Diversity in ensembles of codebooks for visual concept detection

Luca Piras, Roberto Tronci, and Giorgio Giacinto

Department of Electrical and Electronic Engineering, University of Cagliari, Italy
{luca.piras,roberto.tronci,giacinto}@diee.unica.it

Abstract. Visual codebooks generated by the quantization of local descriptors allows building effective feature vectors for image archives. Codebooks are usually constructed by clustering a subset of image descriptors from a set of training images. In this paper we investigate the effect of the combination of an ensemble of different codebooks, each codebook being created by using different pseudo-random techniques for subsampling the set of local descriptors. Despite the claims in the literature on the gain attained by combining different codebook representations, reported results on different visual detection tasks show that the diversity is quite small, thus allowing for modest improvement in performance w.r.t. the standard random subsampling procedure, and calling for further investigation on the use of ensemble approaches in this context.

Keywords: bag of words, clustering, SVM

1 Introduction

As far as an increasing amount of information is available in digital form, smart tools are needed to search into multimedia collections by semantic content. The automatic extraction of semantic content from images can still be a problem in an unconstrained scenario, while, for a number of tasks, some effective techniques are currently available. Due to the difficulty of capturing the complex semantics of images, a number of highly accurate low-level representation of images (e.g., color features, texture, shapes, local descriptors, etc.) have been proposed.

For example, if the goal is to classify a photo as depicting one out of a set of specific scenes, or as containing a certain object of interest, a representation based on a collection of numerical global feature vectors can be used. In the field of visual concept detection, many global features have been proposed to describe image contents such as color [3, 8], and texture [4] histograms. These approaches work very well as long as the concept the user is interested in can be captured by global features. Indeed, global features are not useful to distinguish different parts of the images, e.g., the foreground from the background, and thus information from different parts can be mixed together. On the other hand, descriptors based on local features, i.e., image patterns that differs from the immediate neighborhood for changes in intensity, color, or texture, allows

recognizing objects or scenes in an image, but lose the capability of capturing the global concept as a whole. A way to make a trade-off between global/local features is the use of the so-called bag-of-features methods that have extensively exhibited high levels of performance [7, 23].

These kind of methods have been proposed in the literature under different names such as 'textons' [13], 'object parts' [9] and 'codebooks' [10], but the main idea is more or less the same: the use of vector quantization techniques on local descriptors extracted in some particular areas of an image. The quantization is performed according to a visual codebook where each descriptor is assigned to the codebook element which is closest in the feature space.

Codebooks are usually constructed by clustering local descriptors from a set of training images by using standard clustering algorithms such as the k-means. Clustering is usually performed by sampling either densely or sparsely the set of local descriptors. The final result is the set of centroids of the clusters, so that an image is represented by a vector where the i-th component represents the number of local descriptors that falls in the i-th cluster. The algorithm for generating the codebook is composed of three main parts: the detection of interest points, the extraction of effective descriptors, and the clustering algorithm.

The extraction of interest points can be carried out according to a number of approaches that have been proposed in the past years [15, 10, 12, 27]. The easiest approach that allows extracting a large number of interest points is based on exhaustively sampling different sub-parts of the image at each location and scale [6]. However, by far the more severe drawback of this approach lies in its high computational cost. In order to overcome this problem, a fixed grid of image patches has been proposed in [14] in the context of scene classification, where sampling is performed at every n-th pixel, and at fixed multiple scales. Other approaches, instead of focussing on a more or less dense exhaustive pixels sampling, focus on the search of a set of few but informative interest points. In [27] the authors propose a distinction of corner detectors, blob detectors, and region detectors. Among all, the Harris-Laplace corner detector [17], and the Hessian-Laplace blob detector [18] have proved successful in many applications.

Intensity-based descriptors have been widely used so far to extract distinctive invariant features from interest points. The Scale Invariant Feature Transform (SIFT) descriptor proposed in [15] is one of the most used approaches. It describes the patches of an image by edge orientation histograms. This descriptor is not invariant to changes in light color, because the intensity channel is a combination of the R, G and B channels. In order to add color invariance, and increase its discriminative power, several color descriptors have been proposed [21].

