

Designing Eigenspace Manifolds: With Application to Object Identification and Pose Estimation

Randy C. Hoover and Anthony A. Maciejewski
Dept. of Electrical and Computer Eng.
Colorado State University
Fort Collins, CO 80523-1373, USA
Email: {hoover, aam}@colostate.edu

Rodney G. Roberts
Dept. of Electrical and Computer Eng.
Florida A & M - Florida State University
Tallahassee, FL 32310-6046, USA
Email: roberts@eng.fsu.edu

Abstract—Eigendecomposition has been used to classify three-dimensional objects from two-dimensional images in a variety of computer vision and robotics applications. The biggest on-line computational expense associated with using eigendecomposition is the determination of the closest point on an image manifold embedded in a high-dimensional space. The dimensionality and complexity of the space is a result of the p principal eigenimages that are selected. Unfortunately, for some real-time applications, this search may be prohibitively expensive. This work presents a method to reduce the on-line expense associated with using eigendecomposition for pose estimation. The approach is based on selecting a linear combination of the principal eigenimages to design an eigenspace manifold having a desirable geometric structure that reduces the cost associated with classification.

Index Terms—Object identification, pose estimation, eigendecomposition, manifolds.

I. INTRODUCTION

Over the last several decades, object identification and pose estimation (referred to here as object classification) of three-dimensional (3-D) objects from two-dimensional (2-D) images has become an important issue in computer vision and robotics applications. Subspace methods, also referred to as eigenspace methods, principal component analysis, or the Karhunen-Loeve transformation [1], [2], represent one computationally efficient approach for dealing with object classification and have been applied to a variety of application domains. Specific examples include robot vision [3], [4], face recognition [5]–[10], object recognition [11]–[13], pose estimation [14]–[18], visual tracking [19], [20], and automated inspection [21]. All of these applications are based on the fact that a set of highly correlated images can be approximately represented by a small set of eigenimages [4], [11], [22].

The major drawbacks to using eigenspace methods are threefold. First, eigenspace methods typically require a large off-line expense to compute the principle eigenimages of a set of images. Recent work in this area however has shown that for highly correlated image sequences, the principle eigenimages can be estimated efficiently using spectral theory [15]–[18], [23]. The second drawback to eigenspace methods is that because they are strictly appearance based, they are typically sensitive to background clutter and occlusion, i.e., they typically assume perfect segmentation of the object from the scene. A solution to this problem was recently proposed by Chang *et al.* in [24] where the authors use the image gradient to localize

the object in a cluttered scene (the object may be partially occluded), and then use a quadtree eigendecomposition to classify the object. The third drawback to using eigenspace methods is the resulting search of the eigenspace manifold may be computationally prohibitive when the number of images and/or the dimension of the eigenspace is large. This third drawback has been addressed in various ways, and is the subject of this work.

The naive approach to searching the eigenspace manifold is to use an exhaustive search, however when the dimension of the eigenspace is large this is impractical. Other methods include space partitioning algorithms such as R -tree, k - d tree, and Voronoi polygons [25]–[28]; hashing techniques [29]; and random search methods [30]. The drawback of these techniques is that they require a large amount of storage space, or their computational complexity becomes large as the dimension of the space increases. In [31], Nene and Nayar propose a method to search for nearest neighbor points in high dimensions using a space partitioning algorithm that is based on the projection search paradigm. The algorithm consists of generating hyperplanes in the eigenspace that ultimately “box-in” the point of interest with a hypercube, each side of the hypercube having length 2ϵ . An exhaustive search is then performed within this hypercube to find the closest matching point. The authors of [31] show that this algorithm is computationally less expensive than previous search techniques, however its complexity is still a function of the dimension of the space.

In this paper a fundamentally different approach for classifying 3-D objects using eigenspace methods is presented. It will be shown that rather than choosing the principal eigenimages of the resulting eigendecomposition, a linear combination of these basis vectors can be computed so that the resulting eigenspace manifold has a desirable geometric structure. This geometric structure allows for the classification of the object to be computed using simple calculations rather than searching the eigenspace. The advantage of this technique stems from the fact that because it requires no search of the eigenspace, its computational expense is constant.

The remainder of this paper is organized as follows. In Section II, the fundamentals needed to apply an eigendecomposition to a related image data set is explained, much

of which is discussed in [23]. Section II also gives a brief overview of the classification problem. In Section III, an outline of the proposed eigenspace mapping is presented. The proposed technique is applied to a general set of objects to provide a comparison with the naive approach in Section IV. Finally, Section V provides some concluding remarks and future research directions.

