

Exploiting Unlabeled Data in Content-Based Image Retrieval

Zhi-Hua Zhou, Ke-Jia Chen, and Yuan Jiang

National Laboratory for Novel Software Technology
Nanjing University, Nanjing 210093, China
{zhouzh, chenkj, jiangy}@lamda.nju.edu.cn

Abstract. In this paper, the SSAIR (Semi-Supervised Active Image Retrieval) approach, which attempts to exploit unlabeled data to improve the performance of content-based image retrieval (CBIR), is proposed. This approach combines the merits of semi-supervised learning and active learning. 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 classifies the unlabeled images in the database and passes the most relevant/irrelevant images to 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 relevant with high confidence are returned as the retrieval result, while these judged with low confidence are put into the *pool* which is used in the next round of relevance feedback. Experiments show that semi-supervised learning and active learning mechanisms are both beneficial to CBIR.

1 Introduction

With the rapid increase of the volume of digital image collections, content-based image retrieval (CBIR) has attracted a lot of research interest in recent years [16]. The user could pose an example image, i.e. user query, and ask the CBIR system to bring out relevant images from the database. One of the difficulties 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 [14]. In relevance feedback, the user has the option of labeling a few of images according to whether they are relevant or not. The labeled images are then given to the CBIR system as complementary queries so that more images relevant to the user query could 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 into two classes, i.e. relevant or not. Since the classification is usually with different confidence, the learner produces a rank of the images according to how relevant they are 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 could 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, and pure supervised learning from such a small training set may be hard to obtain good generalization performance.

During the past years, using unlabeled data to help supervised learning has become a hot topic. Considering that in CBIR there are lots of unlabeled images in the database, this paper proposes to exploit them to improve the retrieval performance. In detail, the SSAIR (Semi-Supervised Active Image Retrieval) approach, which employs semi-supervised learning and active learning mechanisms, is proposed. Experiments show that utilizing unlabeled images could greatly enhance the performance of CBIR.

The rest of this paper is organized as follows. Section 2 briefly introduces semi-supervised learning and active learning. Section 3 presents SSAIR. Section 4 reports on the experiments. Section 5 discusses on some related works. Finally, Section 6 concludes and raises several issues for future work.

2 Semi-supervised/Active Learning

Semi-supervised learning deals with methods for exploiting unlabeled data in addition to labeled data to improve learning performance. Such methods include using transductive inference for support vector machines to optimize performance on a specific test set [7], using a generative model built from unlabeled data to perform discriminative classification [6], and using Expectation-Maximization to estimate maximum *a posteriori* parameters of a generative model [13]. A prominent achievement in this area is the *co-training* paradigm proposed by Blum and Mitchell [2], which trains two learners separately on two different views, i.e. two independent sets of attributes, and uses the prediction of each learner on unlabeled examples to augment the training set of the other.

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*. The former methods such as [9] train a single learner and then query the unlabeled instances on which the learner is the least confident. The latter methods such as [1][15] generate a committee of several learners and select the unlabeled instances on which the committee members disagree the most. A recent progress is the *co-testing* paradigm proposed by Muslea et al. [12], 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.

3 SSAIR

In CBIR, the user usually 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 data¹. Let \mathcal{U} denote the unlabeled data set while \mathcal{L} denote 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 whole database DB , \mathcal{P} is $\{query\}$, and \mathcal{N} is empty. Let $|\mathcal{X}|$ denote the size of a set \mathcal{X} . Then the size of \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 *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 be $\mathcal{U} - (\mathcal{P}^* \cup \mathcal{N}^*)$.

After obtaining the enlarged \mathcal{P} and \mathcal{N} , in each round of relevance feedback, a traditional CBIR system would re-train a learner which then would 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 labeled training examples. However, since it is not convenient to ask the user to label a lot of images in the relevance feedback process, in most cases the enlarged training set is still very small.

Inspired by the co-training paradigm [2], SSAIR attempts to exploit \mathcal{U} to improve the performance of retrieval. In detail, SSAIR employs two learners. After obtaining \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 assumed to be 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. The most relevant/irrelevant images, i.e. images with the biggest/smallest ranks, judged by each learner are passed on to the other learner as additional labeled positive/negative training examples. Then, both the learners are re-trained with enlarged labeled training sets and each of them produces a new rank for images in \mathcal{U} . Note that these additional images 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.

