ETH ZURICH, D-INFK

Computer Vision Exercise 9: Image Segmentation

Xingchang Huang

December 21, 2017

1 IMAGE PREPROCESSING

In the first step, what we need to do is to filter and change the color space of given image from RGB to L*a*b* color space, which is designed to approximate human vision and its L component closely matches human perception and therefore brings better segmentation result. The filter step is quite easy in MATLAB using imfilter function and fspecial to define the gaussian kernel. The size of gaussian filter is 5×5 and standard deviation is 0.5. Then apply the color transform from RGB to LAB based on the smoothed image. After that we would reshape the image and the later computation (e.g., distance) are based on the new color space instead of RGB or distance on coordinate. Basically I use the original size of image instead of using the image resize function for tuning the parameters. I found that if the radius and threshold of mean-shift segmentation method is defined appropriately, the convergence rate would be faster.

2 MEAN-SHIFT SEGMENTATION

Basically, this algorithm starts with selecting a pixel and then do mean shift based on its surrounding pixels inside a circle with radius r. Repeat the mean shifting step for each pixel and finally we can generate peaks, whose number is the same as the number of pixels. After that, we can merge those peaks within distance of $r/2$. Then we can not find the clusters that based on the distance of pixel to the peaks. For me, I have recorded the pixels during the path of mean shift to their corresponding peaks, and set the state of these pixels to visited. In this way, those pixels on the way may belong to this peak recorded by counting on all the

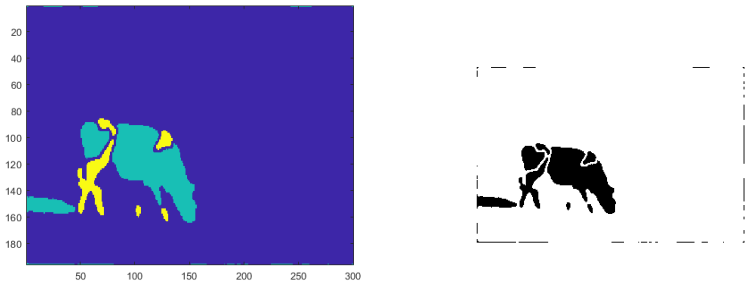

Figure 2.1: Mean-shift segmentation: radius=30, bandwidth=0.3.

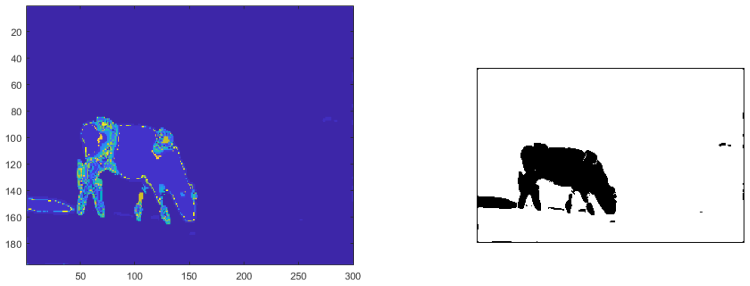

Figure 2.2: Mean-shift segmentation: radius=5, bandwidth=0.1.

peaks. And as they are visited, we can start from unvisited pixels in the next iteration, which can largely reduce the complexity of computing each pixel's mean shift. At the same time, if a pixel belongs to a peak with most frequency (counting), it should belong to that peak. The result of mean shift is shown as the following with different values of radius and threshold (bandwidth). The first result should be better with 3 peaks after merge. The three obviously different colors have been apart but some parts of the cow still be wrongly considered as the background. The second one's radius is much smaller and the final number of peaks have reached over 70, which does not make sense because the original image does not have too much colors and the leg has been considered as the background.

I suppose that large radius could have faster convergence because each pixel can move faster and the threshold for merge ($r/2$) would be larger so that the number of clusters can converge to a reasonable value.

3 EM Segmentation

The EM segmentation method is based on GMM model, which can predefine the number of gaussian distribution and cluster the pixels to the most probable gaussian distribution. So there are mainly two steps, the first one is to calculate the probability that a pixel belongs to

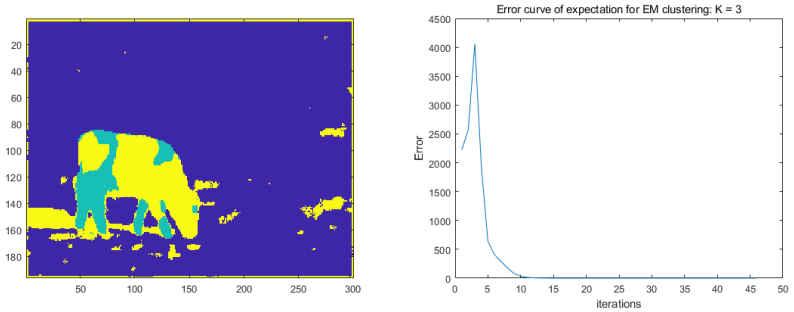

Figure 3.1: EM segmentation: k=3, iteration=51.

