Published in IET Image Processing
Received on 19th May 2013
Revised on 1st October 2013
Accepted on 6th October 2013
doi: 10.1049/iet-ipr.2013.0375

ISSN 1751-9659

Weighting scheme for image retrieval based on bag-of-visual-words

Lei Zhu, Hai Jin, Ran Zheng, Xiaowen Feng

Services Computing Technology and System Laboratory, Cluster and Grid Computing Laboratory, School of Computer Science and Technology, Huazhong University of Science and Technology, Wuhan, People's Republic of China
E-mail: hjin@hust.edu.cn

Abstract: Inspired by the success of bag-of-words in text retrieval, bag-of-visual-words and its variants are widely used in content-based image retrieval to describe visual content. Various weighting schemes have also been proposed to integrate different yet complementary visual-words. However, most of these weighting schemes tend to use fixed weight for every visual-word extracted from the query image, which may lose the discriminative information. This study presents a novel combining method which captures query-specific weights for visual-words in query image. The method mainly contains two stages. Firstly, in offline weight learning, the authors introduce a linear classifier to build a query-category mapping table, and max-margin learning to build category-weight mapping table. Query-category mapping table is used to map the query image to the most likely image class, and category-weight mapping table is used to map image class to the weights of visual-words. Secondly, in online weight mapping, the weights of visual-words are determined efficiently by looking into the pre-learned mapping tables. Experimental results on WANG database and Caltech 101 demonstrate that the proposed weighting scheme can effectively weight visual-words of query image according to their discriminative information. In addition, comparative experiments demonstrate the proposed weighting scheme can obtain higher retrieval performance than other weighting schemes.

1 Introduction

With the popularity of image sharing website and high-quality imaging devices, multimedia data including images has dramatically increased in our daily life. How to accurately retrieve useful information from the mass multimedia data is still a challenging issue. To solve it, content-based image retrieval (CBIR) is developed to retrieve the most similar images in response to a query automatically according to visual content analysis and similarity comparison [1–3].

Visual feature is one of the most important components in CBIR [4]. In order to obtain better retrieval performance, various features are designed to transform visual content to mathematical vector representation. Global features are extracted to capture the global statistics of colour, texture or shape. These features represent image by a single vector. The dissimilarity between images is then calculated by the distance between their corresponding vectors in vector space model (VSM) [5]. The main defects of these features are their failures to handle some complex circumstances such as occlusion, clutter, viewpoint change etc. Local feature has been introduced to represent the interest points detected from the image. Image is described as a set of elementary local features and the similarity between images is calculated by matching sets of local features. Although these features are proved to be reasonably invariant to changes in illumination, noise, rotation, scaling and viewpoint, the matching process suffers from high-computational cost [6].

In order to make local feature more adaptive for image retrieval, bag-of-words (BoW) is applied to CBIR, called as bag-of-visual-words (BoVW) [7]. BoVW quantises the order-less local features to visual-words and represents image as frequency histograms of visual-words. As BoVW transforms the set of local features to a single vector, image retrieval can be performed in VSM and the retrieval process can be greatly accelerated by the inverted files technique [7].

The retrieval accuracy based on BoVW can be further improved by calculating variable weights for different visual-words. Various weighting schemes have been studied while applying BoVW to image retrieval. One well-known scheme is based on term frequency-inverse document frequency (TF-IDF), which describes the importance of visual-words in both single image and image collections. As visual-words are generated without the specific semantics as words, directly applying the weighting schemes for words in text retrieval may fail to capture the characteristics of image retrieval [8].

This paper presents a classification-driven weighting scheme (CWS) for BoVW-based CBIR system. The method contains two major components, a query-category mapping table and a category-weight mapping table, both of which are learned on randomly selected images. Linear classifiers are trained to build the query-category mapping table, which establishes the corresponding relation between

images and image classes. The weights of visual-words for each image class are learned to build category-weight mapping table. In online image retrieval, weights for query image are determined after two-stage mapping. Based on the calculated weights, query-specific visual-words can be captured and more similar images can be ranked at the top positions in the returned retrieval results.

The rest of the paper is organised as follows: related work is reviewed in Section 2. Section 3 introduces the basis of BoVW and the CWS-based CBIR system. Section 4 presents our classification-driven weighting scheme. Experimental results are presented in Section 5. Section 6 concludes the paper.

2 Related work

BoVW and its variants [7, 9–16] have been widely used in the CBIR systems. The BoVW-based CBIR system was originally reported in [7]. When query image is uploaded to retrieval system, local features are extracted and assigned to its nearest visual-words by vector quantisation. The image is described as BoVW feature vector, where each dimension represents the occurrence frequency of one visual-word. In the past literature, numerous approaches are proposed to improve the performance of original BoVW image representation. To address the efficiency of BoVW extraction, vocabulary tree [9] and w-tree [10] are adopted to reduce the time cost of assigning local features to the corresponding visual-words. To speedup the retrieval process of BoVW-based CBIR, hashing structure [11] and semantic indexing [12] are designed to decompose the retrieval space and prune many unnecessary computations. Moreover, soft quantisation, instead of the original hard one, is adopted in [13, 14] to assign local feature to multiple visual-words, to preserve more information during vector quantisation. In [15, 16], further propose to encode the spatial information in BoVW to obtain more discriminative image representation.

For a particular query image, some visual-words are more informative, and thus they are more discriminative for the later tasks (e.g. image retrieval). Therefore designing a weighting scheme that assigns more weights to the informative visual-words can boost the retrieval performance. One well-known weighting scheme is based on the term frequency-inverse document frequency (TF-IDF), which is originally used for text retrieval to describe the importance of words. Tirilly *et al.* [8] reviews the weighting schemes, which exploits how the usage of weighting schemes developed for text retrieval can improve the performance of image retrieval. However, the improvements in performance, as concluded in [8], can only be observed on specific image datasets. The main reason is that these weighting schemes consider the CBIR process as a classical text retrieval problem, without any adaptations to the characteristics of this context.

Other approach contains fuzzy weighting scheme (FWS) described in [17], which considers the distinctive importance between textual words and visual-words. Chen *et al*. [18] develops a spatial weighting scheme (SWS), which uses Gaussian mixture model (GMM) to encode spatial structure while extracting BoVW. Elsayad *et al.* [19] further presents a new SWS (NSWS) to weight the visual-words according to the spatial constitution of an image. The difference between these two approaches is the way to use the probability of each visual-word belonging to each of the Gaussian. However, a lot of these weighting schemes do not consider the discriminative information of visual-words that could capture the distinctive visual-words for a particular query.

Relevance feedback-based weighting scheme (RFWS), which is similar to our work, can mining the discriminative information of visual-words and capture the query-specific weights [20–23]. In traditional CBIR, RFWS uses relevance feedback to give more weights to feature dimensions that can better reflect the visual similarity. The main principle of these approaches is to learn query-specific feature weights in the framework of online learning that keeps a distance margin between the manually selected positive and negative features. However, as the dimension of BoVW is generally high, several rounds of feedback are needed to mining weights of visual-words, which make the process too time-inefficient to meet the practical real-time requirements. In order to reduce the time consumed by RFWS, CWS is proposed to embed the time-consuming weight learning into offline process. Two-stage mapping are designed to map the query image to the pre-learned weights quickly.

