Berkeley Segmentation Dataset and Benchmark

Juan Camilo Martinez
Universidad de los Andes
Departamento de Ingeniera Industrial
jc.martinez10@uniandes.edu.co

Abstract

The Berkeley Segmentation Dataset and Benchmark provides an empirical basis for evaluation of image segmentation methodologies. In this work, implementations of the traditional methodologies K-Means and GMM were benchmarked to establish their applicability and compare them to morw sophisticated techniques, namely the Ultrametric Contour Map.

Keywords: segmentation, unsupervised learning, superpixel

1. Introduction

In order to advance the development of algorithms for computer vision tasks, objective metrics need to be used to assess their performance. To this end, The Berkeley Segmentation Dataset and Benchmark presents a platform to test image segmentation methodologies.

In this work, two methodologies were implemented: K-Means and the Gaussian Mixture Model(GMM). They were run on the 200 test images provided on the BSDS500.

2. Materials and Methods

2.1. Segmentation Algorithms

K -means clustering is an unsupervised algorithm that relies on the iterative refinement of cluster centroids according to the least euclidean distance between the centroid and the points assigned to that cluster.

The Gaussian Mixture Model addresses some flaws present in K-Means. Whereas K-Means works best with ”circularly shaped” data, GMM uses Gaussian Means instead of Euclidean distances, describing each cluster by its centroid, covariance and the size. In essence, instead of a distance to a centroid, each point is assigned a probability of being related to a centroid.

These two methods were chosen to analize both a computationally inexpensive approach (K-Means) and one that relies on a similar principle but with a greater adaptability (GMM)

2.2. Testing Methodology

The database includes 500 images of 481x321 pixels, both in portrait and landscape orientation. 300 of the images are meant for training (i.e. parameter tuning) while 200 images are reserved for bench marking.

2.3. Testing Methodology

The main hyper parameter relevant to the tests carried out was the number of clusters to define the segmentation. This parameter was varied within a range of 4-100, where 4 clusters might prove insufficient to adequately segment real world images, and 100 can result in overly small super pixels for 481x321. Figure 1 shows an example of a K-Means segmentation from the HSV color space, using 4 clusters.

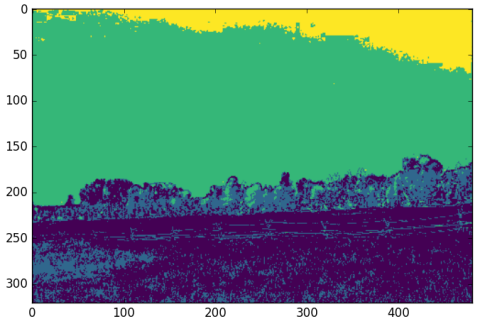

Figure 1. Example of K-Means clustering for k=4

2.4. Evaluation

In order to run the benchmark, MATLAB structures were obtained by saving the different segmentation results (mul-

tiple k values) into cell arrays for every test image. These results were then passed as input to the functions included with the dataset.

200 .mat files were generated for each of the clustering methods.

2.5. Results

The results shown on figure 2 compare the precision-recall curves for K-Means with UCM results as a reference.

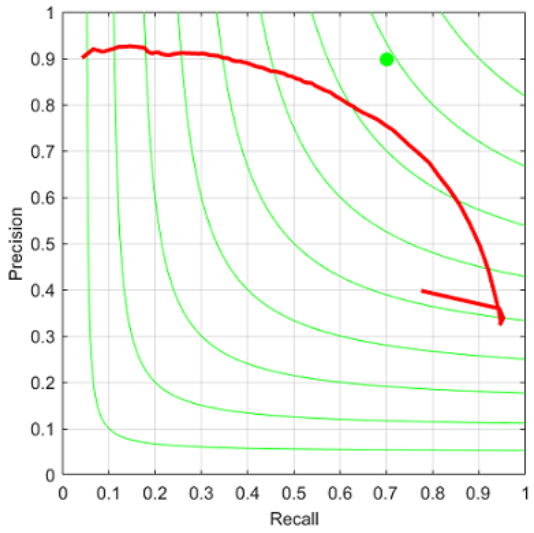

Figure 2. Benchmark results for K-Means

Figure 3 does the same for GMM compared with UCM.

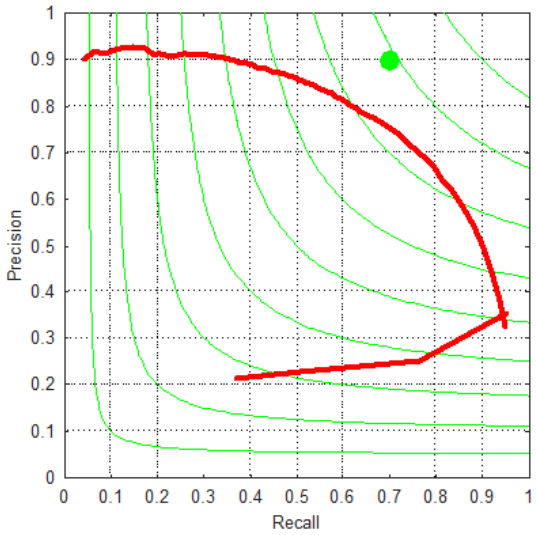

Figure 3. Benchmark results for GMM

Neither result covered a significant portion of the precision recall space. The GMM method presented a longer curve, but with lower precision values. This is surprising, given that the algorithm was expected to work at least as well as K-Means. This suggests that the number of models to be used has a different effect on the method compared to number of centroids for K-Means, or that other hyper parameters need to be adjusted to get the most out of GMM.

It appears that the higher the number of clusters used, the worse the precision becomes for GMM.

Metrics for K-Means:

```
ODS: F(0.81, 0.39)=0.53 [th=1.23]
ODS: F(0.81, 0.39)=0.53
Area_PR=0.07
```

Metrics for GMM:

```
ODS: F(0.95, 0.35)=0.51 [th=1.00]
ODS: F(0.95, 0.35)=0.52
Area_PR=0.95
```

Coverage is better for GMM, but if K-Means presents a better precision at lower recall values.

3. Conclusions

Two algorithms were bench marked according to the procedure established for the BSDS500 dataset. Neither algorithm performed as well as the state of the art.

References