Enhancing Relevance Feedback in Image Retrieval Using Unlabeled Data

ZHI-HUA ZHOU, KE-JIA CHEN, and HONG-BIN DAI
Nanjing University

Relevance feedback is an effective scheme bridging the gap between high-level semantics and low-level features in content-based image retrieval (Cbir). In contrast to previous methods which rely on labeled images provided by the user, this paper attempts to enhance the performance of relevance feedback by exploiting unlabeled images existing in the database. Concretely, this paper integrates the merits of semi-supervised learning and active learning into the relevance feedback process. In detail, in each round of relevance feedback, two simple learners are trained from the labeled data, i.e. images from user query and user feedback. Each learner then labels some unlabeled images in the database for the other learner. After re-training with the additional labeled data, the learners classify the images in the database again and then their classifications are merged. Images judged to be positive with high confidence are returned as the retrieval result, while those judged with low confidence are put into the *pool* which is used in the next round of relevance feedback. Experiments show that using semi-supervised learning and active learning simultaneously in Cbir is beneficial, and the proposed method achieves better performance than some existing methods.

Categories and Subject Descriptors: H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval—*relevance feedback*; H.2.8 [**Database Management**]: Database Applications—*image database*; I.2.6 [**Artificial Intelligence**]: Learning

General Terms: Algorithm, Design, Experimentation

Additional Key Words and Phrases: Content-based image retrieval, relevance feedback, machine learning, learning with unlabeled data, semi-supervised learning, active learning

1. INTRODUCTION

With the rapid increase in the volume of digital image collections, content-based image retrieval (Cbir) has attracted a lot of research interest in recent years [Smeulders et al. 2000]. The user could pose an example image, i.e. user query, and ask the Cbir system to bring out relevant images from the database. A main difficulty here is the gap between high-level semantics and low-level image features, due to the rich content but subjective semantics of an image. Relevance feedback has been shown as a powerful tool for bridging this gap [Rui et al. 1998; Zhou and Huang 2003]. In

This research was supported by the Foundation for the Author of National Excellent Doctoral Dissertation of China under the Grant No. 200343 and the the National Science Fund for Distinguished Young Scholars of China under the Grant No. 60325207.

Author's address: National Laboratory for Novel Software Technology, Nanjing University, Nanjing 210093, China; email: {zhouzh, chenkj, daihb}@lamda.nju.edu.cn.

relevance feedback, the user has the option of labeling a few images according to whether they are relevant to the target or not. The labeled images are then given to the CBIR system as complementary queries so that more images relevant to the user query can be retrieved from the database.

In fact, the retrieval engine of a CBIR system can be regarded as a machine learning process, which attempts to train a learner to classify the images in the database as two classes, i.e. positive (relevant) or negative (irrelevant). Since the classification is usually with different confidence, the learner produces a rank of the images according to how confident it believes the images are relevant to the user query. The higher the rank, the more relevant the corresponding image. Upon receiving the user feedback, the machine learning process uses the newly labeled images along with the original user query to re-train the learner, so that a new rank can be produced which typically puts more relevant images at higher ranks than the original one did. It is obvious that the above is a typical supervised learning process, where only labeled data are used in the training of the learner. In CBIR, since it is not convenient to ask the user to label many images, the labeled training examples given by the user query and relevance feedback may be very small.

During the past few years, using unlabeled data to help supervised learning has become a hot topic in machine learning and data mining. Considering that in CBIR there are lots of unlabeled images in the database, this paper proposes to exploit them to enhance the performance of relevance feedback. Based on a preliminary work [Zhou et al. 2004], the SSAIRA (Semi-Supervised Active Image Retrieval with Asymmetry) method is proposed, which integrates the merits of semi-supervised learning and active learning into the relevance feedback process. Here the semi-supervised learning mechanism is used to help complement the small training set, while the active learning mechanism is used to help enlarge the useful information conveyed by user feedback. Considering that the training examples are usually asymmetrical in CBIR, i.e. positive images can be regarded as belonging to the same target semantic class while negative images usually belong to different semantic classes, the positive and negative images are processed in different ways, where virtual negative examples derived by generalizing the real negative ones are used. Furthermore, since in CBIR the user is interacting with the system in real time, very simple learners are employed. In other words, the proposed method tries to tackle the three special issues of relevance feedback [Zhou and Huang 2003], i.e. small sample, asymmetrical training sample, and real time requirement.

The rest of this paper is organized as follows. Section 2 briefly introduces the research background of the paper. Section 3 presents the SSAIRA method. Section 4 reports on the experiments. Finally, Section 5 concludes and raises several issues for future work.

2. BACKGROUND

2.1 Relevance Feedback

The concept of relevance feedback was introduced into CBIR from text-based information retrieval in the 1990s [Rui et al. 1998] and then became a popular technique in CBIR. This is not strange because images are more ambiguous than texts, making user interaction desirable. With relevance feedback, a user can label a few more images as new examples for the retrieval engine if he or she is not satisfied with the current retrieval result. Actually, these new images refine the original query implicitly, which enables the relevance feedback process to bridge the gap between high-level image semantics and low-level image features. There is a good recent review on relevance feedback [Zhou and Huang 2003], therefore this subsection only introduces issues that are highly related to the work of this paper.

From the view of machine learning, the retrieval engine in fact accomplishes a learning task, that is, classifying the images in the database as positive or negative images. Here an image is positive if it is relevant to the user query and negative otherwise. In contrast to typical machine learning settings, this learning task has some special characteristics [Zhou and Huang 2003], i.e. small sample, asymmetrical training sample, and real time requirement.

The small sample problem is due to the fact that few users will be so patient to provide a lot of example images in the relevance feedback process. Indeed, in most scenarios the number of example images is very small, especially when considering that there are usually a large number of potential image classes and that the images are described by a lot of features. Although there are many machine learning algorithms focusing on learning with a finite number of training examples, learning with an extremely small number of training examples remains a very difficult problem. This means that most popular machine learning algorithms can hardly be applied to Cbir directly. In general, there are two ways to tackle the small sample issue. The first one is to design a smart mechanism that would deal with the limited number of training examples directly. For example, Ishikawa et al. [1998] tried to replace the regular inverse by the Moore-Penrose inverse or pseudo-inverse matrix in computing the sample covariance matrix and its inverse. A better solution proposed by Zhou and Huang [2001] added regularization terms on the diagonal of the sample covariance matrix before the inversion. The second way for tackling the small sample issue is to exploit unlabeled images in the database, as is done in this paper. Some related work on this will be introduced in the next subsection.

The asymmetrical training sample problem is caused by the fact that the Cbir problem is not a real binary classification problem. Typical machine learning algorithms regard the positive and negative examples interchangeably and assume that both sets are distributed approximately equally. However, in Cbir although it is reasonable to assume that all the positive examples belong to the same target class, it is usually not valid to make the same assumption for the negative ones because different negative examples may belong to different irrelevant classes and the small number of negative examples can hardly be representative for all the irrelevant classes. Therefore, it may be better to process the positive and negative examples differently. Picard et al. [1996] chose image sets that most efficiently describe positive examples under the condition that they don't describe negative examples well. Nastar et al. [1998] proposed empirical formulae to take into account negative examples while estimating the distribution of positive examples along each feature component. Vasconcelos and Lippman [2000] used a Bayesian model where the classes under which negative examples score well are penalized. Kherfi et al. [2002] used positive examples in an initial query and then used negative examples to refine the query by considering the occurrences of features in positive and negative examples. Zhou and Huang [2003] assumed that positive examples have a compact low-dimensional support while negative examples can have any configuration, and therefore they used BDA (Biased Discriminant Analysis) to find the low-dimensional space in which positive examples cluster while the negative ones scatter away. Zhang and Zhang [2004] assumed that each negative example represents a unique potential semantic class and used a kernel density estimator to determine the statistical distribution of the irrelevant classes. Similarly, this paper assumes that examples belonging to the same irrelevant class cluster in a certain space, and therefore, virtual examples derived from the negative example and its neighbors can be better representatives of the class.

The real time requirement problem is due to the fact that the user usually wishes to get the retrieval results as soon as possible, and few users will be so patient to take part in a time-consuming interaction process. A reasonable way to address the real time issue is to adopt efficient image storage structures, such as the hierarchical tree structure used in [Chen et al. 2000]. However, using such a structure may make the learning task more difficult because the structure has to be updated once new knowledge is discovered through user interaction. Another feasible way is to use a set of few features that can be evaluated rapidly in processing the queries. For example, Tieu and Viola [2000] defined a very large set of highly selective features each of which will respond to only a small percentage of images in the database, and then a Boosting algorithm was used to quickly select a small number of features which distinguish the presented images well. This paper chooses a different direction, where very simple online learners are used such that a complicated time-consuming learning process is avoided.

It is noteworthy that, according to [Zhou and Huang 2003], there are different styles of relevance feedback implementations based on different user models. This paper assumes that the user is looking for a class of images instead of a specific image, the user only gives binary feedback for positive and negative examples instead of giving relevant scores, and the user is eager to get satisfying retrieval results as soon as possible. But note that the methods and the underlying ideas presented in the paper are quite general, and they can also be applied to other relevance feedback styles.

2.2 Learning from Unlabeled Examples

Learning from unlabeled examples has become a hot topic during the past few years because in many real-world applications labeled training examples are fairly expensive to obtain while unlabeled examples are abundantly available. There are two main machine learning paradigms for this purpose: semi-supervised learning and active learning.

Semi-supervised learning deals with methods for exploiting unlabeled data in addition to labeled data to improve learning performance. Many current semi-supervised learning methods use a generative model for the classifier and employ Expectation-Maximization (EM) [Dempster et al. 1977] to model the label estimation or parameter estimation process. For example, mixture of Gaussians [Shahshahani and Landgrebe 1994], mixture of experts [Miller and Uyar 1997], and naïve Bayes [Nigam et al. 2000] have been respectively used as the generative model, while EM is used to combine labeled and unlabeled data for classification. There are also many other methods such as using transductive inference for support vector machines to optimize performance on a specific test set [Joachims 1999], constructing a graph on the examples such that the minimum cut on the graph yields an optimal labeling of the unlabeled examples according to certain optimization functions [Blum and Chawla 2001], etc.

