Optimal Factor Analysis and Applications to Content-Based Image Retrieval

Yuhua Zhu[1], Washington Mio[2], and Xiuwen Liu[1]

[1] Dept. of Computer Science, Florida State University, Tallahassee FL 32306, U.S.A.
[2] Dept. of Mathematics, Florida State University, Tallahassee FL 32306, U.S.A.

Abstract. We formulate and develop computational strategies for *Optimal Factor Analysis* (OFA), a linear dimension reduction technique designed to learn low-dimensional representations that optimize discrimination based on the nearest-neighbor classifier. The methods are applied to content-based image categorization and retrieval using a representation of images by histograms of their spectral components. Various experiments are carried out and the results are compared to those that have been previously reported for some other image retrieval systems.

Keywords: Linear dimension reduction, image classification, content-based image retrieval, optimal factor analysis.

1 Introduction

We develop *Optimal Factor Analysis* (OFA), a linear dimension-reduction technique that optimizes the discriminative ability of the nearest-neighbor classifier for a given data classification problem. We apply the technique to content-based categorization and retrieval of images using a representation based on the statistics of their spectral components. This investigation is motivated by the need to develop intelligent and scalable systems capable of indexing and retrieving images from large and complex image libraries in an automated manner. Classical approaches based on "expert" annotations are not viable for large data sets.

For the image categorization problem, we shall assume that a training database of labeled images representing various different classes of objects is available and the goal is to learn optimal low-dimensional features or "signatures" to assign a query image to the correct class. In content-based image retrieval, one of the objectives is to find the top ℓ matches in a database to a query image, where the number ℓ is prescribed by the user. In the proposed approach, categorization and retrieval are closed related. We use a categorization algorithm to organize an entire database according to features learned from a training set. Given a query image I, we first rank the classes using the nearest neighbor classifier applied to the learned low-dimensional features and then retrieve images sequentially starting from the top ranked class.

The problem of classifying images in a database into semantic categories arises in many different levels of generality. For example, the problem can be as broad as separating images that depict an indoor or outdoor scene, or it may involve much more specific categorization into classes such as cars, people, and flowers. As the breadth of the semantic categories may vary considerably, the development of general strategies poses

J. Braz et al. (Eds.): VISIGRAPP 2007, CCIS 21, pp. 164–176, 2008.

significant challenges. This motivated us to approach the problem in two stages. First, we extract "stable" features that are able to capture information about the structure and semantic content of an image. Subsequently, we employ learning techniques to identify the factors that have the highest discriminating power for a particular classification problem.

The histogram of an image is useful, however, it tends to have limited discriminating power because it contains little information about the finer structure of an image. To remedy the situation, we use histograms of multiple spectral components, as they retain a significant amount of information about texture patterns and edges. The statistics of spectral components have been studied in the past primarily in the context of texture analysis and synthesis. In [11], it is demonstrated that marginal distributions of spectral components suffice to characterize homogeneous textures; other studies include [5] and [9]. To provide some preliminary evidence of the suitability of spectral histogram (SH) features, in Section 3, we report the results of a retrieval experiment on a database of 1,000 images representing 10 different semantic categories. The relevance of an image is determined by the nearest-neighbor criterion applied to a number of SH-features combined into a single feature vector. Even without a learning component, we already observe a performance comparable to those exhibited by some existing retrieval systems.

Optimal Factor Analysis will be employed with a twofold purpose: (a) to identify and split off the most discriminating factors of the SH-features; (b) to lower the dimension of the representation to reduce complexity and improve computational efficiency. A preliminary form of OFA was introduced in [3] as Splitting Factor Analysis. Given a (small) positive integer k, the goal of OFA is to find an "optimal" k-dimensional linear reduction of the original image features for a particular categorization or indexing problem. Image categorization and retrieval will be based on the nearest neighbor classifier applied to the reduced features, as explained in more detail below. We employ OFA in the context of SH-features, but it will be presented in a more general feature learning framework.

Image retrieval strategies employing a variety of methods have been investigated in [8], [1], [6], [7], [10], [2]. Further references can be found in these papers. Some of these proposals employ a relevance feedback mechanism in an attempt to progressively improve the quality of retrieval. Although not discussed in this paper, a feedback component can be incorporated to the proposed strategy by gradually adding to the training set images for which the quality of retrieval was low.

