Available online at www.sciencedirect.com
SCIENCE DIRECT®
Image and Vision Computing xxx (2006) xxx–xxx
ELSEVIER
image and vision COMPUTING
www.elsevier.com/locate/imavis

Kernel-based distance metric learning for content-based image retrieval

Hong Chang, Dit-Yan Yeung *

Department of Computer Science, Hong Kong University of Science and Technology, Clear Water Bay, Kowloon, Hong Kong

Received 24 August 2005; received in revised form 16 February 2006; accepted 16 May 2006

Abstract

For a specific set of features chosen for representing images, the performance of a content-based image retrieval (CBIR) system depends critically on the similarity or dissimilarity measure used. Instead of manually choosing a distance function in advance, a more promising approach is to learn a good distance function from data automatically. In this paper, we propose a kernel approach to improve the retrieval performance of CBIR systems by learning a distance metric based on pairwise constraints between images as supervisory information. Unlike most existing metric learning methods which learn a Mahalanobis metric corresponding to performing linear transformation in the original image space, we define the transformation in the kernel-induced feature space which is nonlinearly related to the image space. Experiments performed on two real-world image databases show that our method not only improves the retrieval performance of Euclidean distance without distance learning, but it also outperforms other distance learning methods significantly due to its higher flexibility in metric learning.

Keywords: Metric learning; Kernel method; Content-based image retrieval; Relevance feedback

1. Introduction

1.1. Content-based image retrieval

With the emergence and increased popularity of the World Wide Web (WWW) over the past decade, retrieval of images based on content, often referred to as *content-based image retrieval* (CBIR), has gained a lot of research interests [1]. On the WWW where many images can be found, it is convenient to search for the target images in possibly very large image databases by presenting query images as examples. Thus, more and more Web search engines (e.g., Yahoo) are now equipped with CBIR facilities for retrieving images on a query-by-image-example basis.

The two determining factors for image retrieval performance are the features used to represent the images and the distance function used to measure the similarity between a query image and the images in the database. For a specific feature representation chosen, the retrieval performance depends critically on the similarity measure used. Let $\mathbf{f}^i = (f_1^i, f_2^i, \ldots, f_n^i)$ denote a feature vector representing image i, where n is the number of features. For example, $\mathbf{f}^i$ represents a color histogram with n being the number of histogram bins. There exist many methods for measuring the distance between feature vectors. Swain and Ballard [2] proposed the intersection distance measure $d_\cap = \sum_{k=1}^n \min(f_k^i, f_k^j)$, which has the same ordinal properties as the L_1 norm (distance). In [3], the distance between two histograms is defined as the weighted form $d_{\mathbf{W}}(\mathbf{f}^i, \mathbf{f}^j) = \sqrt{(\mathbf{f}^i - \mathbf{f}^j)^{\mathrm{T}} \mathbf{W} (\mathbf{f}^i - \mathbf{f}^j)}$, where each weight w_{ij} in $\mathbf{W}$ denotes the similarity between features i and j. Note that this distance measure includes the Mahalanobis distance as a special case. Other commonly used distance functions for color histograms include the Minkowski distance $d_r(\mathbf{f}^i, \mathbf{f}^j) = (\sum_{k=1}^n |f_k^i - f_k^j|^r)^{1/r}$. However, this distance metric may lead to high false negative rate [4].

* Corresponding author. Tel.: +852 2358 6977; fax: +852 2358 1477.
E-mail addresses: hongch@cs.ust.hk (H. Chang), dyyeung@cs.ust.hk (D.-Y. Yeung).

0262-8856/$ - see front matter © 2006 Elsevier B.V. All rights reserved.
doi:10.1016/j.imavis.2006.05.013

Unfortunately, the effectiveness of these distance functions is rather limited. Instead of choosing a distance function in advance, a more promising approach is to learn a good distance function from data automatically. Recently, this challenging new direction has aroused great interest in the research community.

1.2. Related work

Relevance feedback has been used in the traditional information retrieval community to improve the performance of information retrieval systems based on user feedback. This interactive approach has also emerged as a popular approach in CBIR [5]. The user is provided with the option of labeling (some of the) previously retrieved images as either relevant or irrelevant. Based on this feedback information, the CBIR system can iteratively refine the retrieval results by learning a more appropriate (dis)similarity measure. For example, relevance feedback can be used to modify the weights in the weighted Euclidean distance [5] or the generalized Euclidean distance [6]. The same approach has also been applied to a correlation-based metric [7,8], which usually outperforms Euclidean-based measures. In [9], the authors presented an approach to generate an adaptive quasiconformal kernel distance metric based on relevance feedback. Dong and Bhanu [10] proposed a new semi-supervised expectation-maximization (EM) algorithm for image retrieval tasks, with the image distribution in the feature space modeled as Gaussian mixtures. Pseudo-feedback strategy based on peer indexing was proposed recently to optimize the similarity metric and the initial query vectors [11], where the global and personal image peer indexes are learned interactively and incrementally from user feedback information. Some recent work makes use of the manifold structure of image data in the feature space for image retrieval [12,13]. Other methods include biased discriminant analysis [14], support vector machine (SVM) active learning [15–17], boosting methods [18], and so on.

In the machine learning literature, supervisory information for semi-supervised distance learning usually takes the form of limited labeled data or *pairwise similarity or dissimilarity constraints*. The latter type of information is weaker in the sense that pairwise constraints can be derived from labeled data but not vice versa. Relevance feedback, which has been commonly used in CBIR, may be used to obtain the pairwise constraints. Recently, some machine learning researchers have proposed different metric learning methods for semi-supervised clustering with pairwise similarity or dissimilarity side information [19–22]. Most of these methods try to learn a global Mahalanobis metric corresponding to linear transformation in the original image space [19,20,22]. In particular, an efficient, noniterative algorithm called relevance component analysis (RCA) [19,20] has been used to improve image retrieval performance in CBIR tasks. This work was later extended in [19] by incorporating both similarity and dissimilarity constraints into the EM algorithm for model-based clustering based on Gaussian mixture models. More recently, Hertz et al. [23,24] proposed a nonmetric distance function learning algorithm called DistBoost by boosting the hypothesis over the product space with Gaussian mixture models as weak learners. Using DistBoost, they demonstrated very good image retrieval results in CBIR tasks.

