ECE 549 Homework 4

Fangwen Wu (fwu11)
April 10, 2017

1 SLIC Superpixels

a) The distance function is given as:
$D = \sqrt{(d_c^2 + (\frac{d_s}{S})^2)m^2}$
which is a combination of the color distance d_c and spatial distance d_s, the distance between cluster center S and the compactness factor m allows us to weigh the relative importance between color similarity and spatial proximity.
$d_c = \sqrt{(r_j - r_i)^2 + (g_j - g_i)^2 + (b_j - b_i)^2}$
$d_s = \sqrt{(x_j - x_i)^2 + (y_j - y_i)^2}$

b) We fix number of cluster centers to K = 128 and we choose 3 different weights: 25,100,150

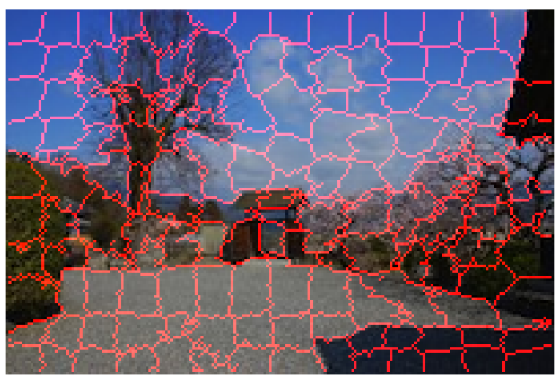


Figure 1: m = 25

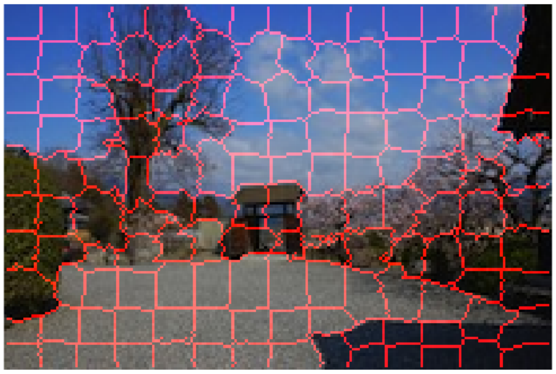


Figure 2: m = 100

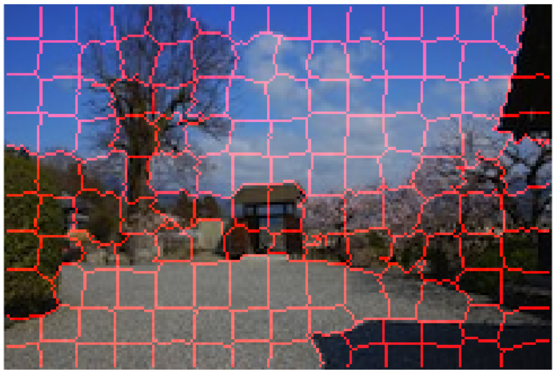
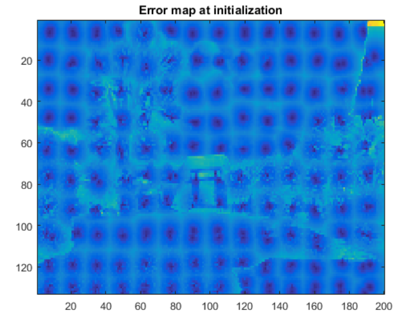
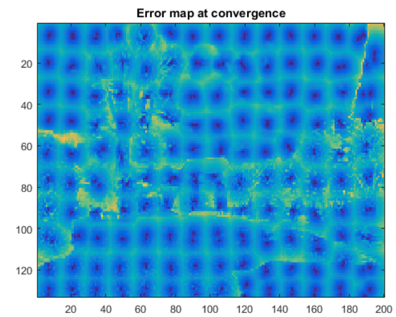


Figure 3: m = 150

When m is large, more weights on the spatial proximity and hance the segmentation is more like rectangular grids. When m is small, the segmentation depends more on the color similarity and the pixels that share similar colors are grouped together.

