Image Segmentation using k-means clustering, EM and Normalized Cuts

Suman Tatiraju
Department of EECS
University Of California - Irvine
Irvine, CA 92612
statiraj@uci.edu

Avi Mehta
Department of EECS
University Of California - Irvine
Irvine, CA 92612
avim@uci.edu

Abstract

This project addresses the problem of segmenting an image into different regions. We analyze two unsupervised learning algorithms namely the K-means and EM and compare it with a graph based algorithm, the Normalized Cut algorithm. The K-means and EM are clustering algorithms,which partition a data set into clusters according to some defined distance measure. The Normalized Cut criterion takes a measure of the similarity between data elements of a group and the dissimilarity between different groups for segmenting the images.

1 Introduction

Images are considered as one of the most important medium of conveying information. Understanding images and extracting the information from them such that the information can be used for other tasks is an important aspect of Machine learning. An example of the same would be the use of images for navigation of robots. Other applications like extracting malign tissues from body scans etc form integral part of Medical diagnosis. One of the first steps in direction of understanding images is to segment them and find out different objects in them. To do this, features like the histogram plots and the frequency domain transform can be used. In this project, we look at three algorithms namely K Means clustering, Expectation Maximization and the Normalized cuts and compare them for image segmentation. The comparison is based on various error metrics and time complexity of the algorithms. It has been assumed that the number of segments in the image are known and hence can be passed to the algorithm.

The report is organized as follows.Section 2 describes each segmentation algorithm in detail. Results generated from the algorithms are presented in section 3. Finally, section 4 draws some conclusions.

2 Image Segmentation Algorithms

Images can be segmented into regions by the following algorithms:

2.1 K-means Clustering Algorithm

K-Means algorithm is an unsupervised clustering algorithm that classifies the input data points into multiple classes based on their inherent distance from each other. The algorithm assumes that the data features form a vector space and tries to find natural clustering in them. The points are clustered around centroids $\mu_i \forall i = 1 \dots k$ which are obtained by minimizing the objective

$$V = \sum_{i=1}^{k} \sum_{x_j \in S_i} (x_j - \mu_i)^2 \tag{1}$$

where there are k clusters S_i, $i = 1, 2, \dots, k$ and μ_i is the centroid or mean point of all the points $x_j \in S_i$

As a part of this project, an iterative version of the algorithm was implemented. The algorithm takes a 2 dimensional image as input. Various steps in the algorithm are as follows:

1. Compute the intensity distribution(also called the histogram) of the intensities.
2. Initialize the centroids with k random intensities.
3. Repeat the following steps until the cluster labels of the image does not change anymore.
4. Cluster the points based on distance of their intensities from the centroid intensities.

$$c^{(i)} := \arg\min_j ||x^{(i)} - \mu_j||^2 \tag{2}$$

5. Compute the new centroid for each of the clusters.

$$\mu_i := \frac{\sum_{i=1}^{m} 1\{c_{(i)} = j\} x^{(i)}}{\sum_{i=1}^{m} 1\{c_{(i)} = j\}} \tag{3}$$

where k is a parameter of the algorithm (the number of clusters to be found), i iterates over the all the intensities, j iterates over all the centroids and μ_i are the centroid intensities.

2.2 EM Algorithm

Expectation Maximization(EM) is one of the most common algorithms used for density estimation of data points in an unsupervised setting. The algorithm relies on finding the maximum likelihood estimates of parameters when the data model depends on certain latent variables. In EM, alternating steps of Expectation (E) and Maximization (M) are performed iteratively till the results converge.The E step computes an expectation of the likelihood by including the latent variables as if they were observed, and a maximization (M) step, which computes the maximum likelihood estimates of the parameters by maximizing the expected likelihood found on the last E step[1]. The parameters found on the M step are then used to begin another E step, and the process is repeated until convergence.