Most existing systems only make use of relevance feedback within a single query session. More recently, some methods have been proposed for the so-called *long-term learning* by accumulating relevance feedback from multiple query sessions which possibly involve different users [25,12,13,26]. However, [12,13] are based on the assumption that the feature vectors representing the images form a Riemannian manifold in the feature space. Unfortunately, this assumption may not hold in real-world image databases. Moreover, the log-based relevance feedback method [26] is expected to encounter the scale-up problem as the number of relevance feedback log sessions increases.

1.3. This paper

Metric learning based on pairwise constraints can be categorized into linear and nonlinear methods. Most existing metric learning methods learn a Mahalanobis metric corresponding to performing linear transformation in the original image space. However, for CBIR tasks, the original image space is highly nonlinear due to high variability of the image content and style. In this paper, we define the transformation in the kernel-induced feature space which is nonlinearly related to the image space. The transformation is then learned based on side information in the form of pairwise (dis)similarity constraints. Moreover, to address the efficiency problem for long-term learning, we boost the image retrieval performance by adapting the distance metric in a stepwise manner based on relevance feedback.

Our kernel-based distance metric learning method performs kernel PCA on the whole data set, followed by metric learning in the feature space. It does not suffer from the small sample size problem encountered by traditional Fisher discriminant analysis methods. Therefore, our method is significantly different from many existing methods which aim to address the small sample size problem in multimedia information retrieval, e.g., the kernel-based biased discriminant analysis method proposed in [14].

In Section 2, we will propose a kernel-based method for nonlinear metric learning. In Section 3, we will describe how this method can be used to improve the performance of CBIR tasks. Our method will then be compared with other distance learning methods based on two real-world image databases. The stepwise kernel-based metric learning algorithm that pays attention to both effectiveness and efficiency will be presented in Section 4. Finally, some concluding remarks will be given in the last section.

2. Kernel-based metric learning

Kernel methods typically comprise two parts. The first part maps (usually nonlinearly) the input points to a feature space often of much higher or even infinite dimensionality, and then the second part applies a relatively simple (usually linear) method in the feature space. In this section, we propose a two-step method which first uses kernel principal component analysis (PCA) [27] to embed the input points in terms of their nonlinear principal components and then applies metric learning there.

2.1. Centering in the feature space

Let $\mathbf{x}_i$ $(i = 1, \ldots, n)$ be n points in the input space $\mathcal{X}$. Suppose we use a kernel function $\hat{k}$ which induces a nonlinear mapping $\hat{\phi}$ from $\mathcal{X}$ to some feature space $\mathcal{F}$.[1] The "images" of the n points in $\mathcal{F}$ are $\hat{\phi}(\mathbf{x}_i)$ $(i = 1, \ldots, n)$, which in general are not centered (i.e., their sample mean is not zero). The corresponding kernel matrix $\hat{\mathbf{K}} = [\hat{k}(\mathbf{x}_i, \mathbf{x}_j)]_{n\times n} = [\langle \hat{\phi}(\mathbf{x}_i), \hat{\phi}(\mathbf{x}_j)\rangle]_{n\times n}$.

We want to transform (simply by translating) the coordinate system of $\mathcal{F}$ such that the new origin is at the sample mean of the n points. As a result, we also convert the kernel matrix $\hat{\mathbf{K}}$ to $\mathbf{K} = [k(\mathbf{x}_i, \mathbf{x}_j)]_{n\times n} = [\langle \phi(\mathbf{x}_i), \phi(\mathbf{x}_j)\rangle]_{n\times n}$.

Let $\mathbf{Y} = [\phi(\mathbf{x}_1), \ldots, \phi(\mathbf{x}_n)]^{\mathrm{T}}$, $\hat{\mathbf{Y}} = [\hat{\phi}(\mathbf{x}_1), \ldots, \hat{\phi}(\mathbf{x}_n)]^{\mathrm{T}}$ and $\mathbf{H} = \mathbf{I} - \frac{1}{n}\mathbf{1}\mathbf{1}^{\mathrm{T}}$, where $\mathbf{1}$ is a column vector of ones. We can express $\mathbf{Y} = \mathbf{H}\hat{\mathbf{Y}}$. Hence,

$$\mathbf{K} = \mathbf{Y}\mathbf{Y}^{\mathrm{T}} = \mathbf{H}\hat{\mathbf{Y}}\hat{\mathbf{Y}}^{\mathrm{T}}\mathbf{H} = \mathbf{H}\hat{\mathbf{K}}\mathbf{H}. \tag{1}$$

2.2. Step 1: Kernel PCA

We briefly review the kernel PCA algorithm here. More details can be found in [27].

We first apply the centering transform as in Eq. (1) to get the kernel matrix $\mathbf{K}$. We then solve the eigenvalue equation for $\mathbf{K}$: $\mathbf{K}\alpha = \xi\alpha$. Let $\xi_1 \geqslant \cdots \geqslant \xi_p > 0$ denote the $p \leqslant n$ positive eigenvalues of $\mathbf{K}$ and $\alpha_1, \ldots, \alpha_p$ be the corresponding eigenvectors. The embedding dimensionality p may be set to the rank of $\mathbf{K}$, or, more commonly, a smaller value to ignore the insignificant dimensions with very small eigenvalues, as in ordinary PCA.