A prominent achievement in this area has been the *co-training* paradigm proposed by Blum and Mitchell [1998], which trains two classifiers separately on two *sufficient and redundant views*, i.e. two attribute sets each of which is sufficient for learning and conditionally independent of the other given the class label, and uses the predictions of each classifier on unlabeled examples to augment the training set of the other. Dasgupta et al. [2002] have shown that when the requirement of sufficient and redundant views is met, the co-trained classifiers could make fewer generalization errors by maximizing their agreement over the unlabeled data. Unfortunately, such a requirement can hardly be met in most scenarios. Goldman and Zhou [2000] proposed an algorithm that does not need two views. This algorithm requires two different supervised learning algorithms that partition the instance space into a set of equivalence classes, and employs a cross validation technique to determine how to label the unlabeled examples and how to produce the final hypothesis. Zhou and Li [2005b] proposed the *tri-training* algorithm which requires neither two different views nor two different supervised learning algorithms. Through employing three classifiers, this algorithm can implicitly measure the labeling confidence whereas previous algorithms require explicit measurement. Moreover, it could utilize ensemble learning to help improve generalization ability. It is worth noting that although the requirement of sufficient and redundant views is quite strict, the co-training paradigm has already been used in many domains such as statistical parsing and noun phrase identification [Pierce and Cardie 2001; Sarkar 2001; Hwa et al. 2003; Steedman et al. 2003].

A few approaches have tried to apply semi-supervised learning to Cbir. Wu et al. [2000] cast Cbir as a transductive learning problem and proposed the D-em algorithm to solve the problem. On a small subset of Corel which contains 134 images, they reported that their approach had achieved good results, regardless of what physical and mathematical features had been used. Dong and Bhanu [2003] proposed a new semi-supervised EM algorithm where the image distribution in feature space is modelled as a mixture of Gaussian densities. It is noteworthy that they attempted to utilize meta-knowledge in Cbir, i.e. the previous experience of each query image with various users, which is quite different from other approaches. Tian et al. [2004] studied the usefulness of unlabeled data in Cbir and they reported that if the distribution of the unlabeled data is different from that of the labeled data, using unlabeled data may degrade the performance. Zhang and Zhang [2004] used a roulette wheel selection strategy to select unlabeled examples to help improve the estimation of the distribution of the irrelevant semantic class corresponding to the labeled negative example, where the unlabeled examples with smaller distances to the concerned negative example have larger probabilities to be selected. Yao and Zhang [2005] proposed a method which uses *perceived accuracy* to help estimate the real accuracy of the classifiers refined in the semi-supervised learning process, such that the refinement can be terminated when it causes the deterioration of the real accuracy. This method was then applied to aerial imagery object detection, a task different from but related to Cbir, and resulted in good performance.

Active learning deals with methods that assume the learner has some control over the input space. In utilizing unlabeled data, it goes a different way from semi-supervised learning, where an *oracle* can be queried for labels of specific instances, with the goal of minimizing the number of queries required. There are two major schemes, i.e. *uncertainty sampling* and *committee-based sampling*. Methods of the former such as [Lewis and Gale 1994] train a single learner and then query the unlabeled instances on which the learner is least confident. Methods of the latter such as [Seung et al. 1992; Abe and Mamitsuka 1998] generate a committee of several learners and select the unlabeled instances on which the committee members disagree the most. A recent advance is the *co-testing* paradigm proposed by Muslea et al. [2000], which trains two learners separately on two different views as co-training does, and selects the query based on the degree of disagreement among the learners.

As early as in 1983, Bookstein [1983] conjectured that having the user label the top-ranked documents, while desirable from a user interface standpoint, might not be optimal for learning. But until Lewis and Gale [1994] showed that labeling documents with a current estimated probability of 0.5 relevance could improve effectiveness of a text classifier over labeling top-ranked documents, active learning had not been introduced into information retrieval. As for Cbir, active learning began to be used only recently. In the Svm$_{Active}$ approach developed by Tong and Chang [2001], in each round of relevance feedback, a support vector machine is trained on labeled data and then the user is asked to label the images closest to the support vector boundary. Cox et al. [2000] used entropy-minimization in search of the unlabeled images that, once labeled, will minimize the expected amount of future feedbacks. Note that this method takes feedback in the form of relative judgements ("image a is more relevant than image b") instead of binary feedback for positive and negative. Another work on introducing active learning to Cbir was done by Zhang and Chen [2002]. Their system randomly chooses a small number of images to annotate at first. Then, the system starts to repeatedly select the image with the maximum knowledge gain for the user to annotate, until the user stops or the database has been fully annotated. Note that this work focuses on hidden annotation instead of relevance feedback. As Zhang and Chen [2002] indicated, relevance feedback does not accumulate semantic knowledge while hidden annotation, on the other hand, tries to accumulate all the knowledge given by the user.

3. THE PROPOSED METHOD

In Cbir, the user normally poses an example image as the query. From the view of machine learning, such a user query is a labeled positive example, while the image database is a collection of unlabeled data[1]. Let $\mathcal{U}$ denote the unlabeled data set while $\mathcal{L}$ denotes the labeled data set, $\mathcal{L} = \mathcal{P} \cup \mathcal{N}$ where $\mathcal{P}$ and $\mathcal{N}$ respectively denote the sets of labeled positive examples and negative examples. Originally, $\mathcal{U}$ is the

[1] For simplicity of discussion, here it is assumed that the database contains no annotation.

whole database DB, $\mathcal{P}$ is $\{query\}$, and $\mathcal{N}$ is empty. Let $|\mathcal{D}|$ denote the size of a set $\mathcal{D}$. Then the sizes of the original $\mathcal{U}$, $\mathcal{P}$ and $\mathcal{N}$ are $|DB|$, 1, and 0, respectively.

In relevance feedback, the user may label several images according to whether they are relevant or not to a *query*, which could be viewed as providing additional positive or negative examples. Let $\mathcal{P}^*$ and $\mathcal{N}^*$ denote the new positive and negative examples, respectively. Since the feedback is usually performed on images in the database, both $\mathcal{P}^*$ and $\mathcal{N}^*$ are subsets of DB. Therefore, the relevance feedback process changes $\mathcal{L}$ and $\mathcal{U}$. As for $\mathcal{L}$, its positive subset $\mathcal{P}$ is enlarged to be $\mathcal{P} \cup \mathcal{P}^*$, and its negative subset $\mathcal{N}$ is enlarged to be $\mathcal{N} \cup \mathcal{N}^*$; but as for $\mathcal{U}$, since some of its elements have been moved to $\mathcal{L}$, it is shrunk to $\mathcal{U} - (\mathcal{P}^* \cup \mathcal{N}^*)$.

In each round of relevance feedback, after obtaining the enlarged $\mathcal{P}$ and $\mathcal{N}$, a traditional CBIR system will re-train a learner which then will give every image in $\mathcal{U}$ a rank expressing how relevant the image is to *query*. It is obvious that such a rank could be more accurate than the one generated by the learner trained with only the original $\mathcal{P}$ and $\mathcal{N}$ because now the learner is fed with more training examples. It can be anticipated that if more training examples could be obtained, the performance could be further improved.

Inspired by the co-training paradigm [Blum and Mitchell 1998], SSAIRA attempts to exploit $\mathcal{U}$ to improve the performance of retrieval. Concretely, SSAIRA employs two learners. After obtaining the enlarged $\mathcal{P}$ and $\mathcal{N}$, both learners are re-trained and then each of them gives every image in $\mathcal{U}$ a rank. Here the rank is a value between -1 and $+1$, where positive/negative means the learner judges the concerned image to be relevant/irrelevant, and the bigger the absolute value of the rank, the stronger the confidence of the learner on its judgement. Then, each learner will choose some unlabeled images to label for the other learner according to the rank information. After that, both the learners are re-trained with the enlarged labeled training sets and each of them will produce a new rank for images in $\mathcal{U}$. The new ranks generated by the learners can be easily combined via summation, which results in the final rank for every image in $\mathcal{U}$. Then, images with the top *resultsize* ranks are returned. Here *resultsize* specifies how many relevant images are anticipated to be retrieved. This parameter could be omitted so that all the images in the database are returned according to descending order of the real value of their ranks. Note that the above process can be repeated many times until the user no longer provides feedback.

The learners used by SSAIRA may be implemented in different ways. In this paper, in order to avoid a complicated learning process so that the real time requirement may be met, a very simple model is used, as shown in Eq. 1, where $i \in \{1, 2\}$ is the index of the learner, $\boldsymbol{x}$ is the image or feature vector[2] to be classified, $\mathcal{P}_i$ and $\mathcal{N}_i$ are respectively the set of labeled positive and negative examples in the current training set of L_i, $\mathcal{Z}_{norm}$ is used to normalize the result to $(-1, 1)$, ε is a small constant used to avoid a zero denominator, and Sim_i is the similarity measure adopted by L_i.

[2] Images can be represented as feature vectors after appropriate feature extraction.

$$L_i\left(\boldsymbol{x},\mathcal{P}_i,\mathcal{N}_i\right)=\left(\sum_{\boldsymbol{y}\in\mathcal{P}_i}\frac{Sim_i\left(\boldsymbol{x},\boldsymbol{y}\right)}{\left|\mathcal{P}_i\right|+\varepsilon}-\sum_{\boldsymbol{z}\in\mathcal{N}_i}\frac{Sim_i\left(\boldsymbol{x},\boldsymbol{z}\right)}{\left|\mathcal{N}_i\right|+\varepsilon}\right)/\mathcal{Z}_{norm} \tag{1}$$

Here the similarity between the two d-dimensional feature vectors $\hat{\boldsymbol{x}}$ and $\hat{\boldsymbol{y}}$ is measured by the reciprocal of the Minkowski distance, as shown in Eq. 2 where ξ is a small constant used to avoid a zero denominator.

$$Sim_i\left(\boldsymbol{x},\boldsymbol{y}\right)=1/\left(\left(\sum\nolimits_{j=1}^{d}\left|\hat{\boldsymbol{x}}_j-\hat{\boldsymbol{y}}_j\right|^{p_i}\right)^{1/p_i}+\xi\right) \tag{2}$$

It is worth noting that the learners should be diverse because if they were identical, then for either learner, the unlabeled examples labeled by the other learner may be the same as those labeled by the learner for itself. Thus, the process degenerates into *self-training* [Nigam and Ghani 2000] with a single learner. In the standard setting of co-training, the use of sufficient and redundant views enables the learners to be different. In the variant of co-training which does not require sufficient and redundant views [Goldman and Zhou 2000], the diversity among the learners is achieved by using different supervised learning algorithms that partition the instance space into a set of equivalence classes. Since SSAIRA does not assume sufficient and redundant views, nor does it employ different supervised learning algorithms that partition the instance space into a set of equivalence classes, the diversity of the learners should be sought from other channels.

