Learning Distance Metrics with Contextual Constraints for Image Retrieval

Steven C. H. Hoi[†], Wei Liu[†], Michael R. Lyu[†] and Wei-Ying Ma[‡]

[†]Chinese University of Hong Kong, Hong Kong
[‡]Microsoft Research Asia, Beijing, P.R. China

Abstract

Relevant Component Analysis (RCA) has been proposed for learning distance metrics with contextual constraints for image retrieval. However, RCA has two important disadvantages. One is the lack of exploiting negative constraints which can also be informative, and the other is its incapability of capturing complex nonlinear relationships between data instances with the contextual information. In this paper, we propose two algorithms to overcome these two disadvantages, i.e., Discriminative Component Analysis (DCA) and Kernel DCA. Compared with other complicated methods for distance metric learning, our algorithms are rather simple to understand and very easy to solve. We evaluate the performance of our algorithms on image retrieval in which experimental results show that our algorithms are effective and promising in learning good quality distance metrics for image retrieval.

1 Introduction

Machine learning algorithms have been popularly applied to image retrieval for bridging the semantic gap between low-level image features and high-level semantic concepts [15]. Many machine learning algorithms, such as k-Means and k-Nearest Neighbor, usually define some distance metrics or functions to measure the similarity of data instances. For example, Euclidean distance is often used for distance measure in many applications. Typically, a good quality distance metric can influence the performance of the learning algorithm significantly. Thus, it is important to choose appropriate distance metrics when applying a learning algorithm to image retrieval under given different contexts [2].

Many research tasks in image retrieval are required to choose a good distance metric or function in order to solve the problems effectively. The first widely studied task is data clustering under unsupervised settings [9]. A suited distance metric can importantly improve the performance of the clustering algorithms, such as k-Means or graph based techniques [13]. The second application is for supervised classification tasks. Choosing a good distance metric is also critical for these tasks. For example, face recognition or general image classification tasks usually use distance based techniques, such as k-Nearest Neighbor, whose performance normally relies on the given distance metric. Moreover, many retrieval tasks in multimedia information retrieval also need to learn a good distance metric in order to retrieve the users' query targets effectively. In content-based image retrieval (CBIR), images are usually represented by low-level features, such as color, texture, and shape. It is simply too restricted to employ the rigid Euclidean distance to measure distances of images. Learning effective distance metrics for image retrieval has attracted more and more attentions in recent years [5].

Here we illustrate an example to show that different distance metrics are important for the applications with different contexts. Figure 1 shows an example of grouping the data instances on different contextual conditions. Figure 1 (a) is the given data. Figure 1 (b)-(d) show three different grouping results under different context environments, e.g., (b) groups by proximity, (c) groups by shape, (d) groups by size. This example shows that it is important for the clustering algorithms to choose the right distance metrics for achieving the correct group results under different contextual information.

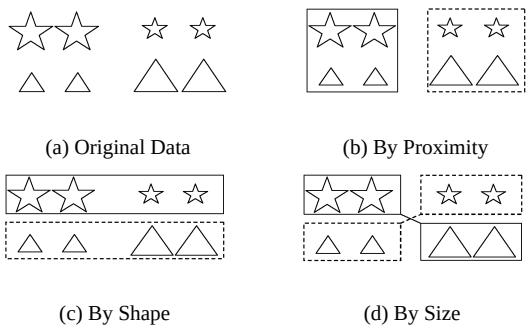


Figure 1. Clustering with different contexts.

In general, the approach to find a good distance metric for various learning algorithms is equivalent to looking for a good data transformation function $f : X \longmapsto Y$, which transforms the data X into another representation of Y [2]. These two problems can be solved together in a unified framework. Hence, our goal is to find a good distance metric which not only can be used for similarity measure of data, but also can transform the data into another better representation of the original data.

For learning distance metrics and data transformation, traditional techniques normally need to acquire explicit class labels. However, in many real-world applications, explicit class labels might be too expensive to be obtained. For example, in image retrieval, obtaining the exact class label of images is usually quite expensive due to the difficulty of image annotation. However, it is much easier to know the relevance relationship between images, which can be obtained from the logs of user relevance feedback [6, 7]. Therefore, it is more attractive to learn the distance metrics or data transformation directly from the pairwise constraints without using explicit class labels.

In this paper we study the problem of learning distance metrics from contextual constraints among data instances. We first propose Discriminative Component Analysis (DCA) to learn the linear data transformation for the optimal Mahalanobis distance metric with contextual information. Based on DCA, we further develop Kernel DCA to learn the nonlinear distance metric by kernel transformations.

