

CS415 HW4 Report

Histogram-Based Skin Color Detector

Xichen Liu

The algorithm of the detector is to use several skin images to train the model, to get an overall distribution of the skin colors in HSV color space, s.t. a particular color has some probability to be skin color. And then apply the probability to every pixel in the test images, to find its likelihood to be a skin color.

The os and sys package is to read the local files, cv2 and np is to process the points of an image, plt is to show the histogram. And I also use the csv package to output the 2d array of histogram into a csv file.

The detailed algorithm is, firstly, initialize a 2d array to be histogram, and the domain is all available (H, S) pairs, and the value is the number of appearance of a pair, and finally will be normalized to a probability.

As for this part, I tried different size of the bin of histogram.

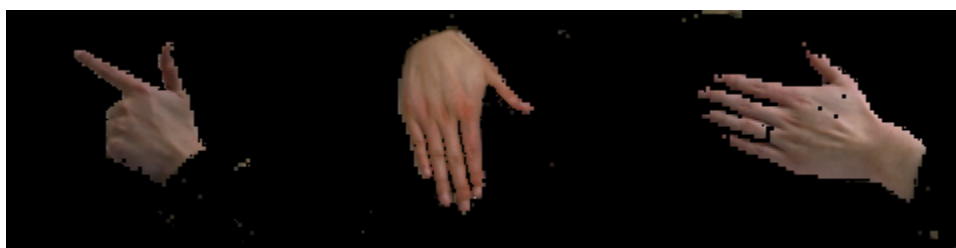
Second, after we got the histogram model, apply the model to the test image. For every pixel of the test image, it has a color, and find the corresponding color from the histogram, we will have a probability for the color to be the skin color. Compare the probability with a given threshold, if the probability is greater than the threshold, it means the pixel is likely to be a piece of skin.

Here are the images for training:



Results:

Divide the (H, S) pairs into 11 bins & threshold is 0.0001:



Divide the (H, S) pairs into 11 bins & threshold is 0.001:



Divide the (H, S) pairs into 11 bins & threshold is 0.01:



Divide the (H, S) pairs into 150 bins & threshold is 0.0001:



Divide the (H, S) pairs into 150 bins & threshold is 0.001:



Divide the (H, S) pairs into 150 bins & threshold is 0.01:



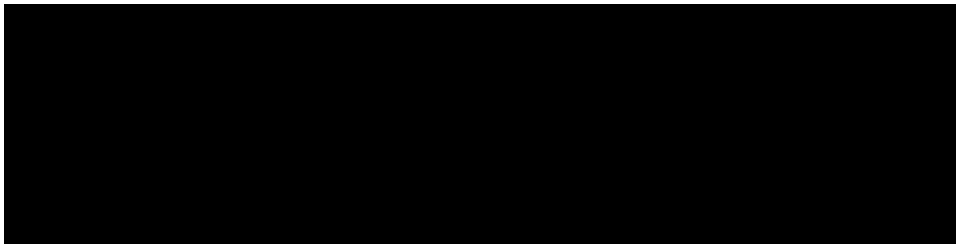
Divide the (H, S) pairs into 256 bins & threshold is 0.0001:



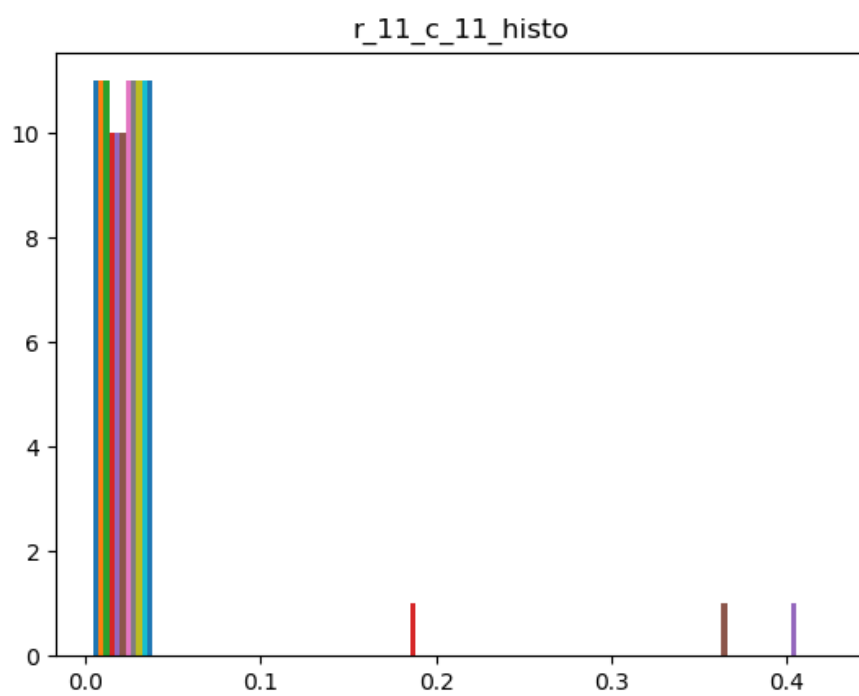
Divide the (H, S) pairs into 256 bins & threshold is 0.001:



