Proceedings of the 1998 IEEE/RSJ
Intl. Conference on Intelligent Robots and Systems
Victoria, B.C., Canada • October 1998

Fast Eigenspace Decomposition of Correlated Images

C-Y. Chang, A. A. Maciejewski, and V. Balakrishnan
Purdue University
1285 Electrical Engineering Building
West Lafayette, Indiana 47907-1285

Abstract

We present a computationally efficient algorithm for the eigenspace decomposition of correlated images. Our approach is motivated by the fact that for a planar rotation of a two-dimensional image, analytical expressions can be given for the eigendecomposition, based on the theory of circulant matrices. These analytical expressions turn out to be good first approximations of the eigendecomposition, even for three-dimensional objects rotated about a single axis. We use this observation to automatically determine the dimension of the subspace required to represent an image with a guaranteed user-specified accuracy, as well as to quickly compute a basis for the subspace. Examples show that the algorithm performs very well on a range of test images composed of three-dimensional objects rotated about a single axis.

I. Introduction

One of the fundamental problems in computer vision is the recognition and localization of three-dimensional objects. Subspace methods represent one computationally efficient approach for dealing with this class of problems. Variously referred to as eigenspace methods, principal component analysis methods, and Karhunen-Loeve transformation methods [1], these have been used extensively in a variety of applications such as face characterization [2] and recognition [3], lip-reading [4, 5], object recognition, pose detection, visual tracking, and inspection [6, 7, 8, 9]. All of these applications are based on taking advantage of the fact that a set of highly correlated images can be approximately represented by a small set of eigenimages. Once the set of principal eigenimages is determined, online computation using these eigenimages can be performed very efficiently. However, the offline calculation required to determine both the appropriate number of eigenimages as well as the eigenimages themselves can be prohibitively expensive. This issue has been previously addressed by three different approaches. One class of techniques find the eigenimages iteratively. For example, the power method [10] and the conjugate gradient algorithm [11, 12] calculate one eigenimage at a time, while the block power method and Lanczos iteration [13] calculate a set of eigenimages together. Another class of techniques relies on updating a small set of eigenimages by recursively adding one image at a time. In [14], the number of eigenimages is fixed, while in [15], this number is adjusted based on the content of the added image. Another approach is based on structuring the computations in order to efficiently perform the matrix calculations involved [16]. The computational complexity of this approach is essentially independent of the number of desired eigenimages.

Our work addresses the computational expense of computing the desired eigenimages in a fundamentally different manner, resulting in considerable computational savings as compared to previous approaches. We present a brief overview of subspace methods in the next section, followed by the problem statement. In §III, we use the theory of circulant matrices to derive an analytical expression for the eigendecomposition of images resulting from planar rotations. In §IV, we illustrate that these analytical expressions represent a good approximation for the eigendecomposition of images of three-dimensional objects rotated about a single axis. We use this observation as the core of an algorithm, outlined in §V, to quickly compute the desired portion of the eigendecomposition based on a user-specified measure of accuracy. In §VI, we describe the performance of the algorithm evaluated on twenty different three-dimensional objects. The results of this evaluation verify the accuracy and computational efficacy of the proposed technique.

II. Preliminaries

An image is an $h \times v$ array of square pixels with intensity values normalized between 0 and 1. Thus, an image will be represented by a matrix $\mathcal{X} \in [0,1]^{h \times v}$. Since we will be considering sets of related images, it will be convenient to represent an image equivalently as a vector, obtained simply by "row-scanning", i.e.,

This work was supported by the Sze Tsao Chang Memorial Engineering Fund and by the Office of Naval Research under contract no. N00014-97-1-0640.

0-7803-4465-0/98 $10.00 © 1998 IEEE

concatenating the rows to obtain the *image vector* $\mathbf{x}$ of length $m = hv$:

$$\mathbf{x} = \text{vec}(\mathcal{X}^T).$$

The *image data matrix* of a set of images $\mathcal{X}_1, \ldots, \mathcal{X}_n$ is an $m \times n$ matrix, denoted X, and defined as

$$X = [\mathbf{x}_1 \ \cdots \ \mathbf{x}_n],$$

with typically $m \gg n$.

The *average image vector* is denoted $\bar{\mathbf{x}}$ and defined as

$$\bar{\mathbf{x}} = (\mathbf{x}_1 + \cdots + \mathbf{x}_n)/n.$$

The corresponding *average image data matrix*, denoted $\bar{X}$, is

$$\bar{X} = [\bar{\mathbf{x}} \ \cdots \ \bar{\mathbf{x}}].$$

The matrix $X - \bar{X}$, which we denote $\hat{X}$, has the interpretation of an "unbiased" image data matrix.

