ASSIGNMENT 3

YONGJIN SHIN
20090488
Industrial Management Engineering, POSTECH
dreimer@postech.ac.kr

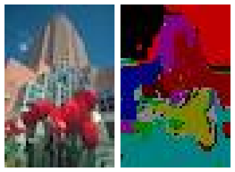

Figure 1. Result of test sample of small size
(left: original, right: 1/8 from original)

1. PROBLEM 1

Since two sets of faces have same co-variance matrix, we can assume that their distribution are quite similar. Furthermore, the results of projection toward lower dimensions, regarding to both PCA and LDA, are nearly same in order to reduce the error. For solving this problem, the proof of proposing that the following is needed

$$S_{PCA} = S_{within} + S_{between}$$

Unfortunately, I was not be able to finish the work because of limited time, so that I just commented the idea of solving.

2. PROBLEM 2

2.1. Result

At first I tried to test out the normalized cut without any attempt to reduce the number of pixels. As the result, the output was made like Figure 1. According to Donghui Yan, there is a way of fast spectral clustering with using K-means clustering. The result are shown as Figure 2 and Figure 3.

2.2. Discussion

2.2.1 Different Weights

Since Lab color space was used, there were different weight to be needed. Though there possibly are two segments of different brightness, as far as they have similar color (a dimension, b dimension, respectively), then I considered they

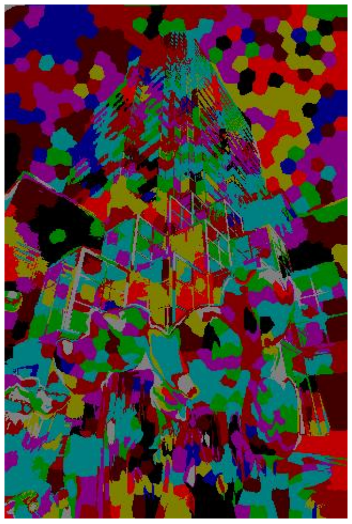

Figure 2. Result of test sample of small size
(Elapsed time: 105.22 seconds)

are the same. Since this reason, I put less weight on the brightness (L dimension) so the difference between three dimensions are 1, 3, 3. However, empirically, there are too much gap between three, the segmentation work was getting worse. Figure 1 was the best output of the algorithm.

2.2.2 Reduction Nodes

As implying the algorithm without reduction of nodes takes too much time, we have to consider the way to make small nodes without affecting the result.

I cited the paper, called 'Fast Approximate Spectral Clustering' written by Donghui Yan in 2009. He suggested following algorithm.

Input: n data points ${x_i}_{i=1}^{n}$, number of representative points k
Output: m-way partition of the input data
1. Perform k-means with k clusters on $x_1, ..., x_n$ to:

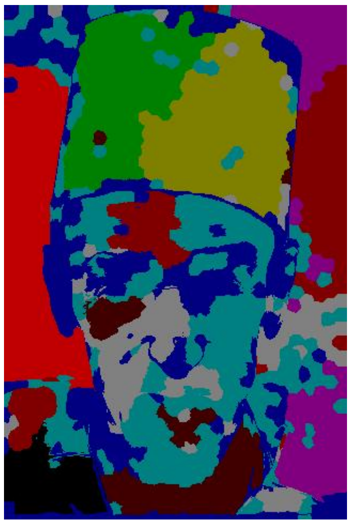

Figure 3. Result of test sample of small size (Elapsed time: 100.93 seconds)

a) Compute the cluster centroids $y_1, ..., y_k$ as the k representative points.
b) Build a correspondence table to associate each x_i with the nearest cluster centroid y_j.
2. Run a spectral clustering algorithm on $y_1, ..., y_k$ to obtain an m-way cluster membership for each of y_i.
3. Recover the cluster membership for each x_i by looking up the cluster membership of the corresponding centroid y$_j inthecorrespondencetable.$

The idea is that we only pick 1/100 of pixels from the original data by using kmeans clustering. Since we should consider the distance between points including both feature and spatial difference, we have to concatenate the image data(L,a,b) with x,y coordinates. After getting the smaller centralized data, we could do same job with the small pictures. After finishing the normalized cut, we can recover the membership of each centralized data toward the whole image pixcels.

As the result, the segmentation time was impressively short, however, the quality is a little worse. In figure 3, the output was not bad. But, figure 2 cannot be said as the good result. Because of limited time, I could guess some clue to improve the quality. First, there should be some mistakes to recover the cluster membership. Both figure 2 and figure 3 have same drawback which is that the clustered blocks are scattered. As recover the membership, I needed to trace back to find the indices of all pixel values. However, the size of array is way too huge that I couldn't check the accuracy of inside the loop. If the traceback work would be okay, secondly, the threshold to accept the useful affinity is too low. Even though I have checked the size of r (if two points are far over "r", then it doesn't have any affinity value), there might be some possible case for two points in same cluster with long distance.

2.3. Reference

[1] Donghui Yan, Ling Huang and Michael I. Jordan. Fast approximate spectral clustering 15th ACM Conference on Knowledge Discovery and Data Mining (SIGKDD), Paris, France, 2009.

[2] Ulrike von Luxburg, A Tutorial on Spectral Clustering, Max Planck Institute for Biological Cybernetics Spemannstr. 38, 72076 Tubingen, Germany, 2007