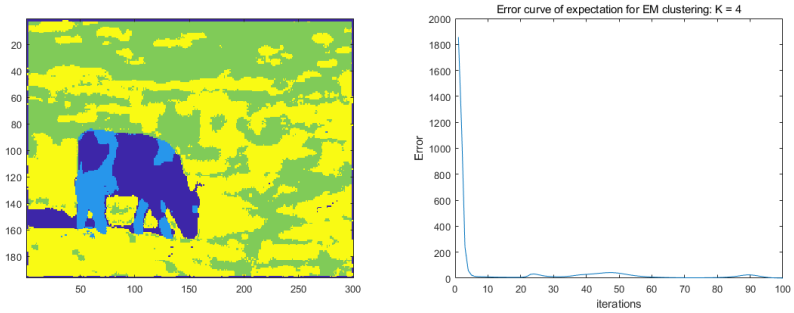

Figure 3.2: EM segmentation: k=4, iteration=100.

all the k clusters $p(z_k|x_i)$, where k is $3,4,5$ and i is the index of pixels. Based on the computed expectation, we can update the parameters in θ to maximize the expectation, which is also a gradient-based method for updating the θ. By iterating two steps, I use the convergence of calculated expectation over all pixels with a small threshold to terminate the iteration.

What we do here actually is to find k gaussian distributions and the pixels belonging to them iteratively following the equations. The initialization of the mean, covariance are randomly selected from k pixels using their LAB values and the their corresponding diagonal matrix. The results are shown as the following under different setting of k. Also, I plot the error curve of expectation, which shows that the convergence rate is faster when $k = 3$.

The threshold for stopping is 0.01 here and the max iteration of this experiment is 100 and you can see that the iteration when $k = 3$ is around 50, which is much less than the number when $k = 4,5$ and $k = 3,4$ give the better result because the original image have mainly 3 or 4 different colors considering the grass and the cow itself. The final values of θ are listed in the params file and the following matrices shows the parameters when $k = 3$.

$$\alpha = \begin{bmatrix} 0.042 & 0.8080 & 0.15 \end{bmatrix}, \mu = \begin{bmatrix} 131.9122 & 124.6612 & 141.1629 \\ 89.4226 & 114.4455 & 149.0930 \\ 48.6125 & 121.3320 & 139.4786 \end{bmatrix}$$

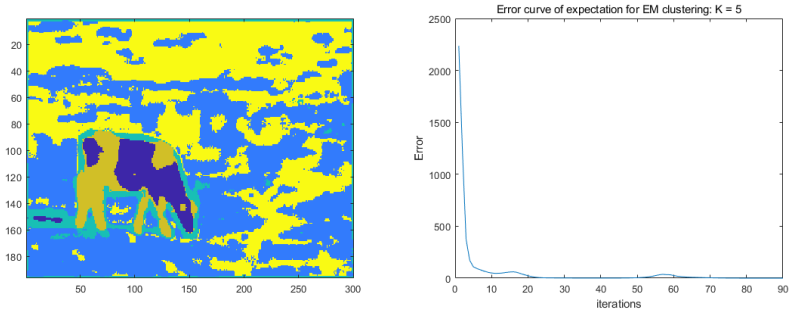

Figure 3.3: EM segmentation: k=5, iteration=90.

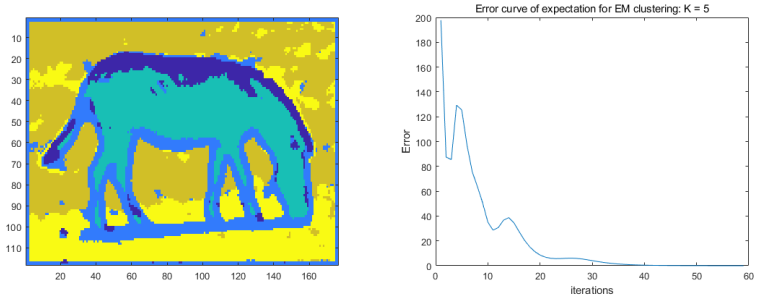

Figure 3.4: EM segmentation on zebra: k=5, iteration=59, scale=0.3.

$$\sigma(:,:,1) = \begin{bmatrix} 2.4225 & 0.0897 & 0.0116 \\ 0.0897 & 0.0142 & -0.0106 \\ 0.0116 & -0.0106 & 0.0280 \end{bmatrix}, \sigma(:,:,2) = \begin{bmatrix} 53.0283 & 0.0350 & 0.5146 \\ 0.0350 & 0.7797 & -0.1604 \\ 0.5146 & -0.1604 & 1.5199 \end{bmatrix}$$

$$\sigma(:,:,3) = \begin{bmatrix} 721.1648 & -140.4151 & 216.4471 \\ -140.4151 & 36.4980 & -48.6405 \\ 216.4471 & -48.6405 & 73.1736 \end{bmatrix}$$

As there are too many parameters for different configurations, I have listed them in the "params" file, which contains the results of parameters, including α, μ and σ when $k = 3,4,5$, respectively on this cow image. I have also tested the EM segmentation method on the zebra image. I set $k = 5$ for prior knowledge and resize the image by a factor 0.3, which makes the running of EM algorithm faster. The result is shown in the above figure and after 59 iterations, the error of expectation is smaller than the threshold 0.01.

In conclusion, I think these two algorithms have the same drawback that the model simply consider the difference of color instead of the semantics. In this way, it may happen that for example, some parts of cow are considered the background because of their similarity in color space.