2 Related Work

The problems for learning distance metrics and data transformation have become more and more popular in recent research due to their broad applications. One kind of approaches is to use the class labels of data instances to learn distance metrics in supervised classification settings. We briefly introduce several traditional methods. Hastie et al. [4] and Jaakkola et al. [8] used the labeled data instances to learn distance metrics toward classification tasks. Tishby et al. [16] considered the joint distribution of two random variables X and Y to be known, and then learned a compact representation of X that enjoys high relevance of Y. Most recently, Goldberger et al. [3] proposed the Neighborhood Component Analysis to learn a distance measure for kNN classification by directly maximizing a stochastic variant of the leave-one-out kNN score on the training set. Zhou et al. proposed a kernel partial alignment scheme to learn kernel metrics for interactive image retrieval [23]. Most of these studies need to explicitly use the class labels as the side-information for learning the representations and distance metrics.

Recently, some work has addressed the problems of learning with contextual information in terms of pairwise constraints. Wagstaff et al. [18] suggested the K-means clustering algorithms by introducing the pairwise relations. Xing et al. [21] studied the problem of finding an optimal Mahalanobis metric from contextual constraints in combination with constrained K-means algorithm. But their method requires solving the convex optimization problem with gradient descent and iterative projections which often suffers from large computation cost. Later on, Bar-Hillel et al. [2] proposed a much simpler approach called Relevance Component Analysis (RCA), which enjoys comparable performance with Xing's method. As our approach is motivated by RCA, we will discuss it in detail below.

Let us first introduce some basic concepts. Mathematically, the Mahalanobis distance between two data instances is defined as:

$$d_M(\mathbf{x}_i, \mathbf{x}_j) = \sqrt{(\mathbf{x}_i - \mathbf{x}_j)^\top M (\mathbf{x}_i - \mathbf{x}_j)} \tag{1}$$

where M must be positive semi-definite to satisfy the properties of metric, i.e., non-negativity and triangle inequality. The matrix M can be decomposed as $M = A^\top A$, where A is a transformation matrix. The goal of RCA learning is to find an optimal Mahalanobis matrix M and the optimal data transformation matrix A using the contextual information.

The basic idea of RCA for learning the distance metric is to identify and down-scale global unwanted variability within the data. RCA changes the feature space used for data representation via a global linear transformation in which relevant dimensions are assigned with large weights [2]. The relevant dimensions are estimated by chunklets [2], each of them is defined as a group of data instances linked together with positive constraints. More specifically, given a data set $X = \{\mathbf{x}_i\}_{i=1}^N$ and n chunklets $C_j = \{\mathbf{x}_{ji}\}_{i=1}^{n_j}$, RCA computes the following matrix:

$$\hat{C} = \frac{1}{N} \sum_{j=1}^{n} \sum_{i=1}^{n_j} (\mathbf{x}_{ji} - \mathbf{m}_j)(\mathbf{x}_{ji} - \mathbf{m}_j)^\top \tag{2}$$

where $\mathbf{m}_j$ denotes the mean of the j-th chunklet, $\mathbf{x}_{ji}$ denotes the i-th data instance in the j-th chunklet and N is the number of data instances. The optimal linear transformation by RCA is then computed as $A = \hat{C}^{-\frac{1}{2}}$ and the Mahalanobis matrix is equal to the inverse of the matrix C, i.e., $M = \hat{C}^{-1}$. RCA is simple and effective for learning distance metrics and data transformation, yet it has two critical disadvantages. One is the lack of including the negative constraints which can provide important discriminative clues. The other is that RCA can learn only the linear relation between data instances which may be too restricted to discover the nonlinear relations in many applications. To this end, we propose the Discriminative Component Analysis (DCA) and Kernel DCA to overcome the two drawbacks.

Summary of Contributions. In this paper we study the problem of learning data transformations for distance

metrics with contextual constraints with application to image retrieval. We propose the Discriminative Component Analysis and Kernel Discriminative Component Analysis algorithms to learn both linear and nonlinear distance metrics. Our algorithms need no explicit class labels, which can be applicable to many broad applications. The rest of this paper is organized as follows. Section 3 formulates the Discriminative Component Analysis and presents the algorithm. Section 4 suggests kernel transformations to extend DCA for learning nonlinear distance metrics. Section 5 discusses our experimental evaluations on image retrieval. Section 6 concludes this work.