The most relevant work to our approach is [24], where the authors propose a framework that uses linear classifier for similarity measurement while retrieving images. Coarse feature weights are defined for each image class manually. However, this trick of weight setting cannot be applied to BoVW. There are two main reasons. Firstly, as the dimension of BoVW feature is generally high, brutally defining the weights with manual power for each feature dimension is almost impossible. Secondly, as semantic of each visual-word is unclear, the significance of each visual-word is hard to be determined manually too. Different from [24], in this work, we propose to learn the weights of visual-words automatically in offline process by the max-margin learning, whose objective is to make the distance between images from same classes closer, and meanwhile keep the distance between images from different classes further. Max-margin learning can describe the importance of visual-words with more accuracy, and thus improve the discriminative ability of BoVW.

3 BoVW and CWS-based CBIR system

In this section, we first give an overview of the BoVW method, and then introduce the proposed CWS-based CBIR system.

3.1 BoVW extraction

BoVW is proposed to convert local feature based image representation to frequency histograms of visual-words, where each bin denotes the frequency of visual-word. Its extraction process usually contains three procedures: local feature extraction, visual-words construction and BoVW generation.

Step 1: Local feature extraction. In this paper, densely sampling strategy is adopted to detect the interest points and scale invariant feature transform (SIFT) [25] is chosen as the local descriptor. Each interest point is represented by a vector of 128 dimensions which summarise edge information in image patch centered at the interest point.
Step 2: Visual-words construction. The extracted local features are then partitioned into M clusters via clustering. The generated cluster centres $V = [v_1, v_2, \ldots, v_M]$ are visual-words, where M is the number of visual-words.

doi: 10.1049/iet-ipr.2013.0375

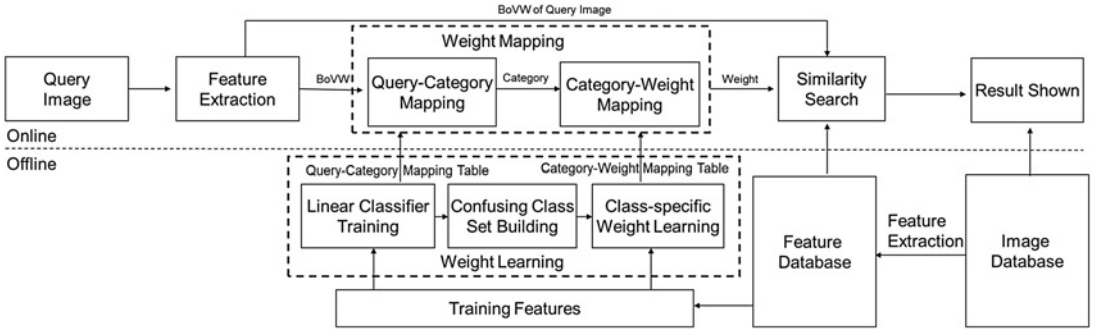

Fig. 1 *Block diagram of CWS-based CBIR system*

Step 3: BoVW generation. BoVW is extracted based on the visual-words. Each local feature is assigned to its nearest visual-word. Occurrence frequency of each visual-word is then accumulated and the image I is represented as $f_I = [f_1, f_2, \ldots, f_M]$, where f_i denotes occurrence frequency of the i_{th} visual-word.

3.2 CBIR system based on CWS

In this paper, CWS is embedded in CBIR system to utilise the discriminative information of visual-words for query image. The block diagram of CWS-based CBIR system is shown in Fig. 1. CWS is comprised of two parts: weight learning and weight mapping. Weight learning is an offline process to build two mapping tables, whereas weight mapping is designed as an online process to map the query image to its optimal weights of visual-words. In weight learning, to learn the weights of visual-words, typical image classes of the whole image database are first determined via visual clustering. Representative images for each image class are then randomly selected to construct the learning dataset, on which BoVW features are extracted and linear classifiers are trained. With the trained linear classifiers, query-category mapping table is built to establish the corresponding relationship between image and image class. In CWS, weights of visual-words are learned for each image class via max-margin learning. With the learned weights, category-weight mapping table is built to establish the corresponding relationship between image class and weights of visual-words. To this end, two mapping tables have been built and weight learning process is completed.

In online retrieval of CWS-based CBIR system, instead of performing direct similarity matching as general CBIR system, we first calculate the image category of query image via query-category mapping. Then, weights of visual-words are determined according to the obtained image category of query and category-weight mapping table. Finally, the weights are used in similarity search to return similar images.

4 Classification-driven weighting scheme

Weights of visual-words calculated from weighting scheme are used to represent the importance of visual-words. The main principle of weighting scheme is that visual-words with more importance should be assigned with larger weights, and visual-words with less importance should be assigned with smaller weights. However, the available weighting schemes are heuristic and fail to measure the importance of visual-words. In order to measure the importance of visual-words with more accuracy, we propose the weighting scheme CWS by combining offline weight learning and online weight mapping. Weight learning is designed to learn the weights of visual-words and build two mapping tables. Weight mapping is used to generate weights of visual-words for the query image during online retrieval.

4.1 Weight learning

The procedure of weight learning contains three major steps. First, linear classifiers are trained to establish the query-category mapping table, which is used to map query image to the image class with the maximum probability. Second, confusing class set is constructed for each class to exclude irrelevant classes and reduce the computational pressure in process of weight learning. Last, class-specific weight learning is used to build the category-weights mapping table, which is used to map image class to its class-specific weight.

4.1.1 Linear classifier training: The task of classifiers is to predict the image category of query with the model learned on training images. In CWS, linear support vector machine (LinearSVM) [26] is chosen as the classifier model since it has empirical success and high time efficiency. However, original BoVW features are linearly inseparable, and linear SVM cannot be trained on them directly. To solve it, explicitly mapping (proposed in [27]) is utilised to map the features into high-dimensional and linearly separable features. With this optimisation, linear classifiers can be trained on these mapped features to learn the optimal hyper-plane. The time complexity of linear classification is $O(1)$, which has almost no influence on the efficiency of online retrieval.

As there are multiple image classes in image database, multiclass classifiers should be built. The dominant method for multiclass classifiers training is reducing the multiclass problem into multiple binary classification problems. One-against-all and one-against-one are two common approaches designed for this reduction. For one-against-all approach, a new image is classified by a winner-takes-all strategy. Assuming C image classes are in image database, C binary classifiers should be built and the classifier with the highest output assigns the class for query image. Therefore the time complexity to classify image using one-against-all approach is $O(C)$. For one-against-one approach, image is classified by a max-wins voting strategy. $C(C-1)/2$ binary classifiers should be built and the classification of image is determined as the class which

wins the most votes. The time complexity of classification process using one-against-one approach is $O(C^2)$.