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

¹ For simplicity of discussion, here it is assumed that the database contains no annotated images.

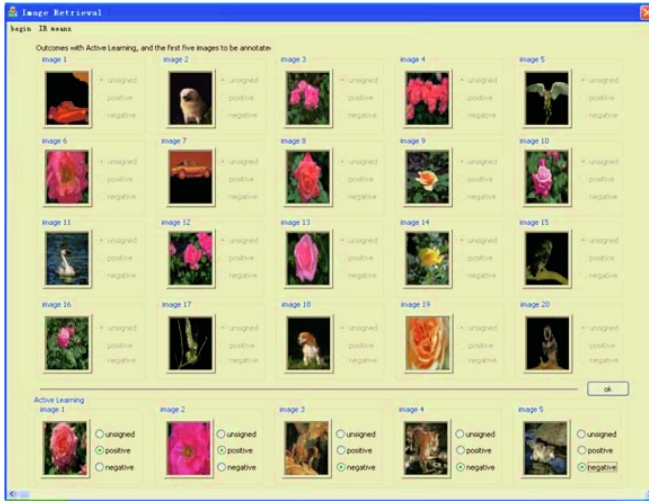


Fig. 1. User interface of a prototype system

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.

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 some 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 images that are most helpful to the improvement of the retrieval performance. For example, labeling an image that has already been well learned may be helpless.

Inspired by the co-testing paradigm [12], since SSAIR 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, SSAIR puts images with the bottom *poolsize* absolute ranks 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, SSAIR does not passively wait 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 our prototype system is shown in Fig. 1, where the region above the dark line is for the display of the retrieved images while the region below the dark line is for the display of the pooled images for relevance feedback.

In summary, the pseudo-code of SSAIR is presented in Table 1. Note that the learners used by SSAIR could be implemented in different ways. In this paper we use a very simple model as shown in Eq. 1, where \mathcal{Z}_{norm} is used to normalize the

Table 1. Pseudo-code describing the SSAIR approach

SSAIR(*query*, *DB*, L_1 , L_2 , *poolsize*, *resultsize*)

Input: *query*: User query
DB: Image database
 L_i ($i \in \{1..2\}$): Learners
poolsize: No. images in the pool
resultsize: No. images to be returned

$\mathcal{P} \leftarrow \{query\}; \mathcal{N} \leftarrow \emptyset; \mathcal{U} \leftarrow DB$

repeat until user is satisfied
 if user want to give feedback
 then
 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**
 $\mathcal{P}_i \leftarrow \mathcal{P} \cup \{\arg \max_{x \in \mathcal{U}} L_{(3-i)}(x, \mathcal{P}, \mathcal{N})\}$
 $\mathcal{N}_i \leftarrow \mathcal{N} \cup \{\arg \min_{x \in \mathcal{U}} L_{(3-i)}(x, \mathcal{P}, \mathcal{N})\}$
 else for $i \in \{1..2\}$ **do** $\mathcal{P}_i \leftarrow \mathcal{P}; \mathcal{N}_i \leftarrow \mathcal{N}$
 for $x \in \mathcal{U}$ **do** $Rank(x) \leftarrow \frac{1}{Z_{norm}} \sum_{i \in \{1..2\}} L_i(x, \mathcal{P}_i, \mathcal{N}_i)$
 $Pool \leftarrow \emptyset; Result \leftarrow \emptyset$
 for $i \in \{1..poolsize\}$ **do** $Pool \leftarrow Pool \cup \{\arg \min_{x \in \mathcal{U}} |Rank(x)|\}$
 for $i \in \{1..resultsize\}$ **do** $Result \leftarrow Result \cup \{\arg \max_{x \in \mathcal{U}} Rank(x)\}$
 end of repeat

Output: *Result*

result to $(-1, 1)$, ε is used to avoid zero denominator, and Sim_i is a similarity measure.

