A Novel Design of an Intangible Hand Gesture Controlled Computer Mouse using Vision Based Image Processing

Rokhsana Titlee, Ashfaq Ur Rahman, Hasan U. Zaman, and Hafiz Abdur Rahman
Department of Electrical and Computer Engineering, North South University, Dhaka, Bangladesh
rokhsanatitlee@gmail.com, ashfaqnadim@yahoo.com, hasan.zaman@northsouth.edu, hafiz.rahman@northsouth.edu

Abstract— **In this paper we presented an intangible hand gesture based computer mouse control system. Our approach uses a novel skin color segmentation technique to control mouse movement. The system uses morphological operations like structuring elements and blob counting. Our system can remove other skin like objects from the background. One of the key features of the system is its simplicity. We have successfully tested our system for an intangible interface between human hand and computer mouse without any complex processing. Various mouse operations like cursor movements, right click and left click have been performed. The system has been implemented using webcam and MATLAB.**

Keywords— Computer vision, Hand gesture, YCbCr, Skin color detection, blob, structuring element.

I. INTRODUCTION

The technological infusion into our lifestyles is unavoidable and in this 21st century it is happening in the massive form. From office room to our bedroom there could be a range of electronic devices that need commands from us to perform their tasks. People always seek the easiest and natural way. In the beginning, we used to command the devices manually, then using the remote controller, after that, the touch and voice controlled technologies came. Now people are trying to use one of the most natural communication modes of human being, which is gesture [1]. It would be a great matter of comfort and joy if we could control our devices just with body movement as hand gesture [2][3][4]. Communication between human and robot (machine) instinctively and directly is still a challenging task. Using gestures for controlling devices is a technology yet to be exploited. In everyday life, our hand is used widely as a natural human to human interaction by gesturing. Hence, it could be also exploited for the communication between human and machine.

The application is countless: virtual gaming, recognition of sign language, robot controlling, smart house or office, controlling TV, driving vehicles, making young children to interact with devices, robot control, lie detection, augmented reality, etc. Hollywood movies already simulated the idea of hand gesture controlled mega-robots and war machine.

Hand gestures are mainly categorized as static gestures and dynamic gestures [5]. Static gestures make use of the hand shape and feature. Dynamic gestures make use of the hand movements. Former gesture recognition technique uses computer vision, whereas the later uses accelerometer.

Vision based recognition is contactless and most natural. It is becoming popular and many works have been done using this method [6][2][7]. However, there are some difficulties in this approach due to the difference in hand sizes, hand position and orientation and the lighting conditions etc. In most of the works, color pointer, gloves, color stickers or other devices are being used [8][10] which are not flexible for interacting in real-time environment. Moreover, these methods also have consequences like additional cost, lack of availability among common people, more amount of maintenance etc. The intangible interfaces with devices would be a major help for taking care of aged and handicapped people since in future the uses of robots will be feasible in human social environment. It would also help to reduce the disadvantages [9] of the usage of tangible devices like mouse, keyboard etc.

In our work, an intangible interface is designed and implemented using vision based real time dynamic hand gestures to perform mouse operations like moving the mouse pointer, clicking left, clicking right with hand gesture. The 2D hand images of bare hand [3] are captured using the computer web camera and are processed using feature extraction algorithm implemented in MATLAB in real-time, so that a mouse can interact with its user in fraction of a second. Since webcam is easily available device, it has been used as an initial testing device for our system.

II. RELATED WORK

There have been many approaches developed to control mouse movement using video devices with different methods. In [10] the authors have tried to control the mouse pointer using a color detection technique based camera. However, the user needs to wear colored tapes to provide information to the system. In [8] gesture control is performed using the color based segmentation technique. Here red glove is used which makes segmentation easier but the drawback is it adds an extra cost and could be a discomfort to the users. In [11] color pointers have been used. Here, 'Flipping of Image' was needed. Use of color pointer might not be comfortable and it also increases the cost. In [12] the background is very simple, and no dynamic gesture was being performed. We are inspired by a work by Alisha Pradhan and B.B.V.L Deepak [8] where they have designed an interface for mouseless computer handling. However, they have used colored glove for performing the hand gestures. In our

978-1-5386-2307-7/17/$31.00 ©2017 IEEE

work, we have avoided the use of any glove. Rather we used our bare hand and simpler algorithms for controlling the mouse.

III. Proposed Method

Our system for real-time vision-based gesture controlled mouse contains several stages that are explained in the flowchart of Fig. 1.

