

Mouse on Finger Tips using ML and AI

Nilesh Patil¹, Gaurav Sali², Nikhil Lokhande³

^{1,2,3}BE(CSE), JSPM's BSIOTR, Wagholi, Pune.

Abstract - Many modules have been developed to help the physical world interact with the digital world. Here we present a novel approach for Human Interaction with Computer (HCI) where, we control cursor movement using a real-time camera and color pointers (fingertips). Our method is to use a Web camera and computer vision technology, such as image segmentation, background subtraction and color tracking, to control mouse tasks (left clicking, right clicking, and scrolling actions) and we show how it can perform everything as current mouse devices can. A color pointer (fingertips) has been used for the object recognition and tracking, so as to implement the module without any physical contact with the system. Click events of the mouse have been achieved by detecting the number of pointers (fingertips) on the images. The application has been created on Python IDE with operating system as windows 10. This method mainly focuses on the use of a Web Camera to develop a virtual human computer interaction device in the cost effective manner.

Key Words: Human Computer Interaction, Background Subtraction, Colour Detection, Web Camera, Computer Vision, Fingertips

1. INTRODUCTION

The importance of computers has increased to a great extent now a days. Then it can be used for general purpose or at work places. Even there is a demand of more and more application based devices, where the latest example can be of smart phones. In general, we use hardware devices i.e. the mouse and keyboard to interact with the Computer system. But due to the growth of technology and demand of virtualization, this technique has been proposed. An alternative way for touch screen can be creation of a virtual human computer interactive device.

1.1 Overview

Many researchers in computer science and human computer interaction developed various technologies related to virtual keyboard and mouse. However all of them used different techniques.

The various HCI Interfaces that are being used since earlier times having some limitations. They block the betterment of computer dependent devices. It is now a general proclivity to lessen human efforts and overcome the usual, traditionally being used computer dependent devices. Considering an example of a keyboard, it is not possible to reduce the entire keyboard in a very small size so that it can

accommodate the fingers. Thus it becomes somewhat difficulty while working where correctness is needed.

Gesture is defined as an expressive, meaningful body motion that expresses an idea, opinion, emotion, etc. Human interaction with a computer in which human gestures, usually hand motions, are recognized by the computer. Gesture recognition is used to interact with computers, such as interpreting sign language. The hand gesture recognition technique can be classified into two types: 3D-hand model approach and the appearance based approach. A static gesture is observed at the sudden burst of time whereas a dynamic gesture is intended to change over a period of time. A waving hand means goodbye is an example of dynamic gesture and the stop sign is an example of static gesture. To understand a full message, it is necessary to interpret all the static and dynamic gestures over a period of time. This complex process is called gesture recognition.

1.2 Objective

The main objective is to develop an object tracking application to interact with the computer, and a virtual human computer interaction device, where in no physical contact with the device is required and also to provide an easier human-machine interaction routine.

1.3 Purpose

The purpose of the project is to create a virtual mouse that works with the help of a web camera. In this project a camera continuously takes images of hand movements of a user, which is then mapped into mouse inputs. This means that we can give inputs to computer without having any physical connection with the computer and without having any hardware movements.

1.4 Problem Definition

To develop a software solution to a problem, the first step is to understand the problem. The problem here is to develop a way so that humans can interact with a computer without having any physical connection with the computer. Many ideas were put forward but they all required physical movement of hardware. So the final decision is to develop a virtual mouse which uses simple and cheap image processing techniques.

2. PROPOSED WORK

The proposed system is a real time video processing that is based on a real time application system. This can replace one of the traditionally used input device i.e. mouse so that simply by using the hand gestures the user will be able to interact naturally with their computer. The basic block diagram of the overall proposed system is as shown below in the figure.