3 Discriminative Component Analysis

3.1 Overview

Let us first give an overview of the concept of Discriminative Component Analysis (DCA). In the settings of DCA learning, we assume the data instances are given with contextual constraints which indicate the relevance relationship (positive or negative) between data instances. According to the given constraints, one can group the data instances into chunklets by linking the data instances together with positive constraints. The basic idea of DCA is to learn an optimal data transformation that leads to the optimal distance metric by both maximizing the total variance between the discriminative data chunklets and minimizing the total variance of data instances in the same chunklets. In the following part, we formalize the approach of DCA and present the algorithm to solve the DCA problem.

3.2 Formulation

Assume we are given a set of data instances $X = \{\mathbf{x}_i\}_{i=1}^N$ and a set of contextual constraints. Assume that n chunklets can be formed by the positive constraints among the given constraints. For each chunklet, a discriminative set is formed by the negative constraints to represent the discriminative information. For example, for the j-th chunklet, each element in the discriminative set D_j indicates one of n chunklets that can be discriminated from the j-th chunklet. Here, a chunklet is defined to be discriminated from another chunklet if there is at least one negative constraint between them. Note that RCA can be considered as a special case of DCA in which all discriminative sets are empty sets that ignore all negative constraints.

To perform Discriminative Component Analysis, two covariance matrices $\hat{C}_b$ and $\hat{C}_w$ are defined to calculate the total variance between data of the discriminative chunklets and the total variance of data among the same chunklets respectively. These two matrices $\hat{C}_b$ and $\hat{C}_w$ are computed as follows:

$$\begin{aligned} \hat{C}_b &= \frac{1}{n_b}\sum_{j=1}^{n}\sum_{i\in D_j}(\mathbf{m}_j-\mathbf{m}_i)(\mathbf{m}_j-\mathbf{m}_i)^\top \\ \hat{C}_w &= \frac{1}{n}\sum_{j=1}^{n}\frac{1}{n_j}\sum_{i=1}^{n_j}(\mathbf{x}_{ji}-\mathbf{m}_j)(\mathbf{x}_{ji}-\mathbf{m}_j)^\top \end{aligned} \tag{3}$$

where $n_b = \sum_{j=1}^{n}|D_j|$, $|\cdot|$ denotes the cardinality of a set, $\mathbf{m}_j$ is the mean vector of the j-th chunklet, i.e., $\mathbf{m}_j = \frac{1}{n_j}\sum_{i=1}^{n_j}\mathbf{x}_{ji}$, $\mathbf{x}_{ji}$ is the i-th data instance in the j-th chunklet, and D_j is the discriminative set in which each element is one of n chunklets that has at least one negative constraint to the j-th chunklet.

The idea of Discriminative Component Analysis is to look for a linear transformation that leads to an optimal distance metric by both maximizing the total variance of data between the discriminative chunklets and minimizing the total variance of data among the same chunklets. The DCA learning task leads to solve the optimization as follows:

$$J(A) = \arg\max_A \frac{|A^\top \hat{C}_b A|}{|A^\top \hat{C}_w A|}\ , \tag{4}$$

where A denotes the optimal transformation matrix to be learned. When the optimal transformation A is solved, it leads to obtain the optimal Mahalanobis matrix $M = A^\top A$.

3.3 Algorithm

According to the Fisher theory [11, 12], the optimal solution in Equation (4) is corresponding to the transformation matrix that diagonalizes both the covariance matrices $\hat{C}_b$ and $\hat{C}_w$ simultaneously [10]. To obtain the solution effectively, we propose an algorithm to find the optimal transformation matrix, which was used to solve LDA in the previous study [22]. The details of our algorithm are shown in **Algorithm 1**.

In our algorithm, a matrix U is first found to diagonalize the covariance matrix $\hat{C}_b$ of between-chunklets. After discarding the column vectors with zero eigenvalues, we can obtain a $k*k$ principal sub-matrix D_b of the original diagonal matrix. This procedure leads to obtain a set of projected subspaces, i.e., $Z = RD_b^{-1/2}$, that can best discriminate the chunklets. Further, we form a matrix $C_z = Z^\top \hat{C}_w Z$ and find a matrix V to diagonalize the matrix C_z. If dimension reduction is required, such that r is the desired dimensionality, then we extract the first r column vectors of V with the smallest eigenvalues to form a lower rank matrix $\hat{V}$. This leads to obtain the reduced diagonal matrix $D_w = \hat{V}^\top C_z \hat{V}$. Finally, the optimal transformation matrix and the optimal Mahalanobis Matrix are given as $A = Z\hat{V}D_w^{-1/2}$ and $M = A^\top A$, respectively.

