CMPE362 HW4

Mean Shift Segmentation

Eren Kılıç

Department of Computer Engineering

TED University

INTRODUCTION

In this assignment, we are asked to apply segmentation to the input images by using Lab color space. The assignment contains two parts. In the first part, pixels were represented with 3 features whereas, in the second part depth value was added which makes 4 features in total.

ABBREVIATIONS

- BW – Bandwidth
- Mean Shift – MS
- Res. - result

STEPS FOLLOWED

Part 1:

- Initially, I downloaded the NYU Labeled dataset to extract my input image.
- I loaded the dataset into MATLAB.
- In my script, I extracted an RGB image.
- I converted the RGB color space to Lab color space with MATLAB's built-in *rgb2lab* function.
- I created a matrix X whose rows represent pixels, and columns represent features.
- I normalized the matrix X with *normalize* function. Thus, each of its columns has zero mean and standard deviation one.
- I got transpose of the matrix X using the *transpose* function.
- I applied the mean shift algorithm using various bandwidth values.
- I computed the mean RGB value for each segment by using the *mean* function.
- I reshaped the matrix for visualizing purposes. Each pixel had a mean RGB value of its segment id.
- I converted the output from Lab space to RGB space with *lab2rgb* function.
- Finally, I displayed the result with the *imshow* function.

Part 2:

I followed the similar steps for part 2 but this time as extras.

- I extracted depth information belonging to my input image.
- I added depth info to my feature matrix X. Hence, in total each pixel was represented by 4 features (Lab color values and depth value). I used *cat* function in order to do complete this step.

Test Step:

To check if my scripts are working properly, I used the image in pdf as input (id=171). I compared the results I get with the results in the pdf. In the end, I was sure that my implementation is correct.

RESULTS OF THE PROGRAM

Part 1:

Figure 1: Input RGB image (id: 202)

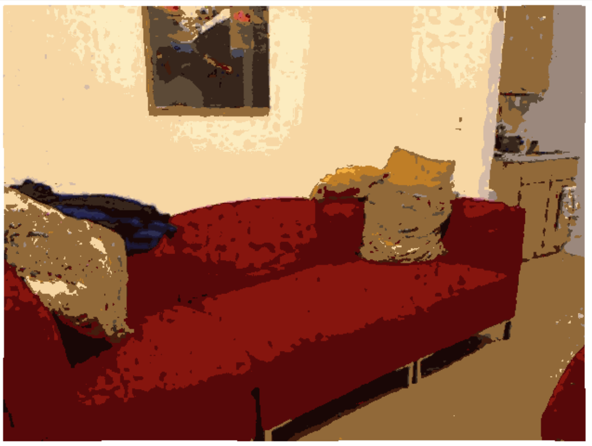

Figure 2: MS Result using Lab (BW: 0.3)

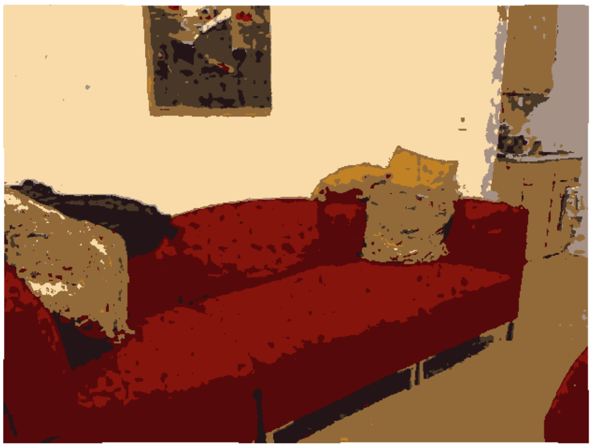

Figure 3: MS Result using Lab (BW: 0.5)

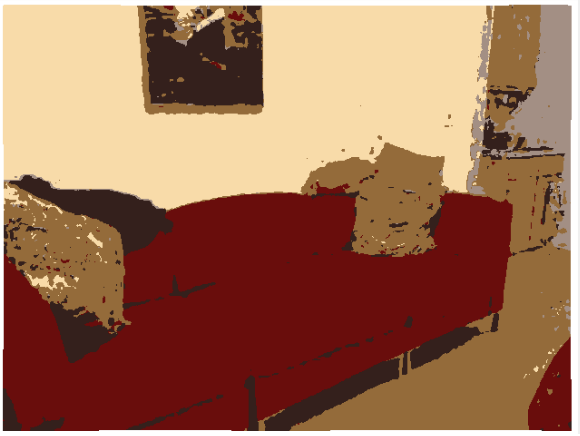

Figure 4: MS Result using Lab (BW: 0.75)

Figure 5: MS Result using Lab (BW: 1)

Figure 6: MS Result using Lab (BW: 1.5)

Part 2:

Figure 7: Input RGB image (id: 202)

