Arpen Matinyan

Image Processing - Final Project

Project report

Stage1.1

Image 1 - Standard behavior ✔

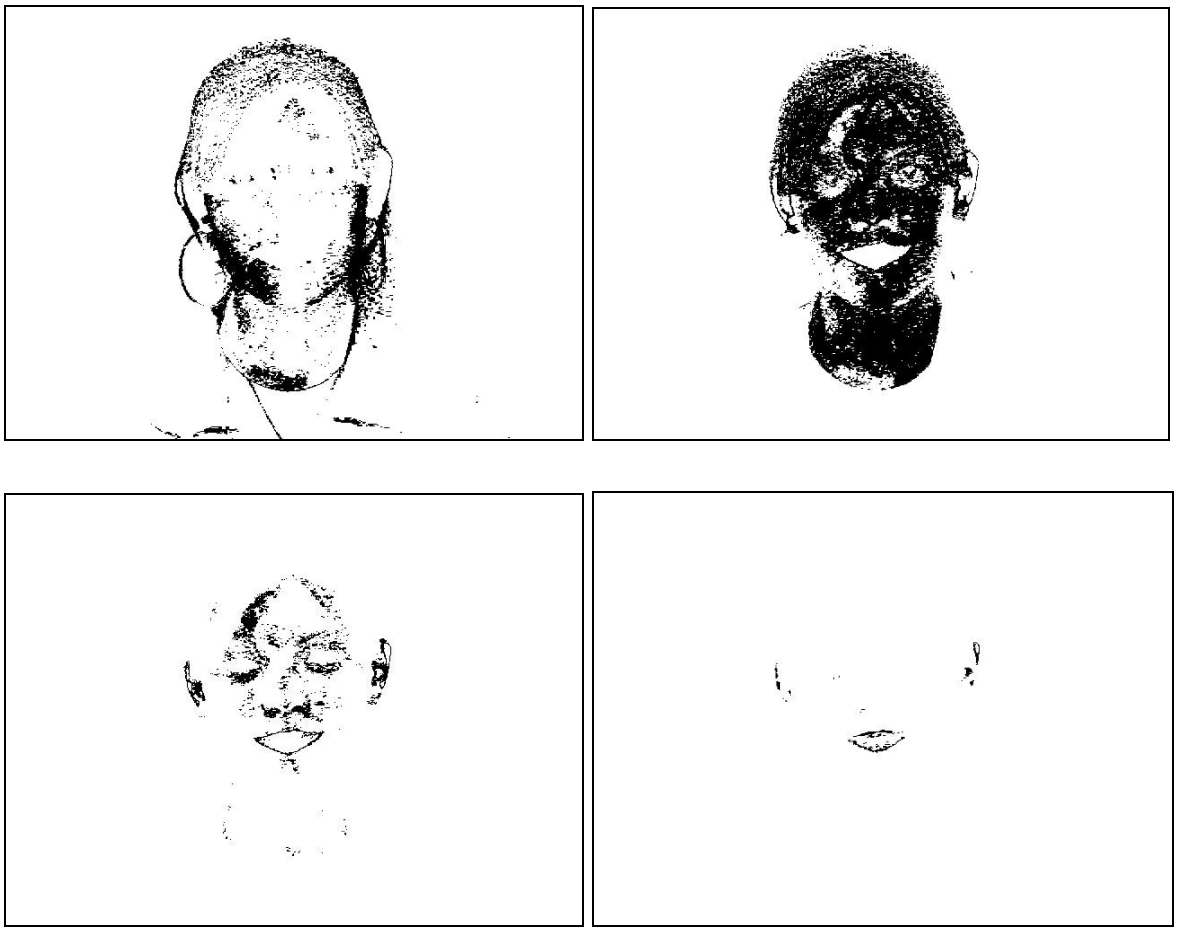


Image 2 - Standard behavior ✔