A. Flowchart

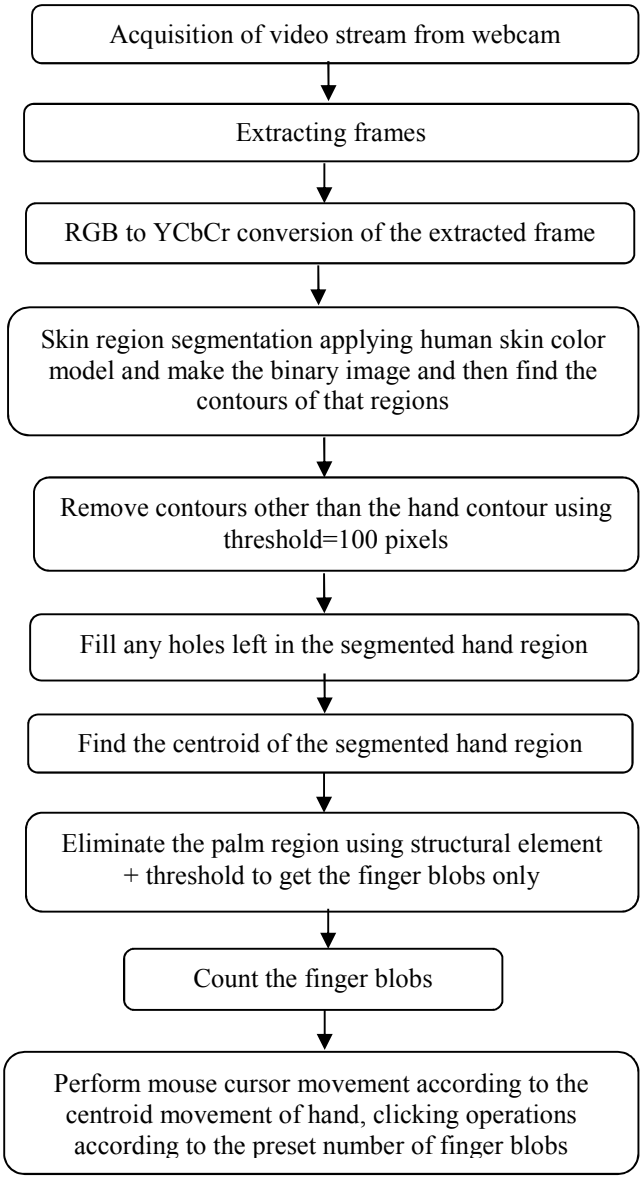

Fig. 1. Flowchart of the proposed method.

B. Hand Segmentation using YCbCr and Skin Color Modeling

One of the most important steps in our work is image segmentation. There are several processes for hand segmentation. We have used the color based segmentation explained below.

YCbCr color space is commonly used in digital video processing and for image and video compression work. It makes the process easy to get rid of some redundant color information. The Y, Cb and Cr values are as follows

$$Y = 0.299R + 0.587G + 0.114B \quad (1)$$
$$Cr = R - Y \quad (2)$$
$$Cb = B - Y \quad (3)$$

Here R, G and B are the red, green and blue components of an RGB image. This color space is popular for skin color modelling due to its transformation simplicity and easier process for separating the luminance component and the chrominance component.

A MATLAB function rgb2ycbcr is used for this conversion. Human skin detection model is a two-class classifier: Skin pixel and non- skin pixel. We assume a pixel $p(x, y)$ in an image I. It will be considered as skin if it follows the conditions below:

The luma component: $pY(x, y) > 80$, and
The blue component: $85 <$ pCb (x,y) < 135, and
The red component: $135 <$ pCr (x,y) < 180

After that we have created a binary image as of same dimension where each pixel's color is either black if it is not considered as a skin color or white if it is considered as a skin color pixel. In Fig. 4, the skin like other objects are being segmented as like a portion of dress of the user.

Fig. 2. RGB image

Fig. 3. YCbCr image

Fig. 4. Skin color segmented image

C. Contour extraction

After getting the binary image, contour extraction is performed to get the boundary information of the desired regions. The different characteristics of a contour are examined and used as features. So, correct extraction of contour produces accurate feature.

D. Region filling

Regions are filled to improve accuracy. This completes the hand portion where due to bad lighting conditions erroneous or bad image of gestures were captured. This actually fills the holes left in the gestures. Here 8-connected filling procedure was

applied. There is a function called "imfill" in MATLAB to accomplish this task.

E. Smaller regions reduction

The object of interest here is the hand. Since, there might be other objects than hand at the background, a threshold area is defined which approximately matches with the hand area. Other objects with skin color are smaller and counted as a noise. Here the skin color objects are represented as blobs and hand is the biggest blob in this case. In this step a threshold of 100 was used, that removed all the connected components that have pixel size lower than 100, it means remove all the objects that have pixels smaller than 100. As a result, only the biggest object is extracted, which is hand in this case. This uses 8-connected neighbors. The functions "bwareaopen" and "bwconncomp" in MATLAB were used.

