International Journal of Research in Engineering and Science (IJRES)
ISSN (Online): 2320-9364, ISSN (Print): 2320-9356
www.ijres.org Volume 4 Issue 9 ǁ September. 2016 ǁ PP. 49-51

An Implementation of Virtual Reality Application of Cursor Control Using Webcam

Ayush Jain[1,] Monika Verma[2,]
Hmritm, Delhi

Abstract: This paper presents a design of an application using hand gestures contributing in the development and implementation of HCI i.e. human-computer interface system and which is more feasible to the users to interact with their pc's. Our approach contains steps for detecting the hand region, locating the fingers, classifying the gesture and calibrating it with mouse functions like clicking and scrolling operations. We demonstrate the effectiveness of the technique on real imagery.

I. INTRODUCTION

Humans communicate mainly by vision and sound; therefore, a man-machine interface would be more intuitive if it's made greater use of vision and audio recognition. Another advantage is that the user not only can communicate from a distance, but need have no physical contact with the computer. Vision-based approach offers a more natural way of HCI as it requires no physical contact with any devices. HCI primarily deals with design, implementation and assessment of new interfaces that improves the interaction of human and machines. Camera(s) placed at a fixed location or on a mobile platform in the environment are used for capturing the input image usually at a frame rate of 30 HZ or more. To recognize human gestures, these images are interpreted to produce visual features which can then be used to interpret human activity.

In our Virtual reality application where there is no need of hardware like keyboard and mouse.The idea behind this work is to develop and implement a gesture-based HCI system using webcam for all mouse pointing work.In the field of computer vision, many prototypes of intelligent vision-based interface systems have been developed that are more intuitive and cost-effective than the conventional interface devices such as the mouse and the keyboard

This research paper is directed towards new era of virtual reality application using emgu.cv libraries for image processing in visual studio. Paper is aimed to build a man-machine interface i.e. human computer interface using hand gestures in front of webcam making it interactive for the human. We have developed a windows application where a mouse control can be transferred to the webcam and accordingly using hand gestures, we are able to create HCI system.

Several Hand gesture recognition techniques already existed are based on Hidden Markov Models, Fuzzy Logic, Neural Networks, etc. These methods provide accurate recognition of hand gestures and are not robust enough for real-time implementation. But, computational cost required to achieve this is pretty high and processor speed is affected .To overcome this, we have developed an effortless method for recognizing simple hand gestures which depend purely on the simple segmentation and techniques.

This paper conveys vision-based fingertip detection and tracking system on a 2d plane using a camera as an input interface to the computer. An intensity based approach is used to detect the arbitrary shaped, uniform colored 2d area on which the hand operates, and then the fingertip is effectively detected and tracked using the sampled hand contour. According to finger count i.e. done using its detection by convexity defects algorithm, the application is interpreted it to simulate different mouse events.

II. LITERATURE SURVEY

Many methods have been developed by several researchers for controlling the mouse movement using a real time camera. Most of them are not robust enough for real time implementation and all of them use ambiguous methods for making a click event of a mouse .An approach developed by Chu-Feng Lien used finger tips for mouse movement and actions. Erdem used finger tracking for mouse control and the click was performed when the hand passed over a specified region . The action of clicking of mouse was done by keeping a track of the finger tips . Movement of the hand while making a special hand sign moved the mouse pointer.
Erosion trims down the image area where the hand is not present and Dilation expands the area of the Image pixels which are not eroded.
Mathematically, Erosion can be defined as,

$$A \ominus B = \bigcap\nolimits_{(b \in B)} A - b$$

Mathematically, Dilation can be defined as,

$$A \oplus B = \bigcup_{b \in B} Ab$$

In our paper, we performed erode function with dilate function. It could be seen that much of the background noise has been removed by erosion and dilation process.

III. SYSTEM FLOW

Three key stages are involved in the development of the proposed HCI system. Firstly, the webcam must be able to track skin colour of a user to detect the hand before any gestures can be recognized and processed. Secondly, each gesture must be properly recognized. Thirdly, gestures recognized must be interpreted to carry out the actions related to it. In general, these three key stages spans around skeleton tracking and gesture recognition functions. This section discusses how this is achieved by following steps:

3.1. Hand Detection

The most important aspect of developing a gesture-based HCI system is to first track the skin colour and hand detection can occur.

According to laptop specifications, the camera powered by "Exmor for PC" (Resolution: 1280 x 1024, Effective Pixels: 1.31 Mega pixels).

The images captured by camera are recorded in terms of frames. Each frame uses tracking system on a 2d plane using a camera as an input interface to the computer. An intensity based approach is used to detect the arbitrary shaped, uniform colored 2d area on which the hand operates.

