Re-Implementation of "Modeling the shape of the scene: a holistic representation of the spatial envelope"

Mehdi Mirza-Mohammadi

Dept. Matemàtica Aplicada i Anàlisi, Universitat de Barcelona, Gran Via 585, 08007, Barcelona, Spain.
e-mail:me@memimo.net

1 Introduction

The current project is the re-implementation of [1] in python for scene recognition . Matlab implementation provided by the authors [1] was the main source, and this module is implemented in the same way. The dataset used for testing is also the same dataset as the one authors use.

2 Technical Details

The libraries used inside the module are all non-commercial, open-source libraries:

- SciPy-NumPy http://www.scipy.org/
- matplotlib http://matplotlib.sourceforge.net/
- Shogun http://www.shogun-toolbox.org/

The module has been written and tested on Ubuntu 9.04 64bit system.

[1] http://people.csail.mit.edu/torralba/code/spatialenvelope/

3 Results

3.0.1 Feature Computation

The feature computation section's result, which is the main novelty of the work, has been compared with original Matlab results, and they are exactly the same for same images.

For comparison of the features in the two implementation, a Matlab function $Gist\_Feature.m$ has been implemented, which compute features with different parameters for the given images and save the results along with parameters in output files. The python module *compare.py* read those parameters and compute the features with those parameters, then compare the feature vector with the corresponded one from Matlab.

The norm of difference between vectors are from order of 10^-7 which is the result of difference between data type definition and manipulation in two programming languages.

3.0.2 Classification

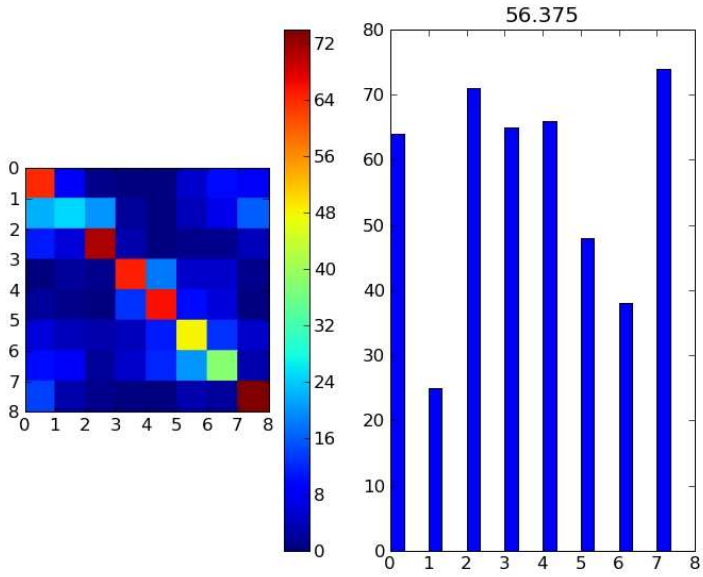
The original Matlab code use a different Super Vector Machine toolbox than shogun, and the kernel that is used is RBF. In the re-implementation shogun toolbox has been used with Gaussian kernel. For better evaluation of the classification result it would be better, to apply 10-fold cross validation, but here the same method as the author's has been used, and the performance results from both implementations has been shown in fig. 1

As you can see, both have relatively same results. The mean performance in python version is 56.375, and in Matlab version is 54.196, although maximum performance is higher in Matlab version. And we can see that the best performance is for "street", "opencountry" and "forest" categories, and lowest for "insidecity" and "mountain" categories.

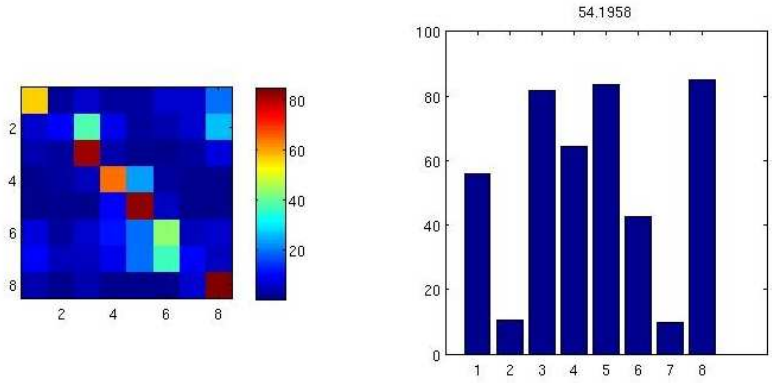
The execution time for 2688 images which was included in the dadset is shown in Table. 1

Table 1
Execution time on a Intel© Core2Duo 2000GHz laptop machine

	Python Code	Matlab Code
Feature Computation	7 minutes and 49.98 seconds	4 minutes and 46.73 seconds
Classification	22.11 seconds	4 minutes and 30.06 seconds
Total	8 minutes and 12.09 seconds	9 minutes and 16.79 seconds



(a)



(b)

Fig. 1. On right confusion matrix and on left, performance for each category: (a) Python implementation, (b) Matlab impelentation

References

[1] A. Oliva, A. Torralba, Modeling the shape of the scene: A holistic representation of the spatial envelope, Int. J. Comput. Vision 42 (3) (2001) 145–175.