Universidad de los Andes
Laboratorio de Vision Artificial

Development and evaluation of image segmentation algorithms

Felipe López[1,*]

Abstract

Image segmentation has long been identified as a computer vision problem with a far seen solution. However, since benchmark databases like the Berkeley Segmentation Database (BSDS) were created, now it is possible to measure progress in this discipline and take us closer to the solution of this problem. In this paper, a segmentation algorithm was developed based on pre-existing methods and was later evaluated in the BSDS to measure the capabilities of the designed algorithm against the strong methods already known in the community. The whole purpose of this project is to get familiar with the bench marking of algorithms in popular databases, and by that compare the efficiency and accuracy.

Keywords

Image Segmentation — BSDS — Computer Vision

[1] *Department of Biomedical Engineering, Universidad de los Andes, Bogotá, Colombia*
[*]**Corresponding author**: jf.lopez718@uniandes.edu.co

Contents

Introduction

Since the implementation of the Berkeley Segmentation Datasets and Benchmark(BSDS), a lot of progress has been made in computer vision. It allows for a quantitative comparison between the different existing methods of boundary detection or segmentation, and allows the user to rank their newly proposed algorithm between the existing ones. This was a leap toward progress and provided wonderful results in this field in the past decades.

1. Development of the segmentation algorithm

The algorithm to be implemented consisted of a function that takes different parameters to do the segmentation in a particular way. The parameters of this function are:

- Image to be segmented
- Feature space
- Clustering method
- Number of clusters

1.1 Image to be segmented

The image to be segmented is just a standard image in RGB space, in this case the images were the 200 test images available in the BSDS. An example of one of these images is shown in Figure 1.

1.2 Feature space

Different feature spaces were used to evaluate which results in a better segmentation.

RGB The RGB feature space corresponds to the usual space of the image and it uses the Red, Green and Blue channels to represent the image.

HSV The HSV feature space represents an image by a combination of Hue, Saturation and Value channels. To segment the image in this space, one has to perform a transformation from RGB to HSV space first

La*b* This feature space represents an image by a combination of Illumination (L) and two color channels (a and b). To use this feature space, a transformation from RGB to La^*b^* needs to be done.

Space +xy Three other spaces were used in which the spatial position of the pixels was taken into account in the previous three spaces. To use this feature space, two more channels are added to the previous three channels of each of the feature space. The additional channels contain information of the x and y position of each pixel.

1.3 Clustering method

To evaluate the performance of different clustering methods, three very different methods were selected. Each perform the segmentation in a very different way from each other but the output is the same, a matrix with the label of the cluster that the pixel belongs to in the position were each pixel is located.

K-means

Mixture of Gaussians (GMM)

Watersheds

1.4 Number of clusters

For each clustering method to perform the segmentation, the user needs to provide a very important parameter for it to work. The number of clusters is the number of segments in which the image is to be divided into. K-means and GMM, both use this parameter to generate the number of clusters provided. However, because of the way the watersheds method is implemented, it uses this parameter as the threshold to cut the regional minima and provide a hierarchical segmentation. This means that the greater this parameter for GMM, the least number of clusters generated, and for K-means and GMM this number results in the exact number of clusters generated.

2. Results

The algorithm was implemented by alternating the parameters for the test images provided by the BSDS. An example of this implementation for the different parameters is shown in Figures 1-16. Here the image in Figure 1 was segmented by the different methods to evaluate different parameters before running the methods on the entire data set. In this case the number of clusters was 6 for K-means and GMM, and the threshold was 6 and 60 for Watersheds. The results of each segmentation were compared to the human segmentation in Figure 2 to evaluate the accuracy of the methods. Additionally the computational time was measured for each method, which is a very important characteristic of a method because if it is too big it will account for a lot of computational cost when running the algorithm on the entire dataset. This evaluation showed that Watersheds and K-means were the fastest and thus the more efficient computationally. Also, it could be noted that for each of the +xy feature spaces, K-means prioritizes the position of each pixel over its color and results in a segmentation with kind of a grid form.