F. Calculating the Centroid of the hand region

It is important to find the centroid of the segmented hand region for performing the morphological operations like blob counting and using the structural elements. To find the geometric moments of an $M \times N$ image $i(x, y)$ for the moment of order $(p + q)$, the formula is as follows:

$$\sum_{x=0}^{m}\sum_{y=0}^{n} X^p Y^q I(x,y)$$

m_{00}: Moment of zero order. It is equivalent to the total intensity of the image.
m_{10}: 1st order moment about the X- axis.
m_{01}: 1st order moment about the Y-axis.
Intensity centroid gives the geometric center of the image. It is given by:

$$\bar{X} = \frac{m_{10}}{m_{00}} \qquad \bar{Y} = \frac{m_{01}}{m_{00}}$$

This formula was implemented using the MATLAB procedure "regionprops".

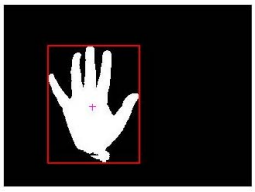

Fig. 5. Centroid of the hand region after performing contour extraction, region filling and smaller regions reduction processes.

G. Morphological Operations

The objective of this section is to remove the palm region from the segmented hand region so that only the fingers remain in the image. This was done using structural elements. MATLAB has various structuring element shapes like diamond, disk, octagon, square, rectangle, cube etc. Here, we have used "disk". Here disk size is dynamically selected upon minimum distance on the contour area from the centroid. Because depending on distance, the hand image exhibited different sizes.

The structural element was calculated as

strel('disk', r), where

$$r = radius = MinimumDistance + \delta$$

Where δ = 50% of minimum distance

Here MinimunDistance was calculated as the distance of the first black pixel from the centroid. We added δ to ensure the blobs created after the operation are separate enough. So, we increased the radius of the structural element by 50%

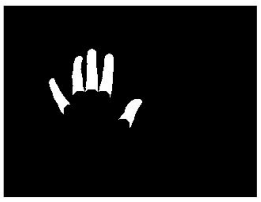

Fig. 6. Palm region removal using Structuring element

H. Blob counting

A blob is a region of an image in which some properties are constant or approximately constant; all the points in a blob are considered in some sense similar to each other. Here, the number of finger blobs in the hand gesture was counted out. For this, the "bwlabel" function of MATLAB was used.

Fig. 7. Blob counting and showing

I. Set hand gestures to control the mouse functions

The mouse functions was performed according to the predefined gestures. Here, it was depended on the number of finger blobs counted in the previous part.

No. of blob = 1, perform movement of the cursor on the screen
No. of blob = 4, perform left click.
No. of blob = 5, perform right click

J. Interfacing and Controlling the mouse pointer

The java class "java.awt.Robot" was used to access the mouse driver after calculating the center coordinates of hand. The mouse cursor moved according to these coordinates. The MATLAB function "set(0,'PointerLocation',[x,y])" was used for the moving operation. The clicking operations were performed according to the blob counting. Resolution of camera was also an important issue, since it was directly proportional to the resolution of mouse pointer. In our project, the resolution of the input image was 320x240 and the resolution of the laptop monitor was 1280x800.

IV. RESULTS

The system was tested indoor (classroom and reading room) with good lighting condition during day and night. The distance

of the hand was about 1-2 feet from the screen/webcam, for which we got about 97% accuracy as we have tested the system with different people and with several trials (ex. For testing 10 times, 9 trials were accurate). At 5-6 feet distance, its accuracy was about 60%. This limitation we expect to improve in future so that gesture can be recognize from more practical scenarios. We tested with different people. In Fig. 8, the mouse cursor movement is shown.

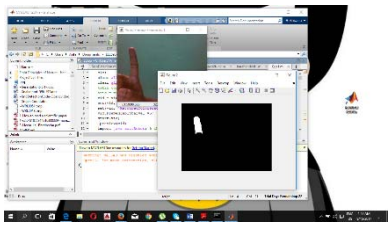

Fig. 8. Movement of mouse pointer on screen

The video preview window and the figure window show us the movement of mouse cursor according to the hand movement. The "PrntScr" command was used to get the snapshot. The pictures in Fig. 9 and Fig. 10 show the mouse on the screen from these screen shots.

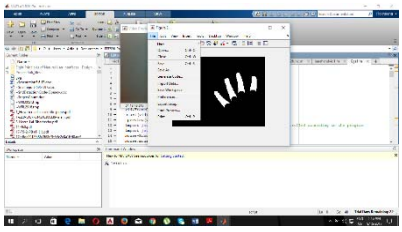

Fig. 9. Left click

In Fig. 9, the left cllick is shown on the figure window as the number of finger blob is 4. Here, the user cllicked on the "File" menu.

