Image Segmentation Testing

Julio Nicolás Reyes Torres
Universidad de los Andes
jn.reyes10@uniandes.edu.co

Juan David Triana
Universidad de los Andes
jd.triana@uniandes.edu.co

Abstract

This article presents the testing of two segmentation functions using the complete database (BSD500), the implemented methods were chosen previously because of its behavior was stood out within a set of 4 different algorithms using a short version of the dataset. The methods switch with different hyperparameters which have the property of changing the segmentation modes and implicitly its accuracy. In this article is reviewed how is the behavior of the two methods but now using the complete database and as a bonus part, our two segmentation functions are proved against the UCM algorithm designed by the teacher' team, which is based into reducing the problem of image segmentation to that of contour detection, looking for to approach to his result.

1. Introduction

A way to extract information from an image is to use the segmentation technique that seeks to decompose an image among its components, that is, extract each object and the background that belongs to it [3, 10]. The mechanism for evaluating each segmented object is to identify whether a pixel belongs to a region of homogeneous characteristics, such as color and intensity, texture, magnitude of the gradient, direction of edges, shades of gray, in general, any parameter that relates similarities between regions in order to discriminate the zones that directly identify each object. [1]

Previously, a work was developed regarding the evaluation of grouping methods applied to the segmentation of images. In that work, it was determined that among the algorithms evaluated, the ones that had the best efficiency were: K-means and hierarchical segmentation. The hyperparameters and conditions evaluated for each method were tested in a small sample of the database.

The current goal is to test the two segmentation functions in the complete database (BSDS500) to assess effectiveness. Additionally, using a graph of presicion-recall we present the comparison of our results with the OWT-UCM method that combines the advantages of the segmentation methods: Oriented Watershed Transform and Ultrametric Contour Map (UCM) [6].

2. Methods

2.1. Data description

In this practice the resources are supplied into folder which have the respective data and code to implement the UCM algorithm and the complete Benchmark database (BSDS500) [12]:

- BSDS500 Large Dataset: images, ground-truth data and benchmarks.
- ucm2: Ultrametric Contour Map segmentation algorithm.
- Documentation: Contour Detection and Hierarchical Image Segmentation (article)[6]

The data set is divided into three different folders: train, test and validation. Test and train folders contain 200 481X312 images each and validation folder contains 100 images of the same resolution.

2.2. Segmentation Methods

K-means

It is one of the simplest unsupervised learning algorithms to solve the problem of clustering. It seeks to group objects in **k** clusters depending on their characteristics. The algorithm solves an optimization problem by minimizing the sum of the quadratic distances each object to the centroid of its cluster [13]. In images, the intensity of the color is the quantization vector that provides the information about the attributes in an image. [5]

Hierarchical

Rather than the other methods, the main idea of hierarchical clustering is to not think of clustering as having groups to begin with [4]. The main goal of Hierarchical clustering's is be depicted as a tree or dendrogram of fusions between clusters [2]. The method implemented was the named **Ward**, the only one among the agglomerative clustering methods based on a classical sum-of-squares criterion, it produce different groups that minimize within-group dispersion at each binary fusion.[9]

The algorithms K-means and hierarchical segmentation will be used to be tested in the complete database, because they were the ones that showed the best performance values in the segmentation. The number of clusters (k) was the most sensitive parameter, for a wide range of tests, these algorithms showed an average fidelity in the data.

No type of modification was made in the segmentation functions because the processing time was not considerably extended in the tests that were done in the small database.

The methods implemented have some hyperparameters, the ones that were used are:

K-means

- **n clusters (int, optional, default: 8):** The number of clusters.
- **random state (int, RandomState instance or None (default)):** Determines random number generation for centroid initialization

Hierarchical

- **n clusters (nt, default=2):** The number of clusters.
- **linkage("ward", "complete", "average", "single"):** Which linkage criterion to use. The linkage criterion determines which distance to use between sets of observation.
- **connectivity (array-like or callable, optional):** Connectivity matrix.

The main hyperparameter is the number of clusters (k), clearly it was the most sensitive parameter to the changes reflected in the final segmentation, a number of low groupings, does not represent all the linked objects in the images, but a very high value generates a over-grouping that does not allow to differentiate objects either. Therefore, choosing a suitable value is of vital importance for a good segmentation.