Algorithm 1:The DCA Algorithm

Input

- a set of N data instances: $X = \{\mathbf{x}_i\}_{i=1}^N$
- n chunklets C_j and discriminative sets D_j, j=1,. . .,n

Output

- optimal transformation matrix A
- optimal Mahalanobis matrix M

Procedure

1. Compute $\hat{C}_b$ and $\hat{C}_w$ by Equation (3) ;
2. Diagonalize $\hat{C}_b$ by eigenanalysis
   2.1. Find U to satisfy $U^\top \hat{C}_b U = \Lambda_b$ and $U^\top U = I$, here Λ_b is a diagonal matrix sorted in increasing order ;
   2.2. Form a matrix $\hat{U}$ by the last k column vectors of U with nonzero eigenvalues ;
   2.3. Let $D_b = \hat{U}^\top \hat{C}_b \hat{U}$ be the $k * k$ submatrix of Λ_b ;
   2.4. Let $Z = \hat{U} D_b^{-1/2}$ and $C_z = Z^\top \hat{C}_w Z$;
3. Diagonalize C_z by eigenanalysis
   3.1. Find V to satisfy $V^\top C_z V = \Lambda_w$ and $V^\top V = I$, here Λ_w is a diagonal matrix sorted in decreasing order ;
   3.2. If dimension reduction is needed, assume the desired dimension is r, then form $\hat{V}$ by the first r column vectors of V with the smallest eigenvalues and let $D_w = \hat{V}^\top C_z \hat{V}$; otherwise, let $\hat{V} = V$ and $D_w = \Lambda_w$;
4. Final Outputs
   $A = Z\hat{V} D_w^{-1/2}$ and $M = A^\top A$.

End of Algorithm

4 Kernel DCA

4.1 Overview

Similar to the RCA learning [2], DCA is so far also a linear technique that is insufficient to discover nonlinear relationships among real-world data. In the machine learning area, the kernel trick is a powerful tool to learn the complex nonlinear structures from the input data [17, 14]. In the literature, the kernel trick has been successfully applied on many linear analysis techniques, such as Kernel Principal Component Analysis (PCA) [19], Kernel Fisher Discriminant Analysis [10, 12], Support Vector Machines [17], Kernel Independent Component Analysis [1], etc. Similar to these approaches, we can also apply the kernel trick on DCA toward more powerful analysis performance in real-world applications.

In general, the kernel technique first maps input data into a high dimensional feature space. A linear technique applied on the data in the feature space is able to achieve the goal of nonlinear analysis. For example, in Kernel PCA, input data are first projected into an implicit feature space via the kernel trick, then the linear PCA is applied on the projected feature space to extract the principal components in the feature space. This enables the Kernel PCA to extract the nonlinear principal components in the input data space using the kernel trick.

Similar to the kernel techniques, we propose the Kernel Discriminative Component Analysis (KDCA) to overcome the disadvantage of RCA and DCA by applying the kernel trick. We first project input data into an implicit feature space via the kernel trick. Then the linear DCA is applied on the projected feature space to find the optimal linear transformation in the feature space. Consequently, we are able to find the nonlinear structures of the given data using the Kernel DCA technique.

4.2 Formulation

Let us now formulate Kernel Discriminative Component Analysis formally. Typically, a kernel-based analysis technique usually implicitly maps original data in input space I to a high-dimensional feature space F via some basis function $\phi : \mathbf{x} \rightarrow \phi(\mathbf{x}) \in F$. The similarity measure of data in the projected feature space is achieved by the kernel function which is defined as an inner product between two vectors in the projected space F as follows:

$$K(\mathbf{x}_i, \mathbf{x}_j) = (\phi(\mathbf{x}_i), \phi(\mathbf{x}_j)). \tag{5}$$

Assume that a set of N data instances $X = \{\mathbf{x}_i\}_{i=1}^N$ is given in an original input space I. To do kernel DCA learning, we first choose a basis function ϕ to map the data in the original input space I to a high-dimensional feature space F. For any two data instances, we compute their distance via the kernel function defined in the projected feature space as follows:

$$d_\phi(\mathbf{x}_i, \mathbf{x}_j) = \sqrt{(\phi(\mathbf{x}_i) - \phi(\mathbf{x}_j))^\top M (\phi(\mathbf{x}_i) - \phi(\mathbf{x}_j))} \tag{6}$$

where M is a full rank matrix that must be positive semi-definite to satisfy the metric property and is often formed by a transformation matrix W. The linear transformation matrix W can be represented as $W = [\mathbf{w}_1, \ldots, \mathbf{w}_m]^\top$ in which each of the m column vectors is a span of all l training samples in the feature space, such that

