Incremental Semi-Supervised Subspace Learning for Image Retrieval

Xiaofei He
Department of Computer Science,
The University of Chicago
xiaofei@cs.uchicago.edu

ABSTRACT
Subspace learning techniques are widespread in pattern recognition research. They include Principal Component Analysis (PCA), Locality Preserving Projection (LPP), etc. These techniques are generally unsupervised which allows them to model data in the absence of labels or categories. In relevance feedback driven image retrieval system, the user provided information can be used to better describe the intrinsic semantic relationships between images. In this paper, we propose a semi-supervised subspace learning algorithm which incrementally learns an adaptive subspace by preserving the semantic structure of the image space, based on user interactions in a relevance feedback driven query-by-example system. Our algorithm is capable of accumulating knowledge from users, which could result in new feature representations for images in the database so that the system's future retrieval performance can be enhanced. Experiments on a large collection of images have shown the effectiveness and efficiency of our proposed algorithm.

Categories and Subject Descriptors
H.3.1 [**Information Storage and Retrieval**]: Content Analysis and Indexing – *Algorithms, Indexing methods.*

General Terms
Algorithms, Measurement, Performance, Experimentation, Theory.

Keywords
Locality Preserving Projections, Image Retrieval, Relevance Feedback, Subspace Learning, Principal Component Analysis, Linear Discriminat Analysis

1. INTRODUCTION
Image representation and indexing have been fundamental problems for efficient clustering [4][7], classification and retrieval [3][12][13][14]. An image can be represented as a point in the vector space $\mathbf{R}^n$. Throughout this paper, we denote by *image space* the set of all image vectors. The image space is typically a subspace of $\mathbf{R}^n$, either linear or non-linear. It would be optimal to utilize clustering, classification and retrieval techniques in the subspace rather than the ambient space.

The typical subspace learning algorithms used for image indexing include Linear Discriminant Analysis (LDA) [1][15], Principal Component Analysis (PCA) [14][18], Locality Preserving Projections (LPP) [9][10], etc. PCA is an eigenvector method designed to model linear variation in high-dimensional data. PCA performs dimensionality reduction by projecting the original *n*-dimensional data onto the k ($<< n$)-dimensional linear subspace spanned by the leading eigenvectors of the data's covariance matrix. Its goal is to find a set of mutually orthogonal basis functions that capture the directions of maximum variance in the data. If the image space is a linearly embedded manifold, PCA is guaranteed to uncover the dimensionality of the manifold and produces a compact representation. Unfortunately, the image space is probably highly nonlinear. In such a case, PCA fails to uncover the intrinsic manifold structure of the image space. Moreover, PCA is unsupervised. Therefore, the subspace obtained by PCA can not reflect human perception.

Different from PCA which is unsupervised, LDA is a supervised learning algorithm. Instead of maximizing the overall variance, LDA maximize the variance between clusters, and minimize the variance inside clusters. LDA is optimal in the sense of classification. However, in image retrieval, the images in database are unlabelled. Therefore, LDA can not be simply applied to learn a semantic subspace for image retrieval.

In this paper, we propose a semi-supervised algorithm to learn a semantic subspace for image retrieval based on LPP [9]. LPP is a linear dimensionality reduction algorithm. Different from PCA which implicitly assumes that the data space is Euclidean, LPP assumes that the data space is a manifold, either linear or nonlinear. LPP aims to preserve the local structure of the image space. Since the neighboring images (data points in high dimensional space) probably are related to the same semantics, LPP can have more discriminating power than PCA. To be specific, an adjacency graph is constructed to model the local structure of the image space. Once the graph structure is obtained, LPP finds a projection which respects the graph structure.

LPP can be performed in either supervised or unsupervised manner. It depends on how the adjacency graph is constructed. During the process of image retrieval based on user's relevance feedback, the accumulated knowledge can be incorporated into the adjacency graph, which could result in new feature representations for

MM'04, October 10–16, 2004, New York, New York, USA.

images in the database so that the system's future retrieval performance can be enhanced.

The rest of this paper is organized as follows: Section 2 gives a brief description of Locality Preserving Projections. Section 3 describes our proposed semi-supervised algorithm for learning a semantic subspace which incorporates the user's relevance feedback. The experimental results are shown in Section 5. Finally, we give concluding remarks and future work in Section 6.

2. LOCALITY PRESERVING PROJECTIONS

2.1 The Problem

The problem of subspace learning for image indexing and representation is the following. Given a set of images $\mathbf{x}_1$, $\mathbf{x}_1$, …, $\mathbf{x}_m$ in $\mathbf{R}^n$, find a lower dimensional representation $\mathbf{y}_i$ of $\mathbf{x}_i$ such that $\|\mathbf{y}_i - \mathbf{y}_j\|$ reflects the semantic relationship between $\mathbf{y}_i$ and $\mathbf{y}_j$. In other word, if $\|\mathbf{y}_i - \mathbf{y}_j\|$ is small, then $\mathbf{x}_i$ is semantically related to $\mathbf{x}_j$. Here, we assume that the images reside on a submanifold embedded in the ambient space $\mathbf{R}^n$.

