efficiency of robots to a great extent that will be also able to interact with humans in near future.

A great deal of computer research and image processing tasks are dedicated to the implementation of systems that is capable of detecting user movements such as facial gestures, hand gestures, cheek movements and eye movements.

In many cases, such systems are created with specific goals aimed at compensating lost senses of people with disabilities or limited motor skills by means of computer systems. However, these systems are not cost-effective. The motivation here is to design an inexpensive alternative to interact with the digital world (primarily computer applications) in a simplistic way as they interact in the physical world.

In the present world, most of the mobile phones as well as the smart phones are using touch-screen technology as the primary means of communication but it is still not cheap enough for implementations in desktops or laptops. Creating a virtual human-computer interaction device such as mouse or keyboard using minimal hardware such as a webcam, computer vision techniques and few color markers can be an alternative way for the touch screen. Our objective for this paper is to implement Sixth-sense technology to create a virtual mouse system using Web camera and color markers to interact with the computer so as to bring closer the physical world with that of digital world. This mouse system shall be able to control all tasks that a mouse does, that includes right click, left click, double clicks, mouse movements, mouse press, mouse release and scrolling. Several image processing algorithms have been implemented in the proposed system. The proposed system also aims to be portable, Cost-effective and shall be able to access data directly from real time devices.

II. RELATED WORKS

A camera and computer vision technology have been used for gesture recognition and image segmentation by Erden et al[1]. Mouse clicks were implemented by defining a screen in which when a user's hand passed over the defined region, the click event took place [1]. Hojoon Park [2] has used computer vision technology to implement mouse movements which is an inspiration for this paper. Many researchers in robotics fields have researched and formulated several methods to control mouse movement using video devices such as web camera. However, their implementation and algorithm were different. The way they implemented clicking events were also different.

Chu-Feng Lien [3] used another approach to control the mouse cursor and the clicking events using only the finger-tips. The method was based on image density. The user had to hold the gesture long enough at the desired spot for a short period of time to make the clicking events occur. There is another work of LI WENSHENG where he controlled various mouse functions and movements of the cursor by Computer vision technology and Web camera. The mouse cursor was controlled by fingertips and the clicking actions were controlled by the angle between the thumb and index finger. Finally, the mouse pointer or the cursor was controlled by the fingertip.

Pranav Mistry has taken the notion of sixth sense technology to a whole new level. He has used color recognition, edge detection, and color tracking and gesture tracking algorithms along with sound recognition and co-ordinate detection in three dimensions to give an output which is very intuitive and with a smaller hardware in pendant form that hangs on the chest of the user like a pendant controlled by a mobile computing device and the output was given by a pocket projector on any semi-rough surface. However, the system needed high speed internet connection to function very intuitively as proposed.

Another inspiration of this paper is the research work of Asanterabi Malima et al. [5] where the behavior of a robot was controlled by a finger counting system. They used an efficient algorithm to get the radius of the region of the hand, and another algorithm for image segregation to get a result that is more efficient. The "AllSee" system which is also a sixth sense device proposed and developed by Bryce Kellogg and other members of that project is a milestone in the road to gesture recognition to connect the physical and the digital world.[4].

III. INTRODUCTION TO THE SYSTEM

In our work, we have tried to implement all mouse functions using only a camera and few color markers based on color recognition technique. The computing device is primarily a windows phone or laptop, though there is a scope of implementation using Android or IOS as well. Here the procedure of our work is simple. Real time video is captured using a camera which, in this case, is a Web-camera. The color markers are worn by user on the fingertips. The camera captures the video whose individual frames are processed separately. The colors are detected using image subtraction technique used in the proposed algorithm. After the detection of the colors and removal of additional noises present in the image initially, the fingertips works as mouse. All the mouse movements are then controlled by user using gestures. The color markers and the web camera are the only required hardware in the system apart from the computing device. Additionally, a pocket projector or mini-projector can be connected to the system to further eliminate the requirement of monitors or other display device and can use any semi-rough surface as the display screen. This furthers reduces hardware complexity.

IV. SYSTEM DESCRIPTION

Object recognition and the tracking of the color markers are the primary concern for the system. Real time video is obtained from the associated web camera in the system. Individual frames are then processed by the system. The frame thus obtained will have the components like a mirror image, so the image frame shall be flipped. This needs to be done with all the acquired frames.

The image is then converted to a grayscale image. The next step will consist of the primary part of the system, i.e., color detection. RGB components are obtained from the grayscale image obtained earlier. The image is then converted into a binary image from which the centroid and the bounding boxes are calculated to find the regions of the image. The co-ordinates acquired from the calculation of the centroid. Finally the implementation of the mouse events like Left click, Right click, Double click, Scroll, Mouse press and Mouse release are performed using the combination of the colors of the color pointers.