The singular value decomposition (SVD) of $\hat{X}$ is given by

$$\hat{X} = \hat{U}\hat{\Sigma}\hat{V}^T,$$

where $\hat{U} \in \mathbb{R}^{m \times m}$ and $\hat{V} \in \mathbb{R}^{n \times n}$ are orthogonal, and $\hat{\Sigma} \in \mathbb{R}^{m \times n}$, with $\hat{\Sigma} = [\hat{\Sigma}_d \ \mathbf{0}]^T$, where $\hat{\Sigma}_d = \text{diag}(\hat{\sigma}_1, \ldots, \hat{\sigma}_n)$, with $\hat{\sigma}_1 \geq \hat{\sigma}_2 \geq \cdots \geq \hat{\sigma}_n \geq 0$, and $\mathbf{0}$ is an n by $m - n$ zero matrix. (When the singular values $\hat{\sigma}_i$ are not ordered, we will refer to the decomposition as an "unordered" SVD.) The SVD of $\hat{X}$ plays a central role in several important imaging applications such as image compression, pattern recognition and pose detection. The columns of $\hat{U}$, denoted $\hat{\mathbf{u}}_i$, $i = 1, \ldots, m$, are referred to as the eigenimages of $\hat{X}$; these can be interpreted as estimates of the eigenvectors of the covariance matrix of the image vector. The eigenimages provide an orthonormal basis for the columns of $\hat{X}$, ordered in terms of importance; the corresponding singular values measure how "aligned" the columns of $\hat{X}$ are, with the associated eigenimage. The components of the ith column of $\hat{V}$ measure how much each individual image contributes to the ith eigenimage.

In practice, the singular vectors $\hat{\mathbf{u}}_i$ are not known or computed exactly, and instead estimates $\mathbf{q}_1, \ldots, \mathbf{q}_k$ which form a k-dimensional basis are used. The accuracy of a practical implementation of subspace methods then depends on three factors: the properties of X, the dimension k, and the quality of the estimates $\mathbf{q}_i$. The measure we will use for quantifying this accuracy is the "energy recovery ratio", denoted ρ, and defined as

$$\rho(\hat{X}, \mathbf{q}_1, \ldots, \mathbf{q}_k) = \frac{\sum_{i=1}^{k} \|q_i^T \hat{X}\|^2}{\|\hat{X}\|_F^2},$$

where $\|\cdot\|_F$ denotes the Frobenius norm. Note that if the $\mathbf{q}_i$ are orthonormal, $\rho \leq 1$, and for any given k achieves a maximum value of $(\sum_{i=1}^{k} \hat{\sigma}_i^2)/(\sum_{i=1}^{n} \hat{\sigma}_i^2)$ when $\text{span}(\mathbf{q}_1, \ldots, \mathbf{q}_k) = \text{span}(\hat{\mathbf{u}}_1, \ldots, \hat{\mathbf{u}}_k)$.

The principal calculation required with subspace methods is the precomputation of estimates of the singular vectors $\hat{\mathbf{u}}_1, \ldots, \hat{\mathbf{u}}_k$ of the $m \times n$ matrix $\hat{X}$. This is a very computationally expensive operation when m and n are very large. Reducing this computational expense by exploiting any correlation between image vectors has been the subject of much previous work [10]–[16]. Our solution to this problem uses a fundamentally different approach that is considerably faster than these methods when the image vectors are "correlated", as in many pose-detection problems. Our technique is motivated by the observation that the SVD of $\hat{X}$ can be determined in a closed form when the images are derived by a planar rotation of a single image about the surface normal, thus resulting in $\hat{X}^T\hat{X}$ being circulant.[1] We describe this in the next section.

III. Planar Rotation

Consider an image data matrix where each $\mathbf{x}_{i+1}$ is obtained from $\mathbf{x}_i$ by a planar rotation[2] of $\theta = 2\pi/n$. Equivalently, $\mathbf{x}_i$ and $\mathbf{x}_j$ are related by a planar rotation of $|i - j|\theta$. Consider

$$X^T X = \begin{bmatrix} \mathbf{x}_1^T\mathbf{x}_1 & \mathbf{x}_1^T\mathbf{x}_2 & \cdots & \mathbf{x}_1^T\mathbf{x}_n \\ \mathbf{x}_2^T\mathbf{x}_1 & \mathbf{x}_2^T\mathbf{x}_2 & \cdots & \mathbf{x}_2^T\mathbf{x}_n \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{x}_n^T\mathbf{x}_1 & \mathbf{x}_n^T\mathbf{x}_2 & \cdots & \mathbf{x}_n^T\mathbf{x}_n \end{bmatrix}. \quad (1)$$