Clustering is the last step of the codebook generation, and it is the step this paper is focused in. In fact, as above mentioned, a large number of works investigated different ways of detecting interest points, and extracting effective descriptors, but very few works investigated how to combine different codebooks by manipulating the clustering process [16].

Clustering is usually performed by taking into account the interest points of all the images in the training set. The clustering process aims at detecting the so-

called *visual words*, i.e., the basic low-level building blocks that allows detecting the concepts of interests. The underlying assumption is that each concept can be detected by representing the images in terms of the number of interest points falling in each cluster. It is easy to see that the clustering process depends on the way interest points are detected, and on the employed descriptor.

Usually, from a dataset of a few thousands of images, it is possible to extract from a few millions to several dozens of million of interest points depending on the employed detector. To reduce the computational cost of the clustering process, the set of interest points is usually under-sampled up to a few hundreds of thousand of elements. However, this means that, on average, we are actually considering less than ten points for each image in datasets containing some dozens of thousands images. On the other hand, a larger number of interest point can be taken into account by resorting to the ensemble paradigm. Different codebooks can be created by using different random or pseudo-random under-samplings of the set of interest points so that the overall number of points used to train the system is as large as possible. Then, a set of classifiers can be trained using the set of different codebook representations and their results combined.

In this paper, we aim at investigating the diversity of the codebooks that can be created according to this approach, and the related performance. Previous works showed that some improvement can be attained by using different clustering algorithms or by employing codebooks of different sizes [16]. Reported results showed that ensemble members where characterized by some kind of diversity, but the overall performance improvements where often modest with respect to other state of the art methods. The approaches investigated in this work are more easy to implement as they don't require different clustering algorithms, and the size of the codebook is kept constant. Experiments have been performed on two distinct visual concept detection tasks. Reported results show that some modest improvements can be attained, even if the correlation of outputs of the ensemble members is quite large. However, as the base performance of concept classifiers are not very high, even modest improvements provide a useful gain in real scenarios.

This paper is organized as follows: Sect.2 presents the proposed pseudo-random under-sampling approach; Sect.3 presents the three different combination rules used in order to combine the score gained from the different classifiers; Experimental results are reported in Sect.4 and conclusions are drawn in Sect.5.

2 Random selection of interest points for visual feature extraction

We focus this section on the clustering process. The clustering process is the key step that transforms the set of interest point descriptors of a set of training images into "bags of visual words" and codebooks.

The process of creating a codebook is quite simple. First, m interest points are extracted from all the training images by resorting to one of the techniques

revised in Section 1. Then, these points are clustered by using any of the clustering algorithms available in the literature. Typically, k-means clustering is used, and the number of clusters k is chosen a-priori, where $k << m$. For each cluster, the centroid is computed, so that each image can be mapped into a codebook of size k by assigning the interest points of the image to the nearest centroid. Finally, this new feature vector representation can be used for image classification.

However, some *tricky* settings are needed for the clustering process to be effective such as the estimation of the "optimal" number of clusters (i.e., codebook's dimension) in the case of the k-means clustering algorithm, and the number of interest points to be clustered. The "optimal" codebook's dimension is usually empirically determined by testing the performance of different codebook sizes.

A common procedure to set the total number of interest points to be clustered is based on a random extraction of a fixed number of points from the totality of points belonging to all images in the training set. This number is fixed regardless of the number of images in the dataset in order to limit the computational complexity of the clustering algorithm. Consequently, the number of points per image automatically decreases as long as the number of images increases. For example, the *Color Descriptors* toolkit [21] (i.e., one of the most popular toolkits available to generate codebooks) employs the k-means clustering algorithm to produce a "bag of visual words" representation by randomly selecting 250,000 points from the set of points of the images of the input dataset, regardless of the dimension of the dataset.

In this paper we investigate how the number of points selected for clustering affects the performance of the concept detector. In particular, we investigate if a larger number of points allows extracting a larger amount of information from the training set. However, in order to limit the computational complexity of the clustering phase, we use different codebooks of the same size generated by selecting different subsets of points from the totality of points of the training set. Then, the detectors trained using these different codebooks are combined in order to exploit the information embedded in a larger number of interest points.

