



Repositorio Institucional de la Universidad Autónoma de Madrid

<https://repositorio.uam.es>

Esta es la **versión de autor** de la comunicación de congreso publicada en:
This is an **author produced version** of a paper published in:

Adaptive Multimedia Retrieval. Understanding Media and Adapting to the User: 7th International Workshop, AMR 2009, Madrid, Spain, September 24-25, 2009, Revised Selected Papers. Lecture Notes in Computer Science, Volumen 6535. Springer, 2011. 87-100.

DOI: http://dx.doi.org/10.1007/978-3-642-18449-9_8

Copyright: © 2011 Springer-Verlag

El acceso a la versión del editor puede requerir la suscripción del recurso
Access to the published version may require subscription

Prosemanic features for content-based image retrieval

Gianluigi Ciocca¹, Claudio Cusano¹, Simone Santini² and Raimondo Schettini¹

¹ Università degli Studi di Milano-Bicocca,
Dipartimento di Informatica Sistemistica e Comunicazione,
viale Sarca 336, 20131 Milano, Italy

² Escuela Politécnica Superior,
Universidad Autónoma de Madrid,
C/ Tomas y Valiente 11, 28049 Madrid, Spain

Abstract. We present here, an image description approach based on prosemanic features. The images are represented by a set of low-level features related to their structure and color distribution. Those descriptions are fed to a battery of image classifiers trained to evaluate the membership of the images with respect to a set of 14 overlapping classes. Packing together the scores vectors of prosemanic features are obtained, and used to index the images in an image retrieval system. To verify the effectiveness of the approach, we designed a target search experiment in which both low-level and prosemanic features are embedded into a content-based image retrieval system exploiting relevance feedback. The experiments show that the use of prosemanic features allows for a more successful and quick retrieval of the query images.

1 Introduction

Many content based-retrieval systems have been proposed to manage and retrieve images on the basis of their content. Among the others we can cite [1–6]. A survey of some of the most important techniques used in Content-Based Image Retrieval (CBIR) systems can be found in [7]. To overcome the necessity of manually describing the images content, many of these systems are essentially based on low-level image features that are directly and automatically computed from the images themselves. However, the use of low-level features can't overcome the gap between the content and the semantic of the images. In order to cope with this problem and to provide satisfactory retrieval performance, new techniques are introduced in the retrieval process that take into account the subjectivity of human perception. One of these techniques is *relevance feedback* [8]. Relevance feedback is based on the interaction with the user who provides the system with examples of images relevant to the query. The system then refines its result depending on the selected images. The user's feedback provides a way to learn short term and case-specific query semantics. An example of this can be found in [9] where the system learns a non-linear embedding that maps clusters of images

into a hidden space of semantic attributes. Long term learning can be achieved by logging the previous user's interactions for further processing [10].

Other systems explicitly extract and embed in the retrieval process semantic information about the image content by exploiting automatic classification techniques [11]. These techniques can then be employed to automatically annotate the image content by keywords, which are then used in the retrieval process. If the underlying annotation is reliable, text-based image retrieval can be semantically more meaningful than other indexing approaches [10]. Concept detection techniques categorize images into general concepts such as city, landscape, sunset, forest, sea, etc. . . , via supervised classification [12, 13].

The annotation approaches described above can be considered as crisp annotation: if an image is annotated with a given label then the image expresses that concept or belong to that class. In [14] the authors tested two classification approaches, support vector machines (SVMs) and Bayes point machines (BPMs), to perform a soft image annotation. At the end of the annotation process, each image is annotated with a label vector, and a confidence factor is assigned to each label in the vector. These confidence factors can then be exploited in a text-based search where images are retrieved and ranked according to the confidence factors of the matching labels.

One of the first works that try to bring semantic information under the same model vector paradigm used in query-by-example systems is [15]. Semantic information is learned directly from the image content and forms a vector of semantic weights. Each weight is associated to a concept and is derived from the confidence score obtained by a support vector machine trained to recognize that concept. Retrieval in the semantic space corresponds to performing a similarity comparison between two model vectors using the L_2 measure. A similar approach is followed in [16].

With the exception of a few examples, all the above techniques tackle the problem of semantic image retrieval from the point of view of indexing, viz. they focus on the accuracy of the indexing scheme. Few have been used and evaluated in CBIR systems or tested on large image databases.