$$\mathbf{w}_i = \sum_{j=1}^{l} \alpha_{ij} \phi_j \ , \tag{7}$$

where α_{ij} are the coefficients to be learned in the feature space. Therefore, for a given data instance $\mathbf{x}$, its projection onto the i-th direction $\mathbf{w}_i$ in the feature space can be computed as follows:

$$(\mathbf{w}_i \cdot \phi(\mathbf{x})) = \sum_{j=1}^{l} \alpha_{ij} K(\mathbf{x}_j, \mathbf{x}) \ . \tag{8}$$

Hence, Equation (6) can be represented as

$$d_\phi(\mathbf{x}_i, \mathbf{x}_j) = \sqrt{(\vec{\tau}_i - \vec{\tau}_j)^\top M (\vec{\tau}_i - \vec{\tau}_j)} \ , \tag{9}$$

where $\vec{\tau}_i = [K(\mathbf{x}_1, \mathbf{x}_i), \ldots, K(\mathbf{x}_1, \mathbf{x}_l)]^\top$, and A is the linear transformation matrix formed by $A = [\vec{\alpha}_1, \ldots, \vec{\alpha}_m]$ in which $\vec{\alpha}_i = [\alpha_{i1}, \ldots, \alpha_{il}]^\top$. Hence, we can similarly compute the two covariance matrices in the projected feature space as follows:

$$\begin{aligned} K_b &= \frac{1}{n_b} \sum_{j=1}^{n} \sum_{i \in D_j} (\vec{u}_j - \vec{u}_i)(\vec{u}_j - \vec{u}_i)^\top \\ K_w &= \frac{1}{n} \sum_{j=1}^{n} \frac{1}{n_j} \sum_{i=1}^{n_j} (\vec{\tau}_j - \vec{u}_i)(\vec{\tau}_j - \vec{u}_i)^\top \end{aligned} \tag{10}$$

where $\vec{u}_j = [\frac{1}{n_j} \sum_{i=1}^{n_j} K(\mathbf{x}_1, \mathbf{x}_i), \ldots, \frac{1}{n_j} \sum_{i=1}^{n_j} K(\mathbf{x}_l, \mathbf{x}_i)]^\top$ denotes the mean vector. Consequently, the Kernel DCA task leads to solve the optimization problem as follows:

$$J(A) = \arg\max_A \frac{|A^\top K_b A|}{|A^\top K_w A|} \ . \tag{11}$$

Solving the above optimization problem gives the optimal linear transformation A in the projected space. It also leads to the optimal Mahalanobis matrix in the projected space.

4.3 Algorithm

The method to solve the optimization of Kernel DCA is similar to that for the linear DCA, i.e., to find the linear transformation matrix A that can diagonalize both K_b and K_w. For limited space, please kindly refer to **Algorithm 2** for the details of Kernel DCA algorithm.

Algorithm 2:The Kernel DCA Algorithm

Input

- a set of N data instances: $X = \{\mathbf{x}_i\}_{i=1}^N$
- n chunklets C_j and discriminative sets D_j, j=1,. . .,n

Output

- optimal transformation matrix A
- optimal Mahalanobis matrix M

Procedure

1. Compute K_b and K_w by Equation (10) ;
2. Diagonalize K_b by eigenanalysis
   2.1. Find U to satisfy $U^\top K_b U = \Lambda_b$ and $U^\top U = I$, here Λ_b is a diagonal matrix sorted in increasing order ;
   2.2. Form a matrix $\hat{U}$ by the last k column vectors of U with nonzero eigenvalues ;
   2.3. Let $D_b = \hat{U}^\top K_b \hat{U}$ be the $k * k$ submatrix of Λ_b ;
   2.4. Let $Z = \hat{U} D_b^{-1/2}$ and $K_z = Z^\top K_w Z$;
3. Diagonalize K_z by eigenanalysis
   3.1. Find V to satisfy $V^\top K_w V = \Lambda_w$ and $V K_z V = I$, here Λ_b is a diagonal matrix sorted in decreasing order ;
   3.2. If dimension reduction is needed, assume the desired dimension is r, then form $\hat{V}$ by the first r column vectors of V with the smallest eigenvalues and let $D_w = \hat{V}^\top K_z \hat{V}$; otherwise, let $\hat{V} = V$ and $D_w = \Lambda_w$;
4. Final Outputs
   $A = Z\hat{V} D_w^{-1/2}$ and $M = A^\top A$.