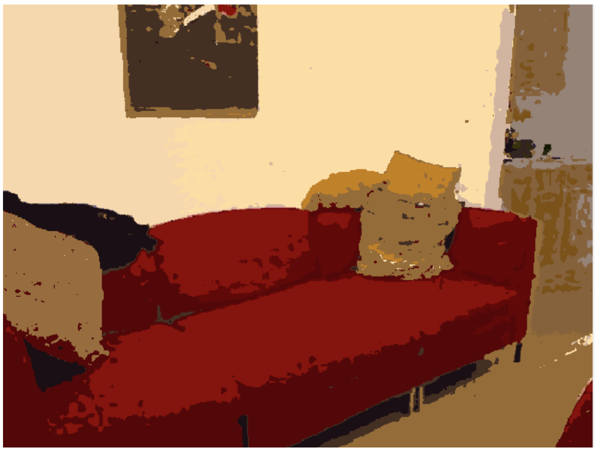

Figure 10: MS Res. with Lab & Depth (BW: 0.5)

Figure 8: Depth information (id: 202)

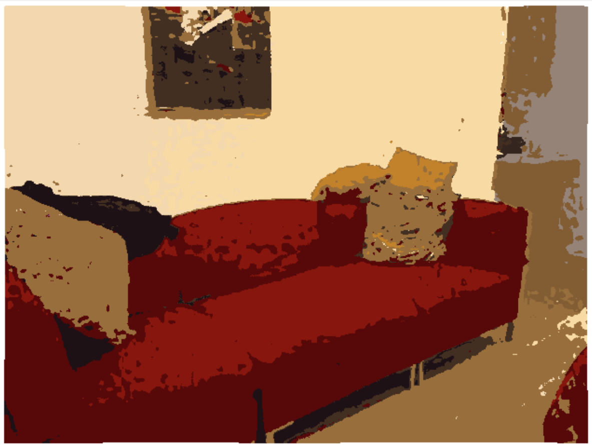

Figure 11: MS Res. with Lab&Depth (BW: 0.75)

Figure 9: MS Res. with Lab&Depth (BW: 0.3)

Figure 12: MS Res. with Lab & Depth (BW: 1)

Figure 13: MS Res. with Lab & Depth (BW: 1.5)

DEDUCTIONS

Part 1:

If we check outputs from figure 2 to figure 6, we see that the outputs are generated with different bandwidth values. The bandwidth values are increasing in order starting from 0.3 and end at 1.5.

We can conclude that, if we increase the bandwidth value, the number of clusters coming from the mean shift algorithm decreases as the outputs proved. For example, number of clusters for bandwidth 0.3, 0.5, 0.75, 1, 1.5 are 65, 15, 6, 5, 2 respectively. Although bandwidth values do not increase that much, the number of clusters decreases noticeably. It is because of that mean shift works with a window.

The number of clusters coming from the mean shift algorithm affects segmentation results. For instance, in figure 5, the bandwidth value is 1 and we have 5 clusters. This means that the image is segmented with 5 different colors, as it can be seen in the image. If we increase the bandwidth value and make it 1.5, the number of clusters will be 2 and we will have just 2 different colors which will cause poor segmented image. Similarly, if we have low bandwidth such as 0.3 in this example, the image will be over segmented. Hence, we should decide the bandwidth value carefully, based on our purpose.

Part 2:

Like part 1, in part 2 if we check the outputs from figure 9 to figure 13, we see that:

- If the bandwidth value increases, the number of clusters coming from the MS algorithm decreases.
- The more clusters mean, the more colors we have in segmentation.
- We notice a sudden decrement on cluster despite regular increment on bandwidth values.
- A low BW can cause poor segmentation like figure 9 (over segmented) and high segmentation could be the reason for poor segmentation as well.
- We have fewer colors on higher bandwidth valued images which affect segmentation result directly.

However, the only difference in this part was related to depth information as can be seen in figure 8. If we compare the outputs from part 1 and part 2 whose bandwidth values are the same, we can see how depth information has an effect on segmentation results. We realize that when we add depth information, we obtain more clusters from the MS algorithm which causes more colors on segmentation. For example, in figure 5, the wall of the kitchen was not segmented. Yet, in figure 12, it is segmented with gray color even though the bandwidth values are the same. This situation makes depth value useful. Please see figure 8 where the kitchen’s wall located. (it is white)

Please note that the computer calculates higher bandwidth values faster. Thus, using depth value information could be useful for a high-resolution image or a sized dataset.

APPENDICIES

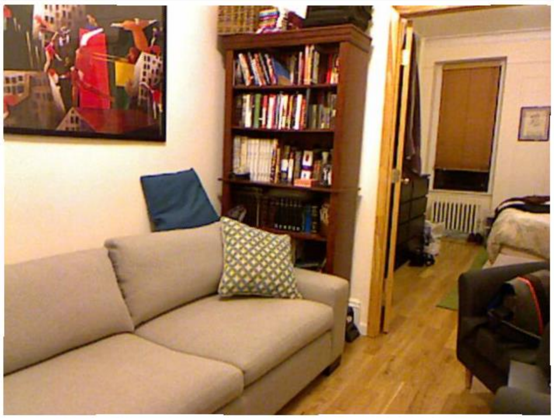

Figure 14: Input RGB image (id: 171)

Figure 15: MS Result using Lab (BW: 0.3)

Figure 16: MS Result using Lab (BW: 0.5)

Figure 17: MS Result using Lab (BW: 0.3)

Figure 18: MS Result using Lab (BW: 0.3)

- The images above are from my program and they were used to check if my implementation is correct.