One of the first attempts to integrate and compare semantic keyword and low-level features into a single CBIR framework is the SIMPLIcity system [17]. The semantic classification is used to categorize images so that different semantically-adaptive search methods can be applied to each category. The system is also able to narrow down the subset of images to be searched by selecting those in the same category as the query. The reference categories chosen by the author are textured vs. non textured and graph-photograph. A more recent work [14] defines a new paradigm denoted as query-by-semantic-example (QBSE) that combines a query-by-example approach with semantic retrieval. Using the vector model to describe image content, the authors define a vector of semantic multinomial values, where each value is associated to a specific concept. They compared the QBSE and the query-by-visual-examples approaches in a CBIR system within a minimum probability error retrieval framework.

Following a similar paradigm, we designed an approach to CBIR based on the information provided by several image classifiers. One of the main problems in integrating automatic image classification into a content-based retrieval system is the choice of classes. It is very hard to identify a set of categories that are representative of the majority of the pictures and that can be used to reliably approximate their semantics. Moreover, state of the art image classification systems are far from perfect and, consequently, their use in image retrieval requires a high degree of tolerance with respect to misclassification errors.

To circumvent these problems, we did not exploit the classifiers to obtain a “crisp” semantic description of the images (e.g. “sunset on the beach”), but rather to provide a rich description of visual content that correlates low-level features to prototypical scenes (e.g. “image with an edge distribution that can easily be found in seaside scenes”). In our approach, this level of description is provided by a set of *prosemantic* features. These features are obtained by training several image classifiers so designed that their output can be interpreted as membership values of an image in the class that they embody. For each class, we trained multiple classifiers using different low-level features. This choice is not motivated by the need of a more robust classification (which is the most common reason for adopting a multiple classifiers strategy), but because we wanted to exploit the relationship between the classes and the individual features. We let the retrieval system, which is based on a relevance feedback algorithm, to select which features and which classes are appropriate on a case by case basis.

The proposed approach consists of three major steps: first, the images are described by a set of low-level features; then, those descriptions are fed to a battery of image classifiers trained to evaluate the membership of the images with respect to a set of 14 overlapping classes; finally, the output of the classifiers is used to index the images in an image retrieval system, using relevance feedback.

2 Image description by low-level features

Our aim is to train several classifiers for a set of classes. Therefore, we need a fairly general description of the images in terms of low-level features. We considered four features: two that convey shape information, and two that describe color distribution.

For their simplicity and satisfactory performance, bag-of-features representations have become widely used for image classification and retrieval [18–20]. The basic idea is to select a collection of representative patches of the image, compute a visual descriptor for each patch, and use the resulting distribution of descriptors to characterize the whole image. In our work, the patches are the areas surrounding distinctive key-points and are described using the Scale Invariant Feature Transform (SIFT) which is invariant to image scale and rotation, and has been shown to be robust across a substantial range of affine distortions, changes in 3D viewpoint, additions of noise, and changes in illumination [21]. More in detail, we adopted the implementation described in [22] for both key-points detection and description. The SIFT descriptors extracted from an image

are then quantized into “visual words”, which are defined by clustering a large number of descriptors extracted from a set of training images [23]. The final feature vector is the normalized histogram of the occurrences of the visual words in the image (1096 components).

Statistics about the direction of edges may greatly help in discriminating between images depicting natural and man made objects [24]. To describe the most salient edges we used a 8 bin edge direction histogram: the gradient of the luminance image is computed using Gaussian derivative filters tuned to retain only the major edges. Only the points for which the magnitude of the gradient exceeds a set threshold will contribute to the histogram. The image is subdivided into 8×8 blocks, and a histogram for each block is computed (for a total of 512 components).

Spatial color distribution is one of the most widely used feature in image content analysis and categorization. In fact, some classes of images may be characterized in terms of layout of color regions, such as blue sky on top or green grass on bottom. Similarly to Vailaya et al. [12], we divided each image into 9×9 blocks and computed the mean and standard deviation of the values of the color channels of the pixels in each block. The LUV color space is used here, since moments in this color space are more discriminant than in other spaces, at least for image retrieval [25]. This feature includes 486 components (six for each block).