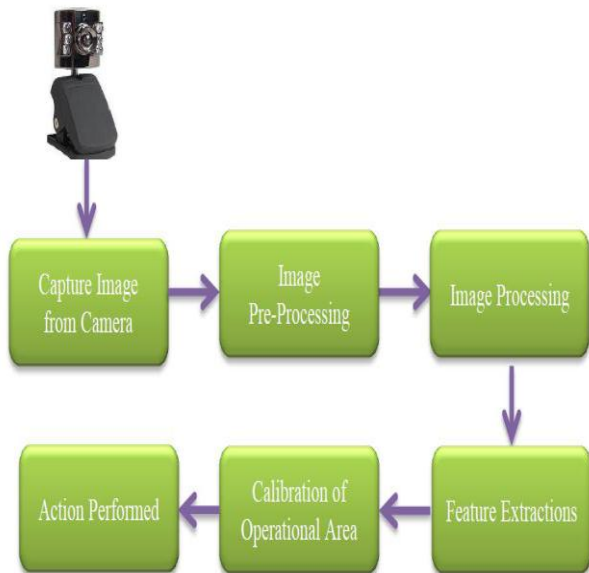


Fig -1: Single frame flow

As soon as the color coded fingertips are detected by the camera, it starts capturing the images and takes out the current images and starts processing. The processed image is given as input to the next phase of processing where it starts detecting the color coded fingertips. Following the centers and edges. These all extracted features are used for the further processing such as detecting hand gestures and working according to it. The flowchart of the proposed system above is shown for the single frame which is captured. The system is a real time system so the flowchart is a continuous process for each and every frame that is captured by the web camera.

In the protest following application one of the principle issues is question identification. Rather than fingertips, a shading pointer has been utilized to make the question recognition simple and quick. A circle blue tip is utilized as a shading pointer in this review. To mimic the snap occasions of the mouse three fingers with three shading pointers(fingertips) has been utilized.

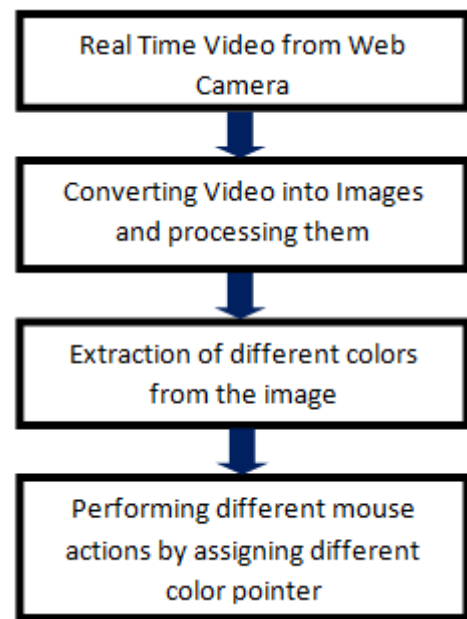


Fig -2: Basic flowchart.

System Approach:

- Capturing real time video using Web-Camera.
- Processing the individual image frame.
- Flipping of each image frame.
- Conversion of each frame to a grey scale image.
- Color detection and extraction of the different colors (RGB) from flipped gray scale image.
- Conversion of the detected image into a RGB to HSV color format.
- Finding the region of the image and calculating its centroid.
- Tracking the mouse pointer using the coordinates obtained from the centroid.
- Simulating the left click and the right click events of the mouse by assigning different color pointers.

3. PROPOSED METHODOLOGY

3.1 Capturing the real time video

For the framework to work we require a sensor to distinguish the hand developments of the client. The webcam of the PC is utilized as a sensor. The webcam catches the constant video at a settled casing rate and determination which is controlled by the equipment of the camera. The casing rate and determination can be changed in the framework if required. Computer Webcam is utilized

to catch the Real Time Video. Video is partitioned into Image outlines in light of the FPS (Frames every second) of the camera Processing of individual Frames.



Fig -3: Capturing Video

3.2 Flipping of Images

At the point when the camera catches a picture, it is upset. This implies on the off chance that we move the shading pointer towards the left, the picture of the pointer moves towards the privilege and the other way around. It's like a picture acquired when we remain before a mirror (Left is distinguished as right and right is recognized as cleared out). To maintain a strategic distance from this issue we have to vertically flip the picture. The picture caught is a RGB picture and flipping activities can't be straightforwardly performed on it. So the individual shading channels of the picture are isolated and afterward they are flipped independently. In the wake of flipping the red, blue and green hued channels separately, they are connected and a flipped RGB picture is gotten.