For any input $\mathbf{x}$, the kth principal component $\tilde{y}_k$ of $\phi(\mathbf{x})$ is given by

$$\tilde{y}_k = \frac{1}{\sqrt{\xi_k}} \sum_{i=1}^{n} \alpha_{ik} \langle \phi(\mathbf{x}_i), \phi(\mathbf{x})\rangle. \tag{2}$$

If $\mathbf{x} = \mathbf{x}_j$ for some $1 \leqslant j \leqslant n$, i.e., $\mathbf{x}$ is one of the n original points, then the kth principal component $\tilde{y}_{jk}$ of $\phi(\mathbf{x}_j)$ becomes

$$\tilde{y}_{jk} = \frac{1}{\sqrt{\xi_k}} (\mathbf{K}\alpha_k)_j = \frac{1}{\sqrt{\xi_k}} (\xi_k \alpha_k)_j = \sqrt{\xi_k}\,\alpha_{jk}, \tag{3}$$

which is proportional to the expansion coefficient α_{jk}. Thus, the input points $\mathbf{x}_i$ $(i = 1, \ldots, n)$ are now represented as $\tilde{\mathbf{y}}_i$ $(i = 1, \ldots, n)$.

2.3. Step 2: Linear metric learning

To perform metric learning, we further transform $\tilde{\mathbf{y}}_i$ $(i = 1, \ldots, n)$ by applying a linear transform $\mathbf{A}$ to each point based on the pairwise similarity and dissimilarity information in $\mathcal{S}$ and $\mathcal{D}$, respectively.

We define a matrix $\mathbf{C}_{\mathcal{S}}$ based on $\mathcal{S}$ as follows:

$$\begin{aligned}\mathbf{C}_{\mathcal{S}} &= \frac{1}{|\mathcal{S}|} \sum_{(\mathbf{x}_i,\mathbf{x}_j)\in\mathcal{S}} \left[\left(\tilde{\mathbf{y}}_i - \frac{\tilde{\mathbf{y}}_i + \tilde{\mathbf{y}}_j}{2}\right)\left(\tilde{\mathbf{y}}_i - \frac{\tilde{\mathbf{y}}_i + \tilde{\mathbf{y}}_j}{2}\right)^{\mathrm{T}} \right. \\ &\quad \left. + \left(\tilde{\mathbf{y}}_j - \frac{\tilde{\mathbf{y}}_i + \tilde{\mathbf{y}}_j}{2}\right)\left(\tilde{\mathbf{y}}_j - \frac{\tilde{\mathbf{y}}_i + \tilde{\mathbf{y}}_j}{2}\right)^{\mathrm{T}} \right] \\ &= \frac{1}{2|\mathcal{S}|} \sum_{(\mathbf{x}_i,\mathbf{x}_j)\in\mathcal{S}} (\tilde{\mathbf{y}}_i - \tilde{\mathbf{y}}_j)(\tilde{\mathbf{y}}_i - \tilde{\mathbf{y}}_j)^{\mathrm{T}},\end{aligned} \tag{4}$$

where $|\mathcal{S}|$ denotes the number of similar pairs in $\mathcal{S}$. Note that this form is similar to that used in RCA [19] by treating each pair in $\mathcal{S}$ as a chunklet. This slight variation makes it easier to extend the method to incorporate pairwise dissimilarity constraints into metric learning, as illustrated here. Similarly, we define a matrix $\mathbf{C}_{\mathcal{D}}$ based on $\mathcal{D}$:

$$\mathbf{C}_{\mathcal{D}} = \frac{1}{2|\mathcal{D}|} \sum_{(\mathbf{x}_k,\mathbf{x}_l)\in\mathcal{D}} (\tilde{\mathbf{y}}_k - \tilde{\mathbf{y}}_l)(\tilde{\mathbf{y}}_k - \tilde{\mathbf{y}}_l)^{\mathrm{T}}, \tag{5}$$

where $|\mathcal{D}|$ denotes the number of similar pairs in $\mathcal{D}$.

The linear transform $\mathbf{A}$ is defined as

$$\mathbf{A} = \mathbf{C}_{\mathcal{D}}^{\frac{1}{2}} \mathbf{C}_{\mathcal{S}}^{-\frac{1}{2}}. \tag{6}$$

Each point $\tilde{\mathbf{y}}$, whether or not corresponding to one of the n original points, is then transformed to $\mathbf{z} = \mathbf{A}\tilde{\mathbf{y}} = \mathbf{C}_{\mathcal{D}}^{\frac{1}{2}} \mathbf{C}_{\mathcal{S}}^{-\frac{1}{2}} \tilde{\mathbf{y}}$. The Euclidean metric in the transformed feature space thus corresponds to a modified metric in the original space to better characterize the implicit similarity relationships between data points.

3. Image retrieval experiments

In this section, we apply the kernel-based metric learning method to improve the retrieval performance of CBIR tasks. We also compare the retrieval performance of this method with other distance learning methods.

3.1. Image databases and feature representation

Our image retrieval experiments are based on two image databases. One database is a subset of the Corel Photo Gallery, which contains 1010 images belonging to 10 different classes. The 10 classes include bear (122), butterfly (109), cactus (58), dog (101), eagle (116), elephant (105), horse (110), penguin (76), rose (98), and tiger (115). Another database contains 547 images belonging to six classes

[1] We use RBF kernel in this paper.

that we downloaded from the Internet. The image classes are manually defined based on high-level semantics.

We first represent the images in the HSV color space, and then compute the *color coherence vector* (CCV) [28] as the feature vector for each image, as was done in [23,24]. Specifically, we quantize each image to $8 \times 8 \times 8$ color bins, and then represent the image as a 1024-dimensional CCV $(\alpha_1, \beta_1, \ldots, \alpha_{512}, \beta_{512})^{\mathrm{T}}$, with α_i and β_i representing the numbers of coherent and noncoherent pixels, respectively, in the ith color bin. The CCV representation stores the number of coherent versus noncoherent pixels with each color and gives finer distinctions than the use of color histograms. Thus, it usually gives better image retrieval results. For computational efficiency, we first apply ordinary PCA to retain the 60 dominating principal components before applying metric learning as described in the previous section.

