Image segmentation using non-supervised classifiers

Mauricio Neira
Universidad de los Andes
Cra 1 Nº 18A - 12, Bogotá - Colombia
m.neira10@uniandes.edu.co

Daniel Rodriguez
Universidad de los Andes
Cra 1 Nº 18A - 12, Bogotá - Colombia
da.rodriguez1253@uniandes.edu.co

Abstract

In this paper we segment images from a small subset of the Berkeley Segmentation Database using 4 clustering algorithms: K Means, Gaussian Mixture Models, Agglomerative Hierarchical Clustering and Watersheds. We study the quality of our segmentation using a new and unrealistic metric. We analyze the segmentation focusing on three aspects: the color space underlined, the unsupervised method, and the number of clusters. The metric implement and the non equivalence of parameters make our results difficult to extend to other studies. However, we give some guidance as to how these evaluation challenges may be overcome in future works on unsupervised segmentation.

1. Introduction

In mathematical terms, an image is simply a matrix whose values we call pixels. However, for humans, an image is a collection of objects. We unconsciously segment visual input to extract information; there is no algorithm. Consequently, this abstract process is difficult to automatize. Several methods have been devised to segment images based on the similarity between pixels and, sometimes, spatial information. These approaches have several limitations since objects can be composed of pixels with very different values, or less commonly, similar pixels can represent different objects.

In this paper we segment 200 images of the BSDS dataset using 4 different unsupervised learning methods. We study the quality of segmentation on the color spaces RGB, HSV, LAB, RGB+xy, HSV+xy and LAB+xy. The nonsupervised methods used were: Kmeans, GMM, Hierarchical and Watersheds. The authors build an oversimplified metric to rapidly evaluate the segmentation. A description of the nonsupervised methods employed are given they are compared terms of the spaces employed and the tuning parameters of the methods. Finally the authors discuss the main computational costs of our algorithm and the changes required to improve the accuracy of their classifiers.

2. Images used

The images were taken from a subset of the Berkeley segmentation data set (BSDS) [1] which can be obtained in the following link: `http://157.253.196.67/BSDS_small.zip`. Each image in the data set is of variable length and width and has 5 related annotations. The data set is partitioned into training and evaluation and the evaluation set contains 28 images. The metrics calculated throughout this paper are based on those 28 images.

3. Approach

Given that each image has 5 human segmentations, we aim to compare the segmentation clustering algorithm to one of the groundtruths. In particular, the first of them.

Segmentation parameter tuning and Image preprocessing.

3.1. Color spaces

The first step in the process was to convert the image from its RGB representation into one of the following color spaces.

3.1.1 RGB

The image was left untouched.

3.1.2 LAB

The image was transformed into the LAB color space where L represents the black and white channel, A the channel that determines the red and green quantities of the pixel and B the channel that quantifies the amount of blue and yellow in the image.

3.1.3 HSV

The image was transformed into the HSV color space where V (value) represents the brightness of the pixel, S (saturation) the purity of the color and H (hue) the color of the pixel.

3.1.4 RGB+xy

This spaie cs the same as the RGB color space with the addition of 2 spatial dimentions correspoding to the indices of the pixels.

3.1.5 LAB+xy

This space is the same as the LAB color space with the addition of 2 spatial dimentions correspoding to the indices of the pixels.

3.1.6 HSV+xy

This spice as the same as the HSV color space with the addition of 2 spatial dimentions correspoding to the indices of the pixels.

3.2. Unsupervised methods

After mapping the image to a color space, the pixels in the new colors pace were clustered together under a parameter k using one of the following algorithms.

3.2.1 K-means

Clusters the data points into k spherical concentrations. This is done by calculating the center of mass of the points that are nearest to each centroid.

3.2.2 GMM

This cl suerint glaorighmtis analogous to k-means except that it adjusts k n-dimensional gaussians to the datapoints.

3.2.3 Hierarchical

The hierarchical method implemented is known in the literature as agglomerative clustering. It is a clustering algorithm that takes a bottom up approach, combining every pair of closest pixels in a color space in each step until the entire image forms a single group. This construction produces a dendogram that can be cut at any stage of the process to obtain k partitions.

3.2.4 Watershed

This algorithm starts with k markers corresponding to the k most significant local minima of the image. The algorithm then simulates a flood starting from these k markers and determines the point at which the water from the different pools of water will meet. The formed pools correspond to the image segments.

3.3. Metric

A general segmentation problem should be compared against average human segmentations or groundtruths. Multiple groundtruths generated by humans make the segmentation problem difficult to asses. However, human variability is mainly due to the number of objects segmented rather than the pixels considered to contain edges or object contours.

We choose an oversimplified metric to evaluate the segmentation quality. Let N denote the number of classes in the groundtruth and M the number of segments of the automatic segmentation. First, an $N \times M$ map matrix $\mathbb{M}$ is built where $\mathbb{M}[i][j]$ contains the number of pixels in the original image with value i that were mapped to the predicted segmentation with value j.

Then for each row in $\mathbb{M}$, the maximum value is extracted and added to a counter. Finally, the counter is divided by N. Let ϕ denote our metric. It would me expressed as:

$$\phi = \frac{1}{N} \sum_{i=1}^{N} max(\mathbb{M}[i]) \tag{1}$$

3.4. Implementation Details

Since color spaces have different range of values and our clustering algorithms are based on the euclidean distance, we normalize all of the channels in a range from [0 to 255] conferring equal weight to them. We do not alter the images in any way. It is common to reduce images resolution, rescale size and apply blurring filters so that big edges (that usually correspond to object divisions) are enhanced. Considering this is a first approach, these steps were ignored for the moment.

