Segmentation

Maria Camila Escobar
Universidad de los Andes
Bogotá D.C, Colombia
mc.escobar11@uniandes.edu.co

Laura Gongas
Universidad de los Andes
Bogotá D.C, Colombia
l.gongas10@uniandes.edu.co

***Abstract*- Image segmentation is a growing area in the computer vision field. Nowadays, with the design of self driving cars robots that can move through spaces we are in need of efficient pattern recognition methods and scene analysis. Both of this applications have their grounds on a correct image segmentation. Here we develop an algorithm for segmenting images with three different color spaces and four clustering methods. We create our own evaluation metric and evaluate the performance of our method in a fragment of the BSDS database. Finally, for each clustering method we analyze the outcomes and conclude about the overall performance of our method.**

1. Introduction

Image segmentation is generally essential as a preliminary step in scene analysis and automatic pictorial pattern recognition problems [1]. Segmentation algorithms aim to group pixels in homogeneous regions generally by a common feature approach between pixels. Features might be represented using space of color, texture and gray levels [4]. Also, the position of the pixel can be taken into account as a way to take into account spatial information.

Clustering is a classification method that works by representing each pixel with a vector of measurements, then it's compared with a similarity measurement to other pixel vectors and finally each pixel is labeled at a corresponding cluster. In other words, clustering is an indicator of similarity of image regions and therefore it's useful for segmentation. Also, it's important to take into account that it's an unsupervised learning method and that it doesn't require labels. Clustering methods are divided into hierarchical, partitional and bayesian. The first one finds successive clusters using previously established clusters and partitional algorithms generally determine clusters at once.

K means is a partitional algorithm that assumes that the number of clusters is known. The main idea of the algorithm is to represent clusters with their centroids and find the centroids and clusters that minimize squared distances between elements and centroids. Lloyd's algorithm is an approximation that consists of: choosing k random centroids, assigning elements to clusters using a Voronoi diagram, computing new centroids and iterating until convergence is reached. The convergence is defined when there's no reassignments of elements to other clusters, no variance of centroids or when there's a minimum decrease in the sum of squared error [3].

Other algorithms of partitional clustering are model based like mixture of gaussians (gmm). The main idea of this method is to represent groups with gaussian distributions. In other words, clusters are formed by describing the probability density function of variables by a mixture of multivariate normal densities. Therefore, the objective is to find the parameters of a mixture of Gaussians that explain best the data [3].

Hierarchical methods can be divisive or agglomerative. This methods use dendrograms but the first one starts splitting all the data while the second one starts comparing pairwise data. This paper uses agglomerative clustering which starts merging the most similar pairs of clusters and stops when there's only one single cluster conformed by all the data.

Watershed is a grouping method for gray level images that considers the graphs of images as topographic surfaces. The domain of the functions are divided into catchment basins of regional minima and these are divided by watershed lines. The main idea is that a regional minimum can be represented as a whole in the topographic surface. Then the surface is flooded and when two lakes of different holes merge, a watershed line is constructed to separate the regions. However, images are generally over segmented so it's recommended to use marker based watersheds by imposing minima in the marker's positions. A way of obtaining automatic markers is by specifying a height h were regional minima are connected by a path of this height [3].

The objective of this paper is to create a region based segmentation function that uses as features the color spaces rgb, lab and hsv and also x and y spatial components if desired. Also, the method of segmentation can be chosen from k means, watershed, gmm and hierarchical. Additionally, an evaluation method was proposed.

2. Materials and methods

The implementation of the segmentation method is divided in two main parts, the feature extraction in each colorspace and the segmentation process by each clustering method.

Feature space

The method has six possible feature spaces. Three for the channels of each colorspace, rgb, lab and hsv and the other three for the channels of each colorspace plus the spatial x and y coordinates of the pixel. First, to obtain the image in lab and hsv representation we used the Matlab built-in functions *rgb2lab* and *rgb2hsv*. Then, we iterated through every pixel of the matrix to extract the features, whether it were only the three channels of the colorspace or the 5 channels including location. For the spatial coordinates we assumed that the upper left corner of the image was the pixel (1,1) and went through the whole image until the last pixel ath the bottom right corner. With these features we created a vector of m*n rows (m being the number of rows of the original image and n of columns) and the number of columns of the feature vector.

