Textons

Johana M. Ramirez Borda
Los Andes University
jm.ramirez11@uniandes.edu.co

Francisco A. Rozo Forero
Los Andes University
fa.rozo1843@uniandes.edu.co

Abstract

On of the most studied problems in the image processing field is the segmentation one. This problem is based on the need to cluster o regroup the pixels of an image that belongs to the same object or region. In this project, four clustering methods (kmeans,gmm,hierarchical and watershed) were tested in order to segmentate an RGB image. Also, LAB ans HSV spaces were taken into account. Moreover, spatial information such as the (x,y) coordinates of each pixel were used. In general, the kmeans and gmm methods showed similar -and visually good- results. On the contrary, watersheds didn't show a good performance even when varying the number of clusters used.

1. Introduction

The segmentation problem in image processing is based on the clustering of pixels that belongs to the same regions or objects. Therefore, segmentation can be used for objects recognition, occlusion boundary estimation, image editing, etc [2].

In this project, several segmentation strategies were tested by varying between four clustering methods: kmeans, gmm,hierarchical and watershed. Also, RGB, LAB and HSV feature spaces were used. However, not only color information was used in the clustering methods, but also the spatial information by taking into account the *(x,y)* coordinates of each pixel: RGB+xy, LAB+xy, HSV+xy.

Firslty, kmeans clustering performs partitions of the color channels of the image into k clusters and return the cluster index of each pixel. This information allows to classify each pixel within a region or object and then display it on screen with a specific color.

Second of all, the gmm (gaussian mixture distribution) method, fits a gaussian mixture distribution with k components to the color channels of the image. Then, k clusters are created and this information is reshaped into an image.

On the other hand, hierarchical clustering is an strategy to built a hierarchy of groups. This method has two main approches: in the agglomerative approach, each cluster starts by beeing one single pixel and then pairs of clusters are merged; in the divisive approach, all the pixels are contained in one cluster and then splits are performed recursively[2].

Finally, the watershed clustering method was used to "cluster" histograms of the image. The main idea is to consider the gray level version of the image and consider it as a topographic relief, in which actuating an inmersion process[1].

2. Materials and methods

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

- Kmeans
- watershed: This function segments the image based on the minimum markers.
- imextendedmin: computes the extended-minima transform, which is the regional minima of the H-minima transform. h is a nonnegative scalar. (source: mathworks)
- imimposemin: Based on a marker, marked pixels become 0.
- linkage: This function creates the hierarchical tree and assigns a value to each row which correspond to their position in the tree.
- dendrogram: Once the hierarchical tree is created, it prints it depending on the number of clusters. Also, it gives the value of the assigned cluster to each pixel.

2.1. Image data pre-processing

Before processing, the pixels of the image were rearranged into a vector, starting in the upper left position and moving up to down and left to right. The first three columns corresponded to each color channel RGB/HSV/LAB in that order, the last two columns were used only when the coordinates are also included, being x and y position the 4th and 5th column respectively. However, these arrange only applies for the kmeans, gmm and hierarchichal methods.

On the other hand, when taking into account the *(x,y)* coordinates, it was found an inconsistency because the pixels of the image was in a range of $[0, 1]$ and the coordinates were much larger than 1. To overcome this, each vector of *(x,y)* coordinates was normalized. The vector of *x* coordinates between the number of rows and the *y* coordinates between the number of columns.

2.2. Kmeans Segmentation

The matrix of the pixels with their respective data values was used on the kmeans function to separate them. The distance metric implemented was euclidean, which has similar results to the cityblock method.
Moreover, other distances such as *cosine* were tested but the performance of the segmentation of lowered (See in *Anex* section).

2.3. GMM Segmentation

In this method, the function *fitgmdist* was firstly used in order to study the matrix that contained the three channels (and the coordinates when necessary) of the imagem and to fit a gaussian mixture distribution with k components. Then, the function *cluster* was used in order to create the k clusters that were regrouped to form the segmented image.

2.4. Hierarchical Segmentation

This method takes every pixel as a single class and then by measuring with the euclidean/cityblock metric, it pairs those that are closer until they all become one unique cluster. For this method, the function linkage revalued the pixels to position them on the tree based on their distance, then the dendrogram function displays the dendrogram and gives the assigned group for each pixel.