It is well known that the performance of the combination of an ensemble of classifiers depends on the "diversity" among the ensemble members, the higher the diversity, the larger the gain in performance that can be expected. To increase the diversity of the ensemble members, we test a pseudo-random procedure to create different codebooks. In particular, we randomly divide the set of training images into n groups, each group containing the same proportion of images per class as the original training set, then a fixed number of interest points is extracted from each group. In this way n different "bags of visual words" representations are obtained, the diversity being attained by using points from distinct set of images from the training set.

3 Combination methods

For each image x_j, n representations are obtained. Let us denote with x_j^k the representation of image x_j in the k representation, let s_{ij}^k and d_{ij}^k be the score,

and the final decision (in the form of a predicted binary label), respectively, produced for x_j^k by a classifier trained for detecting concept i. Without losing generality, let us assume that the score is in the range $[0, 1]$. Then, for each concept i, the scores s_{ij}^k are combined in order to obtain a final score s_{ij}^* for image x_j. We tested the following combination rules:

- The Mean rule
- The Dynamic Score Combination by Majority Vote
- The Dynamic Score Combination by Mean rule

These combination approaches has been chosen for their limited computational cost and because their performance showed to be quite good when compared to more complex combination rules that require a higher computational cost in different application scenarios [25].

For the *Mean rule*, we computed the average of the scores obtained from the set of classifiers [11]:

$$s_{ij}^{mean} = \frac{1}{k} \cdot \sum_k s_{ij}^k \tag{1}$$

In the case of the other two combination rules, we used the Dynamic Score Combination (DSC) approach [25]:

$$s_{ij}^{dsc} = (1 - \alpha) \cdot \min_k \{s_{ij}^k\} + \alpha \cdot \max_k \{s_{ij}^k\} \tag{2}$$

In a two-class problem formulation, the aim of the DSC approach is to combine the scores so that the combined score distributions for the two classes exhibit a larger separation than the score distributions produced by the individual classifiers. Thus, ideally, the maximum separation can be achieved by selecting the maximum score for patterns belonging to the class of interest, and the minimum score otherwise. Equation 2 is a formulation that embeds selection and combination through the use of the weights α. In [25] different methods to compute dynamically the weights α are proposed.

The rule for computing α in the case of *DSC by Majority Vote* is the following:

$$\alpha = \begin{cases} 1, & \text{if at least half of the } d_{ij}^k \text{ are equal to 1} \\ 0, & \text{otherwise} \end{cases} \tag{3}$$

i.e., for each image and class, the maximum score is selected if the majority of classifiers claims that the image belongs to that class, while the minimum score is selected otherwise.

The rule for computing α in the case of *DSC by Mean Rule* is the following:

$$\alpha = \frac{1}{k} \cdot \sum_k s_{ij}^k \tag{4}$$

i.e., the combined score is a combination of the maximum and minimum scores, the value of α being equal to the average of the scores produced by the classifiers.

4 Experimental Results

Experiments have been carried out using two datasets, namely a subset of the MIRFLICKR[1] collection proposed for the ImageCLEF 2012 Photo Flickr Annotation Task [24] and the MICC-Flickr101 dataset [1]. The first dataset comprises 25 thousand multi-labelled images that have been manually annotated using 94 concepts. The MICC-Flickr101 dataset is based on the 101 object categories of the Caltech101 dataset, while the images were obtained by downloading them from Flickr in January 2012. This dataset is made up of 7348 single labelled images with at least about 40 images per class, the median of the number of elements per class being equal to 70. Images are at high resolution, 1024 x 768 pixels on average, and depict objects in daily-life real scenarios. The ImageCLEF 2012 dataset was originally subdivided into a training and a test set, while we randomly subdivided the MICC-Flickr101 dataset.

A dense sampling strategy for interest point detection has been used: features are extracted at 4 scales (0.5, 1, 1.5, and 2) with a regular grid spaced 10 pixels. For extracting the SIFT descriptors, the ISIS Color Descriptors[2] toolkit has been used [21].