The paper is organized as follows. In Section 2, we describe the histogram features that will be used to characterize image content. Preliminary retrieval experiments using these features are described in Section 3. Section 4 contains a discussion of Optimal Factor Analysis, and Sections 6 and 7 are devoted to applications of the machine learning methodology to image categorization and retrieval. Section 8 closes the discussion with a summary and a few remarks on refinements of the proposed methods.

2 Spectral Histogram Features

Let I be a gray-scale image and F a convolution filter. The spectral component I_F of I associated with F is the image I_F obtained through the convolution of I and F, which is given at pixel location p by

$$I_F(p) = F * I(p) = \sum_q F(q)\, I(p-q), \tag{1}$$

where the summation is taken over the pixels of F. For a color image, we apply the filter to its R,G,B channels. For a given set of bins, which will be assumed fixed throughout the paper, we let $h(I, F)$ denote the corresponding histogram of I_F. We refer to $h(I, F)$ as the spectral histogram (SH) feature of the image I associated with the filter F. If the number of bins is b, the SH-feature $h(I, F)$ can be viewed as a vector in $\mathbb{R}^b$. Figure 1 illustrates the process of obtaining SH-features. Frames (a) and (b) show a color image and its red channel response to a Laplacian filter, respectively. The last panel shows the 11-bin histogram of the filtered image.

(a) (b) (c)

Fig. 1. (a) An image; (b) the red-channel response to a Laplacian filter; (c) the associated 11-bin histogram

If $\mathcal{F} = \{F_1, \ldots, F_r\}$ is a bank of filters, the SH-features associated with the family $\mathcal{F}$ is the collection $h(I, F_i)$, $1 \leqslant i \leqslant r$, combined into the single m-dimensional vector

$$h(I, \mathcal{F}) = (h(I, F_1), \ldots, h(I, F_r)), \tag{2}$$

where $m = rb$. For a color image, $m = 3rb$. Banks of filters used in this paper consist of Gabor filters of different widths and orientations, gradient filters, and Laplacian of Gaussians.

3 SH-Features for Image Retrieval

To offer some preliminary evidence that image representation by SH-features may be attractive for retrieval, we perform a simple retrieval experiment using the Euclidean distance between histograms. Even without a learning component, the results are already comparable to those obtained with some existing systems. To compare the results objectively with those reported in [8] for SIMPLIcity and color histograms, we use the same subset of the Corel data set consisting of 10 semantic categories, each with 100 images. We refer to this data set as Corel-1000. The categories are as follows: (1) African people and villages; (2) beach scenes; (3) buildings; (4) buses; (5) dinosaurs; (6) elephants; (7) flowers; (8) horses; (9) mountains and glaciers; (10) food. Three samples from each category are shown in Figure 2. The examples are emblematic of the large variations observed even within a semantic category.

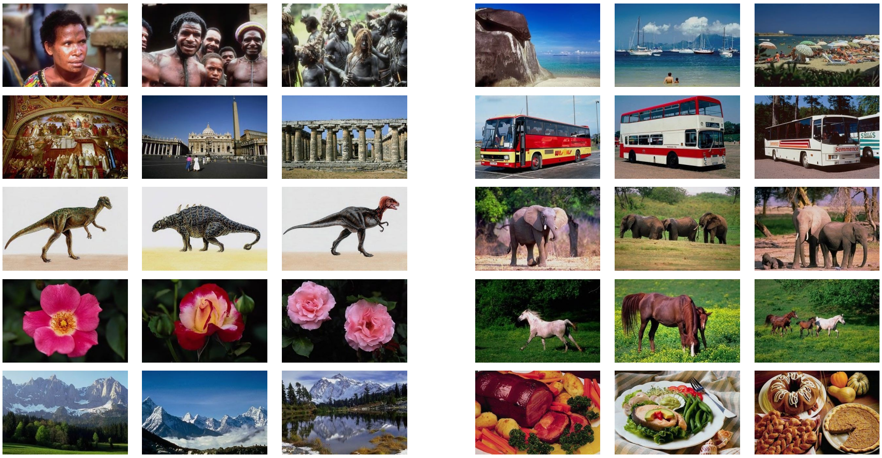

Fig. 2. Samples from Corel-1000: three images from 10 classes, each consisting of 100 images

