

# Segmentation:BSDS

Johana M. Ramirez Borda  
Los Andes University

jm.ramirez11@uniandes.edu.co

Francisco A. Roza Forero  
Los Andes University

fa.roza1843@uniandes.edu.co

## Abstract

*Contour Detection and Image Segmentation Resources* is the name of the project developed by the computer vision group of the University of Berkeley in California. This group provided the BSDS500 data set that is used in this work. Our objective is to segmentate the 500 images of this set using Kmeans with 17 different values of  $K$  and GMM and then, to compare them with the ucm2 method. The last method proved to be the most effective with an  $F$  value of 0.73-0.76 and an area under the curve of 0.74 for the test images.

## 1. Introduction

The computer vision group of the University of Berkeley in California developed a project called "Contour Detection and Image Segmentation Resources". The main objective of this project was to provide an empirical basis for research on image segmentation and boundary detection, promoting scientific progress in the study of visual grouping [3],[4].

The dataset BSDS500 consists of 500 natural images and the human annotations serve as ground-truth. These 500 images are separated into train (200), test (200) and validation (100) [3],[4].

Additionally, the precision-recall curve is used to describe the performance of the algorithm. However, even though this curve is a rich descriptor of performance, it is used the maximum F-measure value too, as a descriptor that assesses the performance into a single number [3],[4].

In this project, the Berkeley Segmentation Dataset (BSDS500) was used in order to test two segmentation methods such as kmeans and gmm and compare them with the ucm2 method.

## 2. Materials and methods

For this lab, the software implemented was MATLAB R2016b. The inside functions implemented were the following:

- segmentByClustering

- allBench\_fast
- boundaryBench\_fast
- collect\_eval\_bdry
- collect\_eval\_reg
- evaluation\_bdry\_image\_fast
- evaluation\_reg\_image
- isoF
- match\_segmentations2
- plot\_eval
- regionBench
- seg2bdry

### 2.1. Kmeans and GMM segmentation

Kmeans had one of the best segmentation results when using the hsv+xy colorspace. Because the segmentation separated different object and/or regions without separating details of the objects different segments. Additionally to kmeans, hierarchical segmentation had a similar performance than kmeans with a hsv+xy colorspace too. However, the hierarchical method requires a large amount of time to process a single image. Hence, it wasn't used because it would spent more time than the available to segment all the database. In consequence, the second method implemented was GMM with a colorspace of Lab+xy, whose overall performance followed Kmeans and hierarchical with colorspace hsv+xy.

Both kmeans and GMM had as an hyper-parameter the number of clusters, which is the number of centroids that represent each cluster.

### 2.2. Evaluation

First, the part 4 of the code test\_benchs\_fast.m was used to evaluate each method with the ground-truth data. To use it, the path of the original images, of the ground-truth

data and the results of each method was saved in a variable. Then, these variables were used to call the function `allBench_fast`, which runs boundary and region benchmarks on dataset. Its output is a folder that contains the necessary information to graph the precision-recall curve, to obtain the area under this curve and the maximum F-measure value, which are the variables that we are going to use in order to compare the three methods. In this sense, at each level the precision and recall quantities are computed and in this way, the precision-recall curve is produced. Precision is the probability that a predicted response is a true positive, also, is a measure of how much noise is in the output. Recall is the probability that a true positive of the groundtruth is detected. Moreover, the precision-recall curve tells us about the trade-off between misses and false positives for different threshold. On the other hand, an "ideal" result can be defined in different ways. First, it could be defined as a curve which maximum F-measure is 1, which is usually unlikely. Second, it could be defined as the curve which maximum F-measure is the one produced by human [3],[4].

### 3. Results and discussion

Below are presented the results obtained for the BSDS benchmark (full test and val sets) for the three methods.

Firstly, the values for ODS F-measure, OIS F-measure and area under the curve, for the test and val sets are shown:

Table 1. Results for GMM segmentation on test and val sets.

GMM			
TEST		VAL	
ODS: F	0.58	ODS: F	0.56
OIS: F	0.64	OIS: F	0.62
Area	0.32	Area	0.30

Table 2. Results for Kmeans segmentation on test and val sets.

KMEANS			
TEST		VAL	
ODS: F	0.56	ODS: F	0.55
OIS: F	0.61	OIS: F	0.59
Area	0.32	Area	0.31

Table 3. Results for ucm2 segmentation on test and val sets.

UCM2			
TEST		VAL	
ODS: F	0.74	ODS: F	0.73
OIS: F	0.77	OIS: F	0.76
Area	0.75	Area	0.74

Secondly, in the Figure 1. can be observed the precision-recall curve for the GMM method. In the Figure 2. is the

graph of the kmeans segmentation and the last figure contains the result of the three methods.

