

Converting Objects in Physical World into Digital World Using Color Recognition

V. Prasanna and S. Gopinath

Abstract---*Gestures do speak well than commands. We usually interact computers with GUI based HID devices. It would be more realistic if the Holy Grail gesture based natural interface enables illiterates even to augment machine usability. This evolution shouldn't be an overhead to the consumerism. Also economical constraint should not force this to be inefficient due to productivity cost. In this paper, we proposed a new genre of interacting with computational devices and our goal is to make the user interface more intuitive. Rather using the mouse and touch pads to achieve the tasks like moving, left & right clicks of the mouse pointer, with our new algorithm you can achieve that with your own color marker tapped finger. In this beginning version of this algorithm we used color markers to reduce false detection of the user finger. With our algorithm it produces a high degree of accuracy in color recognition and make your own finger tapped with the recognized color marker to function as like a computer mouse. Hence such an idea of making the existing wheel to be fit into the new body seems good. Thus by integrating various efficient pragmatics into a new paradigm, syntactically this system is proposed.*

Keywords--- *Color Recognition Technique, Thresholding, Sixth Sense Technology*

I. INTRODUCTION

WE came across lot of computing devices in our day-to-day life with the incredible development in computing and mobile platform. Most probably the way we manipulate such devices may almost same for the current era. We use a mouse, a touch pad, voice input to interact with your computing machine, say a computer or smart phone. We introduce a new genre of user interface, with those computing devices where you don't need to use your physical component, mouse. Rather, you can use your own finger to get the functionality of mouse. In a bid, you can use your own finger as a mouse, indeed the gesture action is reflected as mouse. Here what we use is color recognition, Feature extraction, and a little bit computing algorithms to achieve this. It was used by sixth sense devices [14], [15] previously in 2009. An idea of making the natural and more physical things

to be merged with computers will make human to be more connected with digital devices [1-4]. It also may be useful for disable persons to interact with and make use of computational devices [11], [13]. Here we use the incumbent technology and resources to make the usual user interface much better and more intuitive. With this integration, a new genre to interact with computers is introduced.

II. LITERATURE SURVEY

This paper demonstrates a new genre of interacting with computational devices as model for moving the mouse pointer using finger tapped colour markers. The algorithm presented here is more unique in recognizing the colour of the objects holding in your hand, irrespective of the ambient. These make humans to be connected more with computational devices using your finger, rather using head, eyes, face [14-17] and this way our approach differs from existing models. This provides a better understanding for disables to interact with computers [11], [13]. The smart way of interacting and inputting can be done with this kind of human interactivity [16-18]. Just by looking at an object and asking the computer to do some task over that object were deployed as blink bots [18]. This was deployed more realistic in sixth sense technologies [14], [15]. You can copy text from physical world and paint into the digital devices using a simple gesture of tapping over the region of interest [16]. Related works on this regard uses some tools like iMEMS devices, CCD cameras, hand gestures, finger tips, head movements, eye movements, gyro sensors, teeth, facial expressions and face for moving the mouse pointer of a computer [1-12]. Also when comparing with incumbent models like gesture based mouse move using MEMS devices and Gyro sensor based models, the proposed idea maintains economy considerable. It does not need any new resources to make this, except a camera module [11],[13].

III. BLOCK DIAGRAM

The visual information is fed as input to image acquisition hardware, typically a Camera. In our visual data, it is an image or a sequence of images having the particular object what we use to call as physical objects .we are going to extract those object from that input image. In our case we doesn't need to record it, rather than we just going to process each frames captured at that particular instant by the camera. We are using a camera with 15fps, so that we need to process all the 15 frames (2D images [19]) in order to extract that particular physical object. Here the proposed model undergoes Color Recognition algorithm to extract the region of interest (ROI)[19]. Color Recognition is done using the concept of

V. Prasanna, Student, Electronics and Communication Engineering, St. Joseph's College of Engineering & Technology, Thanjavur, India. E-mail: balaprasannav2009@gmail.com

S. Gopinath, Assistant proessor, Electronics and communication Engineering, St. Joseph's College of Engineering & Technology, Thanjavur, India. E-mail: s.gopinath88@gmail.com

Thresholding [19], which is pixel operation. Threshold level has to be defined by the designer for the colors which he deserves. Based on that threshold values, thresholding function does filtering [19] operation and allows that some pixels to be exist in the output binary image. At the end of thresholding block, we got a binary image consisting of the information about that particular physical object. Once we have a true information about the physical object by thresholding [19], its time to redefine the shape of the extracted information.