To within an accuracy imposed by the resolution, $\mathbf{x}_i^T\mathbf{x}_j$ is a function of $|i - j|$. Thus, row $i + 1$ of X^TX can be obtained by a right-circular shift of row i (the first row is a right-circular shift of the last one). In other words, X^TX is a *circulant* matrix [18]. Much is known about the properties of such matrices; in particular, closed-form expressions can be given for their eigenvalues and eigenvectors: The eigenvalues of X^TX are simply given by the Discrete Fourier Transform of its first row, and the eigenvectors given by the Fourier matrix F. That is,

$$X^T X = F\Lambda F^* \quad (2)$$

where, with $\omega = e^{-j2\pi/n}$ and $P(z) = \sum_{i=0}^{n-1} \mathbf{x}_1^T\mathbf{x}_{i+1} z^i$,

$$\Lambda = \text{diag}\left(P(\omega^0), P(\omega^1), \ldots, P(\omega^{n-1})\right), \quad (3)$$

and

$$F = \frac{1}{\sqrt{n}} \begin{bmatrix} 1 & 1 & 1 & \cdots & 1 \\ 1 & \omega & \omega^2 & \cdots & \omega^{n-1} \\ 1 & \omega^2 & \omega^4 & \cdots & \omega^{2(n-1)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & \omega^{n-1} & \omega^{2(n-1)} & \cdots & \omega^{(n-1)(n-1)} \end{bmatrix}. \quad (4)$$

[1] This observation can be found in [17], which was published while our work was under review.

[2] More precisely, the image $i + 1$ is obtained by rotating the infinite-resolution image represented by the ith image, and then sampling it.

It is easy to verify from (3) that the eigenvalues of X^TX satisfy $P(\omega^i) = P(\omega^{n-i})$ for $i = 1, 2, \ldots, \lfloor (n-1)/2 \rfloor$, and the corresponding eigenvectors (i.e., columns of F) are complex conjugates of each other. Therefore, a *real* eigendecomposition of X^TX is given by

$$X^TX = HDH^T, \tag{5}$$

where D equals the $n \times n$ matrix

$$\text{diag}\left(P(\omega^0), P(\omega^1), P(\omega^1), P(\omega^2), \cdots\right), \tag{6}$$

and H consists of the first n columns of

$$\sqrt{2}\left[\tfrac{1}{\sqrt{2}}\mathbf{f}_1 \;\; \Re\mathbf{f}_2 \;\; \Im\mathbf{f}_2 \;\; \Re\mathbf{f}_3 \;\; \Im\mathbf{f}_3 \;\; \ldots\right] =$$

$$\sqrt{\tfrac{2}{n}}\begin{bmatrix} \frac{1}{\sqrt{2}} & c_0 & -s_0 & c_0 & -s_0 & \cdots \\ \frac{1}{\sqrt{2}} & c_1 & -s_1 & c_2 & -s_2 & \cdots \\ \vdots & \vdots & \vdots & \vdots & \cdots & \cdots \\ \frac{1}{\sqrt{2}} & c_{n-1} & -s_{n-1} & c_{2(n-1)} & -s_{2(n-1)} & \cdots \end{bmatrix} \tag{7}$$

where $c_k = \cos(k\theta)$ and $s_k = \sin(k\theta)$. The above development means that Σ and V corresponding to an unordered SVD of X can be computed in a closed form. In particular, the square roots of the diagonal entries of D are the singular values of X, and $V = H$. To compute U, observe that $U\Sigma = XH$, which can be computed efficiently using FFT techniques. In particular, if Y is a matrix whose ith row is the FFT of the ith row of X, then $Y = \sqrt{n}XF$. The matrix XH can be formed from the first n columns of Y as

$$XH = \sqrt{\frac{2}{n}}\left[\frac{1}{\sqrt{2}}\mathbf{y}_1 \;\; \Re\mathbf{y}_2 \;\; \Im\mathbf{y}_2 \;\; \Re\mathbf{y}_3 \;\; \Im\mathbf{y}_3 \;\; \ldots\right] \tag{8}$$

The above development has focused on obtaining an SVD of X. Note that the (unordered) SVD of $\hat{X}$ can be immediately obtained from the (unordered) SVD of X as follows:

$$\begin{aligned} \hat{X} &= X - \bar{X} \\ &= X - \sigma_1 \mathbf{u}_1 \mathbf{v}_1^T \\ &= \sum_{i=2}^{n} \sigma_i \mathbf{u}_i \mathbf{v}_i^T. \end{aligned}$$

In other words, for $i = 1, \ldots, n-1$, we have $\hat{\sigma}_i = \sigma_{i+1}$ (and similarly for $\hat{\mathbf{u}}_i$ and $\hat{\mathbf{v}}_i$), with $\hat{\sigma}_n = 0$, $\hat{\mathbf{u}}_m = \mathbf{u}_1$, and $\hat{\mathbf{v}}_n = \mathbf{v}_1$.