Here the orders of the Minkowski distance, i.e. p_i in Eq. 2, are set to different values for the two learners. In general, the smaller the order, the more robust the resulting distance metric to data variations; while the bigger the order, the more sensitive the resulting distance metric to data variations. Therefore, with different order settings, L_1 and L_2 could produce different ranks for the images in $\mathcal{U}$. Moreover, such a scheme can also offer another advantage, that is, since it is usually difficult to decide which order of the Minkowski distance is better for the concerned task, the functions of these learners may be somewhat complementary in combining the ranks they produce. It is worth mentioning that such a scheme has been employed in co-training regressors and has achieved success [Zhou and Li 2005a].

Indeed, the learners defined in Eq. 1 are quite trivial, whose performance is determined by the contents of $\mathcal{P}_i$ and $\mathcal{N}_i$. An advantage is that in contrast to many other complicated learners, updating such trivial learners is very easy, which enables the relevance feedback process to be efficient, especially when considering that the learners are to be updated after every round of relevance feedback.

On the other hand, since the learners are not strong, the labels they assign to the unlabeled examples may be incorrect. For a particular query, usually only a small number of images in the database are relevant while most images are irrelevant, therefore the unlabeled images labeled negatively by the learners should be more reliable. Considering this, SSAIRA adopts a very conservative strategy, that is, in each round of relevance feedback each learner only labels for the other learner its two most confident negative examples, i.e. images with the smallest rank (near -1).

In fact, in the experiments reported in Section 4, the two most confident negative examples are always really negative ones while there are cases where the two most confident positive examples are in fact negative.

In order to improve the reliability further, another conservative mechanism is employed, that is, the images labeled by the learners won't be moved from $\mathcal{U}$ to $\mathcal{L}$. In other words, they are only temporarily used as labeled training examples, and in the next round of relevance feedback they will be regarded as unlabeled data again. In this way, the influence of the possible mistakes made by the learners can be limited.

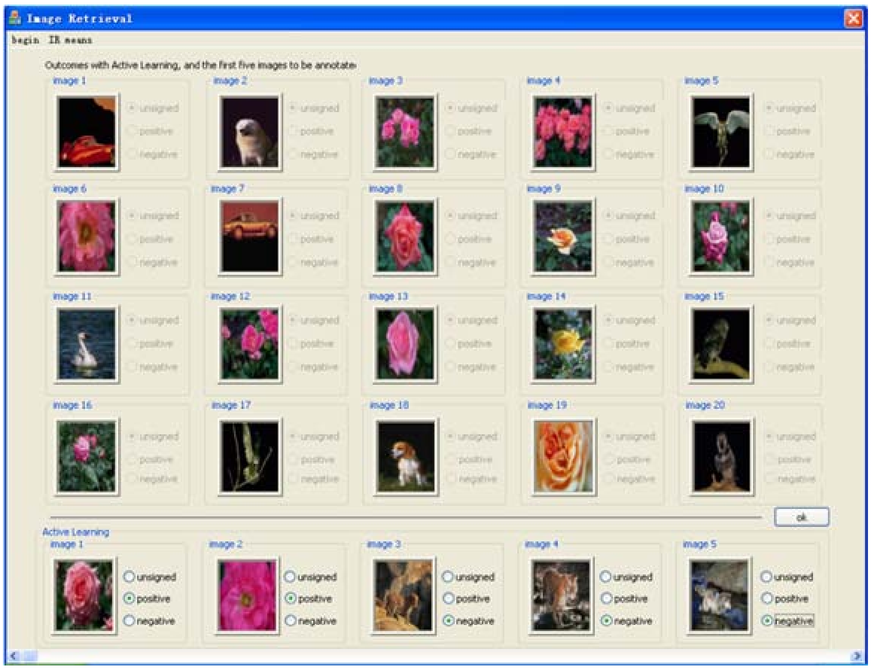
It may be questioned whether the two additional examples can really achieve positive results because "two" is a very small number. In fact, in CBIR the number of labeled examples is very limited because few users will be patient to label more than ten images in each round of relevance feedback. In most scenarios labeling more than five images in each round will make the user feel uncomfortable, while the additional two examples can bring about 40% additional examples if the user labels five images.

In traditional CBIR systems, the pool for the user to give feedback is not distinguished from the retrieved images. That is, the system gives the user the retrieval result, and then the user chooses certain images from the result to label. It is evident that in this way, the images labeled by the user in the relevance feedback process may not be the ones that are most helpful in improving the retrieval performance. For example, labeling an image that has already been well learned is helpless.

Inspired by the co-testing paradigm [Muslea et al. 2000], since SSAIRA employs two learners, it can be anticipated that labeling images on which the learners disagree the most, or both learners are with low confidence, may be of great value. Therefore, SSAIRA puts images with the bottom *poolsize* absolute ranks (near 0) into the pool for relevance feedback. Here *poolsize* specifies how many images can be put into the pool. This parameter could be omitted so that all the images in the database are pooled according to ascending order of the absolute value of their ranks.

Thus, SSAIRA does not passively wait for the user to choose images to label. Instead, it actively prepares a pool of images for the user to provide feedback. A consequence is that in designing the user interface, the retrieval result should be separated from the pool for relevance feedback. For example, the user interface of a prototype system is shown in Fig. 1, where the region above the dark line displays the retrieved images while the region below the dark line displays the pooled images for relevance feedback.

As mentioned before, in CBIR the positive examples can be regarded as belonging to the same relevant class, but the negative examples may belong to different irrelevant classes. Considering that there may exist a large number of potential irrelevant semantic classes, like the strategy adopted in [Zhang and Zhang 2004], this paper assumes that each negative example is a representative of a potential semantic class. Intuitively, examples close to the negative example should have a strong chance to belong to the same potential semantic class. Moreover, as mentioned before, for a particular query usually only a small number of images are



Outcomes with Active Learning, and the first five images to be annotate

image 1 ● unsigned ○ positive ○ negative
image 2 ● unsigned ○ positive ○ negative
image 3 ● unsigned ○ positive ○ negative
image 4 ● unsigned ○ positive ○ negative
image 5 ● unsigned ○ positive ○ negative
image 6 ● unsigned ○ positive ○ negative
image 7 ● unsigned ○ positive ○ negative
image 8 ● unsigned ○ positive ○ negative
image 9 ● unsigned ○ positive ○ negative
image 10 ● unsigned ○ positive ○ negative
image 11 ● unsigned ○ positive ○ negative
image 12 ● unsigned ○ positive ○ negative
image 13 ● unsigned ○ positive ○ negative
image 14 ● unsigned ○ positive ○ negative
image 15 ● unsigned ○ positive ○ negative
image 16 ● unsigned ○ positive ○ negative
image 17 ● unsigned ○ positive ○ negative
image 18 ● unsigned ○ positive ○ negative
image 19 ● unsigned ○ positive ○ negative
image 20 ● unsigned ○ positive ○ negative

Active Learning

image 1 ○ unsigned ● positive ○ negative
image 2 ○ unsigned ● positive ○ negative
image 3 ○ unsigned ○ positive ● negative
image 4 ○ unsigned ○ positive ● negative
image 5 ○ unsigned ○ positive ● negative

Fig. 1. User interface of a prototype system

relevant while most images are irrelevant. Therefore, Ssaira attempts to find a potentially better representative of the semantic class through slightly generalizing the negative examples. Concretely, the k-nearest neighboring unlabeled examples are identified for each negative example and then the feature vectors of these $k + 1$ examples are averaged to derive a virtual example, which is used by the learners in Ssaira instead of the original negative example. Ideally, the k-nearest neighboring examples should be identified in an appropriate subspace that can distinguish the concerned potential semantic class well, which can be implemented by kernel transformation in principle. However, since it is difficult to determine which kernel to use, this paper simply identifies the neighbors using the Euclidean distance in the original feature space.

In summary, the pseudo-code of Ssaira is presented in Table I, where the $Abs(a)$ function is used to produce the absolute value of a while the *Generalize* function is shown in Table II. Note that the function $Neighbors(\boldsymbol{x}, \mathcal{D}_u, k)$ can be computed off-line, that is, the neighbors of all the unlabeled examples can be pre-computed. Moreover, many steps of Ssaira, such as the loop of the 3rd step, can be executed in one scan of the image database. Therefore, Ssaira can be quite efficient in processing online queries. It is also worth noting that the labeled images provided by the user in the relevance feedback process are cumulatively used, which is helpful in enlarging the training sets of the learners.