2.5. Watershed Segmentation

This method consist on h ”flooding” through the regional minimums and barricade where these floods join together. Being these barriers highest peaks between minimums. This way, the objects are segmentated based on their gradient.

3. Results and discussion

Although the function was tested for several images, the results are shown just for the one image, varying the different clustering methods and the color space. The k was set to 7 and for watersheds, the K was set to 20, because instead of segmentating by clusters, it segmentates based on its regional minimum and the k represents how much the image will be flooded.

3.1. Kmeans

Simple RGB segmentation gives a good result even though there is noise in some zones of the image, which says that it is susceptible to outliers. In contrast, adding the xy data separates it more by zone than by objects. Lab colorspace’s segmentation was not too different than using the xy coordinates. Also, the results were similar to RGB. Finally, the HSV gave an alternative segmentation to the RGB and Lab colorspaces, but when including the xy coordinates, the big objects are separated in more than one section.

3.2. Gmm

Simple RGB and RGBxysegmentation gives similar results of HSVxy in kmean, but it separates more the details of the big objects. The Lab gave a bad segmentation with lots of outliers for each cluster. In contrast, the Labxy does not have that many outliers and separates better the objects. Finally, both of the HSV segmentations are highly affected by the shadows of the image and don’t give a good segmentation.

3.3. Hierarchical

Both of the RGB colorspaces have many outliers and only segmentates well one object (flowers). Simmilarly are the results of the HSV colorspace, which also missegmentates the flowers. The HSVxy colorspace gave a better separation of the objects while ignoring the details inside of them. While the Lab colorspaces behave similar than the Lab of the kmeans method.

3.4. Watersheds

The HSV colorspace wasn’t too significative while segmentating, because it is more sensitive to the *k* used. However, when using HSV with the xy position at a lower k, the amount of regions rose and gave better details. The RGB colorspace had many regional minimum, specially in the objects. Also, in this colorspace the results are the same as the RGBxy. Simmilarly, the Lab colorspace had the same results as the Labxy, which indicates that the position of the pixels is not relevant in the watersheds method. Also, the Lab gave less regional minimum than the HSVxy. Finally, if the k increases the objects marked slowly fade until the entire image is one unique cluster.

3.5. Evaluation

Finally, it is well known that a common technique used as a evaluation strategy in segmentation problems is the Jaccard index. This evaluation metric allows to compare the Ground-truth and the segmented image. The index has a high value when the segmented image is similar to the ground truth and a low value otherwise. In this project, an attempt to evaluate the segmentation through this index was done. However, the numerical result was not consistent with the visual one: for example, the value of the index obtained in the kmeans method was 0.0123 when it was visually clear that this method had a better performance than the others.

4. Conclusions

According to the results, the watershed method didn't care about the position of the pixels, because the results were the same regardless of the position data. Also, the hsv colorspace was too sensitive to the parameter k. The Lab colorspace gave the better results when separating big objects and details inside the objects. However, the hierarchical method wasn't good for this colorspace. The HSV colorspace with coordinates gave the better results when separating objects/regions because it also ignored the details inside of them. Finally, The RGB colorspace performed better with the GMM method.

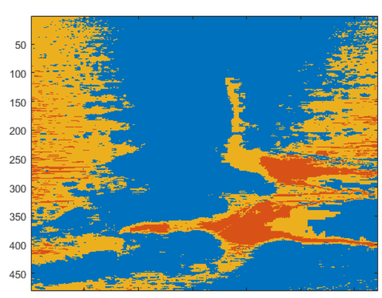
References

[1] M. Bicego, M. Cristani, A. Fusiello, and V. Murino. Watershed-based unsupervised clustering. In *International Workshop on Energy Minimization Methods in Computer Vision and Pattern Recognition*, pages 83–94. Springer, 2003.

[2] A. Jepson and D. Fleet. *Reading on Segmentation*. 2007.

All the codes created and used are attached.