3.2. Comparative study

We want to compare the image retrieval performance of the two-step kernel method with the baseline method of using Euclidean distance without distance learning as well as some other distance learning methods. In particular, we consider two distance learning methods: Mahalanobis distance learning with RCA and distance learning with DistBoost.[2] RCA makes use of the pairwise similarity constraints to learn a Mahalanobis distance, which essentially assigns large weights to relevant components and low weights to irrelevant components with relevance estimated based on the connected components composed of similar patterns. DistBoost, as discussed in Section 1.2, is a nonmetric distance learning method that makes use of the pairwise constraints and performs boosting. Since both DistBoost and our kernel method can make use of dissimilarity constraints in addition to similarity constraints, we conduct experiments with and without such supervisory information for the two methods. In summary, the following four methods are included in our comparative study:

1. Euclidean distance without distance learning.
2. Mahalanobis distance learning with RCA.
3. Nonmetric distance learning with DistBoost (with and without dissimilarity constraints).
4. Metric distance learning with our kernel method (with and without dissimilarity constraints).

3.3. Performance measures

We use two performance measures in our comparative study. The first one, based on *precision* and *recall*, is commonly used in information retrieval. The second one, used in [23,24], is based on *cumulative neighbor purity* curves. Cumulative neighbor purity measures the percentage of correctly retrieved images in the k nearest neighbors of the query image, averaged over all queries, with k up to some value K ($K = 30$ in our experiments).

For each retrieval task, we compute the average performance statistics over all queries of five randomly generated sets of similar and dissimilar image pairs. For both databases, the number of similar image pairs is set to 150, which is about 0.3% and 0.6%, respectively, of the total number of possible image pairs in the databases. The pairs of similar images are randomly selected based on the true class labels. The number of dissimilar image pairs used in DistBoost and our kernel method is also set to 150. For each set of similar and dissimilar image pairs, we set the number of boosting iterations in DistBoost to 50.

3.4. Experimental results

Fig. 1 shows the retrieval results on the first image database based on both cumulative neighbor purity and precision/recall. We can see that metric learning with the two-step kernel method significantly improves the retrieval performance and outperforms other distance learning methods especially with respect to the cumulative neighbor purity measure. The retrieval results on the second image database are shown in Fig. 2. Again, our kernel method significantly outperforms the other methods. For both databases, using dissimilarity constraints in DistBoost and the kernel method can improve the retrieval performance slightly.

Some typical retrieval results on the first and second databases are shown in Fig. 3(a) and (b), respectively. For each query image, we show the retrieved images in three rows, corresponding, from top to bottom, to the use of Euclidean distance without distance learning and distance learning with DistBoost and our kernel method based on similarity and dissimilarity information. Each row shows the seven nearest neighbors of the query image with respect to the distance used, with dissimilarity based on the distance increasing from left to right. The query image is shown with a frame around it. Note that the query image may not be the nearest neighbor using the DistBoost method since it learns nonmetric distance functions which, among other things, may not satisfy $d(\mathbf{x}, \mathbf{x}) = 0$ and the triangle inequality condition. We can see that both DistBoost and our kernel method improve the retrieval performance, with our method outperforming DistBoost slightly.

While the experiments above use the images in the databases as query images, another scenario that exists in some CBIR systems is to use query images that are not in the image databases. We have also performed some experiments on the first database under this setting, with a separate set of query images that are not used for distance learning. We split the database into the training (70%) and

[2] The program code for RCA and DistBoost was obtained from the authors of [19,24,20].

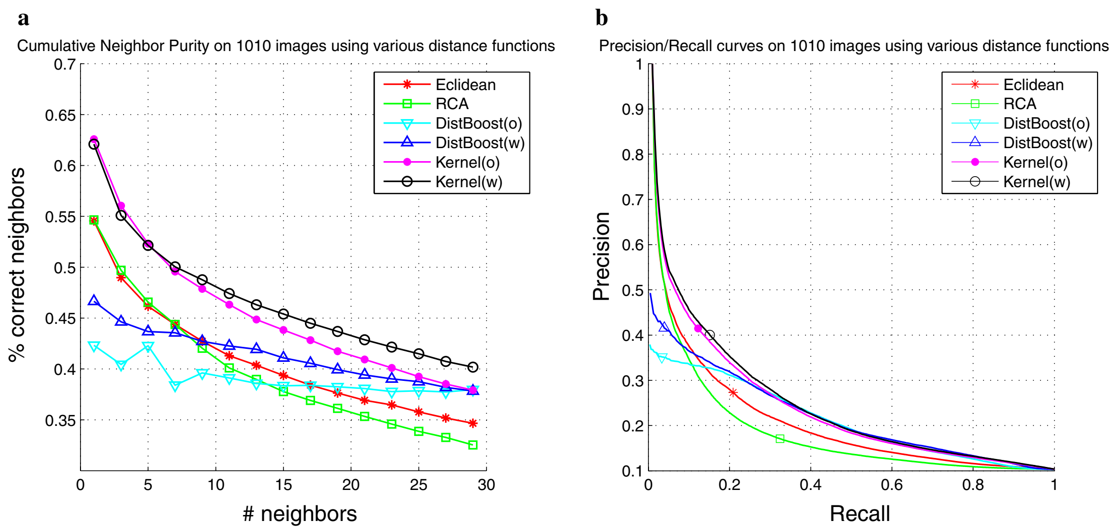

Fig. 1. Retrieval results on the first image database (1010 images, 10 classes). (a) Cumulative neighbor purity curves; (b) precision/recall curves.

