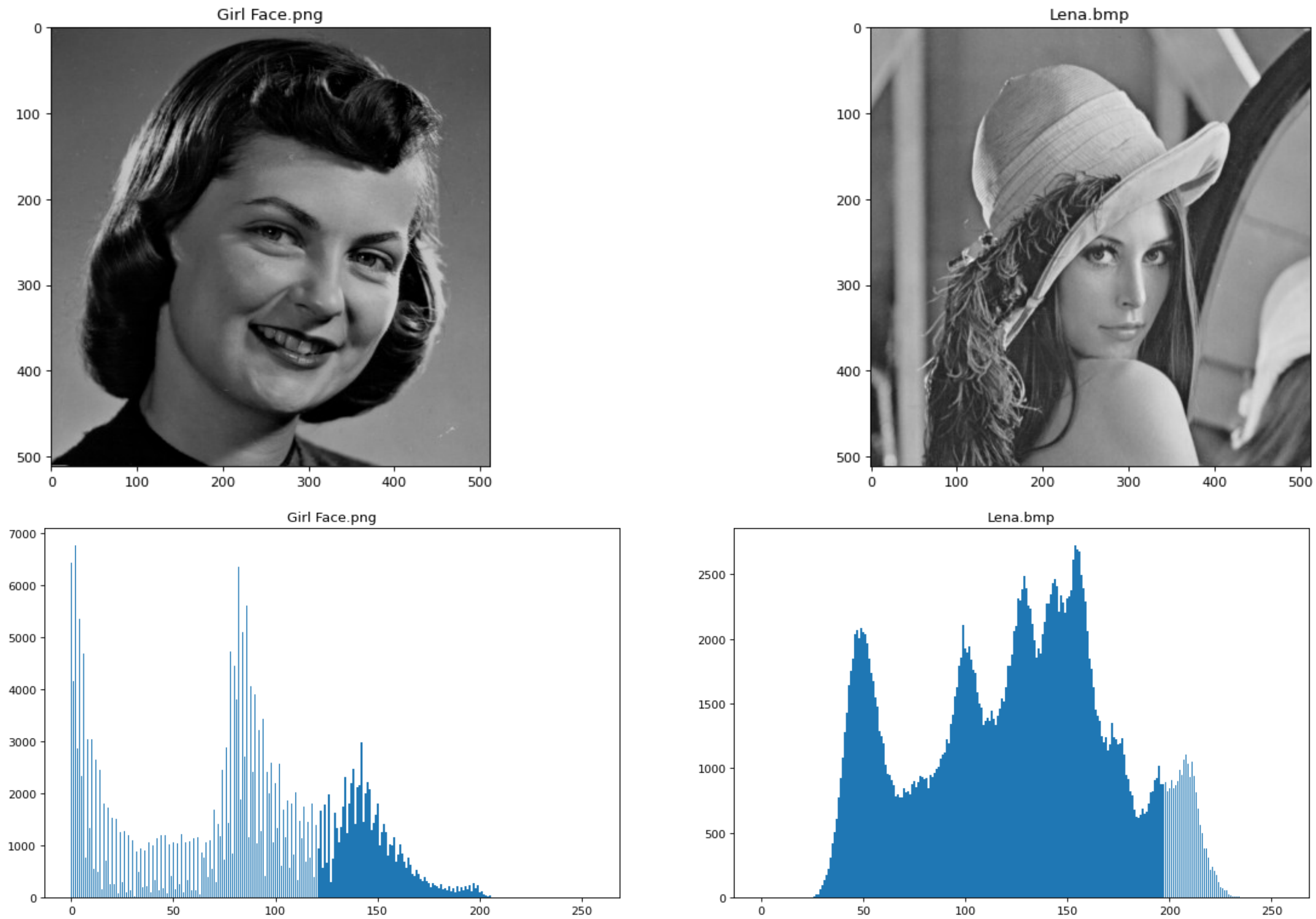
Work 01

Juan Marquina Cancino
0226259

Digital Image Processing I

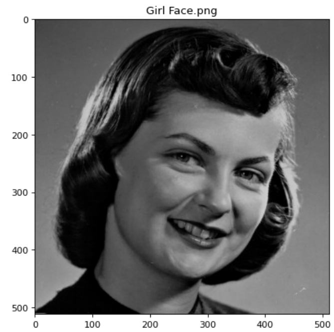
Problem 1.1) Download the Girl Face image and plot a histogram of the grey levels it contains