5. Annex



,

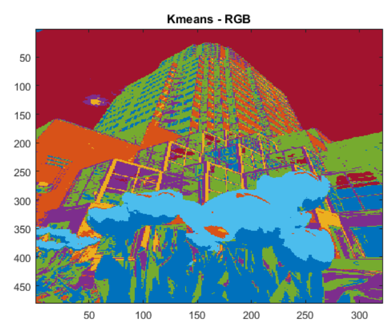
Figure 1. Example of the lowerd performance of the kmeans clustering when usinf the cosine distance.

,

Figure 2. Image to segmentate

5.1. Kmeans Results

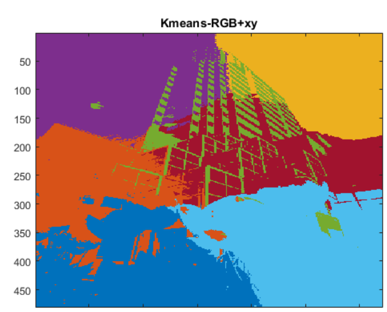
5.1.1 RGB



,

Figure 3. Segmentation through kmeans in RGB space

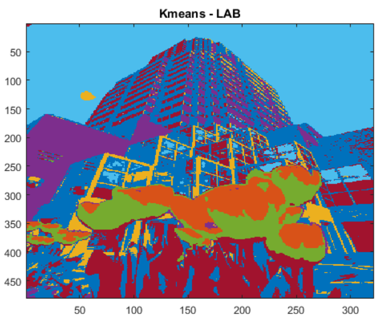
5.1.2 RGB+xy



,

Figure 4. Segmentation through kmeans in RGB+xy space

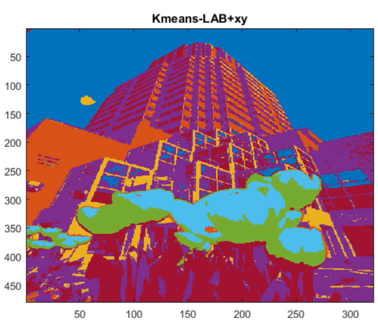
5.1.3 Lab



,

Figure 5. Segmentation through kmeans in LAB space

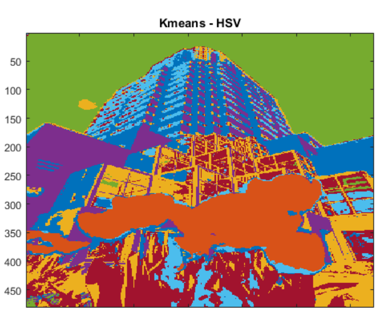
5.1.4 Lab+xy



,

Figure 6. Segmentation through kmeans in LAB+xy space

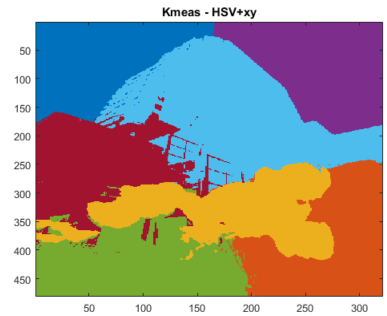
5.1.5 HSV



,

Figure 7. Segmentation through kmeans in HSV space

5.1.6 HSV+xy

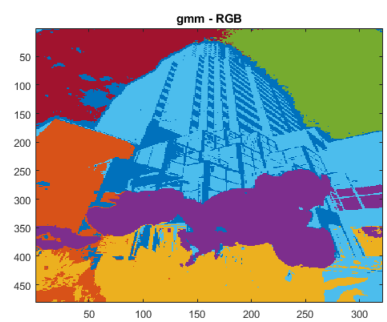


,

Figure 8. Segmentation through kmeans in HSV+xy space

5.2. GMM Results

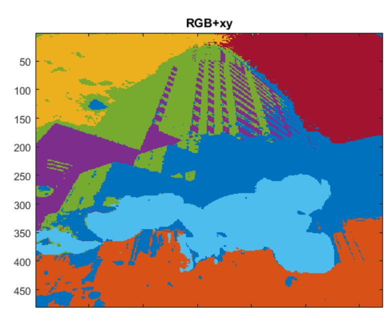