c)Here we show the error map at initialization and at convergence with the setting of K=128 and m =100





d) Here we show the segmentation result with m=100 and K=64,256,1024. Runtime for K = 64 is 0.4486s, K = 256 is 0.6592s, K = 1024 is 1.6962s

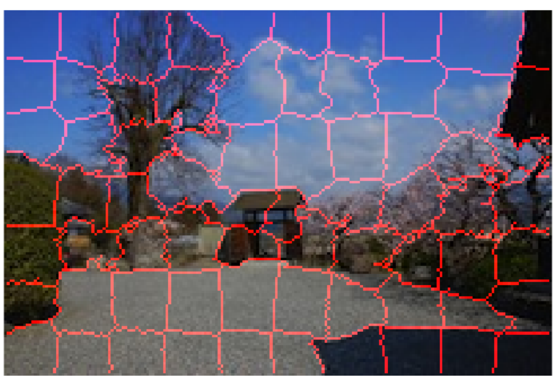


Figure 4: K = 64

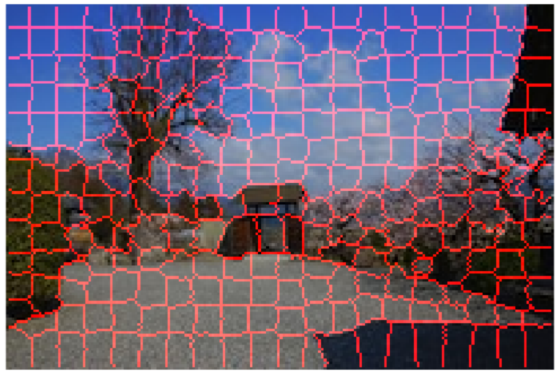


Figure 5: K = 256

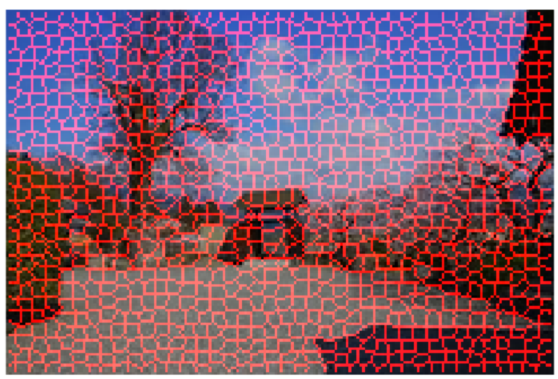


Figure 6: K = 1024

e) The following three images show the test results on the BSD when K = 256:
Average boundary recall = 0.733652
Average undersegmentation error = 0.117846

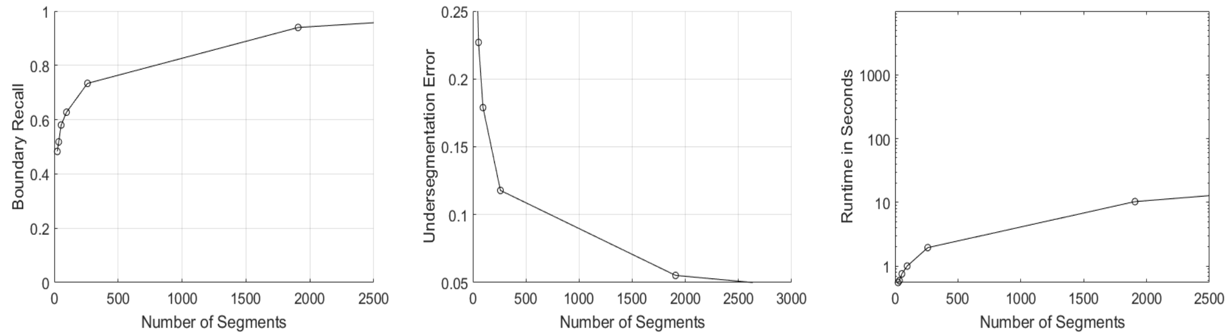


Figure 7: RGB color space evaluation results

f)[extra credit] I tried the CIELab color space and the boundary recall is 0.7921 and under-segmentation error is 0.1141 with K=256. The result is better because Lab color space exceeds the gamuts of the RGB color model hence it is a better color model for segmentation.