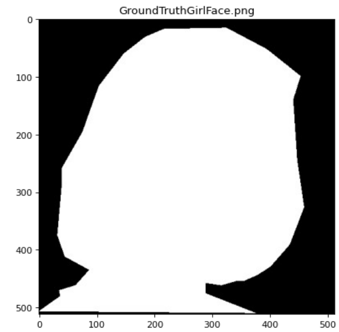


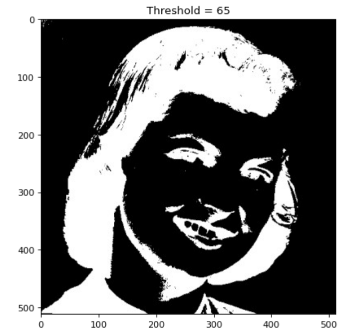
The Girl face histogram has a lot of pixels in the darker spectrum (pixel values from 0 to 5). Also it doesn’t have any pixels with value greater than 200. The image doesn’t have any pixels with value greater than 200. The image doesn’t have a uniform distribution in its histogram and it can be assumed that it could use the range of the spectrum better. A Logarithmic Full Spectrum Stretch might be helpful to improve this image, since it needs to expand the dim values of the image

The Leena histogram is more uniform, is is missing values in ranges from [0-20] and from [220-255]. This image could use a FHSH to improve

Problem 1.2) Write a short program to threshold the image and try to identify a good threshold by trial and error. Create a ground truth segmentation



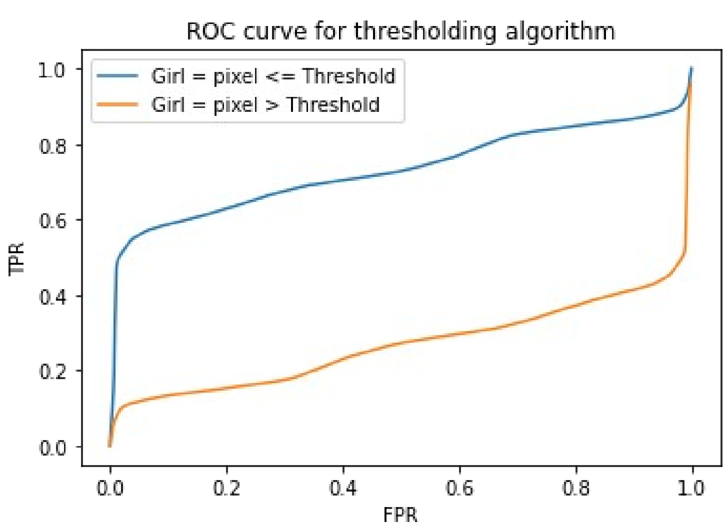


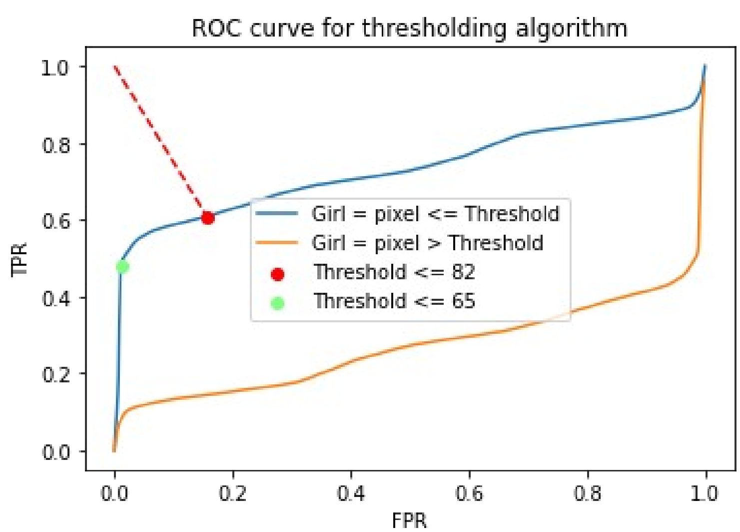


The first set of images, has several thresholds. In each example, values below or equal to the threshold are painted black. Looking at several examples, it is trivial to see that the hair and clothes of the person could be easily segmented using a low threshold. However, the face of the girl and the background has the same distribution of colors, making it impossible to segment the girl from the background.