It is evident that the Ssaira method addresses the three issues mentioned in Section 2.1, i.e. small sample size, asymmetrical training sample, and real time requirement. Concretely, Ssaira combines semi-supervised learning and active

Table I. Pseudo-code describing the SSAIRA method

```
SSAIRA($query$, $DB$, $L_1$, $L_2$, $k$, $poolsize$, $resultsize$)

    Input: $query$: User query
           $DB$: Image database
           $L_i$ ($i \in \{1..2\}$): Learners
           $k$: Number of neighbors used in generalizing negative examples
           $poolsize$: Number of images in the pool
           $resultsize$: Number of images to be returned

    $\mathcal{P} \leftarrow \{query\}$; $\mathcal{N} \leftarrow \emptyset$; $\mathcal{U} \leftarrow DB$
    In each round of relevance feedback:
    $Getfeedback(\mathcal{P}^*, \mathcal{N}^*)$
    $\mathcal{P} \leftarrow \mathcal{P} \cup \mathcal{P}^*$; $\mathcal{N} \leftarrow \mathcal{N} \cup \mathcal{N}^*$; $\mathcal{U} \leftarrow \mathcal{U} - (\mathcal{P}^* \cup \mathcal{N}^*)$
    for $i \in \{1..2\}$ do
        $\mathcal{P}_i \leftarrow \mathcal{P}$
        $\mathcal{N}_i \leftarrow \mathcal{N} \cup \{\arg\min_{\boldsymbol{x} \in \mathcal{U}} L_{(3-i)}(\boldsymbol{x}, \mathcal{P}, Generalize(\mathcal{N}, \mathcal{U}, k))\}$
    for $\boldsymbol{x} \in \mathcal{U}$ do $Rank(\boldsymbol{x}) \leftarrow \frac{1}{Z_{norm}} \sum_{i \in \{1..2\}} L_i(\boldsymbol{x}, \mathcal{P}_i, Generalize(\mathcal{N}_i, \mathcal{U}, k))$
    $Pool \leftarrow \emptyset$; $Result \leftarrow \emptyset$
    for $i \in \{1..poolsize\}$ do $Pool \leftarrow Pool \cup \{\arg\min_{\boldsymbol{x} \in \mathcal{U}} Abs(Rank(\boldsymbol{x}))\}$
    for $i \in \{1..resultsize\}$ do $Result \leftarrow Result \cup \{\arg\max_{\boldsymbol{x} \in \mathcal{U}} Rank(\boldsymbol{x})\}$

    Output: $Result$; $Pool$
```

Table II. Pseudo-code describing the GENERALIZE function

```
GENERALIZE($\mathcal{D}_n$, $\mathcal{D}_u$, $k$)

    Input: $\mathcal{D}_n$: A data set whose elements are to be generalized
           $\mathcal{D}_u$: A data set in which neighbors are to be identified
           $k$: Number of neighbors used in generalizing

    $\mathcal{D}^* \leftarrow \emptyset$
    for $\boldsymbol{x} \in \mathcal{D}_n$ do
        $\mathcal{D}' \leftarrow Neighbors(\boldsymbol{x}, \mathcal{D}_u, k)$      %% $\mathcal{D}'$ stores the $k$-nearest neighbors of $\boldsymbol{x}$ in $\mathcal{D}_u$
        $\mathcal{D}' \leftarrow \mathcal{D}' \cup \{\boldsymbol{x}\}$
        $\boldsymbol{x}' \leftarrow Ave(\mathcal{D}')$      %% $\boldsymbol{x}'$ is the average feature vector of the feature vectors in $\mathcal{D}'$
        $\mathcal{D}^* \leftarrow \mathcal{D}^* \cup \{\boldsymbol{x}'\}$

    Output: $\mathcal{D}^*$
```

learning to exploit unlabeled examples, adopts a special mechanism to generalize the negative examples, and employs very simple learners that can be updated efficiently. As for the technique used in exploiting unlabeled data, SSAIRA gracefully combines co-training and co-testing together. However, it is worth noting that the standard co-training [Blum and Mitchell 1998] and co-testing [Muslea et al. 2000] have a rigorous requirement of sufficient but redundant views. Unfortunately, in real-world applications such as CBIR it is not easy to get sufficient but redundant views to use. Since the mechanisms employed by SSAIRA do not require sufficient but redundant views, their applicability can be broader.

4. EXPERIMENTS

4.1 Comparison Methods

In the experiments SSAIRA is compared with BALAS, a semi-supervised learning method which has been applied to CBIR, recently proposed by Zhang and Zhang [2004]. This method explicitly addresses the small sample issue and the asymmetrical training sample issue by stretching Bayesian learning. Briefly, this method cumulatively uses labeled examples obtained in all rounds of the relevance feedback, and regards some unlabeled examples near the negative examples in a kernel space as additional negative examples, wishing that the kernel space can somewhat represent the semantic classes of the negative examples. It estimates the probability density function (PDF) of the positive class directly while regarding each negative example as a representative of a unique potential semantic class and using the agglomeration of all the negative PDFs as the overall PDF of negative class. Note that the PDFs and the *trustworthy degree*, which is used to weight the features, have to be estimated in the process of relevance feedback because the estimation process relies on the images queried and fed back by the user. Therefore, the running speed of BALAS is slower than that of SSAIRA.

SSAIRA is also compared with SVM$_{Active}$, an active learning method which has been applied to CBIR, proposed by Tong and Chang [2001]. This method exploits the margin of support vector machines for active learning. Firstly, it trains a support vector machine on the labeled examples. Then, the unlabeled images which are close to the support vector boundary are identified and passed to the user for feedback. Note that in each round of relevance feedback a support vector machine has to be trained.

The third method used in the comparison is called NAIVE in this paper, which is the standard relevant feedback method using the base learner of SSAIRA as shown in Eq. 1. After obtaining the labeled examples provided by the user, it searches the database to identify the images close to the positive examples while far from the negative ones according to Euclidean distance. It is evident that this method does not exploit unlabeled data.

Moreover, two degenerated variants of SSAIRA, i.e. SSIRA (Semi-Supervised Image Retrieval with Asymmetry) and AIRA (Active Image Retrieval with Asymmetry), are evaluated in the comparison. The SSIRA method is almost the same as SSAIRA except that SSIRA does not use active learning. Roughly speaking, this method can be obtained by omitting the 8th step in Table I. The AIRA method is almost the same as SSAIRA except that AIRA does not use semi-supervised learning. Roughly speaking, this method can be obtained by replacing the 5th step in Table I by $\mathcal{N}_i \leftarrow \mathcal{N}$.

Furthermore, in order to study whether the mechanism used in SSAIRA to address the asymmetrical training sample issue is useful, another series of degenerated variants are compared, including SSAIR, SSIR and AIR. The difference between SSAIR and SSAIRA is that the former uses the real negative examples directly instead of using virtual ones derived by generalizing the real negative examples. Roughly speaking, the SSAIR method can be obtained through replacing $Generalize(\mathcal{N},\mathcal{U},k)$ by $\mathcal{N}$ in the 5th step and replacing $Generalize(\mathcal{N}_i,\mathcal{U},k)$ by $\mathcal{N}_i$ in the 6th step in Table I. The difference between SSIR and SSIRA and the difference between AIR and AIRA are similar to that between SSAIR and SSAIRA. These degenerated variants share the same parameters of SSAIRA.

In addition, for investigating the influence of the distance metrics used by the two learners in SSAIRA on the retrieval performance, experiments are also performed to compare different versions of SSAIRA which have different distance settings, more concretely, using different p_i values in Eq. 2.

4.2 Evaluation Measures

The evaluation measures used in CBIR have been greatly affected by those used in text-based information retrieval [Müller et al. 2001]. A straightforward and popularly used measure is the *PR-graph* which depicts the relationship between *precision* and *recall* of a specific retrieval system. This measure is used in this paper. Concretely, for every recall value ranging from 0.0 to 1.0, the corresponding precision value is computed and then depicted in the PR-graph.

A deficiency with the PR-graph is that it can hardly reflect the changes of the retrieval performance caused by relevance feedback directly. In other words, in order to exhibit the changes of the retrieval performance caused by relevance feedback, a single PR-graph is not enough. Instead, a series of PR-graphs each graph corresponding to a round of relevance feedback has to be used. Therefore, another graphical measure is employed in this paper. Usually, a CBIR system exhibits a trade-off between precision and recall, to obtain high precision usually means sacrificing recall and vice versa. Considering that in CBIR both the precision and recall are of importance, here BEP (*Break-Event-Point*) is introduced into CBIR as an evaluation measure. By definition, if the precision and recall are tuned to have an equal value, then this value is called the BEP of the system [Lewis 1992]. The higher the BEP, the better the performance. Through connecting the BEPs after different rounds of relevance feedback, a *BEP-graph* is obtained, where the horizontal axis enumerates the round of relevance feedback while the vertical axis gives the BEP value.

In addition, a quantitative measure, i.e. *effectiveness*, is used. This measure was proposed by Mehtre et al. [1995], and then adopted to quantify the utility of relevance feedback mechanisms [Ciocca and Schettini 1999]. The definition is given by Eq. 3, where η_S denotes effectiveness, S denotes the number of relevant images the user wants to retrieve, and R_q^I and R_q^E, respectively, denote the set of relevant images and all images retrieved. The bigger the η_S, the better the performance.

$$\eta_S = \begin{cases} \left|R_q^I \cap R_q^E\right| / \left|R_q^I\right| & if \ \left|R_q^I\right| \leq S \\ \\ \left|R_q^I \cap R_q^E\right| / \left|R_q^E\right| & if \ \left|R_q^I\right| > S \end{cases} \tag{3}$$

4.3 Configurations

One hundred classes of COREL images are used, where each class has 100 images and therefore there are 10,000 images in total. These images are organized into two image databases. The first database contains 20 classes of images (denoted by $C = 20$), and therefore its size is 2,000. The second database contains all the images (denoted by $C = 100$). Experiments are performed on these two databases, and therefore the performance of the compared methods on small and big image databases can be studied.

In the experiments color, texture, and shape features are used to describe the images. The color features are derived from a histogram computed from the H and V components in the HSV space, where the H and V components are equally partitioned into eight and four bins, respectively [Seung et al. 1992]. The texture features are derived from Gabor wavelet transformation according to [Manjunath and Ma 1996]. The shape features are the same as those used in [Wang et al. 2002]. These features are empirically biased by multiplying a weight of 0.5 with color features, 0.3 with texture features, and 0.2 with shape features, respectively. That is, the similarities obtained by comparing the color, texture, and shape features, respectively, are weighted-summed to derive the overall similarity. Note that feature selection methods are usually beneficial to Cbir, but here no feature selection is executed and just a simple weighted sum scheme is used to play with the features. This is because the same set of features and weights will be used by all the compared methods and the relative instead of absolute performance of these compared methods are of concern in the experiments.

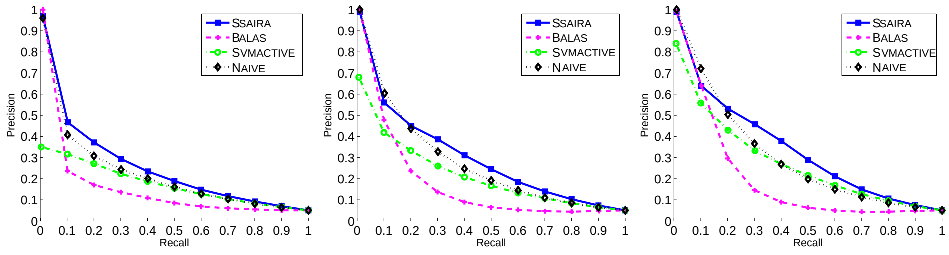
As for Ssaira, the parameter k in Table I is set to 10 in the experiments, and the orders of the Minkowski distance used by the two learners are set to 1 and 2 by default. As for Balas, the parameter w used in dealing with positive examples is set to 0.4, the parameter q used in its Sampling process for dealing with negative examples is set to 5, and the other parameters are set as the same as those set in [Zhang and Zhang 2004]. As for Svm$_{Active}$, an Rbf kernel with $\gamma = 1$ is used.