In CWS, classifiers are embedded in process of online retrieval to obtain the category of newly uploaded query image. Therefore, to speed up the classification, one-against-all approach is used for its low time complexity. In the following, we introduce the detail steps of linear classifier training using one-against-all approach.

Step 1: C typical semantic classes are determined by clustering images, which are sampled from the whole database, into C clusters. N images for each image class are randomly selected as training images.
Step 2: BoVW features are extracted and training images can be denoted as $\{f_n, y_n\}_{n=1}^{N}$, $f_n \in R^{M\times 1}$, $y_n \in \{1, 2, \ldots, C\}$, where f_n, y_n denotes the extracted BoVW feature and class label of the n_{th} image, respectively.
Step 3: In order to make the feature be linearly classified, BoVW is mapped into high-dimensional space with explicitly mapping function φ, and the features of training images are re-formulated as $\{\varphi(f_n), y_n\}_{n=1}^{N}$, $\varphi(f_n) \in R^{M'\times 1}$, $y_n \in \{1, 2, \ldots, C\}$, where M' is the dimension of mapped features.
Step 4: After explicitly mapping, parameters of C binary linear SVMs are trained on mapped features by using one-against-all approach. In detail, the parameters are obtained by solving the following optimisation formula

$$\arg\min_{v_c, b_c}\left\{\sum_{n=1}^{N}\frac{\lambda}{2}||v_c||^2 + \max\{0, 1 - y_n(v_c^{\mathrm{T}}\varphi(f_n) + b_c)\}\right\}$$
$$v_c \in R^{M'\times 1},\ b_c \in R,\ \lambda > 0 \tag{1}$$

where v_c and b_c are the soft margin parameter and bias multiplier of the c_{th} linear SVM, respectively, λ (non-negative scalar) is a regularisation parameter. With these learned parameters, the output of query image q in the c_{th} binary classifiers Classifier^c are obtained as

$$\text{Classifier}^c(q) = v_c^{\mathrm{T}}\phi\left(f_q\right) + b_c, \quad c = 1, 2, \ldots, C \tag{2}$$

Step 5: In CWS, the learned linear classifiers can be considered as the form of query-category mapping table. Once query image is uploaded, BoVW is extracted and mapped into high-dimensional feature space. The mapped features are fed into pre-learned linear classifiers, and category of query image q is determined according to the output confidence score of classification model.

For CBIR, there may exist a case that user's interested query is different from the learned classes. To solve it, we compare the output confidence score of linear SVM to 0 (decision boundary in SVM) to determine the query intention. If the confidence score is above 0, the class label of query image is obtained with much confidence. We regard that the query interest can be categorised into the learned classes with high probability. In contrast, if the confidence score is below 0, the class label of query image is determined with much uncertain. In this context, we identify that the query interest is outside of the learned classes, and then assign a class label $C+1$ for this type of query image for convenience. Formally, the possible image class of query image is given by (see (3))

One should note that this approach is similar to SVM that judges the classification of query image to be uncertain if the output confidence score is still lower than decision boundary. In CWS, we apply the same idea to determine to the degree of certainty of the query-category mapping process. Query image classified with the much uncertainty is categorised into the class labelled with $C+1$. For class $C+1$, weights of visual-words can be determined by other available weighting schemes.

4.1.2 Confusing class set building:

In our approach, weights of visual-words are learned to separate classes which are semantically confusing. Intuitively, the weights of visual-words, which are competent for separating the classes with the more semantic ambiguities, can separate the classes with less semantic ambiguities. In CWS, confusion degree is chosen as the measure metric to determine whether two classes are semantically confused or not. Image classes, whose confusion degree exceeds a certain threshold, are considered as semantically confused classes. In implementation, confusing class set for each semantic class can be built from the confusing matrix which is calculated from classification results. In the following, we introduce an example procedure to construct the confusing class set for particular image class c.

Step 1: We first calculate the confusion degree $P(c, d)$ between class c and d, its formula is

$$P(c, d) = \frac{\sum_{n=1}^{N}\delta(\delta(c, y_n),\ \delta(d,\ \text{Category}(f_n)))}{\sum_{n=1}^{N}\delta(c, y_n)}\ c \neq d$$
$$c,d \in \{1, 2, \ldots, C\} \tag{4}$$

where Category(f_n) is the class label of the n_{th} image determined by linear classifiers. δ is the delta function that is calculated as

$$\delta(\text{a, b}) = \begin{cases} 1 & \text{a} = \text{b} \\ 0 & \text{other else} \end{cases} \tag{5}$$

Step 2: A threshold ζ_c is set for class c to filter out image classes with low confusion degree. Image class with the confusion degree that is above the threshold is considered as confusing class. On the contrary, classes below the threshold are considered semantically irrelevant and the corresponding class should not be included in the confusing set.
Step 3: We define the $U(c)$ as the confusing classes set of class c and it can be built as

$$U(c) = \{d|P(c, d) \geq \zeta_c,\ d \in \{1, 2, \ldots, C\}, \zeta_c < 1\} \tag{6}$$

$$\text{Category}(q) = \begin{cases} \arg\max\limits_{c\in\{1,2,\ldots,C\}} \text{Classifier}^c\left(\varphi\left(f_q\right)\right), & \max\limits_{c\in\{1,2,\ldots,C\}} \text{Classifier}^c\left(\varphi\left(f_q\right)\right) \geq 0 \\ C+1, \quad \text{other else} \end{cases} \tag{3}$$

doi: 10.1049/iet-ipr.2013.0375

There is a case that image class c has almost no confusions with other classes, which leads to empty $U(c)$. To avoid this pitfall, we calculate the similarities between class c and all other classes. The image class who has the highest similarity with image class c constitutes $U(c)$. In this context, $U(c)$ is given by

$$U(c) = \arg\min_{d} \left\{ \sum_{i=1}^{N} \min_{k} \{D(f_i, f_k)\} \right\}, \quad y_i = c\, y_k = d \quad (7)$$

where $D(f_i, f_k)$ is the distance between features f_i and f_k and it can be calculated as

$$D(f_i, f_k) = \sum_{m=1}^{M} (f_{im} - f_{km})^2 \quad (8)$$

One should note that appropriate threshold of confusion degree for each image class is very essential to the effectiveness and efficiency of the weight learning. If the threshold is set to be low, possible confusing image classes will be included from weighting learning as much possible. In this context, the time cost of weight learning will increase, but more discriminative information can be embedded in learned weights. On the contrary, if the threshold is set to be high, less confusing image classes may be included from weight learning, which may generate less discriminative weights.

4.1.3 Class-specific weight learning: We formulate the class-specific weight learning in a discriminative framework. The learning goal is to make distance of images from the same image class c closer, and meanwhile keep the distance of images from image class c and class in its corresponding confusing class set further. The graphic description of this principle is showed in Fig. 2.