In order to create different codebooks, and test the proposed pseudo-random procedure, we randomly divided the training set into 4 groups. Each group contained the same proportion of images per class as the original training set, then 1 million of interest points has been extracted from each group. We compared this approach to the common procedure, i.e. randomly selecting 250,000 points from the set of points extracted from the entire training set. In particular, we performed four random extractions of 250,000 points to create four different codebooks. The dimension of the codebooks, i.e. the number of clusters k in the k-means approach, has been fixed to 512. The choice of the parameters for training set divisions (4) and k-means (512) was carried out to obtain a "good" compromise between the diversity of the codebooks, and the associated computational cost. In particular, we aimed to limit the overall computational cost related to all the phases involved in obtaining the bag-of-visual-words representation, and the later parameter estimation for each SVM that has to be trained for each representation. It is well known that all these phases are highly time consuming. In several preliminary experiments, we also used different values for the k parameter, and the comparison between the proposed mechanism and the usual one exhibits the same behavior as the one reported in the paper.

The Support Vector Machine with RBF kernel has been used as the base classifier for its good performance on various image classification tasks [5, 2]. In particular, we implemented a multi-label classifier by independently training a SVM classifier for each class [22, 26]. SVM parameters have been set by exploring the set of parameters in order to select the detector with the highest performance in the training set. For each classifier, a different decision threshold has been set according to the threshold optimization approach proposed in [19] aimed at

[1] http://press.liacs.nl/mirflickr/
[2] http://koen.me/research/colordescriptors/

maximizing the overall classification performance when, for each pattern, a score value is available for each class. We trained a single SVM for each concept and for each "bags of visual words" representation. Thus, a S_i^k SVM is available for concept i in the k representation.

4.1 Results

Performance are evaluated in terms of Macro-averaged and micro-averaged F-measures [22, 26]. In multi-label classification tasks, the F-measure [20] over all classes can be defined in terms of empirical averages in two different ways. Macro-averaging (denoted with the capital 'M') consists of averaging over all the classes the corresponding class-related measure. It equally weights each class, and thus tends to be dominated by the performance on rare classes, which is usually lower than that attained for common ones [28]. Micro-averaging (denoted with 'u') consists of computing the measure with respect to the sum of the true positive, false positive and false negative values over all classes.

The results of the experiments on the ImageCLEF and the MICC-Flickr101 datasets are reported in the following Tables. Reported results are related to different experimental settings:

- average performance of four concept detectors, each one trained on different codebooks, each created by 250,000 independently random selected points (SE avg);
- combination of the scores of the codebooks generated as in the former experiments by the Mean rule;
- combination of the scores obtained by training four concepts detectors on four different codebooks generated according to the proposed pseudo-random procedure (Mean rule, DSC mean, DSC majority).

Reported results show that the different ways of selecting the points for the creation of the codebooks influences the performance depending on the number of the images in the dataset. Table 1, related to the ImageCLEF dataset, shows that combining the scores of four concept detectors trained on different codebooks (**SE Mean**, **Mean rule**, **DSC mean**, and **DSC majority**) allows obtaining a slight better results w.r.t. the average performance of individual detectors (**SE avg**). In addition, reported results shows that in the case of the ImageCLEF dataset the improvement in performance is larger when ensemble members rely on codebooks generated by four independent random extractions of points (**SE Mean**) than those attained by creating the codebooks by first splitting the training set into four groups (Training set splitting).

Results reported in Table 2 show that the F1u performance of the **Mean rule**, related to the (Training set splitting), is slightly worse than that of the **SE Mean** techniques, related to four random extractions of points. On the other hand, (Training set splitting) exhibits a better performance if the F1M measure is considered. This behavior can be explained by the definition of the two measures. While the F1u measure computes the sum of the true positive, false

	Random		Training set splitting		
	SE avg	**SE Mean**	**Mean rule**	**DSC mean**	**DSC majority**
F1u	49.51%	50.12%	49.90%	49.84%	49.65%
F1M	23.07%	24.04%	23.64%	23.37%	23.41%

Table 1. Results in terms of F1 micro and macro for the ImageCLEF dataset.