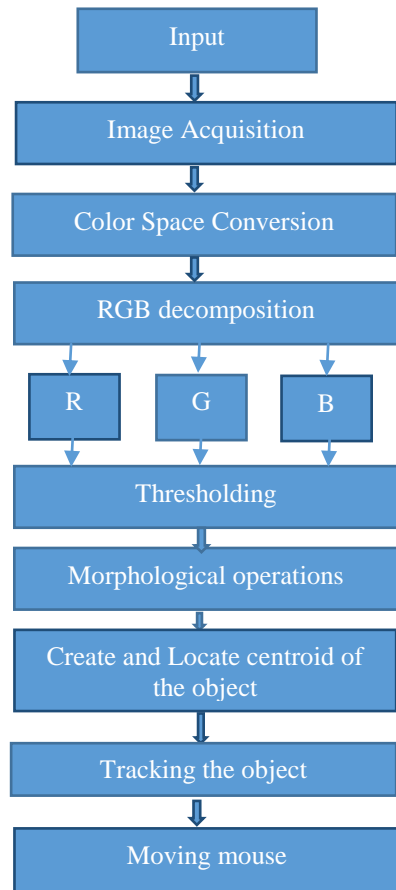


Figure 1: Block Diagram

Typically, it uses morphological operations [21] such as erode, dilate, opening and closing [19],[21]. After extracting the physical object we have to take the centroid value of it, for our application. Locating the Centroid of that object will help us to track the object continuously which leads us to assign some task to that object. With a successful integration between physical and digital world, if a change is made in physical world, then, it will be reflected back in digital world. In our case, we use the object in physical world to move the mouse pointer in digital world. At this stage what we got is that, we integrate the object or things in physical world with the digital world, normally with computer [14-17]. This is the key idea confined with this project and the block diagram is shown in Fig 1.

IV. IMAGE ACQUISITION

Image acquisition is the digitization and storage of an image. In order to process the image, we should acquire the

images with the help of image acquisition hardware, a camera which act as a digital eye to the image processing system [20]. In our model, we used the incumbent resource in a laptop. 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. However, if the image has not been acquired satisfactorily then the intended tasks may not be achievable, even with the aid of some form of image enhancement [19]. In our case, we make use of the digital eye in a portable computer.

V. IMAGE PROCESSING

We humans, having the power of vision with our eyes, which is capable of acquiring continuous video, which is being stored for later retrieval and to be processed within unimaginable time. An intelligent processing and controlling system is our human brain, handling the image processing, feature extraction and lot of complex algorithms, yet even not to be modeled. Based on the processed data it handles some events and tasks. With this scenario, a different genre of modeling Image Processing is provided here. As like the human Eye, the camera which act as a digital eye to image processing system. The brain, which can't even modeled with the incumbent image processing hardware and the incredible software algorithms. As like the connections and links between eye and the brain, here we are having lots of protocols which make us to augment the image processing into a new level called Intelligent Image Processing, The EyeTap [20]. In our proposal, we used all the three basic terms Camera, Image Processing Hardware and Intelligent Software.

VI. COLOR RECOGNITION

Recognizing a color is a complex task. While recognizing a certain color in a scene, its background information is always trying to affect our region of interest [21]. Another one issue is that, it always depends on the background illumination and luminance, chrominance of the scene [21]. Upon overcoming these kinds of issue we got a perfect color recognized module with our threshold values. This module can be recognized by any system with a little bid pre installation, so that it worthwhile independent of background information. Also the second issue of luminance of the light source will not affect this module. Because I chose to recognize the color of an object which is having the property of showing its most extreme value of its color component when it is exposed to luminance of high. And in normal state its color can be easily differentiated with other. This means that it is having a unique value in normal state and also that peculiar value still standalone even when more a bright light is applied aboard it. Here, I am pointing that, not all part of the object is to be extracted. Because the goal is not like that. The goal was to connect the two worlds which I mentioned earlier. Even a single pixel which read from the object is more than enough to augment this connectivity. But as I pointed before we have to reduce the false detection of that object.