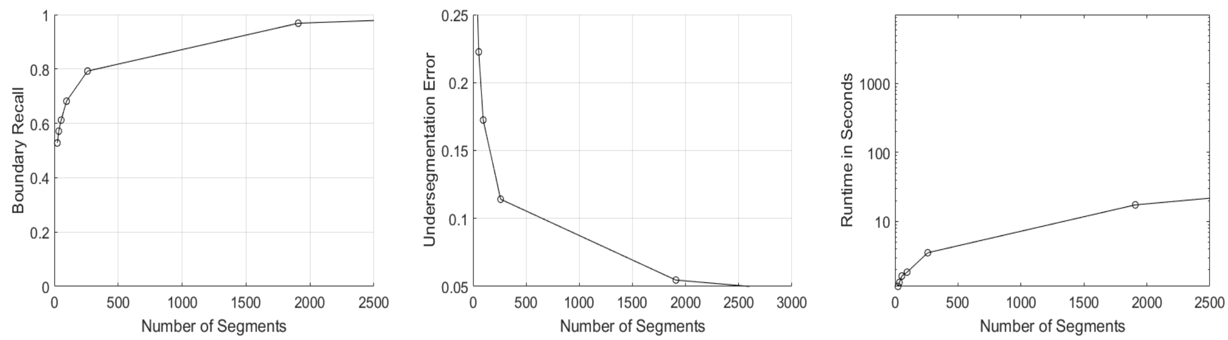


Figure 8: Lab color space evaluation results

2 EM Algorithm: Dealing with Bad Annotations

2.1 Derivation of EM Algorithm

The goal of the EM Algorithm is to use the provided data to estimate the parameters which can be expressed as:

$$\hat{\theta} = \arg\max_{\theta} p(\mathbf{x}|\theta) = \arg\max_{\theta} log \sum_{\mathbf{m}} p(\mathbf{x},\mathbf{m}|\theta) \tag{1}$$

Here m is the latent variable which indicates whether the annotator is good or bad. From Jensen's inequality $f(E[x]) \geq E[f(x)]$ for concave function x, we compute the lower bound of the problem which is to compute the expectation of log of the function instead:

$$\sum_{\mathbf{m}} log(p(\mathbf{x}, m|\theta))p(m|\mathbf{x}, \theta) \tag{2}$$

2.1.1 E-step

In the E step, we try to compute this expectation. For the case when the latent variable m has the label k we first expand this equation to

$$\sum_{i} \sum_{j\in S(i)} \sum_{k} [log(p(x_{ij}|m_j = k, \mu_i, \sigma, \beta))p(m_j = k|\mu_i, \sigma, \beta)]p(m_j = k|x_{i\in I(j)}, \mu_i, \sigma, \beta) \tag{3}$$

Where $S(i)$ is the set of 5 annotators that comment on image i and $I(j)$ is the set of 30 images annotator j gives score to.
One major step in E step is to compute the posterior of the latent variable m, that is the probability that the score is from a good annotator or a bad one given the parameters.

$$p(m_j = k|x_{i\in I(j)}, \mu_i, \sigma, \beta) = \frac{p(x_{i\in I(j)}|m_j = k, \mu_i, \sigma, \beta)p(m_j = k|\mu_i, \sigma, \beta)}{\sum_{k=0}^{1} p(x_{i\in I(j)}|m_j = k, \mu_i, \sigma, \beta)p(m_j = k|\mu_i, \sigma, \beta)} \tag{4}$$

Since m_j can only choose 0 and 1 in our case and $p(x_{i\in I(j)}|m_j = 1, \mu_i, \sigma)$ is the normal distribution for good annotators which is known, we get

$$p(m_j = 1|x_{i\in I(j)}, \mu_i, \sigma, \beta) = \frac{p(x_{i\in I(j)}|m_j = 1, \mu_i, \sigma)\beta}{p(x_{i\in I(j)}|m_j = 1, \mu_i, \sigma)\beta + \frac{1-\beta}{10}} \tag{5}$$

$$p(m_j = 0|x_{i\in I(j)}) = 1 - p(m_j = 1|x_{i\in I(j)}, \mu_i, \sigma, \beta) \tag{6}$$

For convenience we denote $\alpha_j = p(m_j = 1|x_{i\in I(j)}, \mu_i, \sigma, \beta)$ in the following section and the subscript in α_j shows there are 25 α values in total, one for each annotator.