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

Since images can be represented as feature vectors after appropriate feature extraction, in this paper the reciprocal of the Minkowsky distance between two d -dimensional feature vectors \hat{x} and \hat{y} is used to measure the similarity between two images x and y , as shown in Eq. 2 where ε is used to avoid zero denominator. Note that for the first learner ($i = 1$) in SSAIR the order of the Minkowsky distance is 2, while for the second learner ($i = 2$) the order is 3. 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 these two learners could produce different ranks in SSAIR.

$$Sim_i(x, y) = 1 / \left(\left(\sum_{j=1}^d |\hat{x}_j - \hat{y}_j|^{(i+1)} \right)^{1/(i+1)} + \varepsilon \right) \quad (2)$$

Note that other kinds of semi-supervised and active learning paradigms can also be used here to exploit the unlabeled images. SSAIR employs co-training and co-testing like schemes just because this enables semi-supervised learning and active learning to be easily and gracefully integrated together. It is worth mentioning that the standard co-training [2] and co-testing [12] require *sufficient but redundant views*, but some recent work [5] shows that using two different supervised learners instead of two attribute sets can also work well, which is really the way SSAIR goes. These issues will be discussed further in Section 5.

4 Experiments

SSAIR is empirically compared with three image retrieval approaches, i.e. RFIR, SSIR, and AIR. The difference between SSAIR and RFIR is that RFIR uses neither semi-supervised learning nor active learning. The difference between SSAIR and SSIR is that SSIR does not use active learning. The difference between SSAIR and AIR is that AIR does not use semi-supervised learning.

Experiments are performed on a set of 2,000 images from the COREL database. These images belong to 20 classes, each of which has 100 images. The features used to represent the images are color histograms. Note that since the purpose of the experiments is to explore whether employing semi-supervised learning and active learning mechanisms are helpful to CBIR, the relative performance of SSAIR against other approaches is more important than its absolute performance. So, here we have not spent much time in trying to use stronger features, although doing so might greatly improve the absolute retrieval performance.

For each compared approach, after obtaining a query, five rounds of relevance feedback are performed. In each round the user could label 5, 7, or 9 images as his/her feedback. The whole process is repeated for five times with different queries, and the average results are recorded.

The evaluation measures used in CBIR have been affected much by these used in common information retrieval [11]. A straightforward while popularly used measure is the *PR-Graph*, which depicts the relationship between *precision* and *recall* of a specific retrieval system. However, since there is a PR-Graph for each class of images after every round of relevance feedback, it is hard to present these graphs in a paper of limited length as this one. Usually, a CBIR system exhibits a trade-off between precision and recall, to obtain a high precision usually means sacrificing recall and vice versa. Considering that in CBIR precision and recall are of the same importance, here BEP (*break-event-point*) is introduced as an evaluation measure. In 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, *BEP-Graph* is obtained, where the horizontal axis enumerates the round of relevance feedback while the vertical axis tells the BEP value.