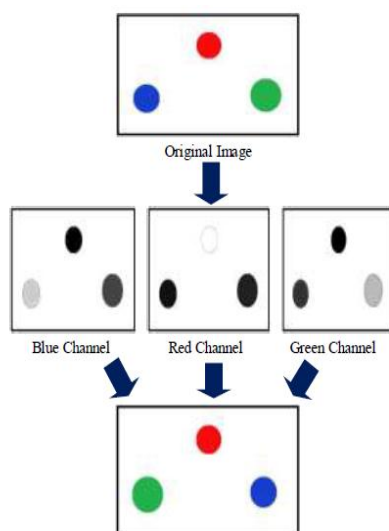


Fig -4: Flipping of an Image

3.3 Conversion of Flipped Image into Gray scale Image

When contrasted with a shaded picture, computational many-sided quality is diminished in a dim scale picture. In this way the flipped picture is changed over into a dim scale picture. All the fundamental operations were performed in the wake of changing over the picture into dark scale.

3.4 Color Detection

This is the most critical stride in the entire procedure. The red, green and blue shading article is recognized by subtracting the flipped shading smothered channel from the flipped Gray-Scale Image. This makes a picture which contains the recognized protest as a fix of dim encompassed by dark space.

3.5 Performing Clicking Actions

The control activities of the mouse are performed by controlling the banners related with the mouse catches. JAVA is utilized to get to these banners. The client needs to perform hand motions keeping in mind the end goal to make the control activities. Because of the utilization of shading pointers, the calculation time required is lessened. Besides the framework ends up noticeably impervious to foundation clamor and low light conditions. The discovery of green and blue hues takes after a similar strategy examined previously. Clicking activity depends on concurrent recognition of two hues.

3.6 Color Detection

```
imgHSV=cv.cvtColor(img,cv.COLOR_BGR2HSV)
mask=cv.inRange(imgHSV,lowerBound,upperBound)

maskOpen=cv.morphologyEx(mask,cv.MORPH_OPEN,kernel
Open)

maskClose=cv.morphologyEx(maskOpen,cv.MORPH_CLOSE,
kernelClose)

maskFinal=maskClose

conts,h=cv.findContours(maskFinal.copy(),cv.RETR_EXTERN
AL,cv.CHAIN_APPROX_NONE)
```

4. CONCLUSION

The proposed system is used to control the mouse cursor and implement its function using a real-time camera. We implemented mouse movement, selection of the icons and its functions like right, left, double click and scrolling. This system is based on image comparison and motion detection technology to do mouse pointer movements and selection of icon. From the results, we can expect that if the algorithms can work in all environments then our system will work more efficiently. This system could be useful in

presentations and to reduce work space. In the future, we plan to add more features such as enlarging and shrinking windows, closing window, etc. by using the palm and multiple fingers.

REFERENCES

- [1] **A. Erdem, E. Yardimci, Y. Atalay, V. Cetin, A. E.** "Computer vision based mouse", Acoustics, Speech, and Signal Processing, Proceedings. (ICASS). *IEEE International Conference*, 2012
- [2] **Hojoon Park**, "A Method for Controlling the Mouse Movement using a Real Time Camera", *Brown University*, Providence, RI, USA, Department of computer science, 2013
- [3] **Chu-Feng Lien**, "Portable Vision-Based HCI – A Realtime Hand Mouse System on Handheld Devices", *National Taiwan University*, Computer Science and Information Engineering Department
- [4] **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
- [5] **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
- [6] **Rafael C. Gonzalez and Richard E. Woods**, *Digital Image Processing*, 2nd edition, Prentice Hall, Upper Saddle River, New Jersey, 07458
- [7] **Shahzad Malik**, "Real-time Hand Tracking and Finger Tracking for Interaction", CSC2503F Project Report, December 18, 2013