Textons

Federico Ulloa
Universidad De Los Andes
Cra. 1 18a-12, Bogotá, Colombia
f.ulloa10@uniandes.edu.co

Esteban Vargas
Universidad De Los Andes
Cra. 1 18a-12, Bogotá, Colombia
e.vargas11@uniandes.edu.co

Abstract

In this laboratory, image classification was made using textons. Textons are texture descriptors, the basic units of the patterns that compose a texture. Database from ponce group was used for this work, being divided into training and test sets. Firstly, a texton dictionary was created by convoluting all the filters in the filter bank to crops of the training images, then the responses were clustered using k-means. These texton dictionary obtained, was then used to train a nearest neighbor classifier and a random forest classifier. Lastly, test images were classified using these trained classifiers.

1.. Introduction

Texture is many times an important feature for image processing. It can be defined as repetitive patterns in a region of the image. The basic unit of one of these patterns is called a texton, and was defined by Julesz in 1981.[1]

A texton map of an image is a representation of the different textures of an image given a texton dictionary. Basically, pixels of an image are filtered by the filters in the filter bank and have a different response to each. With these responses, K-means is used to cluster pixels with similar responses into texture categories, obtaining the texton dictionary. This is used to train a classifier such as nearest neighbor or random forests to classify other images.

The texton representation of an image tells us how is the image composed with respect to texture. In other words, a texton map of an image shows how the different regions of the image responded to the filter bank, giving us an idea of which textures appear (or are closer to) in each part of the image.

Clearly, some filters from the filter bank are more discriminative than others, as from each texture, a different response is obtained to each filter. As there are some elements that are more common among the patterns in textures, these will tend to have a higher chance of a high response in general.

2.. Methodology

2.1.. Database

The database used in this laboratory is from the ponce group. [2] This database consists of 25 different classes of images with 20 images each. Each class contains images with the same kind of texture, and each image is composed of a single texture. All the images are gray scale. The dataset was separated into train and test sets.

2.2.. Training

Firstly, as this process is very expensive computationally and there is a large amount of images relative to this, images were re-sized to 50x50, 100x100 and 200x200. This con be done because we know that each image is composed of a single texture. Then, each re-sized image from the training set was convoluted by every filter in the filer bank provided and the response of each pixel to each filter were used to build the texton dictionary. For this, responses were clustered using k-means with a k of 25, 40 and 55. These numbers were selected because they are not too big, and they work well for 25 classes of single texture images.

Then, with the texton dictionary obtained, two classifiers were trained and evaluated: nearest neighbor and random forest. For both classifiers, distance between histograms were used to train the classifiers.

2.3.. Nearest neighbors

For the nearest neighbors classifier, different experiments were done. First, samples of 50, 100 and 200 pixels of each images were selected with a k of 40 to determine the best performance. Then, for the k amount of nearest neighbors was varied to determine the best which gave the best results.

2.4.. Random forests

For the random forests, different amount of trees were tried and different depths of the trees were run as well.

3.. Results

3.1.. Nearest neighbors

After running experiments with different amount of pixels, the best results were obtained with 100 pixels,as shown in the figures. Apart from the ACA, computation time was also a determining factor in choosing the optimal number of pixels. With 50 pixels and k=40, the algorithm took 315 seconds, 996 seconds for 100 pixels, and 3853 seconds for 200 pixels. Results were not significantly better for 200 pixels compared to 100 pixels and it took a lot more time, reason why we chose 100 pixels as the optimal number of pixels.

For the variation of k, as seen in the figures, the ACA decreases as the k increases, because with a big k, the classifier starts to overfit the data. As seen in figure 4, the best ACA obtained with the optimal number of pixels is near to 0.725 with a very little k.