We utilize a bank of 5 filters and apply each filter to the R, G, and B channels of the images to obtain a total of 15 histograms per image. Each histogram consists of 11 bins so that the SH-feature vector $h(I, \mathcal{F})$ has dimension 165. For a query image I from the database, we calculate the Euclidean distances between $h(I, \mathcal{F})$ and $h(J, \mathcal{F})$, for every J in the database, and rank the images according to increasing distances. For comparison purposes, as in [8], we calculate the weighted precision and the average rank, which are defined as follows. The retrieval precision for the top ℓ returns, is n_ℓ/ℓ, where n_ℓ is the number of correct matches. The weighted precision for a query image I is

$$p(I) = \frac{1}{100} \sum_{\ell=1}^{100} \frac{n_\ell}{\ell}. \tag{3}$$

For a query image I, rank order all 1,000 images in the database, as described above. The average rank $r(I)$ is the mean value of the ranks of all images that belong to the same class as I. Figures 3(a) and 3(b) show the mean values

$$\bar{p}_i = \frac{1}{100} \sum_{I \in C_i} p(I) \;\; \text{and} \;\; \bar{r}_i = \frac{1}{100} \sum_{I \in C_i} r(I), \tag{4}$$

of the weighted precision and average rank within each class C_i, $1 \leqslant i \leqslant 10$. High mean precision and low mean rank reflect high retrieval performance. The results obtained with SH-features are compared to those reported in [8] for SIMPLIcity and for color histograms with the earth mover's distance (EMD) investigated in [6]. In Figure 3, color histograms 1 and 2 refer to EMD applied to histograms with a different number of bins.

4 Optimal Factor Analysis

We develop Optimal Factor Analysis (OFA), a linear feature learning technique whose goal is to find a linear mapping that reduces the dimension of data representation while

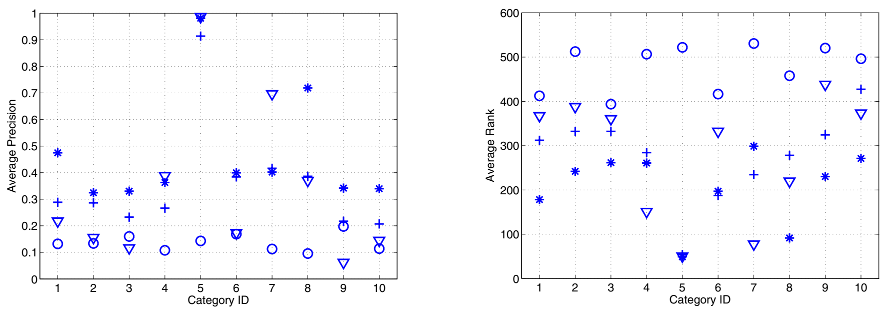

Fig. 3. (a) Plots of $\bar{p}_i$ and $\bar{r}_i$, $1 \leqslant i \leqslant 10$. The methods are labeled as: (∇) spectral histogram; ($*$) SIMPLIcity; ($\circ$) color histogram 1; (+) color histogram 2.

optimizing the discriminative ability of the nearest neighbor classifier, as measured by its performance on training data. We assume that the training set is formed by feature vectors in Euclidean space $\mathbb{R}^m$ and consists of labeled representatives from P different classes of objects. For each integer c, $1 \leq c \leq P$, we denote the vectors in class c by $x_{c,1}, \ldots, x_{c,t_c}$.