2.2 The Algorithm

In this section, we give a brief description of Locality Preserving Projections (LPP) [9]. Different from PCA which assumes that the image space is a Euclidean space, LPP assumes that the image space is a manifold. Note that, Euclidean space is actually a linear manifold which is equipped with a *flat* Riemannian metric.

Let $\mathbf{x}_1$, $\mathbf{x}_1$, …, $\mathbf{x}_m$ denote the set of image sample vectors in $\mathbf{R}^n$. We denote by X the image matrix whose column vectors are images. Let $\mathbf{a}$ denote the transformation vector. Thus, the optimal projections preserving locality can be obtained by solving the following minimization problem [9]:

$$\min_{\mathbf{a}} \sum_{ij} \left(\mathbf{a}^T \mathbf{x}_i - \mathbf{a}^T \mathbf{x}_j\right)^2 S_{ij} \tag{1}$$

where S_{ij} evaluate the local structure of the image space. It can be simply defined as follows:

$$S_{ij} = \begin{cases} 1 & \text{if } \mathbf{x}_i \text{ is among the } k \text{ nearest neighbors of } \mathbf{x}_j \\ & \text{or } \mathbf{x}_j \text{ is among the } k \text{ nearest neighbors of } \mathbf{x}_i \\ 0 & \text{otherwise} \end{cases} \tag{2}$$

The objective function with our choice of symmetric weights S_{ij} ($S_{ij} = S_{ji}$) incurs a heavy penalty if neighboring points $\mathbf{x}_i$ and $\mathbf{x}_j$ are mapped far apart. Therefore, minimizing it is an attempt to ensure that if $\mathbf{x}_i$ and $\mathbf{x}_j$ are "close" then y_i (= $\mathbf{a}^T\mathbf{x}_i$) and y_j (= $\mathbf{a}^T\mathbf{x}_j$) are close as well. S_{ij} can be thought of as a similarity measure between objects. Some more sophisticated definition of S can be found in [9]. By simple algebra formulation, the objective function can be reduced to:

$$\begin{aligned} & \frac{1}{2}\sum_{ij} (\mathbf{a}^T \mathbf{x}_i - \mathbf{a}^T \mathbf{x}_j)^2 S_{ij} \\ &= \sum_i \mathbf{a}^T \mathbf{x}_i D_{ii} \mathbf{a}^T \mathbf{x}_i - \sum_{ij} \mathbf{a}^T \mathbf{x}_i S_{ij} \mathbf{a}^T \mathbf{x}_j \\ &= \mathbf{a}^T X(D-S)X^T \mathbf{a} = \mathbf{a}^T XLX^T \mathbf{a} \end{aligned} \tag{3}$$

where $X = [\mathbf{x}_1, \mathbf{x}_2, …, \mathbf{x}_n]$, and D is a diagonal matrix; its entries are column (or row, since S is symmetric) sums of S, $D_{ii} = \sum_j S_{ji}$.. $L = D - S$ is the Laplacian matrix [5]. We impose a constraint as follows:

$$\mathbf{a}^T \mathbf{a} = 1 \tag{4}$$

This constraint is different from the one in the original LPP algorithm [9]. Finally, the minimization problem reduces to finding:

$$\mathop{\arg\min}_{\substack{\mathbf{a} \\ \mathbf{a}^T\mathbf{a}=1}} \ \mathbf{a}^T XLX^T \mathbf{a} \tag{5}$$

The transformation vector $\mathbf{a}$ that minimizes the objective function is given by the minimum eigenvalue solution to the following problem:

$$XLX^T \mathbf{a} = \lambda \mathbf{a} \tag{6}$$

Note that the matrix XLX^T is symmetric and positive semi-definite. Also, the obtained projections $\mathbf{a}$ are actually the optimal linear approximation to the eigenfunctions of the Laplace Beltrami operator on the manifold [2][9]. Therefore, LPP is capable of uncovering the intrinsic manifold structure to some extent.

3. INCREMENTAL SEMI-SUPERVISED SUBSPACE LEARNING FOR IMAGE RETRIEVAL

In section 2, we have described the LPP algorithm which can obtain a locality preserving subspace for image representation. The original LPP is unsupervised. It only makes use of the local geometrical structure to construct the adjacency graph. However, in image retrieval, we can get semantic relationship between images from user's relevance feedback. In this section, we introduce our incremental semi-supervised subspace learning algorithm which incorporates user's relevance feedbacks. Our algorithm is based on a relevance feedback driven query-by-example system.