The steps stated above are broadly described in this section. The system consists of the following steps to work as proposed;

A. *Block Diagram of the Proposed system*

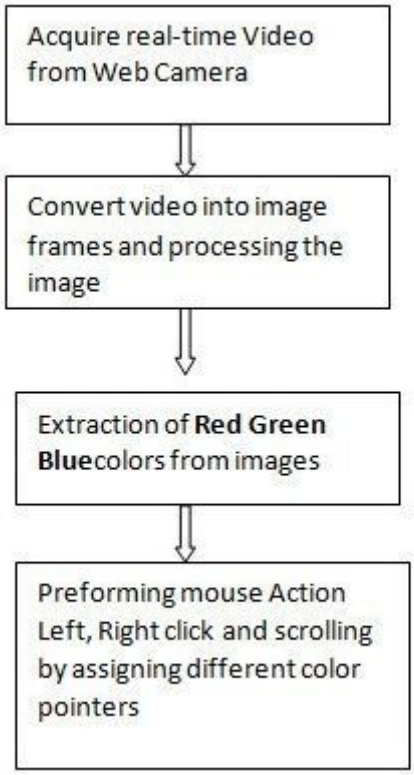

Figure 1: Block Diagram of the proposed system

B. *Capturing the video at real-time*

In order to capture the video at real time, a sensor is needed. For the proposed system, the web camera will function as the video capturing sensor. The only thing that needs to be adjusted is the fixed frame resolution which may vary based on the configuration of the camera. Since it is pre-determined the manufacturer, it will not affect the system and even if it does, it will be very easy to adjust that part. Based on the Frames Per Second of the camera, the captured video will be divided into several image frames which will then be utilized for the processing of each frame.

C. *Flipping of the acquired image*

The acquired image is inverted. This implies that the movement of the pointers to the left will make the cursor move to the right if used with the acquired image. This is because the image acquired is similar to the mirror image of the scene it captures. So the flipping of the image becomes necessary. The image is flipped vertically to eliminate this problem. But the flipping actions cannot be directly performed on an RGB image, so the individual color channels are separated and each channel is flipped separately. After the red, green and blue colored channels are flipped separately, they are concatenated. As a result of this, a flipped RGB image is obtained at the end of this step [6].

D. *Conversion of the flipped real time image into grayscale image*

The computational complexity is very high in RGB images as compared to the grayscale image. So to make the computation feasible and the system faster, the flipped image obtained at the end of the previous step is converted into grayscale image. After converting the image into grayscale, all the required operations are performed to facilitate an intuitive output[6].

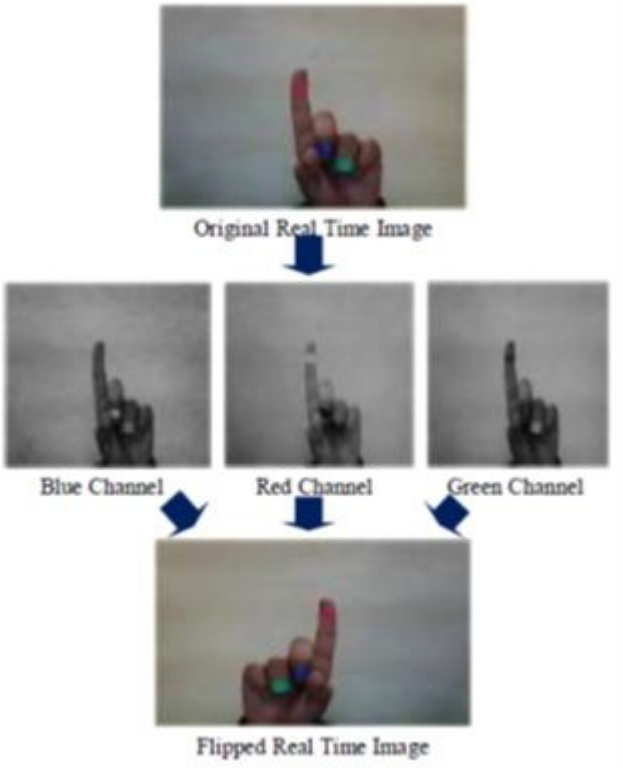

Figure 2: Flipping of a real-time image

E. *Recognition of the colored components*

The RGB colors, i.e., the Red, Green, and Blue colored components are detected at the end of this step. This is done by the subtraction technique stated earlier. The color suppressed channels of the flipped image are subtracted from the flipped grayscale image obtained in the previous step. The output of this step is an image consisting of the object that is detected as a patch of grey tones surrounded by a blank space, usually a black region [7].