This section has considered only planar rotations of images. The next section considers how well these results apply to images obtained from rotations in a three-dimensional world.

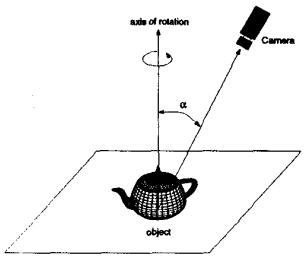

Fig. 1. The framework for obtaining images of a 3D object, rotated about a single axis through the object. The camera view vector makes an angle of α with the axis of rotation, with the entire object always lying completely within the camera's field of view.

IV. Objects rotated about a single axis

We now consider the case when the n images in the image data matrix are obtained from three-dimensional objects rotated about a single axis at increments of $2\pi/n$. We will assume that the camera view vector makes an angle of α with the axis of rotation, and that the entire object is always within the field of view (see Fig. 1).

Suppose that the axis of rotation and the camera view vector are aligned (i.e., $\alpha = 0$). Then, all of the results of the previous section apply directly[3]. If α is nonzero, then in general, the results of §III do not apply. However, consider a *planar* object whose surface normal is aligned with the axis of rotation. Then, the results of §III apply independent of α. To see this, let $\mathbf{x}_i(\alpha)$ denote the image vector of the object with camera view angle α, and note that we have

$$\mathbf{x}_i(\alpha)^T\mathbf{x}_j(\alpha) = \mathbf{x}_i(0)^T\mathbf{x}_j(0)\cos\alpha + C(\alpha), \tag{9}$$

where $C(\alpha)$ represents the contribution due to the background of the image. From the arguments in §III, it follows that $\mathbf{x}_i(0)^T\mathbf{x}_j(0)$ is only a function of $|i-j|$, and consequently, so is $\mathbf{x}_i(\alpha)^T\mathbf{x}_j(\alpha)$.

Although the situation is much more complicated for three-dimensional objects with nonzero view angles (for example, due to curved surfaces and self-occlusion), the analysis of §III often provides a good, approximate method to quickly estimate the SVD. We demonstrate this here with a simple example. (More realistic examples are considered in §VI.)

The object considered in the example is a cylinder that is half-black and half-white, split along the longitudinal axis. If $\alpha = 0$, i.e., when the cylinder is viewed along the longitudinal axis so that it appears as a circle, the results of §III apply and therefore the right singular vectors are pure sinusoids. When $\alpha \neq 0$, the results of §III do not apply (see Fig. 2 for

[3]This assumes that all the light sources rotate with the object, or equivalently the camera rotates and everything else is stationary. We will also assume that the camera is far enough away that perspective effects can be neglected.

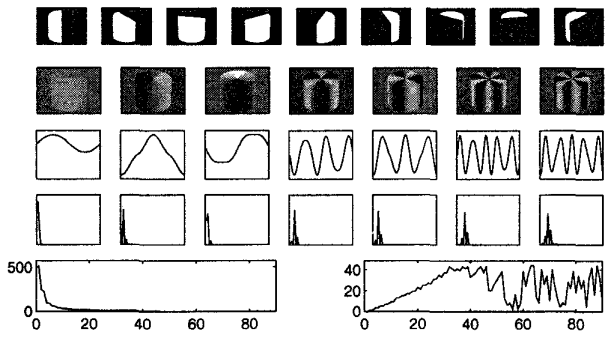

Fig. 2. This figure shows the eigendecomposition of the image matrix X obtained from rotating a half black, half white cylinder, with a view angle of $\alpha = 60°$. The first row shows nine of the 90 images of the image data matrix X. The second row shows the first seven eigenimages (left singular vectors of X) using the same gray scale encoding, with white denoting the maximum positive pixel value and black denoting the maximum negative value. The third row shows the first seven right singular vectors of X. From the fourth row, where the FFT magnitudes of these right singular vectors are shown, it is apparent that though the right singular vectors of X are not pure sinusoids, their spectra are concentrated in a narrow band around frequencies that are harmonics of $2\pi/n$. The plot on the left in the last row shows the singular values of X. Note that the singular values from indices 45 onwards are identically zero, due to the symmetry of the object. The plot on the right shows the frequency at which the FFT of the corresponding right singular vectors achieves a maximum (i.e., the "dominant" frequencies). It can be seen that the dominant frequencies of the spectra of the right singular vectors corresponding to nonzero singular values increase almost linearly with their index.