The second set of images contains a comparison between the original image, a ground truth of a segmentation between the image and the background, and a threshold chosen manually. The threshold was chosen because it is a threshold that doesn’t get any pixels of the background (the false positives of the image is 0). Also it delimits the face of the image.

Problem 1.3) Plot the ROC curve for your thresholding algorithm. What does the ROC curve suggest? You may have to implement an algorithm to find the point closest to a desired Operating Point. How does it compare to your trial and error estimate?

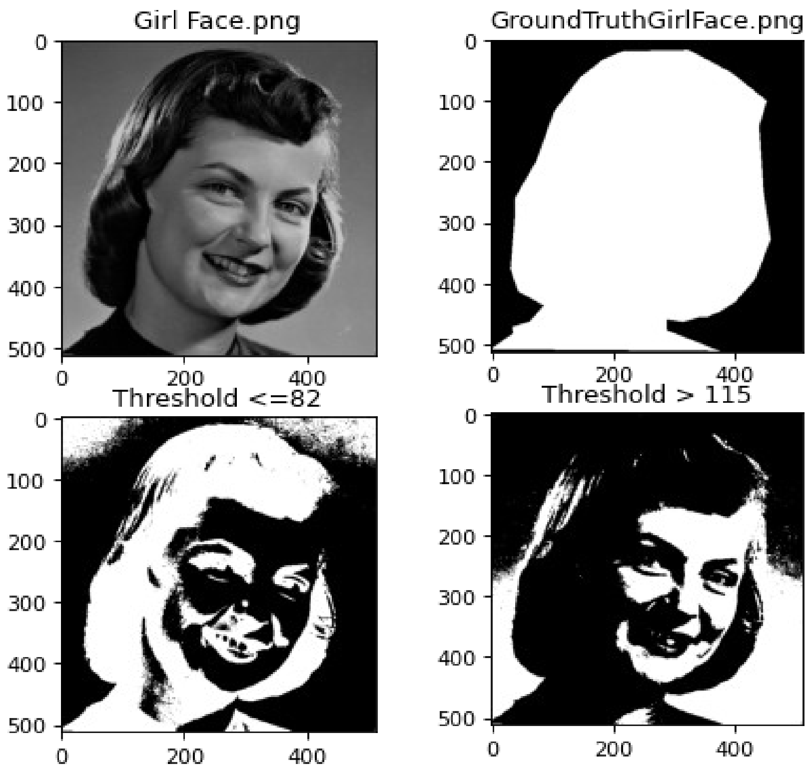




Both figures represent the ROC curve. With the ROC curve we can see that it is always better to select the pixels below our threshold, instead of pixels above our threshold. The ROC curve shows also a critical point, where we start having false positives (we start selecting pixels from the background). That point is where the threshold is 65.

The optimal point of operations is the point closest to (0, 1), since that point has perfect TPR and the lowest FPR. The closest point is where the threshold equals 82. To find that point, we iterated through all the thresholds and measured the distance between the rates of the point and the optimal point (0,1).