After this single evaluation, the algorithm was tested in the entire dataset providing the information that is summarized in Table 1. Precision-Recall (PR) curves were plotted to show this information more graphically and to provide more comparative information.

K-Means The PR curves for the evaluation of K-means in different feature spaces is shown in 17. This figure restates the previous observation that the +xy feature spaces in K-means result in a wrong grid-like segmentation, seen in the HSV+xy curve that gives a maximum F value of only 0.19. However, after it was noted that these curves had a very low AP and due to the fast computational time of these algorithms longer curves were performed by generating more segmentations with different number of clusters. These new curves are shown in Figure 18.

GMM The PR curves for the evaluation of GMM in different feature spaces is shown in Figure 19.

Watersheds The PR curves for the evaluation of Watershed is shown in Figure 20 with the results of the best of each other algorithms to illustrate the differences the algorithms have in the complete PR range. It can be seen the the algorithm provided by Arbelaez et. al [1] gives the best behavior in the entire range. As of the methods used in this implementation, the GMM in HSV+xy method gave the highest maximum F measure of 0.57, but the watersheds method had a very close measure of 0.53 with much faster computational time that allowed to implement the algorithm over a wider range and generated a longer curve that gave a better AP.

Table 1. Benchmark results

Method	ODS	OIS	AP
UCM**	0.70	0.71	0.31
GMM(hsv+xy)	0.53	0.57	0.07
GMM(hsv)	0.49	0.52	0.05
GMM(lab)	0.49	0.51	0.04
K-means(hsv)	0.43	0.44	0.03
K-means(lab)	0.41	0.41	0.02
K-means(hsv+xy)	0.18	0.19	0.02
Watersheds	0.49	0.53	0.33
K-means(hsv)*	0.47	0.50	0.12
K-means(rgb)*	0.46	0.48	0.09

**Arbelaez et. al [1]

References

[1] P Arbelaez, M Maire, C Fowlkes, and J Malik. Contour detection and hierarchical image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(5):898–916, 2011.

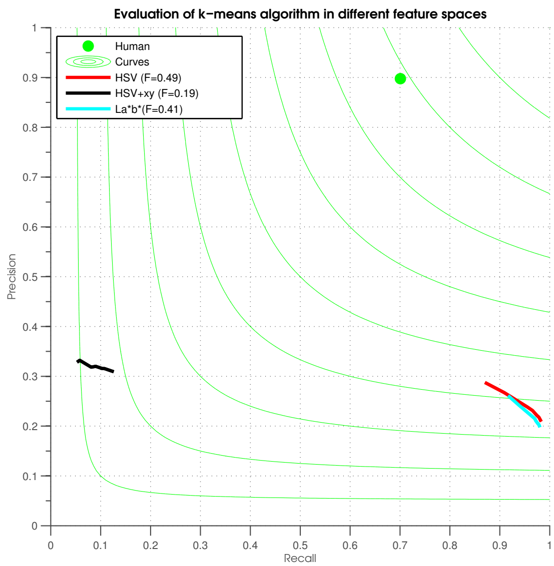

Figure 17. Precision recall curves for the three evaluated k-means segementations

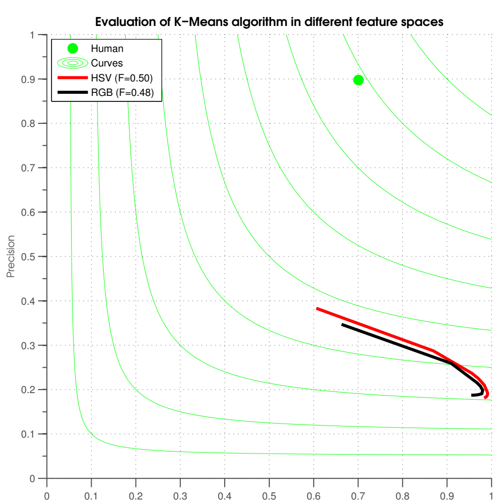

Figure 18. Precision recall curves for the three evaluated k-means segementations

