Journal of King Saud University – Science (2014) 26, 89–100

King Saud University
Journal of King Saud University – Science
www.ksu.edu.sa
www.sciencedirect.com

ORIGINAL ARTICLE

Ensemble of different local descriptors, codebook generation methods and subwindow configurations for building a reliable computer vision system

Loris Nanni [a,*], Alessandra Lumini [b], Sheryl Brahnam [c]

[a] DEI, University of Padua, viale Gradenigo 6, Padua, Italy
[b] DISI, Università di Bologna, Via Venezia 52, 47521 Cesena, Italy
[c] Computer Information Systems, Missouri State University, 901 S. National, Springfield, MO 65804, USA

Received 24 June 2013; accepted 6 November 2013
Available online 18 November 2013

KEYWORDS
Object recognition;
Bag-of-features;
Texture descriptors;
Machine learning;
Support vector machine;
Usage scenarios

Abstract In the last few years, several ensemble approaches have been proposed for building high performance systems for computer vision. In this paper we propose a system that incorporates several perturbation approaches and descriptors for a generic computer vision system. Some of the approaches we investigate include using different global and bag-of-feature-based descriptors, different clusterings for codebook creations, and different subspace projections for reducing the dimensionality of the descriptors extracted from each region. The basic classifier used in our ensembles is the Support Vector Machine. The ensemble decisions are combined by sum rule. The robustness of our generic system is tested across several domains using popular benchmark datasets in object classification, scene recognition, and building recognition. Of particular interest are tests using the new VOC2012 database where we obtain an average precision of 88.7 (we submitted a simplified version of our system to the person classification-object contest to compare our approach with the true state-of-the-art in 2012). Our experimental section shows that we have succeeded in obtaining our goal of a high performing generic object classification system.

The MATLAB code of our system will be publicly available at http://www.dei.unipd.it/wdyn/?IDsezione=3314&IDgruppo_pass=124&preview=. Our free MATLAB toolbox can be used to verify the results of our system. We also hope that our toolbox will serve as the foundation for further explorations by other researchers in the computer vision field.

* Corresponding author. Tel.: +39 0547 339121; fax: +39 0547 338890.
E-mail addresses: loris.nanni@unibo.it, loris.nanni@unipd.it (L. Nanni).

Peer review under responsibility of King Saud University.

Production and hosting by Elsevier

1. Introduction

Given the vast amount of data being collected machine analysis of image content is imperative (Müller et al., 2004; Lew et al., 2006), a key issue is finding effective feature representations for images. Early systems developed in the 1990s, e.g., Candid (Kelly et al., 1995), Photobook (Pentland et al., 1996), and

1018-3647
http://dx.doi.org/10.1016/j.jksus.2013.11.001

Nextra (Ma et al., 1997), exploited simple global features based on image color, texture, and shape. Approaches around the turn of the century, e.g. (Li et al., 2003; Fergus et al., 2004), focused on constellation models to locate distinctive object parts and to determine constraints on the spatial arrangement. The main drawback of these representations is that they typically are unable to handle significant deformations such as large rotations and occlusions. Moreover, they fail to consider objects, such as trees and buildings, with variable numbers of parts.

More recent systems have taken advantage of new developments in the application of local descriptors in pattern recognition, computer vision, and image retrieval. Of particular importance has been the use of such local features as keypoints and image patches, which have shown great promise in several application areas, including wide baseline matching for stereo pairs (Baumberg, 2000; Tuytelaars and Gool, 2004), object retrieval in videos (Sivic et al., 2004), object recognition (Lowe, 2004), texture recognition (Lazebnik et al., 2005), robot localization (Se et al., 2002), visual data mining (Sivic and Zisserman, 2004), and symmetry detection (Turina et al., 2001). A consensus has emerged from that literature supporting the value of the bag-of-words (BoW) technique for image representation (Lowe, 2004). BoW is based on powerful scale-invariant feature descriptors that are used to match identical regions between images by representing regions in a given image that are covariant to a class of transformations.

Region matching using local image features handles illumination changes, blurring, zoom effects, and many degrees of occlusion and of distortions in perspective. Approaches for region description have been proposed that analyze different aspects of images, such as color, texture, edges, and pixel intensities. Some of the most promising descriptors are those based on histogram distributions (Mikolajczyk and Schmid, 2005). Some important examples of these descriptors include the intensity-domain spin image (Lazebnik et al., 2006), an histogram approach that represents regions using the distance from the center point and intensity values; the SIFT descriptor (Lowe, 2004), an histogram that takes the weighed gradient locations and orientations; and the geodesic intensity histogram (Ling and Jacobs, 2005), a histogram that provides a deformation invariant local descriptor. Other descriptors of this type include PCA-SIFT (Ke and Sukthankar, 2004), moment invariants (Gool et al., 1996), and complex filters (Schaffalitzky and Zisserman, 2002). Some powerful texture descriptors include center-symmetric local binary patterns (CS-LBP) Heikkilä et al., 2009, a LBP-based texture descriptor which is computationally simpler than SIFT and more robust to illumination problems. Another interesting result in region description is reported in Nowak et al. (2006), where it is shown that random sampling, in the case where a large number of regions is available, gives equal or better classification rates than the other more complex operators that are in common use. Some recent effort on visual recognition for very large databases are (Lin et al., 2011; Krizhevsky et al., 2012; Perronnin et al., 2010).

Some recent advances in the problem of building recognition are also noteworthy (Hutchings and Mayol-Cuevas, 2005; Jing and Allinson, 2009). The specific difficulties of this task are the various forms of occlusions encountered (e.g., trees and moving vehicles) and the varying viewpoints in the images. In Hutchings and Mayol-Cuevas, (2005); and Jing and Allinson, (2009) global features (intensity and color information at different scales) and local features (Gabor features at several different scales and orientations) were extracted from a database of building images and used as a powerful feature vector. Moreover, in Jing and Allinson (2009) several subspace learning-based dimensionality reductions were tested and compared to improve performance and to alleviate computational complexity.

Starting from these and other results, we report improvements of our previously published generic system for object recognition (Nanni et al., 2012, 2013). The new system reported in this paper is based on the following ideas:

- The utilization of both local and global descriptors to represent images; we fuse several texture descriptors.
- Dimensionality reduction of the texture descriptors using principal component analysis (PCA) according to the PCA-SIFT approach (Ke and Sukthankar, 2004); PCA handles the problems of high correlation among the features as well as the curse of dimensionality. Different projections are performed retaining different training subsets for building different projection matrices. In this way it is possible to build an ensemble of classifiers by varying the projection matrix. For each projection matrix a different classifier is trained.
- The utilization of the BoW approach by computing textons considering different clusterings; each cluster is performed separately using a subset of the images of each class. In this way different global texton vocabularies are created, and for each vocabulary a different SVM is trained.
- A new method proposed in this paper that is based on cloud of features where all the subwindows extracted from a given region of the image are used to train a one-class support vector machine.