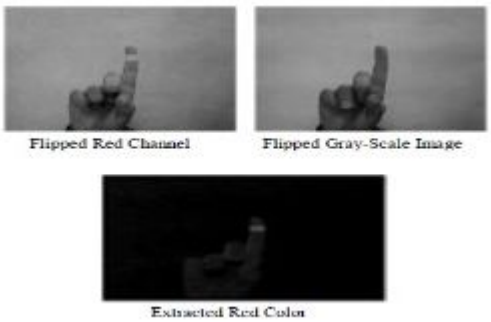

Figure 3: Detection of Red components

F. *Conversion of grayscale image into binary image*

The finding of centroid must be done in an binary image. So, in order to compute the region of the detected object, in this case, the color pointers, the grayscale image is converted to binary image. Since the grayscale image is a matrix containing values in the range 0 to 255, the conversion to binary image becomes simpler. We use a threshold value of 25% , which means that all pixels which have values less than 20% of the maximum pixel are converted to 0, while ones that lies above the threshold value are converted to 1. Here 0 represents pure Black and 1 represents pure White. In the original grayscale image, 0 represents pure Black and 255 represents pure White. The output image at the end of this step is a monochromic image with only Black and White colors. This step becomes necessary because we implement this system in MATLAB and MATLAB can find properties of an image from a monochromatic image only.

Figure 4: Detected region associated with mouse cursor from the color markers

G. *Finding and plotting centroid of the object*

In this step, we implement the mouse pointer that is the base of implementing mouse functions. Now the mouse pointer can be plotted on the screen at some point having co-ordinate value. This value must be sent to the cursor to be positioned accordingly synchronized with the detected object. Once it has been implemented, the mouse pointer can be controlled by the system. We perform this using an inbuilt MATLAB function whose output is a matrix consisting of the vertical(X) and horizontal(Y) value of the co-ordinate. These co-ordinates value changes with time and thus fast processing is required for intuitive view of the output. The co-ordinate values are stored in a variable for further use in the processing.

H. *Tracking the implemented mouse pointer*

The co-ordinates are sent to the cursor after accessing the mouse drivers and on receiving the values of the co-ordinates, the cursor positions itself on the screen at the received co-ordinates. With the change of time, new centroid is calculated, located using the previous step, and sent to the cursor again. The cursor then reposition itself at the new co-ordinates. This continues and the user gets the illusion of a continuous moving cursor. As the user moves his hand in front of the camera, the cursor moves on the screen accordingly. There is no inbuilt function to perform this action. But MATLAB supports integration with other languages like C, C++ or JAVA. In this system, we use Java Robot to perform this operation.

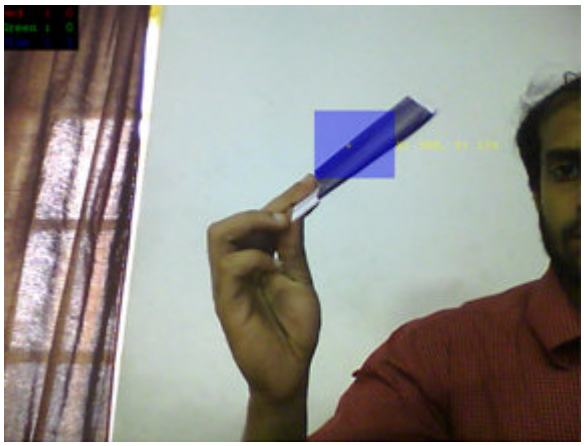

Figure 5: Tracked Blue Color

I. *Implementation of mouse events*

Using Java Robot, we control the flags associated with mouse buttons. The control actions are initiated by the hand gestures by the user. The computation and hence the computational complexity is reduced to a great extent because of the use of the colored markers. Background noise and illumination conditions, although, affects the intuitiveness and accuracy of the system, but it is reduced to a large extent. The colors can be customized based on user requirements.
In this system, we use the following gestures to implement the mouse events:

Mouse movement is based on the detection of blue color. On tracking the movement of the blue colored marker, the cursor moves according to the movement of the blue colored marker.

Other mouse events are based on the number of a particular color that has been detected. If the detected color is green, then, the system further counts the number of green colored markers detected. If the number of detected green colored marker is 1, then left click operation is performed. If the number of detected green colored markers is 2, then right click operation is performed. If the number of detected green colored markers is 3, then double click operation is performed.

If the detected color is red, then the red colored marker is tracked and scrolling operation is performed. Mouse Press and Mouse release operations are based on the simultaneous detection of all three colors. If all three colors are detected together, then Mouse press operation is performed. If Blue and Green colors are detected simultaneously, then Mouse release operation is performed. In other words, if all three colored markers are pointed to the screen together, the mouse press event will occur. To release, the user just needs to remove the red colored markers from the area of detection.