II. PRELIMINARIES

A. Mathematical Description

In this work, a gray-scale image is described by an $h \times v$ array of square pixels with intensity values normalized between 0 and 1. Thus an image is represented by a matrix $\mathcal{X} \in [0, 1]^{h \times v}$. Because sets of related images are considered in this paper, the *image vector* \mathbf{f} of length $m = hv$ is obtained by “row-scanning” an image into a column vector, i.e., $\mathbf{f} = \text{vec}(\mathcal{X}^T)$. The *image data matrix* of a set of images $\mathcal{X}_1, \dots, \mathcal{X}_n$ is an $m \times n$ matrix, denoted X , and defined as $X = [\mathbf{f}_1, \dots, \mathbf{f}_n]$, where typically $m > n$ with fixed n [23].

To construct the image data matrix X , consider capturing images on the surface of the sphere where the object is placed at the sphere’s center, as seen in Fig. 1. In the figure, a sample image is taken at each of the black dots on the surface of the sphere. Capturing images in this manner allows for the image vector to be defined as $\mathbf{f} = \mathbf{f}(\xi_i)$ where $\xi_i, i \in \{0, \dots, n-1\}$ is the unit vector pointing at the angle of co-latitude β measured down from the upper pole, and the angle of longitude $\alpha_i \in [0, 2\pi)$, which is the parameterization of the sphere in spherical coordinates. Setting β to a constant results in a one-dimensionally correlated image data matrix that is correlated on S^1 .

The *thin* singular value decomposition (SVD) of X is given by

$$X = U\Sigma V^T, \quad (1)$$

where $U \in \mathbb{R}^{m \times n}$ is right orthogonal, i.e., $U^T U = I$, $V \in \mathbb{R}^{n \times n}$ is orthogonal, and $\Sigma = \text{diag}(\sigma_1, \dots, \sigma_n)$ with $\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_n \geq 0$. The columns of U , denoted $\mathbf{u}_i, i = 1, \dots, n$, are referred to as the left singular vectors or eigenimages of X , while the columns of V , denoted $\mathbf{v}_i, i = 1, \dots, n$ are referred to as the right singular vectors of X . The left singular vectors (or eigenimages) of X can be interpreted as the eigenvectors of the covariance matrix XX^T . The eigenimages provide an orthonormal basis for the column space of X , ordered in terms of importance; the corresponding singular values measure how “important” the associated eigenimage is. The components of the i^{th} column of V measure how much each individual image in X contributes to the i^{th} eigenimage.

B. The Classification Problem

Classifying objects using eigendecomposition is done in two separate phases, namely, the training phase and the classification phase. The training phase is completed off-line and consists of constructing the image data matrix X , and computing the principle eigenimages U_p , where U_p consists

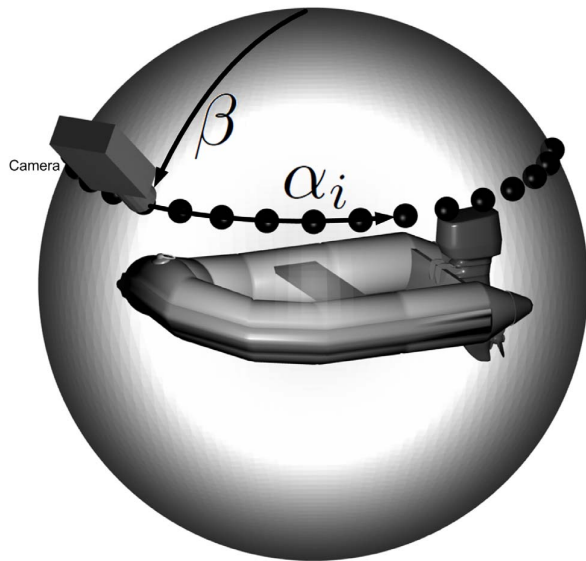


Fig. 1. This figure depicts sampling an object along a line of constant co-latitude, which results in an image data matrix correlated on S^1 . A sample image is captured at each of the black dots that reside on the sphere.

of the first p columns of U . Once the principal eigenimages of an image data matrix have been computed, the images in X are then projected onto the resulting eigenspace using $\mathcal{M}_p = U_p^T X$. This projection generates a set of n points in the eigenspace, each point having dimension p . This set of points is a discrete approximation to the underlying one-dimensional manifold embedded in p -dimensional space.¹

The on-line classification of 3-D objects then consists of projecting an input image \mathbf{f}_t into the eigenspace. Because an object’s projection will likely be close to the eigenspace manifold computed from training images of the same object, distance to the manifold can be used for object identification. Once identified, the pose of the object can be estimated by determining the closest point on the manifold.