The strength of this paper lies in the detailed experiments that, together with the shared code, may provide helpful bases for researchers interested in image classification, especially for students who are new to the topic. Different local descriptors, codebook generation methods, subwindow configurations, etc. are combined together and state-of-the-art results are obtained in the tested datasets.

Our new generic system is compared with other approaches using several well-known and widely used datasets: a 15-class scene dataset (Xiao et al., 2010), a building recognition dataset (Amato et al., 2010), the caltech-256 dataset (Griffin et al.), and the person classification dataset of the object classification contest of VOC2012. The new VOC2012 is the last of a very famous series of computer vision competitions, where our system was submitted as a participant so that we could report a comparison of our system with the true state-of-the-art of 2012. In 2001 the accuracy in the 15-class scene dataset was only 73.3%; by 2012 it had become 88.1% (Xiao et al., 2010). The system proposed in this paper obtains an accuracy of 88.3% in the scene dataset; 95.6% in the building recognition; 40% in the caltech-256 dataset, and 88.7% in the person-classification VOC2012 dataset.

A full-feature MATLAB toolbox containing all the source codes used in our proposed system is available at http://www.dei.unipd.it/wdyn/?IDsezione=3314&IDgruppo_pass=124&preview=. We plan on maintaining this toolbox and

updating it with new descriptors useful for the object/building/scene recognition problem. Our hope is that this toolbox will serve as the foundation for further explorations by other researchers in the field.

The rest of this paper is organized as follows. In Section 2 the texture descriptors used in our system are briefly reviewed. In Section 3 the proposed approach for object/building/scene recognition is explained in detail, and in Section 4 experimental results are presented and discussed. We conclude in Section 5.

2. Descriptors

Below we describe some of the state-of-art descriptors used in our system and experiments.

2.1. SIFT descriptor

The SIFT[1] descriptor (Lowe, 2004) is a 3D histogram that takes the gradient locations (in our case quantized into a 4×4 location grid) and orientations (quantized into eight values) and weighs them by the gradient magnitude and a Gaussian window superimposed over the region. The SIFT descriptor, which is obtained by concatenating the orientation histograms over all bins and normalized to unit length, shows how the local gradients around a point are aligned and distributed at different scales.

2.2. Local ternary patterns (LTP)

LTP is a recent variant (Tan and Triggs, 2007) of LBP. The LBP operator is rotation invariant and evaluates the binary difference between the gray value of a pixel $\mathbf{x}$ and the gray values of P neighboring pixels on a circle of radius R around $\mathbf{x}$. A problem with conventional LBP is its sensitivity to noise in the near-uniform image regions. The three value encoding scheme of LTP overcomes this problem. The implementation of LTP used in our experiments is a modification of the original LBP Matlab[2] code. It includes three value encodings and a normalized histogram.

In our experiments, we used both the rotation invariant bins and the uniform bins, with each descriptor used to train a different classifier. The final descriptor was obtained by concatenating the features extracted with ($R = 1$, $P = 8$) and ($R = 2$, $P = 16$). We tested both LTP with uniform bins (LTP-u) and LTP with rotation invariant uniform bins (LTP-r).

2.3. Local phase quantization (LPQ)

LPQ[3] is a texture descriptor (Ojansivu and Heikkila, 2008) that uses the local phase information extracted from the 2-D short-term Fourier transform (STFT) computed over a rectangular neighborhood of radius R at each pixel position in an image. Only four complex coefficients, corresponding to the 2-D frequencies, are considered and quantized using a scalar quantizer between 0 and 255. The final descriptor is the normalized histogram of the LPQ values. Different LPQ descriptors were evaluated in our experiments, with two selected for our final system. Both of these were extracted by varying the parameter R (specifically, $R = 3$ and $R = 5$), and each descriptor was used to train a different classifier.

2.4. GIST

The GIST descriptor (Oliva and Torralba, 2001) computes the energy of a bank of Gabor-like filters evaluated at 8 orientations and 4 different scales. The square output of each filter is then averaged on a 4×4 grid.

2.5. The histogram of oriented edges (HOG)

One way of looking at HOG (Dalal and Triggs, 2005) is as a simplified version of SIFT. HOG calculates intensity gradients from pixel to pixel and selects a corresponding histogram bin for each pixel based on the gradient direction.

The HOG features extracted in our experiments used a 2×2 version of the HOG. The HOG features were extracted on a regular grid at steps of 8 pixels and stacked together considering sets of 2×2 neighbors to form a longer descriptor with more descriptive power.

2.6. Daubechies wavelets (DW)

As explained in Huang et al. (2003), DW is a feature extraction method where the average energy of the three high-frequency components is calculated up to the *Lth* level decomposition using both the scaling and the wavelet functions of the selected wavelet. In our experiments we use decomposition $L = 10$ coupled with Daubechies 4 wavelet function.

2.7. Laplacian features (LF)

LP, as proposed in Xu et al. (2012), is based on a SIFT-like descriptor extracted at different window sizes. A descriptor called the multifractal spectrum (MFS) then extracts the power-law behavior of the local feature distributions over the scale. Finally, to improve robustness to changes in scale, a multi-scale representation of the multi-fractal spectra under a wavelet tight frame system is proposed.

2.8. Local derivative pattern (LDP)

As detailed in Zhang et al. (2010), LDP is a general framework that encodes directional pattern features based on local derivative variations. The LDP templates extract high-order local information by encoding various distinctive spatial relationships contained in a given local region.

2.9. Speeded up robust features (SURF)

SURF Xiao et al., 2010 is an improvement of the famous SIFT features (Lowe, 2004). SURF extract features starting from interest points detected by a method based on the Hessian matrix. A set of features based on Haar wavelet response around

[1] Matlab code available at http://www.vlfeat.org/~vedaldi/code/sift.html.

[2] Matlab code available at http://www.ee.oulu.fi/mvg/page/lbp_matlab.

[3] LPQ code available at http://www.ee.oulu.fi/mvg/download/lpq/.

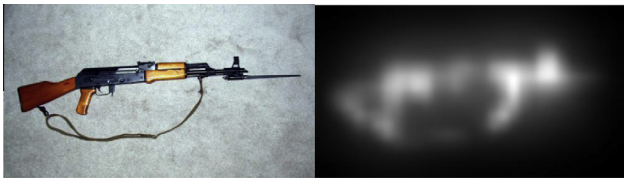

Figure 1 An example of an image and its saliency map.

a point of interest is then extracted. To speed the feature extraction step, the integral image is used.