For each compared method, after obtaining a query, five rounds of relevance feedback are performed. In each round the user can label F (= 5, 7, or 9) images as the feedback. For each query, the process is repeated five times with different users. Moreover, the whole process is repeated five times with different queries. The average results are recorded. The experiments are conducted on a Pentium 4 machine with 3.00GHz Cpu and 1GB memory.

4.4 Results

At first, the performance of Ssaira, Balas, Svm$_{Active}$, and Naive are compared. The geometrical PR-graphs at the 0th, 3rd, and 5th round of relevance feedbacks when $F = 5$, 7, and 9 and $C = 20$ are shown in Figs. 2 to 4, respectively. The geometrical Bep-graphs are presented in Fig. 5, and the geometrical effectivenesses are tabulated in Table III where the best performance at each round of relevance feedback has been boldfaced. *Geometrical* means the results obtained after averaging across all the image classes. Note that the performance at the 0th round corresponds to the performance before starting relevance feedback, that is, the retrieval performance with only the initial query.

Figs. 2 to 5 and Table III explicitly show that Ssaira is better than Balas, Svm$_{Active}$, and Naive when $C = 20$. It is impressive that at all rounds of relevance feedback, the geometrical effectiveness of Ssaira is always the best. Note that the performance of Naive is not bad, which verifies the usefulness of the base learner used in Ssaira. The performances of Balas and Svm$_{Active}$ are not as excellent

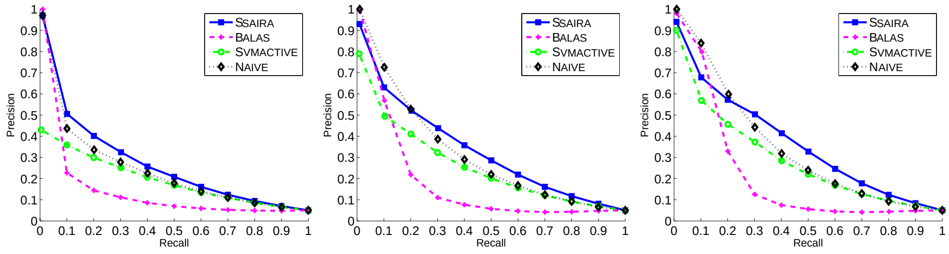


(a) at the 0th round

(b) at the 3rd round

(c) at the 5th round

Fig. 2. Geometrical PR-graphs of SSAIRA, BALAS, SVM$_{Active}$, and NAIVE at the 0th, 3rd, and 5th rounds ($F = 5$, $C = 20$)

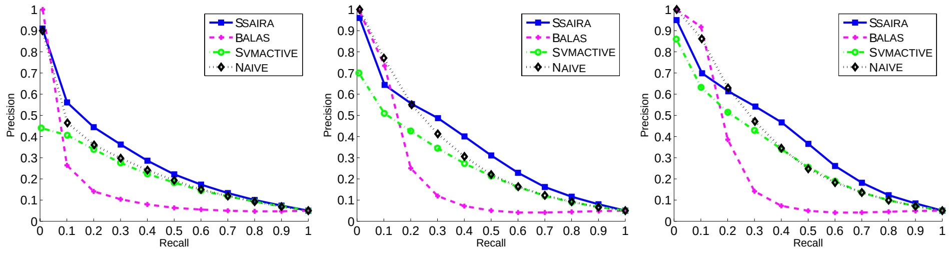


(a) at the 0th round

(b) at the 3rd round

(c) at the 5th round

Fig. 3. Geometrical PR-graphs of SSAIRA, BALAS, SVM$_{Active}$, and NAIVE at the 0th, 3rd, and 5th rounds ($F = 7$, $C = 20$)



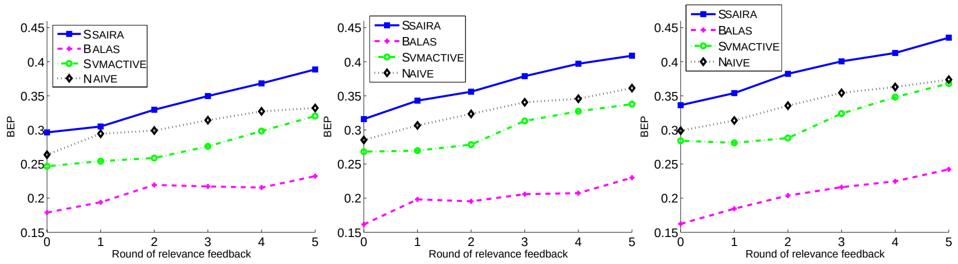
(a) at the 0th round

(b) at the 3rd round

(c) at the 5th round

Fig. 4. Geometrical PR-graphs of SSAIRA, BALAS, SVM$_{Active}$, and NAIVE at the 0th, 3rd, and 5th rounds ($F = 9$, $C = 20$)

as those reported in [Zhang and Zhang 2004; Tong and Chang 2001], which may be because when only a limited number of images is labeled during the relevance feedback process, the probability density estimation employed by BALAS and the SVM employed by SVM$_{Active}$ can hardly get sufficient labeled examples to use and therefore their performance degenerates. At first glance, the figures with different F values look very similar, which indicates that the relative performances of the compared methods are quite consistent no matter which F value is taken. Moreover, the figures suggest that the benefit from increasing the number of feedbacks in each round is not as apparent as increasing the rounds of feedbacks.



(a) $F = 5$ (b) $F = 7$ (c) $F = 9$

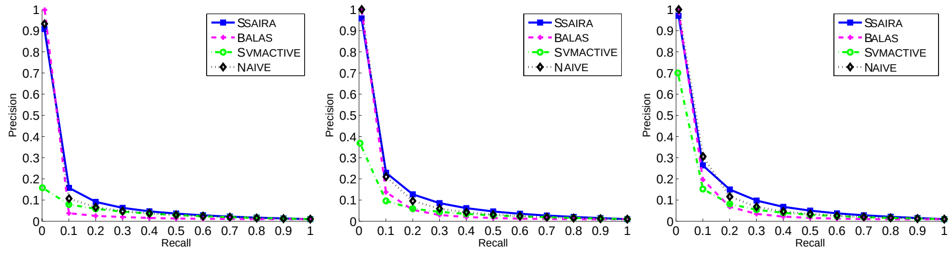
Fig. 5. Geometrical BEP-graphs of SSAIRA, BALAS, SVM$_{Active}$, and NAIVE ($C = 20$)

Table III. Geometrical effectivenesses of SSAIRA (SA), BALAS (B), SVM$_{Active}$ (V), and NAIVE (N) when $C = 20$

	$F = 5$				$F = 7$				$F = 9$			
	SA	B	V	N	SA	B	V	N	SA	B	V	N
$\bar{\eta}^0_{200}$(%)	**43.8**	28.5	38.7	40.2	**45.8**	25.4	40.8	42.7	**47.7**	24.4	42.5	44.4
$\bar{\eta}^1_{200}$(%)	**45.0**	28.5	39.6	41.6	**49.6**	28.0	41.8	44.0	**50.3**	25.6	43.1	44.5
$\bar{\eta}^2_{200}$(%)	**46.8**	29.9	40.3	43.1	**51.1**	26.9	42.9	46.4	**53.0**	26.6	44.6	46.7
$\bar{\eta}^3_{200}$(%)	**49.7**	29.1	40.9	44.2	**52.9**	26.6	45.4	47.3	**54.6**	27.8	46.7	47.7
$\bar{\eta}^4_{200}$(%)	**52.0**	28.8	43.9	45.1	**54.6**	26.0	46.8	47.6	**55.9**	27.8	48.5	49.2
$\bar{\eta}^5_{200}$(%)	**53.2**	29.7	46.7	45.7	**56.1**	28.4	47.5	49.3	**57.6**	29.5	50.5	49.9

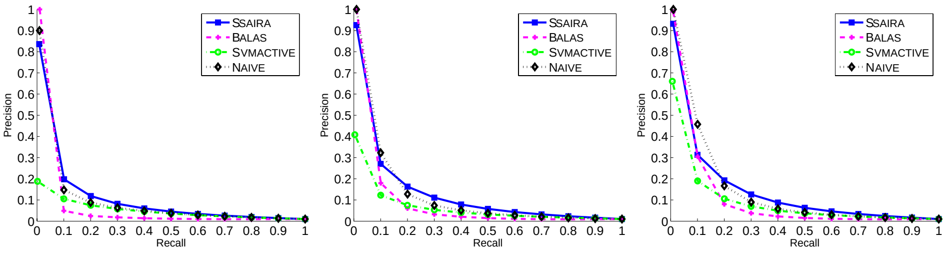
The geometrical PR-graphs at the 0th, 3rd, and 5th round of relevance feedbacks when $F = 5$, 7, and 9 and $C = 100$ are shown in Figs. 6 to 8, respectively. The geometrical BEP-graphs are presented in Fig. 9, and the geometrical effectivenesses are tabulated in Table IV where the best performance at each round of relevance feedback has been boldfaced.