VII. SIGNAL FLOW OF PROPOSED ALGORITHM

A. Step 1

We are able to perceive billions of colors [21]. Obviously how a computer can?. The camera, which act as a digital eye fills this gap, but it depends upon the resolution of the camera and also the number of sensor array used, in the case of CMOS type. So it is necessary to connect a camera module with the computing device for which you wish to give a sense of vision. By simply connecting a USB type webcam will not fulfill our requirement. For that we need to initialize the drivers for that webcam and also the important thing is to integrate it with the software package we are suppose to use for processing the digital eyes data. Here in our model we use MATLAB[19] software as a software package for processing the data from the digital eye, but it is also possible with some other software packages like Scilab, openCV, etc. Now-a-days android codes are directly accessing a camera module and also apps developers are more interested to write their codes to process the images in more real time as like GOOGLE GOGGLES was made.

B. Step 2

For a successful integration, we may create a video object which help us to proceed with further processing of the data from the digital eye. Creating a video object in MATLAB [19] will be done with the help of integration of windows winvideo function with MATLAB. The following command will help us to create a video object using (1).

Video=videoinput (winvideo, 1)

Here in the above command, Video is a variable name of the video object and the index number 1 represents the identity of camera1. It can supports up to 'n' numbers. We can also set the frames per second of the camera. To view the video output in the user preferred size (window size) we use the following command.

Preview (Video)

After a successful integration and preview of the video, its time to snap the images from the real time video data. To get a snap of the present video data, use the get snapshot command along with the respective video object.



Figure 2: Output of Get Snapshot ()

C. Step 3

Color space conversion is much needed for processing digital images. For applications regarding color recognition its much better to use RGB [19] color space format. RGB [19] is a basic type of color image which comprises of three components namely R, G, and B (grayscale type). If your camera acquires images in YCbCr [19] color space, it is necessary to converted into RGB format for our algorithm to work by the image processing software. Once we have converted into RGB [19] image, then store it in a variable for further processing and feature extraction concepts to be applied.



Figure 3: RGB Image Format Converted from Ycbcr Format

D. Step 4

Next to color space conversion, we need to decompose the RGB color image into three different gray scale images, which contains only Red component, Green component, and Blue component respectively. Store the entire three images in 3 variables. Consider that our variables are r, g, b. Apply thresholding to all the three variables separately by properly analyzing the transfer function of the thresholding. Here the transfer function indicates the filter's cutoff value. In our proposal, we recognized the red color with the following pixel cutoff values as the transfer function and filter cutoff which are given below,

$$Tr=(r \geq 185) \& (r \leq 255) \dots [19]$$

$$Tg=(g \geq 45) \& (g \leq 98) \dots [19]$$

$$Tb=(b \geq 30) \& (b \leq 80) \dots [19]$$

In the above equation r, g, b are grayscale images. And Tr, Tg, Tb are variable name of binary images which comprises of only black and white pixels. This will clearly explains us, that, where ever there is a white pixel, it ensures that the presence of our recognized object.



Figure 4: R Component Gray Scale Image



Figure 5: G Component Gray Scale Image



Figure 6: B Component Gray Scale Image

E. Step 5

At the end of step 4, we are having 3 binary images named Tr, Tg, Tb, which are now being converted into a single binary image using &(logical AND) operation. This helps us to filter some false recognition of unwanted colors. Finally as a result of AND operation you will be getting one single binary image which partially describes our object (color of the object which we used to recognize, here in our proposal it is red color). If same object is present at two places in an image, how our system should smartly understand that? This problem can be solved by simply labeling (bwlabeled, [19] is the command used for labeling a binary image) the final binary image. This labeling concept will be discussed in further steps.