3. Proposed approach

In this section, we explain the steps of our proposed approach. In an outline form, they are the following:

- *STEP 1*: *PRE-PROCESSING*. The image is normalized using contrast-limited adaptive histogram equalization.[4] The image is then resized so that the lower dimension is at least 50 pixels.
- *STEP 2*: *GLOBAL DESCRIPTORS*. The whole image is divided into four equal regions without overlap, and a central region of the same dimension is extracted. Since in most computer vision applications it is important to extract features only in the foreground region, we used a method proposed in Hou et al. (2012) for extracting a saliency map from the image. For each region we extract three sets of descriptors: one from the original image and the other two from two foreground regions (different combinations were tested, see Section 4). Each pixel which saliency higher than a prefixed threshold (0.15 and 0.25 in this work) is determined to be part of the foreground. For each region, different descriptors are extracted, and for each descriptor a different SVM is trained. Results are pooled by sum rule. An example of a given image and its saliency map is shown in Fig. 1.
- *STEP 3*: *SUBWINDOWS*. Each image is divided into overlapping subwindows with the size specified as a percentage (ps) of the original image taken at fixed steps $st = \min(ps \times l, ps \times h)/2$, where $l \times h$ is the size of the original image. We tested different values of ps. In our final version, we used both $ps = 12.5\%$ and $ps = 8\%$ (see the experimental section for more details).
- *STEP 4*: *LOCAL DESCRIPTORS*. A local feature extraction method is performed by evaluating different texture descriptors from each subwindow.
- *STEP 5*: *DIMENSIONALITY REDUCTION BY PCA*. Each local descriptor is transformed according to PCA (calculated as in *TRAINING*2).
- *STEP 6*: *CODEBOOK ASSIGNATION*. Two different approaches are used for the codebook assignation:
  (1) Each descriptor is assigned to one codebook (created as in *TRAINING3*) according to the minimum distance criterion;
  (2) Each image is divided into four equal regions, and a different codebook assignation is performed separately for each region. For each codebook, the method proposed in Feng et al. (2012) is applied. In this way, there are four different codebooks for each image, with each region encoded into a 30-dimensional feature vector; the four 30-dimensional feature vectors are concatenated and used to represent a given image.
- *STEP 7*: *CLASSIFICATION*. Each global and local descriptor extracted from the image is classified by an SVM (trained as in *TRAINING*1).
- *STEP* 8: *FUSION*. The classifier results are combined using the sum rule, i.e., by selecting as the final score the sum of the scores of a pool of classifiers that belong to an ensemble. Before fusion, the scores of each classifier are normalized to a mean of 0 and standard deviation of 1.
- *TRAINING 1*: *SVM*. A different SVM is trained for each local or global descriptor. SVM (Duda et al., 2000) is a general purpose two-class classifier that finds the equation of a hyperplane that maximally separates all the points between the two classes. SVM handles nonlinearly separable problems using kernel functions to project the data points onto a higher-dimensional feature space. Multi-class problems can be discriminated by performing, for example, several "one-versus-all" (OVA) classifications (OVA is used in this paper). We used two different kernels in our experiments: (1) histogram for BoW and (2) the radial basis function for global descriptors. Because our system is general purpose, we used the same kernel and the same parameters in all the tested datasets (see the experimental section).
- *TRAINING 2*: *PCA*. A set of 250,000 subwindows is randomly extracted from the training set (considering the different classes) and used to construct the PCA matrix (one projection matrix for each descriptor). This step is iterated several times (five times retaining 99% of the variance and five times retaining 98% of the variance); for each iteration, a different codebook is created and a different SVM is trained.
- *TRAINING 3*: *CODEBOOK CREATION*. A different set of textons is created for each class of the dataset, and for each NIMG[5] image a texton is built clustering a local descriptor with k-means, with the number k randomly selected between 10 and 40. For each descriptor, the final texton vocabulary (codebook) is obtained by concatenating the textons over all classes. Since k-means is an unstable clustering approach, we run it twice to obtain more codebooks (for both these codebooks a different SVM is trained).

3.1. Cloud of features

In our experiment we also tested a novel method for object recognition based on cloud of features (Lai et al., 2004). A cloud represents image i as M_i feature vectors, storing the information on M_i single points, or patches in the image. Each point/patch is a "subwindow" of fixed size (we run three methods with size = {8, 10, 12}. The cloud points, C_i, of an image, I_i, are formed by these patches. From each point/patch (in this paper we extracted {150, 250, 350} points/patches from each image), we extracted a given descriptor, giving us 9 classifiers (three size × the number of retainer patches). These nine classifiers were combined by sum rule.

[4] Using the function adapthisteq.m in MATLAB.

[5] Images are clustered in groups due to computational issues.

In our experiments, we fit a one-class SVM (Maneivitz and Yousef, 2002), implemented as in libSVM (www.csie.ntu.edu.tw/~cjlin/libsvm/) around the cloud of points C_i and enclose the data by a hypersphere H_i of minimal volume. We can describe this process formally as follows:

Let the hypersphere be described by the center a and the radius R. For a vector $\mathbf{x}$, coming from the cloud of points C_j, representing the image I_j, we define:

$$C^i(\mathbf{x}) = \mathbf{Q},$$

where $\mathbf{x}$ is accepted by the one-class classifier H_i that describes the cloud points of the image pair I_i and where Q is the indicator function (for instance, Q(A) = 1, if the condition A is true and 0 otherwise). In our experiments the acceptance of the vector $\mathbf{x}$ is defined as follows:

1, if x is accepted by the one-class classifier;
0, if x is rejected by the one-class classifier.

An image I_j could be classified by taking into account the fraction of vectors from the cloud representation C_j, which are rejected by the classifier H_i:

$$S_i(I_j) = \left(\frac{1}{M_j}\right)\sum_{x\epsilon C_j}(1 - C^i(\boldsymbol{x}))$$

where M_j is the size of the cloud C_j .

Notice that if only one classifier is used, the performance may suffer from a large overlap between individual clouds of points. For instance, if a cloud subsumes another originating from a different class, the percentage of outliers (vector $\mathbf{x}$ rejected) can still be zero. Such a situation lowers the performance of the whole system. To prevent this from happening, the information given by all the classifiers is combined. The relations among the images of the training set can be evaluated using a "classifier profile," which expresses the dissimilarities among a given image from all the images of the training set. The classifier profile of a test image is now classified by SVM trained using the classifier profile of the images that belong to the training set. We assign each image of the testing set to the class of the nearest neighbor (for a better mathematical description of classifier profile, see Lai et al., 2004).

4. Experimental results

In our experiments we tested our approach on the following different datasets and object classification problems:

- Scene recognition: the 15-class scene dataset widely used in the literature (Oliva and Torralba, 2001);
- Caltech-256: one the most familiar object classification datasets (Griffin et al.);
- Building recognition; 12 buildings in the city of Pisa, Italy (Amato et al., 2010);
- Person recognition: the PASCAL Visual Object Classes Challenge 2012 protocol (VOC2012) Everingham et al., 2012.

4.1. Datasets description

4.1.1. Scene dataset (Oliva and Torralba, 2001)

This dataset has the following fifteen categories (we set NIMG = 50): coast (360 images), forest (328 images), mountain (274 images), open country (410 images), highway (260 images), inside city (308 images), tall building (356 images), street (292 images), bedroom (216 images), kitchen (210 images), living room (289 images), office (215 images), suburb (241 images), industrial (311 images), and store (315 images). Images are approximately 300×250. Fig. 2 shows a few samples.