5.2.1 RGB



,

Figure 9. Segmentation through gmm in RGB space

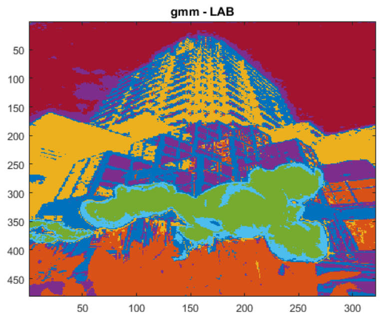
5.2.2 RGB+xy



,

Figure 10. Segmentation through gmm in RGB+xy space

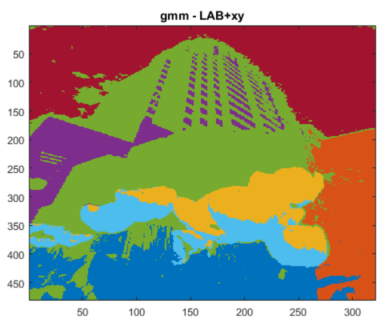
5.2.3 Lab



,

Figure 11. Segmentation through gmm in LAB space

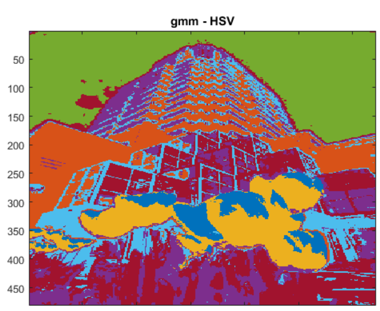
5.2.4 Lab+xy



,

Figure 12. Segmentation through gmm in LAB+xy space

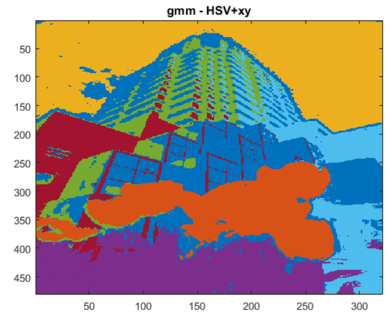
5.2.5 HSV



,

Figure 13. Segmentation through gmm in HSV space

5.2.6 HSV+xy

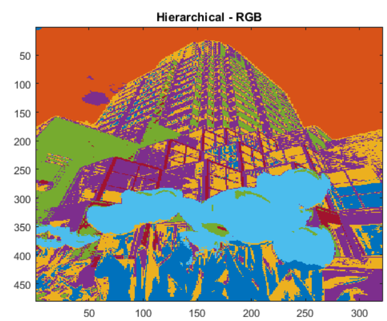


,

Figure 14. Segmentation through gmm in HSV+xy space

5.3. Hierarchical Results

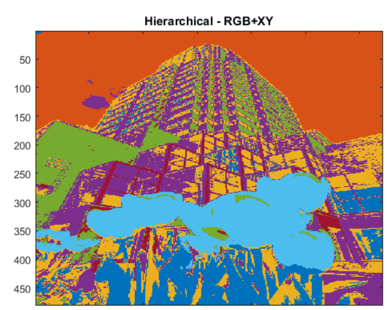
5.3.1 RGB



,

Figure 15. Segmentation through hierarchical in RGB space

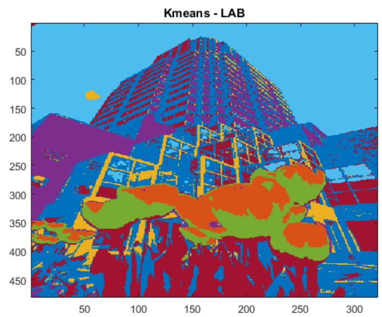
5.3.2 RGB+xy



,

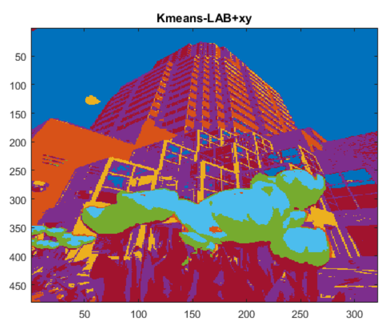
Figure 16. Segmentation through hierarchical in RGB+xy space