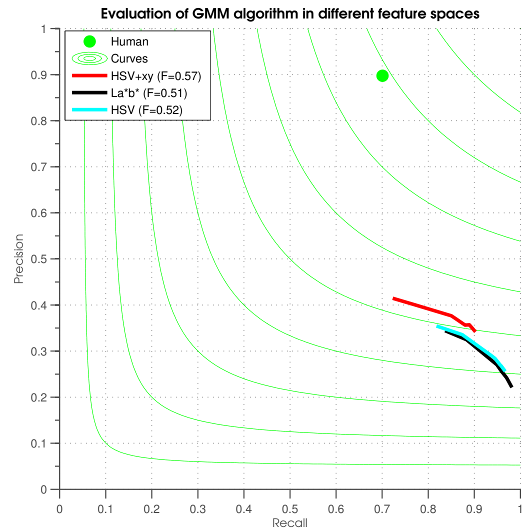

Figure 19. Precision recall curves for the three evaluated k-means segementations

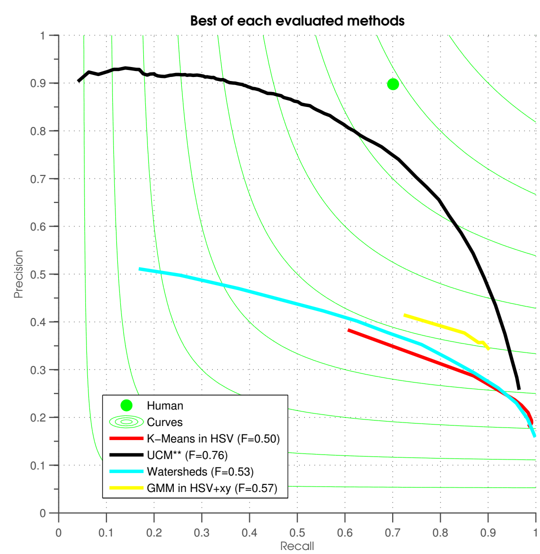

Figure 20. Precision recall curves for the segmentation methods. ** Arbelaez et al [1]