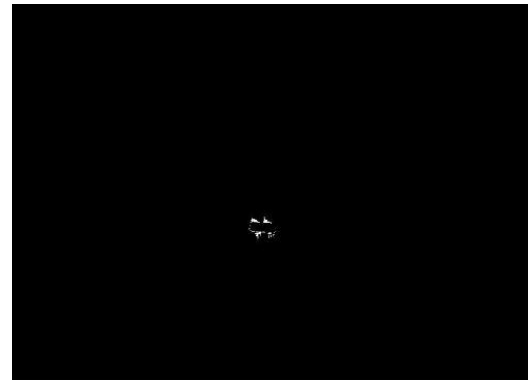


Figure 7: Separately Thresholded and And Operated Binary Image

F. Step 6

Once we extracted the object roughly, we have to give more intelligence to our system to redefine the size of the object which is extracted so called TDD object. So we used morphological techniques like dilate, erode, opening, closing, holes filling, and the like. Upon doing this kind of morphological operation we are in need of using a structural element which is going to override our extracted binary image (our object) to redefine and enhances the TDD object in size, contour, boundary information etc. In our proposal we used a structural element of type 'disk' of size '10'. Disk is round shaped structural element with the second parameter drives its size.

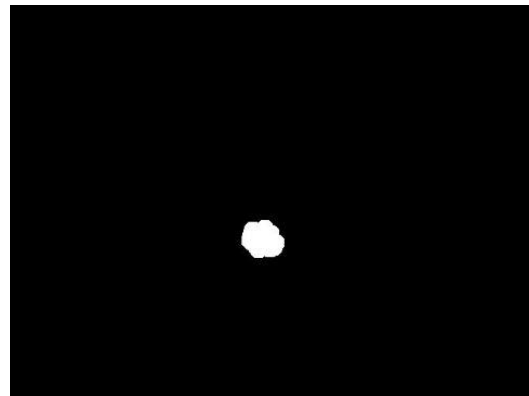


Figure 8: Morphological Operated Output of the Threshold Image (TDD Object)

G. Step 7

Labeling is necessary to count, same number of color objects extracted. For example, if we are having two white pixel regions without connectivity in the final binary image, how the system knows that there are two objects? We humans, and so we can perceive that quantity by the combined action of vision and leverage. But how a system can understand that? The answer for these complex questions is typified by the concept of labeling. In labeling, we can find a connected region of white pixels by pixel operation. Once we have identified the connected pixels and its boundary, it is being stored by a variable at its front which shows the number of objects available in our input image. Here, the object represents our recognized color of physical object.

H. Step 8

Once we perfectly recognized the color of the physical object and converted it into digital object, its time to use it for some applications. In our model, we moved the mouse pointer in a computer with the help of this recognized object. For this application, we need to track the objects motion and to define a black box which move the mouse based upon the tracked location values of the physical object. Here, the black box represents a function, which needs to be defined to move the mouse pointer based on a parameter from the tracking module. The code, which tracks the location of the physical object is constantly sending data to the next function, which moves the mouse pointer based on that received value. For object tracking, we used the concept of getting its (x,y) location in image (X cross Y, x is dimension in x direction of the computing device, we used to apply this algorithm and similarly for Y also). Typically in laptops, we may have X direction with dimension of 1200 and Y direction with dimension of 800. This is useful for us to track the object in the following pixel location.

I. Step 9

With the tracked values of our real world object, we now can able to move the mouse pointer with a help of function called Robot in java. By properly importing this function into the MATLAB environment and redefine that function to integrate with our tracking code, we can move the mouse pointer. This is showed in the output images.

VIII. FEATURE EXTRACTION

After recognizing the color of our object we need to extract some information about the recognized one. We have to reduce the dimensionality of the object to give shape to our virtual object which originates in physical world .so we have to go for feature extraction principles with the help of morphological tasks. Our goal is to augment these two worlds. In the proposed idea, it is targeted over the pointing device. For such precision, dimensionality reduction of the recognized object is necessary without de-qualifying the object. This phase reduces the recognized object to circular meta object to a task defined dimension (TDD).