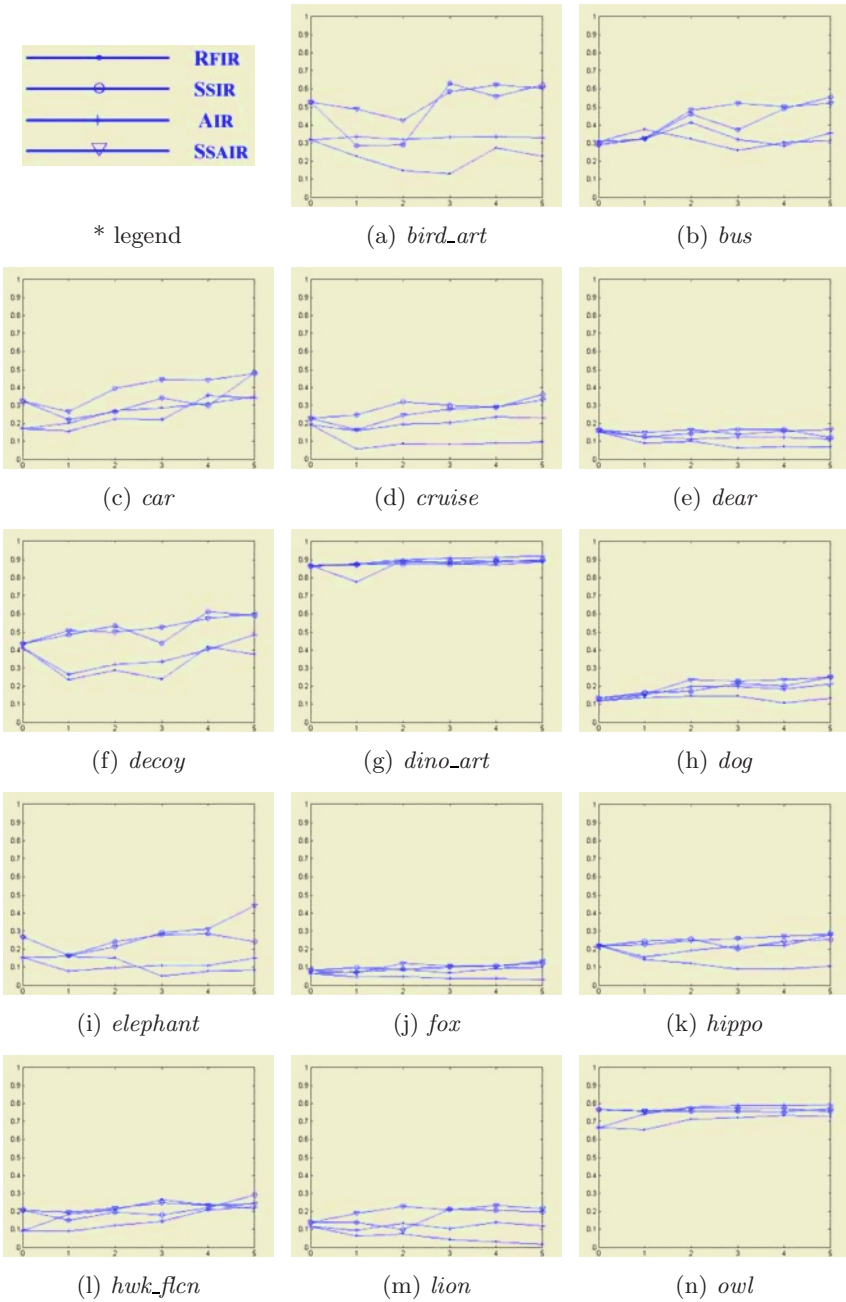


Fig. 2. BEP-Graphs for different classes of images, with 5 feedbacks in each round

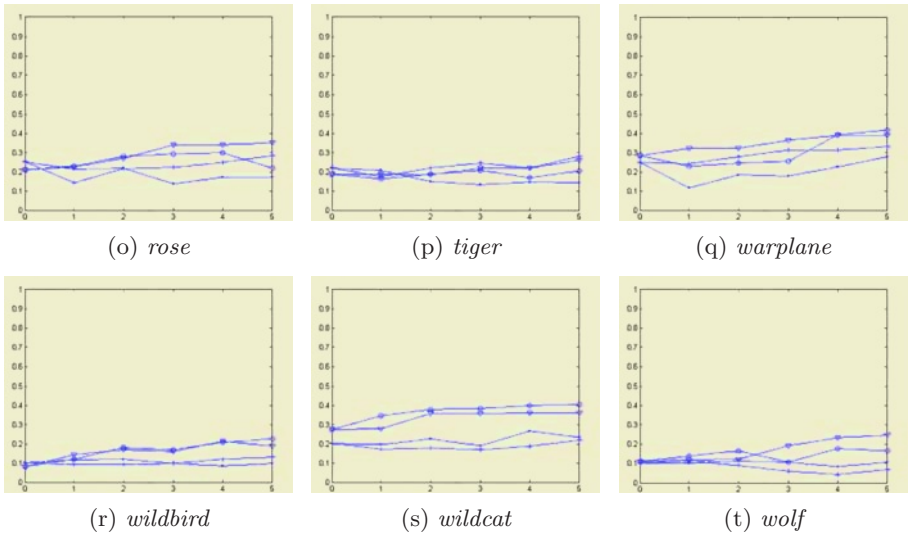


Fig. 2. (continued)