Color moments are less useful when the blocks contain heterogeneous color regions. Therefore, a global color histogram has been selected as a second color feature. The RGB color space has been subdivided in 512 bins by a uniform quantization of each component in eight ranges.

3 Image description by prosemantic features

In order to provide a semantically meaningful information about the content of the images, several categories in which images may be automatically classified have been proposed [12, 24, 26–28]. Based on this work, we selected a set of 14 classes: animals, city, close-up, desert, flowers, forest, indoor, mountain, night, people, rural, sea, street, and sunset. Some classes describe the image at a scene level (city, close-up, desert, forest, indoor, mountain, night, rural, sea, street, sunset) other describe the main subject of the picture (animals, flowers, people). The set of classes is not meant to be exhaustive, or to be able to characterize the content of the images with sufficient specificity for our purposes. Our intent, here, was to select a variegated set of concepts proving a wide range of low-level descriptions of typical scenes.

We queried various image search engines on the web with several keywords related to the classes, and downloaded the resulting pictures. Images have been manually inspected in order to remove those which were not relevant to the classes. Low-quality images have also been removed. The final dataset consist of 30084 pictures, divided into 14 sets of more than 2000 images each. For each class, a set of negative examples has been selected by considering pictures of

the other classes. Since the classes may overlap, a manual inspection was needed to verify that all the selected images were actually negative examples. Note that this dataset is completely separated from the one we used in the retrieval experiments.

For each combination of low-level feature and class, a Support Vector Machine (SVM) has been trained using the implementation described in [29]. We chose to adopt a Gaussian kernel. There are two parameters that need to be tuned (the cost parameter C and the scale of the Gaussian kernel γ), they have been selected by maximizing the cross validation performance of the resulting classifier (see Table 1). The classification performance varies greatly depending on classes and

Table 1. Percentage of classification errors of the classifiers on the 14 classes, using the four low-level feature considered (Bag of features (BoF), color histogram in the RGB color space (RGB), color moments in the YUV color space (YUV), and edge direction histograms (EDH)). The errors have been estimated by a five-fold cross validation on the training sets. For each class, the best result is reported in bold.

Class	BoF	RGB	YUV	EDH
Animals	22.5	30.0	22.9	25.5
City	10.1	20.6	17.1	12.5
Closeup	17.7	27.3	17.2	15.0
Desert	18.7	15.7	14.1	22.0
Flowers	12.8	12.0	12.6	13.3
Forest	7.0	13.6	9.8	9.4
Indoor	14.7	18.5	18.3	12.9
Mountain	14.1	16.8	13.7	20.3
Night	13.5	8.3	6.6	27.5
People	17.0	23.8	20.2	20.5
Rural	18.5	15.7	12.2	22.6
Sea	23.1	21.9	19.4	16.7
Street	18.6	24.5	18.8	17.4
Sunset	12.5	8.4	6.6	16.3
Average	15.8	18.4	15.0	18.0

features, ranging from 6.6% of misclassifications for the “night” class using color moments, to a 30% for the class “animals” using the color histogram. There is not a clearly superior feature and each feature obtained the lowest classification error for at least one class.

Better results can probably be obtained by combining the four scores for each class. However, our goal is not to achieve low misclassification rates, but rather to use the classifiers to warp the high-dimensional feature space into a low-dimensional semantic space without losing valuable information about the visual content of the images. Therefore we decided to keep the information about the individual scores obtained with the four features.

In the end, for each class c and for each low-level feature f , a SVM has been trained. Given a new image Q , represented by the feature vector $\mathbf{x}_Q^{(f)}$, the SVM provide a score $s^{(c,f)}$:

$$s^{(c,f)}(\mathbf{x}_Q^{(f)}) = b^{(c,f)} + \sum_{I \in T^{(c)}} \alpha_I^{(c,f)} y_I^{(c)} \exp\left(-\gamma^{(c,f)} \|\mathbf{x}_I^{(f)} - \mathbf{x}_Q^{(f)}\|^2\right), \quad (1)$$

where $T^{(c)}$ is the training set for class c , $\mathbf{x}_I^{(f)}$ denotes the feature vectors computed on the image I , $y_I^{(c)}$ is the label in $\{-1, +1\}$ which indicates whether I is a positive or a negative example, $b^{(c,f)}$ and $\alpha_I^{(c,f)}$ are the parameters determined by the training procedure, and $\gamma^{(c,f)}$ is the scale parameter of the kernel. The score is expected to be positive when the image belongs to the class c , and negative otherwise. It is well known [30] that the higher the score, the more likely is that the image belongs to the class. Packing together the 56 scores we obtain a compact vector of prosemantic features.