We normalized the values of the feature vectors in order to get a feature space with a coherent magnitude. For example, in Lab colorspace the value range for each channel differ, L goes in a range of [0,100] while a and b go in a range of [-128+127] [2]. To modify the magnitude of this space we added 128 to the value in a and b. Then we divided by 100 for the L channel and by 255 for the other channels. By normalizing the featurespace we ensured that all channels have equal weight in the segmentation.Despite, we found that the features that included colorspace and spatial position needed a downscaling in the spatial coordinates. This was done by dividing the value of the parameters by a factor of 2.

However, since watershed clustering method uses a matrix of the size of the image as parameter we decided to make a matrix out of the average values of the feature vector for each pixel. The results with matrix were then compared with the results of the complete features vector as part of hyperparameter choosing.

Clustering

For the segmentation phase, four different clustering methods were used, kmeans, gmm, hierarchichal and watersheds. The number of clusters for the output segmentation were determined by a qualitative analysis of each image in the dataset.

For kmeans, we ran the features matrix using the matlab function *kmeans* with 5 replicates to ensure that the convergence of the method was accurate.

Also, gmm was done by first creating a gaussian mixture model with the data in the features matrix via *fitgmdist*. Then, we used the function *cluster* to find the cluster configuration that would best adjust the data to the gaussian mixtures. Finally we obtained a vector of labels and the reshaped it with the same size of the original image.

For watershed, the gradient was obtained using a sobel edge mask and the gradient magnitude was computed. Originally, this method didn't work well for images of the database because the edges obtained with the gradient weren't closed. Therefore, the image was dilated with a diamond in a big enough size to join the different parts of the mask and give a closed edge result. Then, h minima were used as markers for watershed, the amount of labels of clusters was obtained for the specific h value and the process was repeated until the number of minima were equivalent to the amount of desired clusters.

Finally, for the hierarchical method it was necessary to rescale the images due to lack of memory and high computing time. Then, an agglomerative tree was constructed implementing the *linkage* function. Then, clusters were obtained employing the *cluster* function and the level of the tree used for the segmented image was determined by the number of clusters parameter.

Evaluation

For the performance evaluation of our method we decided to implement our own strategy. In this strategy we first have one of the groundtruth segmentations, we choose the first labeled region of the segmentation and use it as a mask to extract the same region in the segmented image created by our method. Then we calculate the number of pixels in that region of the segmented image that have the same label and we make a ratio of # pixels of the label in groundtruth over # of pixels with the same label in that section of segmented image. We calculate the ratio for every label of the groundtruth and the make an average of the points obtained for the image. To incorporate all versions of the groundtruth we run the same evaluation with each of the segmented ground truths and then calculate an average.

To improve the consistency of the results we decided to make another label ratio starting from the segmented image and comparing it to the different groundtruths, so it works under the same algorithm as previously explained. The final metric was obtained by making an average of both ratios. A value close to 1 means that the segmentation is very similar to all of the groundtruths and a value close to 0 means that the segmentation was poorly done.

3. Results

Because of the varying types of images in the dataset we decided to analyze the results of the methods with only one image from the dataset. By doing this, we can ensure that the difference in outcome of the method is due to the parameters of featurespace and clustering method and not because of the differences between images. We chose the image of the church because it did not have pronounced textures (since we are not using a texture descriptor) and it was possible to qualitatively select a number of 3 clusters.

The first hyperparameter that we tuned was the use of a set of feature vectors or a matrix including an average of the data in the vectors. We ran the segmentation algorithm with kmeans as the clustering method in Lab colorspace with both the features vector and the average marix and we measured the time it took to execute and the result of the evaluation metric. We obtained a metric of 0.5112 and a time of 1.29 seconds for the configuration that included the average matrix and a metric of 0.5097 with a time of 140 seconds for the features vector.