Comparing Figs. 2 to 5 and Table III with Figs. 6 to 9 and Table IV shows that on the bigger image database ($C = 100$), the performance of all the compared methods degenerates. For example, the geometrical effectiveness of SSAIRA almost drops by half when the image database changes from $C = 20$ to $C = 100$. However,



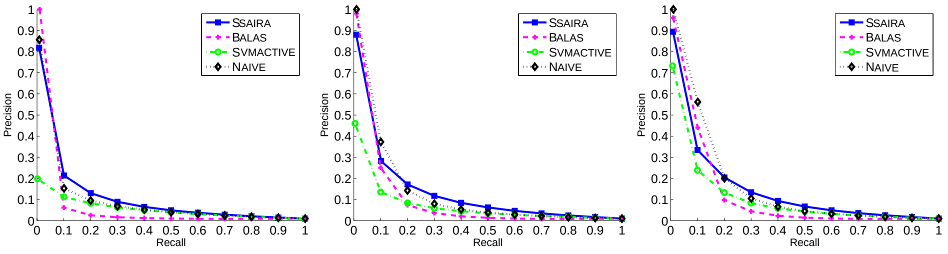
(a) at the 0th round (b) at the 3rd round (c) at the 5th round

Fig. 6. Geometrical PR-graphs of SSAIRA, BALAS, SVM$_{Active}$, and NAIVE at the 0th, 3rd, and 5th rounds ($F = 5$, $C = 100$)



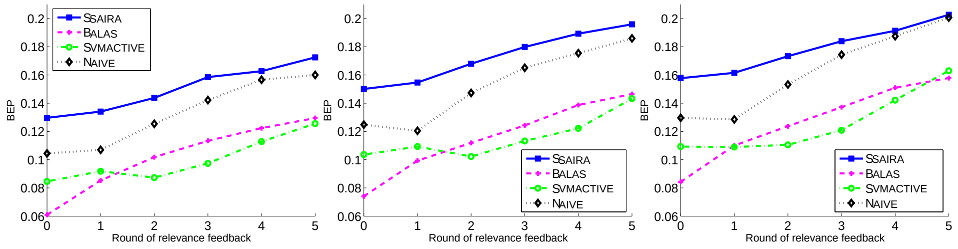
(a) at the 0th round (b) at the 3rd round (c) at the 5th round

Fig. 7. Geometrical PR-graphs of SSAIRA, BALAS, SVM$_{Active}$, and NAIVE at the 0th, 3rd, and 5th rounds ($F = 7$, $C = 100$)



(a) at the 0th round (b) at the 3rd round (c) at the 5th round

Fig. 8. Geometrical PR-graphs of SSAIRA, BALAS, SVM$_{Active}$, and NAIVE at the 0th, 3rd, and 5th rounds ($F = 9$, $C = 100$)



(a) $F = 5$ (b) $F = 7$ (c) $F = 9$

Fig. 9. Geometrical BEP-graphs of SSAIRA, BALAS, SVM$_{Active}$, and NAIVE ($C = 100$)

by studying detailed results, it has been found that even when $C = 100$, the effectiveness of SSAIRA on some image classes is not bad, e.g. on *flags* and *music_ads* the effectiveness is close to 90.0%. The low geometrical effectiveness may have two reasons: The first is that with the increasing number of image classes, the retrieval task becomes more difficult. The second is that some image classes of COREL, e.g. *Africa* and *christmas*, do not really have consistent image content, that is, the images in these classes are grouped based on semantic rather than low-level features. Nevertheless, Figs. 6 to 9 and Table IV show that the performance of SSAIRA is still better than that of BALAS, SVM$_{active}$, and NAIVE.

The geometrical time costs of SSAIRA, BALAS, SVM$_{Active}$, and NAIVE spent in

Table IV. Geometrical effectivenesses of Ssaira (SA), Balas (B), Svm$_{Active}$ (V), and Naive (N) when $C = 100$

	$F = 5$				$F = 7$				$F = 9$			
	SA	B	V	N	SA	B	V	N	SA	B	V	N
$\bar{\eta}_{200}^{0}(\%)$	**19.0**	8.6	13.9	15.7	**22.1**	9.9	16.7	18.6	**23.2**	10.9	17.6	19.5
$\bar{\eta}_{200}^{1}(\%)$	**20.0**	12.3	14.7	15.3	**22.7**	13.4	17.3	16.8	**23.5**	14.6	17.6	17.8
$\bar{\eta}_{200}^{2}(\%)$	**20.9**	14.2	14.0	17.7	**24.4**	15.1	16.5	20.3	**25.2**	16.4	17.7	20.9
$\bar{\eta}_{200}^{3}(\%)$	**22.7**	15.2	14.9	19.7	**26.0**	16.4	17.1	22.3	**26.7**	18.0	18.2	23.3
$\bar{\eta}_{200}^{4}(\%)$	**23.5**	16.2	16.8	21.2	**27.0**	17.8	18.0	23.6	**27.3**	19.2	20.6	24.8
$\bar{\eta}_{200}^{5}(\%)$	**24.6**	17.3	18.1	21.4	**27.7**	18.5	20.7	24.5	**28.6**	19.9	23.1	26.3

Table V. Geometrical time costs (seconds) of Ssaira, Balas, Svm$_{Active}$, and Naive spent in each round of relevance feedback when $C = 100$

	$F = 5$	$F = 7$	$F = 9$
Ssaira	0.545	0.734	0.820
Balas	1.605	2.048	2.495
Svm$_{Active}$	0.131	0.196	0.271
Naive	**0.107**	**0.140**	**0.180**

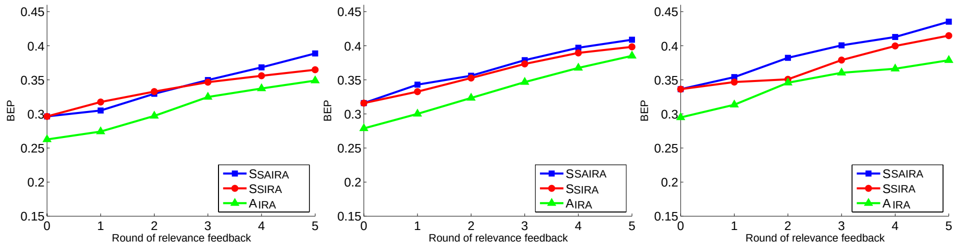
each round of relevance feedback when $C = 100$ are compared in Table V, where the smallest time cost under each F value has been boldfaced.

Table V reveals that Naive, the base learner of Ssaira, is the most efficient, and although Ssaira is not as efficient as Svm$_{Active}$, it is much more efficient than Balas. Considering that the retrieval performance of Ssaira is better than Balas, Svm$_{Active}$, and Naive, the above presented experiments confirm that Ssaira is the best among the compared methods.

In order to study whether the semi-supervised learning and active learning mechanisms employed by Ssaira are beneficial or not, the Ssaira method is compared with Ssira and Aira. The geometrical Bep-graphs when $C = 20$ and 100 are presented in Figs. 10 and 11, and the geometrical effectivenesses are tabulated in Tables VI and VII, respectively.

Figs. 10, 11 and Tables VI, VII show that when $C = 20$, Ssira is better than Aira, but when $C = 100$, the performance of Ssira degenerates as the round of relevance feedback increases. However, it is noteworthy that no matter whether $C = 20$ or 100, the performance of Ssaira is always better than that of Ssira and Aira.

Recall that the semi-supervised learning mechanism of Ssaira picks the most confident negative examples to use, which implies that Ssira only uses one positive example, i.e. the initial *query*. It is evident that when there are lots of negative classes, using only one positive example is not sufficient to distinguish the positive class, and therefore the performance of Ssira degenerates. When both the active learning and semi-supervised learning mechanisms are used, since additional positive examples can be obtained via active learning, the Ssaira method in fact gets more positive examples as well as more negative examples to use. Thus, its perfor-

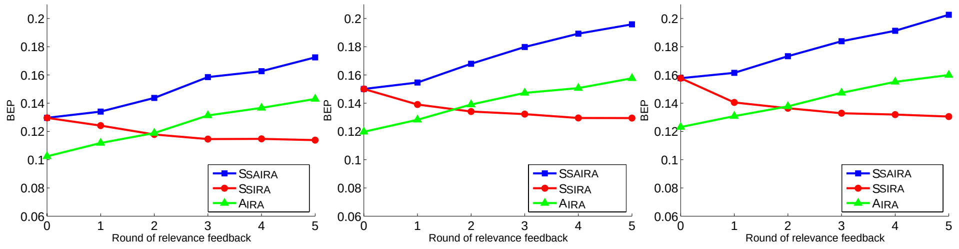


(a) $F = 5$ (b) $F = 7$ (c) $F = 9$

Fig. 10. Geometrical BEP-graphs of SSAIRA, SSIRA, and AIRA when $C = 20$

Table VI. Geometrical effectivenesses of SSAIRA (SA), SSIRA (S), and AIRA (A) when $C = 20$

	$F = 5$			$F = 7$			$F = 9$		
	SA	S	A	SA	S	A	SA	S	A
$\bar{\eta}^0_{200}(\%)$	**43.8**	**43.8**	40.1	**45.8**	**45.8**	42.1	**47.7**	**47.7**	43.7
$\bar{\eta}^1_{200}(\%)$	**45.0**	44.5	40.8	**49.6**	47.4	44.9	**50.3**	48.3	46.3
$\bar{\eta}^2_{200}(\%)$	**46.8**	46.6	44.1	**51.1**	49.1	48.3	**53.0**	49.0	49.5
$\bar{\eta}^3_{200}(\%)$	**49.7**	47.5	46.8	**52.9**	50.8	50.2	**54.6**	51.3	51.1
$\bar{\eta}^4_{200}(\%)$	**52.0**	48.7	48.3	**54.6**	51.8	51.9	**55.9**	53.3	52.0
$\bar{\eta}^5_{200}(\%)$	**53.2**	48.7	48.9	**56.1**	52.7	53.4	**57.6**	54.3	53.4



(a) $F = 5$ (b) $F = 7$ (c) $F = 9$

Fig. 11. Geometrical BEP-graphs of SSAIRA, SSIRA, and AIRA when $C = 100$

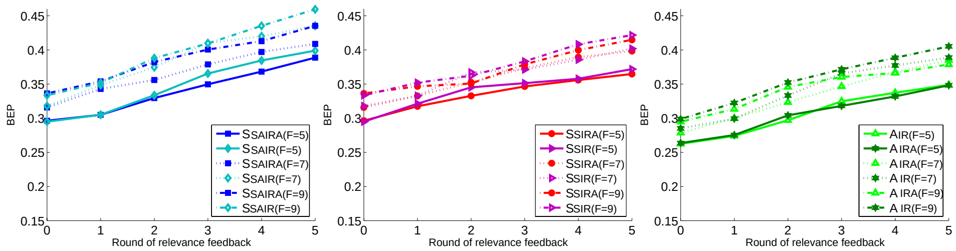
Table VII. Geometrical effectivenesses of SSAIRA (SA), SSIRA (S), and AIRA (A) when $C = 100$

	$F = 5$			$F = 7$			$F = 9$		
	SA	S	A	SA	S	A	SA	S	A
$\bar{\eta}^0_{200}(\%)$	**19.0**	**19.0**	15.4	**22.1**	**22.1**	18.1	**23.2**	**23.2**	18.9
$\bar{\eta}^1_{200}(\%)$	**20.0**	18.7	16.7	**22.7**	20.7	19.3	**23.5**	21.1	19.8
$\bar{\eta}^2_{200}(\%)$	**20.9**	18.0	17.8	**24.4**	20.1	20.7	**25.2**	20.5	20.9
$\bar{\eta}^3_{200}(\%)$	**22.7**	17.5	19.1	**26.0**	19.9	21.7	**26.7**	19.9	22.0
$\bar{\eta}^4_{200}(\%)$	**23.5**	17.5	19.7	**27.0**	19.4	22.2	**27.3**	19.7	22.8
$\bar{\eta}^5_{200}(\%)$	**24.6**	17.4	20.6	**27.7**	19.3	23.0	**28.6**	19.7	23.5

mance can be better than that of Ssira and Aira. In other words, the above observations show that the mechanisms of active learning and semi-supervised learning should be used simultaneously, especially when handling big image databases.