2.3. OWT-UCM algorithm

This algorithm was introduced by "Pablo Arbeláez et al" in the article "Contour Detection and Hierarchical Image Segmentation", it was a significantly better solution than its competitors. it presents a formal solution to the image segmentation problem implementing an algorithm that uses the contour detected into a hierarchical region tree. [6]

The following results were obtained by Pablo and his collaborators, illustrated as a precision-recall curve.

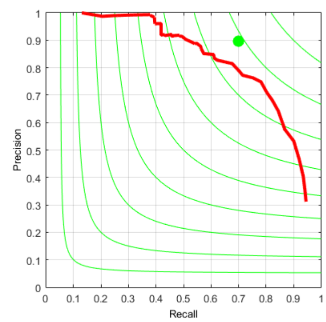


Figure 1: Precision and recall curve for the OWT-UCM method

2.4. Evaluation

Quantitative evaluation metrics for segmentation can be separated in various categories. Spatial overlap based, volume based, pair counting based, information theoretic based, probabilistic based, and spatial distance based [7]. These kinds of evaluation give a measurable performance perspective towards a segmentation method. A more powerful way to evaluate an algorithm, as it can also be evaluated qualitative manner. These methods use human evaluation through visual perspective [8].

For this specific practice, the BSDS benchmark evaluation was used. This broad benchmark introduces three different metrics entitled to evaluate different aspects of segmentation algorithm. These are: variation information (cluster comparison), rand index (general cluster evaluation), and segmentation covering (pixel-wise classification) [6]. Since each of our unsupervised methods must include a specified cluster number (threshold), it is necessary to implement a more broad and general metric to show a description of the images using one numerical value. This is why it is useful to use the optimal dataset scale (ODS), which uses a fixed threshold for the entire training dataset to optimize the output. Moreover, an optimal image scale (OIS) per image metric can show the specified algorithm's performance. These can be applied to segmentation covering and rand index.

Another way to measure a segmentation algorithm's effectiveness is through the precision-recall curve. This metric illustrates a trade-off between precision and recall for different thresholds. It is calculated using the following formulas:

$$P = \frac{T_p}{T_p + F_p}$$

$$R = \frac{T_p}{T_p + F_n}$$

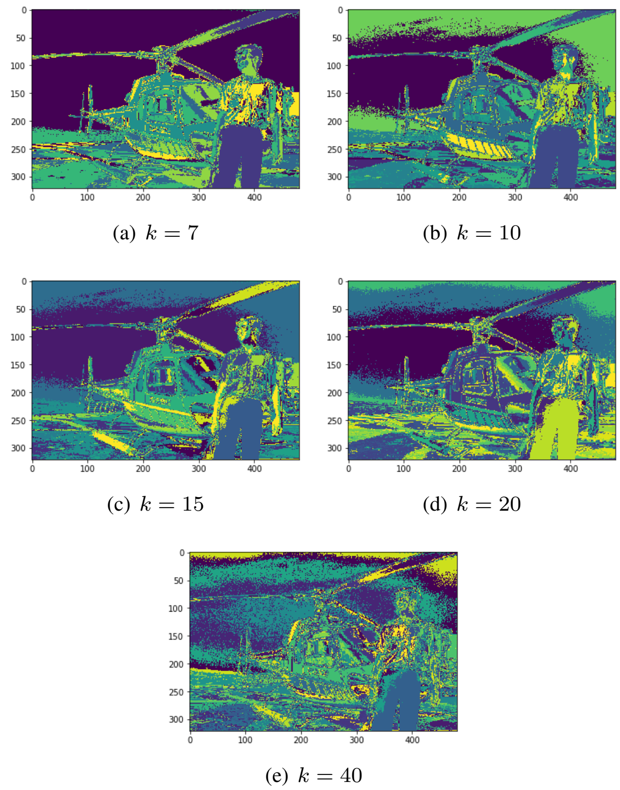
where P and R are the precision and recall values respectively, T_p the true positive values taken into account by the algorithm, F_p the false positive values, and F_n the false negative values [11]. Most likely, a high recall would mean a low precision, and vice versa.

3. Results

Note: First, the visual result of segmenting an image for the different K is shown using the K-means and Hierarchical algorithms. After the comparison between methods is presented using a precision-recall graphic to measure the performance of each segmentation algorithm.

- Image = $146074.jpg$

3.1. Segmentation using K-means



(a) $k = 7$ (b) $k = 10$ (c) $k = 15$ (d) $k = 20$ (e) $k = 40$

Figure 2: **K-means Segmentation.** Images representing the segmentation for different chosen number of clusters (7,10,15,20,40) using K-means algorithm.