IX. DESCRIPTION OF TDD OBJECT

TDD object effects from morphological techniques such erosion, dilation, open, close methods. With the TDD object, we have to assign it to a ROBOT function to handle the task of moving the mouse pointer. In our case, we used it to move the mouse pointer. This is the task we assigned to the TDD object, a virtual object created from our recognized image. We used the concept of descriptors to label the TDD object for later use [21].

X. TASK ASSIGNED TO THE OBJECT

We can do some tasks with the recognized and perfectly extracted object .we have to assign some function for that object. In this experiment, we make the object to act as mouse pointer. Typically we have to hack the functionality of mouse pointer and overload our object with that task, with the help of

simple java native libraries (java.awt). Indeed, on behalf of our application we use java syntax, not reinventing the wheel. We just make use of the incumbent components and make the user interface more intuitive. Finally the goal was achieved with an augmented connectivity between physical and digital world [20]. It's proved that, when you change the state of the object in physical world, that is reflected back into digital world. By properly tracking and extracting the location parameters from the Meta objects, and then passing it to a mouse move function, will now move the mouse as expected. In our case, if we move the object up, down, left or right, based on that, the mouse pointer is getting moved.

XI. RESULT

The output of our proposal is shown below. Here the red colored object is the one which we recognized. The mouse pointer is getting moved based on our physical object's location. These images are snapped version of the recorded output.

The following figures represented the simulated output of the proposed model.



Figure 9: Final Output of Moving UP

Fig 9 shows that, the red colored object is on the top and also based on that mouse pointer is in the top of the working screen.



Figure 10: Output for Moving Down

Fig 10 shows that red colored object is on the bottom and also based on that, the mouse pointer is in the bottom of the working screen.



Figure 11: Output for Holding in Center

Fig 11. Shows that red colored object is in the center and also based on that mouse pointer is at the center position.



Figure 12: Output for Moving Left Side

Fig 12. Shows that red colored object is in the left side and also based on that mouse pointer moves to the left.



Figure 13: Output for Moving Right Side

Fig 13. Shows that red colored object is in the right side and also based on that mouse pointer moves to the right.

XII. CONCLUSION

Our proposal concludes with a key idea that, recognizing the colour of an object is complex task, because of the previously mentioned issues. We provide a robust recognition value of an object of red colour, so that it is providing 99% accuracy of recognizing the red colour. In the area of colour recognition, the accuracy is one of the main factor to be

considered. In our application, we generated as a Meta object, which is seamlessly be in connection with the real world object, and used it for moving the cursor. Finally we fulfilled our goal of getting connected the two worlds, namely physical and digital world through colour recognition.

XIII. FUTURE WORK

With the incredible image processing algorithms, we can do enormous number of application. Here, we use color recognition concept. This can be extended to give touch sense feel to a non-touch device, by placing the camera inclined with line of interaction with the computational device's display. This gesture recognition will leads to a new genre of virtual input devices, for example we can put password to any authentication system, just in air. Finally we conclude with the importance of accuracy in color recognition with a quote "Increasing the accuracy of color recognition, will square the probability of intuitive application in lot of field by means of integrating with that field..."