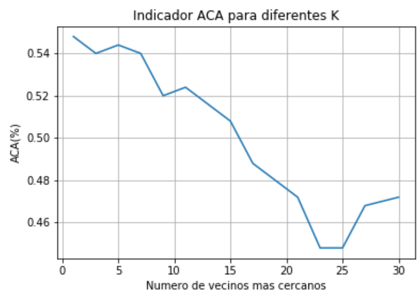

Figura 1: ACA for KNN with 50x50 pixels

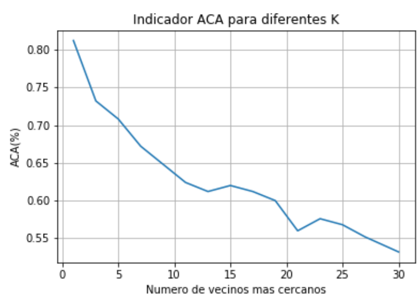

Figura 2: ACA for KNN with 200x200 pixels

3.2.. Random forests

For the random forests, as it can be seen in the figures, the ACA increases with the depth of the trees, until it reaches and optimum and then the classifier also starts to overfit the data. As demonstrated in figure 9, an ACA of near 0.75 was reached for a depth between 10 and 15.

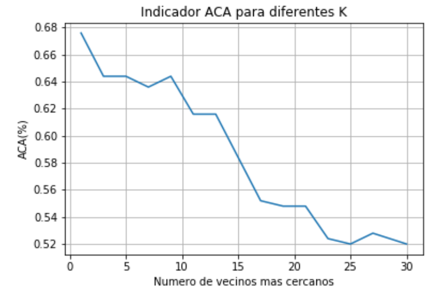

Figura 3: ACA for KNN with 100x100 pixels and k=25

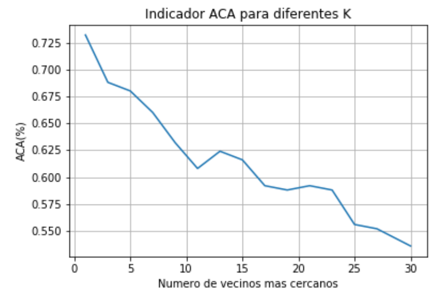

Figura 4: ACA for KNN with 100x100 pixels and k=40

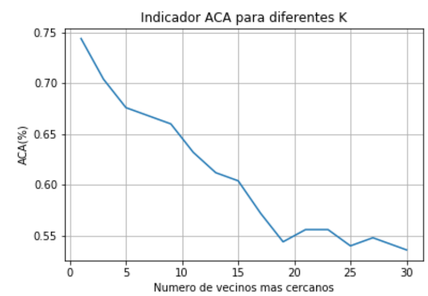

Figura 5: ACA for KNN with 100x100 pixels and k=55

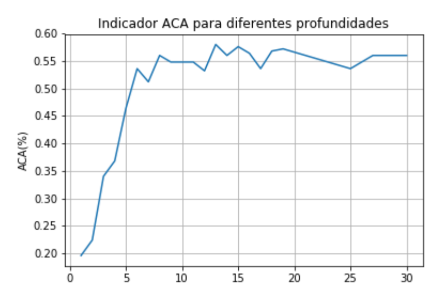

Figura 6: ACA for RF with 50x50 pixels and k=40

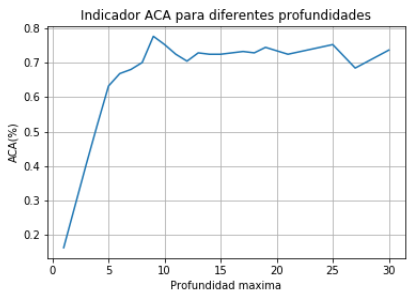

Figura 7: ACA for RF with 200x200 pixels and k=40

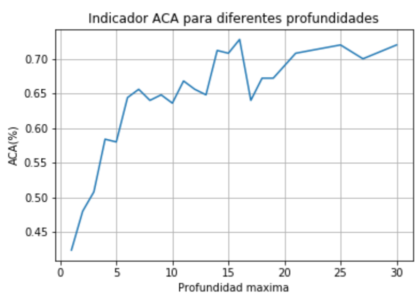

Figura 8: ACA for RF with 100x100 pixels and k=25

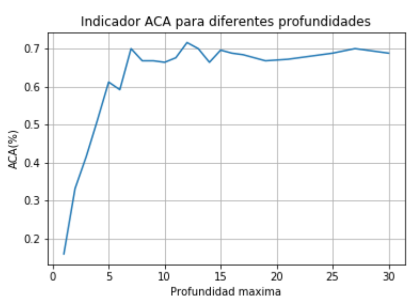

Figura 9: ACA for RF with 100x100 pixels and k=40

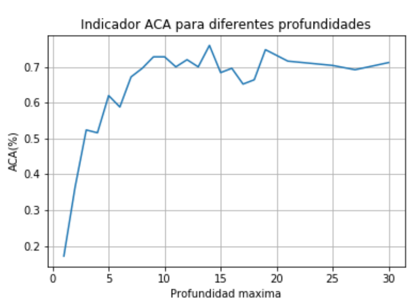

Figura 10: ACA for RF with 100x100 pixels and k=5

4.. Conclusions

Texton representation of images is a good way to do classification in various applications, but it also tends to be very expensive computationally. For this reason, it is very important to find a way to sub-sample the images. In this case, images were re-sized to reduce the computation time.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
8	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
0	10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	9	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	7	0	0	1	0	0	0	0	0	0	0	0	0	0	0	2	0	0	0	0	0	0
0	0	0	0	6	0	0	0	0	0	0	1	0	0	0	1	0	0	0	0	0	2	0	0	0
0	0	0	0	0	5	0	3	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0
0	0	0	2	0	0	6	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1
0	0	0	0	0	0	0	3	1	0	0	0	0	0	0	0	2	0	0	1	0	2	0	0	1