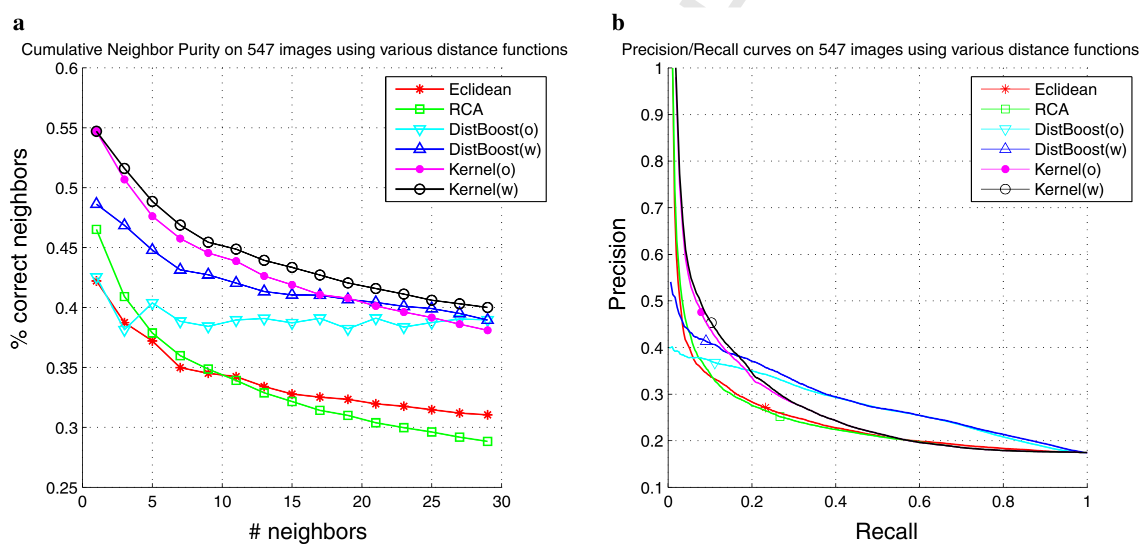

Fig. 2. Retrieval results on the second image database (547 images, 6 classes). (a) Cumulative neighbor purity curves; (b) precision/recall curves.

test (30%) sets, with the former used for distance learning and the latter serving as query images. Fig. 4 presents the retrieval results, which show that the kernel-based metric learning method still outperforms other methods.

3.5. Discussions

We have demonstrated the promising performance of our kernel-based metric learning method for CBIR tasks. Unlike other metric learning methods which learn a Mahalanobis metric corresponding to performing linear transformation in the original image space, we define the transformation in the kernel-induced feature space which is nonlinearly related to the image space. Metric learning estimates a linear transformation in the higher-dimensional feature space induced by the kernel used in kernel PCA. Any query image, either inside or outside the image database, is then mapped to the transformed feature space where the Euclidean metric can capture better the similarity relationships between patterns. Moreover, it is worthy to note that our kernel-based metric learning method is very efficient. In our experiments, it is more than 10 times faster than DistBoost for the same retrieval tasks.

We want to investigate further on how practical it is to incorporate distance learning into real-world CBIR tasks. As discussed above, relevance feedback is commonly used in CBIR systems for improving the retrieval performance [10,7,15,9,6,5,16,17,14]. The pairwise (dis)similarity constraints used by the kernel method can make better use of the relevance feedback from users, not only from one specific query but also from all previous ones. Specifically, similarity (dissimilarity) constraints can be obtained from the relevance feedback, with each relevant (irrelevant) image and the query