After having the segmentation images, we obtained the following precision-recall curve, showing the trade-off using different number of clusters for k-means segmentation.

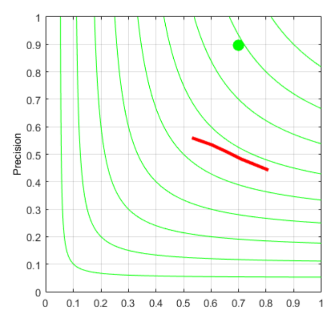
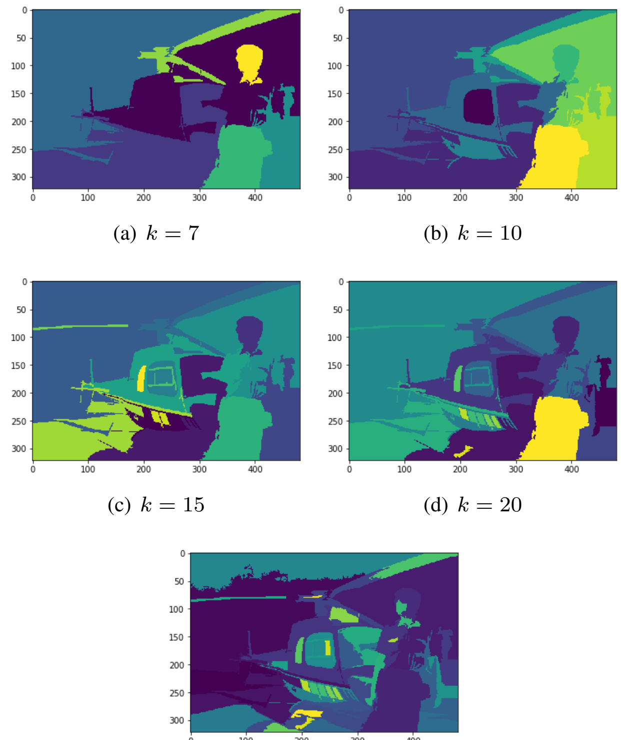


Figure 3: Precision and recall curve for Hierarchical clustering segmentation for five different cluster numbers (7,10,15,20,40)

Regarding the scale dataset validation on GT covering, the following results refer to k-means clustering segmentation.

Metric	Score
ODS	0.28
OIS	0.29

3.2. Segmentation using Hierarchical clustering



(a) $k = 7$ (b) $k = 10$

(c) $k = 15$ (d) $k = 20$

(e) $k = 40$

Figure 4: **Hierarchical Segmentation.** Images representing the segmentation for different chosen number of clusters (7,10,15,20,40) using Hierarchical algorithm.

After having the segmentation images, we obtained the following precision-recall curve, showing the trade-off using different number of clusters for hierarchical segmentation.

Regarding the scale dataset validation on GT covering, the following results refer to hierarchical clustering segmentation.

Metric	Score
ODS	0.4
OIS	0.45

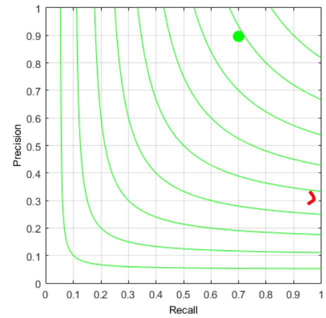


Figure 5: Precision and recall curve for KMeans clustering segmentation for five different cluster numbers (7,10,15,20,40).

4. Discussion

In the figure (2) is presented the segmentation using K-means for a different number of clusters, unfortunately, the more number of clusters, the more there is over-segmentation and the specific information on each object is lost. Besides, the clusters are going to provide specific information about color and that's why this algorithm is sensitive to outliers.

On the other hand, it is observed that in the hierarchical method (figure 4), a greater number of clusters there is a better grouping for each object and a segmentation is more visible.

The hierarchical clustering was the best method because it produces a richer structure than a fixed partition as K-means. It produces a dendrogram Bottom Up merging the most similar pair of clusters, so it is more flexible between distances in clusters and is more adaptable. Also, observing both tables shown on the results section, it was demonstrated that the hierarchical method was better since both values (ODS and OIS) were greater. This means that the hierarchical method has a better score regarding the entire dataset evaluation as well as an individual image segmentation evaluation.