The testing protocol established by other papers in the literature for the scene dataset, which we followed, requires 5 experiments, each using 100 randomly selected images per category for training and the remaining images for testing. The

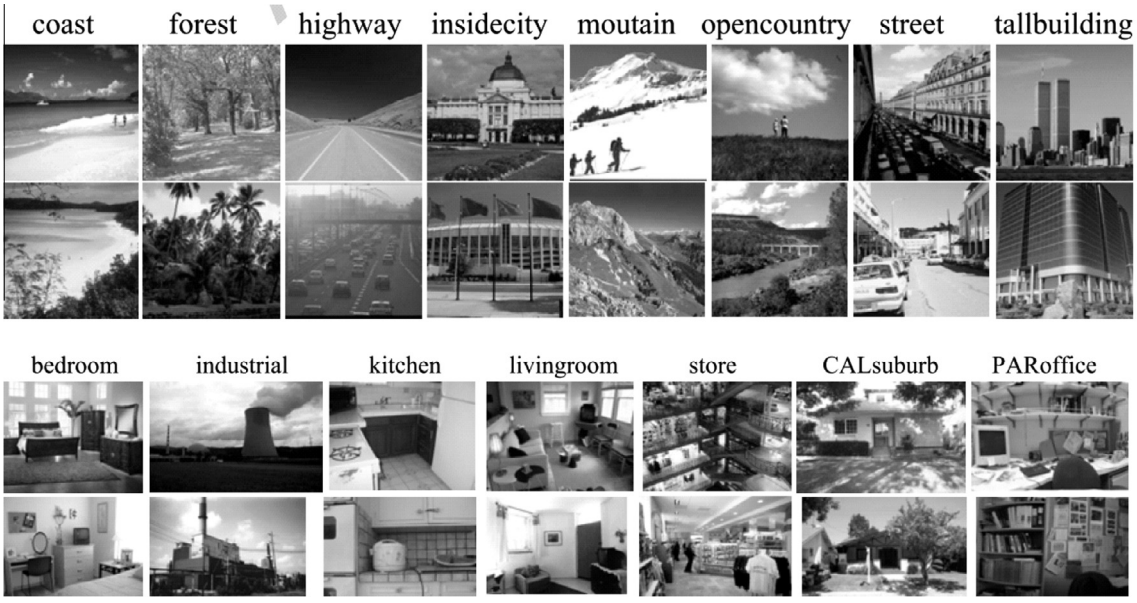

Figure 2 Samples from the scene dataset.

performance indicator is the accuracy, which is averaged across the experiments.

4.1.2. Caltech-256

This dataset includes a challenging set of 256 object categories containing a total of 30,607 images with at least 80 images for each category. The images in Caltech-256 were collected by choosing a set of object categories, and then by downloading examples from Google Images. The final dataset was produced by manually screening out all images that did not fit the chosen category. According to a widely used protocol, we ran 5 split tests using 40 images per class for training (we set NIMG = 40) and 25 for testing. The performance indicator was the accuracy, which was averaged on the 5 experiments. Fig. 3 provides samples of some images contained in the Caltech-256 dataset.

4.1.3. Building recognition

This dataset (Amato et al., 2010) contains 1227 photographs crawled from Flickrs of landmarks located in Pisa. The dataset is divided into 12 classes having a minimum of 46 images per class. According to the official testing protocol (Amato et al., 2010), the dataset should be divided into a training set of 921 photos (we set NIMG = 50), or approximately 80% of the dataset and a testing set of 226, or approximately 20% of the dataset. For comparison purposes these two sets are provided on the web page of one of the authors of Amato et al. (2010). The performance indicator used in our experiments was accuracy. Some sample images are shown in Fig. 4.

The *VOC*2012 *dataset* (Everingham et al., 2012) includes twenty classes of images from four main categories:

- *Person*: people;
- *Animal*: bird, cat, cow, dog, horse, and sheep;
- *Vehicle*: airplane, bicycle, boat, bus, car, motorbike, and train;
- *Indoor*: bottle, chair, dining table, potted plant, sofa, and tv/monitor.

According to the official protocol, we use *Trainval* images for training and the *Testing* set of images for testing. Several example images of the VOC2012 dataset are available at http://pascallin.ecs.soton.ac.uk/challenges/VOC/voc2012/examples/index.html. The official performance indicator is the average precision (AP). For a given task and class, the precision/recall curve is computed from a method's ranked output. *Recall* is defined as the proportion of all positive examples ranked above a given rank. *Precision* is the proportion of all examples above that rank which are from the positive class. The AP summarizes the shape of the precision/recall curve. It is defined as the mean precision at a set of eleven equally spaced recall levels $[0, 0.1, \ldots, 1]$ (Everingham et al., 2010).

Due to restraints on computation time, we focused only on the person image classification contest (we set NIMG = 250).

4.2. Empirical results

The first experiment, reported in Tables 1–3, was aimed at comparing variants of the steps in our proposed approach.

Figure 3 Samples from the Caltech dataset.

Figure 4 Samples from the landmark dataset.

Table 1 Performance of the proposed approach in the scene dataset.

	Scene								
	LTP-u	LPQ(3)	LPQ(5)	LTP-r	HOG	GIST	LDP	LF	DW
B	67.24	57.35	55.34	55.71	50.08	61.81	68.41	58.43	55.04
*B*1	81.21	76.08	74.71	73.50	68.17	75.04	78.59	74.67	71.79
*M*1	79.87	73.74	72.06	71.02	63.42	71.76	77.32	66.20	63.69
*M*2	77.15	69.78	67.04	68.48	59.70	68.58	74.17	62.18	59.10
*Fus*1	81.41	76.88	75.54	74.45	70.65	75.58	78.96	75.51	74.10
*Fus*2	81.17	77.49	75.31	74.71	69.21	74.64	78.56	74.74	73.70
*ALL*1	87.10								
*ALL*2	86.70								
Xiao et al. (2010)	81.2								
Huang et al. (2011)	82.6								

Table 2 Performance of the proposed approaches in the building dataset.

	Building								
	LTP-u	LPQ(3)	LPQ(5)	LTP-r	HOG	GIST	LDP	LF	DW
B	88.05	89.82	88.50	72.57	76.99	83.63	82.74	77.43	65.49
*B*1	91.59	95.58	94.69	81.86	86.28	88.05	85.85	89.38	81.42
*M*1	92.48	94.69	96.02	83.63	86.28	92.92	89.82	89.82	81.86
*M*2	91.15	94.69	94.69	86.28	88.50	89.82	89.38	88.05	78.76
*Fus*1	92.48	95.58	96.02	83.63	88.50	92.92	88.50	73.45	89.82
*Fus*2	93.81	95.58	96.46	85.84	89.82	93.36	89.82	91.15	85.84
*ALL*1	95.13								
*ALL*2	95.13								
Amato et al. (2010)	92								

Table 3 Performance of the proposed approaches in the Caltech-256 dataset.