The triplet index set of training image groups for class c is $G = \{(i, j, k) | y_i = y_j = c, y_i \neq y_k, i, j, k \in \{1, 2, \ldots, N\}$, where image i and j is from the same class c, and image k is from the class in $U(c)$. The raw distance of mth visual-words between image i and j is denoted as $D_{ij}(m)$ $m = 1, 2, \ldots, M$. Accordingly, $D_{ij} = \{D_{ij}(1), D_{ij}(2), \ldots, D_{ij}(M)\}$ is the distance vector between them. We assign weight vector w for visual-words and the weighted distance is represented as $D_w(I_i, I_j)$, which can be calculated as $D_w(I_i, I_j) = w^{\mathrm{T}} D_{ij}$.

The objective of the optimisation is to make the distance of the different labelled images larger than the same labelled images. Formulary, it can be denoted as $D_w(I_i, I_k) > D_w(I_i, I_j)$ or $w^{\mathrm{T}} D_{ik} > w^{\mathrm{T}} D_{ij}$. The whole weight learning for class y_i can be formulated as an optimisation problem, the objective function is

$$\arg\min_{w_{y_i}} \frac{\beta}{2} ||w_{y_i}||^2 + \sum\nolimits_{ijk} \xi_{ijk} \quad (9)$$

where w_{y_i} is weights of visual-words for class y_i, the first item $\frac{\beta}{2} \left\| w_{y_i} \right\|^2$ is regulation term, slack variables $\xi_{ijk} \geq 0$ are added to avoid over-fitting and $\sum_{ijk} \xi_{ijk}$ is the sum of empirical loss. β is the regulation parameter to keep a trade-off between the empirical loss and regularisation (β is set to 0.33 in our approach to maximise the performance). Our formulated objective is to minimise the sum of the empirical loss and regularisation term while subjecting to a series of conditions

$$\begin{aligned} & w_{y_i}^{\mathrm{T}} D_{ik} - w_{y_i}^{\mathrm{T}} D_{ij} \geq 1 - \xi_{ijk} \\ & w_{y_i} \geq 0, \quad \xi_{ijk} \geq 0, \quad \forall (i, j, k) \in G \end{aligned} \quad (10)$$

In our approach, stochastic gradient descent (SGD) is adopted to solve this optimisation problem. Since SGD does not need to remember which examples are visited during the previous iteration, it directly optimises the objective function where examples are randomly drawn from the ground truth distribution [28, 29]. For computation convenience, the objective function H is reformulated as

$$\begin{aligned} H &= \sum_{ijk} H\left(w_{y_i}, D_{ik} - D_{ij}\right) \\ &= \sum_{ijk} \frac{\beta}{2} ||w_{y_i}||^2 + \max\left[0, 1 - w_{y_i}^{\mathrm{T}}\left(D_{ik} - D_{ij}\right)\right] \end{aligned} \quad (11)$$

The gradient of w_{y_i} is

$$\begin{aligned} & \nabla_{w_{y_i}} L\left(w_{y_i}, D_{ik} - D_{ij}\right) \\ & = \begin{cases} \beta w_{y_i} - \left(D_{ik} - D_{ij}\right), & w_{y_i}^{\mathrm{T}}(D_{ik} - D_{ij}) < 1 \\ \beta w_{y_i}, \text{ other else} \end{cases} \end{aligned} \quad (12)$$

The update rule for w_{y_i} is (η is step size and t is the number of iteration)

$$w_{y_i}^{t} = \begin{cases} (1 - \beta\eta) w_{y_i}^{t-1} + \eta\left(D_{ik} - D_{ij}\right), w_{y_i}^{\mathrm{T}}\left(D_{ik} - D_{ij}\right) < 1 \\ (1 - \beta\eta) w_{y_i}^{t-1}, \text{ other else} \end{cases} \quad (13)$$

With the gradient computed, weights of visual-words are updated step-by-step along the direction of gradient descent until the optimal conditions are achieved. To this end, weights of visual-words for each image class are learned. Category-weight mapping table can be built with the learned weights.

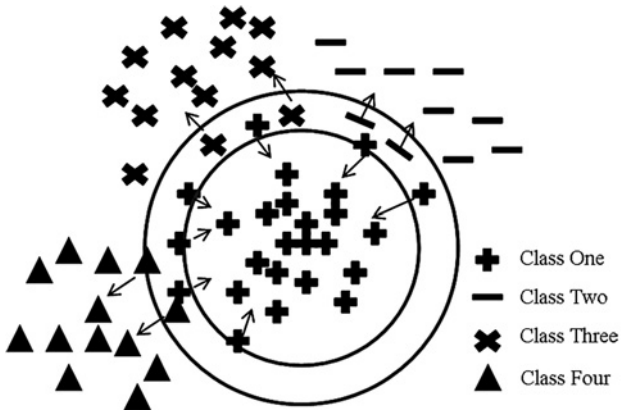

Fig. 2 *Weight learning for visual-words of class one (class two, class three, class four are in the confusing class set of class one)*

4.2 Weight mapping

With the two mapping tables built from weight learning, weights of visual-words for query image are calculated

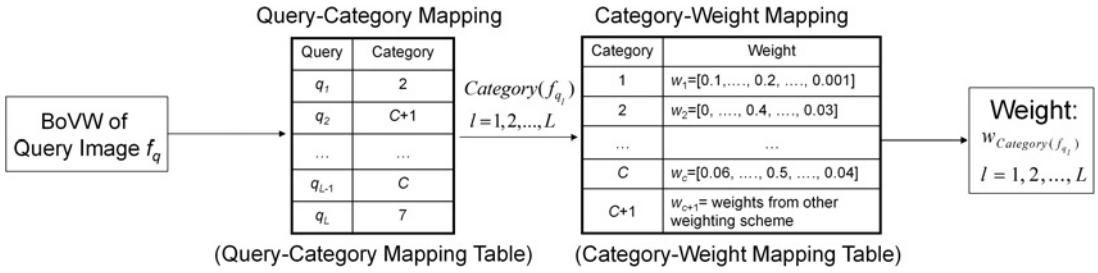

Fig. 3 *Two-stage weight mapping process (L is the number of query images)*

through two-stage mapping in the process of online retrieval. Fig. 3 shows the two-stage mapping process in CWS. The first-mapping stage is query-category mapping that maps query image to its most likely class. In this mapping process, BoVW of query image is first extracted, and then is fed to the pre-trained linear classifiers. Class of query is finally obtained according to classification scores. The second-mapping stage is category-weight mapping that maps class of query image to weights of visual-words, which can be obtained by looking into the category-weight mapping table according to the possible class label of query. After this two-stage mapping, query-specific weights are obtained and can be utilised directly in similarity search to return the most similar images.

5 Experiments and results

In this section, we apply the proposed method on image retrieval tasks and demonstrate its superiorities by comparing it to other methods, under both unsupervised and supervised settings. First, we introduce our test dataset and experimental setup. Second, we observe the variations of performance with the parameters in CWS. Third, we conduct comparative experiments to demonstrate the effectiveness of our approach by comparing it with the current best weighting schemes. Finally, we carry out experiments to evaluate the robustness of the proposed weighting scheme.

