Lab 7 BSDS500

Nathan Mateo Marín
Universidad Los Andes
nm.marin10@uiandes.edu.co

Abstract

This laboratory is the continuation of the lab 6, in the sense that the objective is to evaluated in the BSDS500 framework the two best segmentation algorithms of the lab 6. The algorithms used were kmeans and gmm with the rgb channels. In addition, these two algorithms will be compared against the UCM algorithm of proffesor Pablo. Although none of the two algorithms were capable of surpass the UCM algorithm their perfomance was good taking into account how simple they are.

1. Introduction

The BSDS500 is the Benchmark used for all researches to test and train their segmentation methods, the database has 500 images, 200 are in the Test group, 200 in the Train group and 100 in the Validation group. The images are quite diverse, ranging from buildings, to planes, to animals to humans. Each image has multiple annotations which consist of the boundaries of the objects in the image, multiple persons were ask to draw the boundaries and each result is on of the annotations of the image. The Benchmark evaluates how many boundaries were detected correctly, for that from the regions of the segmentation the boundaries are computed, then every pixel of the image is given one of two labels, one indicates that the pixel is a boundary and the other indicates that it is not. Once the boundaries are computed they are compare with the annotations to determine if the boundary is a true positive or a false negative, to do so a threshold is used, for each pixel with the label of boundary in the boundary image obtained by the algorithm is searched in the annotations if there is a pixel of boundary in the proximity, if the distance between the pixel analyzed and the closest actual pixel of boundary is less than the threshold, then the pixel is classified as a true positive, in the other scenario it is classified as a false positive. Knowing the amount of true boundary pixels and the amount of true positives it is possible to calculate the amount of false negatives. With this information a precision-recall curve is constructed by changing the threshold.

The algorithm is finally evaluated by the precision-recall curve.

2. Methods

For this laboratory the methods used where K-means and GMM, and the channel utilized was in both cases RGB. Let's explain first why to use the RGB channel, in the previous laboratory it was noticed that the spacial information cause big homogeneous regions to be segmented in different parts, therefore making the segmentation worst. Besides, when looking at the biggest Jaccard index in lab6 it was deduced that K-means and GMM with the channel RGB give the best results. Just to recall, the possible color spaces that the segmentation algorithms can use are, RGB, LAB, HSV. HSV is not very useful because the channel S gives information about the saturation of the color and it doesn't help to segregated it from the others. The channel LAB does has the Luminosity (the image in gray scale more or less), but when the algorithms were design this gray scale information was added to RGB too, as a weighted average of the three channels; in that sense LAB doesn't have advantage over RGB. Nonetheless, in the channels A and B the euclidean distance between two points is suppose to give a good idea of how similar the two colors are, but the lab6 data says that RGB works better than LAB.

Now let's give a little explanation of K-Means and GMM:

2.1. K-means

The first step is to chose k random points (centroids) in the representation space, then each pixel is assign to the class of the closest centroid, making a first partition of the space, once here with the data of each pixel the new centroids of each class are calculated. These centroids are used to reassign the pixels and make a new partition of the space. The process is repeated until convergence is reached, that is, between to consecutive steps none of the pixels is reassign.

2.2. GMM

K Gaussians with random mean and variance are made, then to each pixel a probability of belonging to a Gaussian is assign (This probability is the values of the Gaussian over the point that represents the pixel). With these dependencies (probabilities) computed, new means and variances are calculated for the Gaussians. The process is repeated until convergence is reached (the means and variances change less than a given threshold). Finally, it is said that a given pixel belong to the group of the Gaussian that has the biggest value over that point.

3. Observations

A couple of thing must be said about the evaluation.

First, to evaluated the results of my algorithms a matlab function provided for the professor is used, at the moment of use a problem made itself evident: when the algorithm makes a segmentation it gives to every pixel a label that says to which group it belongs, those labels are integers and in my case they start with zero, and sometimes with another integer; the problem is that the matlab function assumes that the labels star with a one, therefor when face with a zero it sends an error. To fix this problem it was necessary to modify all of the segmentations so they would start with a one, this process took about three hours.