Mathematically for a given training dataset $\{x_{(1)}, x_{(i2}, \dots x_{(m)}\}$ and model $p(x, z)$ where z is the latent variable, We have:

$$\begin{aligned} l(\theta) &= \sum_{=1}^{m} \log p(x; \theta) && (4) \\ &= \sum_{i=1}^{m} \log \sum_{z} p(x, z; \theta) && (5) \end{aligned}$$

As can be seen from the above equation, The log likelihood is described in terms of x, z and θ. But since z, the latent variable is not known, We use approximations in its place. These approximations take the form of E & M steps mentioned above and formulated mathematically below.

E Step, for each i:

$$Q_i(z^{(i)}) := p(z^{(i)}|x^{(i)};\theta) \tag{6}$$

M Step, for all z:

$$\theta := \arg\max_{\theta} \sum_i \sum_{z^{(i)}} Q_i(z^{(i)}) \log \frac{p(x^{(i)}, z^{(i)};\theta)}{Q_i(z^{(i)})} \tag{7}$$

where Q_i is the *posterior* distribution of $z^{(i)}$'s given the $x^{(i)}$'s.

Conceptually, The EM algorithm can be considered as a variant of the K Means algorithm where the membership of any given point to the clusters is not complete and can be fractional.

2.3 Normalized Cuts-A graph Partitioning approach

Image segmentation can also be viewed as an optimal partitioning of a graph. The image is presented as a weighted undirected graph $G = (V, E)$. This image graph can be partitioned into two subgraphs A and B by modeling the partition as minimizing the *cut* as defined below:

$$cut(A, B) = \sum_{u \epsilon A, v \epsilon B} w(u, v) \tag{8}$$

where $w(i, j)$, the weight of each edge is a function of the similarity between nodes i and j.However the minimum cut criteria favors cutting small sets of isolated nodes in the graph.To overcome these outliers we can use a modified cost function, *Normalized Cut* as defined below.

$$Ncut(A, B) = \frac{cut(A, B)}{assoc(A, V)} + \frac{cut(A, B)}{assoc(B, V)}, assoc(A, V) = \sum_{u \epsilon A, t \epsilon V} w(u, t) \tag{9}$$

The association value, *assoc(A,V)*, is the total connection from nodes A to all nodes in the graph. *Ncut* value won't be small for the cut that partitions isolating points, because the cut value will be a large percentage of the total connection from that set to the others.

Given a partition of a graph V into two disjoint complementary sets A and B, let x be an $N = |V|$ dimensional indication vector, $x_i = 1$ if node i is in A, -1 otherwise. Let $d_i = \sum_j W(i, j)$ be the total connection from node i to all other nodes. *Ncuts* can be rewritten as (3).

$$Ncut(A, B) = \frac{\sum_{x_i > 0, x_j < 0} -w_{ij} x_i x_j}{\sum_{x_i > 0} d_i} + \frac{\sum_{x_i < 0, x_j > 0} -w_{ij} x_i x_j}{\sum_{x_i < 0} d_i} \tag{10}$$

Let $D = diag(d_1, d_2 .... d_N)$ be an NxN diagonal matrix and W be an NxN symmetric matrix with $W(i, j) = w_{ij}$ Finding global optimum reduces to (4):

$$min_x Ncut(x) = min_y \frac{y^T (D - W) y}{y^T D y} \tag{11}$$

with the condition $y(i) \,\epsilon\, \left\{1, -\frac{\sum_{x_i>0} d_i}{\sum_{x_i<0} d_i}\right\}$ and $y^T D1 = 0$. If y is relaxed to take on real values, we can minimize (4) by solving generalized eigenvalue system

$$(D - W)y = \lambda D y \tag{12}$$

Constraints on y come from the condition on the corresponding indicator vector x. The second smallest eigenvector of the generalized system (4) satisfies the normality constraint. $y1$ is a real valued solution to our normalized cut problem.