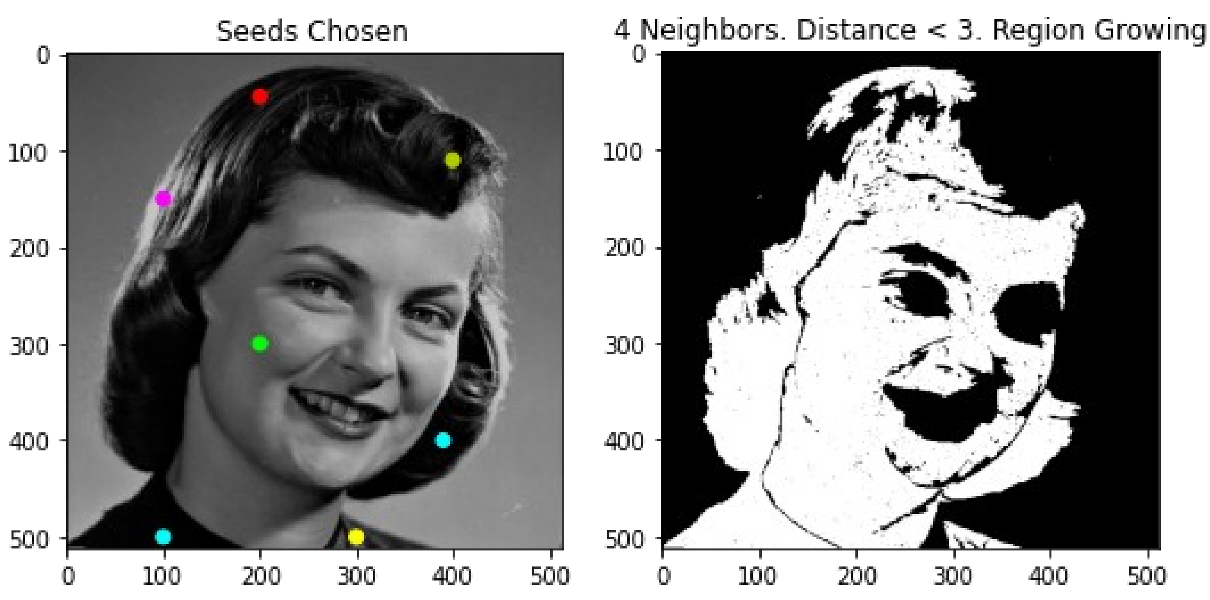
In the graph in can be clearly shown that any threshold greater than 82 increments the number of false positives in a linear manner. Without plotting the image, it can be predicted that the threshold 82 will select 60% of the girl’s pixels, but it will also select a low percentage of the back ground, around 18%.



The image at the bottom left is the image we get after applying the best possible threshold. Even though some pixels of the background are selected, the algorithm chooses a decent amount of pixels corresponding to the correct area.

An observation is that because this algorithm disregards spatial information, we see random areas, such as the corners of this image, selected. Even though there is no spatial relationship between those corners and the rest of the image.

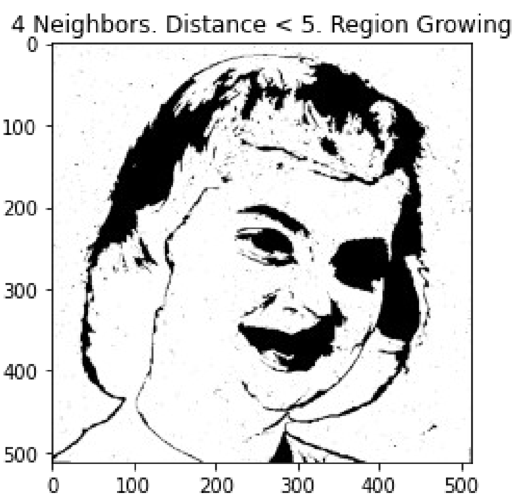
Problem 2.1) Implement a region growing algorithm and compare its performance at segmenting the face in the image above with the performance of the thresholding algorithm. Show the seed you chose.



After some experimentation, I chose 7 seeds for the region growing algorithm. Visually, there are 5 unconnected areas that belong to the girl face, the face, the left part of the hair, the right part of the hair, the right part of the shirt, and the left part of the shirt. The main idea was to choose some seeds in each of the regions.

The main limitation was that it was impossible to make certain regions grow without adding excessive amount of seeds, or using a bigger threshold. For example, increasing the grow threshold, makes the algorithm useless, as it can be seen with the image below. And even in this case, there are areas in the hair that aren't segmented in the algorithm. This areas have the property that in the original image are reflecting a lot of light, therefore the difference of color between one pixel and one of its neighbors is a lot.

2 Seeds, the pink seed, and the light green seed, were chosen with the hope that a region would grow from those seeds. But because of the limitations talked above, that attempt was unsuccessful.