4 The QuickLook² CBIR System

We choose to test the prosemantic features within the framework of the QuickLook² content based retrieval system [5] which easily allows the incorporation and testing of different numerical image representations. The system adopts low-level pictorial features coupled with a relevance feedback mechanism.

With QuickLook², an image database can be queried with the aid of sample images, or user-made sketches, and/or textual image descriptions. When a query is submitted to the system, the retrieved items are presented in decreasing order of relevance, the user is then allowed to progressively refine the system's response by indicating their relevance, or non-relevance. A query refinement mechanism and a relevance feedback algorithm are used to define the new query representing the user needs and to modify the metric used in the retrieval process respectively. For the purpose of this test we use only the low-level pictorial features retrieval capabilities of the system while discarding the textual retrieval functionalities.

Let \mathbf{x}_I be the representation of the image I . Images can be described by different features so \mathbf{x}_I is composed of different numerical vectors, each one representing an image characteristic (e.g. color histogram, shape, etc...). We indicate these vectors for image I as $\mathbf{x}_I^{(1)}, \mathbf{x}_I^{(2)}, \dots, \mathbf{x}_I^{(p)}$. Given a query Q and a image I , the dissimilarity between the two representations is computed as:

$$D(Q, I) = \frac{1}{p} \sum_{f=1}^p D^{(f)}(\mathbf{x}_Q^{(f)}, \mathbf{x}_I^{(f)}) w^{(f)}, \quad (2)$$

where $D^{(f)}$ and $w^{(f)}$ are the dissimilarity metric and the weight associated to the feature f respectively. The weights $w^{(f)}$ allow to tune the contribution of each features in the overall similarity measure. According to the images selected by the user, the weights are determined by the relevance feedback algorithm while the query Q is computed by the query refinement algorithm.

4.1 Relevance Feedback

The QuickLook² system uses a relevance feedback mechanism to update the weights of the similarity function. The key concept of the relevance feedback mechanism, is that the statistical analysis of the image feature distributions the user has judged relevant, or not relevant, can be used to determine what features the user has taken into account (and to what extent) in formulating this judgment, and then accentuate the influence of these features in the overall evaluation of image similarity, as well as in the formulation of a new query. The structure of the relevance feedback mechanism is entirely description-independent, that is, the index can be modified, or extended to include other features without requiring any change in the algorithm as long as the features can be expressed as numerical vectors. The relevance feedback algorithm works as follows: let R_+ the set of relevant images and R_- the set of non relevant images. The feature weights are computed as:

$$w^{(f)} = \begin{cases} \frac{1}{\epsilon} & \text{if } \|R_+\| < 3 \\ \frac{1}{\epsilon + \mu_+^{(f)}} & \text{if } \|R_+\| \geq 3 \text{ and } \|R_-\| = 0 \\ \frac{1}{\epsilon + \mu_+^{(f)}} - \alpha \frac{1}{\epsilon + \mu_*^{(f)}} & \text{otherwise} \end{cases}, \quad (3)$$

where ϵ and α are positive constants, $\mu_+^{(f)}$ is the average of the dissimilarities computed on the f -th feature between each pair of images in R_+ , and $\mu_*^{(f)}$ the average of the dissimilarities computed on the f -th feature between each image in R_+ and each image in R_- . If a weight is negative it is set to 0. A weight is large if the corresponding feature is present in all the relevant images while it is small or dampened if the corresponding feature is variable within the relevant image or is also present in the non relevant images respectively.