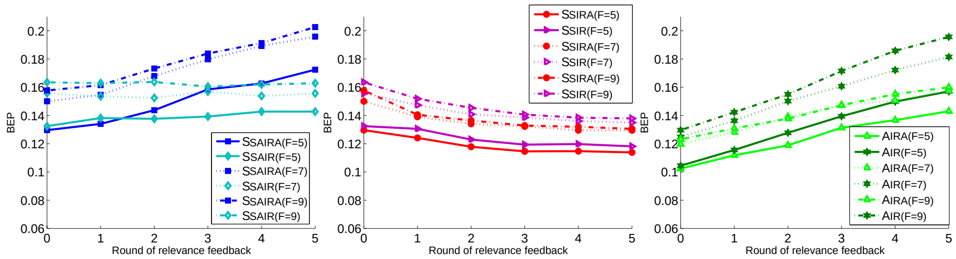
On the other hand, Tables VI and VII show that in the early rounds of relevance feedback, semi-supervised learning contributes more to Ssaira while in the later rounds active learning contributes more. This is not difficult to understand because in the early rounds most images in the retrieval results have not been learned well, therefore randomly picking some images to label in the relevance feedback process is not very different from active feedback. That is, the active learning mechanism is not very helpful in the early rounds. However, as relevance feedback continues, the number of well-learned images increases and therefore randomly picking an image to label is less likely helpful. Thus, the active learning mechanism is more valuable in the later rounds.

Further, in order to study whether the mechanism of dealing with negative examples employed by Ssaira is helpful or not, Ssaira is compared with Ssair, Ssira is compared with Ssir, and Aira is compared with Air. Note that in each pair, the former method uses this mechanism of dealing with negative examples while the latter does not. The geometrical Bep-graphs when $C = 20$ and 100 are plotted in Figs. 12 and 13, respectively.



(a) Ssaira vs. Ssair (b) Ssira vs. Ssir (c) Aira vs. Air

Fig. 12. Geometrical Bep-graphs of Ssaira vs. Ssair, Ssira vs. Ssir, and Aira vs. Air when $C = 20$

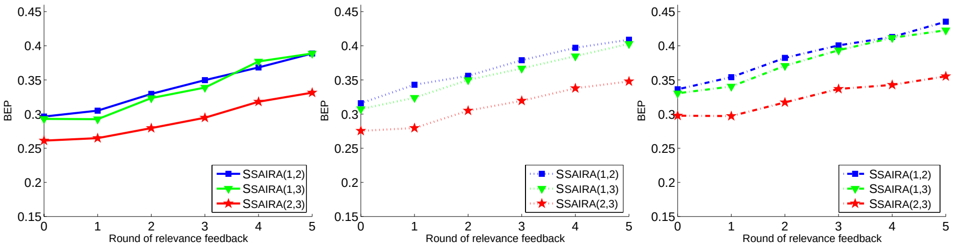


(a) Ssaira vs. Ssair (b) Ssira vs. Ssir (c) Aira vs. Air

Fig. 13. Geometrical Bep-graphs of Ssaira vs. Ssair, Ssira vs. Ssir, and Aira vs. Air when $C = 100$

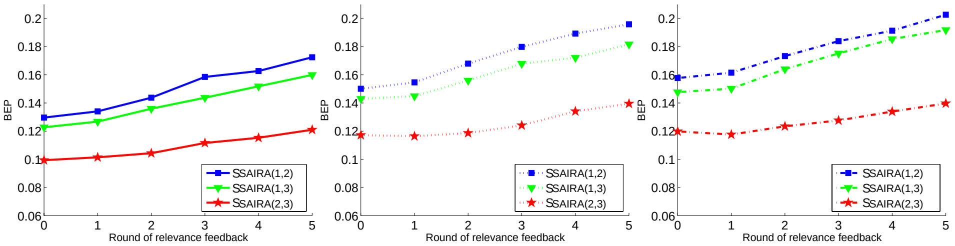
Figs. 12 and 13 show that using the mechanism of dealing with negative examples usually does not help, but when $C = 100$ this mechanism is beneficial to Ssaira with the increasing of the round of relevance feedback. It is conjectured that this is because when there are lots of negative classes, the negative examples are more apt to cluster together, and since Ssaira gets more positive examples as well as more negative examples to use via the active learning and semi-supervised learning mechanisms as the round of relevance feedback increases, the identification of neighboring negative examples for a concerned negative example becomes more reliable.

For studying the influence of the distance metrics used by the two learners in Ssaira on retrieval performance, different versions of Ssaira which are facilitated with different distance settings are evaluated. In detail, Ssaira(1,2), Ssaira(1,3), and Ssaira(2,3) are compared, where Ssaira(a,b) means that the a-order and b-order Minkowski distances are used by the two learners, respectively. The geometrical Bep-graphs when $C = 20$ and 100 are plotted in Figs. 14 and 15, respectively.



(a) $F = 5$ (b) $F = 7$ (c) $F = 9$

Fig. 14. Geometrical Bep-graphs of Ssaira(1,2), Ssaira(1,3) and Ssaira(2,3) when $C = 20$



(a) $F = 5$ (b) $F = 7$ (c) $F = 9$

Fig. 15. Geometrical Bep-graphs of Ssaira(1,2), Ssaira(1,3) and Ssaira(2,3) when $C = 100$

Figs. 14 and 15 show that the performance of Ssaira(1,3) is close to that of Ssaira(1,2), but the performance of Ssaira(2,3) is apparently worse. Recall the property of the order of Minkowski distance, that is, the smaller the order, the more robust the resulting distance metric to data variations; while the bigger the order, the more sensitive the resulting distance metric to data variations. It is evident that in Cbir, due to the gap between the high-level image semantics and low-level image features, the distance metric should not be very sensitive to data variations. Therefore, using the first- and second-order Minkowski distances should be a better choice for Ssaira, which is confirmed by Figs. 14 and 15.

5. CONCLUSION

The research reported here extends a preliminary paper [Zhou et al. 2004], which advocates using semi-supervised learning and active learning together to exploit unlabeled images existing in the database to enhance the performance of relevant feedback in Cbir. Concretely, this paper attempts to address three special issues of relevance feedback, i.e. small sample size, asymmetrical training sample, and real time requirement. Experiments show that the proposed Ssaira method is superior to some existing methods, and employing semi-supervised learning and active learning simultaneously is beneficial to the improvement of the retrieval performance.

Although the utility of Ssaira has been verified by experiments, there is a lack of theoretical analysis. This might have encumbered the exertion of the full power of Ssaira. For example, in the current form of Ssaira, in each round of relevance feedback each learner only labels the two most confident negative images for the other learner. If theoretical analysis on the relationship between the performance of the learners and the possible noise in the labeling process is available, it might be found that letting each learner label more images, including negative as well as positive images, can be beneficial, which may help improve the performance of Ssaira. This is an important issue for future work.

Ssaira addresses the asymmetrical training sample problem by generalizing the negative examples using their neighboring negative examples. Intuitively, negative examples belonging to the same negative class should distribute closely because they share some common properties. However, for different negative examples, the number of neighbors belonging to the same negative classes are usually different. Therefore, using an adaptive instead of fixed neighborhood area is more desirable. It is evident that designing better schemes for dealing with asymmetrical training sample is an important future issue.

Evaluation measures are important in the research of Cbir. This paper uses *PR-graphs* and *effectiveness* to measure retrieval performance. Besides, considering that PR-graphs can hardly reflect the changes of the retrieval performance caused by relevance feedback directly, this paper introduces *Bep* into Cbir and designs the *Bep-graph*. Note that some recent research reveals that the size of the relevant image classes and the number of retrieved images have influence on the evaluation of precision and recall, and therefore the *generality*, i.e. the relevant fraction, should be taken into account [Huijsmans and Sebe 2005]. Using the *GRiP* or *GReP* graphs [Huijsmans and Sebe 2005] to measure the performance of Ssaira and other methods is an issue left for future work. Moreover, studying the deficiencies of current evaluation measures and developing other powerful measures for Cbir are important issues to be investigated in future work.

Note that the performance of semi-supervised learning methods are usually unstable because the unlabeled examples may be wrongly labeled during the learning process [Nigam et al. 2000; Hwa et al. 2003]. It has been shown that when the model assumption does not match the ground-truth, unlabeled data can either improve or degrade the performance, depending on the complexity of the classifier compared with the size of the labeled training set [Cozman and Cohen 2002; Cohen et al. 2004]. Moreover, if the distribution of the unlabeled data is different from that of the labeled data, using unlabeled data may degrade the performance [Tian et al. 2004]. Therefore, if some unlabeled data satisfying the model assumption and the distribution of labeled data can be identified, using them in semi-supervised learning might be better than simply trying to use all the unlabeled data or randomly picking some to use, which is another issue to be explored in the future.

Acknowledgement

The comments and suggestions from the associate editor and the anonymous reviewers greatly improved the paper.

REFERENCES

Abe, N. and Mamitsuka, H. 1998. Query learning strategies using boosting and bagging. In *Proceedings of the 15th International Conference on Machine Learning.* Madison, WI, 1–9.