	Random		Training set splitting		
	SE avg	**SE Mean**	**Mean rule**	**DSC mean**	**DSC majority**
F1u	28.12%	29.47%	29.36%	28.64%	29.05%
F1M	24.15%	24.96%	25.47%	24.93%	25.10%

Table 2. Results in terms of F1 micro and macro for the MICC-Flickr101 dataset.

positive and false negative values over all classes, and then evaluates a "global" average, the F1M measure averages the corresponding class-related measure over all the classes, with equal weights for each class. Consequently, the F1M tends to be dominated by the performance of the classes containing a tiny fraction of patterns. In other words, this means that when the dataset contains a lot of images, and, consequently, a very large number of interest points are extracted, the (Training set splitting) procedure allows using a larger number of points per image w.r.t. the random extraction of points from the entire training set, but it does not translates into an increased diversity of the generated codebooks. On the other hand, when the number of images in the dataset is not so high, the (Training set splitting) procedure allows extracting those points that permit to create diverse codebooks that can improve the performance on rare classes.

It is worth noting that the performance improvements are related to the correlation (i.e,, the diversity) of codebooks. In fact, reported results show that the performance obtained by using an ensemble of diverse codebooks are always higher than those obtained by using just a single codebook. To this end, Figure 1 shows the degree of correlation among the scores of the 8 concept detectors, four of them trained with the codebooks obtained according to the (Training set splitting) procedure (**Split**x), the other four trained with the codebooks obtained by following the common approach (**Trad**x). It is easy to see that, regardless the technique used to produce different codebooks, the output scores tend to exhibit high correlation values. Nonetheless, some improvements in performance is attained.

5 Conclusions

We investigated the effect of the combination of an ensemble of different codebooks, each codebook being created by using different pseudo-random techniques for sub-sampling the set of local descriptors. In particular, we tested different

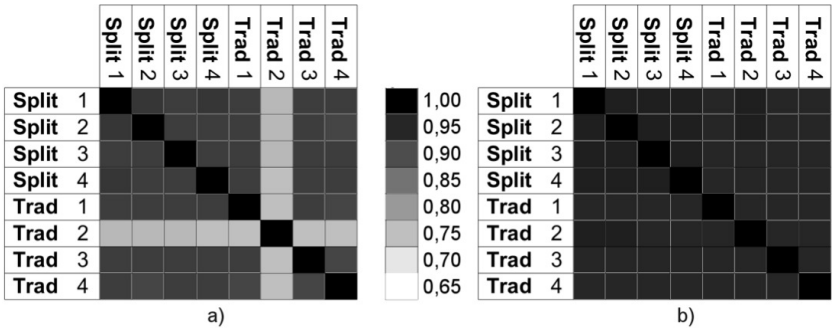

Fig. 1. Correlation matrix between the score obtained using the proposed approach (**Split**x) and the score obtained using the common clustering method (**Trad**x) in the ImageCLEF (**a**) and MICC-Flickr101 (**b**) dataset

combination strategies to combine the scores produced by the detectors trained on such different codebooks. Reported results show that the use of ensemble approaches allows attaining modest improvement in performance even if the correlation of the output score is somewhat high. However, it is worth noting that the proposed approaches allows reducing the computational cost w.r.t. the typical settings. Thus, we can conclude that creating diverse codebooks allows reducing the training cost of the classification system, while retaining or slightly increasing the classification performance.

Acknowledgments

This work is supported by the Regional Administration of Sardinia, Italy, within the project "Advanced and secure sharing of multimedia data over social networks in the future Internet" (CUP F71J11000690002).

References

1. Ballan, L., Bertini, M., Del Bimbo, A., Serain, A.M., Serra, G., Zaccone, B.F.: Combining generative and discriminative models for classifying social images from 101 object categories. In: ICPR 2012. Tsukuba, Japan
2. Chang, C.C., Lin, C.J.: Libsvm: A library for support vector machines. ACM TIST 2(3), 27 (2011)
3. Chang, S.F., Sikora, T., Puri, A.: Overview of the mpeg-7 standard. IEEE Trans. Circuits Syst. Video Techn. pp. 688–695 (2001)
4. Chatzichristofis, S.A., Boutalis, Y.S.: Fcth: Fuzzy color and texture histogram - a low level feature for accurate image retrieval. In: Proc. of the 9th Int. Workshop on Image Analysis for Multimedia Interactive Services. pp. 191–196. IEEE CS (2008)
5. Cristianini, N., Shawe-Taylor, J.: An Introduction to Support Vector Machines and Other Kernel-based Learning Methods. Cambridge University Press (2000)
6. Crowley, J.L., Sanderson, A.C.: Multiple resolution representation and probabilistic matching of 2-d gray-scale shape. IEEE Trans. Pattern Anal. Mach. Intell. 9(1), 113–121 (1987)