4.2 Query Refinement

In content-based retrieval images are sometimes considered relevant because they resemble the query image in just some limited low-level features. Consequently, after an initial query, a given retrieved image may be selected by the user as relevant because it has one of the characteristics of the query (e.g. the same color), and another be selected for another characteristics (e.g. the shape), although the two are actually quite different from each other. To cope with this problem, QuickLook² adopts a new method, called query refinement, for computing the query vector. On the basis of the images selected by the user, the system formulates a new query that better represents the images of interest to the user, taking into account the features of the relevant images, without allowing any one particular feature value to bias the query computation. Let $\mathbf{x}_I^{(f)}(k)$ be the k -th value of the f -th feature of image I . By considering only the images in the relevant set R_+ , the query Q is computed as:

$$Y_k^{(f)} = \{\mathbf{x}_I^{(f)}(k) : |\mathbf{x}_I^{(f)}(k) - \mathbf{x}_Q^{(f)}(k)| \leq 3\sigma_k^{(f)}\}, \quad (4)$$

$$\mathbf{x}_Q^{(f)}(k) = \frac{1}{\|Y_k^{(f)}\|} \sum_{\mathbf{x}_I^{(f)}(k) \in Y_k^{(f)}} \mathbf{x}_I^{(f)}(k), \quad (5)$$

where \bar{Q} is the average query and $\sigma_k^{(f)}$ is the standard deviation of the k -th values in the f -th feature. The query is thus computed from the feature values that mostly agree while the outliers are removed from the computation.

5 Experiments

A user study has been conducted to evaluate the performance of our prosemantic features against the corresponding low-level ones. For our purpose, we substituted the original features in the QuickLook² system with ours and asked 20 subjects to perform ten target search retrieval sessions. All subjects came from the computer science department of the University of Milan - Bicocca: four of them have a background on image processing or computer vision (two Ph.D. students and two post-doctoral fellows), the other 16 are graduate (three) or undergraduate (13) students.

The subjects did the user study one by one on the same desktop with the same instructor. Each subject was constrained to retrieve the target image by selecting any number of relevant and not relevant images within the top 60 retrieved images. They were also allowed to deselect all the previously selected images. Both the search and the deselection accounted as one retrieval operation each and the subjects were instructed that they must retrieve the target image in a maximum of 20 operations without a time limit. During each session the operation performed, the images selected, and the position of the target image within the retrieved results were recorded. In order to minimize user adaptation, the retrieval sessions were conducted alternatively with the low-level features and with the prosemantic features (i.e. one query with the low-level feature and one query with the prosemantic features). For the same reason, each user searched the ten query images in a different order. The subjects were oblivious to what kind of features they were currently using.

The retrieval sessions were organized in such a way that at the end of the user study, each target image was searched half the time by using the low-level and half the time by using the prosemantic features. Before starting each session, the users have been instructed in the use of the system by performing a guided retrieval test.

The dataset used consists of 1875 images taken from the Benchathlon dataset [31]. The dataset includes typical consumer photographs showing a very different distribution of concepts with respect to the dataset used to train the classifiers. For instance, very often the image would fall in the “people” class, while very few images can be considered as belonging to the “desert” or “flowers” classes. The target images have been randomly selected and are shown in Figure 1. Other 60 images have been randomly selected to compose the page from which the users started all their searches. These images are shown in Figure 2.



Fig. 1. The ten images used in the target search retrieval sessions.