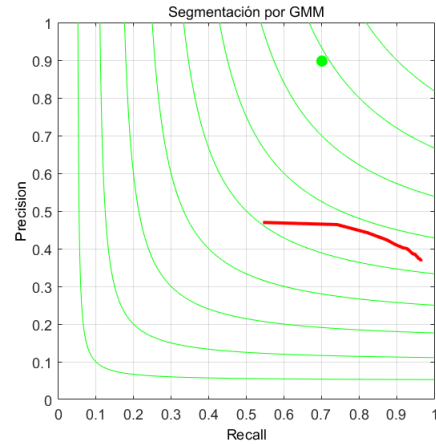


Figure 1. Precision-recall curve of the GMM segmentation.

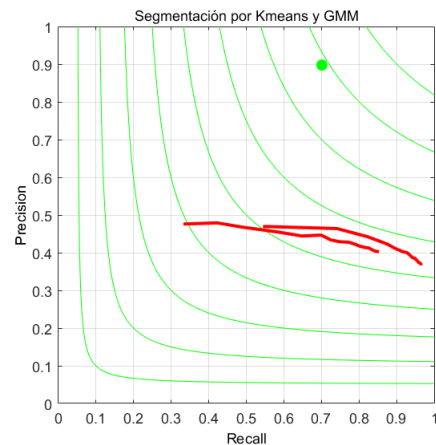


Figure 2. Precision-recall curve of the kmeans segmentation.

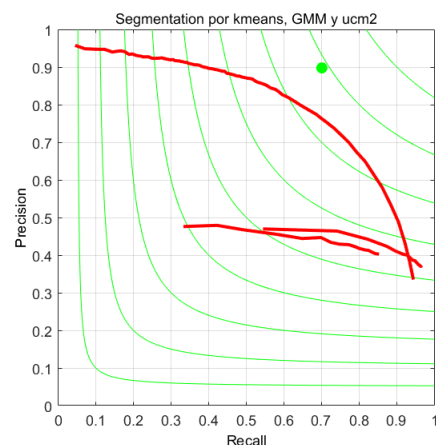


Figure 3. Precision-recall curve of the three methods.

According with the results, the segmentation with GMM gives better results than kmeans. In the first place, it is well known that k-means is only applicable if the mean is defined, moreover it needs the determination of the value of  $k$  and it is very sensible to outliers. It also only joins clusters with spheric forms. On the other hand, GMM can be seen as the generalization of kmeans. This method forms clusters by representing the probability function of a pixel to belong to a certain combination of non-parametric distributions. It is said that gmm method helps to express the uncertainty that kmeans doesn't allow to [2],[1].

The two methods implemented didn't beat the Pablo's UCM method and were far behind its performance. Kmeans and GMM were of the first basic methods implemented for segmentation and they group pixels based only on their similarity. On the other side, the ucm method finds boundaries in the image based on their color, brightness and texture. Additionally, it builds a graph based on the lines present in the image and their different magnitude and angle and then uses normalized cuts based on the eigenvectors of the graph; which is a greater processing to the image than just reorganizing pixels based on distances.

Kmeans algorithm is fast processing 2.76 images per minute, while gmm spent 1.2 minutes for two images with a processor of 8GB RAM. Also, both methods rely only on the pixel information regarding position, colorspace values, distance measurement method and  $\sigma$  for kmeans and gmm respectively. Pixel information ignores boundaries and edges that separate objects. Moreover, it ignores information given by textures, boundaries, lines orientations, local minimum or maximums.

When the pixels characteristics are not normalized, the relevance of the information with low values is meaningless next to those with high values (i.e. L channel of Lab vs xy position). Also, not every image responds the same way to the different colorspace. For example, shadows and reflections in images tend to become segmented objects in the image.

- One way to improve it is to increase the information of the pixels. This can be done by combining the information provided by all the colorspace as well as the position.
- Combine methods, such as adding information of watersheds to the one provided by kmeans or even adding mean shift.
- Detect border or magnitude of images and then cluster pixels bounded by them.

## 4. Conclusions

Kmeans presented one of the best segmentations when testing it with a small set of images, same as hierarchical. However, the results in the whole BSDS database said that GMM was the best method from the chosen ones. Yet, both methods were far below the ucm algorithm because of their low processing and information they use of the images. So, in order to improve the results, one method is not enough to achieve a good performance and the combination of many methods can make the difference.

## References

- [1] P. Arbelaez. Lecture 5: Clustering. In *Computer Vision IBIO4680*.
- [2] U. Baid, S. Talbar, and S. Talbar. Comparative study of k-means, gaussian mixture model, fuzzy c-means algorithms for brain tumor segmentation. 2017.
- [3] C. F. Pablo Arbelaez and D. Martin. The berkeley segmentation dataset and benchmark, jun 2007.
- [4] C. vision group. Contour detection and image segmentation resources, jun.

*All the codes created and used are attached.*