Assignment 2

CS 585 HW 2
Alex Wong and Wei-Hsiang Lin
Feb 12, 2020

Problem Definition

1. Design and Implement algorithm to detect hand gestures and Apply it to graphical display that reacts to the hand gestures
2. Must use skin-detection.
3. Algorithm must detect 4 different hand gestures.
4. Create confusion matrix for the algorithm.

The result of this experiment is very important, because object tracking is a big topic in the area of Computer Vision. In this experimented, we assume that there is consistent background and lighting conditions, as these are required for our project to work (as we are using hsv-thresholding and template matching). Anticipated difficulties include differentiating hand and non-hand class on an image and classifying the correct hand gesture.

Method and Implementation

There are 3 different frames in our program:

1. **Original:** Displays exactly what's captured by the webcam. Here, you tune the hsv thresholds (explained later). The recognized/detected gesture is displayed here with a text indicating the classified gesture and the confidence score.
2. **Filtered:** Displays the hsv-threshold filtered frame. This is useful in seeing what the algorithm sees prior to template matching. You should tune HSV thresholds while looking at this.
3. **Drawing Frame:** Once HSV threshold is setup, you can use hand gestures to draw and download your artwork into the 'download_frame' folder.

From the user perspective, the first step is to tune the HSV threshold so it recognizes your hand only. To help, the user can click on a pixel in the video frame and see the hsv value. This helps to figure out the best hsv threshold.

Once the threshold is fine-tuned, the user should move their hand around (for each gestures) to make sure. The reason is that the HSV threshold is sensitive to lighting conditions and backgrounds. Therefore, even if the detection works on one position, it might not work when your move hands to another place.

The third step is start drawing. Each gesture corresponds to a different function in the drawing. The functions are listed below:

1. **Paper:** All fingers open. This will clear the drawing frame
2. **Pointer:** Only 1 finger sticks out. This allows you to start drawing with your finger (on the drawing frame).
3. **Rock:** All fingers enclosed. This does nothing. It acts as a neutral gesture to allow you to move around the image/frame without drawing anything on the frame. Should be used in combination with 'pointer' to draw better images.
4. **Seven:** Thumb and pointer sticks out. This will download the image to the 'download_frame' folder.

Note that, there are checks put in place for the paper and seven gesture to be triggered only when it is detected 20 frames in a row.

Experiments

Environmental Factors

We experimented with many different backgrounds and lighting conditions. We found that the algorithm works best with backgrounds and lighting conditions that are more consistent in color and brightness.

Algorithms

Additionally, we attempted different methods to perform hsv thresholding and image processing. After experimenting different combinations, we decided that the hsv values for thresholding should be done manually because the lighting conditions, skin color and backgrounds will vary almost every time. As for image processing, after the frame is filtered via hsv-thresholding, we perform a combination of basic image processing techniques. These include gaussian blur, binary thresholding, dilation and erosion. Additionally, we perform 'object proposal'. This is partially inspired by how CNN works. The idea is to use contours to figure out the biggest blob and only focus in on that blob. After that, we draw a bounding box around that blob and perform template matching. Since the left and right is symmetrical (horizontally), we perform template matching on both the blob and the blob when flipped horizontally. To classify the gesture into the 4 possible gestures, we compute a confidence score based on template matching's max/peak values. We select the template with the highest confidence score with the bounding box cropped image and consider this as our region of interest (ROI) or, in simple terms, the gesture recognized.