These results undoubtedly suggest that the best configuration should include the average matrix, not only because it gives a more accurate result but because it takes significantly less time to compute. The better result could be attributed to the fact that, since every channel is normalized, the average will give a representation of the image that can soften differences. Therefore if two pixels that should be in the same category have slightly different values in one channel but very similar values in the others they will have a better chance to bet classified together if the parameter is the average of all of their channels.

3.1. K means

In figures 1, 2 and 3 we can observe the differences between segmentations using rgb, Lab and hsv color spaces respectively. The overall results of the lab and rgb methods are acceptable, each method is able to segment the church from the background and then segment the stairs from the rest of the church. However, it is possible to observe that the hsv segmentation does not recover as much detail as the other two methods. The comparison in metric shows that, for this certain image, an Lab representation is the best possible configuration for kmeans.

Additionally, in figure 4, it is possible to see the results of the segmentation for rgb colorspace including the spatial coordinates. The results for rgb with spatial coordinates are better than the original method, this was achieved by giving a smaller weight in the feature vector to the x and y coordinates and a bigger weight to the rgb channels.Downscaling is important because the spatial coordinates are used to support the similarities between pixels, pixels that are next to each other are more likely to belong to the same category, but this does not give as much information of the image as the channel values.

However, adding spatial coordinates information to a representation matrix that is already giving inaccurate results will not help reduce the error in a significant way. This is shown in figure 5 where the already flawed hsv segmentation becomes worse by adding more parameters. Therefore, to achieve the best results with kmeans we have to use the colorspace that allows to differentiate between segments, in this case Lab, and then use the spatial coordinates as support to the classification.Also, it is possible to identify that the evaluation method is not precise, since the worst result, figure 5, provides the higher value.

3.2. GMM

The outcome obtained with this method was consistent with the results given by kmeans, the best featurespace was Lab, figure 6, and the worst was hsv, figure 7, both with spatial coordinates. The results of the metric with gmm were lower than kmeans but doing a qualitative analysis we are able to identify that the results for some segmentations, like 6 were better than with other methods. It is also possible to observe that the results vary significantly with each featurespace, more than it did using other clustering methods. The reason of this phenomena might be that gmm takes into assumption that the data is normally distributed, therefore it works best for data that fits the assumption.

3.3. Watershed

The most significant limitation observed for this method was obtaining the regional minima markers required to impose them to watershed. It's important to take into account that imposing minima is of extreme importance when implementing watersheds to reduce the superpixels output. However, markers are difficult to select because they're problem specific. However, for this paper, the h minima were selected as markers to impose minima and also to control the amount of clusters generated. The results for both Lab and rgb colorspaces were similar (figures 8 and 9), indicating that the imposed minima tends to be in similar places for both colorspaces. However, the results obtained with hsv were completely inaccurate. When taking into account the feature matrix representation for hsv+xy in figure 13 it is possible to see that the watersheds method will impose minima in the darker areas of the matrix. However, since these areas do not divide accurate components of the image the method will flood the segments that we are trying to divide and end up with small separate clusters.

Additionally, based on the evaluation results presented in figure 10, we can see that the evaluation method proposed isn't very appropriate for the cases were the majority of the image is labeled with the same value.

3.4. Hierarchical

An important limitation of this method was that it was very slow for large scale images. Therefore it was necessary to resize the images. However, as it can be observed in figure 11, important information of the borders and object's shape is lost. Also, large amount of clusters require to resize even more the image which results in loosing a lot of information. However, it's easy to change the maximum number of clusters desired because the tree is already constructed. To obtain more accurate results it's recommended to study other methods that describe features in smaller size vectors.