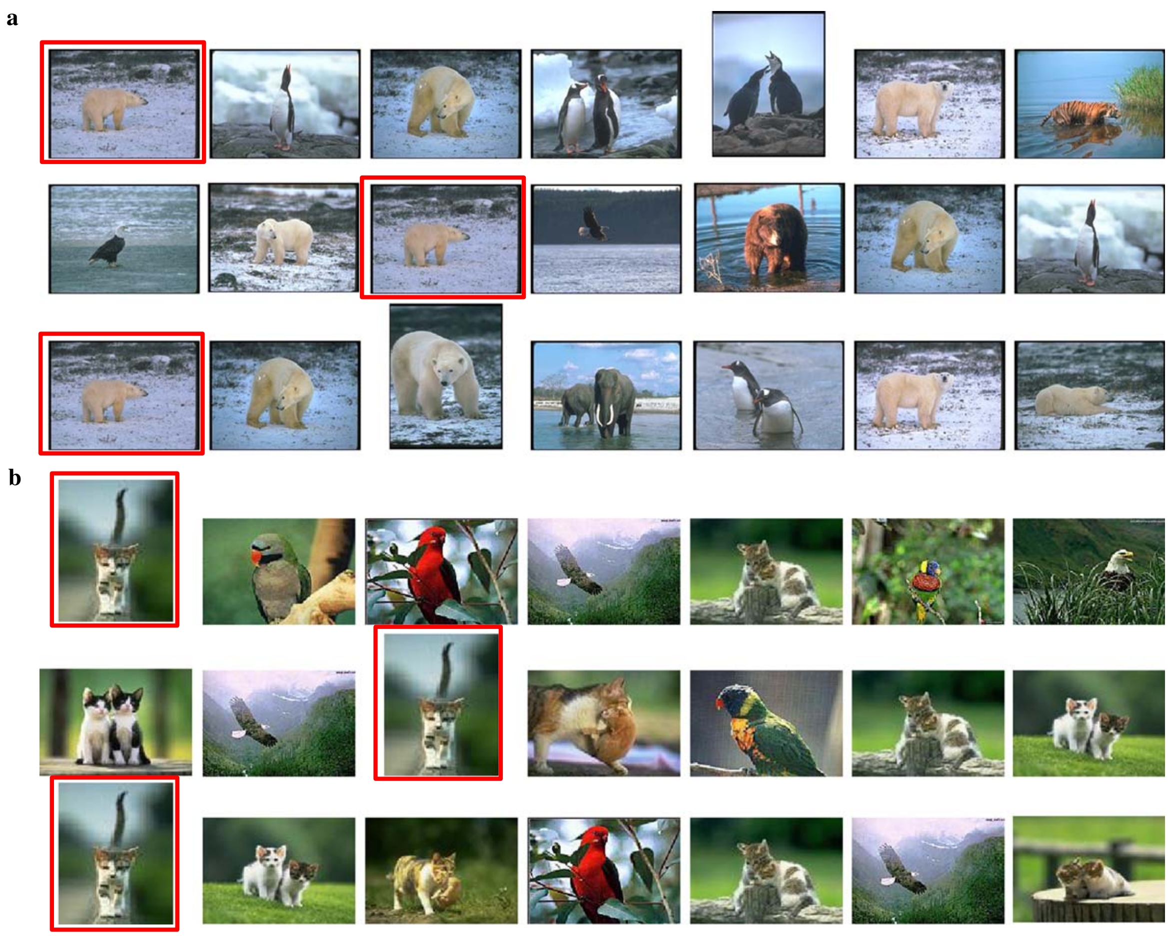

Fig. 3. Typical retrieval results on the two databases (a and b) based on Euclidean distance (top row), DistBoost (middle row) and our kernel method (bottom row). Each row shows the seven nearest neighbors including the query image (framed).

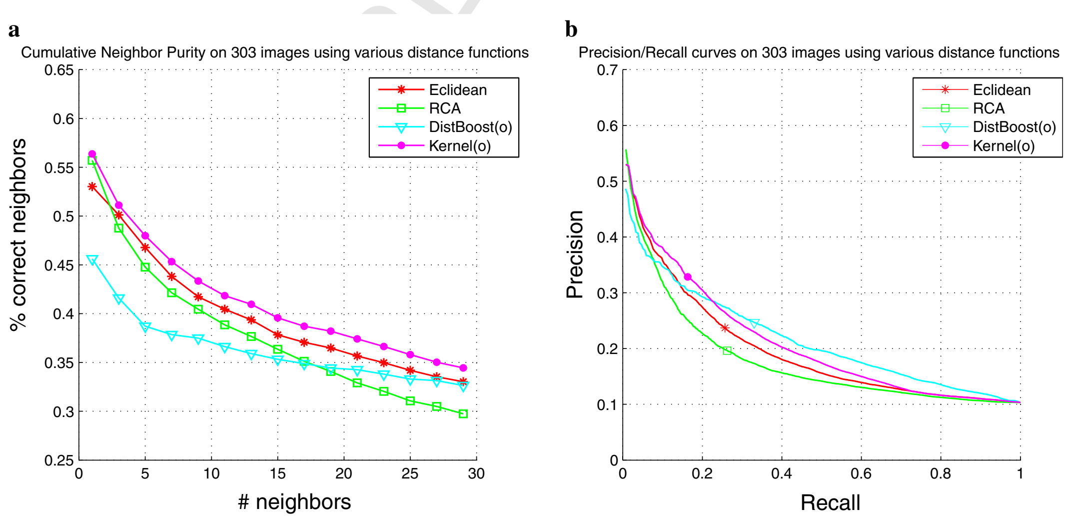

Fig. 4. Retrieval results on the first image database based on a separate set of query images. (a) Cumulative neighbor purity curves; (b) precision/recall curves.

image forming a similar (dissimilar) image pair. The set of similar and dissimilar image pairs (or pairwise similarity and dissimilarity constraints) is incrementally built up as relevance feedback is collected from users. Thus, later retrieval tasks can make use of an increasing set of similar and dissimilar image pairs for metric learning. Fig. 5 gives a functional diagram that summarizes how metric learning can be realized in CBIR systems.

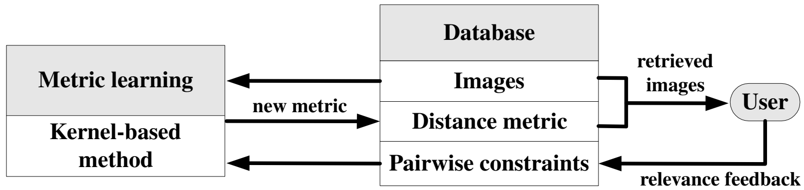

Fig. 5. Functional diagram for metric learning in CBIR.

4. Stepwise metric learning for image retrieval

The kernel-based metric learning algorithm incorporates pairwise constraints to perform metric learning. In the experiments performed in Section 3, we accumulate the similarity constraints over multiple query sessions before applying metric learning. Experimental results show that more pairwise constraints can lead to greater improvement. However, this also implies higher computational demand.

4.1. Stepwise kernel-based metric learning

As a compromise, we can perform stepwise kernel-based metric learning by incorporating the pairwise constraints in reasonably small, incremental batches each of a certain size ω. Whenever the batch of newly collected pairwise constraints reaches this size, metric learning will be performed with this batch to obtain a new metric. The batch of similarity constraints is then discarded. This process will be repeated continuously with the arrival of more relevance feedback from users. In so doing, knowledge acquired from relevance feedback in one session can be best utilized to give long-term improvement in subsequent sessions. This stepwise metric adaptation algorithm is summarized in Fig. 6.

4.2. Evaluation on CBIR tasks

To evaluate the stepwise kernel-based metric learning algorithm described above, we devise an automatic evaluation scheme to simulate a typical CBIR system with the relevance feedback mechanism implemented. More specifically, for a prespecified maximum batch size ω, we randomly select ω images from the database as query images. In each query session based on one of the ω images, the system returns the top 20 images from the database based on the current distance function, which is Euclidean initially. Of these 20 images, five relevant images are then randomly chosen, simulating the relevance feedback process performed by a user.[3] Our kernel-based metric learning method is performed once after every ω sessions.