Let $\mathbf{q}$ denote the query image. During the k^{th} iteration, the user provide a set of positive examples, $A = \{\mathbf{x}_{i_1}, \mathbf{x}_{i_2}, \cdots, \mathbf{x}_{i_m}\}$ and a set of negative examples, $B = \{\mathbf{x}_{j_1}, \mathbf{x}_{j_2}, \cdots, \mathbf{x}_{j_n}\}$. Let G_i denote the adjacency graph corresponding to the k^{th} iteration with the weight matrix S_k and normalized weight matrix W_k. S_0 is defined as in equation (2). Based on the user provided information, we can update S_k and W_k as follows:

1. $S_k = S_{k-1}$
2. $S_k(ij) = \begin{cases} 1 & \text{if } \mathbf{x}_i, \mathbf{x}_j \in A \\ 0 & \text{if } \mathbf{x}_i \in A \text{ and } \mathbf{x}_j \in B, \text{ or } \mathbf{x}_i \in B \text{ and } \mathbf{x}_j \in A \end{cases}$
3. normalization:

   $$W_k(ij) \leftarrow S_k(ij) \big/ \textstyle\sum_j S_k(ij)$$

   This makes the row sums of W_k are 1. The reason for normalization becomes clear in the next subsection.

Correspondingly, we can compute the diagonal matrix D_k and graph Laplacian L_k. based on W_k.

3.1 The Convergence of XL_kX^T

In this section, we discuss the convergence of the matrix XL_kX^T. We first consider the matrix $W_\infty = \lim_{k\to\infty} W_k$. Once we get the matrix W_∞, we can get the matrix $XL_\infty X^T$.

As we described above, the matrix W is updated based on the user's relevance feedback. As more and more relevance feedbacks are provided, ultimately for any two images we can judge if they are semantically related or not. Let n_k denote the number of samples in the k^{th} class. Therefore, W_∞ is defined as follows:

$$W_\infty(ij) = \begin{cases} \frac{1}{n_k} & \text{if } \mathbf{x}_i \text{ and } \mathbf{x}_j \text{ belong to the } k^{th} \text{ category} \\ 0 & \text{otherwise} \end{cases} \tag{7}$$

Note that, W_∞ is normalized. Consequently, we get

$$\begin{aligned} D_\infty(ii) &= \textstyle\sum_j W_\infty = I \\ L_\infty &= D_\infty - W_\infty = I - W_\infty \end{aligned} \tag{8}$$

where I is the identity matrix. In this case, L_∞ does not reflect the local geometrical structure of the image space. Instead, it reflects the discriminant structure of the image database. This observation leads us to consider its connection to Linear Discriminant Analysis (LDA) [6]. Suppose there are l classes. The eigenvector problem of LDA is as follows:

$$\begin{aligned} S_b\mathbf{a} &= \lambda S_w \mathbf{a} \\ S_b &= \sum_{i=1}^{l} n_i \left(\mathbf{m}^{(i)} - \mathbf{m}\right)\left(\mathbf{m}^{(i)} - \mathbf{m}\right)^T \\ S_w &= \sum_{i=1}^{l}\left(\sum_{j=1}^{n_i}\left(\mathbf{x}_j^{(i)} - \mathbf{m}^{(i)}\right)\left(\mathbf{x}_j^{(i)} - \mathbf{m}^{(i)}\right)^T\right) \end{aligned} \tag{9}$$

where $\mathbf{m}$ is the total sample mean vector, n_i is the number of samples in the i^{th} class, $\mathbf{m}^{(i)}$ is the average vector of the i^{th} class, and $\mathbf{x}_j^{(i)}$ is the j^{th} sample in the i^{th} class. We call S_w the *within-class scatter matrix* and S_b the *between-class scatter matrix*.

Suppose the i^{th} class has n_i samples. We define:

$$X_i = [\mathbf{x}_1^{(i)}, \mathbf{x}_2^{(i)}, \cdots, \mathbf{x}_{n_i}^{(i)}] \tag{10}$$

$$L^{(i)} = X_i\left(I_i - \tfrac{1}{n_i}\mathbf{e}_i\mathbf{e}_i^T\right)X_i^T \tag{11}$$

where I_i is a $n_i \times n_i$ identity matrix and $\mathbf{e}_i = \{1, 1, \ldots, 1\}^T$ is a n_i dimensional vector. $L^{(i)}$ can be regarded as the Laplacian matrix [1] of a complete graph defined on X_i with weight matrix $W^{(i)}$:

$$W_{jk}^{(i)} = \tfrac{1}{n_i}, \quad 1 \le j, k \le n_i \tag{12}$$

Or

$$W^{(i)} = \tfrac{1}{n_i}\mathbf{e}_i\mathbf{e}_i^T \tag{13}$$

Note that, $W^{(i)}$ is a $n_i \times n_i$ matrix. Let $D^{(i)}$ be a diagonal matrix whose entries are the column (or row, since $W^{(i)}$ is symmetric) sums of $W^{(i)}$:

$$D_{jj}^{(i)} = \sum_{l=1}^{n_i} W_{jl}^{(i)} \tag{14}$$

Thus,

$$L^{(i)} = D^{(i)} - W^{(i)} \tag{15}$$

Combining all the l classes, we get:

$$\sum_{i=1}^{l} X_i L^{(i)} X_i^T = XL_\infty X^T \tag{16}$$

Now, we can compute the matrix S_w as follows:

$$\begin{aligned} S_w &= \sum_{i=1}^{l} n_i E\left[\left(\mathbf{x}^{(i)} - \mathbf{m}^{(i)}\right)\left(\mathbf{x}^{(i)} - \mathbf{m}^{(i)}\right)^T\right] \\ &= \sum_{i=1}^{l}\left(\sum_{j=1}^{n_i}\left(\mathbf{x}_j^{(i)} - \mathbf{m}^{(i)}\right)\left(\mathbf{x}_j^{(i)} - \mathbf{m}^{(i)}\right)^T\right) \\ &= \sum_{i=1}^{l}\left(\sum_{j=1}^{n_i}\left(\mathbf{x}_j^{(i)}(\mathbf{x}_j^{(i)})^T - \mathbf{m}^{(i)}(\mathbf{x}_j^{(i)})^T - \mathbf{x}_j^{(i)}(\mathbf{m}^{(i)})^T + \mathbf{m}^{(i)}(\mathbf{m}^{(i)})^T\right)\right) \\ &= \sum_{i=1}^{l}\left(\sum_{j=1}^{n_i}\mathbf{x}_j^{(i)}(\mathbf{x}_j^{(i)})^T - n_i\mathbf{m}^{(i)}(\mathbf{m}^{(i)})^T\right) \\ &= \sum_{i=1}^{l}\left(X_iX_i^T - \tfrac{1}{n_i}\left(\mathbf{x}_1^{(i)} + \cdots + \mathbf{x}_{n_i}^{(i)}\right)\left(\mathbf{x}_1^{(i)} + \cdots + \mathbf{x}_{n_i}^{(i)}\right)^T\right) \\ &= \sum_{i=1}^{l}\left(X_iX_i^T - \tfrac{1}{n_i}X_i\left(\mathbf{e}_i\mathbf{e}_i^T\right)X_i^T\right) \\ &= \sum_{i=1}^{l} X_i L^{(i)} X_i^T \\ &= XL_\infty X^T \end{aligned}$$

This shows that, as more and more user's relevance feedbacks are accumulated, the matrix $XL_\infty X^T$ ultimately converges to the within-class scatter matrix S_w. Therefore, by computing the eigenvectors of the matrix XL_kX^T (k=1,2,…), we ultimately obtain a semantic subspace in which the different classes can be best separated. Note that, in this process, the unsupervised subspace learning becomes supervised subspace learning.

3.2 The First Eigenvector of XL_kX^T

We begin with a lemma as follows:

Lemma 1: Let $\{a_n\}$ be a sequence of real numbers. If $\lim_{n\to\infty} a_n = a$, then $\lim_{n\to\infty} \frac{1}{n}\sum_{i=1}^{n} a_i = a$.

In fact, a_n can be elements in any metric space, besides the real space. Thus, we have the following lemma:

Lemma 2: Let $\{M_n\}$ be a sequence of real matrices. If $\lim_{n\to\infty} M_n = M$, then $\lim_{n\to\infty} \frac{1}{n}\sum_{i=1}^{n} M_i = M$.

From section 3.1, we know that $\lim_{k\to\infty} XL_k X^T = XL_\infty X^T$. Thus, we have

$$\lim_{n\to\infty} \frac{1}{n}\sum_{k=1}^{n} XL_k X^T = XL_\infty X^T \tag{17}$$

Let $\mathbf{u}_k$ be the eigenvector of XL_kX^T associated with the eigenvalue λ_k. That is

$$XL_k X^T \mathbf{u}_k = \lambda_k \mathbf{u}_k \tag{18}$$

Let $\mathbf{u} = \lim_{k\to\infty} \mathbf{u}_k$ be the eigenvector of $XL_\infty X^T$ with the eigenvalue λ. That is,

$$XL_\infty X^T \mathbf{u} = \lambda \mathbf{u} \tag{19}$$

We define

$$\mathbf{v}_k = XL_k X^T \mathbf{u}_k \tag{20}$$

Thus, $\lim_{k\to\infty} \mathbf{v}_k = XLX^T\mathbf{u} = \lambda\mathbf{u}$. Now, we define

$$\mathbf{w}_n = \frac{1}{n}\sum_{k=1}^{n} \mathbf{v}_k \tag{21}$$

By lemma 2, we have

$$\lim_{n\to\infty} \mathbf{w}_n = \lambda\mathbf{u} \tag{22}$$

Consequently,

$$\lim_{n\to\infty} \|\mathbf{w}_n\| = \lambda \tag{22}$$

$$\lim_{n\to\infty} \frac{\mathbf{w}_n}{\|\mathbf{w}_n\|} = \mathbf{u} \tag{23}$$

By simple algebra formulation, we have

$$\begin{aligned}
\mathbf{w}_n &= \frac{1}{n}\sum_{k=1}^{n} \mathbf{v}_k \\
&= \frac{1}{n}\sum_{k=1}^{n} XL_k X^T \mathbf{u}_k \\
&= \frac{n-1}{n}\left(\frac{1}{n-1}\sum_{k=1}^{n-1} XL_k X^T \mathbf{u}_k\right) + \frac{1}{n} XL_n X^T \mathbf{u}_n \\
&= \frac{\text{n-1}}{\text{n}} \mathbf{w}_{n-1} + \frac{1}{n} XL_n X^T \mathbf{u}_n
\end{aligned} \tag{24}$$