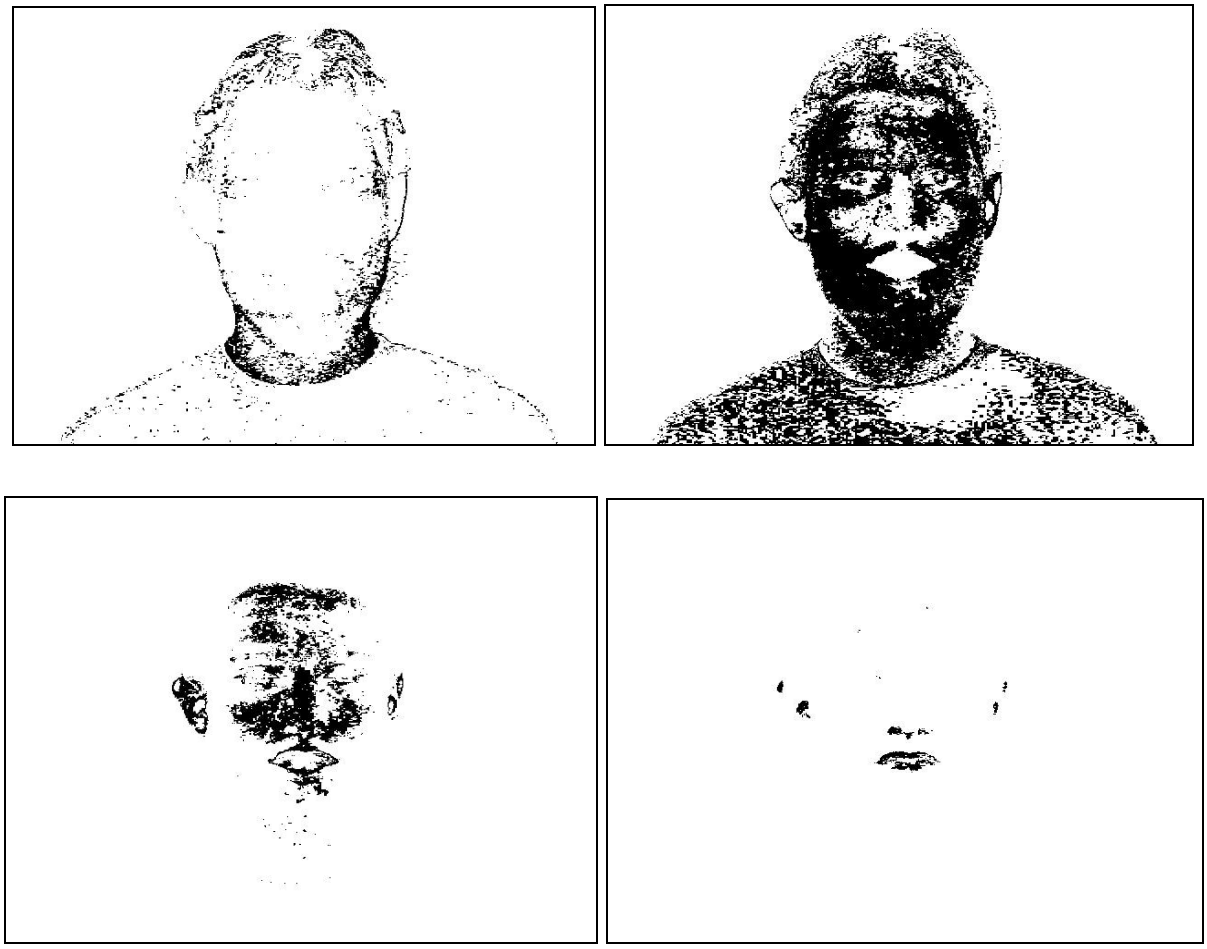


Image 3 - Standard behavior ❌

Binary layer 0 is not selecting the face bounds, instead the skin pixels.

Binary layer 1 is not selecting the skin pixels, instead the face bounds.