Second, at first glance it is easy to think that the final objective of any segmentation algorithm is to achieve the perfect 1,1 in the precision-recall space, but that is not the case. This is due to the fact that not even the humans when face with the problem of segmentation agree, some of them mark a lot of boundaries, others only the more important ones and other are somewhere in between. As a consequence, when comparing the segmentations of one human against the ones of the rest the point 1,1 is never achieve. In this sense, the objective of an algorithm is to achieve the performance of human beings, this performance is obtained with the average of the performance of every single human compared against the others. That objective point is shown in all the curves of the results section.

Third, both of the methods evaluated have only one hyperparameter and that is the number of cluster k, that's why each image was segmented with k=2,3,...,23. Using these values the process of segmentation of the BSDS500 database took around two days.

4. Results

As mentioned before the evaluation is given by the precision-recall curves (p-r curves). Therefore, Figure 7 has the p-r curve for K-means in the group of validation, Figure 6 has the p-r curve for gmm in that same group, Figure 5 has the p-r curve for K-means in the group of Train, Figure 4 has the p-r curve for gmm in that group too, Figure 3 has the p-r curve for K-means in the group of Test, Figure 2 has the p-r curve for gmm in the group of Test and Figure 1 has the p-r curve for UCM in the group of Test.

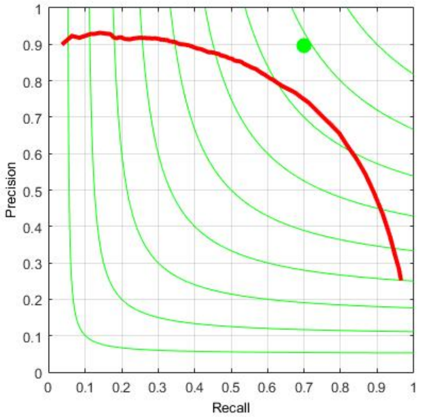

Figure 1. Precision vs recall curve for the UCM algorithm in the test group

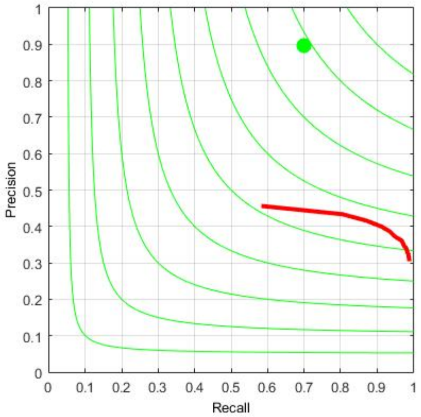

Figure 2. Precision vs recall curve for the gmm algorithm in the test group

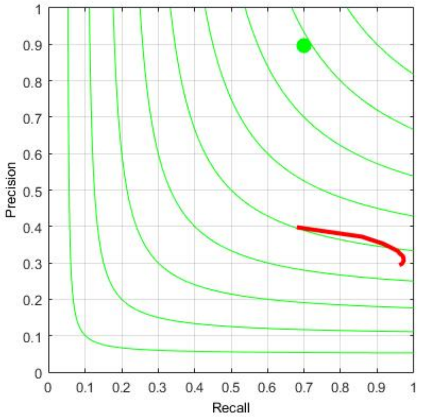

Figure 3. Precision vs recall curve for the K-means algorithm in the test group

One thing that must be said is that every point of the curve is calculated with a different value of k, that number