Input: Image database $\mathcal{X}$, maximum batch size ω

Begin

Set Euclidean metric as initial distance metric

Repeat {

Obtain relevance feedback from new query session

Save relevance feedback to current batch

If batch size $= \omega$

Adapt distance metric by kernel-based metric learning

Clear current batch of feedback information

}

End

Fig. 6. Stepwise kernel-based metric learning algorithm for boosting image retrieval performance.

Fig. 7 shows the cumulative neighbor purity curves for the retrieval results on the Corel image database based on stepwise metric learning with different maximum batch sizes ω. As we can see, long-term metric learning based on stepwise metric learning can result in continuous improvement of retrieval performance. Moreover, to incorporate the same amount of relevance feedback from users, it seems more effective to use larger batch sizes. For example, after incorporating 40 query sessions from the same starting point, the final metric (metric_4) of Fig. 7(a) is not as good as that (metric_2) of Fig. 7(b), which in turn is (slightly) worse than that of Fig. 7(c). Thus, provided that the computational resources permit, one should perform each metric learning step using relevance feedback from more query sessions.

5. Concluding remarks

In this paper, we have proposed an efficient kernel-based distance metric learning method and demonstrated its promising performance for CBIR tasks. Not only does our method based on semi-supervised metric learning improve the retrieval performance of Euclidean distance without distance learning, it also outperforms other distance learning methods significantly due to its higher flexibility in metric learning. Moreover, unlike most existing relevance feedback methods which only improve the retrieval results within a single query

[3] In real-world CBIR tasks, users intuitively select the most relevant images from the returned (say top 20) images. The selected images are not necessarily the nearest ones computed based on the (learned) distance metric. To simulate real-world CBIR tasks, we use five randomly selected images as relevance feedback from the user. In fact, for the purpose of metric learning, selecting more "distant" yet relevant images as similar pairs is even better, as the distance metric can be improved to a greater extent in the subsequent metric learning process.

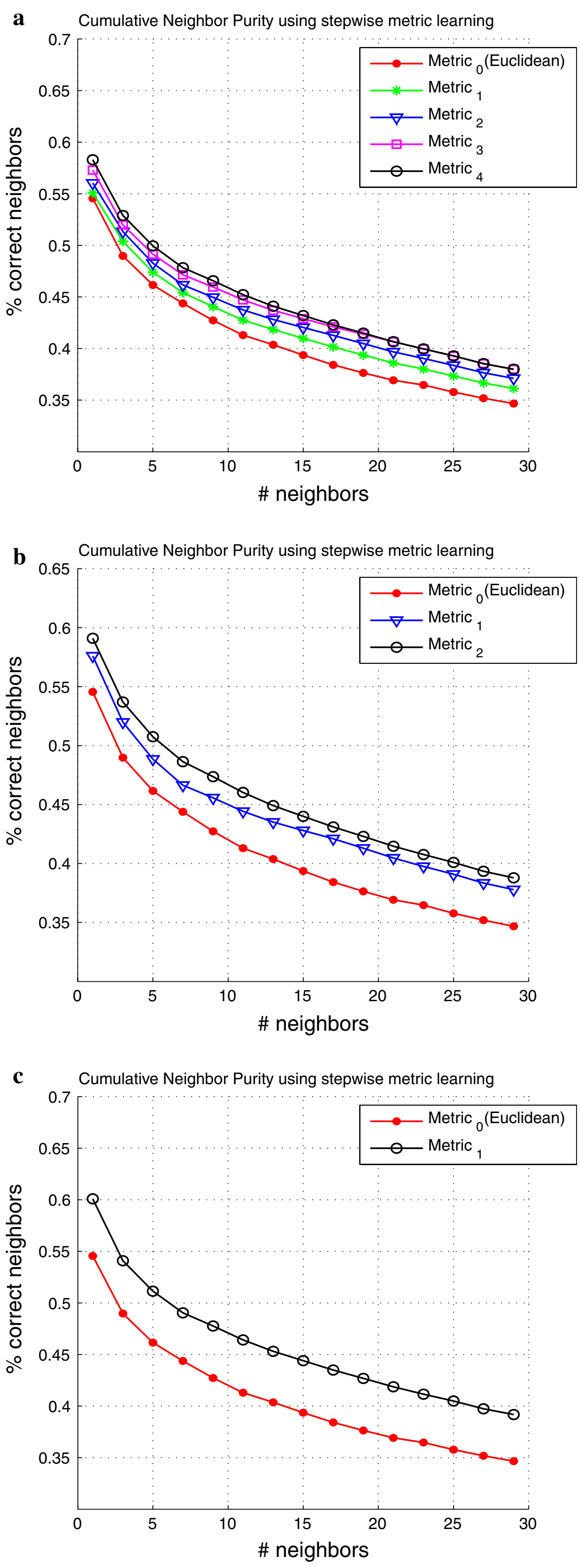

Fig. 7. Retrieval results based on stepwise kernel-based metric learning with different maximum batch sizes. (a) $\omega = 10$ sessions; (b) $\omega = 20$ sessions; (c) $\omega = 40$ sessions.

session, we propose a stepwise metric learning algorithm to boost the retrieval performance continuously by accumulating relevance feedback collected over multiple query sessions.

Despite its promising performance, there is still room to further enhance our proposed method. In our kernel method, the kernel PCA embedding step does not make use of the supervisory information available. One potential direction to pursue is to combine the two steps into one using the kernel trick and reformulate the metric learning problem as a kernel learning problem. Other possible research directions include applying the idea of kernel-based metric learning to other pattern recognition tasks.

Acknowledgements

The research described in this paper has been supported by two grants, CA03/04.EG01 (which is part of HKBU2/03/C) and HKUST6174/04E, from the Research Grants Council of the Hong Kong Special Administrative Region, China.