2.1.2 M-step

M step updates the parameters such as μ_i, σ, β in order to maximize the expectation mentioned above, they can be solved from taking the derivatives of the expectation of the log likelihood function. Here we use θ^t to represent the parameters at each round of the EM algorithm.

$$\begin{aligned}\theta^{t+1} = \arg\max_\theta \sum_i \sum_{j\in S(i)} [log(p(x_{ij}|m_j = 1, \theta^t)) + log(p(m_j = 1|\theta^t))]\alpha_j \\ + \sum_i \sum_{j\in S(i)} [log(p(x_{ij}|m_j = 0, \theta^t)) + log(p(m_j = 0|\theta^t))](1 - \alpha_j)\end{aligned} \tag{7}$$

Substitutes in the normal distribution and the uniform distribution

$$p(x_{ij}|m_j = 1, \theta^t) = \frac{1}{\sqrt{2\pi}\sigma} e^{(-\frac{(x_{ij}-\mu_i)^2}{2\sigma^2})} \tag{8}$$

$$p(x_{ij}|m_j = 0, \theta^t) = \frac{1}{10} \tag{9}$$

$$\theta^{t+1} = \sum_i \sum_{j\in S(i)} [log\frac{1}{2\pi} - log\sigma\frac{(x_i - \mu_i^t)^2}{2\sigma^2} + log\beta^t]\alpha_j + \sum_i \sum_{j\in S(i)} [log(\frac{1}{10}) + log(1 - \beta)](1 - \alpha_j) \tag{10}$$

In this question we have $i = 150$ images and we are working with 150 normal distributions (one for each image) and each image has a different μ value. The derivation for μ will reflect the updates in a single image.

$$\frac{\partial}{\partial \mu_i} \sum_{j\in S(i)} \frac{(x_{ij} - \mu_i)^2}{\sigma^2}\alpha_j = 0 \tag{11}$$

$$\mu^{t+1} = \frac{\sum_{j\in S(i)} \alpha_j x_{ij}}{\sum_{j\in S(i)} \alpha_j} \tag{12}$$

The standard deviation is the same for all images, thus σ should be updated according to every annotation.

$$\frac{\partial}{\partial \sigma} \sum_i \sum_{j\in S(i)} (-log\sigma - \frac{(x_{ij} - \mu_i)^2}{2\sigma^2})\alpha_j = 0 \tag{13}$$

$$\sigma^{2^{t+1}} = \frac{\sum_i \sum_{j\in S(i)} \alpha_j (x_{ij} - \mu_i^{t+1})^2}{\sum_i \sum_{j\in S(i)} \alpha_j} \tag{14}$$

The prior β is same for all the annotators:

$$\frac{\partial}{\partial\beta}\sum_i\sum_{j\in S(i)}\alpha_j log\beta+\sum_i\sum_{j\in S(i)}(1-\alpha_j)log(1-\beta)=0 \tag{15}$$

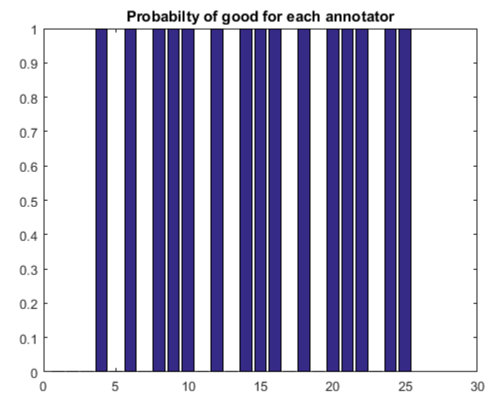
$$\sum_i\sum_{j\in S(i)}\frac{\alpha_j}{\beta}+\sum_i\sum_{j\in S(i)}\frac{(\alpha_j-1)}{1-\beta}=0 \tag{16}$$