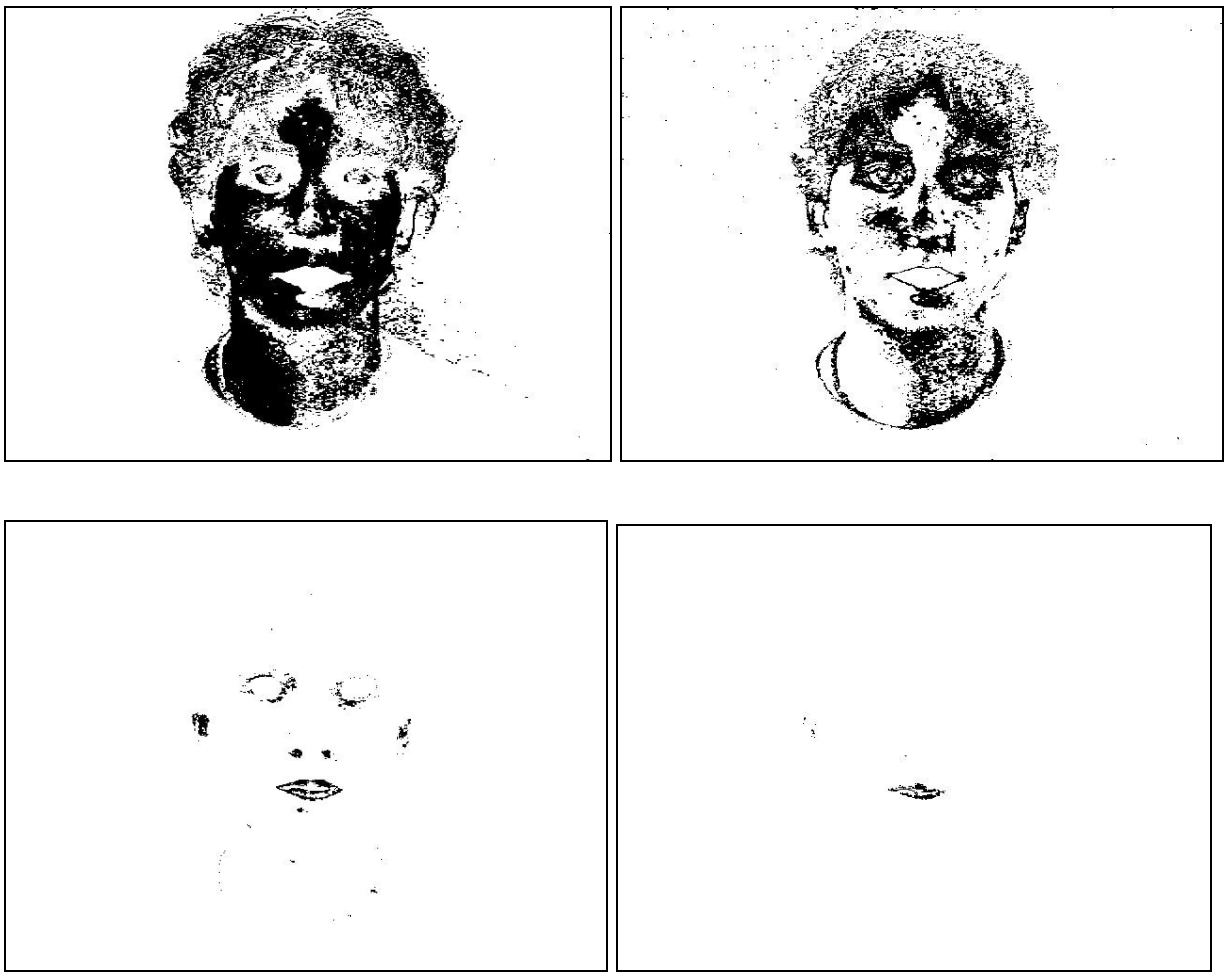