2.3.1 The Grouping Algorithm

The eigenvector corresponding to the second smallest eigenvalue is the real-valued solution that optimally sub-partitions the entire graph, the third smallest value is a solution that optimally partitions the first part into two etc. The grouping algorithm consists of the following steps:

a) Given an image, Construct a weighted graph $G(V, E)$ by taking each pixel as a node and connecting each pair of pixels by an edge. The weight on that edge should reflect the likelihood that the two pixels belong to one object. Using just the brightness value of the pixels and their spatial location, we can define the graph edge weight connecting the two nodes i and j as:

$$w(i, j) = \exp \frac{-\|F_{(i)} - F_{(j)}\|_2^2}{\sigma_I^2} * \exp \frac{-\|X_{(i)} - X_{(j)}\|_2^2}{\sigma_X^2} if \|X_{(i)} - X_{(j)}\|_2 < r \tag{13}$$

else the value of $w(i, j)$ is zero. In the present approach $F(i) = I(i)$, the intensity value, for segmenting brightness images. The weight $w(i, j) = 0$ for any pair of nodes i and j that are more than r pixels apart.

b) Solve $(D - W)y = \lambda D y$ for eigenvectors with the smallest eigenvalues.

c) Use the eigenvector with second smallest eigenvalue to bipartition the graph by finding the splitting points such that $Ncut$ is minimized.

d) Recursively repartition the segmented parts (go to step a).

e) Exit if $Ncut$ for every segment is over some specified value.

3 Results

We applied the partitioning algorithm to gray-scaled images. The segmentation results as generated on different images by each algorithm for varying value of *k* is as shown in figure 1, 2, 3 and 4.

4 Conclusion

We implemented the EM and K-means clustering algorithm and used it for intensity segmentation. For smaller values of k the algorithms give good results. For larger values of k, the segmentation is very coarse, many clusters appear in the images at discrete places. This is because Euclidean distance is not a very good metric for segmentation processes.

Better algorithms like the graph based NCuts give good results for larger value of k. One basic difference between Ncuts and clustering algorithms is that clustering algorithms consider pixels with relatively close intensity values as belonging to one segment, even if they are not locationally close. Ncuts considers such areas as separate segments.NCuts implementation is computationally complex.The eigenvalue method take a longtime for a full scale image.Images have to be resized to get faster results.We used the authors[1] implementation of NCuts to visualize the results. Future work includes analyzing other machine learning algorithms like neural networks and SVM for image segmentation.

5 References

[1] Jianbo Shi & Jitendra Malik (1997) Normalized Cuts and Image Segmentation, *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, pp. 731-737.

[2] T. Kanungo, D. M. Mount, N. Netanyahu, C. Piatko, R. Silverman, & A. Y. Wu (2002) An efficient k-means clustering algorithm: Analysis and implementation *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, pp.881-892.

[3] D. Arthur, & S. Vassilvitskii (2007) k-mean++ The advantage of Careful Seeding. *Symposium of Discrete Algorithms*

