Maggie Zhang, Weiyan Zhu
CSC294 - 137, HW 4 Report
Nov. 2, 2020

In this report, we included details of our implementation for hand gesture tracking and detection in homework 4 of CS294.

Part1

For this part, we extracted the hand from the camera. We used the sample code on Piazza as a starting point. We followed the code snippet to do skin color matching to get the hand, and we modified the color values to make it work better in our settings. In future works, we would consider not only skin color but also skin texture in order to develop an algorithm that detects skin regardless of skin shades.

We then applied erosion/dilation to the area surrounding the skin mask region and gaussian blur within the skin mask region to reduce noise.

Part2

For this part, we detected the finger ring. We first threshold the image, then we use cv2 to find all connected components. Sorting the areas by how large they are, we can find the finger ring, which is the third largest area.

We then use cv2 to find the contours of the finger ring and fit an eclipse to it. We can get the size, position and angle of the eclipse, which we can use in part4.

Part3

For this part, we processed the hand image and detected how many fingers we have in the image. We need to threshold the image first as we did in part2, but unlike part2 we do not invert the gray image. Then we used the code snippet to detect contours and hulls, but before we drew lines and circles, we converted the gray images back to color images in order to add colored annotations and markers on the same image.

We also needed to check the angle between fingers to filter out noises. Because we want to compare the images before and after applying the heuristic, we modified the code sample to store and show two images, one with noise filtered and another without. We also calculated the finger counts by checking how many defects remain after applying the heuristic, and the finger count is simply that number plus one. At the end, we put a text on the image showing how many fingers are detected.

Part4

For this part, we detect gestures and control mouse/keyboards through that.

Since there is some noise in the data we get from images, we do not want to control the button/mouse immediately when the gestures are detected. We have a counter that shows how many times the gesture has been detected, and we control the keyboard only when the counter hits a certain number. So even if some gestures are incorrectly interpreted, the functionality won't be triggered. This largely reduced noise in our algorithm.

Another method that helps reduce noise is using thresholds. Only when the change of area/angle is large enough will we record them in our counters, since even if the user does not move their hands, the variables may still increase/decrease slightly due to noise.

<u>3 simple gestures:</u>

1. Space: We detect the finger counts. If users hold five fingers up, we will press "Space" for them.

2. Click: If users form a fist (0 finger), we will trigger a mouse click for them

3. Exit: If the users hold three fingers up for 90 frames, we will exit the program for them.

<u>4 complex gestures:</u>

1. Zoom in: We record the previous angle between fingers and compare it with current ones. If the angle keeps increasing for a predetermined number (threshold) of frames, we will press "Ctrl" plus "+" for the user. It should also be noted that whenever we detect an increasing angle, we not only increase the increasing_angle counter but also decrease the decreasing_angle counter. This largely reduces the noise in our gesture detection algorithm.

2. Zoom out: If the angle keeps decreasing for a predetermined number of frames, we will press "Ctrl" plus "-" for the user.

3. Volume up: We record the previous finger ring area and compare it with current ones. If the finger ring area keeps increasing for certain frames, we will press the volume up keyboard shortcut for users and increase the volume

4. Volume down: If the finger ring area keeps decreasing for certain frames, we will press the volume down keyboard shortcut for users and decrease the volume