End of Algorithm

5 Experimental Results

To evaluate the performance of our algorithms, we conduct empirical evaluation of learning distance metrics for content-based image retrieval in comparisons with traditional methods in distance metric learning [21, 2]. We describe the details of our empirical evaluation below.

5.1 Experimental Testbed

To test the performance of DCA and Kernel DCA for learning distance metrics for image retrieval, we employ an image dataset from COREL image CDs. 10 image categories are selected to form our dataset, such as *dogs*, *cats*, *horses*, etc. Each of them has a distinct semantic meaning and contains 100 images. In total, 1000 images are engaged in our dataset.

For image retrieval, low-level feature representation is critical. In our experiment, three kinds of low-level features are extracted: color, shape, and texture. For color, we extract the color moments: color mean, color variance and color skewness in each color channel (H, S, and V). Thus, 9-dimensional color moment features are used. For shape, we use the edge direction histogram. Canny edge detector is applied to obtain the edges. Then 18-dimensional edge direction histogram features are computed to represent the shapes. For texture, we use the wavelet-based texture features. The Discrete Wavelet Transformation (DWT) is applied on the gray images of original images by a Daubechies-4 wavelet filter. In total, we perform 3-level decompositions and extract 9-dimension wavelet-based texture features for each image. All together, we use 36 features to represent images in our experiment.

5.2 Performance Evaluation

We now empirically evaluate our algorithms for learning distance metrics with contextual constraints in image retrieval. Although our application is on image retrieval, our algoirthms can also be beneficial to other information retrieval tasks.

In our experiments, six different retrieval methods are compared as follows: (1) Euclidean: retrieval by Euclidean metric; (2) RCA: retrieval by the metric of RCA; (3) Xing: retrieval by the metric of Xing's method with nonlinear optimization [21]; (4) DCA: retrieval by the metric of DCA; (5) KDCA: retrieval by the nonlinear metric of KDCA.

For a real-world image retrieval application, contextual information can be obtained easily. For example, a CBIR

Table 1. Performance Evaluation for Image Retrieval (Average Precision on TOP 20 Returned Images.)

Category	Euclidean	RCA	Xing	DCA	KDCA
Dogs	0.420	0.455 (+8.3%)	0.390 (-7.1%)	0.500 (+19.0%)	**0.600 (+42.9%)**
Cats	0.495	0.590 (+19.2%)	**0.640 (+29.3%)**	0.600 (+21.2%)	**0.640 (+29.3%)**
Horses	0.775	**0.865 (+11.6%)**	0.830 (+7.1%)	0.850 (+9.7%)	0.820 (+5.8%)
Eagles	0.575	0.595 (+3.5%)	**0.665 (+15.7%)**	0.590 (+2.6%)	0.625 (+8.7%)
Penguins	0.215	0.465 (+116.3%)	0.260 (+20.9%)	**0.470 (+118.6%)**	0.325 (+51.2%)
Roses	0.505	0.570 (+12.9%)	0.545 (+7.9%)	**0.610 (+20.8%)**	**0.610 (+20.8%)**
Mountain	0.505	0.605 (+19.8%)	0.570 (+12.9%)	0.635 (+25.7%)	**0.670 (+32.7%)**
Sunset	**0.570**	0.365 (-36.0%)	0.560 (-1.8%)	0.395 (-30.7%)	0.510 (-10.5%)
Butterfly	0.310	0.395 (+27.4%)	0.345 (+11.3%)	0.390 (+25.8%)	**0.430 (+38.7%)**
Balloon	0.260	0.240 (-7.7%)	0.265 (+1.9%)	0.240 (-7.7%)	**0.320 (+23.1%)**
MAP	0.463	0.515 (+11.1%)	0.507 (+9.5%)	0.528 (+14.0%)	**0.555 (+19.9%)**

system often provides the relevance feedback function for users. The relevance feedback records can then be logged for learning the distance metrics [6]. In our experiment, to enable objective evaluation, we generate the contextual constraints automatically according to the ground truth of the image datasets. In total, we generate 1% positive constraints and 1% negative constraints. For performance evaluation, we employ the standard evaluation metric for image retrieval [15, 6], i.e., retrieval precision, which is defined as the ratio of the number of relevant images over the number of returned images.

In our experiment, every image in each category is used as the query for retrieval. In total, 100 queries are performed for each category. We measure the average precision on the top returned images for each compared scheme. Figure 2 shows the evaluation curves of average retrieval performance on several semantic categories. For the experimental results, we can see that the DCA slightly outperforms the RCA approach while the Kernel DCA achieves the best performance in most cases.