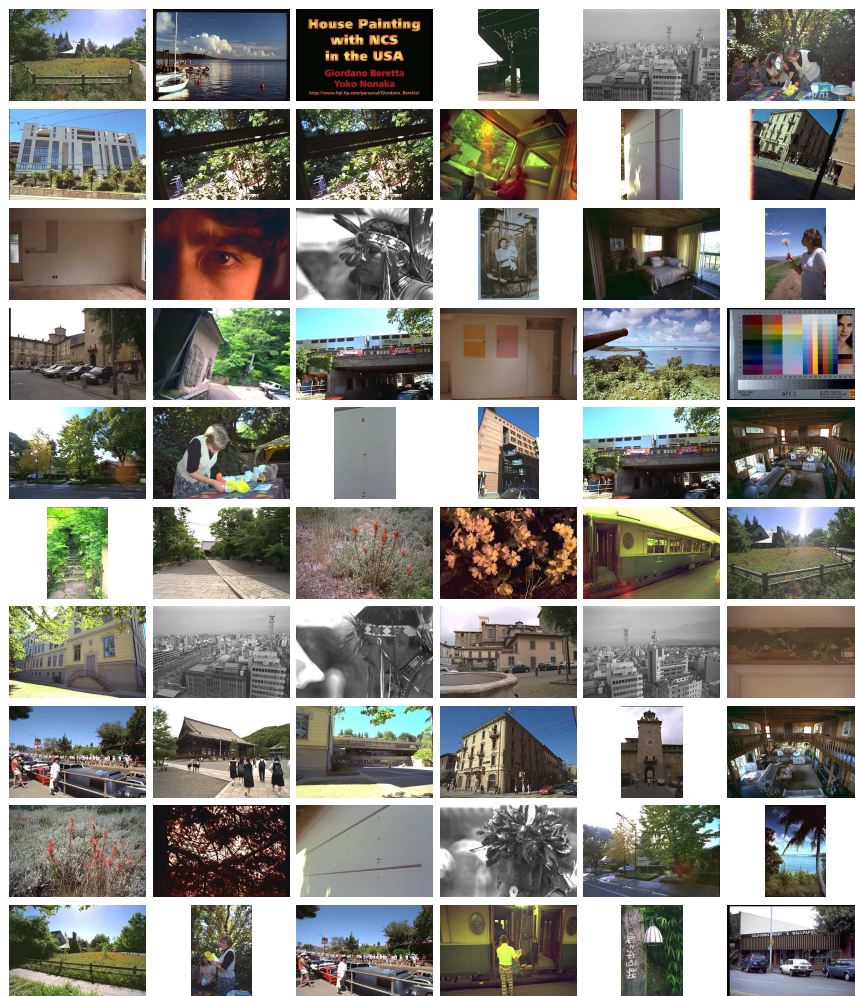


Fig. 2. The 60 images which compose the starting page of the searches.

The outcome of the 200 searches clearly demonstrates the effectiveness of prosemanic features with respect to low-level features. Using the prosemanic features, only seven times were the users not able to retrieve the target images within the limit of 20 retrieval operations. By contrast the limit has been exceeded 49 times in the case of low-level features. Figure 3 shows the cumulative success rate for the two sets of features as a function of the number of iterations. The plot shows how prosemanic features allows the retrieval of more target images and with less iterations. In particular, in the case of prosemanic features in more than one third (35/100) of the cases the retrieval of the target image required only one iteration (i.e. without really exploiting the relevance feedback algorithm). Using low-level features this happened only in 11 cases.

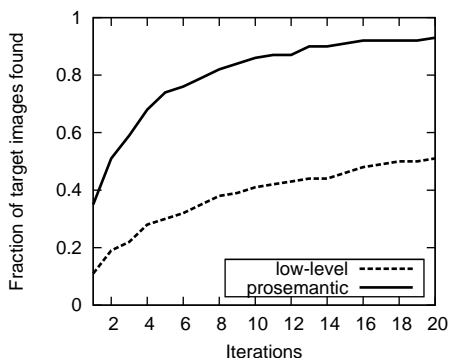












Fig. 3. Fraction of images successfully retrieved as a function of the number of iterations.

Since the performance changes significantly for different target images, we reported in Table 2 the results obtained on each of the ten queries. On nine cases out of ten, the use of prosemanic features obtained a higher success rate. The only exception is query (g) which has been quite difficult to find with both the features considered. Two images have never been found using low-level features (d and f), while they have been considered among the easiest to find using prosemanic features. There are two cases (queries e and h) which present clearly distinguishable visual characteristics (one is a grayscale image, the other presents a strong color cast). This fact has been recognized by the majority of users which exploited it to quickly find the targets using low-level features; however, the few users who have not been able to master how low-level similarity works failed the retrieval task. In these two cases retrieval with prosemanic features required (on average) a higher number of iterations, but with only one failure.

Observing the users and discussing with them after the experiment, we made the hypothesis that the effectiveness of the prosemanic features derives from their capability of encoding characteristics of the images which allow a better match against users' intuition about the similarity of the images. Very often,

Table 2. Detail of the results obtained on the ten query images using the two sets of features considered. For each query image are reported the number of successful searches (over 10 attempts for each feature set), the number of iterations needed to retrieve the image (averaged over the successful searches), and the corresponding standard deviation.