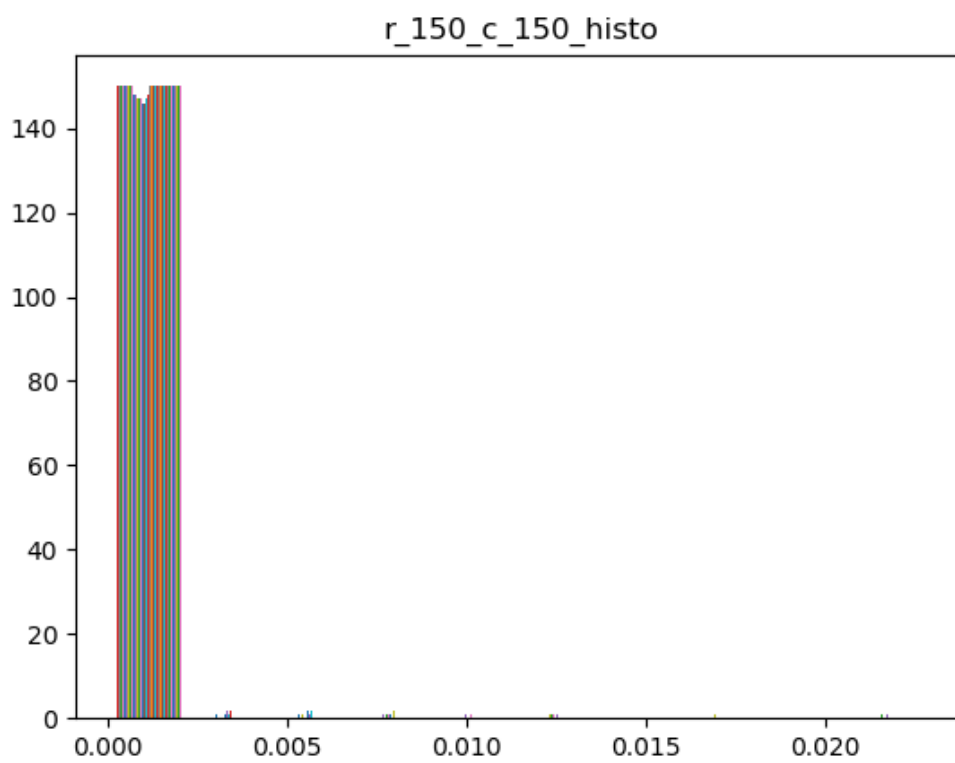
Divide the (H, S) pairs into 256 bins & threshold is 0.01:



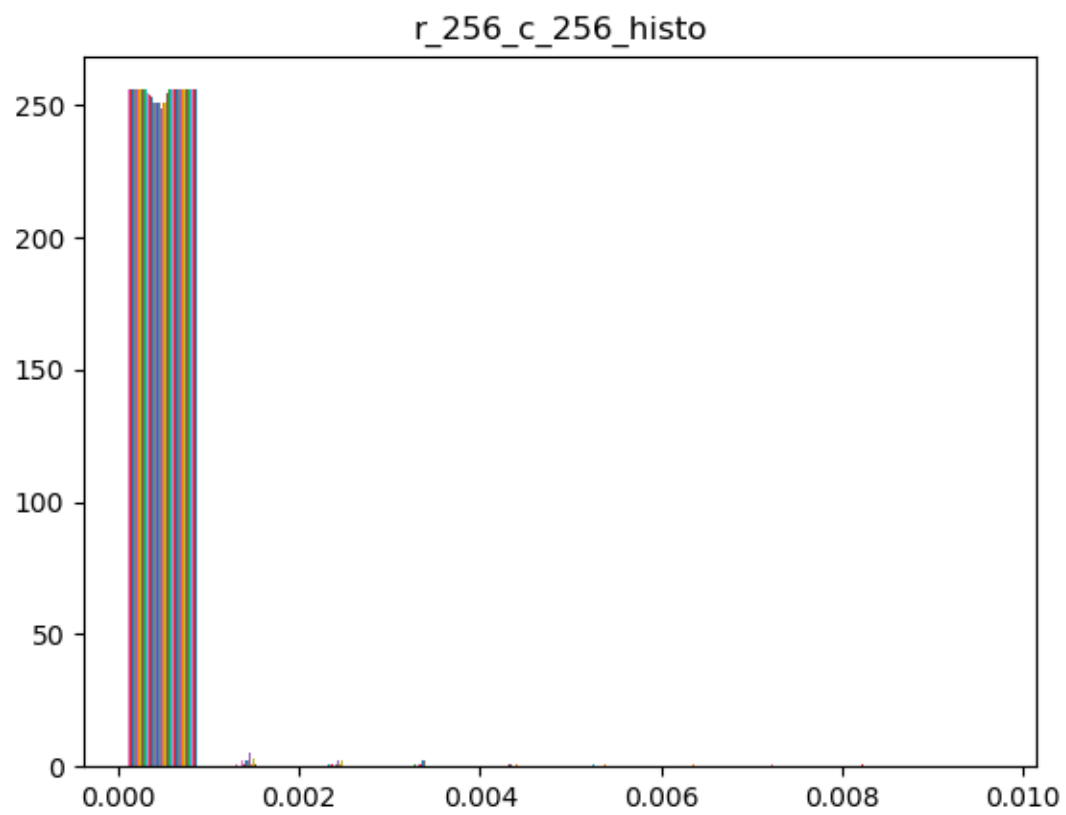
Histogram of 11 bins:



Histogram of 150 bins:



Histogram of 256 bins:



Screenshot of 11-bin histogram 2d array:

0	0	0.008079	0.189159	0.428663	0.34348	0.030619	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0

The 2d array for 150 bins and 256 bins is too large to present here, available from the folder.

As we check the results, the main result is that as the bins increase, the quality of the result decrease, and as the threshold increase, the quality of the result decrease.

The reason, I think, is that if there exist more bins, the distribution of the color will be wider. So, the distribution will be sparser. And less appearance in one bin, cause the less probability. As a result, there will be less color achieve the threshold, so less pieces of skin shown in the final result.

The result of changing the threshold can simply explained, the higher the threshold, the less the point can achieve, so less skin pieces shown in the result.