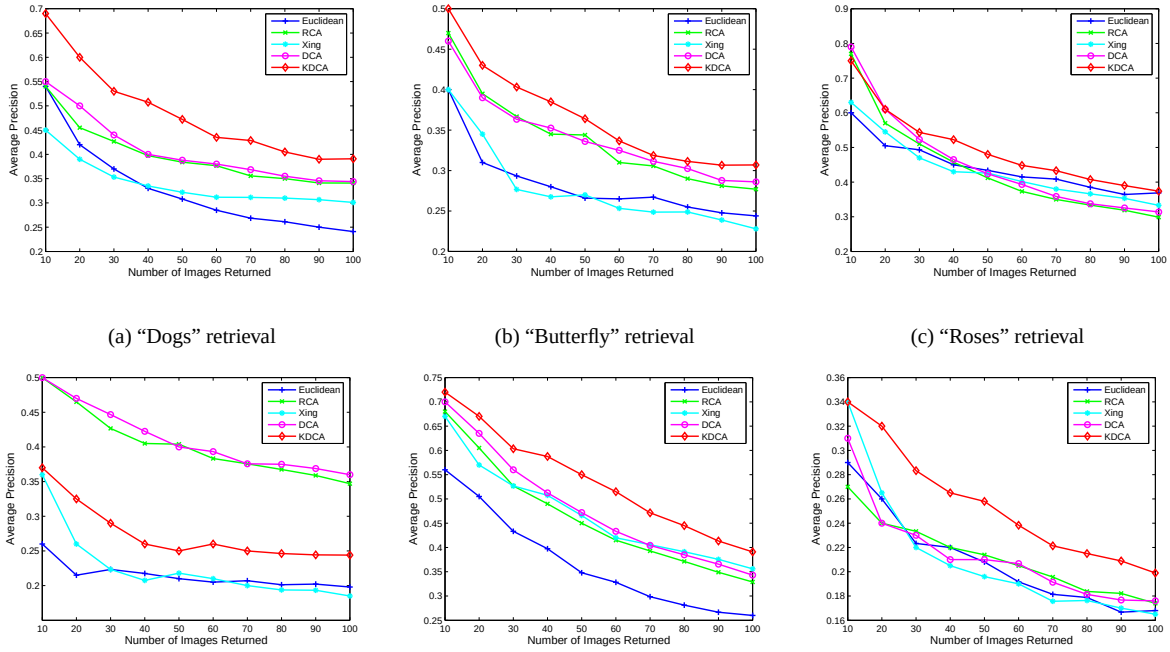
More specifically, we make a comparison on the TOP 20 returned images for each category. Table 1 shows the experimental results. We can see that RCA and Xing's method are comparable, in which RCA achieves 11.1% average improvement over the baseline approach, while Xing's method achieves 9.5% average improvement over the baseline approach. Our DCA algorithm achieves better results than both RCA and Xing's method, i.e., 14.0% average improvement over the baseline method. Our Kernel DCA method achieves the best performance among all, i.e., 19.9% average improvement over the baseline method. Note that there is an exceptional case, i.e., the "sunset" retrieval, in which all metric learning methods were fail to improve the performance. This may be caused by noisy features in the data. In sum, the overall results demonstrate our proposed methods are empirically more effective to learn good quality distance metrics than the traditional approaches for improving the performance of image retrieval. In the future work, we may compare our methods with other more complicated distance function learning techniques, such as kernel target alignments [20].

6 Conclusion

In this paper we studied the problem of learning distance metrics and data transformation using the contextual information for image retrieval. We addressed two important limitations in the previous approach of Relevance Component Analysis. One is the lack of exploiting negative constraints. Another is the limitation of learning the linear distance metrics which are not adequate for describing the complex nonlinear relations of real-world objects. To address the first problem, we proposed the Discriminative Component Analysis (DCA), which can exploit both positive and negative constraints in an efficient learning scheme. For solving the second problem, the Kernel DCA is proposed by applying the kernel trick on the linear DCA. We conducted extensive experiments to evaluate the performance of our algorithms on image retrieval. The promising results show that our algorithms are simple but quite effective in learning good quality metrics for image retrieval. In the future work, we will apply our methodology for other applications, such as data clustering and dimension reduction problems.

Acknowledgements

This work was done when Steven C. H. Hoi and Wei Liu were interns at Microsoft Research Asia. The work described in this paper was partially supported by two grants, one from the Shun Hing Institute of Advanced Engineering, and the other from the Research Grants Council of the Hong Kong Special Administrative Region, China (Project No. CUHK4205/04E).