If $A\colon \mathbb{R}^m \to \mathbb{R}^k$ is a linear transformation, the quantity

$$\rho(x_{c,i}; A) = \frac{\min_{c \neq b,j} \|Ax_{c,i} - Ax_{b,j}\|^p}{\min_{j \neq i} \|Ax_{c,i} - Ax_{c,j}\|^p + \epsilon} \tag{5}$$

provides a measurement of how well the nearest-neighbor classifier applied to the reduced data identifies the element represented by $x_{c,i}$ as belonging to class c. Here, $\epsilon > 0$ is a small number used to prevent vanishing denominators and $p > 0$ is an exponent that can be adjusted to regularize ρ in different ways. In this paper, we set $p = 2$. A large value of $\rho(x_{c,i}; A)$ indicates that, after the transformation A is applied, $x_{c,i}$ lies much closer to its own class than to other classes. A value $\rho(x_{c,i}; A) \approx 1$ indicates a transition between correct and incorrect decisions by the nearest neighbor classifier. The function ρ is similar to that used in the development of Optimal Component Analysis (OCA) [4]. Note that expression (5) can be easily modified to reflect the performance of the K-nearest neighbor classifier.

The idea is to choose a transformation A that maximizes the average value of $\rho(x_{c,i}; A)$ over the training set. To control bias with respect to particular classes, we scale $\rho(x_{c,i}; A)$ with a sigmoid of the form

$$\sigma(x) = \frac{1}{1 + e^{-\beta x}} \tag{6}$$

before taking the average. We identify linear maps $A\colon \mathbb{R}^m \to \mathbb{R}^k$ with $k \times m$ matrices, in the usual way, and define a performance function $F\colon \mathbb{R}^{k \times m} \to \mathbb{R}$ by

$$F(A) = \frac{1}{P} \sum_{c=1}^{P} \left(\frac{1}{t_c} \sum_{i=1}^{t_c} \sigma\left(\rho(x_{c,i}; A) - 1\right) \right). \tag{7}$$

Scaling an entire dataset does not change decisions based on the nearest-neighbor classifier. This is reflected in the fact that F is (nearly) scale invariant; that is, $F(A) \approx F(rA)$, for $r > 0$. Equality does not hold exactly because $\epsilon > 0$, but in practice, ϵ is negligible. Thus, we fix the scale and optimize F over matrices A of unit Frobenius norm. Let

$$\mathbb{S} = \left\{ A \in \mathbb{R}^{k \times m} : \; \|A\|^2 = \mathrm{tr}\,(AA^T) = 1 \right\} \tag{8}$$

be the unit sphere in $\mathbb{R}^{k \times m}$. The goal of OFA is to maximize the performance function F over $\mathbb{S}$; that is, to find

$$\hat{A} = \operatorname*{argmax}_{A \in \mathbb{S}} F(A). \tag{9}$$

Due to the existence of multiple local maxima of F, the numerical estimation of $\hat{A}$ is carried out with a stochastic gradient search. We remark that this optimization problem is simpler than the corresponding problem for OCA because the OFA search is performed over a sphere instead of a Grassmann manifold. While OCA only considers dimension reduction via orthogonal projections to k-dimensional subspaces of $\mathbb{R}^m$, OFA allows more general linear mappings. Thus, OFA may produce k-dimensional features more effective for classification with significant computational gains.

5 Estimating $\hat{A}$

Our computational approach to the estimation of $\hat{A}$ is based on simulated annealing and is similar to the strategy adopted by Liu *et al.* for OCA [4]. We begin with the details of a deterministic gradient search for maxima of F over the unit sphere $\mathbb{S}$ and then outline the routine changes needed to carry out a stochastic search using simulated annealing with a Metropolis-Hastings acceptance-rejection criterion.

5.1 Deterministic Gradient

Given $A \in \mathbb{S}$, to estimate the gradient vector field $\nabla_{\mathbb{S}} F$ on $\mathbb{S}$ associated with the performance function F, we first calculate $\nabla F(A)$, the gradient of F viewed as a function on $\mathbb{R}^{k \times m}$. Since F is nearly scale invariant,

$$\nabla F(A) \approx \nabla_{\mathbb{S}} F(A), \tag{10}$$

as the component of $\nabla F(A)$ normal to the sphere is almost negligible. The numerical estimation of the left-hand side of (10) only involves standard procedures. For $1 \leqslant i \leqslant k$, $1 \leqslant j \leqslant m$, let E_{ij} be the $k \times m$ matrix whose (i, j) entry is 1 and all others vanish. The partial derivative of F in the direction E_{ij} is estimated as