Finally, a common factor in the configurations with all of the clustering methods was that Lab gave the best results while the results given by hsv were completely inaccurate. In figures 12 and 13 we can see the feature matrix for both colorspaces. The Lab matrix has a great resemblance to the original image and it is possible to identify different objects in it.On the other side, the hsv feature matrix is not coherent with the objects that are in the image and it distinguishes small objects that are not originally supposed to be separated. Overall the final results can be seen in figure 14, the best configuration is gmm classifier with Lab colorspace and the worst configuration corresponds to watersheds with hsv space.

4. Conclusions

It is possible to identify that the evaluation method is not completely coherent with the qualitative results. Our evaluation method works well for various categories in which there will always be a small region corresponding to the cluster. However, when the image is segmented in only 2 parts, the regions are big and the evaluation fails to give a correct metric. Because of this, it is possible to use our metric to compare results as long as they consist of medium and small sized clusters. We calculated the average human performance of our metric using the different groundtruths and found a value of 0.5339 therefore, a better enhancement of this metric could take into account the default discrepancies in human performance.

It was possible to observe the importance of using different color spaces because depending on the application, some channels were more discriminative and this aided for a better segmentation. The Lab space, for example, yielded the best results while the hsv yielded the worst. For future studies it's recommended to try methods with automated definition of optimal amount of clusters. Also, an evaluation method such as precision recall curves could be implemented. The methods should be tested with more images and other features such as texture could be taken into account to obtain better segmentation results.

Finally, it is not possible to give a full comparison between clustering methods since the metric we developed is not fully consistent. Therefore, the only way we could give an approximation of which method is better is by doing a mixture of the metric with qualitative analysis and in segmentation methods a qualitative analysis is not a correct approach.

References

[1] Visual computation and multimedia. *Universidade de Beira Interior*.
[2] Introduction to colour spaces, 2018.
[3] P. Arbelaez. Lecture 5: Clustering. *Universidad de los Andes*, 2018.
[4] D. Naik and P. Shah. A review on image segmentation clustering algorithms. *International Journal of Computer Science and Information Technologies*, 5(3):3289–3293, 2014.

Additional attachments

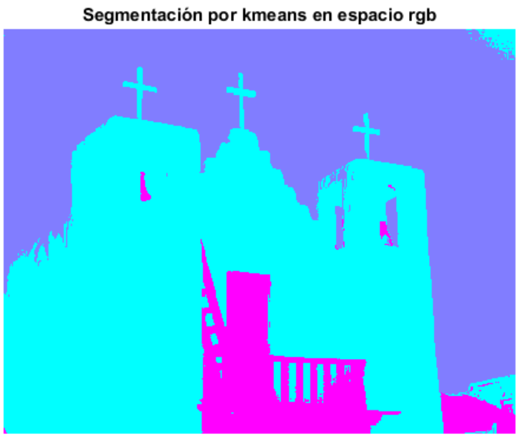

Figure 1. Image segmentation for kmeans with rgb colorspace. Evaluation metric=0.502

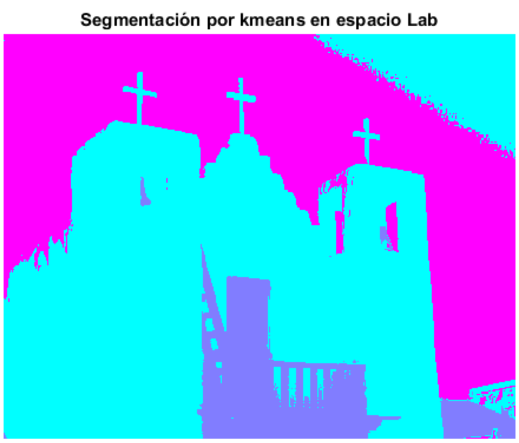

Figure 2. Image segmentation for kmeans with lab colorspace. Evaluation metric=0.5112