Blum, A. and Chawla, S. 2001. Learning from labeled and unlabeled data using graph mincuts. In *Proceedings of the 18th International Conference on Machine Learning.* Williamston, MA, 19–26.

Blum, A. and Mitchell, T. 1998. Combining labeled and unlabeled data with co-training. In *Proceedings of the 11th Annual Conference on Computational Learning Theory.* Madison, WI, 92–100.

Bookstein, A. 1983. Information retrieval: A sequential learning process. *Journal of the American Society for Information Science 34,* 4, 331–342.

Chen, J.-Y., Bouman, C. A., and Dalton, J. 2000. Hierarchical browsing and search of large image databases. *IEEE Transactions on Image Processing 9,* 3, 442–445.

Ciocca, G. and Schettini, R. 1999. A relevance feedback mechanism for content-based image retrieval. *Information Processing and Management 35,* 5, 605–632.

Cohen, I., Cozman, F. G., Sebe, N., Cirelo, M. C., and Huang, T. S. 2004. Semisupervised learning of classifiers: Theory, algorithm, and their application to human-computer interaction. *IEEE Transactions on Pattern Analysis and Machine Intelligence 26,* 12, 1553–1567.

Cox, I. J., Miller, M., Minka, T. P., Papathomas, T., and Yianilos, P. 2000. The Bayesian image retrieval system, PicHunter: Theory, implementation, and psychophysical experiments. *IEEE Transactions on Image Processing 9,* 1, 20–37.

Cozman, F. G. and Cohen, I. 2002. Unlabeled data can degrade classificaion performance of generative classifiers. In *Proceedings of the 15th International Conference of the Florida Artificial Intelligence Research Society.* Pensacola, FL, 327–331.

Dasgupta, S., Littman, M., and McAllester, D. 2002. PAC generalization bounds for co-training. In *Advances in Neural Information Processing Systems 14*, T. G. Dietterich, S. Becker, and Z. Ghahramani, Eds. MIT Press, Cambridge, MA, 375–382.

Dempster, A. P., Laird, N. M., and Rubin, D. B. 1977. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society, Series B 39,* 1, 1–38.

Dong, A. and Bhanu, B. 2003. A new semi-supervised EM algorithm for image retrieval. In *Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition.* Madison, WI, 662–667.

Goldman, S. and Zhou, Y. 2000. Enhancing supervised learning with unlabeled data. In *Proceedings of the 17th International Conference on Machine Learning.* San Francisco, CA, 327–334.

Huijsmans, D. P. and Sebe, N. 2005. How to complete performance graphs in content-based image retrieval: Add generality and normalize scope. *IEEE Transactions on Pattern Analysis and Machine Intelligence 27,* 2, 245–251.

Hwa, R., Osborne, M., Sarkar, A., and Steedman, M. 2003. Corrected co-training for statistical parsers. In *Working Notes of the ICML'03 Workshop on the Continuum from Labeled to Unlabeled Data in Machine Learning and Data Mining*. Washington, DC.

Ishikawa, Y., Subramanya, R., and Faloutsos, C. 1998. MindReader: Query databases through multiple examples. In *Proceedings of the 24th International Conference on Very Large Data Bases*. New York, NY, 218–227.

Joachims, T. 1999. Transductive inference for text classification using support vector machines. In *Proceedings of the 16th International Conference on Machine Learning*. Bled, Slovenia, 200–209.

Kherfi, M. L., Ziou, D., and Bernardi, A. 2002. Learning from negative example in relevance feedback for content-based image retrieval. In *Proceedings of the 16th International Conference on Pattern Recognition*. Quebec, Canada, 933–936.

Lewis, D. 1992. Representation and learning in information retrieval. Ph.D. thesis, Department of Computer Science, University of Massachusetts, Amherst, MA.

Lewis, D. and Gale, W. 1994. A sequential algorithm for training text classifiers. In *Proceedings of the 17th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*. Dublin, Ireland, 3–12.

Manjunath, B. S. and Ma, W. Y. 1996. Texture features for browsing and retrieval of image data. *IEEE Transactions on Pattern Analysis and Machine Intelligence 18,* 8, 837–842.

Mehtre, B. M., Kankanhalli, M. S., Narasimhalu, A. D., and Man, G. C. 1995. Color matching for image retrieval. *Pattern Recognition Letters 16,* 3, 325–331.

Miller, D. J. and Uyar, H. S. 1997. A mixture of experts classifier with learning based on both labelled and unlabelled data. In *Advances in Neural Information Processing Systems 9*, M. Mozer, M. I. Jordan, and T. Petsche, Eds. MIT Press, Cambridge, MA, 571–577.

Müller, H., Müller, W., Squire, D. M., Marchand-Maillet, S., and Pun, T. 2001. Performance evaluation in content-based image retrieval: Overview and proposals. *Pattern Recognition Letters 22,* 5, 593–601.

Muslea, I., Minton, S., and Knoblock, C. A. 2000. Selective sampling with redundant views. In *Proceedings of the 17th National Conference on Artificial Intelligence*. Austin, TX, 621–626.

Nastar, C., Mitschke, M., and Meilhac, C. 1998. Efficient query refinement for image retrieval. In *Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition*. Santa Barbara, CA, 547–552.

Nigam, K. and Ghani, R. 2000. Analyzing the effectiveness and applicability of co-training. In *Proceedings of the 9th ACM International Conference on Information and Knowledge Management*. Washington, DC, 86–93.

Nigam, K., McCallum, A. K., Thrun, S., and Mitchell, T. 2000. Text classification from labeled and unlabeled documents using EM. *Machine Learning 39,* 2-3, 103–134.

Picard, R. W., Minka, T. P., and Szummer, M. 1996. Modeling user subjectivity in image libraries. In *Proceedings of the International Conference on Image Processing*. Lausanne, Switzerland, 777–780.

Pierce, D. and Cardie, C. 2001. Limitations of co-training for natural language learning from large data sets. In *Proceedings of the 2001 Conference on Empirical Methods in Natural Language Processing*. Pittsburgh, PA, 1–9.

Rui, Y., Huang, T. S., Ortega, M., and Mehrotra, S. 1998. Relevance feedback: A power tool for interactive content-based image retrieval. *IEEE Transactions on Circuits and Systems for Video Technology 8,* 5, 644–655.

Sarkar, A. 2001. Applying co-training methods to statistical parsing. In *Proceedings of the 2nd Annual Meeting of the North American Chapter of the Association for Computational Linguistics*. Pittsburgh, PA, 95–102.

Seung, H., Opper, M., and Sompolinsky, H. 1992. Query by committee. In *Proceedings of the 5th ACM Workshop on Computational Learning Theory*. Pittsburgh, PA, 287–294.

Shahshahani, B. and Landgrebe, D. 1994. The effect of unlabeled samples in reducing the small sample size problem and mitigating the hughes phenomenon. *IEEE Transactions on Geoscience and Remote Sensing 32,* 5, 1087–1095.

Smeulders, A. W. M., Worring, M., Santini, S., Gupta, A., and Jain, R. 2000. Content-based image retrieval at the end of the early years. *IEEE Transactions on Pattern Analysis and Machine Intelligence 22,* 12, 1349–1380.

Steedman, M., Osborne, M., Sarkar, A., Clark, S., Hwa, R., Hockenmaier, J., Ruhlen, P., Baker, S., and Crim, J. 2003. Bootstrapping statistical parsers from small data sets. In *Proceedings of the 11th Conference on the European Chapter of the Association for Computational Linguistics*. Budapest, Hungary, 331–338.

Tian, Q., Yu, J., Xue, Q., and Sebe, N. 2004. A new analysis of the value of unlabeled data in semi-supervised learning for image retrieval. In *Proceedings of the IEEE International Conference on Multimedia Expo*. Taibei, 1019–1022.

Tieu, K. and Viola, P. 2000. Boosting image retrieval. In *Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition*. Hilton Head, SC, 228–235.

Tong, S. and Chang, E. 2001. Support vector machine active learning for image retrieval. In *Proceedings of the 9th ACM International Conference on Multimedia*. Ottawa, Canada, 107–118.

Vasconcelos, N. and Lippman, A. 2000. Learning from user feedback in image retrieval systems. In *Advances in Neural Information Processing Systems 12,* S. A. Solla, T. K. Leen, and K.-R. Müller, Eds. MIT Press, Cambridge, MA, 977–986.

Wang, H. F., Jin, X. Y., and Sun, Z. 2002. Semantic image retrieval (in Chinese). *Journal of Computer Research and Development 39,* 5, 513–523.

Wu, Y., Tian, Q., and Huang, T. S. 2000. Discriminant-EM algorithm with application to image retrieval. In *Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition*. Hilton Head, SC, 222–227.

Yao, J. and Zhang, Z. 2005. Object detection in aerial imagery based on enhanced semi-supervised learning. In *Proceedings of the 10th IEEE International Conference on Computer Vision*. Beijing, China, 1012–1017.

Zhang, C. and Chen, T. 2002. An active learning framework for content-based information retrieval. *IEEE Transactions on Multimedia 4,* 2, 260–268.

Zhang, R. and Zhang, Z. 2004. Stretching Bayesian learning in the relevance feedback of image retrieval. In *Proceedings of the 8th European Conference on Computer Vision*. Prague, Czech, 355–367.

Zhou, X. S. and Huang, T. S. 2001. Small sample learning during multimedia retrieval using BiasMap. In *Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition*. Kauai, HI, 11–17.

Zhou, X. S. and Huang, T. S. 2003. Relevance feedback in image retrieval: A comprehensive review. *Multimedia Systems 8,* 6, 536–544.

Zhou, Z.-H., Chen, K.-J., and Jiang, Y. 2004. Exploiting unlabeled data in content-based image retrieval. In *Proceedings of the 15th European Conference on Machine Learning*. Pisa, Italy, 525–536.

Zhou, Z.-H. and Li, M. 2005a. Semi-supervised learning with co-training. In *Proceedings of the 19th International Joint Conference on Artificial Intelligence*. Edinburgh, Scotland, 908–913.

Zhou, Z.-H. and Li, M. 2005b. Tri-training: Exploiting unlabeled data using three classifiers. *IEEE Transactions on Knowledge and Data Engineering 17,* 11, 1529–1541.

Received September 2005; revised December 2005, March 2006; accepted March 2006