$$\partial_{ij} F(A) \approx \frac{F(A + \delta E_{ij}) - F(A)}{\delta},$$

with $\delta > 0$ small. Then, $\nabla F(A)$ can be approximated by

$$\bar{\nabla} F(A) = \sum_{i,j} \partial_{ij} F(A) E_{ij}. \tag{11}$$

The vector $\bar{\nabla} F(A)$ is nearly tangential to $\mathbb{S}$ at A. We enforce full tangentiality and obtain a more accurate estimation of $\nabla_{\mathbb{S}} F(A)$ by subtracting the component normal to the sphere $\mathbb{S}$, as follows. For any $A \in \mathbb{S}$, the outer unit normal vector to $\mathbb{S}$ at A in $\mathbb{R}^{k\times m}$ is A itself. Thus, we adopt the estimate

$$\nabla_{\mathbb{S}} F(A) \approx \bar{\nabla} F(A) - \left\langle \bar{\nabla} F(A), A \right\rangle A. \tag{12}$$

A deterministic gradient search for (local) maxima of F on $\mathbb{S}$ can be carried out with the following algorithm.

Algorithm: Deterministic Gradient Search

1. Choose a threshold value $\epsilon > 0$ and a step size $\delta > 0$.
2. Initialize the search with some $A \in \mathbb{S}$.
3. Calculate $\nabla_{\mathbb{S}} F(A)$ using Eqns. 11 and 12.
4. If $\|\nabla_{\mathbb{S}} F(A)\| < \epsilon$, set $\hat{A} = A$ and stop. Else, update A according to

$$A = A \cos\left(\delta \left\|\nabla_{\mathbb{S}} F(A)\right\|\right) + \frac{\nabla_{\mathbb{S}} F(A)}{\|\nabla_{\mathbb{S}} F(A)\|} \sin\left(\delta \left\|\nabla_{\mathbb{S}} F(A)\right\|\right).$$

5. Go to Step 3.

Remarks:

(a) The update of A described in Step 4 of the algorithm has the effect of displacing A by $\delta \|\nabla_{\mathbb{S}} F(A)\|$ units of length along the great circle of $\mathbb{S}$ through A in the direction $\nabla_{\mathbb{S}} F(A)$.
(b) We often initialize the search with a linear mapping obtained from classical dimension reduction techniques such as principal component analysis or linear discriminant analysis.

5.2 Stochastic Search

Our next goal is to add a stochastic component to the deterministic gradient field $\nabla_{\mathbb{S}} F$ on $\mathbb{S}$. To simplify the calculation, instead of considering stochastic processes on the sphere, we first add a random component to $\nabla_{\mathbb{S}} F(A)$ as a vector in $\mathbb{R}^{k\times m}$ and then project it to the tangent space to $\mathbb{S}$ at A. We adopt the notation $\Pi_A \colon \mathbb{R}^{m\times k} \to T_A\,\mathbb{S}$ for the orthogonal projection of $\mathbb{R}^{m\times k}$ onto the tangent space of $\mathbb{S}$ at A, which is given by $\Pi_A(X) = X - \langle X, A\rangle A$.

Algorithm: Stochastic Gradient Search

1. Choose $A \in \mathbb{S}$, a cooling ratio $\gamma > 1$, an initial temperature $T_0 > 0$, a step size $\delta > 0$, and a positive integer N to control the number of iterations.
2. Set $t = 0$ and initialize the search with $A_t = A \in \mathbb{S}$.
3. Calculate $\nabla_{\mathbb{S}} F(A_t)$ using Eqns. 11 and 12.

4. Generate samples $w_{ij}(t) \in \mathbb{R}$, $1 \leq i \leq m$, $1 \leq j \leq k$, from the standard normal distribution and construct the tangent vector

$$f(t) = \delta \, \nabla_{\mathbb{S}} F(A_t) + \sqrt{2\delta T_t} \; \mathit{\Pi}_{A_t} \left(\sum_{i,j} w_{ij}(t) E_{ij} \right).$$

5. Moving along an arc of length $\|f(t)\|$ on the great circle through A_t in the direction of $f(t)$, define a candidate $B \in \mathbb{S}$ for update by

$$B = A_t \cos(\|f(t)\|) + \frac{f(t)}{\|f(t)\|} \sin(\|f(t)\|) \,.$$

6. Calculate $F(B)$, $F(A_t)$, and the increment $dF = F(B) - F(A_t)$.
7. Accept B with probability $\min\{e^{dF/T_t}, 1\}$. If B is accepted, set $A_{t+1} = B$. Else, set $A_{t+1} = A_t$.
8. If $t < N$, set $T_{t+1} = T_t/\gamma$ and $t = t + 1$, and go to Step 3. Else, let $\hat{A} = A_t$ and stop.