Since there is no k *a priori* that would maximize the chosen metric, a range of values of k were considered. In our exploration, $k \in \{5, 10, 15, 20\}$.

Due to performance issues, the agglomerative hierarchical clustering method required the implementation of a connectivity matrix that is calculated from a grey-scale image. This grey-scale image is calculated by averaging the values across all of the dimensions of the color space for each pixel.

Since the watersheds implementation requires a grey-scale image as input as well, the same method for constructing the grey-scale image was used.

4. Experiments and discussiooon

In total, there are 3 parameters that can be tweaked to obtain a segmentation. The clustering algorithm, k and the colorspace. That results in a total of $|methods| \times |k| \times |colorspaces| = 4 \times 4 \times 6 = 96$ combinations. Each one was evaluated over the test set. ϕ was calculated for each image and the average value of ϕ was given as the metric for the database.

In Figure 1, we compare the quality of segmentation of the 4 non supervised methods with our metric. As a general rule, the quality of segmentation linearly decreased as the number of cluster increases. This counter-intuitive fact shows that our metric is ill-defined. When the predicted segmentation produces large regions, multiple ground truth segments will be mapped almost completely to large regions. This translates to a high ϕ. Since small k produce large regions, the segmentation results appear to work better for small k.

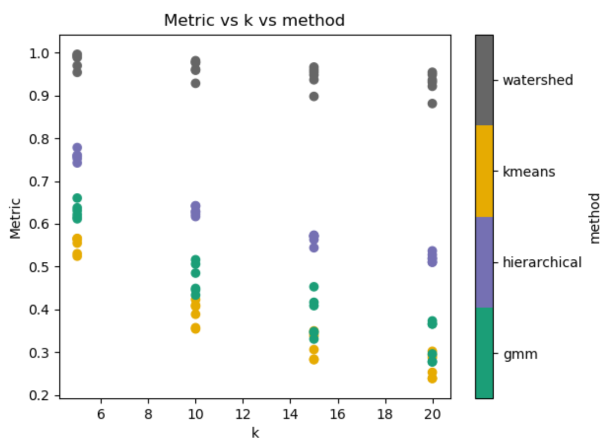

Figure 1. Metric (ϕ) as a function of k. The different algorithms are painted with different colors.

Additionally, there is an evident trend between the performance of the clustering algorithms. It is evident that the order of performance is:

1. Watershed
2. Hierarchical
3. GMM
4. k-means

This makes sense if one realises that $\phi \propto \mathit{predicted\ segment\ size}$. Watersheds tends to cluster tiny islands and leave a large background. Hierarchical manages to cluster images into a set of closed segments that are large relative to GMM and k-means. GMM and k-means both cluster with a really high level of granularity. In the resulting image, a human can make an analogy with sand when seeing the segmentations.

In Figure 2, we analyze ϕ as a function of k and paint the colorspaces to see if there is an underlying pattern. Visually, it is evident that the colorspace does not determine the performance of the algorithms given a k.

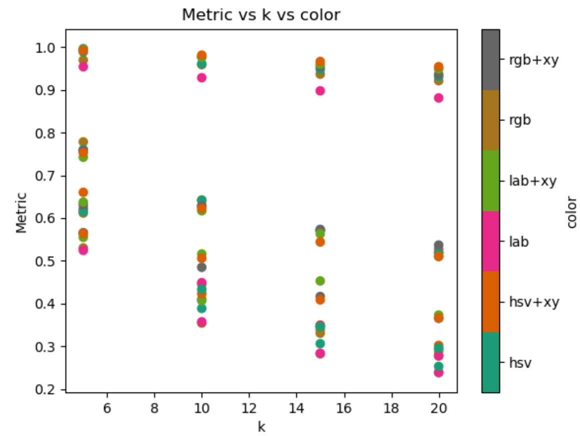

Figure 2. Metric (ϕ) as a function of k. The different color spaces are painted with different colors.

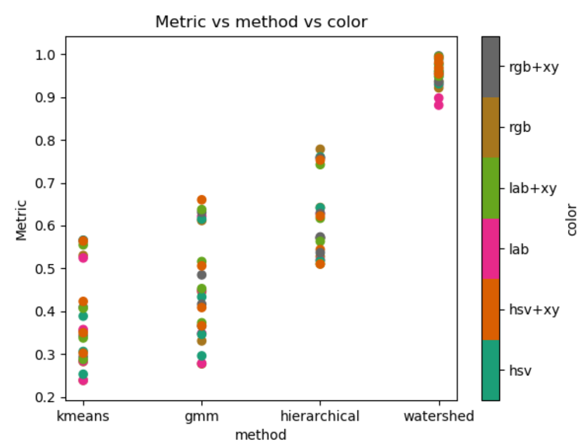

Figure 3. Metric (ϕ) as a function of the clustering method. The different color spaces are painted with different colors.

Lastly, Figure 3 reinforces two previously stated arguments. Firstly, the order of performance of the clustering methods is the same as that of Figure 1. Secondly, it emphasizes the fact that the color space has little to no effect on the performance of the clustering algorithm.

5. Conclusions

First of all, it is worth noting that the evaluation metric ϕ is largely skewed towards results that return large segments as large portions of the labeled segments will be mapped to the same predicted segment and this is favored by the proposed metric. Thus the proposed metric ϕ is not an adequate metric to evaluate image segmentation under a supervised context.

Consequently, since $\phi \propto predicted\ segment\ size$, the parameters that return lare segment sizes are favoured by ϕ. This results in low values of k being favored and the following order for clustering algorithms:

1. Watershed
2. Hierarchical
3. GMM
4. k-means

Finally, the color space has little to no effect in the performance of the algorithms under ϕ. This is due to the fact that the color space does not affect the size of the prediced segments in any significant way.

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.