	Caltech-256								
	LTP-u	LPQ(3)	LPQ(5)	LTP-r	HOG	GIST	LDP	LF	DW
B	12.15	10.62	10.92	9.77	7.26	17.43	11.27	6.87	5.45
*B*1	22.49	21.50	23.45	13.43	12.14	28.83	18.43	12.91	9.65
*M*1	21.80	22.12	23.50	13.49	10.94	27.69	17.76	11.79	8.22
*M*2	20.63	20.73	21.51	12.92	10.38	24.46	17.23	10.64	7.57
*Fus*1	23.80	23.78	25.75	14.80	12.30	31.16	19.17	14.50	10.79
*Fus*2	24.10	24.79	25.98	15.07	12.14	31.43	19.40	14.92	11.25
*ALL*1	37.67								
*ALL*2	38.60								

In this table performance of the descriptors detailed in Section 2 is reported.

In particular, we evaluated the performance obtained by considering different global descriptors (STEP2 in Section 3). In these tests we used only a stand-alone SVM, with the radial basis function kernel and no dataset parameter tuning[6] for each dataset. We compared the following different approaches for the global descriptors:

- B: the features were extracted from the whole image.
- B1: the image was divided in four equal regions without overlap and a central region of size 1/2 of the original image. For each region a different SVM was trained; the five descriptors were then used to train four SVMs combined by sum rule.
- M1: as in B1, but instead of the original image, we used the foreground region extracted method using the saliency map with $TH = 0.15$.
- M2, as in B1, but instead of the original image, we used the foreground region extracted using the saliency map with $TH = 0.25$.

The following fusions among descriptors were also compared:

- *Fus*1: sum rule among the descriptors based on B1 and M1;
- *Fus*2: sum rule among the descriptors based on B1, M1, and M2;
- *ALL*1: sum rule among the descriptors of *Fus*1;
- *ALL*2: sum rule among the descriptors of *Fus*2.

[6] Parameters −g 0.1–c 1000.

In the Scene dataset (see Table 1), *B*1 outperforms *M*1. This is due to the fact that the background region discarded by the saliency map contained important information for classifying a given image in a given scene class. The fusion, *Fus*1 (sum rule between *B*1 and *M*1), outperforms B1 because different descriptors are extracted from the two different images (the original image when B1 is applied and the foreground region when M1 is used).

In the building recognition task, it is typically important to discard background regions. For this reason, *M*1 outperforms *B*1 in the Building dataset (see Table 2). The best results are also obtained in this dataset by combining different approaches, with *Fus*2 outperforming the other approaches. LPQ(5) works better than ALL1 or ALL2 in the Building dataset but not on others: when using this approach, it would be desirable to test it using the training data to determine whether it is suited to that specific classification.

In the Caltech-256 dataset (see Table 3), the saliency approaches do not outperform B1, but both *Fus*1 and *Fus*2 outperform *B*1, *M*1, and *M*2.

From the results reported in Tables 1–3, the following conclusions can be drawn:

- In the building dataset, it is clear that global descriptors work well. Notice that several of our approaches outperform the SIFT based method reported in Amato et al. (2010), which obtains an accuracy of 92%.
- It is interesting to note how differently the same descriptor works in each of the datasets. In the scene/landmark datasets, e.g., GIST works poorly with respect to the object classification dataset. Fusion, it should be noted, works well in all the tested datasets.
- For a statistical validation of our experiments, we have used the Wilcoxon signed rank test (Demsar, 2006) to compare *FUS*2 and *B*1 (considering the different datasets and the different descriptors); we found a statistical difference with a *p*-value of 0.05.
- Finally, we should note that some state-of-art stand-alone approaches obtain a performance that is similar to our stand-alone best methods. The SIFT based method tested in Xiao et al. (2010) (which was the best method using an ensemble approach), for example, obtained an accuracy of 81.2% in the scene dataset, while the salient coding approach (Huang et al., 2011), which is a better performing variant of LLC (the winner of VOC2009), obtained 82.6% in the scene dataset.

Table 4 Performance of the proposed approach using different descriptors in the scene dataset.

	−1	M−1	−2	M−2	Local
LPQ R = 3	72.86	74.10	67.54	71.12	**79.50**
LPQ R = 5	72.60	74.14	63.18	68.61	**78.73**
LTu	69.18	71.26	66.60	70.45	**76.98**
LTr	63.28	65.59	64.76	67.77	**74.57**
GI	75.41	75.95	70.45	74.67	**82.01**
HO	72.73	74.54	70.59	74.30	**80.50**
Sift	–	–	–	–	**62.11**
Lap	69.55	70.49	60.34	67.91	**77.55**
LDP	73.47	74.64	67.71	72.09	**80.54**
Surf	–	–	–	–	**66.77**
Fusion	85.80				
Xiao et al. (2010)	81.2				
Huang et al. (2011)	82.6				

Higher accuracy for each descriptor.

Table 5 Accuracy obtained by the state-of-the-art approaches in the scene dataset.

Approach	Year	Accuracy (%)
Oliva and Torralba (2001)	2001	73.3
Lazebnik et al. (2006)	2006	81.4
Liu and Shah (2007)	2007	83.3
Wu and Rehg (2009)	2009	83.1
Xiao et al. (2010)	2010	88.1 (ensemble) 81.2 (stand-alone)
Gu et al. (2011)	2011	83.7
Meng et al. (2012)	2012	84.1
Nanni et al. (2012)	2012	82.0
Nanni et al. (2013)	2012	87.1
Elfiky et al. (2012)	2012	85.4
EDL	2012	**88.3**

In the BoW approach proposed in this work, we used a stand-alone SVM as our classifier with the histogram intersection kernel (without any parameters tuning for each dataset[7]: the same settings were used in all the datasets and with all the descriptors). Using the same kernel with the same parameters for all the datasets and all the descriptors is very useful for practitioners since they can use this same set of features in their datasets.

In our second set of experiments, reported in Table 4, we compare some variants in the steps of our proposed approach using the scene dataset. We tested several descriptors derived from the original image – all those described in Section 2 and their fusion by sum rule. We label the methods as follows:

- −1: a version of our system based on a single codebook creation (only one PCA projection) with patches with *ps* = 8% and only the first method of STEP6 for codebook creation is used.
- *M*−1: a version of our system based on using all the PCA projections for building patches (see Section 3 – *TRAINING2:PCA*) with *ps* = 8% and only the first method of STEP6 for codebook creation.
- −2: as in −1, but the second method of STEP6 is used.
- *M*−2: as in *M*−1, but the second method of STEP6 is used.
- Local[8]: a complete version of our system based only on a local descriptor; we combine, by sum rule, the scores obtained by the SVMs trained using the textons vocabulary obtained both with *ps* = 12.5% and *ps* = 8%.
- *Fusion*: the fusion of all the *Local* descriptors by sum rule.