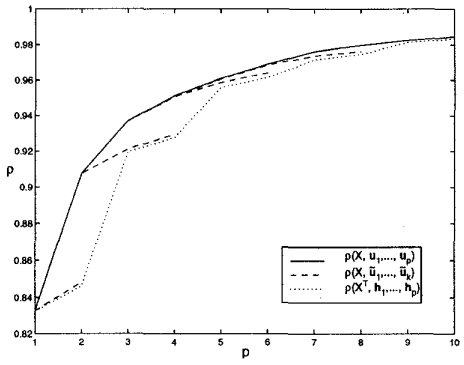

Fig. 3. This figure illustrates the typical relationship between several energy recovery ratios as a function of k, $1 \leq k \leq p$, for several fixed values of p. (The plots shown here correspond to Object 1 from Fig. 4.) For fixed p, $\rho(X, \tilde{\mathbf{u}}_1, \ldots, \tilde{\mathbf{u}}_k)$ behaves as a very good lower bound to $\rho(X, \mathbf{u}_1, \ldots, \mathbf{u}_k)$ for small k, and is very well approximated from below by $\rho(X^T, \mathbf{h}_1, \ldots, \mathbf{h}_k)$ for large k.

the case $\alpha = 60°$). However, two properties are immediately apparent: (i) The right singular vectors are well-approximated by sinusoids of frequencies that are multiples of $2\pi/n$ radians, and the spectra of the right singular vectors consist of a narrow band around the corresponding dominant harmonics. (ii) The dominant frequencies of the spectra of the (ordered) singular vectors increase approximately linearly with their index. Empirical evidence (see §VI) suggests that these properties hold approximately true for a large class of three-dimensional objects. This forms the basis of a fast approximate eigenimage computation technique that we present in the next section.

V. A FAST EIGENDECOMPOSITION ALGORITHM

Our objective is to determine the first k left singular vectors of X. Let p be such that the spectra of the first k singular vectors are essentially restricted to the band $[0, 2\pi p/n]$. Owing to the properties of the singular vectors discussed above, p is typically not much larger than k. Let H_p denote the matrix comprising the first p columns of H (i.e., the first p columns of the matrix given in (7)). Then the first k singular values $\tilde{\sigma}_1, \ldots, \tilde{\sigma}_k$ and the corresponding left singular vectors $\tilde{\mathbf{u}}_1, \ldots, \tilde{\mathbf{u}}_k$ of XH_p serve as excellent estimates to those of X. (Note that XH_p typically has far fewer columns than X, so that its SVD can be computed much more quickly.)

Moreover, the accuracy of the approximated singular vectors with spectra concentrated around "lower" frequencies will tend to be better, i.e., the smaller i is, the better estimate $\tilde{\mathbf{u}}_i$ is of $\mathbf{u}_i$. This is illustrated in Fig. 3 for a typical image (Object 1 from Fig. 4), where we have used ρ to measure the quality of the estimates of the $\mathbf{u}_i$. The solid line shows $\rho(X, \mathbf{u}_1, \ldots, \mathbf{u}_p)$ as a function of p, while the dashed lines show $\rho(X, \tilde{\mathbf{u}}_1, \ldots, \tilde{\mathbf{u}}_k)$ for $k = 1, 2, ..., p$ and $p = 2, 4, 6, 8$. It is evident that for any p, the error $\rho(X, \mathbf{u}_1, \ldots, \mathbf{u}_k) - \rho(X, \tilde{\mathbf{u}}_1, \ldots, \tilde{\mathbf{u}}_k)$ increases as k increases from 1 to p.

Our ultimate goal is to guarantee, upon termination, that $\rho(X, \tilde{\mathbf{u}}_1, \ldots, \tilde{\mathbf{u}}_k)$ exceeds a user-specified threshold μ. However, note that $\rho(X, \tilde{\mathbf{u}}_1, \ldots, \tilde{\mathbf{u}}_k)$ depends critically on k and $\tilde{\mathbf{u}}_1, \ldots, \tilde{\mathbf{u}}_k$, neither of which are available a priori. However, it can be shown that

$$\rho(X, \tilde{\mathbf{u}}_1, \ldots, \tilde{\mathbf{u}}_p) \geq \rho(X^T, \mathbf{h}_1, \ldots, \mathbf{h}_p), \qquad (10)$$

where $\mathbf{h}_i$ denotes the ith column of H. The right-hand side of (10) is readily computed; and ensuring that $\rho(X^T, \mathbf{h}_1, \ldots, \mathbf{h}_p) \geq \mu$ in turn guarantees that $\rho(X, \tilde{\mathbf{u}}_1, \ldots, \tilde{\mathbf{u}}_p) \geq \mu$.