$$\sum_i\sum_{j\in S(i)}\alpha_j-\sum_i\sum_{j\in S(i)}\beta=0 \tag{17}$$

$$\beta^{t+1}=\frac{\sum_i\sum_{j\in S(i)}\alpha_j}{750} \tag{18}$$

2.2 Application to Data

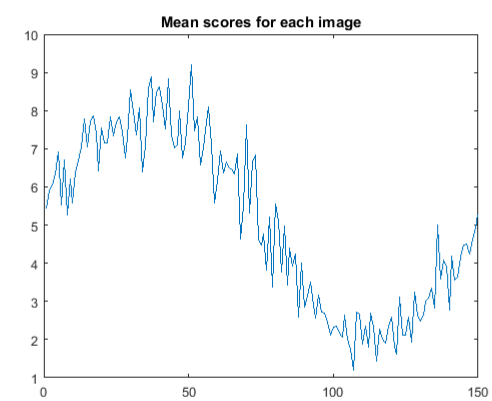
a)



From the bar plot of probability of "good" for each annotator, we can see the bad annotators are 1,2,3,5,7,11,13,17,19,23.

b) $\sigma = 0.8681$

c)



3 Graph-cut Segmentation

a)We first get two gaussian mixture models one for background and one for foreground each with a mixture of 5 gaussian functions using the MATLAB function and then the foreground and background likelihood of each pixel comes from the Gaussian Mixture Model.
b)

$$\textbf{unary potential}(x_i) = -log\frac{p(c(x)|foreground)}{p(c(x)|background)} \tag{19}$$

$$\textbf{edge potential}(x_i \neq x_j) = k_1 + k_2 e^{\frac{-||c(x_i)-c(x_j)||^2}{2\sigma^2}} \tag{20}$$

$$\textbf{edge potential}(x_i = x_j) = 0 \tag{21}$$

$$Energy(\mathbf{y}) = \sum_i \Phi_1(x_i) + \sum_{i,j} \Phi_2(x_i, x_j) \tag{22}$$

c)

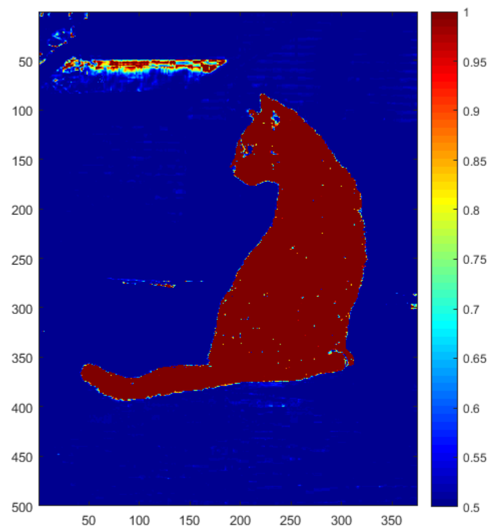


Figure 9: Foreground likelihood map

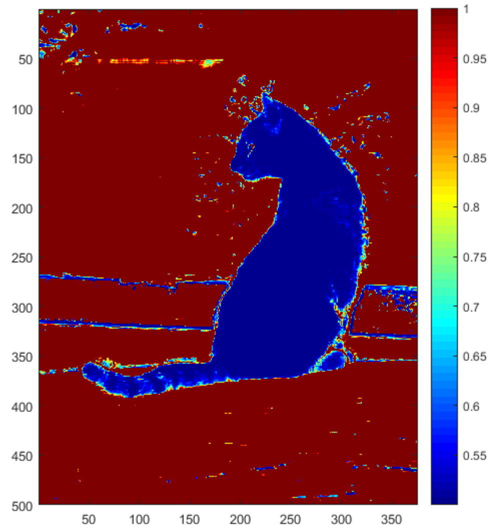


Figure 10: Background likelihood map

d)

Figure 11: Final segmentation