As can be seen in Table 4, our ensemble approach improves the performance of stand-alone approaches (compare −1 with *Local*), thus gaining a performance similar to state-of-the-art stand-alone approaches: the SIFT based method tested in Xiao

[7] $C = 0.25$.

[8] Both the approaches for the codebook assignation are used, all the 10 PCA projections are used.

Table 6 Accuracy obtained by bag of word approaches.

		−1 (%)	M−1 (%)	−2 (%)	M−2 (%)	Local (%)
Building	*LPQ R* = 3	92.4	94.7	10.2	42.0	95.1
	LPQ R = 5	89.8	92.5	52.2	49.1	93.0
	LTP-u	75.7	79.6	65.5	73.0	80.5
	LTP-r	70.4	77.9	59.3	64.2	79.7
	GI	92.9	93.8	76.5	80.1	95.1
	HO	91.6	91.2	9.8	42.5	92.0
	Fusion	95.6				
Caltech	*LPQ R* = 3	16.66	16.90	18.07	18.33	20.43
	LPQ R = 5	17.29	17.64	13.78	14.66	22.04
	LTP-u	11.28	11.39	10.38	10.46	14.63
	LTP-r	9.36	9.50	9.25	9.38	12.73
	GI	18.06	20.49	12.00	13.91	23.57
	HO	14.30	14.32	15.28	15.66	17.25
	Fusion	28.3				

Table 7 Accuracy obtained by the state-of-the-art approaches.

Dataset	Approach	Accuracy (%)
Building	***EDL***	95.6
	Nanni et al. (2013)	95
	SIFT (Amato et al., 2010)	92
	Color-SIFT (Amato et al., 2010)	82
	SURF (Amato et al., 2010)	90
Caltech	***EDL***	40.0
	Nanni et al. (2013)	40.0
	Gehler and Nowozin (2009)	48.9
	Yang et al. (2009) (*Ntrain* = 45)	37.5
	Perronnin et al. (2010) (*Ntrain* = 45)	45
	Lin et al. (2011) (*Ntrain* = 45)	45.3

Table 8 Accuracy obtained by cloud of features approach.

	Cloud	Local	Sum
Building			
LPQ R = 3	84.25 (77.52)	95.1	**95.4**
HO	83.21 (78.21)	92.0	**92.3**
LTP-u	74.11 (67.54)	80.5	**82.5**
Scene			
LPQ R = 3	72.53 (69.08)	79.50	**80.45**
HO	73.11 (69.95)	80.50	**80.85**
LTP-u	71.42 (68.21)	76.98	**78.75**

Higher accuracy for each descriptor.

et al. (2010) obtained 81.2%, while salient coding based method proposed in Huang et al. (2011) obtained 82.6%.

In Table 5 we report the results obtained by our system compared to the best methods reported in the literature for the scene dataset. In the following tables we named the whole system detailed in Section 3 as ensemble of different local descriptors (EDL), i.e. global descriptors combined with the local descriptors.

Due to computational time factors, we used fewer descriptors for our bag of feature approach in the Caltech and Building datasets. Results are reported in Table 6. In the Building dataset, we used images extracted from the foreground region with the saliency map TH = 0.15, since this method using global descriptors produced the best results. We used the original images in the Caltech dataset because this produced the best results with this dataset.

In the Building dataset, the second approach for building the codebook works very poorly, so in *Local* this approach was not considered in this dataset. It is clear in Table 6 that the ensemble outperforms the stand-alone approach (compare *Local* with −1).

In Table 7 we compare results obtained by our complete system with the best methods reported in the literature. It should be noted that in Demsar (2006), Gehler and Nowozin (2009) and Yang et al. (2009) 45 images in Caltech-256 dataset were used for training each class, making their training sets larger than those used in our system. In Torresani et al. (2010) several approaches are compared (see Fig. 2 of that paper[9]), and only LPbeta (i.e. Gehler and Nowozin, 2009) among the approaches based on 40 training images obtained a performance higher than 40%.

In Table 8, we report the results obtained by the cloud of feature approach (*Cloud*), comparing it with the local approach detailed in Section 2 (*Local*) and their fusion by weighted sum rule (*SUM*1) where the weight of *Local* is three and the weight of *Cloud* is 1. *Cloud* is applied separately in the five regions (*B*1), and then these five scores are combined by sum rule.

To reduce the computational time, we used only three descriptors and ran tests on only two datasets. These preliminary results show that the method proposed here could be considered a new approach for object classification.

In Table 8, each cell of *Cloud* reports two values. The first is the performance obtained considering the fusion among the nine approaches detailed in Section 3.1. The value between the brackets is the performance obtained by a standalone method, with *DIMsize* = 10 and 150 patches retained in each image. It is clear in these experiments that *Cloud* is very well suited for building an ensemble of classifiers.

[9] http://www.cs.dartmouth.edu/~lorenzo/Papers/tsf-eccv10.pdf.

Table 9 Results on a subset of the validation set.

	−1	M−1	−2	M−2	Local
Local					
LPQ R = 3	44.25	52.68	40.25	53.25	62.35
LPQ R = 5	48.85	56.36	41.19	55.76	66.55
LTP-u	53.03	59.03	42.88	46.85	63.88
LTP-r	48.58	52.10	36.52	42.21	57.36
GI	53.25	60.67	47.45	57.54	65.48
HO	55.50	56.36	46.05	58.27	65.73
Global					
LPQ R = 3	48.82				
LPQ R = 5	45.31				
LTP-u	60.32				
LTP-r	41.25				
GI	58.90				
HO	41.59				
*F*1	73.70				
*F*2	73.90				
*F*1 + *F*2	78.90				
Fp	87.40				

Finally, in Table 9, we report the results on the VOC2012 contest. We ran our approach on only the person classification dataset. Since our ensemble approach needs bounding boxes that contain a given object to classify, we used the method proposed in Bourdev and Malik (2009) for extracting subwindows that might contain a person (we simply retained the 15 regions with higher similarity to the person template used (Bourdev and Malik, 2009). Since each subwindow was of a different dimension, we resized each so that the minimum size was 40 pixels. Each subwindow was classified as *person*/*non-person* using our approach.

The results reported in Table 9 are a subset of the validation set (500 person images and 1500 non-person images) and only the *B*1 approach is used for the global descriptor (the saliency map is not considered). Moreover, the TRAINING2 step is iterated only four times (two times retaining 99% of the variance and two times retaining 98% of the variance). These restrictions were due to constraints in computation time. It should be remarked that this dataset is built by images with complex backgrounds that contain no information about the class of the images. Increasing the dimension of the ensemble would likely boost the performance of our system. Examining the results in Table 9, it is clear that the ensemble once again improves performance.

Since a validation set is available, we also ran some experiments for optimizing the approach:

- F1, the best fusion by sum rule of LPQ R = 3, GIST, and LTP-u among the global approaches, obtained an average precision (*ap*) of 73.7;
- F2, the fusion among the local approaches, obtained an *ap* of 73.9;
- F1 + F2, the fusion between F1 and F2, obtained an *ap* of 78.9;
- Fp, the fusion by weighted sum rule of F1, F2, and the score obtained by poselet (Bourdev and Malik, 2009) (weight of poselet = 4), obtained an *ap* of 87.4.