5.1 Dataset and experimental setup

The performance of the proposed method is evaluated on WANG database [30] and Caltech 101 [31]. WANG database contains 10 image categories with 100 images in each category. Caltech 101 contains 8197 images in 101 categories which have huge variances in shape, colour and texture. The number of images in each class varies from 31 to 800. In the following experiments, unless specifically noted, 10 images for each class are randomly selected for training and the remaining images are utilised as query images. We choose the 10 as the number of training images so that more number of query images can be obtained.

The performance of retrieval system is evaluated using the standard metric in terms of precision and recall [32]. Images in the same class are considered as relevant images. Precision is defined as the ratio between the number of retrieved relevant images and the total number of retrieved images. Recall is defined as the ratio between the number of retrieved relevant images and the number of relevant images in database. For direct comparisons, mean average precision (MAP) is calculated by averaging the precisions on all recall levels.

In the experiments, SIFT descriptors are extracted from local patches of 16×16 pixels, which are centred on the interest points densely sampled on a grid with step size of 3 pixels. The dimension of mapped feature is three times of BoVW ($M' = 3 \times M$). To train the linear classifiers, Pegasos SVM [33] is utilised as our SVM solver for its high efficiency. Its soft margin parameter and bias multiplier is set to 3 and 1, respectively, to maximise the retrieval performance.

All our experiments have been run on the platform equipped with an Intel Core i7 920 CPU running at 2.67 GHz. The operating system is 64-bit RHEL AS 5.4 with Linux kernel 2.6.18.

5.2 Evaluation for the parameters in CWS

In the following subsections, we observe the variations of retrieval performance with three parameters: the size of visual-words M, the threshold of confusion degree and the number of training images N.

5.2.1 Evaluation for the size of visual-words M: In experiment, SIFT is employed as the local descriptor and the k-means is used to generate visual-words. In this section, empirical evaluation is investigated to show the variations of retrieval performance with the size of visual-words M. For each M, the retrieval performance of CWS and BoVW is reported.

Fig. 4 shows the MAP of the retrieval system on two dataset when the size of visual-words changes from 100 to 1000. The figure shows that MAP varies with the size of visual-words. Also, we can see CWS outperforms BoVW on all values of M. The difference of MAP on the two datasets is roughly 20%. On both datasets, the best

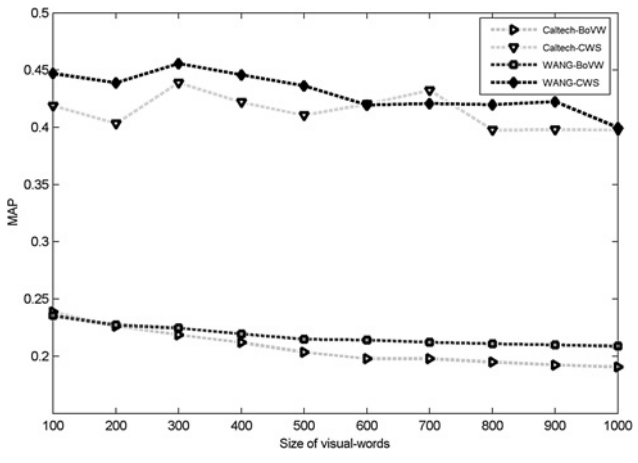

Fig. 4 *Evaluation for the size of visual-words M*

doi: 10.1049/iet-ipr.2013.0375

Table 1 Retrieval performance variations when different thresholds of confusion degree are set (time taken in seconds)

Dataset	WANG database						
threshold	−0.1	0	0.1	0.2	0.3	0.4	0.5
learning time	742.5	255.1	58.6	53.7	48.8	48.8	48.8
MAP	0.465	0.454	0.433	0.423	0.414	0.414	0.414
Dataset	Caltech						
threshold	−0.1	0	0.1	0.2	0.3	0.4	0.5
learning time	22542.5	5902.2	1180.4	1157.4	1157.4	1157.4	1157.4
MAP	0.453	0.441	0.405	0.392	0.392	0.392	0.392

Table 2 Retrieval performance variations with the number of training images

Dataset	WANG database								
N	5	10	15	20	25	30	35	40	45
MAP	0.422	0.464	0.471	0.478	0.480	0.486	0.491	0.496	0.512
Dataset	Caltech								
N	5	10	15	20	25	30	N/A	N/A	N/A
MAP	0.402	0.468	0.471	0.478	0.480	0.486	N/A	N/A	N/A

performance of retrieval is achieved when the size of visual-words is 300. The highest MAP of CWS is 45.6% on WANG database, and 43.9% on Caltech, respectively. These experiments confirm our analysis that the retrieval performance of CBIR can be improved by capturing the weights of visual-words.

5.2.2 Evaluation for the threshold of confusion degree: In this section, we investigate the influence of different thresholds of confusion degree to the retrieval performance. Threshold of confusion degree is set for each image class in process of confusing class set building to filter image classes with less confusion.

Table 1 reports retrieval performance on two dataset when different thresholds are set. The threshold is set to negative value (−0.1) means that no image class is filtered and all classes are included for weight learning. The table shows that, when lower threshold is set, the weight learning consumes more time, but improves the retrieval accuracy. The main reason is that more images are included for learning when the threshold is low. On both dataset, when the threshold is 0, we obtain a small accuracy loss compared with the one achieved by including all the remaining image classes for learning. Another experimental phenomenon we can observe on this threshold is that the process of weight learning consume less time. Therefore, in the following experiments, for both datasets, confusion degree is set to 0 by considering the efficiency and effectiveness.

We can also observe from the table that, on WANG database, both the learning time and the retrieval accuracy becomes unchanged when threshold is more than 0.3. This phenomenon can also be observed on Caltech when the threshold is above 0.2. The main reason to explain this experimental phenomenon is that, when the threshold is above certain value, their confusing class set is only comprised of the class who has the highest similarity with it.

5.2.3 Evaluation for the number of training images: In this section, we present the results of an experiment to investigate the variations of retrieval performance with the number of training images *N*. In CWS, training images are used to learn the linear classifiers and weights of visual-words. We claim that learning on more training images can embed more discriminative information in weights.

In this experiment, we rebuild the image dataset to verify our claim. For WANG database, a half of images in each class (50 images for each) are used as training images. For Caltech dataset, the number of training images is set to 30 (considering the characteristics of this dataset). For both dataset, the remaining images are used as the queries.

We run the experiments to observe the variations of retrieval performance when the number of training images is changed. Table 2 summarises the main retrieval accuracy on two dataset. It is shown that, on both dataset, the retrieval accuracy increases steadily with the number of training images. These results demonstrate that learning on more training images can generate more discriminative weights, and naturally improve the retrieval accuracy.