3.2. Filtering Image

1.) YCBCR filter is applied, with required thresholds to obtain skin colored parts of the image.
2.) Eroding and Dilation of the filtered image is done using cv.erode() and cv.dilate () functions.
3.) Smoothing of the image is done using gaussian filter and from the resulting image the biggest contour is extracted.

3.3. Gesture Recognition

Hand is detected by the application and filtering for clear and accurate image is executed .After that we have effectively detected fingers using convexity defect algorithm and tracked using it a sampled hand contour is constructed. Using convexity defect algorithm we find the start point end point of the finger and draw a line between these points to show the counts of the finger The next step is to find centre of the contour, using moments of contour. As soon as we were able to get the co ordinates of the center we noticed there was too much fluctuation in the center of contour, as the hand continuously flickered. For this problem we divided the co ordinates of center by 10 to remove unit part of them as there was no fluctuation in the tenths of the co-ordinates.

3.4. Cursor Assignment To Hand &Cursor Movement

Before assigning the cursor to the hand and moving the cursor, a Win32 function called SendInput must be imported. This function can send Windows messages to the topmost window on the computer screen to control inputs (mouse, keyboard, hardware). It uses some basic Win32 structures defined for the mouse input.
After the user's hand has been detected and scaled to a Windows PC screen size, the X andY position of the cursor needs to be assigned to the user's hand. These parameters are then sent to the Windows OS which in this case causes the MouseEventMove flag to be raised, thus once the user moves the right hand (or left hand if left-handed), the cursor also moves in the direction of the hand.

3.5. Modelling Of Clicks And Detecting Gestures

The mouse Left-Click event flag is usually raised when a user hold the mouse left button down and then release it within a certain time frame. To simulate this action for example, the Windows OS user library described in Figure 10 above is used. The MouseEventLeftDown is sent to the OS followed by the MouseEventLeftUp corresponding to a user actually pressing a physical mouse left.

Each of the mouse events is triggered by gestures. The description of the gestures used for each mouse click is given as follows:

Cursor Movement: On 0 Or 1 Finger count.
Left click: On 4 Finger count
Double Click: On Two Finger Count.
Right Click: On Three Finger Count.

Scroll Function: On Five Finger Count

IV. EXPERIMENTAL RESULTS: OUR APPLICATION

This application uses emguCV, a .NET wrapper for openCV, to perform image processing, through which we try to recognize hand gestures and control mouse using these gestures. In this app, cursor movement is controlled by the movement of hand and click events are triggered using hand gestures.

The code uses a well known library for image processing, known as, openCV in the form of emguCV which is a .NET wrapper of it. The code is really simple and easy to understand. First of all the application tries to catch a video input device. If it is successful in doing so, it calls the function ProcessFramAndUpdateGUI() where all the processing is done.

REFERENCES

[1]. A. Chaudhary et al., "Intelligent approaches to interact with machines using hand gesture recognition in a natural way: A survey," International Journal of Computer Science and Engineering Survey(IJCSES), vol. 2, no. 1, Feb 2011.

[2]. A. Albiol, L. Torres, and E. J. Delp, "Optimum color spaces for skin detection," in Proceedings of 2001 Image Processing International Conference, vol. 1, pp. 122-124, 2001.

[3]. A. Pandit, D. Dand, S. M. Sabesan, A. Daftery, "A simple wearable hand gesture recognition device using iMEMS," Soft Computing and Pattern Recognition, 2009. SOCPAR '09. International Conference,vol. 4, no. 7, pp. 592-597, Dec. 2009.

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

[5]. Hojoon Park, "A Method for Controlling the Mouse Movement using a Real Time Camera", Brown University, Providence, RI, USA, Department of computer science, 2008

[6]. Chu-Feng Lien, "Portable Vision-Based HCI – A Realtime Hand Mouse System on Handheld Devices", National Taiwan University, Computer Science and Information Engineering Department

[7]. Kamran Niyazi, Vikram Kumar, Swapnil Mahe, Swapnil Vyawahare, "Mouse Simulation Using Two Coloured Tapes", Department of Computer Science, University of Pune, India, International Journal of Information Sciences and Techniques (IJIST) Vol.2, No.2, March 2012

[8]. K N. Shah, K R. Rathod and S. J. Agravat, "A survey on Human Computer Interaction Mechanism Using Finger Tracking" International Journal of Computer Trends and Technology, 7(3), 2014, 174-177

[9]. Rafael C. Gonzalez and Richard E. Woods, Digital Image Processing, 2nd edition, Prentice Hall, Upper Saddle River, New Jersey, 07458

[10]. Shahzad Malik, "Real-time Hand Tracking and Finger Tracking for Interaction", CSC2503F Project Report,