Note that, $\mathbf{u}_n$ is the eigenvector of XL_nX^T. Since $\lim_{n\to\infty} \mathbf{u}_n = \lim_{n\to\infty} \mathbf{w}_{n-1}/\|\mathbf{w}_{n-1}\| = \mathbf{u}$, $\mathbf{u}_n$ can be estimated by $\mathbf{w}_{n-1} / \|\mathbf{w}_{n-1}\|$. Thus, we get

$$\mathbf{w}_n \approx \frac{n-1}{n}\mathbf{w}_{n-1} + \frac{1}{n} XL_n X^T \frac{\mathbf{w}_{n-1}}{\|\mathbf{w}_{n-1}\|} \tag{25}$$

TABLE 1

Image Features Used in Our System

Color-1	Color histogram in HSV space with quantization 256
Color-2	First and second moments in Lab space
Color-3	Color coherence vector in LUV space with quantization 64
Texture-1	Tamura coarseness histogram
Texture-2	Tamura directionary
Texture-3	Pyramid wavelet texture feature

As we described previously, at each iteration during the retrieval process, the matrix L_n can be updated based on user's relevance feedbacks. Let $C_n = L_n - L_{n-1}$. C_n is a very sparse matrix since typically only a small number of feedbacks can be obtained from the user. Thus, the matrix XL_nX^T can be computed as follows:

$$XL_n X^T = XL_{n-1} X^T + \sum_{ij} C_{n,ij} \mathbf{x}_i \mathbf{x}_j^T \tag{26}$$

Clearly, the matrix XL_nX^T can be updated very efficiently. The main computations during the eigenvector update process are due to equation (25) and (26). The computational complexity is very small. The first eigenvector and eigenvalue are $\mathbf{w}_n/\|\mathbf{w}_{n-1}\|$ and $\|\mathbf{w}_{n-1}\|$, respectively.

3.3 Other Eigenvectors of XL_kX^T

In section 3.2, we described how to compute the first eigenvectors. Note that the matrix XL_kX^T is symmetric, so the eigenvectors are orthogonal to each other. Thus, the other eigenvectors can be computed in the complementary space which is orthogonal to the subspace spanned by the first eigenvector. Let $\mathbf{a}^k$ denote the estimated k^{th} eigenvector associated with the estimated k^{th} eigenvalue. For each data point, we first subtract its projection on the first eigenvector $\mathbf{a}^1$, i.e.

$$\mathbf{x} \leftarrow \mathbf{x} - \mathbf{a}^1(\mathbf{a}^1)^T\mathbf{x} \tag{27}$$

Using the same method described in Section 3.2, we can compute the second eigenvector. Because the eigenvector is a linear combination of the data points (after transformation) and the data points (after transformation) are orthogonal to the first eigenvector, the second eigenvector is automatically orthogonal to the first eigenvector. By repeating this process, we can get the third, forth eigenvectors, and so on. Thus, for any image $\mathbf{x}$, we can get its map in the k–dimensional semantic subspace as follows:

$$\begin{aligned}
\mathbf{y} &= A^T\mathbf{x} \\
A &= \left[\mathbf{a}^1, \mathbf{a}^2, \cdots, \mathbf{a}^k\right]
\end{aligned}$$

3.4 Image Retrieval in Semantic Subspace

The retrieval process can be viewed as an on-line learning process in which the image retrieval system acts as a learner and the user

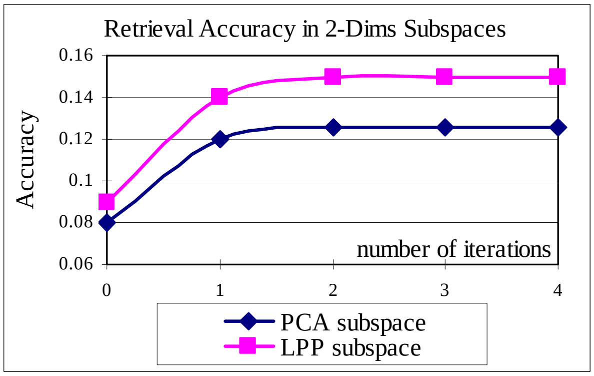

Figure 1. The image retrieval performances in PCA subspace and LPP subspace with 2 dimensions.

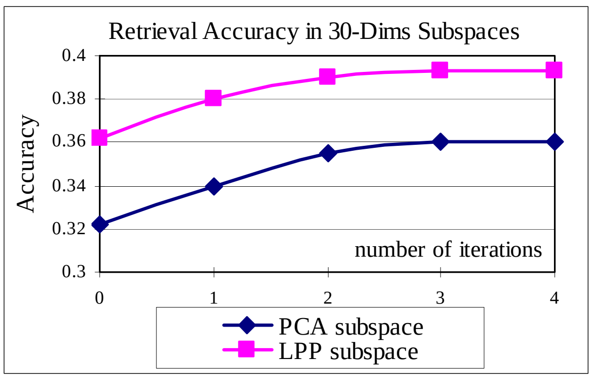

Figure 2. The image retrieval performances in PCA subspace and LPP subspace with 30 dimensions.