5.3 Comparison with other weighting schemes

In this section, we conduct experiments to compare CWS with other weighting schemes, under both unsupervised and supervised settings.

Table 3 Unsupervised weighting schemes in comparative experiments

Weighting schemes	Abbreviation	Best size of visual-words
classification-driven weighting scheme [proposed]	CWS	WANG: 300 Caltech: 300
fuzzy weighting scheme [17]	FWS	WANG: 300 Caltech: 400
spatial weighting scheme [18]	SWS	WANG: 400 Caltech: 500
new spatial weighting scheme [19]	NSWS	WANG: 400 Caltech: 500
term frequency-inverse document frequency [7]	TF-IDF	WANG: 300 Caltech: 300

5.3.1 Comparison with unsupervised weighting schemes: Table 3 summarises the weighting schemes, their abbreviations and best size of visual-words. In experiments, degree of fuzziness is set to 1 to maximise the retrieval performance of FWS-based CBIR. For GMM modelling in spatial weighting, five-dimensional (5D) feature (3D LUV colour feature and 2D position coordinate) is used to represent each pixel.

Figs. 5 and 6 describe the precision-recall curves of all weighting schemes. The figures show that CWS outperforms all other unsupervised weighting schemes. In addition, we discover an interesting phenomenon that these unsupervised weighting schemes obtain different performance on two image datasets. FWS achieves the best performance on WANG database, whereas NSWS achieves the best performance on Caltech. MAP of CWS outperforms FWS about 13.7% on WANG database, whereas MAP of CWS improves NSWS about 9.4% on Caltech. The detailed comparison of MAP is shown in Fig. 7. The concrete values of MAP are shown in Table 4.

To test the time efficiency of CWS, the average response time (ART) of these weighting schemes is compared on two datasets. In our experiments, ART is recorded from the time when query image is uploaded to the time when similar images are returned. Fig. 8 shows the graphic description of ART and Table 4 describes the concrete values of ART. As shown in Fig. 8, CWS, FWS, SWS, NSWS and TF-IDF cost almost the same response time. CWS prolongs the process of image retrieval slightly (<0.01 s). The reason to account for this is that CWS embeds an additional classification process for query image in process of online retrieval. This experimental phenomenon demonstrates that the additionally linear classification has almost no impact on the whole retrieval.

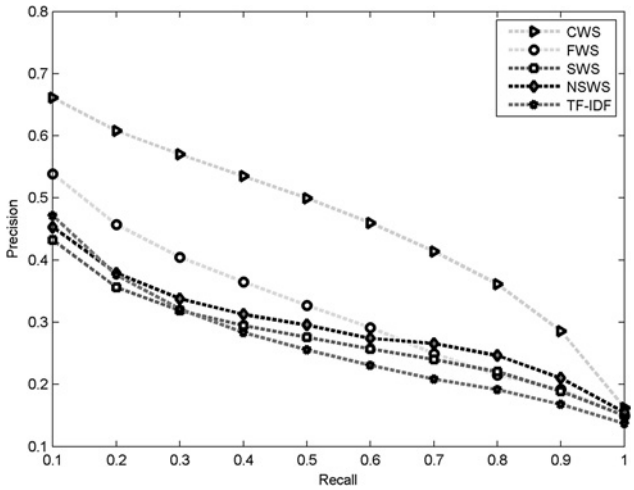

Fig. 5 *Precision-recall curve of weighting schemes on WANG*

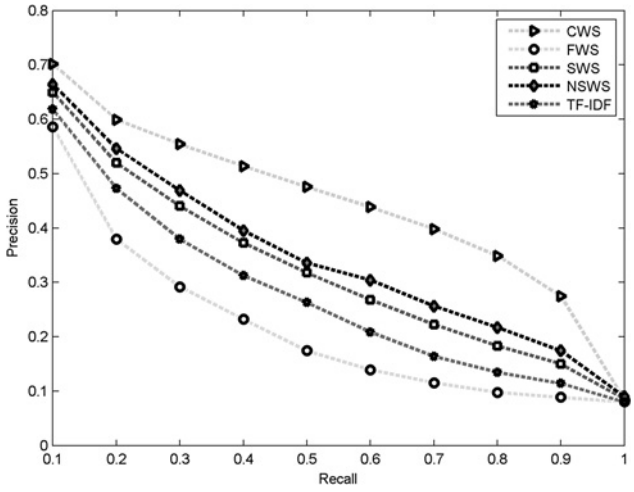

Fig. 6 *Precision-recall curve of weighting schemes on Caltech 101*

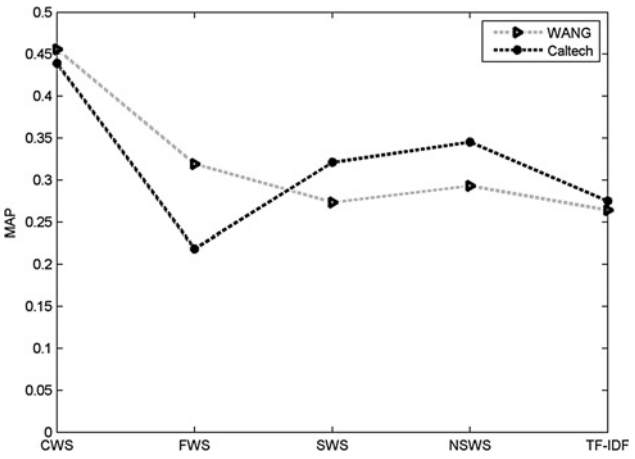

Fig. 7 *MAP comparison on WANG database and Caltech 101*

Table 4 MAP and ART (time taken in seconds) of weighting scheme on WANG database and Caltech 101

Weights schemes	CWS	FWS	SWS	NSWS	TF-IDF
MAP (WANG)	0.456	0.319	0.273	0.293	0.264
MAP (Caltech)	0.439	0.218	0.321	0.345	0.275
ART (WANG)	0.0206	0.0184	0.0185	0.0186	0.0187
ART (Caltech)	0.0406	0.0324	0.0330	0.0330	0.0329