Figure 1

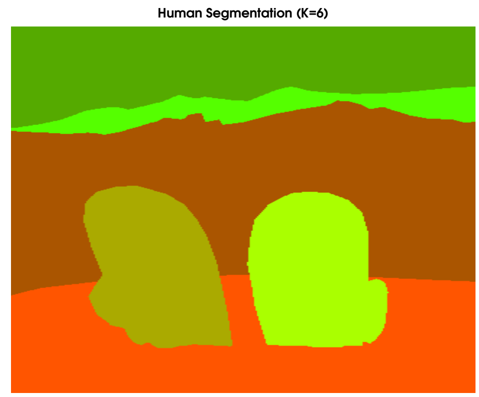

Figure 2

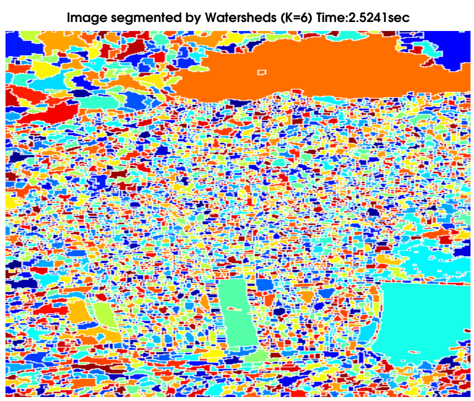

Figure 3

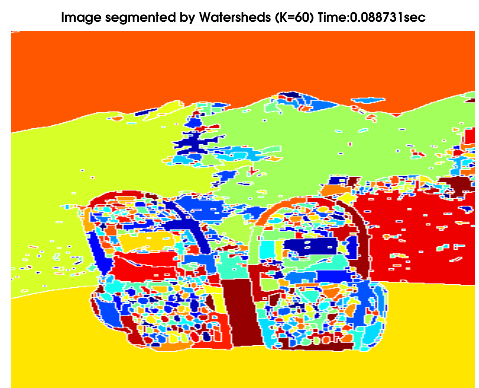

Figure 4

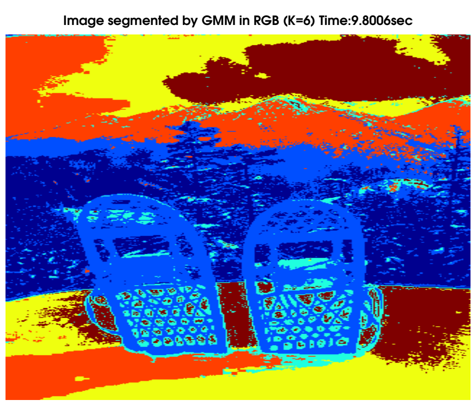

Figure 5

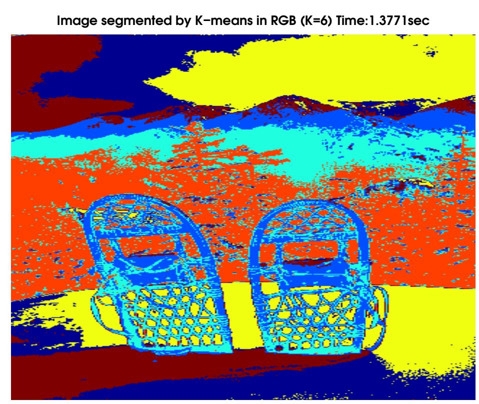

Figure 6

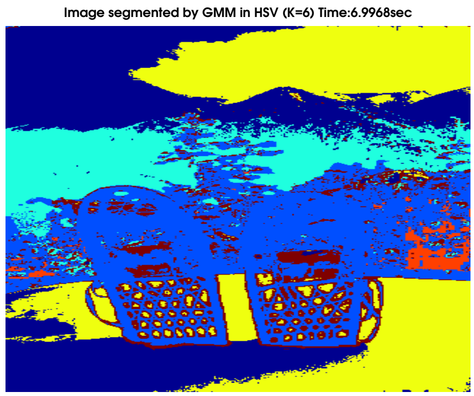

Figure 7

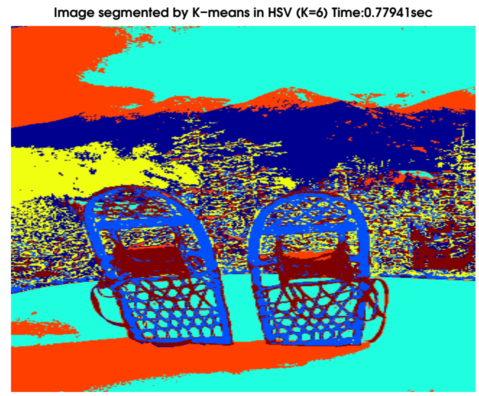

Figure 8

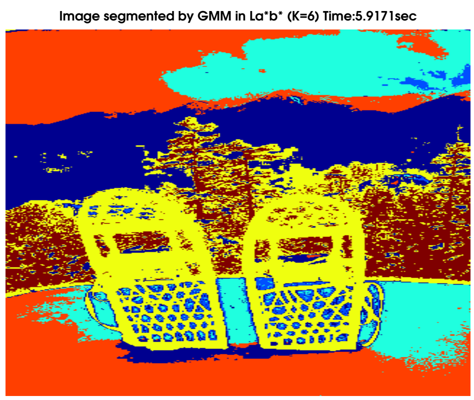