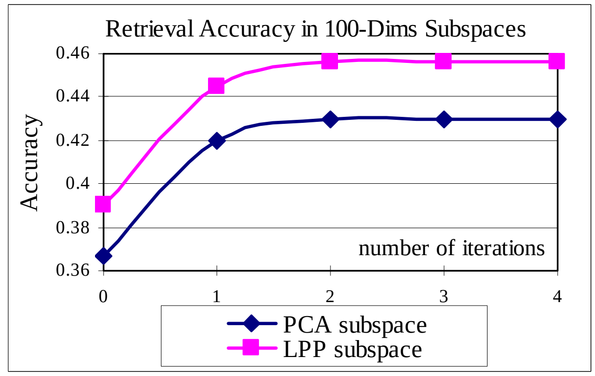

Figure 3. The image retrieval performances in PCA subspace and LPP subspace with 100 dimensions.

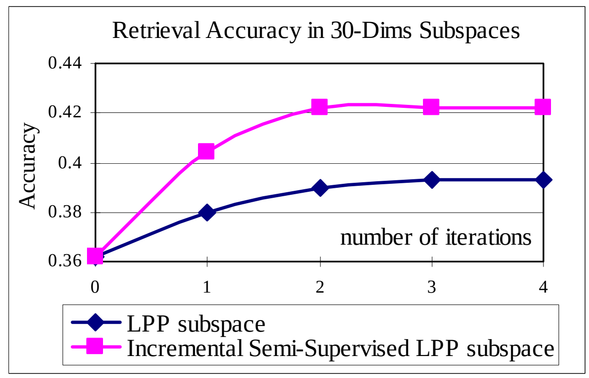

Figure 4. The image retrieval performances in LPP subspace and incremental semi-supervised LPP subspace with 30 dimensions.

acts as a teacher. They typical retrieval process is outlined as follows:

1. The user submits a query image example to the system. The system ranks the images in the database according to some pre-defined distance metric and presents to the user the top ranked images.
2. The user provides his relevance feedbacks to the system by labeling images as "relevant" or "irrelevant".
3. The system uses user's provided information to re-rank the images in the database and returns to the user the top images. Go to step 2 until the user is satisfied.

Traditionally, the user's relevance feedbacks are used to update the query vector [12] or adjust the weighting of different dimensions [11]. The user's relevance feedbacks are not used to obtain better representations of the images. Therefore, the future retrieval can not benefit from the current feedbacks provided by the user.

Using our proposed incremental semi-supervised subspace learning algorithm based on LPP, we can periodically update the vector representations of the images. The new representation respects user's judgments on the images and can better describe the intrinsic semantic relationships between images.

4. EXPERIMENTAL RESULTS

We performed several experiments to evaluate the effectiveness of the proposed algorithm on a large image database. The image database we used consists of 3,000 images of 30 semantic categories, from the Corel data set. Each category contains 100 images. It is a large and heterogeneous image set. For the sake of simplicity of evaluation, all the query images are from the database. However, the query images can also be outside the database. A retrieved image is considered correct if it belongs to the same category as the query image. Three types of color features and three types of texture features are used in our system; they are listed in Table 1. Each image is represented as a 435-dimensional vector.

We designed an automatic feedback scheme to model the retrieval process. At each iteration, the system selects the first four correct images (according to the ground truth) as relevant examples (relevant examples from the previous iterations are excluded from the selection). These automatically generated feedbacks are used to modify the similarity matrix and learn a semantic subspace which respects human perceptions. To evaluate the performance of our algorithms, we define the retrieval accuracy as follows:

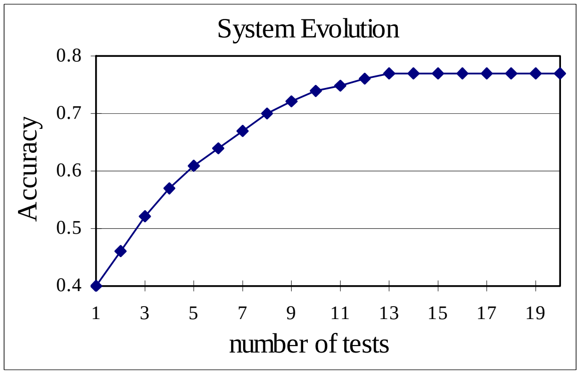

Figure 5. The retrieval accuracy of the system improves as the user's feedbacks are accumulated.

$$Accuracy = \frac{relevant\ images\ retrieved\ in\ top\ N\ returns}{N}$$

Since our algorithm is fundamentally based on Locality Preserving Projections, we begin with the comparisons between LPP and PCA.