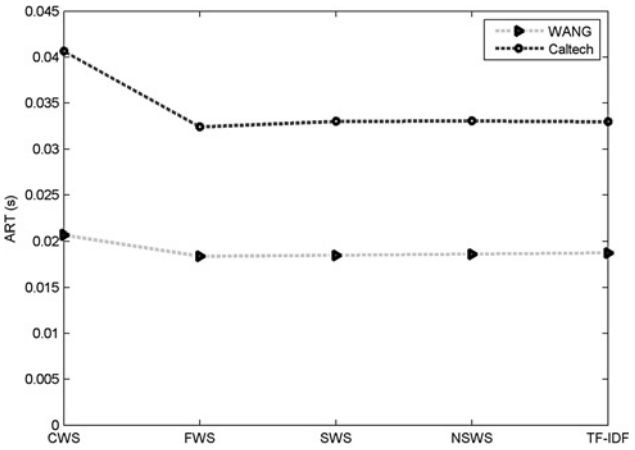

Fig. 8 *Evaluation for ART on WANG database and Caltech 101*

5.3.2 Comparison with RFWS: Recent image retrieval studies also rely on the usage of RFWS to learn the weights of visual-words in CBIR. RFWS consists in modifying the weights of visual-words in online retrieval process by learning the weights on user feedbacks. Therefore RFWS can be considered as a supervised weighting scheme. In our

doi: 10.1049/iet-ipr.2013.0375

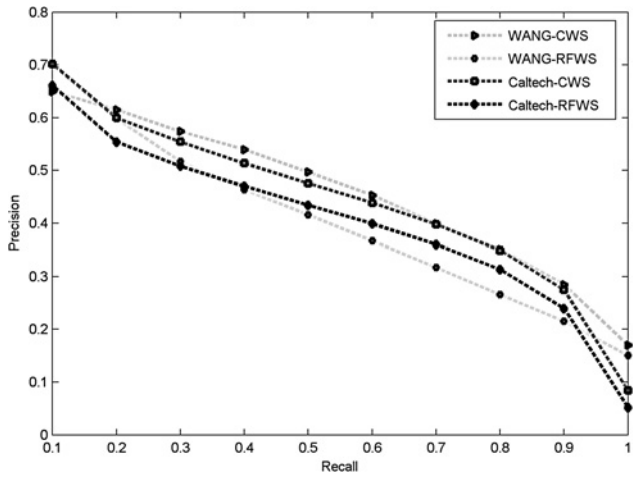

Fig. 9 *Retrieval performance comparison with RFWS*

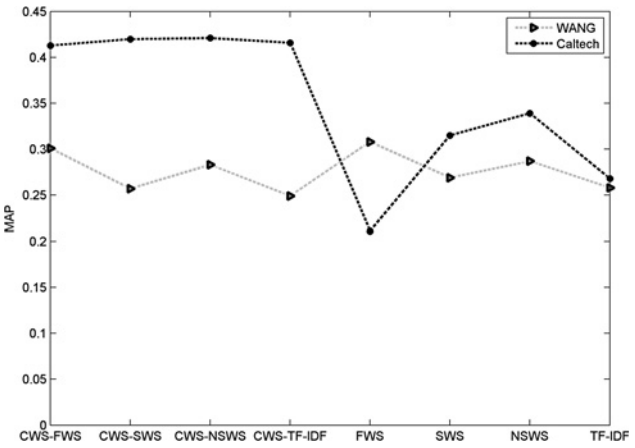

Fig. 10 *Retrieval performance of different query types*

experiments, max-margin model is also utilised to formulate the weight learning in RFWS. To guarantee the fairness of comparison, ten positive images (visual similar with query image) and ten negative images (visual dissimilar with query image) are selected to construct the temporary image dataset for weight learning.

Fig. 9 shows the detail comparison of retrieval performance between CWS and RFWS on two datasets. The figure shows that CWS outperforms RFWS on most of the recall levels. The MAP of CWS outperforms RFWS. The difference of MAP is roughly 5% on WANG database, and 4% on Caltech. The MAP is improved by CWS since the offline learning of weights can involve more discriminative information. More importantly, it should take several rounds for RFWS to select images and learn the weights of visual-words. However, for CWS, it just cost the same time with the original BoVW. Obviously, as weight learning is designed in offline process, CWS can obtain great improvement on time efficiency than RFWS. Depending on the application context of CBIR, computational time has to be carefully considered when a new weighting scheme is designed. So from this point of view, CWS is more advantageous than RFWS.

5.4 Robustness experiments

In this section, experiments are carried out to evaluate the robustness of the proposed CWS in the scenario that user query interest is different from the learned concepts. In CWS, output confidence score of linear classifier is compared with determine whether query interest fails within our learned concepts or not. With this judgment process, different weights are mapped for query image.

To simulate this practical scenario, we build a new image dataset named WANG-Caltech by mixing the query images from WANG database and Caltech 101. Similarly, training images from each class in Caltech are used to train the weights, and all the remaining images are chosen as queries. Obviously, the query interest of images from WANG database is not within the learned concepts in Caltech. In this experiment, FWS, SWS, NSWS and TF-IDF are chosen as the available weighting schemes for queries whose interest is not with the learned concepts. For presentation convenience, we rename this weighting scheme as CWS–FWS, CWS–SWS, CWS–NSWS and CWS–TF-IDF respectively.

With the prepared dataset, we run experiments to investigate the variations of retrieval performance of different query types when different weighting schemes are used. Fig. 10 summarises the results for different weighting schemes. For query images from WANG database, we observe a little accuracy decrease compared with theses unsupervised weighting schemes. The performance decrease is attributed to the mistakes the system made when determining the query interests. Also, it can be easily observed that, for queries from Caltech, CWS still outperform the other weighting scheme. The main reason is that the weights generate by correct categorisation of query image rank more similar images at the top positions. These experimental results clearly demonstrate that the proposed weighting scheme is still robust even when the interest of user query is outside of learned concepts. Note that, in practice, interest of most query images will fall into the learned concepts in image database, and our weighting scheme can still achieve superior performance.

6 Conclusions

Although BoVW has achieved great successes in CBIR, the retrieval performance of BoVW-based image representation can be further improved if the weights of visual-words are properly calculated. This paper proposes a classification-driven weighting scheme which computes the weights of visual-words with the combination of offline weight learning and online weight mapping. Query-category mapping table is designed to predict the semantic class of query image, whereas the category-weight mapping table learned in offline process is developed to obtain the weights of visual-words. Weights of visual-words are learned to involve discriminative information that is lost in traditional weighting schemes. Linear classifier is utilised to drive the whole weighting scheme. Experimental results on both two datasets demonstrate that the better retrieval performance can be obtained by our weighting scheme. In the future, we will continue to check the consistence of these results when different interest point detectors and local descriptors are adopted.

7 Acknowledgments

The authors would like to thank Liang Xie and Xiaobai Liu for their work on the revised manuscript. The authors thank the anonymous reviewers for their helpful and valuable

suggestions. This work is supported by the National Natural Science Foundation of China (Grant no. 61133008) and the National High Technology Research and Development Program of China, under Grant 2012AA01A306.