From Fig. 3 it can be seen that $\rho(X^T, \mathbf{h}_1, \ldots, \mathbf{h}_p)$ is a very conservative lower bound for $\rho(X, \mathbf{u}_1, \ldots, \mathbf{u}_p)$, with the quality of the bound improving uniformly with increasing p. For fixed p, $\rho(X, \tilde{\mathbf{u}}_1, \ldots, \tilde{\mathbf{u}}_k)$ behaves as a very good lower bound to $\rho(X, \mathbf{u}_1, \ldots, \mathbf{u}_k)$ for small k, and is very well approximated from below by $\rho(X^T, \mathbf{h}_1, \ldots, \mathbf{h}_k)$ for large k.

In summary, when p is chosen so as to satisfy $\rho(X^T, \mathbf{h}_1, \ldots, \mathbf{h}_p) \geq \mu$, the quantity $\rho(X, \tilde{\mathbf{u}}_1, \ldots, \tilde{\mathbf{u}}_k)$ turns out to exceed μ for some $k \leq p$, with $\tilde{\mathbf{u}}_1, \ldots, \tilde{\mathbf{u}}_k$ being very good estimates for $\mathbf{u}_1, \ldots, \mathbf{u}_k$; and $\tilde{\sigma}_1, \ldots, \tilde{\sigma}_k$ being very good estimates for $\sigma_1, \ldots, \sigma_k$. The energy recovery ratio $\rho(X, \tilde{\mathbf{u}}_1, \ldots, \tilde{\mathbf{u}}_k)$ can be efficiently approximated by $\sum_{i=1}^{k} \tilde{\sigma}_i^2 / \|X\|_F^2$.

The entire algorithm for the fast computation of a partial SVD of X can be summarized as follows.

1. Form the matrix Y, whose ith row is the FFT of the ith row of X.
2. Determine the smallest number p such that $\rho(X^T, \mathbf{h}_1, \ldots, \mathbf{h}_p) > \mu$, where μ is the user-specified reconstruction ratio. The key observation here is that the matrix XH_p can be constructed as the first p columns of the matrix $\sqrt{\frac{2}{n}}[\frac{1}{\sqrt{2}}\mathbf{y}_1 \ \Re\mathbf{y}_2 \ \Im\mathbf{y}_2 \ \Re\mathbf{y}_3 \ \Im\mathbf{y}_3 \cdots]$, where $\mathbf{y}_i$ denotes the ith column of Y.
3. With Z_p denoting the first p columns of the matrix $[\frac{1}{\sqrt{2}}\mathbf{y}_1 \ \Re\mathbf{y}_2 \ \Im\mathbf{y}_2 \ \Re\mathbf{y}_3 \ \Im\mathbf{y}_3 \cdots]$, compute the SVD of the matrix Z_p.
4. Return $\tilde{\mathbf{u}}_1, \ldots, \tilde{\mathbf{u}}_k$ such that $\rho(X, \tilde{\mathbf{u}}_1, \ldots, \tilde{\mathbf{u}}_k) \geq \mu$.

The above algorithm computes the partial SVD of X. If instead the partial SVD of $\hat{X}$ is sought, the algorithm is modified as follows. In Step 2, p is estimated as the smallest number p such that $\sum_{i=2}^{p} \|X\mathbf{h}_i\|^2 > \mu(\|X\|_F^2 - \|X\mathbf{h}_1\|^2)$. In Step 3, the SVD of the matrix comprising the second through p columns of Z is computed.

We briefly analyze the computational expense of our algorithm. The cost incurred in Step 1, i.e., performing the FFT of each row of X, requires $O(mn\log_2 n)$ flops. Step 2, that of estimating p, requires of $O(mp)$ flops. In Step 3, the cost of computing the SVD of the matrix comprising the first p columns of $\sqrt{\frac{n}{2}}XH$ is of order $O(mp^2)$. Step 4, determining the needed dimension k, requires $O(mnk)$ flops. If $p \ll n$, then the total computation required is approximately $O(mn\log_2 n)$. This compares very favorably with the direct SVD approach which requires $O(mn^2)$ flops, and in most cases with the updating SVD method as well, which requires $O(mnk^2)$ flops.

VI. Example

We illustrate our approach on a database of images provided by [19]. There are twenty different objects available, with each image data matrix being of size $128^2 \times 72$. A single image of each object is shown in Fig. 4.