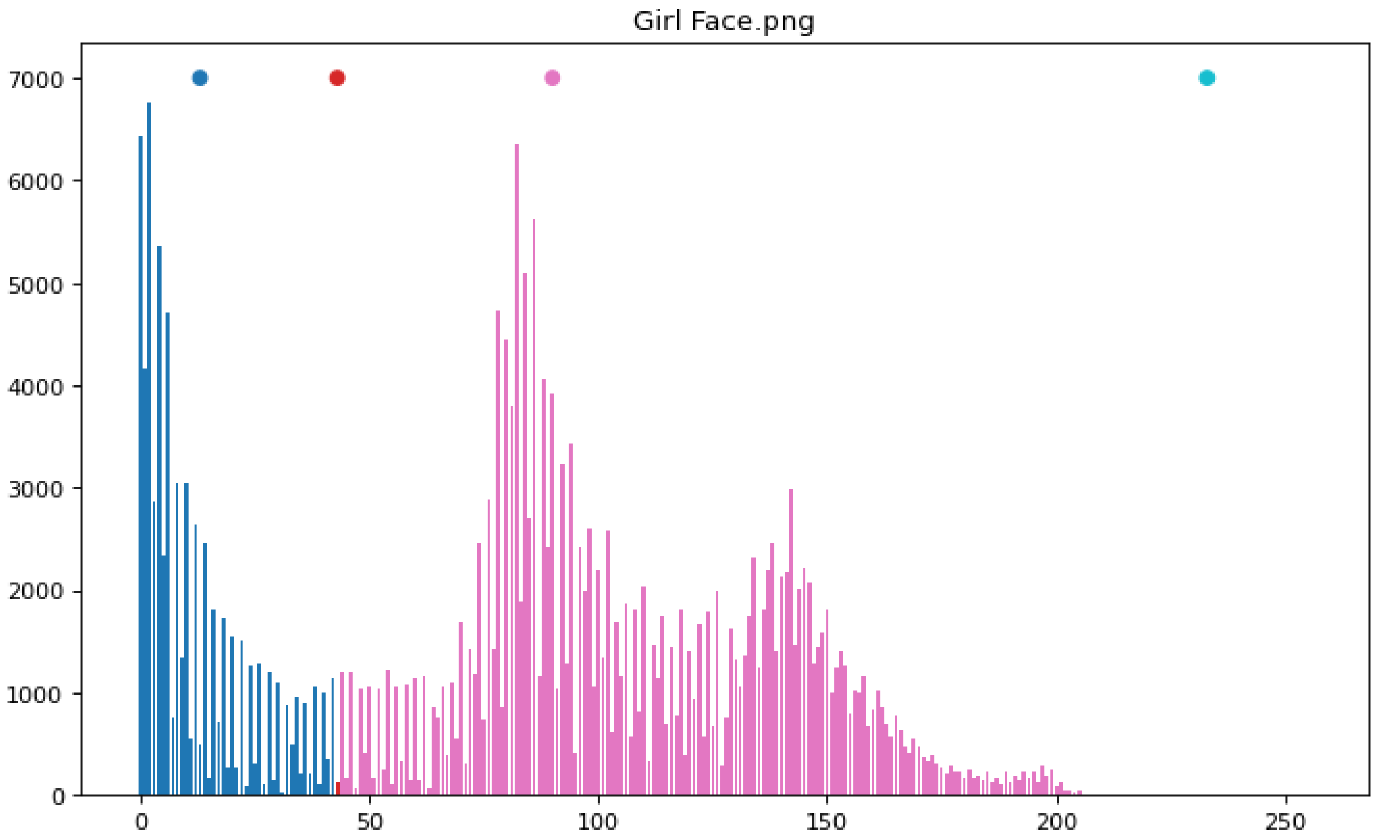

```
Stats for Thresholding Algorithm
tp =  103591
fp =  14395
fn =  66522
tn =  77636
tpr = 0.6089540481914962
fpr =  0.1564146863556845
Distance to optimal point(0,1) = 0.421168
```

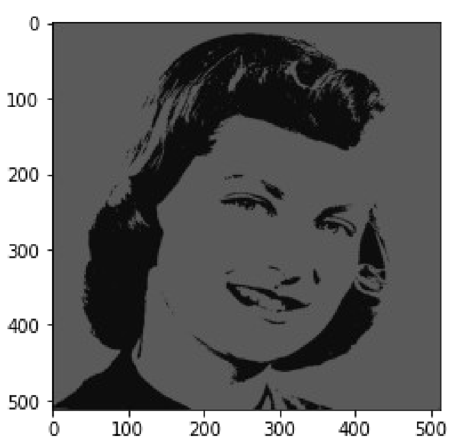
```
Stats for region growing algorithm
tp =  108328
fp =  1014
fn =  61785
tn =  91017
tpr = 0.636800244543333
fpr =  0.011018026534537276
Distance to optimal point(0,1) = 0.363366
```

Even with this limitations, the region growing algorithm performs better than the thresholding algorithm. The region growing algorithm has a near perfect False Positive Rate, and it still has a better TPR.