Query Image	Features	Successful searches	Iterations Average	Std deviation
(a) 	low-level	8	9.75	5.49
	prosesemantic	10	6.80	4.21
(b) 	low-level	5	4.20	4.35
	prosesemantic	9	4.00	2.11
(c) 	low-level	6	3.67	5.09
	prosesemantic	9	1.11	0.31
(d) 	low-level	0	-	-
	prosesemantic	9	3.33	1.70
(e) 	low-level	7	1.29	0.45
	prosesemantic	10	3.80	3.16
(f) 	low-level	0	-	-
	prosesemantic	10	1.30	0.64
(g) 	low-level	9	7.78	4.39
	prosesemantic	7	8.00	5.63
(h) 	low-level	7	5.29	4.40
	prosesemantic	9	8.11	4.56
(i) 	low-level	6	7.50	5.41
	prosesemantic	10	1.10	0.30
(j) 	low-level	3	9.00	5.10
	prosesemantic	10	1.80	1.60

the users started by selecting pictures with the same “general theme” of the target image (e.g. pictures of people, city shots, . . .). Conversely, reasoning about low-level features would require specific training. To verify this intuition we considered the variation of the number of successfully retrieved images during the sessions. Therefore, we counted for each feature set how many images has been retrieved among the first two searches of each user. We did the same for the second two searches and so on. . . We considered pairs of searches because the two feature sets have been used alternatively by each subject. The results are shown in Figure 4. Using the prosemantic features performances are very close to the maximum attainable (i.e. 20 successes) straight from the beginning of the retrieval sessions. Therefore, it is not possible to distinguish any user adaption phenomenon. For what concern low-level features, instead, it seems that performance actually increased during the sessions: from only four retrieved images within the first two searches, to 14 within the last two searches. So it is possible that, provided a sufficient amount of training of the user, low-level features may reach the same retrieval performance of prosemantic features.

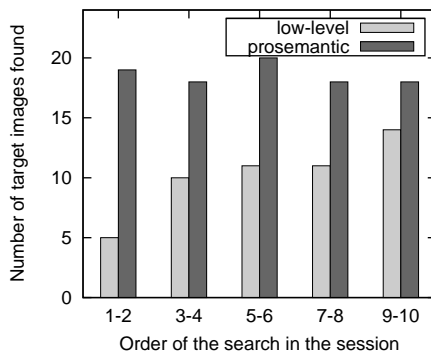


Fig. 4. Number of images successfully retrieved as a function of the order in the sequence of searches.

6 Conclusions

We have presented here, an image description approach based on prosemantic features. These features are obtained by multiple classifiers trained to identify 14 semantic concepts, on the basis of different low-level representations. To verify the effectiveness of the approach, we designed an image retrieval experiments in which low-level and prosemantic features are embedded into a content-based image retrieval system based on relevance feedback. The experiments show that the use of prosemantic features allows for a more successful and quick retrieval of the query images.

To further assess the generalization capabilities of prosemantic features, we plan to extend the experimentation by recruiting more subjects and by considering additional queries. We are also considering to test prosemantic features in other application scenarios such as automatic image annotation and classification.