4.1 Comparisons between PCA and LPP

In this subsection, we compared the retrieval performance in PCA subspace and LPP subspace. The original 435-dimensional image space was reduced to lower-dimensional subspaces with different dimensionalities by using PCA and LPP. For each semantic class, we randomly selected 10 query images. Hence there are 300 query images in total. The query images are also projected into the lower dimensional subspaces. Many retrieval algorithms have been proposed in recent years [16][17][19]. In this paper, we used the Rocchio's formula described in [11][13] for its simplicity. The average retrieval performances were computed. Figure 1-3 show the fraction of relevant images among the top N = 15 images returned by the system, as a function of the number of rounds of user feedback. As can be seen, the retrieval performance in LPP subspace is better than that in PCA subspace.

4.2 Retrieval in the Semantic Subspace Obtained by Our Semi-Supervised Subspace Learning Algorithm

4.2.1 Image Retrieval in LPP subspace and Semi-Supervised LPP subspace

As we have demonstrated, LPP is superior to PCA as to learning a semantic subspace. In this section, we evaluate our proposed incremental semi-supervised LPP algorithm. The image database and query images used in this experiment are the same as those in Section 4.1. At each iteration, the user's relevance feedbacks are used to update the eigenvectors which span the semantic subspace. The images in the database and the query images are projected into the semantic subspace using these eigenvectors. The retrieval was then performed in the semantic subspace. Note that, at the beginning when there is no user's relevance feedback, our semi-supervised LPP obtained the same subspace as the ordinary LPP.

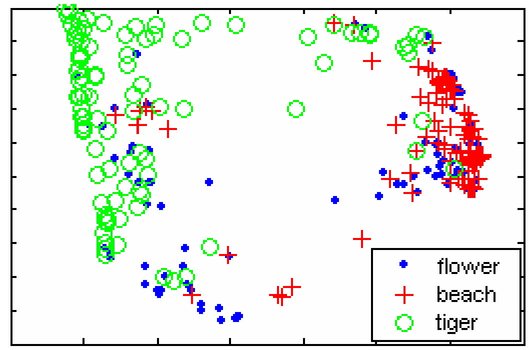

Figure 6. 2-D visualization of three image classes using the LPP algorithm.

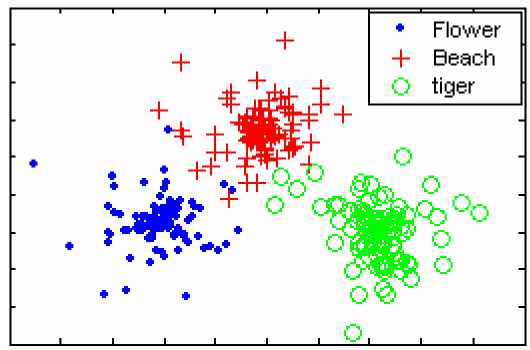

Figure 7. 2-D visualization of three image classes using incremental semi-supervised LPP algorithm.

Figure 4 shows the retrieval performance in LPP subspace and our incremental semi-supervised LPP subspace with 30 dimensions. As can be seen, our algorithm consistently outperforms LPP during the retrieval process.

As we described earlier, the matrix XL_kX^T will ultimately converge to the *within-class scatter matrix* S_w. We did an experiment to evaluate how our system evolves. At each test, we randomly selected 30 query images from the database, one per class. For each query, we accumulated the user's feedbacks and computed the retrieval accuracy using Rocchio's formula [11][13]. The 30 retrieval performances were averaged as the retrieval accuracy for this test. The user's feedbacks collected during this test were used to update the eigenvectors using our proposed algorithm. In this experiment, 30 eigenvectors are used. In other word, the dimensionality of the semantic subspace is 30. Figure 5 shows that the system performance can be enhanced as more and more user's feedbacks were collected. Moreover, the system performance tended to converge after 12 tests. The final semantic subspace we obtained is called *optimal subspace*.

4.2.2 2-D Visualization

Theoretical analysis and experimental results show that our incremental semi-supervised subspace learning algorithm will converge. It would be worthwhile to compare the 2-D visualization of the image database using the ordinary LPP algorithm and our proposed algorithm. We randomly select three semantic classes, i.e. flower, beach, and tiger. There are 300 images in total, 100 images for each class. These 300 images were projected into a 2-dimensional subspace obtained by LPP. Figure 6 shows the result. Similarly, these 300 images were also projected into the 2-dimensional *optimal subspace* obtained in previous section using our proposed algorithm. The result is shown in Figure 7. As can be seen, these three image classes can be best separated in the subspace obtained by our incremental semi-supervised subspace learning algorithm.

5. CONCLUSIONS

In this paper, we described a semi-supervised subspace learning algorithm that makes use of relevance feedbacks to enhance the performance of an image retrieval system from both short- and long-term perspectives. Our algorithm is based on Locality Preserving Projections [9] which is a recently proposed linear dimensionality reduction algorithm. Different from Principal Component Analysis which respects the global structure, our algorithm respects the local structure and human perceptions which might be more important for real world image retrieval applications. Theoretical analysis shows that our incremental subspace learning algorithm has intrinsic connection to Linear Discriminant Analysis. Our algorithm will ultimately obtain a semantic subspace in which the different image classes can be best separated. Several experiments on a large collection of images have shown the effectiveness and efficiency of our proposed algorithm.