V. SIXTHSENSE MOUSE ALGORITHM

Step1: *Initialize the thresholds for each R, G, B components.*
Step2: *Input: Real time video and convert the videos into individual image frames.*
Step3: *Flip each image frames.*
Step4: *Convert each frame into a gray scale image.*
Step5: *Color detection and extract the Red Green Blue colors.*
Step6: *Convert the image into binary image.*
Step7: *Calculate the centroid of detected color of the image frame.*
Step8: *Track mouse pointer using co-ordinate of the centroid.*

If detected color is ***red***
Then perform ***Scrolling***
Else if detected color is ***blue***
Then perform ***mouse movement***
Else if detected color is ***green*** *then*
If number of ***green is*** *1 then*
Perform ***left click***
if number of ***green*** *is 2 then*
Perform ***right click***
if number ***green is 3*** *then*
Perform ***double click.***
Else
if detected color is ***green*** *,* ***blue*** *and* ***red*** *then*
Perform ***Mouse Press***
Else if detected color is ***green*** *and* ***blue*** *then*
Perform ***Mouse release***
End

VI. PROBLEMS WITH THE SYSTEM

Though there are many advantages with this efficient sixth sense mouse, there are certain problems as well. Since the system is based on image recognition, the illumination factor will affect the performance of the system. The illumination factor cannot be eliminated. If there are other Red, Green, or Blue hues in the background which falls within the detected threshold, then the system will give erroneous output. But this problem can be eliminated by adjusting the threshold values but the output will be more intuitive if the background does not contain the colors used in the system for detection of objects. It is also favorable for the background to be light. The system computes certain complex calculations internally, therefore, with inadequate processing speed, the system may run slower. However, the requirements are not very high. Therefore, the system will run optimally on standard laptops or Computers. The implementation of the system can be slower on smartphones.

VII. FUTURE SCOPE

The system has several future scopes. Without a hardware mouse, the running of the computers will be more flexible. The user can stay far from the computer and still be able to operate it. Implementations of several functions are possible. Gestures can be modified and upgraded to an enhanced system with several gestures for image capture, media player control, drawing in any painting application, typing through virtual, or on-screen keyboards. The possibilities are endless. Moreover, adding a pocket projector or mini projector can take away another hardware device from the system. The projector will eliminate the monitor associated with computers for display. It can find any semi-rough surface and project on it to make any semi-rough surface as the display screen.

VIII.CONCLUSION

In this paper, the design and implementation of Sixth sense mouse is presented and developed using Webcam and colored markers. An algorithm has been presented and its working is detailed thoroughly. Since the updating possibilities are endless, updating the system has been kept as a future scope. The built device is cheap, and is easy to carry from one place to another. The addition of the pocket projector will make it easier to carry further. The limitation of the hardware being associated with a system has been reduced to a great extent. As an end thought, the system will prevent us from being machines sitting in front of another machine. It will allow us to stay more human.

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] Hojoon Park, "A Method For Controlling The MouseMovement using a Real Time Camera", 2008, Brown University,Providence ,RI ,USA, Department of computer science
[3] 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
[4] UW team works on gesture recognition technology *http://allsee.cs.washington.edu/files/allsee.pdf*
[5] Asanterabi Malima, Erol Ozgur, and Mujdat Cetin, *A Fast Algorithm for Vision-Based Hand Gesture Recognition for Robot Control*
[6] Rafael C. Gonzalez and Richard E. Woods, *Digital Image Processing*, 2nd edition, Prentice Hall, Upper Saddle River, New Jersey, 07458
[7] Cloud Computing., Retrieved at 03. June 2011 [http://fclose.com/b/cloud-computing/article/mrcc-a-distributed-ccompiler-system-on-mapreduce/]
[8] M. I. Khalil, 2010, "Red Color Recognition Using the Template Matching Method", International Journal of Computer Theory and Engineering, Vol. 2, No. 5, pp 683-687.
[9] S. Du, M. Ibrahim, M. Shehata and W. Badawy, 2013, "Image Recognition: A State of the Art Review", IEEE Transactions on Circuits And Systems For Video Technology, Vol. 23, No. 2, pp. 311-325.
[10] L. S. Chang, S. L. Chen, C. Y. Chung and W. S Chen, 2004, "Image Recognition", IEEE Transactions on Intelligent Transaction Systems, Vol. 5, No.1, pp. 42-57.
[11] R. Kate, 2012, "Color Recognition Using Segmentation", International Journal of Engineering Research and Technology (IJERT), Vol. 1, No. 9.