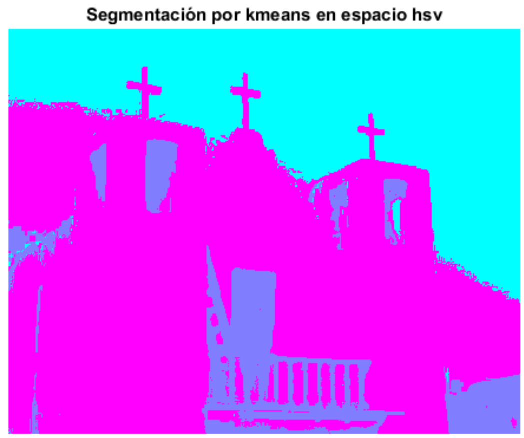

Figure 3. Image segmentation for kmeans with hsv colorspace. Evaluation metric=0.355

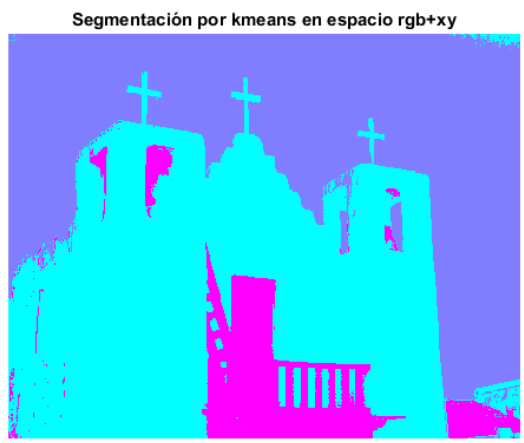

Figure 4. Image segmentation for kmeans with rgb+xy colorspace. Evaluation metric=0.510

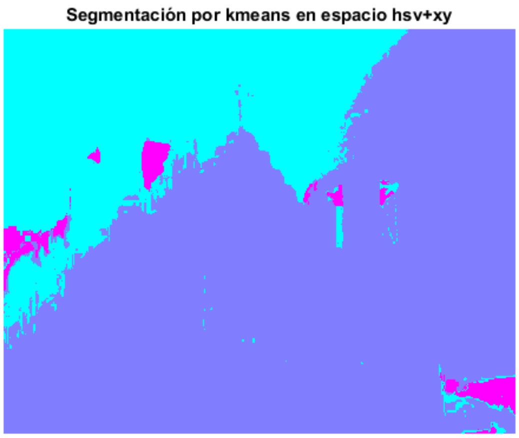

Figure 5. Image segmentation for kmeans with hsv+xy colorspace. Evaluation metric=0.52

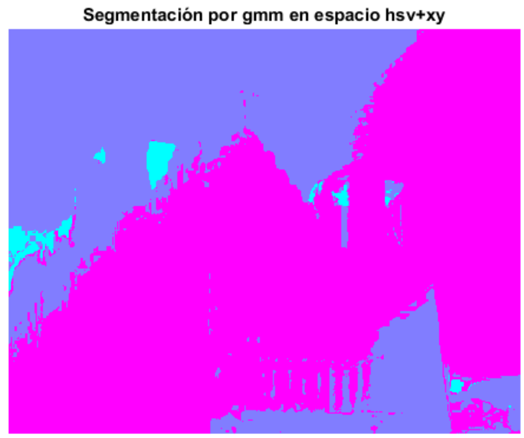

Figure 7. Image segmentation for gmm with hsv+xy colorspace. Evaluation metric=0.446

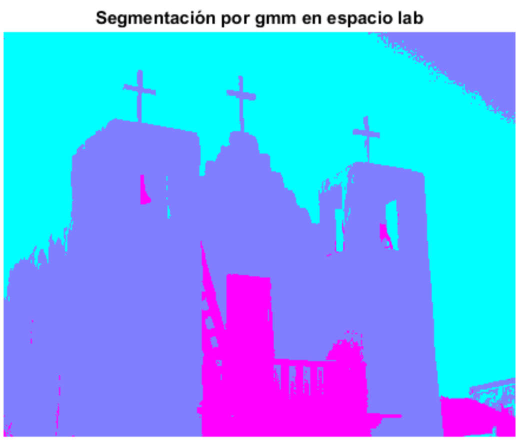

Figure 6. Image segmentation for gmm with lab+xy colorspace. Evaluation metric=0.3984