5.3 An Alternative Interpretation of OFA

Unlike linear dimension-reduction methods that rely only on orthogonal projection onto a subspace of the original feature space, OFA allows general linear mappings to a k-dimensional feature space. In this section, we show that if we are willing to consider metrics other than the Euclidean metric, then dimension reduction and subsequent data classification with OFA may be viewed as obtained from an orthogonal projection onto a subspace of the original feature space. If A is a rank r matrix, take a singular value decomposition

$$A = U \mathit{\Sigma} V^T, \tag{13}$$

where U and V are orthogonal matrices of dimensions k and m, respectively, and $\mathit{\Sigma}$ is a $k \times m$ matrix whose $r \times r$ northwest quadrant is diagonal with positive eigenvalues and whose remaining entries are all zero. Let H be the r-dimensional subspace of $\mathbb{R}^m$ spanned by the first r columns of V and denote the orthogonal projection of a vector $x \in \mathbb{R}^m$ onto H by x_H. Then,

$$Ax \cdot Ay = y^T (A^T A) x = y^T K x = y_H^T K x_H, \tag{14}$$

for any $x, y \in \mathbb{R}^m$, where $K = A^T A$ is a positive semi-definite symmetric matrix. In particular,

$$\|Ax - Ay\|^2 = (x_H - y_H)^T K (x_H - y_H). \tag{15}$$

This means that the Euclidean distance between feature vectors in the reduced space $\mathbb{R}^k$ can be interpreted as the distance between the projected vectors x_H and y_H in the original feature space with respect to the new metric

$$d(x_H, y_H) = \sqrt{(x_H - y_H)^T K (x_H - y_H)} \,. \tag{16}$$

Note that the subspace H is spanned by the eigenvectors of K associated with its non-zero eigenvalues, so that (16) does define a metric on H. Thus, OFA may be viewed as a technique to learn a subspace H of $\mathbb{R}^m$ for orthogonal dimension reduction and a positive definite quadratic form on H that are optimal for categorization based on the nearest-neighbor classifier.

6 Image Categorization

We report the results of several image categorization experiments with the Corel-1000 data set described in Section 3. In each experiment, we placed an equal number of images from each class in the training set and used the remaining ones as query images to be indexed by the nearest-neighbor classifier applied to a reduced feature learned with OFA. Initially, an image is represented by an SH-feature vector $h(I, \mathcal{F})$ of dimension 165 obtained from the 11-bin histograms associated with 5 filters applied to the R, G, and B channels. OFA was used to reduce the dimension to $k = 9$. Table 1 shows the categorization performance: T denotes the total number of images in the training set and categorization performance refers to the rate of correct indexing using all $1,000 - T$ images outside the training set as queries.

Table 1. Results of categorization experiments with the Corel-1000 data set. T is the number of training images and the dimension of the reduced feature space is 9.

T	Categorization Performance
600	85.5%
400	84.5%
200	71.7%

7 Image Retrieval

We now use the reduced features learned with OFA to retrieve images from the database. We begin with the remark that the reduced representation was optimized to categorize query images with the nearest neighbor classifier, but not necessarily to rank matches to a query image correctly according to distances in feature space. Thus, in contrast with the retrieval strategy based solely on distances adopted, for example, in [8] and [2], we propose to exploit the strengths of the image categorization method in a more essential way.

Let $A\colon \mathbb{R}^m \to \mathbb{R}^k$ be the optimal linear dimension-reduction map learned with OFA. If I is an image and $h(I, \mathcal{F}) \in \mathbb{R}^m$ is the associated SH-feature vector, we let x denote its projection to $\mathbb{R}^k$; that is,

$$x = Ah(I, \mathcal{F}). \tag{17}$$

If there are P classes of images, for each $1 \leqslant i \leqslant P$, let x_i be the reduced feature vector of the training image in class i closest to x and let

$$d_i(I) = \|x - x_i\| \tag{18}$$

be the distance from I to class i in reduced feature space.