A key difference between this 2 algorithms is that in the thresholding algorithm, the TPR and FPR could be increased by selecting a more "aggressive" threshold, and both would be increased with a linear rate. In the region growing algorithm increasing the threshold/distance parameter, would make the False Positive Rate = 1, making the result useless. In other words, the TPR of this algorithm cannot be increased further than 0.6368

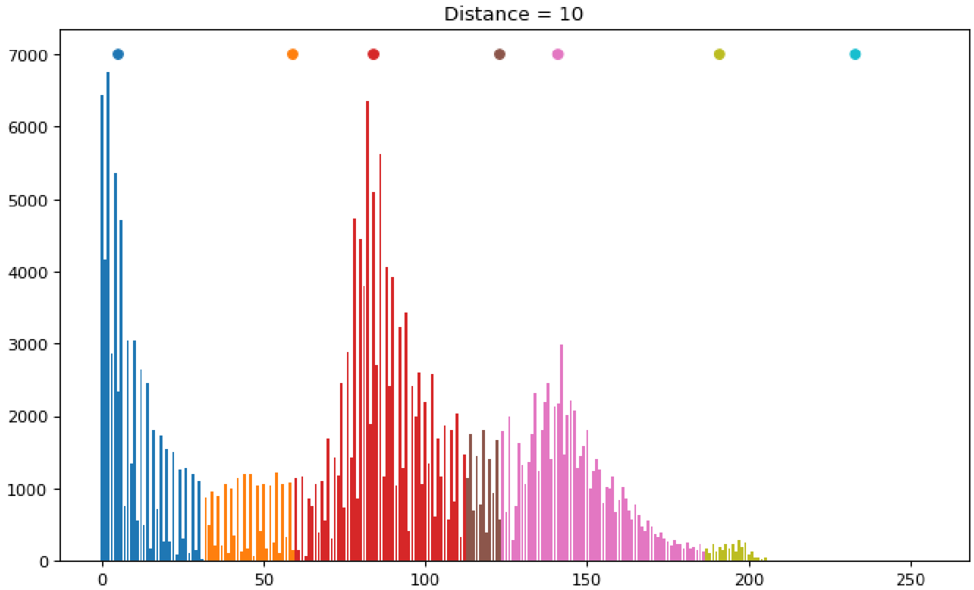
Problem 2.1) Implement the mean-shift algorithm and show the results that you achieve for different values of the “radius” parameter

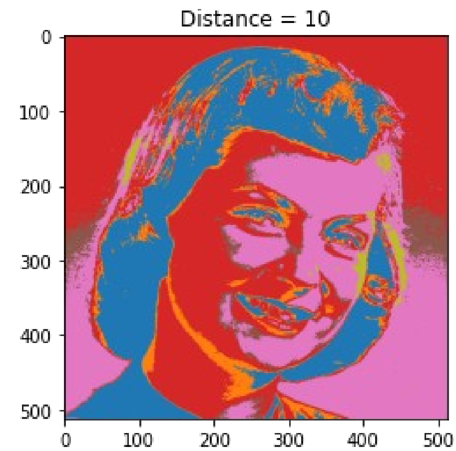




We applied the mean-shift algorithm to the histogram of the image. The result of the algorithm are the local maximums of the histogram and a map of each color, to its corresponding maximum.

Since we are disregarding any spatial information, we are expecting this algorithm to behave like a threshold algorithm, but with more thresholds. For example. The image above is a mean-shift with radius 35. The resulting image it to the left.