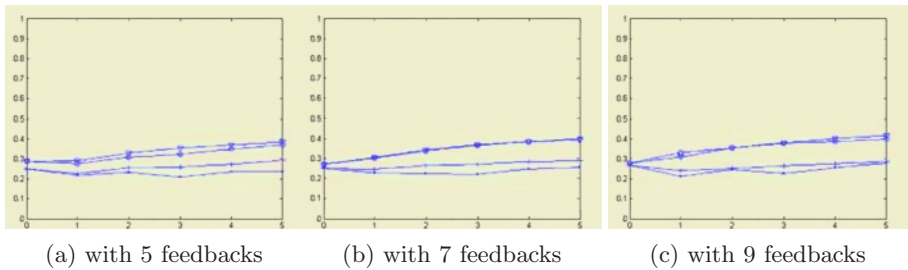


Fig. 3. Geometrical-BEP-Graphs with 5, 7, or 9 feedbacks in each round

The BEP-Graphs for the twenty classes of images are shown in Fig. 2. Here in each round of relevance feedback, the user provides labels for five images. After averaging across all the classes of images, the *Geometrical-BEP-Graphs* with 5, 7, or 9 feedbacks in each round of relevance feedback are presented in Fig. 3, where the legend used is the same as that used in Fig. 2.

Fig. 2 shows that the performance of the compared approaches are not identical on different classes of images. There are classes such as *car* on which SSAIR is distinctly superior to the other three approaches, but there are also classes such as *dino_art* on which the performance of the compared approaches are very similar. Nevertheless, Fig. 2 reveals that SSAIR is almost always the best or among the best approaches on all these image classes. This finding is supported by Fig. 3(a), which exhibits a clear rank of the performance of the compared approaches, that is, $SSAIR > SSIR > AIR > RFIR$. This observation tells that introducing semi-supervised learning and active learning into CBIR is beneficial,

and the benefit from semi-supervised learning is bigger than that from active learning. This is not difficult to understand because semi-supervised learning enables the utilization of images in addition to these labeled by the user in relevance feedback, although the user feedback process in AIR is more effective than that in SSIR. Moreover, Figs. 3(b) and 3(c) show that when the number of feedback in each round increases, the benefit from active learning decreases so that the performance of SSAIR is very comparable to that of SSIR. This is also not difficult to understand because as the number of feedback increases, the images that the user chosen from the retrieval result to label would have more chances to overlap with the images that the system actively asks the user to label.

Besides BEP-Graph, this paper uses another measure, i.e. *effectiveness*, to evaluate the compared approaches. This measure was proposed by Mehtre et al. [10], and adopted to quantify the utility of relevance feedback mechanisms recently [4]. The definition is shown in Eq. 3, where S denotes the number of relevant images the user wants to retrieve, R_q^I and R_q^E respectively denote the set of relevant images and all images retrieved in the shortlist. The bigger the η_S , the better the performance.

$$\eta_S = \begin{cases} |R_q^I \cap R_q^E|/|R_q^I| & \text{if } |R_q^I| \leq S \\ |R_q^I \cap R_q^E|/|R_q^E| & \text{if } |R_q^I| > S \end{cases} \quad (3)$$

Due to the page limit, here we only present the *geometrical-effectiveness*, i.e. the average effectiveness across all the image classes, as shown in Table 2 where $\bar{\eta}_{200}^i$ denotes the geometrical-effectiveness of the i -th round of relevance feedback with the size of shortlist being 200.