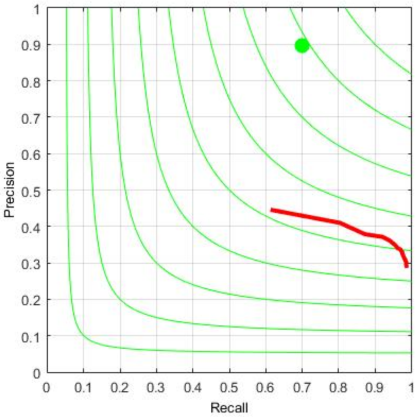

Figure 4. Precision vs recall curve for the GMM algorithm in the train group

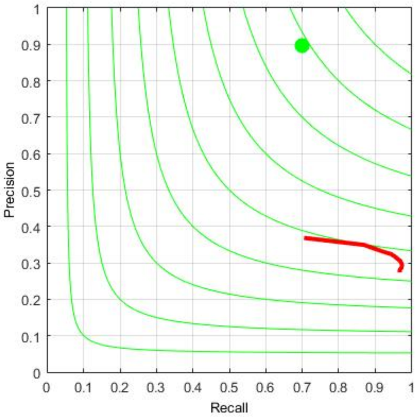

Figure 5. Precision vs recall curve for the K-means algorithm in the train group

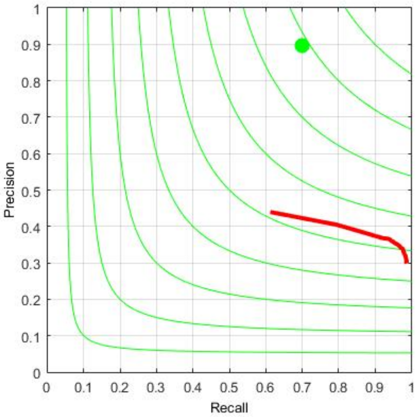

Figure 6. Precision vs recall curve for the GMM algorithm in the validation group

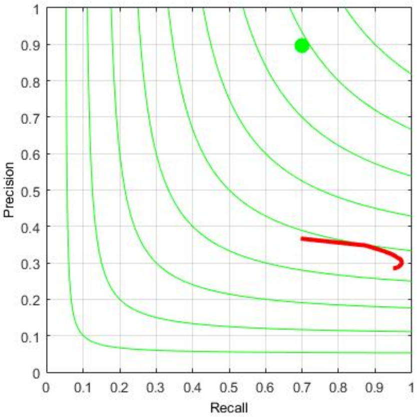

Figure 7. Precision vs recall curve for the K-mens algorithm in the validation group

k gives a value of recall, so for interpretation purposes the vertical lines in the figure will be seen as a given value of k. This is important because the idea is to obtain the optimal value o k from the Training and Validation group and then use this value for the test. That's why, when looking at the Test curves of the gmm and k-mens methods, the point of interest to evaluated the method won't be the one that has the highest F-mean, but the one that corresponds to the optimal k obtain from the validation and Training curves.

Knowing this and looking at Figure 7 and Figure 5 it can be seen that the value of recall with highest F-mean for K-means is 0.85. Similarly, looking at Figure 6 and Figure 4 it's deduced that the value of recall with the biggest F-mean for GMM is 0.8. Having these two values in mind and looking at Figure 3 and Figure 2 it is noticed that the F-mean of K-means with channel RGB is $F_k = 0.51$, and the F-mean of gmm with channel RGB is $F_{GMM} = 0.56$.

Finally looking at Figure 1 it can be seen that the maximum F-Mean of UCM is $F_{UCM} = 0.73$

It's quite curious that the curves behave mostly the same for the three groups (Trainig, Validation, Test), this is a sign that the algorithms are consistent. They always make the same mistake, only segment objects by their color.

Another interesting fact is that GMM always wins over K-means, this is not a surprise because GMM allows the clusters to have a ellipsoid form, while K-means only lets them be a sphere.

The last observation that needs to be made is the fact that in lab6 our evaluation method obtained Jaccard indexes around 0.9 (which is great) but now that we use the BSDS500 evaluation method it's easy to see that the algorithms aren't perfect (not even close). The discrepancies between the two evaluations are due to two facts: First, one evaluated only the boundaries while the other sees the segmented region itself. Second, the evaluation of lab6 was made over only one image of the 500.

5. Conclusions

It's quite clear that K-means and GMM with the RGB channel are not the best solution to the segmentation problem. Part of the problem is that these two algorithms only see the color information and asume that the clusters have a 'spherical' form. A first approach to boost the performance of the methods is to give them texture information, for that to the RGB channels it would be needed to add channels corresponding to the response of the pixels to a bank o filters. But no even this would be enough, the next step is to stop assuming the number of clusters and their shape, for that the K-means and GMM algorithms will have to be replace by a Meanshift algorithm working in the space of color and texture (as response to the filters). With these two variations the algorithm should work a lot better, probably not as good as UCM, but way better than now.

References