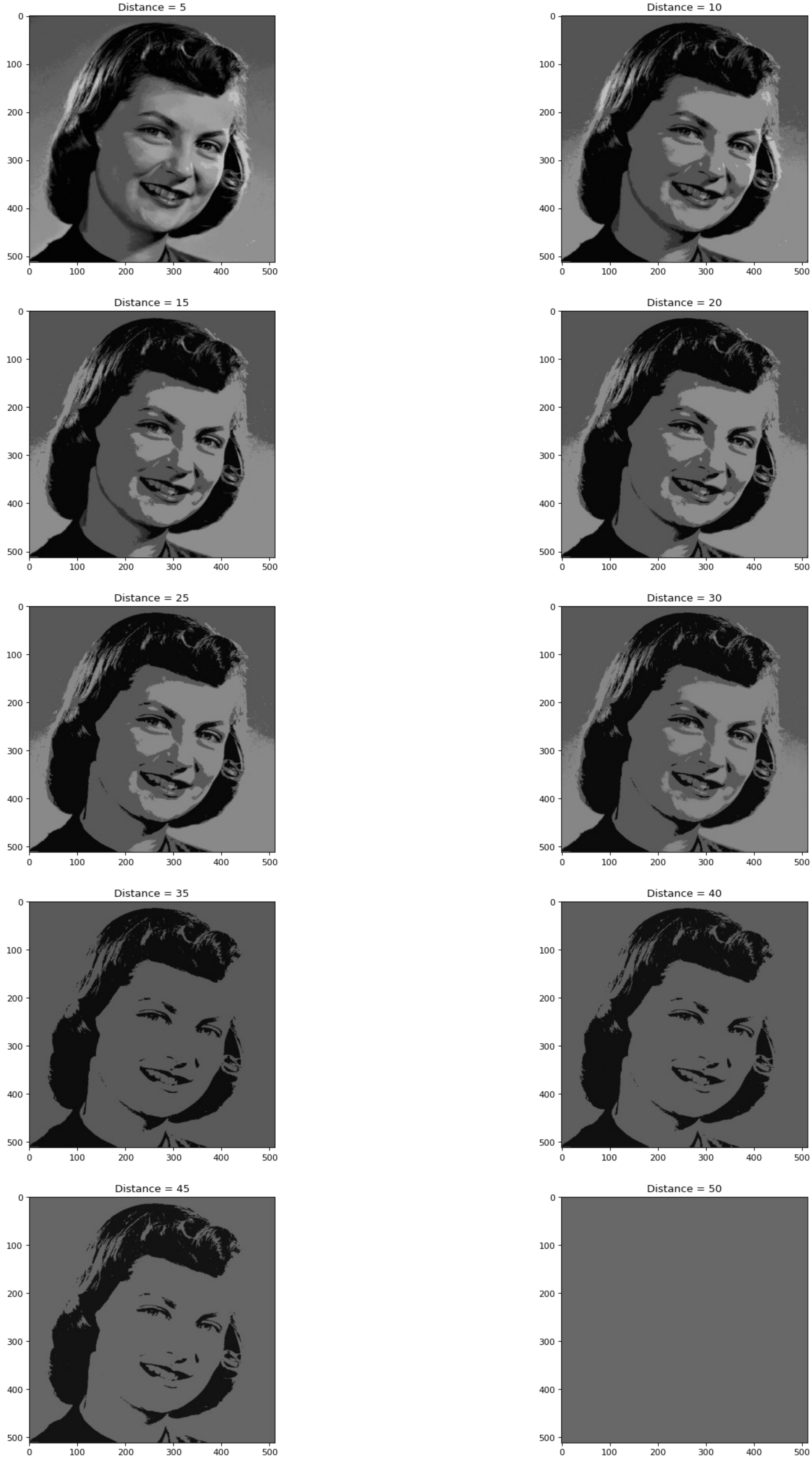


The examples above represent a segmentation using a mean-shift of distance/radius 10. In the histogram, each circle at the topside of the graph represents a local maximum, and each color represents the cluster that each point belongs to. We can see clear segmentation on the distribution of the histogram, however when seeing the final result, we see that face of the girl in the image, is in the same cluster as the background of the image. Not only that, but the face of the image is divided in several clusters.

The problems that this algorithm has, are similar to the problems that the threshold algorithm had. Since we are disregarding any spatial information, and the distribution of the colors in the background are similar to the ones of the face of the person, we cannot segment the image properly.

A solution could be to run the mean-shifts algorithm, but including spatial information such as x,y. However the computational complexity of running this algorithm prevented us from implementing such idea.

Using different radius parameters lead to similar result, as seen in the images below. For this concrete image, using only the color space information is not enough to get good results.


Distance = 5
Distance = 10
Distance = 15
Distance = 20
Distance = 25
Distance = 30
Distance = 35
Distance = 40
Distance = 45
Distance = 50