References

[1] A. Smeulders, M. Worring, S. Santini, A. Gupta, R. Jain, Content-based image retrieval at the end of the early years, IEEE Transactions on Pattern Analysis and Machine Intelligence 22 (12) (2000) 1349–1380.
[2] M.J. Swain, D.H. Ballard, Color indexing, International Journal of Computer Vision 7 (1) (1991) 11–32.
[3] J. Hafner, H.S. Sawhney, W. Equitz, M. Flickner, W. Niblack, Efficient color histogram indexing for quadratic from distance functions, IEEE Transactions on Pattern Analysis and Machine Intelligence 17 (7) (1995) 729–736.
[4] M. Stricker, M. Orengo, Similarity of color images, in: Storage and Retrieval for Image and Video Databases (SPIE), vol. 2, 1995, pp. 381–392.
[5] Y. Rui, T.S. Huang, M. Ortega, S. Mehrotra, Relevance feedback: a power tool for interactive content-based image retrieval, IEEE Transactions on Circuits and Systems for Video Technology 8 (5) (1998) 644–655.
[6] Y. Ishikawa, R. Subramanya, C. Faloutsos, Mindreader: query databases through multiple examples, in: Proceedings of the 24th VLDB Conference, 1998.
[7] A. Doulamis, N. Doulamis, T. Varvarigou, Efficient content-based image retrieval using fuzzy organization and optimal relevance feedback, International Journal of Image and Graphics 3 (1) (2003) 1–38.
[8] N. Doulamis, A. Doulamis, Fuzzy histograms and optimal interactive relevance feedback, IEEE Transactions on Image Processing, to appear.
[9] D.R. Heisterkamp, J. Peng, H.K. Dai, Adaptive quasiconformal kernel metric for image retrieval, in: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol. 2, 2001, pp. 388–393.
[10] A. Dong, B. Bhanu, A new semi-supervised EM algorithm for image retrieval, in: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol. 2, 2003, pp. 662–667.
[11] J. Yang, Q. Li, Y. Zhuang, Towards data-adaptive and user-adaptive image retrieval by peer indexing, International Journal of Computer Vision 56 (1/2) (2004) 47–63.
[12] X. He, Incremental semi-supervised subspace learning for image retrieval, in: Proceedings of the 12th Annual ACM International Conference on Multimedia, 2004, pp. 2–8.
[13] X. He, W.Y. Ma, H.J. Zhang, Learning an image manifold for retrieval, in: Proceedings of the 12th Annual ACM International Conference on Multimedia, 2004, pp. 17–23.

[14] X.S. Zhou, T.S. Huang, Small sample learning during multimedia retrieval using biasmap, in: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2001, pp. 11–17.

[15] G. Guo, A.K. Jain, W. Ma, H. Zhang, Learning similarity measure for natural image retrieval with relevance feedback, IEEE Transactions on Neural Networks 13 (4) (2002) 811–820.

[16] D. Tao, X. Tang, Random sampling based SVM for relevance feedback image retrieval, in: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol. 2, 2004, pp. 647–652.

[17] S. Tong, E. Chang, Support vector machine active learning for image retrieval, in: Proceedings of the Ninth ACM International Conference on Multimedia, 2001, pp. 107–118.

[18] K. Tieu, P. Viola, Boosting image retrieval, International Journal of Computer Vision 56 (1/2) (2004) 17–36.

[19] A. Bar-Hillel, T. Hertz, N. Shental, D. Weinshall, Learning distance functions using equivalence relations, in: Proceedings of the Twentieth International Conference on Machine Learning, Washington, DC, USA, 21–24 August 2003, pp. 11–18.

[20] T. Hertz, N. Shental, A. Bar-Hillel, D. Weinshall, Enhancing image and video retrieval: learning via equivalence constraints, in: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol. 2, Madison, WI, USA, 18–20 June 2003, pp. 668–674.

[21] K. Wagstaff, C. Cardie, S. Rogers, S. Schroedl, Constrained k-means clustering with background knowledge, in: Proceedings of the Eighteenth International Conference on Machine Learning, Williamstown, MA, USA, 2001, pp. 577–584.

[22] E.P. Xing, A.Y. Ng, M.I. Jordan, S. Russell, Distance metric learning with application to clustering with side-information, in: S. Becker, S. Thrun, K. Obermayer (Eds.), Advances in Neural Information Processing Systems, vol. 15, MIT Press, Cambridge, MA, USA, 2003, pp. 505–512.

[23] T. Hertz, A. Bar-Hillel, D. Weinshall, Boosting margin based distance functions for clustering, in: Proceedings of the Twenty-First International Conference on Machine Learning, Banff, Alberta, Canada, 4–8 August 2004, pp. 393–400.

[24] T. Hertz, A. Bar-Hillel, D. Weinshall, Learning distance functions for image retrieval, in: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol. 2, Washington, DC, USA, 27 June–3 July 2004, pp. 570–577.

[25] X. He, O. King, W.Y. Ma, M. Li, H.J. Zhang, Learning a semantic space from user's relevance feedback, IEEE Transactions on Circuits and Systems for Video Technology 13 (1) (2003) 39–48.

[26] C.H. Hoi, M.R. Lyu, A novel log-based relevance feedback technique in content-based image retrieval, in: Proceedings of the 12th Annual ACM International Conference on Multimedia, 2004, pp. 24–31.

[27] B. Schölkopf, A.J. Smola, K.-R. Müller, Nonlinear component analysis as a kernel eigenvalue problem, Neural Computation 10 (1998) 1299–1319.

[28] G. Pass, R. Zabih, J. Miller, Comparing images using color coherence vectors, in: Proceedings of the Fourth ACM International Conference on Multimedia, 1996, pp. 65–73.