Textons

Cristina Isabel González Osorio
ci.gonzalez10@uniandes.edu.co

Raúl Santiago Molina Rodríguez
rs.molina10@uniandes.edu.co

Abstract

1. Introduction

The spatial, color and intensity information is important for the development of resolutive algorithms for problems of image classification and object segmentation. However, a space of representation limited to these characteristics does not provide enough information to obtain the desired results. In this sense, the representation of the texture in an image or object is relevant and can be obtained through local patterns in it [1].Thus, it is intended to implement a method of classifying images from the texture information represented by textons.

2. Materials and methods

2.1. Dataset

Texton analysis was made using the *Texture Database* from the Ponce Research Group [2]. This database contains 40 samples of 25 different texture classes, each sample is a grayscale JPEG image of 640x480 pixels. Two different subsets of this database were taken for the classifier's training:

- Class provided set, composed of 750 training images (30 images per class) and 250 for testing (10 images per class)
- Random subset, 20 images were randomly chosen for each class, equally split between training (250 total) and testing (250 total).

2.2. Texture based image classification

Texture in images can be characterized using 'patch' information from across the image, convolving different size, orientation and contrast filters, multiple activation maps can be obtained. A multidimensional representation of the image, with as many channels as filters used, contains texture information pixelwise, each pixel being a datapoint in high dimensional space. Common texture patterns in the image (texton dictionary) are then established through clustering methods (K-means algorithm was used), so that each pixel can be classified to a texton. 128 different textons were used; although there are only 25 textures in the dataset, each texture can be composed of different textons (as might be the case for the same texture in different orientations).

The texton representation of the image groups each pixel to a corresponding texture in the dictionary, images with the same dominating textures are going to have similar distributions in their histograms. To find the appropriate textons, the dictionary must be created with a representative sample of the dataset (training samples), as described in the previous section. Image preprocessing was made to optimize the texton dictionary computing times: for each image in the training set, a 64x64 section was cropped, convolved with the filter bank and then, its activation map, concatenated in a single matrix containing the activation maps for previous samples.

The convolution of each filter can be interpreted as the similarity between an image section and the filter, so images that contain 'oriented' textures can produce high activation of same oriented filters. The same principle applies to blob filters or any arbitrary shaped filter, for this reason is important to apply different scaled filters to the same section, so high frequency and low frequency activations can be obtained, distinguishing pattern features from image edges.

2.3. Classifiers

On one hand, when using the K-Nearest Neighbors method as a classifier, it is possible to specify the number of neighbors to be taken into account, the weight of the same in the classification, and the distance metrics used. For this, the kernel of the intersection as distance metric was used and the number of neighbors with uniforms weights was varied. On the other hand, although the random forest method as a classifier has a large number of hyperparameters, the evaluation of all of them may require a lot of experimentation time, so it is proposed to evaluate the parameters that have a greater influence on the accuracy of the predictions and generalization of the method. Among these are the number of trees, the depth, randomness of the node and the percentage of characteristics that are used by each trained tree.

3. Results

For the classification of the images by KNN, the results reported in the figure 1 were obtained when using the kernel of the intersection and varying the number of neighbors that are taken into account

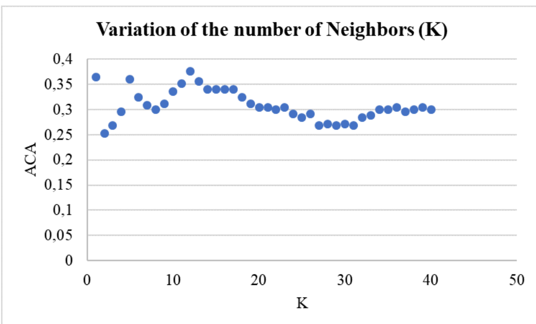

Figure 1: ACA for classification of images using the KNN method with different values of k

According to the above, when using 12 neighbors, a maximum ACA of 0.376 is obtained. Additionally, when using the random forest method with 700 trees and a maximum depth of 6, an ACA of 0.43 was obtained. Finally, when using the database reduced to 10 images per category, ACA of 0.276 and 0.459 were obtained for the KNN and random forest methods, respectively.

4. Discussion of results

the classification in K classes of the response of each pixel in the selected window. In addition, this depends on the number of images in the training database. However, the training and application of both classifiers, once the representation spaces have been extracted, does not have a considerable temporal complexity.

According to the reported results, it is possible to affirm that the random forest method has a performance as a classifier significantly higher with respect to the K-Nearest Neighbors method because a greater ACA is obtained. This can be understood according to the difference in complexity of these methods. In this sense, the simplicity of the KNN method is evidenced by the decrease in the accuracy of the predictions. Although, the random forest method is more complex, this can be counterproductive by overfitting the classifier to the training data. The above can be solved with the variation of the hyperparameters mentioned above until finding those that produce the smallest difference between the test error and the training error.

Finally, regarding the limitations of the method, it should be mentioned that for the creation of a texton dictionary sufficiently general and complete it is necessary to use a large number of pixels. In addition, the classification of all this information through Kmeans is delayed. Consequently, the temporal and spatial complexity of the algorithm is high. With respect to the classification methods used, it should be considered that their results depend on different parameters and the metric used. In this way, the process in the estimation of these parameters requires a great time of experimentation in order that the precision in the predictions does not diminish due to phenomena of overfitting or underfitting of the classifier.

References

[1] R. M. Haralick, K. Shanmugam, et al. Textural features for image classification. *IEEE Transactions on systems, man, and cybernetics*, (6):610–621, 1973.

[2] S. Lazebnik, C. Schmid, and J. Ponce. A sparse texture representation using local affine regions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(8):1265–1278, 2005.