0	0	0	0	0	0	0	0	8	0	0	1	0	0	0	1	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	1	6	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
0	0	0	0	0	0	0	0	0	0	9	1	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	1	0	0	9	0	0	0	0	0	0	0	0	0	0	0	0
0	0	1	0	0	0	0	0	1	1	0	0	0	5	0	2	0	0	0	0	0	0	0	0	0
0	0	0	0	1	0	0	0	0	0	0	1	0	2	5	1	0	0	0	0	0	0	0	0	0
0	0	0	0	1	0	0	0	2	0	0	0	0	0	0	7	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	2	0	0	0	0	1	0	0	0	6	1	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	1	0	0	3	1	1	1	0	3	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0	0
1	0	0	0	0	0	0	1	0	0	0	0	1	0	0	0	0	0	0	5	0	2	0	0	0
0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	9	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	2	0	0	0	1	0	0	0	7	0	0	0
2	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	2	0	0	0	0	4	0	0
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	9	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	10

Figura 11: Confusion matrix for random forests

For the two classifiers used, it is also important to choose the parameters correctly to obtain the best results with an optimal time. Overall, random forests worked better than nearest neighbors. This is because random forests varies the amount of characteristics evaluated in each tree randomly, while with nearest neighbors, the number of characteristics is not random. Also, nearest neighbors overfit the data faster than random forests.

Creating the texton dictionary takes a very long time because for this, each training image has to be convoluted by every filter in the filter bank, and all the responses are clustered. All this takes a big amount of computation.

The method could be improved by sub-sampling the images in another way and comparing other classifiers, like support vector machine, for instance.

Referencias

[1] S. Zhu, C. Guo, Y. Wang and Z. Xu, "What are Textons?", International Journal of Computer Vision, vol. 62, no. 12, pp. 121-143, 2005.

[2] "Ponce Research Group: Datasets", 2018. [Online]. Available: http://www-cvr.ai.uiuc.edu/ponce$_{g}rp/data/.[Accessed : 26-Feb-2018]$.