The algorithm outlined in §V was used to compute the eigendecomposition of $\hat{X}$ corresponding to each of the image data matrices, with an energy recovery ratio threshold of 0.90. Table I summarizes the performance of the algorithm, showing k, p, and computation times. In addition, Table I also shows the data when the direct SVD of MATLAB is used to compute the first n singular values and vectors. Compared to the direct SVD which took about 40 seconds for each object, the median speedup factor with our algorithm was approximately three. We also compared our algorithm against

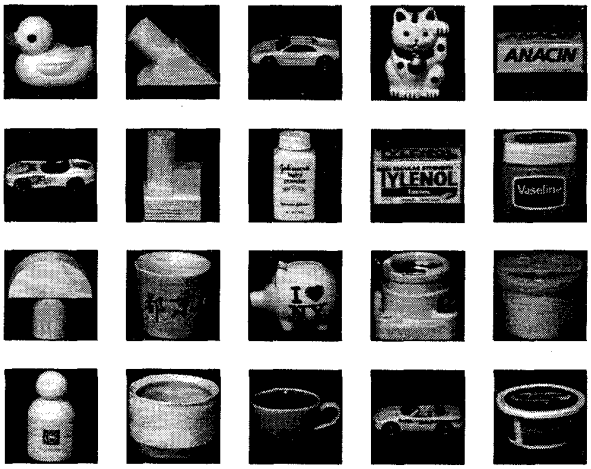

Fig. 4. This figure shows the objects used to evaluate the proposed algorithm (provided by [19]). The objects are rotated throughout 360 degrees and 72 images were taken for each of them. Each image is of size 128 × 128 and is scale normalized such that the object touches a boundary.

other algorithms [10, 11, 13, 14], using a fair and reasonable stopping criterion. It was observed that our algorithm yielded significant computational savings as compared to these algorithms as well. Details of these comparisons can be found in [20].

With our algorithm, the difference between $\rho(\hat{X}, \hat{\mathbf{u}}_1, \ldots, \hat{\mathbf{u}}_k)$ and $\rho(\hat{X}, \tilde{\mathbf{u}}_1, \ldots, \tilde{\mathbf{u}}_k)$ for each object is less than 2.22%, with an average of 0.85%, which reveals that $\{\tilde{\mathbf{u}}_1, \ldots, \tilde{\mathbf{u}}_k\}$ provides a very good approximate basis for the span of the first k eigenimages $\{\hat{\mathbf{u}}_1, \ldots, \hat{\mathbf{u}}_k\}$. As discussed in §IV, this is a consequence of the following empirical facts: (i) The FFT of the right singular vector $\mathbf{v}_i$ turns out to be approximately band-limited. (ii) The frequency at which the magnitude of the FFT of $\mathbf{v}_i$ achieves a maximum roughly increases with increasing i. Thus, the span of $\{\mathbf{h}_2, \ldots, \mathbf{h}_p\}$ effectively "covers" the span of $\{\hat{\mathbf{v}}_1, \ldots, \hat{\mathbf{v}}_k\}$.

We next turn to image-specific conclusions that can be inferred from Fig. 4 and Table I. While Object 1 requires a value of $k = 9$ to achieve an energy recovery ratio of 0.90, the value of k for Object 9 is four times as large. This illustrates that determining a priori the dimension of the subspace required to achieve a prespecified quality of reconstruction is difficult. Thus, other algorithms such as the updating SVD which do use a fixed value of k cannot be expected to perform uniformly well over all images. In contrast, our online estimate of k (given by p) can be seen to perform extremely well for most objects. In cases when the estimate p of k is poor, it can be seen that the corresponding object is rotationally symmetric; thus the associated pose-detection problem is ill-conditioned (see Object 17, and also 12 and 16).

VII. Conclusions

We have illustrated a very computationally efficient algorithm for computing the eigenspace decomposition of correlated images. In addition to its speed, the algorithm enjoys the advantage that the dimension of the subspace required to achieve a desired fidelity of representation is determined automatically; thus the amount of computation is "adapted" to meet accuracy requirements. Examples show that the algorithm performs very well on a range of test images composed of three-dimensional objects rotated about a single axis. The approach presented herein was demonstrated on rotationally correlated images; its applicability to images with other types of correlation is a topic that is currently under investigation.