7. Csurka, G., Dance, C.R., Fan, L., Willamowski, J., Bray, C.: Visual categorization with bags of keypoints. In: In Workshop on Statistical Learning in Computer Vision, ECCV. pp. 1–22 (2004)
8. Deselaers, T., Keysers, D., Ney, H.: Features for image retrieval: an experimental comparison. Inf. Retr. 11(2), 77–107 (2008)
9. Fergus, R., Perona, P., Zisserman, A.: Object class recognition by unsupervised scale-invariant learning. In: CVPR (2). pp. 264–271. IEEE Computer Society (2003)
10. Jurie, F., Triggs, B.: Creating efficient codebooks for visual recognition. In: ICCV. pp. 604–610. IEEE Computer Society (2005)
11. Kittler, J., Hatef, M., Duin, R.P.W., Matas, J.: On combining classifiers. IEEE Trans. Pattern Anal. Mach. Intell. 20(3), 226–239 (1998)
12. Lazebnik, S., Schmid, C., Ponce, J.: Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In: CVPR (2). pp. 2169–2178. IEEE Computer Society (2006)
13. Leung, T.K., Malik, J.: Representing and recognizing the visual appearance of materials using three-dimensional textons. International Journal of Computer Vision 43(1), 29–44 (2001)
14. Li, F.F., Perona, P.: A bayesian hierarchical model for learning natural scene categories. In: CVPR (2). pp. 524–531. IEEE Computer Society (2005)
15. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. International Journal of Computer Vision 60(2), 91–110 (2004)
16. Luo, H.L., Wei, H., Hu, F.: Improvements in image categorization using codebook ensembles. Image Vision Comput. 29(11), 759–773 (2011)
17. Mikolajczyk, K., Schmid, C.: Indexing based on scale invariant interest points. In: ICCV. pp. 525–531 (2001)
18. Mikolajczyk, K., Schmid, C.: Scale & affine invariant interest point detectors. International Journal of Computer Vision 60(1), 63–86 (2004)
19. Pillai, I., Fumera, G., Roli, F.: Threshold optimisation for multi-label classifiers. Pattern Recognition 46(7), 2055 – 2065 (2013)
20. Van Rijsbergen, C.J.: Information Retrieval. Butterworth (1979)
21. Van de Sande, K.E.A., Gevers, T., Snoek, C.G.M.: Evaluating color descriptors for object and scene recognition. IEEE Trans. Pattern Anal. Mach. Intell. 32(9), 1582–1596 (2010)
22. Sebastiani, F.: Machine learning in automated text categorization. ACM Comput. Surv. 34(1), 1–47 (2002)
23. Sivic, J., Zisserman, A.: A text retrieval approach to object matching in videos. In: ICCV. pp. 1470–1477. IEEE Computer Society (2003)
24. Thomee, B., Popescu, A.: Overview of the imageclef 2012 flickr photo annotation and retrieval task. Tech. rep., CLEF 2012 working notes, Rome, Italy (2012)
25. Tronci, R., Giacinto, G., Roli, F.: Dynamic score combination: A supervised and unsupervised score combination method. Machine Learning and Data Mining in Pattern Recognition 5632, 163–177 (2009)
26. Tsoumakas, G., Katakis, I., Vlahavas, I.P.: Mining multi-label data. In: Maimon, O., Rokach, L. (eds.) Data Mining and Knowledge Discovery Handbook, pp. 667–685. Springer (2010)
27. Tuytelaars, T., Mikolajczyk, K.: Local invariant feature detectors: A survey. Foundations and Trends in Computer Graphics and Vision 3(3), 177–280 (2007)
28. Yang, Y.: A study on thresholding strategies for text categorization. In: ACM (ed.) Proceedings of the International ACM SIGIR Conference on Research and Development in Information Retrieval,. p. 137145 (2001)