5.3.3 Lab



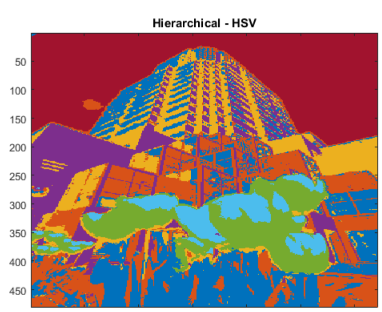
,
Figure 17. Segmentation through hierarchical in LAB space

5.3.4 Lab+xy



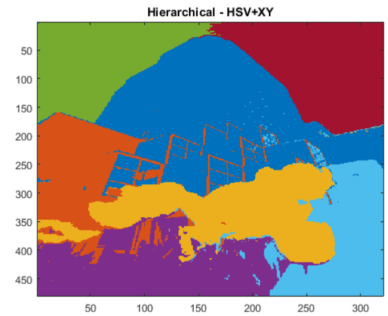
,
Figure 18. Segmentation through hierarchical in LAB+xy space

5.3.5 HSV



,
Figure 19. Segmentation through hierarchical in HSV space

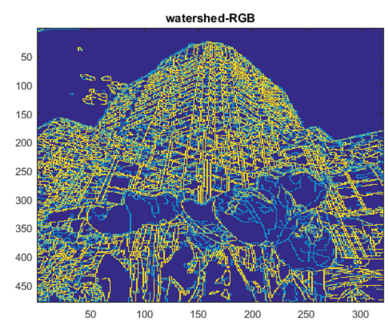
5.3.6 HSV+xy



,
Figure 20. Segmentation through hierarchical in HSV+xy space

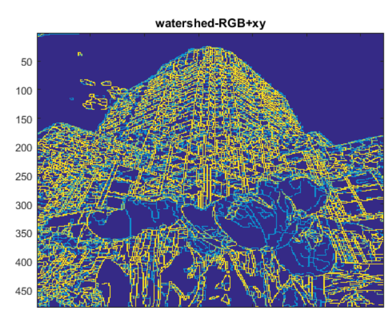
5.4. Watersheds Results

5.4.1 RGB



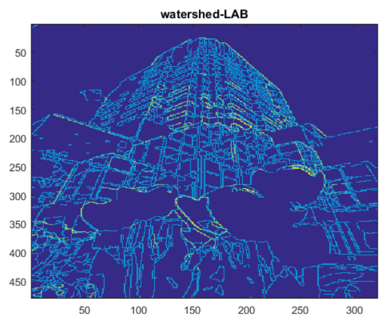
,
Figure 21. Segmentation through watersheds in RGB space

5.4.2 RGB+xy



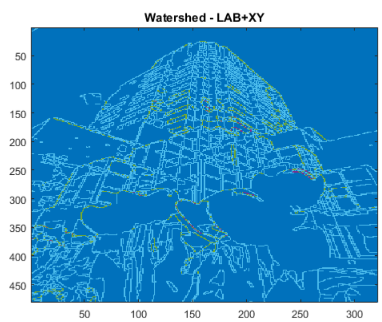
,
Figure 22. Segmentation through watersheds in RGB+xy space

5.4.3 Lab



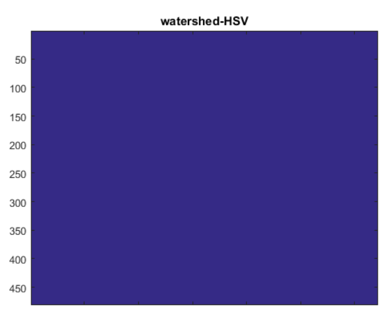
,
Figure 23. Segmentation through watersheds in LAB space

5.4.4 Lab+xy



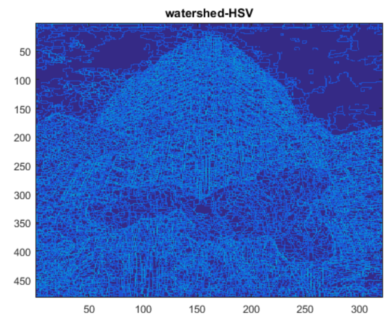
,
Figure 24. Segmentation through watersheds in LAB+xy space

5.4.5 HSV



,
Figure 25. Segmentation through watersheds in HSV space

5.4.6 HSV+xy



,
Figure 26. Segmentation through watersheds in HSV+xy space