REFERENCES

- [1] E. Ul Haq, S.J.H.pirzada, M.W Biag and Hyunchal Shin "A new hand gesture recognition method for mouse operation", The 5th International Midwest symposium in circuits and systems IEEE, (MWSCAS), Pp. 1 – 4, 2011.
- [2] A.Pandit, D.Dand, S.Mehta, S.Sabesan and A.Daferly,"A simple wearable hand gesture recognition device using iMEMS", International Conference on soft computing and recognition (SOCPAR'09), Pp., 2009.
- [3] K Manchanda and B Bing, "Advanced mouse using trajectory-based gesture recognition", proceeding of the IEEE southeastcon (southeastcon), Pp. 412-415, 2010.
- [4] S Kumar, A Agarwal, A Rai and N Bachani,"Vision based human interaction system for disable", 2nd international conference on Image processing theory tools and application (IPTA), Pp. 441-446, 2010.
- [5] Young wook kim, "Development of headset-type computer mouse using gyro sensors for the handicapped", Electronics letters, volume 38, issue 22, Pp. 1313 – 1314, 2002.
- [6] A W Mohamed and R Koggalage,"Control of mouse movements using human facial expression", 3rd international conference on Information and Automation for sustainability(ICI AFS),Pp. 13-18, 2007.
- [7] T , Palreja, E Rubion, M Teixido, M Tresanchez, A F Delviso, C Rebate and J Palacin, "Simple and robust implementation of a relative virtual mouse controlled by head movements", Conference on Human System interactions, Pp. 221 – 224, 2008.
- [8] Dong Hyun yoo and Myung Jin Chung,"Eye-mouse under large head movements for human-computer interface", IEEE international conference on Robotics and Automation (ICRA'04), volume 1, Pp. 237 – 242, 2004.
- [9] D Ionescu, "Gesture control and the new and intelligent Man-Machine interface", 6th international symposium on applied computational intelligence and informatics (SACI), Pp. 291, 2011.
- [10] Jilin Tu ,T Huan and Hai Tao, "face as mouse through visual face tracking ", 2nd Canadian conference on computer and robot vision, Pp. 339 – 346, 2005.
- [11] M Betke , J Gips and P Fleming, "the camera mouse : visual tracking of body features to provide computer access for people with severe disabilities", IEEE transactions on Neural systems and Rehabilitation Engineering ,Volume 10, Issue 1, Pp. 1 – 10 ,
- [12] Tsang W M and Kong – Pang Pun,"A finger – tracking virtual mouse realized in an embedded system", International symposium on Intelligent signal processing and communication system (ISPACS), Pp. 781 – 784 , 2005.
- [13] M Atyabi, M S K Hosseini and M Mokhtari, "A webcam mouse: visual 3D tracking of body features to provide computer access for people with severe disabilities", Annual IEEE conference India, Pp. 1 – 6 ,2006.

- [14] P mistry and p maes,” sixth sense –A wearable Gestural interface”, proceedings of SIGGRAPH Asia 2009, Emerging technologies, yohohama, Japan, 2009.
- [15] P Mistry, “the thrilling potential of sixth sense technology”, TED India 2009, Mysore, India, 2009.
- [16] P mistry, S C Nanayakkara and P Maes, “Touch and Copy, Touch and paste”, international conference on Human Factors in Computing interactivity (CHI 2012), 2012.
- [17] Pranav mistry and pattie Maes, “Mouseless”, proceedings of UIST2010, poster NY, USA, 2010.
- [18] Pranav mistry, kentaro ishii, masahiko inami and takeo igarashi, ”BlinkBot- look at, Blink and Move”, proceeding of UIST2010, demo- NY,USA,2010.
- [19] C Rafael Gonzalez, E Richard Woods and L Steven Eddins, ”Digital Image Processing using MATLAB”, by Pearson Education, Inc. ,2004
- [20] Steve Mann, “INTELLIGENT IMAGE PROCESSING”, by A john Wiley & sons, Inc., publication, 2002.
- [21] C Rafael Gonzalez, E Richard Woods and L Steven Eddins, ”Digital Image Processing”, by Pearson Education, Inc. ,2004

V. Prasanna I am V Prasanna, final year student, pursuing Bachelor degree in Electronics and Communication Engineering discipline, at St. Joseph's College of Engineering and Technology, tanjore, Tamilnadu, India. My field of interest are machine vision, computer vision, image processing, and human-machine interface, sixthsense technology. (E-mail:balaprasannav2009@gmail.com)

S. Gopinath I am gopinath, pursued my BE degree in College of Engineering Guindy, Anna UniversityChennai, India. Also ME degree in College of Engineering Guindy, Anna UniversityChennai, India. Presently working as an assistant professor in St. Joseph's College of Engineering and Technology, tanjore, Tamilnadu, India. (E-mail:s.gopinath88@gmail.com)