(a) “Dogs” retrieval (b) “Butterfly” retrieval (c) “Roses” retrieval

(d) “Penguins” retrieval (e) “Mountain” retrieval (f) “Balloon” retrieval

Figure 2. Performance Evaluation for Image Retrieval (Average Precision on Top Returned Images).

References

[1] F. R. Bach and M. I. Jordan. Kernel independent component analysis. *JMLR*, 3:1–48, 2003.

[2] A. Bar-hillel, T. Hertz, N. Shental, and D. Weinshall. Learning a mahalanobis metric from equivalence constraints. *Journal of Machine Learning Research*, 6:937–965, 2005.

[3] J. Goldberger, S. Roweis, G. Hinton, and R. Salakhutdinov. Neighbourhood components analysis. In *Proc. NIPS*, pages 513–520, 2004.

[4] T. Hastie and R. Tibshirani. Discriminant adaptive nearest neighbor classification and regression. In *Proc. NIPS*, pages 409–415, 1996.

[5] T. Hertz, N. Shental, A. Bar-Hillel, and D. Weinshall. Enhancing image and video retrieval: Learning via equivalence constraints. In *Proc. IEEE CVPR*, pages 668–674, 2003.

[6] C. H. Hoi and M. R. Lyu. A novel log-based relevance feedback technique in content-based image retrieval. In *Proc. ACM Multimedia*, pages 10–16, New York, Oct. 2004.

[7] S. C. Hoi, M. R. Lyu, and R. Jin. A unified log-based relevance feedback scheme for image retrieval. *IEEE Trans. on Knowledge and Data Engineering*, 18(4):509–524, 2006.

[8] T. Jaakkola and D. Haussler. Exploiting generative models in discriminative classifiers. In *Proc. NIPS*, 1998.

[9] A. K. Jain and M. N. Murty. Data clustering: A review. *ACM Computing Surveys*, 32(3):264–323, 1999.

[10] Q. Liu, H. Lu, and S. Ma. Improving kernel fisher discriminant analysis for face recognition. *IEEE Trans. on Circuits and Systems for Video Technology*, 14(1):42–49, 2004.

[11] G. McLachlan. *Discriminant Analysis and Statistical Pattern Recognition*. John Wiley, 1992.

[12] S. Mika, G. Ratsch, J. Weston, B. Scholkopf, and K. Muller. Fisher discriminant analysis with kernels. In *Proc. IEEE NN for Signal Processing Workshop*, pages 41–48, 1999.

[13] A. Ng, M. Jordan, and Y. Weiss. On spectral clustering: Analysis and an algorithm. In *Proc. NIPS*, 2001.

[14] J. Shawe-Taylor and N. Cristianini. *Kernel Methods for Pattern Analysis*. Cambridge University Press, 2004.

[15] A. W. M. Smeulders, M. Worring, S. Santini, A. Gupta, and R. Jain. Content-based image retrieval at the end of the early years. *IEEE Trans. PAMI*, 22(12):1349–1380, 2000.

[16] N. Tishby, F. C. Pereira, and W. Bialek. The information bottleneck method. In *Proc. Annual Allerton Conf. on Communication, Control and Computing*, pages 368–377, 1999.

[17] V. N. Vapnik. *Statistical Learning Theory*. John Wiley, 1998.

[18] K. Wagstaff, C. Cardie, S. Rogers, and S. Schroedl. Constrained k-means clustering with background knowledge. In *Proc. 18th ICML*, pages 577–584, 2001.

[19] C. K. I. Williams. On a connection between kernel pca and metric multidimensional scaling. *Machine Learning*, 46(1–3):11–19, 2002.

[20] G. Wu, N. Panda, and E. Y. Chang. Formulating context-dependent similarity functions. In *ACM International Conference on Multimedia (MM)*, pages 725–734, 2005.

[21] E. P. Xing, A. Y. Ng, M. I. Jordan, and S. Russell. Distance metric learning with application to clustering with side-information. In *Proc. NIPS*, 2002.

[22] H. Yu and J. Yang. A direct lda algorithm for high-dimensional data with application to face recognition. *Pattern Recognition*, 34:2067–2070, 2001.

[23] X. S. Zhou, A. Garg, and T. S. Huang. A discussion of nonlinear variants of biased discriminants for interactive image retrieval. In *CIVR 2004 (LNCS 3115)*, pages 353–364, 2004.