Also, this algorithm continue being the best even in this complete database (BSDS500) comparing it with the other method chosen by us (K-means), because of the characteristics mentioned above.

Our implemented function for segmenting images didn't beat Pablo's algorithm because our method only use the properties of each one (K-means, Hierarchical) to segment clusters with supposed similar characteristics, instead, OWT-UCM algorithm implements a robust solution to segmentation using an algorithm that constructs hierarchical segmentations from contours without losing performance on the boundary benchmark.

Unsupervised clustering for segmentation can have a big amount of limitations. As it was seen on figures 2 and 4, clustering needs a specific amount of clusters for it to be optimized. This number varies depending on the image's objects and lighting. For example, an image that has 5 objects in it would be hard to avoid over segmentation using a random number of clusters that are over 5. Moreover, the feature descriptor used on the algorithm training was based on plain pixel intensity values. This is not a good descriptor since spatial units matter in segmentation problems.

There are clear error patterns on out algorithms. As it was mentioned before, segmentation is based on a number of clusters. In this case, objects contained on each image are different in size and number, which means that evident errors will be held using the same amounts of clusters. This is why our algorithm had such a low ODS and OIS for GT coverage.

Having said that, our algorithm needs to have a spatial descriptor for each object segment as a pixel. Moreover, it would be useful to implement a hierarchical boundary detection to be able to separate objects within an image that may be distorted by lighting or image perception. This way, specific object details can be separated from the actual object boundaries. As it was implemented by Pablo, probability of object detection can be very useful to integrate the hierarchical levels of the clustering segmentation. This means that it would be possible to merge objects even though a first over segmentation was held before.

5. Conclusions

- K-means algorithm assumes a number of clusters (k) representing each of it by its centroid, the problem is simplified to find the minimum quadratic distance between the objects and centroids. The clusters on an image are organized by the color, it is the quantization vector that provides information about the attributes on the image, so the clusters are going to provide specific information about color and that's why this algorithm is sensitive to outliers, the different images of database have pixels that damage the cluster because those are very far away from other data points, in figure 2 it can appreciate the same colors in different sectors of the image.

- Unlike the K-means algorithm, the hierarchical clustering produces a richer structure than a fixed partition because it is depicted as a tree or dendrogram of fusions between clusters. The algorithm is built as type agglomerative which produces a dendrogram Bottom Up merging the most similar pair of clusters. In figure 4 it can be appreciated how the algorithm cluster the objects according to its similarities, so the perspective is that it is an appropriate algorithm to segment, however as it is observed for the cases of k = 7 and k = 10, some parts of the helicopter as the helix are not demarcated, this is because the segmentation is not complete due to the lack of clusters.

- The number of clusters (k) was the most sensitive parameter, a low number of clusters does not represent all the linked objects in the images, and in the other hand, a very high value generates a over-grouping that does not allow to differentiate objects either. So, is very important to choose a suitable value for a good segmentation.

- Spatial description of images is a powerful way to train an algorithm, as it was shown by Pablo and his collaborators, as well as gradient contour detection using filtering.

References

[1] Capítulo 6 Segmentación de imágenes 6.1 Introducción. Technical report.

[2] Hierarchical Clustering Big Ideas. Technical report.

[3] Tema 4:Segmentación de imágenes. Technical report.

[4] Distances between Clustering, Hierarchical Clustering. Technical report, 2009.

[5] P. Arbelaez. Computer Vision Perceptual Grouping - Segmentation. Technical report, Universidad de os Andes.

[6] P. Arbeláez, M. Maire, C. Fowlkes, and J. Malik. Contour Detection and Hierarchical Image Segmentation. Technical report.

[7] H. A. Aziz, A. Evaluation metrics for medical organ segmentation and lesion detection, 2017.

[8] A. A. O. A. Babakano, J. Performance metrics for image segmentation techniques: A review, 2015.

[9] F. Murtagh and P. Legendre. Ward's Hierarchical Agglomerative Clustering Method: Which Algorithms Implement Ward's Criterion? *Journal of Classification*, 31:274–295, 2014.

[10] P. Roncagliolo and B. Nº. Procesamiento Digital de Imágenes. Technical report.

[11] Scikit-learn. Precision-Recall.

[12] UC Berkeley Computer Vision Group. Contour Detection and Image Segmentation Resources.

[13] Universidad de Oviedo. El algoritmo k-means aplicado a clasificación y procesamiento de imágenes.