Creating a instrument based on movement in webcam view

Hamid Shayestehmanesh

October 28, 2016

Contents

0.1 Main Project Idea

The basic idea was to create a device which use a webcam and a simple electronic element which is physically like conductor baton and also can be identified by the webcam. Then the user stands in front of the webcam and moves the electronic element, a sound will generates the hole time, but the frequency and amplitude will change according to the movement of the element.

0.1.1 Goal

Goal is to create a new music instrument.

0.2 First Sample

To decide what can be build practically we should have a minimal model of the system first.

0.2.1 Abstract

A minimal model implemented using Python, OpenCV(Tracking the Object) and Pygame(Generating Sounds). In this model moving the element (conductor baton) Vertically changes the frequency of sound and moving it horizontally changes sounds' amplitude. Coordination of the element identified by color segmentation.
Movement of the element also was computed using it's coordination in previous frame.

0.2.2 Details

The sample code works with one specific element. The idea was to clean all the view of the webcam from one color and create that element in that color, so filtering on that color will let us to find the element easily. Filtering the color or color segmentation will give us a black and white frame which everything should be black but the element. In practice some other small areas will be converted to white, to solve this problem the image has to be cleaned. *Eroding* and *dilating* the image will fairly solve our problem here. In next step we should identify the element by one point to track it's movement,thus to do this first draw a contour around the white color. As the contour won't usually have a good shape, finding the minimum circle around the contour can give us a good presentation of the element coordination in frame, also center of circle is a good one-point identifier.
Well now comparing two last center of this imaginary circle can help us detect any movement of the element. Now that the movement is detected, we can update our frequency and amplitude. Generate and play a sound with specified frequency and amplitude using Pygame.

0.3 Percussion Idea

The idea is to create a virtual percussion. Percussion contains many type of instruments. The most practical ones to implement here is congas-likes. Congas generates sounds after the musician strikes it. The characteristic of the generated sounds depends of congas physical features, the point musician strikes it and the power of the strike.

0.3.1 Abstract

Four virtual congas' will be shown on the frame. The user will also see his/her hand on the frame. His/Her hands will be detected, and sounds will be generated for each strike user does, exactly the way a real congas works.

The frame of program will be something like this.(without hands)

0.3.2 Details

To generate a good sounds strikes should be detected correctly.

Detecting Strike

Element size

To identify strikes using webcam, first idea is to use radius of the circle around the element obtained in sample model, larger radius means closer element. But because webcam pictures have some delay and cleaning the image is not very accurate for exact size, in practice it won't help too much. The most important problem with this method is that if you want to find speed or power of strike which is necessary here, not only the size of the element in frames depends on it's distance from webcam but also on how well the cleaning phase works. Using this method (Eroding and delighting) won't work well, in practice it's not fast enough therefor some frames of webcam will be lost which means we won't be able to calculate a strike's speed or sometimes a strike. To solve the problem we should change our approach to problem. so it didn't work well.

0.4 Object Tracking

The idea is to move and click with mouse using object movements in the read frame from camera.

0.4.1 Improving the Tracking system

Step 1

The first two Objects are recognized by their size and color in the picture. The problem with these two objects is that if any object with the same color appears in the frame and has an appropriate size, the system will follow the wrong object of it will repeatedly change the recognized object between the correct one and the other, so the system is not robust.

FigureObject1 Used from top (second image)

Step 2

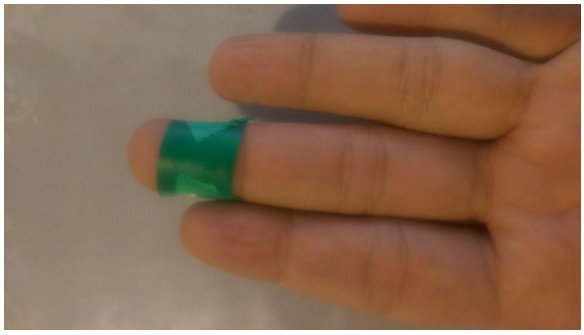
One easy idea to solve the problem is to increase the size of the object.

FigureObject2 Used from side

Due to the fact that this step still recognize the object by size and color increasing the size won't help, mostly because of noises, sometimes the contour around the detected color becomes small specially in fast pace movements. (Notice that system searches for a hard coded size of contour.)

Step 3

One Idea to solve the problem is to create a specific background, therefore the colored point won't be lost in the background color. First attempt with this idea is one stick on hand.



FigureThe green color was seperated to detect the point

Step 4

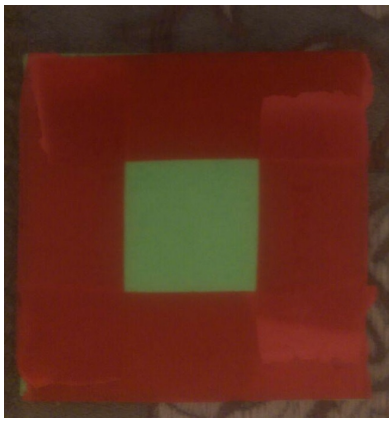
Even though a green stick on the finger seems detectable but in practice the color of skin is not very stable so it isn't very helpful as a background, also in light has a enormous effect on the color.

Step 5

To solve the problem some red stick are added in up and down and then on left and right. Even the system works fine with only up and down red sticks but

adding left and right ones will help the system when very bright lights exists in the frame.

Due to the fact that sticks are very annoying I created something with papers.



FigureCreated with papers

Step 6

The papered object worked fine but because of the small background sometimes following the point became hard. To solve the problem we use the fact that one's hand won't move very fast, so if call current position of point $p_1 = (x_1, y_1)$ and

next position $p_2 = (x_2, y_2)$ then the distance between p_1, p_2 is less than C. So if we know the place of point, so we don't have to search the hole next frame to find the point. We have to search only a part of it, so a square with upper left point of $p_1 - (C, C)$ and the lower right point of $p_1 + (C, C)$

Step 7

If we could ever make the background bigger the system could work more robust. We created a glove with the point one it. The big background it used to create help the system to track the point using only mentioned square.

This object works very well but yet light can make huge distractions. To solve the problem we change many different color for the glove (as the background) (Colors are shown below) and the Red Dark seemed to work better than others.

0.5 Future Work

It can be implemented on Android or any other mobile frameworks.

0.6 Other Technologies

A programming language called Chuck has been created in Princeton university. ChucK is a programming language for real-time sound synthesis and music creation. The problem with this language is that it's doesn't support any library which can use webcam or any type of camera, neither easy way to connect it to another programming language.