Given a query image I and a positive integer ℓ, the goal is to retrieve a ranked list of ℓ images from the database. We assume that all images in the database have been indexed using the representation learned with OFA. Given I, rank the classes according to increasing values of the distances $d_i(I)$. We retrieve images as follows: select as many images as possible from the first class; once that class is exhausted, we proceed to the second and iterate the procedure until ℓ images are obtained. Within each class, the images are retrieved and ranked according to their Euclidean distances to I as measured in the reduced feature space.

7.1 Experimental Results

We report the results of retrieval experiments with the Corel-1000 dataset. To make objective comparisons with other systems, we only use query images that are part of the database. Since each class contains 100 images, the maximum possible number of matches to a query image is 100, where a match is an image that belongs to the same class. We first compare retrieval results using OFA learning with those obtained with SIMPLIcity and spectral histograms, as described in Section 3. We calculated the mean values $\bar{p}_i$ and $\bar{r}_I$ of the weighted precision and rank as defined in (4). The plots shown in Figure 4 show a significant improvement in retrieval performance with a learning component. OFA was used with 400 training images (OFA-400).

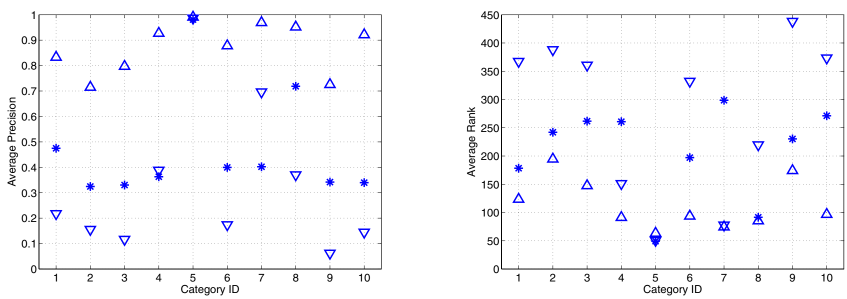

Fig. 4. (a) Plots of $\bar{p}_i$ and $\bar{r}_i$, $1 \leqslant i \leqslant 10$. The methods are labeled as: (∇) spectral histogram; ($*$) SIMPLIcity; ($\triangle$) OFA-400.

We further quantify retrieval performance, as follows. For an image I and a positive integer ℓ, let m_ℓ be the number of matching images among the top ℓ returns. Let

$$p_\ell(I) = \frac{m_\ell(I)}{\ell} \quad \text{and} \quad r_\ell(I) = \frac{m_\ell(I)}{100} \tag{19}$$

be the precision and recall rates for ℓ returns for image I. The average precision and average recall for the top ℓ returns are defined as

$$p_\ell = \frac{\sum_I p_\ell(I)}{1000} \quad \text{and} \quad r_\ell = \frac{\sum_I r_\ell(I)}{1000}, \tag{20}$$

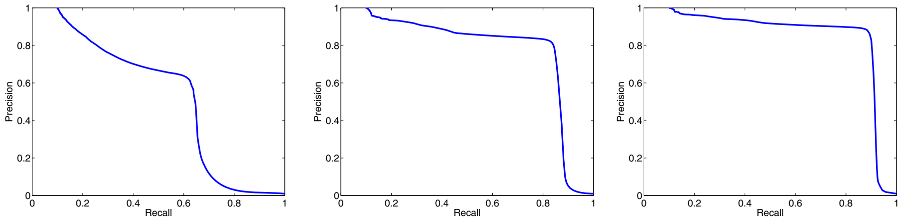

Fig. 5. Corel-1000: average-precision $\times$ average-recall plots for 200, 400 and 600 training images

Table 2. Retrieval results for OFA with T training images. Average retrieval precision (p_ℓ) and average recall (r_ℓ) for the top ℓ matches.