Figure 1: Original Images

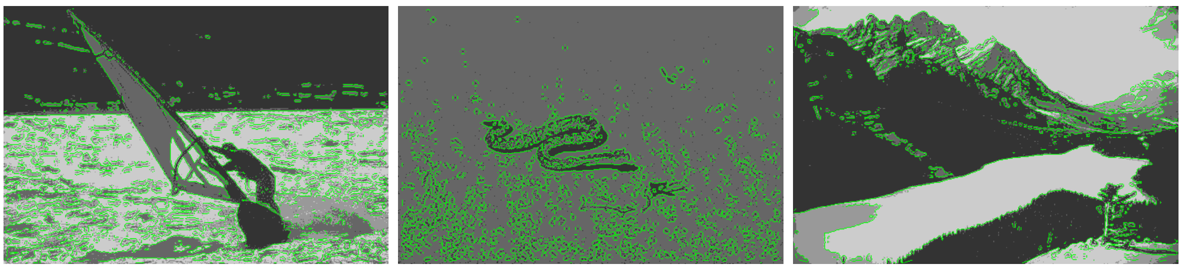


Figure 2: Expectation Maximization

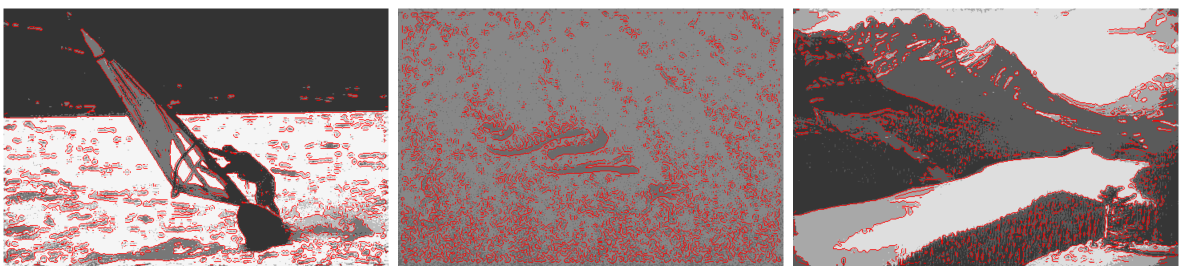
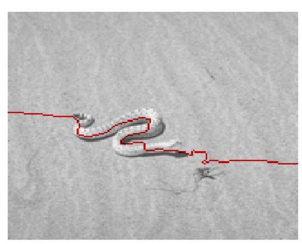
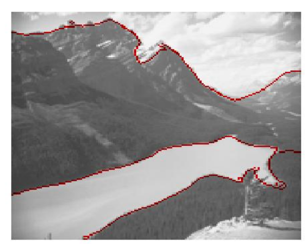


Figure 3: K Means

(a) Rowing

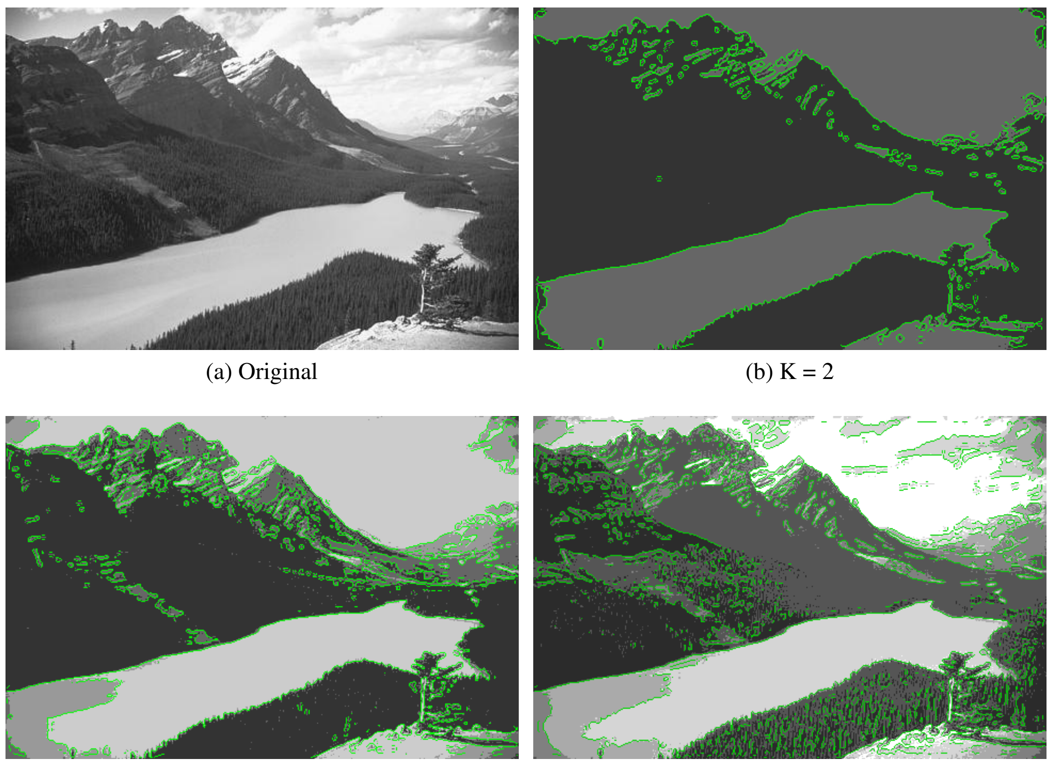


(b) Snake



(c) Wolf

Figure 4: Normalized Cuts



(a) Original

(b) K = 2

(c) K = 4

(d) K = 6

Figure 5: Results from Expectation Maximization Algorithm for various K’s