References

1. Brunelli, R., Mich, O.: Image retrieval by examples. *IEEE Transactions on Multimedia* **2**(3) (2000) 164–171
2. Flickner, M., Sawhney, H., Niblack, W., Ashley, J., Huang, Q., Dom, B., Gorkani, M., Hafner, J., Lee, D., Petkovic, D., Steele, D., Yanker, P.: Query by image and video content: the QBIC system. *IEEE Computer* **28**(9) (1995) 23–32
3. Gevers, T., Smeulders, A.: PicToSeek: combining color and shape invariant features for image retrieval. *IEEE Transactions on Image Processing* **9**(1) (2000) 102–119
4. Smith, J.R., Chang, S.F.: VisualSEEK: a fully automated content-based image query system. In: *Proceedings of the fourth ACM international conference on Multimedia*. (1996) 87–98
5. Ciocca, G., Gagliardi, I., Schettini, R.: Quicklook²: An integrated multimedia system. *Journal of Visual Languages & Computing* **12**(1) (2001) 81–103
6. Ahmad, I., Grosky, W.I.: Indexing and retrieval of images by spatial constraints. *Journal of Visual Communication and Image Representation* **14**(3) (2003) 291–320
7. Smeulders, A.W.M., Worring, M., Santini, S., Gupta, A., Jain, R.: Content-based image retrieval at the end of the early years. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **22**(12) (2000) 1349–1380
8. Zhou, X.S., Huang, T.S.: Relevance feedback in image retrieval: a comprehensive review. *Multimedia Systems* **8**(6) (2003) 536–544
9. Lee, C.S., Ma, W.Y., Zhang, H.: Information embedding based on user’s relevance feedback for image retrieval. In: *Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series*. Volume 3846. (1999) 294–304
10. Datta, R., Li, J., Wang, J.Z.: Content-based image retrieval: approaches and trends of the new age. In: *Proceedings of the 7th ACM SIGMM international workshop on Multimedia information retrieval*. (2005) 253–262
11. Fan, J., Gao, Y., Luo, H., Xu, G.: Automatic image annotation by using concept-sensitive salient objects for image content representation. In: *Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval*. (2004) 361–368
12. Vailaya, A., Figueiredo, M., Jain, A., Zhang, H.J.: Image classification for content-based indexing. *IEEE Transactions on Image Processing* **10**(1) (2001) 117–130
13. Chen, X., Wang, J.Z.: Image categorization by learning and reasoning with regions. *Journal of Machine Learning Research* **5** (2004) 913–939
14. Chang, E., G., K., Sychay, G., Gang, W.: CBSA: content-based soft annotation for multimodal image retrieval using bayes point machines. *IEEE Transactions on Circuits and Systems for Video Technology* **13**(1) (2003) 26–38
15. Smith, J.R., Naphade, M., Natsev, A.: Multimedia semantic indexing using model vectors. In: *Proceedings of IEEE International Conference on Multimedia and Expo*. (2003) 445–448
16. Lu, J., Ma, S., Zhang, M.: Automatic image annotation based on model space. In: *Proceedings of IEEE International Conference on Natural Language Processing and Knowledge Engineering*. (2005) 455–460

17. Wang, J., Li, J., Wiederhold, G.: Simplicity: Semantics-sensitive integrated matching for picture libraries. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **23**(9) (2001) 947–963
18. Zhang, J., Marszalek, M., Lazebnik, S., Schmid, C.: Local features and kernels for classification of texture and object categories: A comprehensive study. *International Journal of Computer Vision* **73**(2) (2007) 213–238
19. Wallraven, C., Caputo, B., Graf, A.: Recognition with local features: the kernel recipe. In: *Proceedings of the Ninth IEEE International Conference on Computer Vision*. Volume 1. (2003) 257–264
20. Grauman, K., Darrell, T.: The pyramid match kernel: discriminative classification with sets of image features. In: *Proceedings of the Tenth IEEE International Conference on Computer Vision*. Volume 2. (2005) 1458–1465
21. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision* **60**(2) (2004) 91–110
22. Vedaldi, A.: Sift++ a lightweight c++ implementation of sift. <http://vision.ucla.edu/~vedaldi/code/siftpp/siftpp.html>
23. Nister, D., Stewenius, H.: Scalable recognition with a vocabulary tree. In: *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*. Volume 2. (2006) 2161–2168
24. Vailaya, A., Jain, A., Zhang, H.J.: On image classification: city images vs. landscapes. *Pattern Recognition* **31**(12) (1998) 1921–1935
25. Furht, B.: Content-based image indexing and retrieval. In: *Handbook on Multimedia Computing*. CRC Press, Inc. (1998)
26. Vogel, J., Schiele, B.: Semantic modeling of natural scenes for content-based image retrieval. *International Journal of Computer Vision* **72**(2) (2007) 133–157
27. Schettini, R., Brambilla, C., Cusano, C., Ciocca, G.: Automatic classification of digital photographs based on decision forests. *International Journal of Pattern Recognition and Artificial Intelligence* **18**(5) (2004) 819–845
28. Szummer, M., Picard, R.: Indoor-outdoor image classification. In: *Proceedings of IEEE International Workshop on Content-Based Access of Image and Video Database*. (1998) 42–51
29. Chang, C.C., Lin, C.J.: LIBSVM: a library for support vector machines. (2001) Software available at <http://www.csie.ntu.edu.tw/~cjlin/libsvm>.
30. Platt, J.C.: Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. In: *Advances in Large Margin Classifiers*. (1999) 61–74
31. Gunther, N.J., Beretta, G.: A benchmark for image retrieval using distributed systems over the internet: BIRDS-I. Technical Report HPL-2000-162, HP Labs, Palo Alto (2001)