8 References

1 Rui, Y., Huang, T.S., Chang, S.F.: 'Image retrieval: current techniques, promising directions, and open issues', *J. Vis. Commun. Image R.*, 1999, **10**, (1), pp. 39–62

2 Smeulders, A.W.M., Worring, M., Santimi, S., Gupta, A., Jain, R.: 'Content-based image retrieval at the end of the early years', *IEEE Trans. Pattern Anal. Mach. Intell.*, 2000, **22**, (12), pp. 1349–1380

3 Datta, R., Joshi, D., Li, J., Wang, J.Z.: 'Image retrieval: ideas, influences, and trends of the new age', *ACM Comput. Surv.*, 2008, **40**, (2), pp. 1–60

4 Deselaers, T., Keysers, D., Ney, H.: 'Features for image retrieval: an experimental comparison', *Inf. Retr.*, 2008, **11**, (2), pp. 77–107

5 Salton, G., Wong, A., Yang, C.S.: 'A vector space model for automatic indexing', *Commu. ACM*, 1975, **18**, (11), pp. 613–620

6 Zhang, J., Marszalek, M., Lazebnik, S., Schmid, C.: 'Local features and kernels for classification of texture and object categories: a comprehensive study', *Int. J. Comput. Vis.*, 2007, **73**, (2), pp. 213–238

7 Sivic, J., Zisserman, A.: 'Video Google: a text retrieval approach to object matching in videos'. Proc. Int. Conf. on Computer Vision, Nice, France, October 2003, pp. 1470–1477

8 Tirilly, P., Claveau, V., Gros, P.: 'Distance and weighting schemes for bag of visual words image retrieval'. Proc. Int. Conf. on Multimedia Information Retrieval, Philadelphia, America, March 2010, pp. 323–332

9 Nister, D., Stewenius, H.: 'Scalable recognition with a vocabulary tree'. Proc. Int. Conf. on Computer Vision and Pattern Recognition, New York, America, June 2006, pp. 2161–2168

10 Shi, M.J., Xu, R.X., Tao, D.C., Xu, C.: 'W-tree indexing for fast visual word generation', *IEEE Trans. Image Process.*, 2013, **22**, (3), pp. 1209–1222

11 Kong, W.H., Li, W.J., Guo, M.Y.: 'Manhattan hashing for large-scale image retrieval'. Proc. ACM SIGIR Conf. on Research and Development in Information Retrieval, Portland, America, August 2012, pp. 45–54

12 Deng, J., Berg, A.C., Li, F-F.: 'Hierarchical semantic indexing for large scale image retrieval'. Proc. Int. Conf. on Computer Vision and Pattern Recognition, Providence, America, June 2011, pp. 785–792

13 Liu, L.Q., Wang, L., Liu, X.W.: 'In defense of soft-assignment coding'. Proc. Int. Conf. on Computer Vision, Barcelona, Spain, November 2011, pp. 2486–2493

14 Ai, L.F., Yu, J.Q., Guan, T.: 'Spherical soft assignment: improving image representation in content-based image retrieval'. Proc. Int. Conf. on Advances in Multimeida Information Processing, Singapore, December 2012, pp. 801–810

15 Philbin, J., Chum, O., Isard, M., Sivic, J., Zisserman, A.: 'Object retrieval with large vocabularies and fast spatial matching'. Proc. Int. Conf. on Computer Vision and Pattern Recognition, Minneapolis, America, June 2007, pp. 1–8

16 Cao, Y., Wang, C.H., Li, Z.W., Zhang, L.Q.: 'Spatial-bag-of-features'. Proc. Int. Conf. on Computer Vision and Pattern Recognition, San Francisco America, June 2010, pp. 3352–3359

17 Bouachir, W., Kardouchi, M., Belacel, N.: 'Improving bag of visual words image retrieval: a fuzzy weighting scheme for efficient indexation'. Proc. Int. Conf. on Signal-Image Technology & Internet-Based Systems, Marrakesh, Morocco, November 2009, pp. 215–220

18 Chen, X., Hu, X.H., Shen, X.J.: 'Spatial weighting for bag-of-visual-words and its application in content-based image retrieval'. Proc. Int. Conf. on Advances in Knowledge Discovery and Data Mining, Bangkok, Thailand, April 2009, pp. 27–30

19 Elsayad, I., Martinet, J., Urruty, T., Djeraba, C.: 'A new spatial weighting scheme for bag-of-visual-words'. Proc. Int. Conf. on Content-Based Multimeida Indexing, Grenoble, France, June 2010, pp. 1–6

20 Das, G., Ray, S., Wilson, C.: 'Feature re-weighting in content-based image retrieval'. Proc. Int. Conf. on Image and Video Retrieval, Tempe, America, July 2006, pp. 193–200

21 Lee, R.S., Chung, C.W., Lee, S.L., Kim, S.H.: 'Confidence interval approach to feature re-weighting', *Multimed. Tools Appl.*, 2008, **40**, (3), pp. 385–407

22 Su, J.H., Huang, W.J., Yu, P.S., Tsent, V.S.: 'Efficient relevance feedback for content-based image retrieval by mining user navigation patterns', *IEEE Trans. Knowl. Data En.*, 2011, **23**, (3), pp. 360–372

23 Wang, X.Y., Zhang, B.B., Yang, H.Y.: 'Active SVM-based relevance feedback using multiple classifiers ensemble and features reweighting', *Eng. Appl. Artif. Intell.*, 2013, **26**, (1), pp. 640–646

24 Rahman, M.M., Antani, S.K., Thoma, G.R.: 'A learning-based similarity fusion and filtering approach for biomedical image retrieval using SVM classification and relevance feedback', *IEEE Trans. Inf. Technol. Biomed.*, 2011, **15**, (4), pp. 640–646

25 Lowe, D.G.: 'Distinctive image features from scale-invariant keypoints', *Int. J. Comput. Vis.*, 2004, **60**, (2), pp. 91–110

26 Fan, R.E., Chang, K.W., Hsieh, C.J., Wang, X.R., Lin, C.J.: 'LIBLINEAR: a library for large linear classification', *J. Mach. Learn. Res.*, 2008, **9**, pp. 1871–1874

27 Vedaldi, A., Zisserman, A.: 'Efficient additive kernels via explicit feature maps', *IEEE Trans. Pattern Anal. Mach. Intell.*, 2012, **34**, (3), pp. 480–492

28 Bordes, A., Bottou, L., Gallinari, P.: 'SGD-QN: careful quasi-newton stochastic gradient descent', *J. Mach. Learn. Res.*, 2009, **10**, pp. 1737–1754

29 Bottou, L.: 'Large-scale machine learning with stochastic gradient descent'. Proc. Int. Conf. on Computational Statistics, Paris, France, August 2010, pp. 177–187

30 Li, J., Wang, J.Z.: 'Automatic linguistic indexing of pictures by a statistical modeling approach', *IEEE Trans. Pattern Anal. Mach. Intell.*, 2003, **25**, (9), pp. 1075–1088

31 Li, F.-F., Fergus, R., Perona, P.: 'Learning generative visual models from few training examples: an incremental Bayesian approach tested on 101 object categories', *Comput. Vis. Image Underst.*, 2007, **106**, (1), pp. 59–70

32 Singha, M., Hemachandran, K., Paul, A.: 'Content-based image retrieval using the combination of the fast wavelet transformation and the colour histogram', *IET Image Process.*, 2012, **6**, (9), pp. 1221–1226

33 Shai, S.S., Singer, Y., Srebro, N., Cotter, A.: 'Pegasos: primal estimated sub-gradient solver for SVM', *Math. Program.*, 2011, **127**, (1), pp. 3–30

doi: 10.1049/iet-ipr.2013.0375