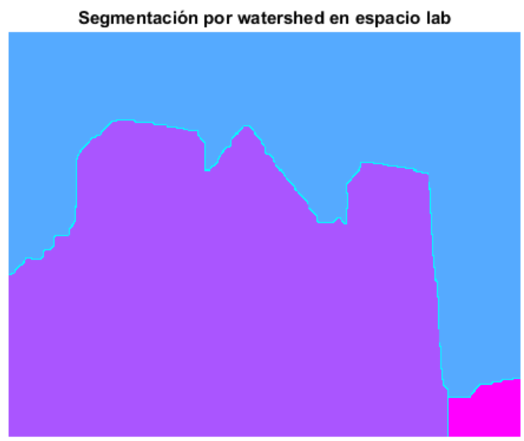

Figure 8. Image segmentation for watershed with Lab colorspace. Evaluation metric=0.5227

Figure 9. Image segmentation for watershed with rgb colorspace. Evaluation metric=0.497

Figure 10. Image segmentation for watershed with hsv+xy colorspace. Evaluation metric=0.5591

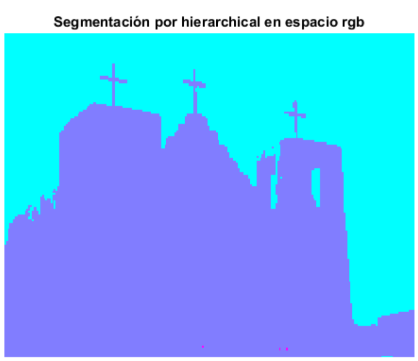

Figure 11. Hierarchical segmentation with the image resized to 50% and 2 clusters used. The evaluation metric yielded 0.642.

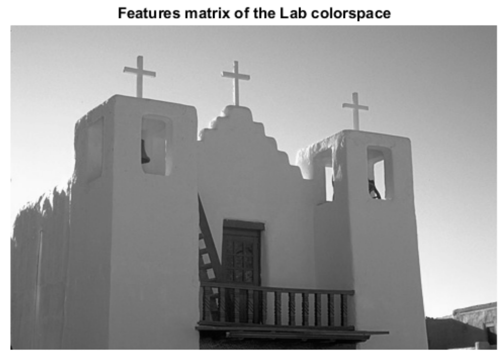

Figure 12. Feature matrix for Lab colorspace

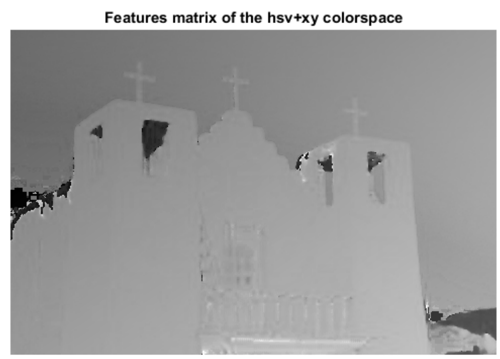

Figure 13. Feature matrix for Hsv colorspace

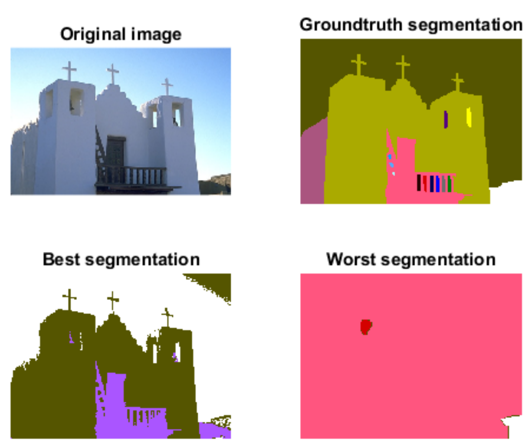

Figure 14. Final results of the algorithm