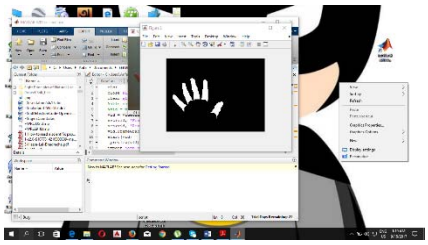

Fig. 10. Right click

In Fig. 10, the right click of mouse is shown as the number of finger blob is 5. To show it clearly, the clicking was performed on the screen.

V. CONCLUSION

In this paper we have presented the theoretical model and implementation of a real time vision based bare hand gesture controlled mouse system. Our system works using bare hand and built-in webcam. Mouseless computing is not far from reality anymore. Hand gesture recognition system has its own advantages over other intangible interfacing system like touch and voice controlling computers or other devices in certain scenarios and distance and noise conditions.

VI. FUTURE WORK

Our hand gesture recognition system is not limited for only mouse. It can be used for controlling many other devices, such as TV, robots inside a hazardous nuclear reactor in a convenient way for human and also in some other industrial setup. It can further be applied to control multimedia applications, multiplayer gaming etc. Sensibility of the system to lighting and background conditions can also be improved.

REFERENCES

[1] Stefan Reifinger, Frank Wallhoff, Markus Ablassmeier, Tony Poitschke, Gerhard Rigoll, "Static and Dynamic Hand-Gesture Recognition for Augmented Reality Applications," Intl. Conf. Human-Computer Interaction, pp. 727-737, 2007.

[2] Cristina Manresa-Yee, Javier Varona, Ramon Mas and Francisco J. Perales, "Real –Time Hand Tracking and Gesture Recognition for Human-Computer Interaction," Electronic Letters on Computer Vision and Image Analysis, ISSN 1577-5097 E(1):1, Jan. 2000.

[3] Heera Lal Bhadrecha, "Finger Gesture Control Computer Mouse With Image Processing," InternatIonal Journal of electronIcs & communIcatIon technology IJECT, Vol. 6, Issue 1, Jan. – Mar. 2015.

[4] Randeep Singh, Bhartendu Seth, Bhartendu Seth, "A Real-Time Framework for Vision based Human Robot Interaction," Intl. Conf. Intelligent Robots and Systems, Oct. 9 - 15, 2006, Beijing, China.

[5] Erdem Yörük, Ender Konukoglu, Bülent Sankur, and Jérôme Darbon, "Shape-Based Hand Recognition," IEEE Trans. Image Processing, Vol. 15, No. 7, July 2006.

[6] Pragati Garg, Naveen Aggarwal and Sanjeev Sofat,"Vision Based Hand Gesture Recognition," World Academy of Science, Engineering and Technology, International Journal of Computer, Electrical, Automation, Control and Information Engineering, Vol.3, No.1, 2009.

[7] David J. Rios-Soria, Satu E. Schaeffe, Sara E. Garza-Villarreal, "Hand-gesture recognition using computer-vision techniques," 21st International Conference on Computer Graphics, Visualization and Computer Vision, 2013.

[8] Alisha Pradhana, B.B.V.L. Deepak, "Design of Intangible Interface for Mouseless Computer Handling using Hand Gestures," Procedia Computer Science 79 (2016) 287 – 292, 7th International Conference on Communication, Computing and Virtualization 2016. Available [online]: www.sciencedirect.com.

[9] Radhika Bhatt, Nikita Fernandes, Archana Dhage, "Vision Based Hand Gesture Recognition for Human Computer Interaction," International Journal of Engineering Science and Innovative Technology (IJESIT), Vol. 2, Issue 3, May 2013.

[10] R. Lulu, Vanshri Kurhade, Neha Pande, Rohini Gaidhane, Nikita Khangar, Priya Chawhan, Ashwini Khaparde, "Web Camera Based Mouse Controlling Using RGB Colours," International Journal of Advanced Research in Computer Engineering & Technology (IJARCET), Vol. 4, Issue 2, Feb. 2015.

[11] Shridhar Kolap, Madhu Khaparkhande, Pooja Latane, "Mouse Control Using Image Processing," International Journal Of Engineering And Computer Science, ISSN: 2319-7242, Vol. 5, Issue 09, Sep. 2016.

[12] Ram Pratap Sharma and Gyanendra K. Verma, "Human Computer Interaction using Hand Gesture", 11th Intl. Multi-Conference on Information Processing (IMCIP-2015), Haryana, India, 2015.