We submitted our best approach as a competitor for the VOC2012 contest. The result obtained in the VOC2012 contest was 88.7%. Notice that in this paper we have not used the images used in the test set of VOC2012, since their labels are unavailable, in this work we have used a validation set. Since the VOC2012 classification dataset is the same as that used in 2011 and no additional data has been annotated, we can fairly compare our approach with both the competitors of VOC2011 and VOC2012 (http://pascallin.ecs.soton.ac.uk/challenges/VOC). Among the 26 competitors of VOC2011/VOC2012 contests, we rank in the 10th position. Except for Panasonic and the National Laboratory of Pattern Recognition, Institute of Automation Chinese Academy of Science, we obtained a performance that was similar to the other state of the art approaches. Moreover, since we simplified our approach for the competition by reducing the number of components due to restrictions in the computation time. We would expect to obtain an even better performance using the complete system described in this paper.

5. Conclusion

In this paper we presented a new method that combines a feature extraction approach from regions of the image by considering the saliency of the image using a bag of features approach. We explored variations of both based on a combination of different descriptors for recognizing object categories and scene.

For improving the performance of each descriptor, we combined different codebooks obtained in different ways (e.g., via different descriptors and different clusterings for different codebook creations) to enrich the power of codebook representations. Finally, the descriptors are used to train a standalone SVM. Without any ad hoc optimization of SVM per dataset, our approach obtains a very high performance on different object recognition datasets.

In the tested datasets the classes are often imbalanced. It is widely known in the literature (Gang Wu and Chang, 2006) that several classifiers tend to treat data in the minority class as noise, resulting in a class boundary that unduly benefits the majority class. For handling this problem several approaches have been proposed in the literature. A very popular one is SMOTE, which increases diversity by generating pseudo minority class data (Chawla et al., 2002). We have tried to couple SVM with SVM but the performance of whole system is only slightly better, and we have not found any statistical difference, also with a *p*-value of 0.1.

As future work we want to couple SVM with some more recent systems, see (Wang and Yao, 2012), for handling the unbalancing problem.

As further future work we will try to improve the results by improving the classifier. Empirical results, reported in Li et al. (2003), show that the bagging SVM outperforms the standalone SVM and other ensemble of classifiers.

Acknowledgement

The authors would like to thank all the other researchers that have shared their MATLAB code.

References

Amato, G., Falchi, F., Bolettieri, P., 2010. Recognizing landmarks using automated classification techniques: evaluation of various visual features. In: Second International Conferences on, Advances in Multimedia, pp. 78–83.

Baumberg, A., 2000. Reliable feature matching across widely separated views. In: IEEE Conference on Computer Vision and Pattern Recognition, Hilton Head Island, SC, pp. 774–781.

Bourdev, Lubomir, Malik, Jitendra, 2009. Poselets: body part detectors trained using 3D human pose annotations. In: ICCV.

Chawla, N.V., Bowyer, K.W., Hall, L.O., Kegelmeyer, W.P., 2002. SMOTE: synthetic minority over-sampling technique. Journal of Artificial Intelligence Research 16, 321–357.

Dalal, N., Triggs, B., 2005. Histograms of oriented gradients for human detection. In: 9th European Conference on Computer Vision, San Diego, CA, 2005.

Demsar, J., 2006. Statistical comparisons of classifiers over multiple data sets. Journal of Machine Learning Research 7, 1–30.

Duda, R.O., Hart, P.E., Stork, D.G., 2000. Pattern Classification, second ed. Wiley, New York.

Elfiky, N.M., Khan, F.S., Weijer, J.v.d., Gonzalez, J., 2012. Discriminative compact pyramids for object and scene recognition. Pattern Recognition 45 (4), 1627–1636.

Everingham, M., Van Gool, L., Williams, C.K.I., Winn, J., Zisserman, A., 2010. The PASCAL Visual Object Classes (VOC) Challenge. International Journal of Computer Vision 88 (2), 303–338.

Everingham, M., Van Gool, L., Williams, C.K.I., Winn, J., Zisserman, A., 2012. The PASCAL Visual Object Classes Challenge 2012 (VOC2012) Results, vol. <http://www.pascal-network.org/challenges/VOC/voc2012/workshop/index.html>.

Feng, J., Ni, B., Yan, S., 2012. Histogram contextualization. IEEE Transactions on Image Processing 21 (2), 778–788.

Fergus, R., Perona, P., Zisserman, A., 2004. A visual category filter for google images. In: European Conference on Computer Vision (ECCV), pp. 242–256.

Gang Wu, Edward Y., Chang, K.B.A., 2006. Kernel boundary alignment considering imbalanced data distribution. IEEE Transactions on Knowledge and Data Engineering 17 (6), 786–796.

Gehler, P., Nowozin, S., 2009. On feature combination for multiclass object detection, ICCV.

Gool, L.J.V., Moons, T., Ungureanu, D., 1996. Affine/photometric invariants for planar intensity patterns. In: 4th European Conference on Computer Vision, pp. 642–651.

Griffin, G., Holub, A., Perona, P., 2007. Caltech-256 Object Category Dataset, vol. California Institute of Technology. <http://resolver.caltech.edu/CaltechAUTHORS:CNS-TR-2007-001>.

Gu, Guanghua, Zhao, Yao, Zhu, Zhenfeng, 2011. Integrated image representation based natural scene classification. Expert Systems with Applications 38 (9), 11273–11279.

Heikkilä, M., Pietikäinen, M., Schmid, C., 2009. Description of interest regions with local binary patterns. Pattern Recognition 42 (3), 425–436.

Hou, X., Harel, J., Koch, C., 2012. Image signature: highlighting sparse salient regions. IEEE Transactions on Pattern Analysis and Machine Intelligence 34 (1), 194–201.

Huang, K., Velliste, M., Murphy, R.F., 2003. Feature reduction for improved recognition of subcellular location patterns in fluorescence microscope images. In: SPIE, pp. 307–318.

Huang, Y., Huang, K., Yu, Y., Tan, T., 2011. Salient coding for image classification. In: Computer Vision and Pattern Recognition (CVPR).

Huang, Yongzhen, Huang, Kaiqi, Yu, Yinan, Tan, Tieniu, 2011. Salient Coding for Image Classification. Computer Vision and Pattern Recognition (CVPR).

Hutchings, R., Mayol-Cuevas, W., 2005. In: Building recognition for mobile devices: incorporating positional information with visual features, vol. CSTR-06-017. Computer Science, University of Bristol.

Jing, L., Allinson, M., 2009. Subspace learning-based dimensionality reduction in building recognition. Neurocomputing 73 (1–3), 324–330.

Ke, Y., Sukthankar, R., 2004. PCA-SIFT: a more distinctive representation for local image descriptors. In IEEE Conference on Computer Vision and, Pattern Recognition, pp. 506–513.