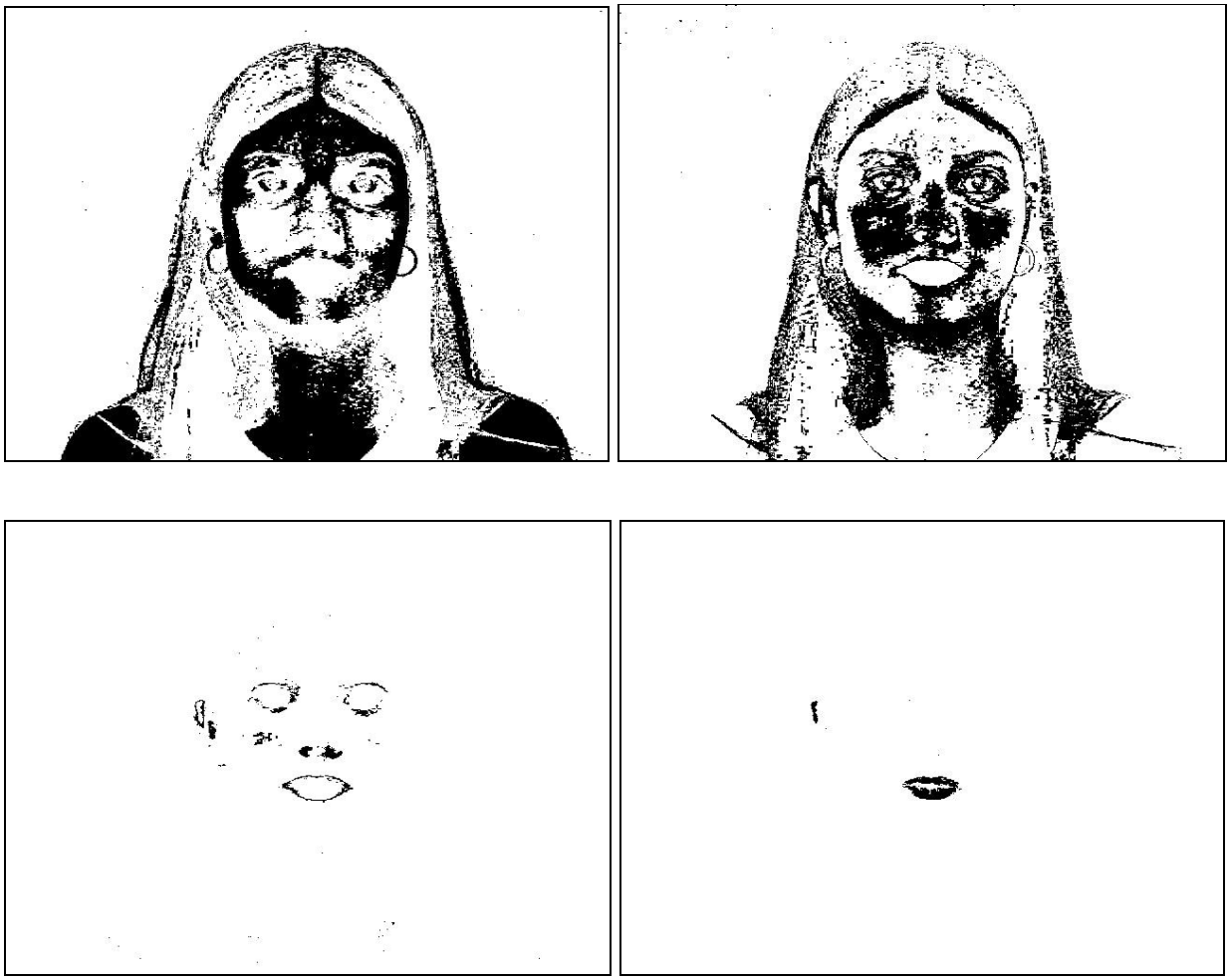