Table 2. Geometrical-effectiveness with 5, 7, or 9 feedbacks in each round

	$\bar{\eta}_{200}^1$		$\bar{\eta}_{200}^2$		$\bar{\eta}_{200}^3$		$\bar{\eta}_{200}^4$		$\bar{\eta}_{200}^5$	
	RFIR	AIR	RFIR	AIR	RFIR	AIR	RFIR	AIR	RFIR	AIR
	SSIR	SSAIR	SSIR	SSAIR	SSIR	SSAIR	SSIR	SSAIR	SSIR	SSAIR
5 feedbacks	30.2	33.6	30.8	35.9	28.8	36.7	31.1	39.9	32.4	41.8
	36.5	37.0	41.4	43.2	43.1	46.2	45.6	48.0	47.0	49.2
7 feedbacks	29.0	34.2	29.5	37.0	30.5	38.1	33.1	40.0	34.9	41.6
	40.7	41.4	43.5	45.3	46.3	46.9	47.7	48.5	49.7	49.9
9 feedbacks	29.0	34.2	29.5	37.0	30.5	38.1	33.1	39.8	34.9	41.6
	43.4	41.5	45.7	47.3	48.7	49.6	49.0	51.3	51.4	52.4

Table 2 shows that except on the 1st round of 9 feedbacks where SSIR is better than SSAIR, SSAIR is always better than SSIR, SSIR is always better than AIR, and AIR is always better than RFIR. The table also exhibits that the gap between the performance of SSIR and AIR is relatively big. These observations verify again that introducing semi-supervised learning and active learning into

CBIR is beneficial, and the benefit from semi-supervised learning is bigger than that from active learning.

5 Related Works

As early as in 1983, Bookstein [3] 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 relevance of 0.5 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 [17], 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 most close to the support vector boundary. Since the learner used in SSAIR is far simpler than the support vector machine used in SVM_{Active} , SSAIR is more efficient than SVM_{Active} . Moreover, since SSAIR employs semi-supervised learning mechanism while SVM_{Active} does not, SSAIR is stronger than SVM_{Active} in exploiting unlabeled data.

Another work on introducing active learning to CBIR was done by Zhang and Chen [19]. 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. This work is quite different from SSAIR not only because it does not use any semi-supervised learning mechanism, but also because it mainly works on hidden annotation while SSAIR works on relevance feedback. As Zhang and Chen [19] said, relevance feedback does not accumulate semantic knowledge while hidden annotation, on the other hand, tries to accumulate all the knowledge given by the user.

Wu et al. [18] has tried to apply semi-supervised learning to CBIR. They cast the problem of CBIR as a transductive learning problem, and proposed the D-EM algorithm to solve this problem. On a small subset of COREL which contains only 134 images, they reported that their approach had achieved good results nevertheless physical or mathematical features were used. It is worth noting that their approach works in the Expectation-Maximization framework, while SSAIR was inspired by the co-training paradigm. Moreover, their approach does not use any active learning mechanism while SSAIR does.

The standard co-training algorithm [2] requires *sufficient but redundant views*, that is, the attributes be naturally partitioned into two sets, each of which is sufficient for learning and conditionally independent to the other given the class label. It is obvious that SSAIR does not utilize two sufficient but redundant views. In fact, the semi-supervised learning mechanism embedded in SSAIR is somewhat like the algorithm proposed by Goldman and Zhou [5], which uses two different supervised learners instead of two attribute sets. In contrast to Goldman and Zhou's algorithm, SSAIR does not employ time-consuming cross validation technique to determine how to label the unlabeled examples and how to produce

the final hypothesis. Therefore, the semi-supervised learning mechanism of SSAIR is more efficient.

The active learning mechanism embedded in SSAIR was inspired by co-testing [12]. In contrast to the algorithm proposed by Muslea et al. [12], SSAIR does not utilize two sufficient but redundant views. Instead, it employs two learners obtained through setting different parameter values of the same model. Since sufficient but redundant views are not often available, the active learning mechanism of SSAIR has better applicability.

6 Conclusion and Future Work

This paper advocates applying semi-supervised learning and active learning together to CBIR. As an example, the proposed SSAIR approach gracefully integrates the merits of these learning mechanisms in exploiting unlabeled data, and experiments show that it could effectively improve the retrieval performance.

Although the utility of SSAIR has been verified by experiments, there is a lack of theoretical analysis. This might have encumbered the exertion of the full power of SSAIR. For example, in current form of SSAIR, in each round of relevance feedback each learner only labels for the other learner two images, i.e. the most relevant/irrelevant images it judged to be. If theoretical analysis on the relationship between the performance of the learners and the possible noises in the labeling process is available, it might be found that letting each learner label more images is still safe, which may help improve the performance of SSAIR. This is an important issue for future work.