TABLE I
ALGORITHM PERFORMANCE

	Dimension		Time (sec)	
Object	k	p	Our method	Direct SVD
1	9	11	6.72	40.53
2	12	15	8.46	40.53
3	25	29	16.53	41.12
4	16	19	10.21	40.33
5	31	33	19.57	41.95
6	27	29	16.70	40.67
7	12	17	8.99	40.24
8	15	19	9.94	40.67
9	36	39	24.52	41.30
10	18	19	10.76	40.19
11	15	19	10.14	40.96
12	24	45	27.47	39.13
13	22	25	13.80	40.32
14	18	25	13.41	41.15
15	15	21	11.10	39.86
16	14	31	16.32	39.42
17	18	39	22.83	40.39
18	22	29	16.09	40.53
19	20	23	12.59	41.66
20	31	37	22.19	40.57

The performance of the proposed algorithm is compared against that of the direct SVD algorithm, using the 20 objects shown in Fig.4. In all cases, the eigenimages of $\hat{X}$ were computed with a desired energy recovery ratio of 0.90. All computations were performed using MATLAB on a HP9000/C110 workstation.

References

[1] K. Fukunaga, *Introduction to Statistical Pattern Recognition*, Academic Press, London, second edition, 1990.

[2] L. Sirovich and M. Kirby, "Low-dimensional procedure for the characterization of human faces," *Journal of Optical Society of America*, vol. 4, no. 3, pp. 519–524, March 1987.

[3] M. Turk and A. Pentland, "Eigenfaces for recognition," *Journal of Cognitive Neuroscience*, vol. 3, no. 1, pp. 71–86, March 1991.

[4] H. Murase and R. Sakai, "Moving object recognition in eigenspace representation: Gait analysis and lip reading," *Pattern Recognition Letters*, vol. 17, no. 2, pp. 155–162, Feb. 1996.

[5] G. Chiou and J.-N. Hwang, "Lipreading from color video," *IEEE Trans. on Image Processing*, vol. 6, no. 8, pp. 1192–1195, Aug. 1997.

[6] H. Murase and S. K. Nayar, "Illumination planning for object recognition using parametric eigenspaces," *IEEE Trans. PAMI*, vol. 16, no. 12, pp. 1219–1227, Dec. 1994.

[7] H. Murase and S. K. Nayar, "Visual learning and recognition of 3-D objects from appearance," *Int. J. Computer Vision*, vol. 14, no. 1, pp. 5–24, 1995.

[8] H. Murase and S. K. Nayar, "Detection of 3D objects in cluttered scenes using hierarchical eigenspace," *Pattern Recognition Letters*, vol. 18, no. 4, pp. 375–384, 1997.

[9] S. K. Nayar, S. A. Nene, and H. Murase, "Subspace method for robot vision," *IEEE Trans. Robot. Automat.*, vol. 12, no. 5, Oct. 1996.

[10] S. Shlien, "A method for computing the partial singular value decomposition," *IEEE Trans. PAMI*, vol. 4, no. 6, pp. 671–676, Nov. 1982.

[11] R. Haimi-Cohen and A. Cohen, "Gradient-type algorithms for partial singular value decomposition," *IEEE Trans. PAMI*, vol. PAMI-9, no. 1, pp. 137–142, Jan. 1987.

[12] X. Yang, T. K. Sarkar, and E. Arvas, "A survey of conjugate gradient algorithms for solution of extreme eigen-problems for a symmetric matrix," *IEEE Trans. ASSP*, vol. 37, no. 10, pp. 1550–1556, Oct. 1989.

[13] C. R. Vogel and J. G. Wade, "Iterative SVD-based methods for ill-posed problems," *SIAM J. Sci. Comput.*, vol. 15, no. 3, pp. 736–754, May 1994.

[14] H. Murakami and V. Kumar, "Efficient calculation of primary images from a set of images," *IEEE Trans. PAMI*, vol. 4, no. 5, pp. 511–515, Sept. 1982.

[15] S. Chandrasekaran, B. Manjunath, Y. Wang, J. Winkeler, and H. Zhang, "An eigenspace update algorithm for image analysis," *CVGIP: Graphic Models and Image Processing*, vol. 59, no. 5, pp. 321–332, Sept. 1997.

[16] H. Murase and M. Lindenbaum, "Partial eigenvalue decomposition of large images using the spatial temporal adaptive method," *IEEE Trans. on Image Processing*, vol. 4, no. 5, pp. 620–629, May 1995.

[17] M. Uenohara and T. Kanade, "Optimal approximation of uniformly rotated images: Relationship between Karhunen-Loeve expansion and discrete cosine transform," *IEEE Trans. on Image Processing*, vol. 7, no. 1, pp. 116–119, Jan. 1998.

[18] P. J. Davis, *Circulant Matrices*, John Wiley and Sons, Inc., 1979.

[19] S. Nene, S. Nayar, and H. Murase, *Columbia Object Image Library (COIL-20)*, CUCS-006-96, Department of Computer Science, Columbia University, 1996.

[20] C.-Y. Chang, *Eigenspace Analysis of Correlated Images*, PhD thesis, Purdue University, 1999.