Kelly, P.M., Cannon, M., Hush, D.R., 1995. Query by image example: the Candid approach. In: Proceedings of the of SPIE Conference on Storage and Retrieval for Image and Video Databases III, pp. 238–248.

Krizhevsky, Alex, Sutskever, Ilya, Hinton, Geoffrey, 2012. ImageNet classification with deep convolutional neural networks. In: NIPS.

Lai, C., Tax, D.M.J., Duin, R.P.W., Pekalska, E., Paclik, P., 2004. A study on combining image representations for image classification and retrieval. International Journal of Pattern Recognition and Artificial Intelligence 18 (5), 867–890.

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

Lazebnik, S., Schmid, C., Ponce, J., 2006. Beyond bags of features: spatial pyramid matching for recognizing natural scene categories. In: IEEE Computer Society Conference on Computer Vision and, Pattern Recognition, pp. 2169–2178.

Lew, M.S., Sebe, N., Djeraba, C., Jain, R., 2006. Content-based multimedia information retrieval: state of the art and challenges. ACM Transactions on Multimedia Computing, Communications, and Applications 2 (1), 1–19.

Li, F.-F., Fergus, R., Perona, P., 2003. A bayesian approach to unsupervised one-shot learning of object categories. In: Ninth IEEE International Conference on Computer Vision, pp. 1134–1141.

Lin, Yuanqing, Lv, Fengjun, Zhu, Shenghuo, Yang, Ming, Cour, Timothee, Yu, Kai, Cao, Liangliang, Huang, Thomas, 2011. Large-scale image classification: fast feature extraction and SVM training. In: IEEE International Conference on Computer Vision and Pattern Recognition (CVPR).

Ling, H., Jacobs, D.W., 2005. Deformation invariant image matching. In 10th IEEE International Conference on Computer Vision, pp. 1466–1473.

Liu, J., Shah, M., 2007. Scene modeling using co-clustering. In: IEEE International Conference on Computer Vision.

Lowe, D.G., 2004. Distinctive image features from scale-invariant keypoints. International Journal of Computer Vision 60 (2), 91–110.

Ma, W.Y., Deng, Y., Manjunath, B.S., 1997. Tools for texture- and color-based search of images. In: Proceedings of the of SPIE Conference on Human Vision and Electronic Imaging, San Jose, CA, pp. 496–507.

Maneivitz, L.M., Yousef, M., 2002. One-class SVMs for document classification. Journal of Machine Learning Research 2, 139–154.

Meng, X., Wang, Z., Wu, L., 2012. Building global image features for scene recognition. Pattern Recognition 45 (1), 373–380.

Mikolajczyk, K., Schmid, C., 2005. A performance evaluation of local descriptors. IEEE Transactions on Pattern Analysis and Machine Intelligence 29 (10), 1615–1630.

Müller, H., Michoux, N., Bandon, D., Geissbuhler, A., 2004. A review of content-based image retrieval systems in medical applications – clinical benefits and future directions. International Journal of Medical Informatics 73, 1–23.

Nanni, L., Brahnam, S., Lumini, A., 2012. Random interest regions for object recognition based on texture descriptors and bag of features. Expert Systems with Applications 39 (1), 973–977.

Nanni, L., Brahnam, S., Lumini, A., 2013. Heterogeneous bag of features for object/scene recognition. Applied Soft Computing 13 (4), 2171–2178.

Nowak, E., Jurie, F., Triggs, B., 2006. Sampling strategies for bag-of-features image classification. In: European Conference on Computer Vision (ECCV), pp. 490–503.

Ojansivu, V., Heikkila, J., 2008. Blur insensitive texture classification using local phase quantization. In ICISP, pp. 236–243.

Oliva, A., Torralba, A., 2001. Modeling the shape of the scene: a holistic representation of the spatial envelope. International Journal of Computer Vision 42 (3), 145–175.

Pentland, A., Picard, R.W., Sclaroff, S., 1996. Photobook: tools for content-based manipulation of image databases. International Journal of Computer Vision 18 (3), 233–254.

Perronnin, F., Sánchez, J., Mensink, T., 2010. Improving the Fisher Kernel for large-scale image classification. In: European Conference on Computer Vision.

Perronnin, Florent., Sánchez, Jorge., Mensink, Thomas., 2010. Improving the Fisher Kernel for large-scale image classification. ECCV (4), 143–156.

Schaffalitzky, F., Zisserman, A., 2002. Multi-view matching for unordered image sets. In: 7th European Conference on Computer Vision, pp. 414–431.

Se, S., Lowe, D., Little, J., 2002. Mobile robot localization and mapping with uncertainty using scale-invariant visual landmarks. International Journal of Robotics Research 21 (8), 735–758.

Sivic, J., Zisserman, A., 2004. Video data mining using configurations of viewpoint invariant regions. In: IEEE Conference on Computer Vision and Pattern Recognition, Washington, DC, pp. 488–495.

Sivic, J., Schaffalitzky, F., Zisserman, A., 2004. Object level grouping for video shots. In 8th European Conference on Computer Vision, Prague, Czech Republic, pp. 724–734.

Tan, X., Triggs, B., 2007. Enhanced local texture feature sets for face recognition under difficult lighting conditions. Analysis and Modelling of Faces and Gestures LNCS 4778, 168–182.

Torresani, Lorenzo, Szummer, Martin, Fitzgibbon, Andrew, 2010. Efficient object category recognition using classemes. European Conference on Computer Vision.

Turina, A., Tuytelaars, T., Gool, L.V., 2001. Efficient grouping under perspective skew. In: IEEE Conference on Computer Vision and Pattern Recognition, Hawaii, pp. 247–254.

Tuytelaars, T., Gool, L.V., 2004. Matching widely separated views based on affine invariant regions. International Journal of Computer Vision 59 (1), 61–85.

Wang, S., Yao, X., 2012. Multiclass imbalance problems: analysis and potential solutions. IEEE Transactions on Systems, Man, and Cybernetics, Part B 42 (4), 1119–1130.

Wu, J., Rehg, J.M., 2009. CENTRIST: a visual descriptor for scene categorization, Technical Report GIT-GVU-09-05, Georgia Institute of Technology.

Xiao, J., Hays, J., Ehinger, K., Oliva, A., Torralba, A., 2010. SUN database: large-scale scene recognition from abbey to zoo. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR).

Xu, Y., Huang, S., Ji, H., Fermüller, C., 2012. Scale-space texture description on SIFT-like textons. Computer Vision and Image Understanding 116 (9), 999–1013.

Yang, J., Yu, K., Gong, Y., Huang, T.S., 2009. Linear spatial pyramid matching using sparse coding for image classification. Computer Vision and Pattern Recognition (CVPR), pp. 1794–1801.

Zhang, B., Gao, Y., Zhao, S., Liu, J., 2010. Local derivative pattern versus local binary pattern: face recognition with high-order local pattern descriptor. IEEE Transactions on Image Processing 19 (2), 533–544.