Hand Gestures

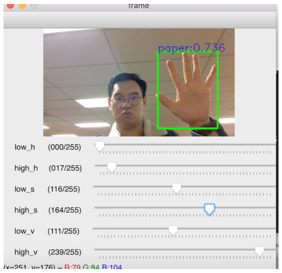
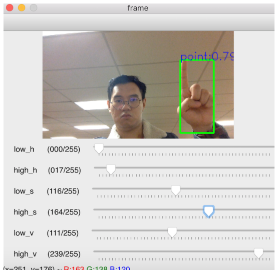
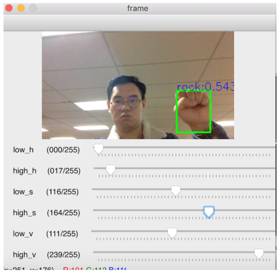
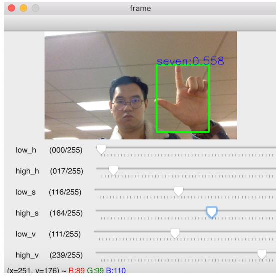
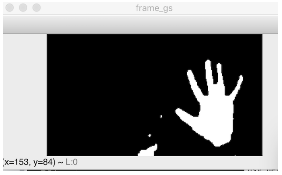
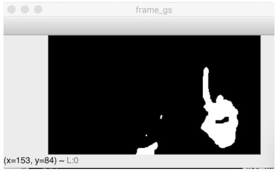
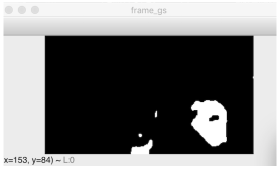
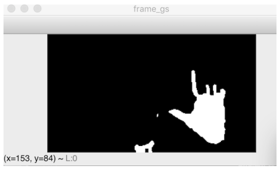
We found that not all gestures work equally. There are some gestures that are very similar, and thus, is more prone to misclassification. There are some gestures that are harder to be detected after hsv-thresolding because the way the gesture is positioned might affect how the skin color is displayed. For example, flipping your hand upside down might change the light refracted from your skin because the angle of your hand from the light is now changed and not accounted when you manually perform the hsv thresholding.

After experimenting with different gestures, we deicided to go with **paper, pointer, rock and seven** gestures. due to the consideratiosn mentioned above

Results

The results were good as long as the environmental factors (explained earlier) and hsv threshold values are optimal.

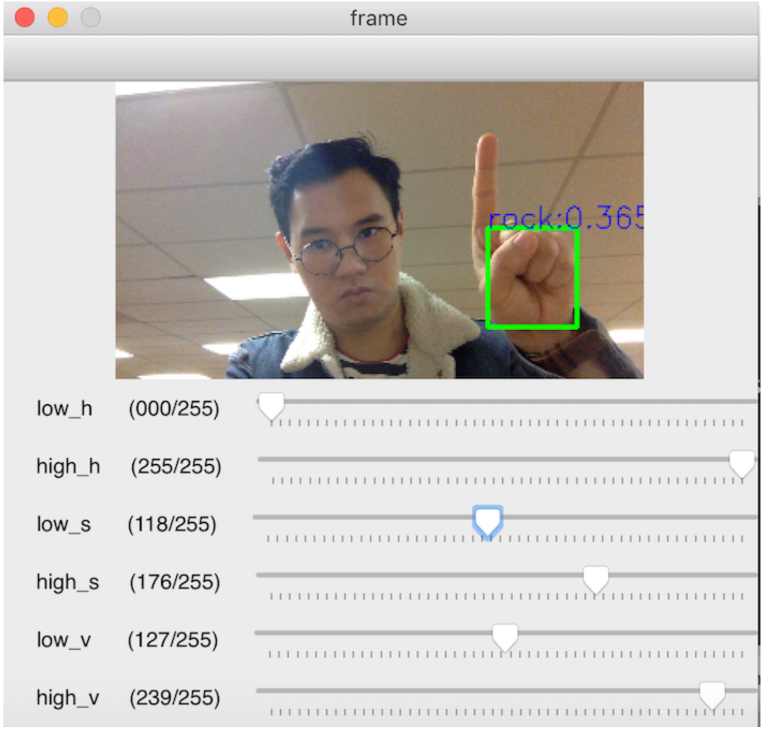
Examples of Detecting Hand Gestures

	Paper Gestures	Pointer Gestures	Rock Gestures	Seven Gestures
Main Frame (Object Detection Bounding Box + HSV Threshold Tuning)	 frame paper:0.736 low_h (000/255) high_h (017/255) low_s (116/255) high_s (164/255) low_v (111/255) high_v (239/255) (x=251, y=176) ~ R:79 G:84 B:104	 frame point:0.79 low_h (000/255) high_h (017/255) low_s (116/255) high_s (164/255) low_v (111/255) high_v (239/255) (x=251, y=176) ~ R:163 G:138 B:120	 frame rock:0.543 low_h (000/255) high_h (017/255) low_s (116/255) high_s (164/255) low_v (111/255) high_v (239/255)	 frame seven:0.558 low_h (000/255) high_h (017/255) low_s (116/255) high_s (164/255) low_v (111/255) high_v (239/255) (x=251, y=176) ~ R:89 G:99 B:110
HSV-Thresholded Frame	 frame_gs x=153, y=84) ~ L:0	 frame_gs (x=153, y=84) ~ L:0	 frame_gs x=153, y=84) ~ L:0	 frame_gs (x=153, y=84) ~ L:0