There are still several questions that remain to be investigated in the future. First, the feedback provided by the real world users often contains inaccurate information. Hence the noise could be introduced into the system when the semantic subspace is constructed. It is unclear how the system works under a noisy environment, even though the original LPP algorithm seems to be insensitive to noise [9]. It may be desirable to conduct filtering to remove unreliable feedback before using it for training the system. Finally, and the most importantly and challengingly, it remains unclear to what extent the geometrical structure of the image space (characterized by color, texture, etc.) can be consistent to the human perception. Specifically, it is unclear how to learn a metric tensor defined on the image manifold which can induce a Riemannian metric consistent to human perception. Some preliminary results have been reported in [8].

6. REFERENCES

[1] P. N. Belhumeur, J. P. Hespanha and D. J. Kriegman, "Eigenfaces vs. Fisherfaces: Recognition using class specific linear projection," *IEEE Trans. On Pattern Analysis and Machine Intelligence*, vol. 19, pp. 711-720, 1997.

[2] M. Belkin and P. Niyogi, "Laplacian Eigenmaps and Spectral Techniques for Embedding and Clustering," *Advances in Neural Information Processing Systesms*, vol. 14, 2002.

[3] E. Chang, K. Goh, G. Sychay and G. Wu, "CBSA: Content-based soft annotation for multimodal image retrieval using bayes point machines," *IEEE Trans. On Circuits and Systems for Video Technology*, vol. 13, no. 1, pp. 26-38, January 2003.

[4] Yixin Chen, James Z. Wang and Robert Krovetz, "Content-based image retrieval by clustering", *Proc. 5th International Workshop on Multimedia Information Retrieval*, pp. 193-200, Berkeley, CA, 2003.

[5] Fang Chung, Spectral Graph Theory, *Regional Conferences Series in Mathematics*, number 92, 1997.

[6] Richard O. Duda, Peter E. Hart, and David G. Stork, *Pattern Classification*, John Wiley and Sons, New York, 2001.

[7] S. Gordon, H. Greenspan and J. Goldberger, "Applying the information bottleneck principle to unsupervised clustering of discrete and continuous image representations", *IEEE International Conference on Computer Vision*, Nice, France, 2003.

[8] Xiaofei He, Wei-Ying Ma, and Hong-Jiang Zhang, "Learning an Image Manifold for Retrieval", *ACM Conference on Multimedia*, New York, 2004.

[9] Xiaofei He and Partha Niyogi, "Locality Preserving Projections", *Advances in Neural Information Processing Systems* 16, Vancouver, Canada, 2003.

[10] Xiaofei He, Shuicheng Yan, Yuxiao Hu and Hong-Jiang Zhang, "Learning a locality preserving subspace for visual recognition", *IEEE International Conference on Computer Vision*, Nice, France, 2003.

[11] Y. Ishikawa, R. Subranmanya, and C. Faloutsos, "MindReader: Query Databases Through Multiple Examples," in *Proc. Of the 24th Int. Conf. on Very Large Databases*, pp. 218-227, New York, 1998.

[12] W. Y. Ma and B. S. Manjunath, "Netra: a toolbox for navigating large image databases", *Multimedia Systems*, vol. 7, no. 3, Springer-Verlag, May 1999.

[13] Y. Rui, T. S. Huang and S. Mehrotra, "Content-Based Image Retrieval with Relevance Feedback in MARS," in *Proc. IEEE International Conference on Image Processing*, pp. 815-818, Santa Barbara, CA, Oct. 1997.

[14] Z. Su, S. Li and H.-J. Zhang, "Extraction of Feature Subspace for Content-Based Retrieval Using Relevance Feedback", in *Proc. ACM International Conference on Multimedia*, Ottawa, Canada, 2001.

[15] D. L. Swets and J. Weng, "Using Discriminant eigenfeatures for image retrieval," *IEEE Trans. On Pattern Analysis and Machine Intelligence*, vol. 18, no. 8, pp. 831-836, 1996.

[16] K. Tieu and P. Viola, "Boosting Image Retrieval," in *Proc. IEEE International Conference on Computer Vision and Pattern Recognition*, Hilton Head Island, SC, June, 2000.

[17] S. Tong and E. Chang, "Support Vector Machine Active Learning for Image Retrieval," in *Proc. ACM Multimedia 2001*, Ottawa, Canada, Sept. 2001.

[18] M. A. Turk and A. P. Pentland, "Face recognition using eigenfaces", *IEEE Conference on Computer Vision and Pattern Recognition*, pages 586-591, 1991.

[19] X. S. Zhou and T. S. Huang, "Comparing Discriminating Transformations and SVM for Learning during Multimedia Retrieval," in *Proc. ACM Multimedia 2001*, Ottawa, Canada, 2001.