	ℓ	10	20	40	70	100	200	500
$T = 600$	p_ℓ	0.925	0.921	0.916	0.909	0.883	0.461	0.192
	r_ℓ	0.093	0.184	0.366	0.636	0.883	0.922	0.962

	ℓ	10	20	40	70	100	200	500
$T = 400$	p_ℓ	0.889	0.882	0.875	0.861	0.825	0.437	0.188
	r_ℓ	0.089	0.176	0.345	0.603	0.825	0.876	0.938

	ℓ	10	20	40	70	100	200	500
$T = 200$	p_ℓ	0.731	0.690	0.660	0.649	0.620	0.361	0.176
	r_ℓ	0.073	0.138	0.264	0.454	0.620	0.722	0.881

respectively. Here, the sum is taken over all 1,000 images in the database. Note that, for a perfect retrieval system, $p_\ell = 1$, for $1 \leqslant \ell \leqslant 100$, gradually decaying to $p_{1000} = 0.1$ as ℓ increases. Similarly, $r_\ell = 1$, for $\ell \geq 100$, and decays with ℓ to $r_1 = 0.01$.

Table 2 shows several values of the average precision and the average recall based on a 9-dimensional representation learned with T training images. The full average-precision $\times$ average-recall plots are shown in Figure 5. Figure 6 shows the top 10 returns for a few images in the database in an experiment with 400 training images. In each group, the first image is the query image, which is also the top return.

8 Summary and Discussion

We employed a representation of images by the histograms of their spectral components for content-based image categorization and retrieval. A feature learning technique, referred to as Optimal Factor Analysis, was developed to reduce the dimension of the representation and optimize the discriminative ability of the nearest-neighbor classifier. Several experiments were carried out and the results demonstrate a significant improvement in retrieval performance over a number of existing retrieval systems. Refinements

Fig. 6. Examples of top-10 returns. In each group, the first image is the query, which is also the top return.

of the methods will be investigated in future work to obtain sparse representations and to incorporate kernel techniques to cope with nonlinearity in data geometry. Computational strategies for faster retrieval as well as a modified version of the OFA cost function that allows a more efficient estimation of the gradient also will be investigated in future work.

Acknowledgements. This work was supported in part by NSF grants CCF-0514743 and IIS-0307998.

References

1. Carson, C., Thomas, M., Belongie, S., Hellerstein, J., Malik, J.: Blobworld: a system for region-based image indexing and retrieval. In: Proc. Visual Information Systems, pp. 509–516 (1999)
2. Hoi, S., Liu, W., Lyu, M., Ma, W.-Y.: Learning distance metrics with contextual constraints for image retrieval. In: Proc. CVPR 2006 (2006)
3. Liu, X., Mio, W.: Splitting factor analysis and multi-class boosting. In: Proc. ICIP 2006 (2006)
4. Liu, X., Srivastava, A., Gallivan, K.: Optimal linear representations of images for object recognition. IEEE Trans. Pattern Analysis and Machine Intelligence 26, 662–666 (2004)
5. Portilla, J., Simoncelli, E.: A parametric texture model based on joint statistics of complex wavelet coeficients. International Journal of Computer Vision 40, 49–70 (2000)
6. Rubner, Y., Guibas, L., Tomasi, C.: The earth mover's distance, multi-dimensional scaling, and color-based image databases. In: Proc. DARPA Image Understanding Workshop, pp. 661–668 (1997)
7. Smith, J., Li, C.: Image classification and querying using composite region templates. Computer Vision and Image Understanding 75(9), 165–174 (1999)
8. Wang, J., Li, J., Wiederhold, G.: SIMPLIcity: Semantics-sensitive integrated matching for picture libraries. IEEE Trans. on Pattern Analysis and Machine Intelligence 23(9), 947–963 (2001)
9. Wu, Y., Zhu, S., Liu, X.: Equivalence of Julesz ensembles and FRAME models. International Journal of Computer Vision 38, 247–265 (2000)
10. Yin, P.-Y., Bhanu, B., Chang, K.-C., Dong, A.: Integrating relevance feedback techniques for image retrieval using reinforcement learning. IEEE Trans. on Pattern Analysis and Machine Intelligence 27(10), 1536–1551 (2005)
11. Zhu, S., Wu, Y., Mumford, D.: Filters, random fields and maximum entropy (FRAME). International Journal of Computer Vision 27, 1–20 (1998)