Figure 9

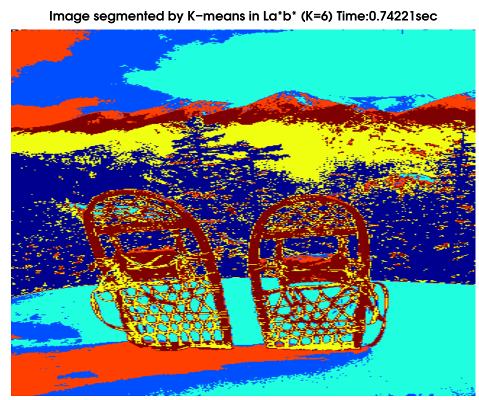

Figure 10

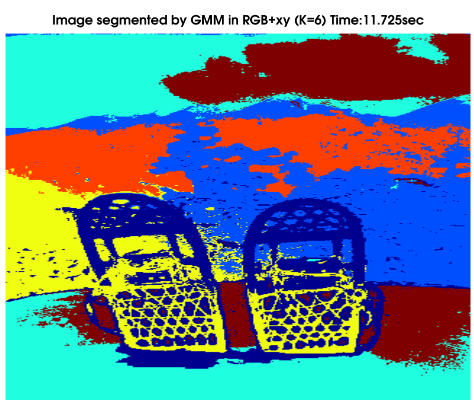

Figure 11

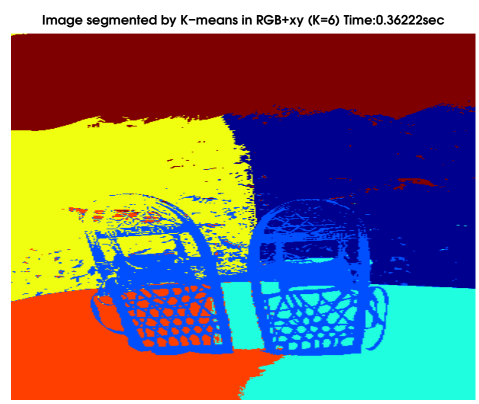

Figure 12

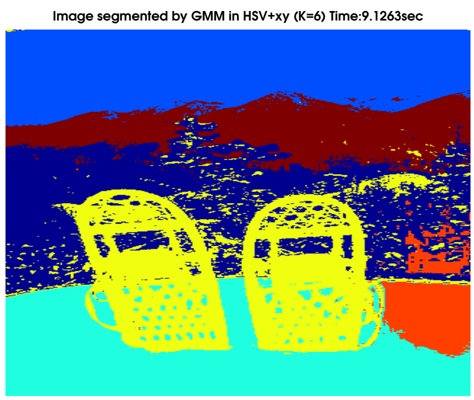

Figure 13

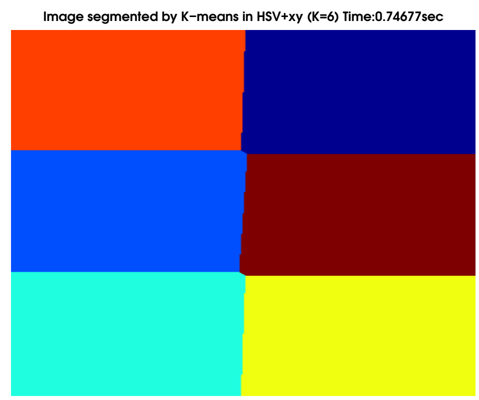

Figure 14

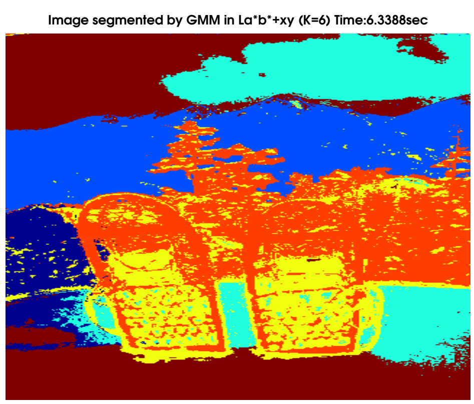

Figure 15

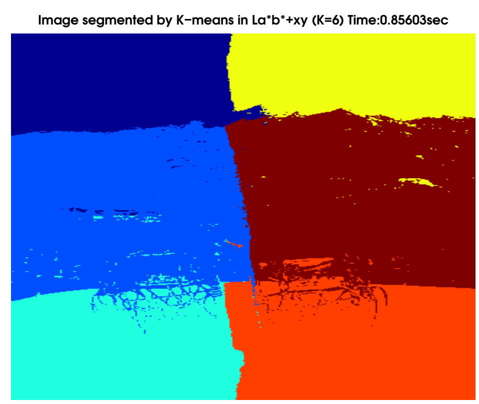

Figure 16