ECE4580 Homework #9

Due: Apr. 06, 2017

Problem 1. (25 pts) Before considering the implementation of k-means on images, let's go with the implementation as though an image just provided a colleciton of 3D vectors. These 3D vectors then feed into the k-means algorithm. Of course there is the issue of initialization. Since this homework involves Matlab's `kmeans` function you can either go with Matlab;s guess at the means or select your own. If selecting your own, please give the initial guess used.

Since we are using Matlab's algorithm, the only real thing that is needed it to convert the image data to vectorized data, apply the algorithm, then reconvert there data:

1. Convert image data (slices) to vectorized data (in a matrix).
2. Select the number of clusters, k.
3. Use Matlab's `kmeans` algorithm to cluster the data.
4. Convert the results back into an image.

The function stub is called `imkmeans.m`, and the script file you should prepare is called `segmentM.m`. It should select two of the images from the given Matlab file to process and go for it. Turn in the code and the results. Explain the output and your selection (image plus initial conditions).

Note: You shouldn't be using a k value that is crazy large, otherwise you won't get a sense for what k-means is doing. Likewise, selecting $k = 1$ or $k = 3$ may not be so sensible for your image. In this homework problem, you should go for an under-segmentation, but not an extreme one. For your personal exploration (e.g., done on your own but not turned in), feel free to try large k values.

Note: This implementation can be used as a sanity check on the image-based version.

Problem 2. (25 pts) Implement the k-means algorithm for image segmentation. Of course initialization is always a problem, so for the images provided give the initial guess of means to start out with.

Recall that the energy to minimize is:

$$\mathcal{E}(\mu, s) = \int_D \left|\left|I(\mathbf{x}) - \mu_{\mathbf{s}(\mathbf{x})}\right|\right|_\Sigma^2 \,\mathrm{d}\mathbf{x}\,,$$

where $I : D \to \mathbb{R}$ is the image proper, $\mu = (\mu_1, \dots, \mu_k)$ are the means, and $s : D \to \{1, \dots, k\}$ is the segmentation map. Usually, Σ is one or the identity matrix. It will be an optional argument for the function.

The algorithm is:

1. Start with initial guess of means.
2. Using means, segment by minimizing error energy.
3. Recompute means based on the segmentation.
4. Repeat (2)-(3) until segmentation does not change or maximum iterations hit.

The function stub is called `segKmeans.m`, and the script file you should prepare is called `segmentK.m`. As usual, it should select two of the images from the given Matlab file to process and go for it. Turn in the code and the results. Explain the output and your selection (image plus initial conditions).

Problem 3. (35 pts) Implement the Iterative Conditioning Mode algorithm for image segmentation, which is essentially k-means with regularization. Of course initialization is always a problem, so for the images provided give the initial guess of means to start out with.

Recall that the energy to minimize is:

$$\mathcal{E}(\mu, s) = \int_D \left|\left|I(\mathbf{x}) - \mu_{\mathbf{s}(\mathbf{x})}\right|\right|_\Sigma^2 \,\mathrm{d}\mathbf{x} + \lambda \int_{\mathbf{D}} \int_{\mathcal{N}(\mathbf{x})} \left(\mathbf{1} - \delta_{\mathbf{1}}(\mathbf{s}(\mathbf{x}), \mathbf{s}(\mathbf{y}))\right) \mathrm{d}\mathbf{y}\ \mathrm{d}\mathbf{x}$$

where $I : D \to \mathbb{R}$ is the image proper, $\mu = (\mu_1, \dots, \mu_k)$ are the means, $s : D \to \{1, \dots, k\}$ is the segmentation map, and δ_1 is the Kronecker delta function.

One algorithm is:

1. Starting with initial guess of means, generate segmentation by using standard k-means (w/out regularization).
2. Using means, update segmentation by minimizing error energy.
   - Here, this involves minimizing both the matching energy and the neighbor disagreement penalty (e.g., the regularization term).
3. Recompute means based on the segmentation.
4. Repeat (2)-(3) until segmentation does not change or maximum iterations hit.

The function stub is called `segICM.m`, and the script file you should prepare is called `segmentI.m`. Select two of the images from the Matlab file to process and go for it. Turn in the code and the results. Explain the output and your selection. Compare to just k-means (one way is to set lambda to zero or to perform no iterations and run `segICM`).

Problem 4. (20 pts) Move on to the next activity of the learning module. The group submission should reflect the work of the group, and should also be submitted individually with the name of your partner in the document. If submitting video or links to video for the pair, then only one member need to do so, while the other member should just note as much. The prior expectaction for deliverables continues to hold.