Image 4 - Standard behavior ❌

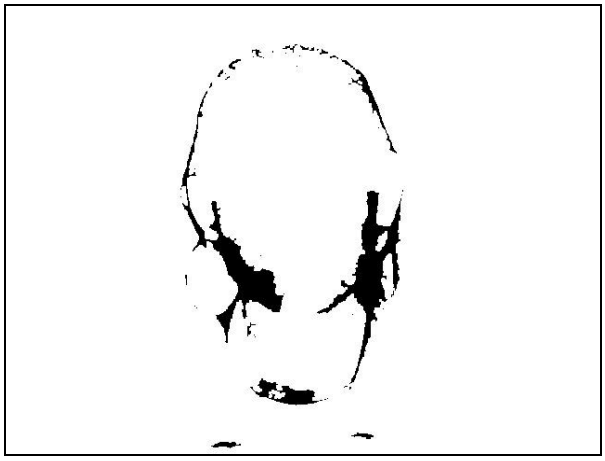
Binary layer 0 is not selecting the face bounds, instead the skin pixels.

Binary layer 1 is not selecting the skin pixels, instead the face bounds.

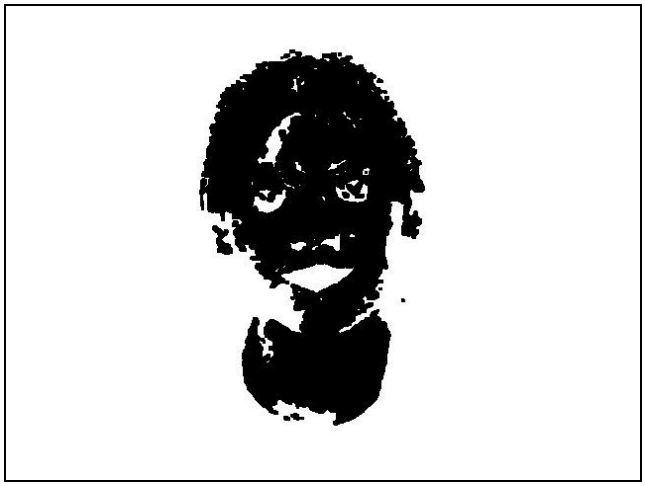


Stage 1-1

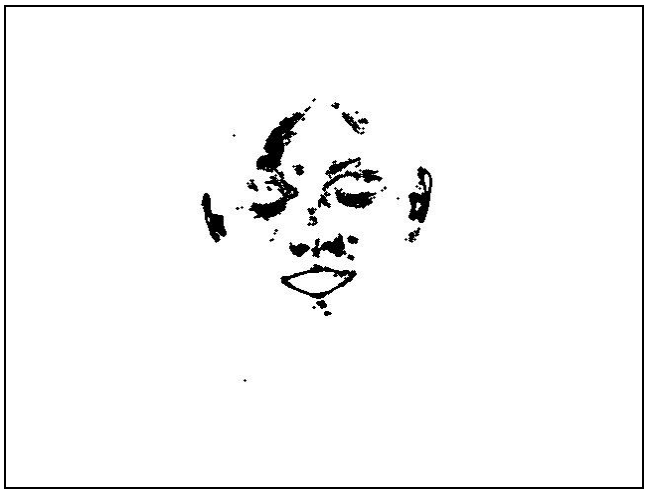
I have noticed that it is better to apply the filters and then the binary layer plugins, to get more accurate results. For example, in this picture after applying the sequence of Mean(3) + BinaryLayer0 + Max(1) I get the most accurate result. When I use Max Filter in the end, the noise from the inside part of the face is eliminated.



For the skin pixels, the best combination that I found was applying Binarylayer1 first, after that median(2) and then smoothing the result with Min(1).



For selecting the central part of the face I have used Median(3) filter + Binary layer 2 plugin + Min(1) in the end to reduce the noise. I have noticed that median filter makes the central part “darker”, that is why applying the Binary filter after that selects a smoothened region.



At last, for selecting the lips and possibly the ears I have used BinaryLayer3 + Min(1) + Median(3). It is important to get the binary image and then apply the filters, because in the opposite case most filters are distorting the image of lips.

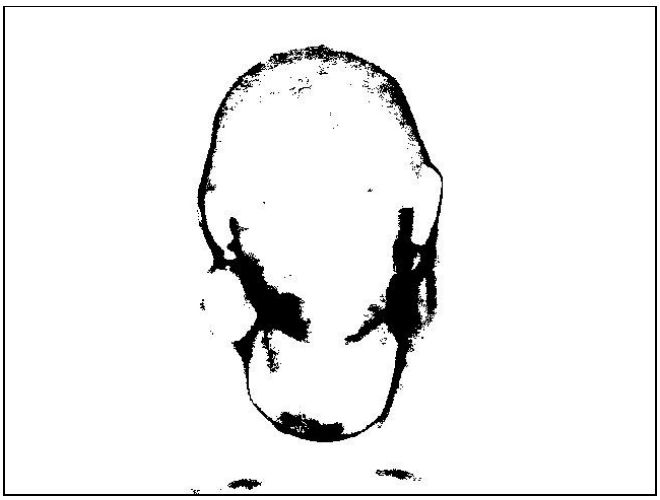
The rest of the pictures having standard behavior are uploaded in the corresponding folder.

Stage 1.2

I have noticed that the bigger the convolutional matrix, the better smoothing I am having. However, for constructing the minimal size matrix I decided that the dimentions for the convolutional matirx need to be at least 7x7.

I have tried different modifications of matrices. The best results were obtained using the following matrix.

```
-5 -5 -5 -5 -5 -5 -5
-5 -3 -3 -3 -3 -3 -5
-5 -3 -1 -1 -1 -3 -5
-5 -3 -1  0 -1 -3 -5
-5 -3 -1 -1 -1 -3 -5
-5 -3 -3 -3 -3 -3 -5
-5 -5 -5 -5 -5 -5 -5
```



The rest of the images are uploaded.
I don't know whether this is a convoltion of different filters, or just a random matrix, I just came up with different modifications and variants.

Stage 1.3

Orientation - For determining the head rotation we can calculate the orientation of the head. I have tried calcutaing the orientation of the head, however I did not get reasonable results. The obtained numbers were mostly randon. Then, I have tried calculating the orientation of specific parts of the face like nose, eyes.

I suppose that better calculations may have results, however I didn not get any during my experiments.

Another idea is binary threshholding the image and calculating the orienttion, hoping that ImageJ will be able to calculate the head rotations. We can combine the binary layers for doing that.

Eccentricity - For determining whether a person is smiling or not, we can calculate the eccentricity of the mouth. If it is elongated, then a person is probably smiling. I have tried cropping only the mouth and calculating its exccentricity. Theoretically, I was assuming to have nice results, however I did not get some.

Maybe some other ways should be developed for understanding the head position in a given image and also whether a person is smiling or not. Some other tools should be developed for that.