PROJECT REPORT

TOPIC: *CURSOR MOVEMENT USING HAND GESTURES*

Abstract

This project is a **mouse simulation system** which performs all the functions performed by your mouse corresponding to your hand movements and gestures. Simply speaking, a camera captures your video and depending on your hand gestures, you can move the cursor and perform **left click, right click, drag, select and scroll up and down.** The predefined gestures make use of **only three fingers marked by different colors.**

The code is written on Python. It uses of the cross-platform image processing module OpenCV and implements the mouse actions using Python specific library PyAutoGUI. Thus, in addition to a webcam (which almost all laptops are already loaded with) a computer needs to be pre-equipped with the following packages for the project to run such as:

- **Python 2.7** interpreter
- **OpenCV:** for screen capture, taking inputs and recognition
- **Numpy:** for slicing up the index and coordinates. Numpy is the core library for scientific computing in Python.
- **PyAutoGUI:** Lets Python control the mouse and keyboard, and other GUI automation task

Video captures by the webcam is processed and only the three colored finger tips are extracted. Their centers are calculated using method of moments and depending upon their relative positions it is decided that what action is to be performed.

Introduction

The project “Mouse control using Hand Gestures” is developed aiming to better the process of human-computer interaction. It aims to provide the user a better understanding of the system and to let them use alternate ways of interacting with the computer for a task.

The task here is to control the mouse even from a distance just by using hand gestures. It uses a program in python and various libraries such as PyAutoGUI, Numpy and image processing module OpenCV to read a video feed which identifies the user’s fingers represented by three different colors and track their movements. It retrieves necessary data and implements it to the mouse interface of the computer according to predefined notions.

Methodology

The HSV color space was chosen since it was found by to be the best color space for skin detection. The next step would be to use a method that would differentiate selected color pixels from non-color pixels in the image (color detection).
The system can be broken down in four main components, which are:

- Color detection
- Color Contour Extraction
- Hand Tracking
- Gesture Recognition
- Cursor Control

Implementation

The first thing that we do is convert the captured video into HSV format. Now the user gets to calibrate the color ranges for three of his fingers individually. Depending on the calibrations, only the three fingertips are extracted from the video, one by one. In order to remove noise in the video feed, we apply a two-step morphism i.e. erosion and dilation. The noise filtered image referred to as mask in the program is then sent for locating the centers.
Depending upon its output, function carries out either of the following using the PyAutoGUI library:

- free cursor movement
- left click
- right click
- drag/select
- scroll up
- scroll down

Project Specifications

Software Specifications:

- 64-bit Operating System: Windows 8 or Higher
- OpenCV 2.4.9 needs to be installed prior to running.
- Windows Administrator permissions are needed for some parts of the program to function properly.

Hardware Specifications:

A Webcam

Other Specifications:

- A clear white background
- There should be no other objects (especially red, blue, yellow colored) in front of the webcam other than on the fingers.

FREE MOVEMENT

LEFT CLICK

RIGHT CLICK

DRAG / SELECT

SCROLL DOWN

SCROLL UP