Fail Example

As I explained earlier, sometimes, the detection doesn't work as intended. Below is an example:

Main Frame (Object Detection Bounding Box + HSV Threshold Tuning)

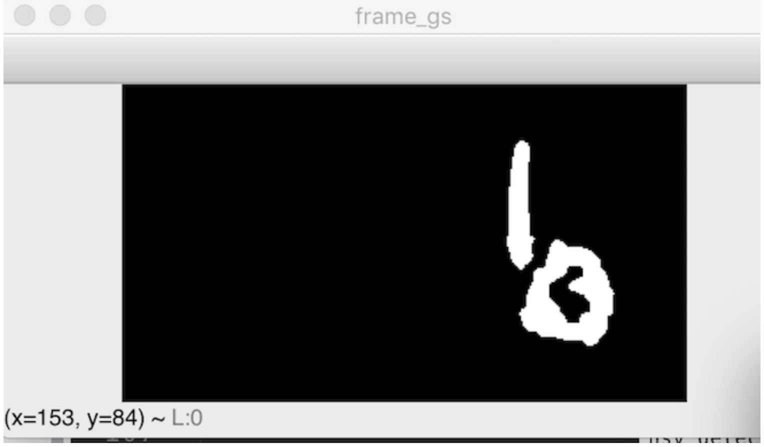


frame

rock:0.365

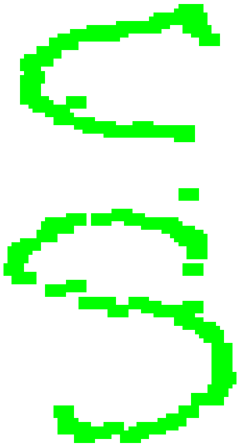
low_h (000/255)
high_h (255/255)
low_s (118/255)
high_s (176/255)
low_v (127/255)
high_v (239/255)

HSV-Thresholded Frame



In the image above, the template matching of the hand failed because there is a disconnect between the pointer finger and the palm. This led the algorithm to think that it is a rock gesture. The reason this happened is because the HSV thresholding was too strong. However, at the same time, if it not strong enough, then environmental noise would not be filtered. Eventually, the cause of the problem becomes a deadlock. In short, the algorithm still have many weaknesses and suggestions for improvements will be discussed later (in the 'Discussion' section.

Example of Drawing



Confusion Matrix

To demonstrate our performance metrics, we plotted a confusion matrix. Note that this follows that assumption that environmental factors (explained above) is optimal and that the hsv threshold values are tuned optimally.

Confusion Matrix	True Class			
	Hand	Rock	Pointer	Seven

Hypothesized Class	Hand	**18**	0	0	0
	Rock	0	**15**	2	3
	Pointer	1	4	**17**	1
	Seven	1	1	1	**16**

I think the confusion matrix demonstrates that the accuracy of our end-product has good results (although it can be improved further), as each gesture's max lies in its True Positive.

Recall, Precision and Accuracy

Measurements	Paper Gesture	Pointer Gesture	Rock Gesture	Seven Gesture
Recall	0.5625	0.625	0.68	0.5926
Precision	1	0.75	0.739	0.842
Accuracy	0.825	0.825	0.825	0.825

Discussion

Future work can focus on:

1. **Frame-to-Frame Differencing:** This can be very interesting. Because the application is drawing, it implies that the pointer will be moving. If we track the motion, we can filter out the background (since they are not moving) much better. Whether hsv-thresholding is required when this is applied is debatable. More experiments will be required to decide this.
2. **Background Differencing:** If we can take the image of the background and then remove the background from our frame, then we have much less noise to begin with and detecting hand gesture might be easier.
3. **HSV Thresholding by clicking on ROI:** This was almost implemented by didn't make it due to time constraints. We can retrieve the HSV of our ROI by clicking the video and using that, we can figure out the HSV thresholding values, rather than manually tuning it. However, one of the bottlenecks we faced was that, even on a hand, there are many different HSV values. When one finger covers over the palm, a shadow exist and this changes the HSV values greatly. Our suggested solution is allow several different clicks and have a list of HSV thresholds.

Conclusions

This was a good assignment. We were very skeptical whether template matching would work when we started. In our initial experiments, template matching was horrible due to the noise generated from environmental factors. After applying image processing and some form of object proposal, we were able to denoise and achieved significantly better results. I can see how this technique can be researched further to achieve more complicated CV tasks.

Credits and Bibliography

OpenCV documentation