Moreover, in this paper images are described by simple color histogram features. Although it is anticipated the retrieval performance could be further improved through utilizing stronger image features, there is a bare possibility that utilizing unlabeled images is not so beneficial when the features are strong. Therefore, investigating the performance of SSAIR facilitated with strong image features is also an interesting issue for future work.

Acknowledgement

This work was supported by the National Outstanding Youth Foundation of China under the Grant No. 60325207, and the Foundation for the Author of National Excellent Doctoral Dissertation of China under the Grant No. 200343.

References

1. Abe, N., Mamitsuka, H.: Query learning strategies using boosting and bagging. In: Proceedings of the 15th International Conference on Machine Learning, Madison, WI (1998) 1–9
2. Blum, A., Mitchell, T.: Combining labeled and unlabeled data with co-training. In: Proceedings of the 11th Annual Conference on Computational Learning Theory, Madison, WI (1998) 92–100
3. Bookstein, A.: Information retrieval: a sequential learning process. *Journal of the American Society for Information Science* **34** (1983) 331–342

4. Ciocca, G., Schettini, R.: A relevance feedback mechanism for content-based image retrieval. *Information Processing and Management* **35** (1999) 605–632
5. Goldman, S., Zhou, Y.: Enhancing supervised learning with unlabeled data. In: *Proceedings of the 17th International Conference on Machine Learning*, San Francisco, CA (2000) 327–334
6. Jaakkola, T., Haussler, D.: Exploiting generative models in discriminative classifiers. In: Kearns, M.S., Solla, S.A., Cohn, D.A. (eds.): *Advances in Neural Information Processing Systems*, Vol. 11. MIT Press, Cambridge, MA (1999) 487–493
7. Joachims, T.: Transductive inference for text classification using support vector machines. In: *Proceedings of the 16th International Conference on Machine Learning*, Bled, Slovenia (1999) 200–209
8. Lewis, D.: Representation and learning in information retrieval. PhD thesis, Department of Computer Science, University of Massachusetts, Amherst, MA (1992)
9. Lewis, D., Gale, W.: 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 (1994) 3–12
10. Mehtre, B.M., Kankanhalli, M.S., Narasimhalu, A.D., Man, G.C.: Color matching for image retrieval. *Pattern Recognition Letters* **16** (1995) 325–331
11. Müller, H., Müller, W., Squire, D.M., Marchand-Maillet, S., Pun, T.: Performance evaluation in content-based image retrieval: overview and proposals. *Pattern Recognition Letters* **22** (2001) 593–601
12. Muslea, I., Minton, S., Knoblock, C.A.: Selective sampling with redundant views. In: *Proceedings of the 17th National Conference on Artificial Intelligence*, Austin, TX (2000) 621–626
13. Nigam, K., McCallum, A., Thrun, S., Mitchell, T.: Text classification from labeled and unlabeled documents using EM. *Machine Learning* **39** (2000) 103–134
14. Rui, Y., Huang, T.S., Ortega, M., Mehrotra, S.: Relevance feedback: a power tool for interactive content-based image retrieval. *IEEE Transactions on Circuits and Systems for Video Technology* **8** (1998) 644–655
15. Seung, H., Opper, M., Sompolinsky, H.: Query by committee. In: *Proceedings of the 5th ACM Workshop on Computational Learning Theory*, Pittsburgh, PA (1992) 287–294
16. 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** (2000) 1349–1380
17. Tong, S., Chang, E.: Support vector machine active learning for image retrieval. In: *Proceedings of the 9th ACM International Conference on Multimedia*, Ottawa, Canada (2001) 107–118
18. Wu, Y., Tian, Q., Huang, T.S.: Discriminant-EM algorithm with application to image retrieval. In: *Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition*, Hilton Head, SC (2000) 222–227
19. Zhang, C., Chen, T.: An active learning framework for content-based information retrieval. *IEEE Transactions on Multimedia* **4** (2002) 260–268