An example of this process is shown in Fig. 2, where the top row shows five of 128 images of a boat sampled according to Fig. 1. The curve in the figure shows a linear approximation to the corresponding projection of the image data matrix X onto the first three eigenimages U_3 , i.e., the curve is a linear approximation of $\mathcal{M}_p = U_p^T X$ where $p = 3$. The dots in the figure show the projection of five of the original training images into the eigenspace. An input image is also shown in the figure, along with its projection into the three-dimensional eigenspace. The projected input image, in general, will not lie directly on the one-dimensional manifold. Instead, the pose of the object in the input image is estimated by the pose corresponding to the closest point on the manifold. This figure also shows the projection of an image of an object other than

¹This assumes the object’s pose is the only variable in generating the image data matrix X .

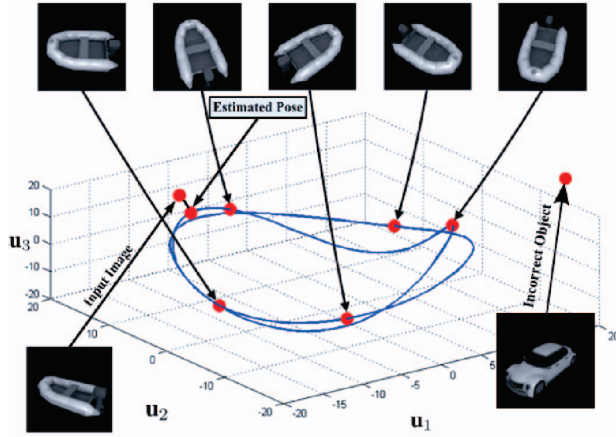


Fig. 2. An example subspace showing the image data matrix X (generated from sampling on S^1) being projected onto the first three eigenimages U_3 . The projection generates a one-dimensional manifold in \mathbb{R}^3 with five of the images in the image data set highlighted. An input image f_t is shown along with its projection onto the three-dimensional eigenspace. The closest point on the one-dimensional manifold determines the estimated pose of the object in the test image. The projection of an image of an object other than the one used to generate X is also shown.

the boat. Notice that because images of this object were not contained in X , the projection of the image lies nowhere near the eigenspace manifold. (This will typically, but not always, be true.)

The projection of the input image onto the eigenspace can be computed relatively quickly by calculating p dot products of the image with the orthogonal vectors defining the eigenspace, i.e., $U_p^T f_t$. This projection generates a new p -dimensional point in the eigenspace. The classification of the object is then determined by comparing this new point in the eigenspace with all other points that constitute the underlying manifold. Because the number of principle eigenimages p is typically small compared to the number of images, the series of dot products can be computed relatively quickly. The problem of searching the manifold for the closest matching point however can be computationally expensive. This computational expense has been addressed using several different search methods, however these methods either require a large amount of storage space or their computational expense becomes large as the dimension of the space increases [31]. Therefore, it is desirable to either reduce the dimensionality of the search space, or eliminate the search all together.

III. MANIFOLD MAPPINGS

One method of reducing the dimensionality of the search space, or eliminating the search all together, is to guarantee that the eigenspace manifold has a desirable geometric structure. In particular, the manifold shown in Fig. 2 most closely represents a circular coil in the eigenspace. This geometric structure is a function of both the image data matrix X , and the subspace (eigenimages) chosen. If a linear combination of the principal eigenimages are chosen as the subspace however, the resulting

geometric structure of the manifold in the eigenspace can be forced to have certain desirable characteristics. For example, if the images in X are correlated in one-dimension, it would be desirable for the eigenspace manifold to be approximately represented by a circle in \mathbb{R}^2 rather than a circular coil in \mathbb{R}^p . This representation would allow for a very fast determination of where the projection of the input image f_t resides in the eigenspace.

Unfortunately, simply choosing two basis vectors that map the image data matrix X onto a unit circle in \mathbb{R}^2 may not be sufficient for accurate classification because this space may not capture the intrinsic characteristics of X , i.e., the eigenimages are special because they represent the best subspace approximation of X . Therefore, we construct our transformation from the eigenspace to retain as much of the norm of X as possible. The problem of mapping the eigenspace manifold onto a unit circle in \mathbb{R}^2 can be formulated as follows:

Find a matrix $A_p \in \mathbb{R}^{p \times 2}$ such that

$$\|\mathcal{M}_p^T A_p - B\|_F^2 < \epsilon \quad (2)$$

for the user specified value ϵ , and

$$B = \begin{bmatrix} \cos(\theta_0) & \sin(\theta_0) \\ \cos(\theta_1) & \sin(\theta_1) \\ \vdots & \vdots \\ \cos(\theta_{n-1}) & \sin(\theta_{n-1}) \end{bmatrix} \quad (3)$$

where $\theta_i = \frac{2\pi i}{n}$ for $i \in \{0, 1, \dots, n-1\}$. Note that the columns of B represent a unit circle in \mathbb{R}^2 and

$$\begin{aligned} \mathcal{M}_p &= U_p^T X \\ &= \Sigma_p V_p^T, \end{aligned} \quad (4)$$

is the p -dimensional manifold determined by the principal eigenimages. Therefore,

$$\begin{aligned} \mathcal{M}_p^T A_p &= (\Sigma_p V_p^T)^T A_p \\ &= V_p \Sigma_p A_p \end{aligned} \quad (5)$$

where Σ_p is the upper $p \times p$ sub-matrix of Σ and $V_p \in \mathbb{R}^{n \times p} = [v_1, \dots, v_p]$.

This results in the following proposition:

Proposition 1: Assume that the matrix X has full rank. Then $A_p = \Sigma_p^{-1} V_p^T B$ is the unique matrix that minimizes the objective function $\|\mathcal{M}_p^T A_p - B\|_F^2$ over the family of $p \times 2$ matrices A_p . Furthermore, the value of the objective function monotonically decreases to zero as p varies from 1 to n .

Proof: We have that

$$\begin{aligned} \|\mathcal{M}_p^T A_p - B\|_F^2 &= \|V_p \Sigma_p A_p - B\|_F^2 \\ &= \|V^T (V_p \Sigma_p A_p - B)\|_F^2 \end{aligned}$$

where the last equality follows from the fact that the Frobenius norm is invariant under multiplication by an orthogonal matrix such as V^T . The objective function then simplifies to

$$\begin{aligned} \|\mathcal{M}_p^T A_p - B\|_F^2 &= \left\| \begin{bmatrix} \Sigma_p A_p \\ \mathbf{0} \end{bmatrix} - \begin{bmatrix} B_1 \\ B_2 \end{bmatrix} \right\|_F^2 \\ &= \left\| \begin{bmatrix} \Sigma_p A_p - B_1 \\ -B_2 \end{bmatrix} \right\|_F^2 \end{aligned}$$

where $B_1 = V_p^T B$, $B_2 = W_p^T B$, and W_p consists of the last $n - p$ columns of V . Since the Frobenius norm squared is simply the sum of the matrix elements squared, it readily follows that

$$\|\mathcal{M}_p^T A_p - B\|_F^2 = \|\Sigma_p A_p - B_1\|_F^2 + \|B_2\|_F^2.$$

The second term on the right is independent of A_p , so the problem becomes that of minimizing the first term on the right, which is uniquely minimized to the value zero by $A_p = \Sigma_p^{-1} B_1 = \Sigma_p^{-1} V_p^T B$.

Consider now the minimum remaining error:

$$\begin{aligned} \|V_p \Sigma_p \Sigma_p^{-1} V_p^T B - B\|_F^2 &= \|(V_p V_p^T - I)B\|_F^2 \\ &= \|(I - V_p V_p^T)B\|_F^2. \end{aligned}$$

Each column of B can be expressed as a linear combination of the vectors $\mathbf{v}_1, \dots, \mathbf{v}_n$. In particular, $\mathbf{b}_j = \alpha_{1j} \mathbf{v}_1 + \dots + \alpha_{nj} \mathbf{v}_n$, where \mathbf{b}_j is the j^{th} column of B . The error is then given by

$$\begin{aligned} \|(I - V_p V_p^T)B\|_F^2 &= \left\| \begin{bmatrix} \sum_{i=p+1}^{n-1} \alpha_{i1} \mathbf{v}_i & \sum_{i=p+1}^{n-1} \alpha_{i2} \mathbf{v}_i \\ \vdots & \vdots \end{bmatrix} \right\|_F^2 \\ &= \sum_{i=p+1}^{n-1} [\alpha_{i1}^2 + \alpha_{i2}^2], \end{aligned}$$

and is zero when $p = n$. Note that the error clearly decreases monotonically to zero as p varies from 1 to n . \square

Note that if the rank of X is $r < n$, then $A_p = \Sigma_p^+ V_p^T B$ is a minimizer where Σ_p^+ is the pseudoinverse of Σ . However, for $p > r$, this minimizer is not unique, although the quantity $\mathcal{M}_p^T A_p$ is unique. Furthermore, the error does not go to zero as p goes to n unless B is in the column space of X^T .

A typical example illustrating the behavior of this proposition on an image data matrix is shown in Fig. 3. The upper four plots show the convergence of the manifold \mathcal{M}_p onto the circle in \mathbb{R}^2 as the value p increases. The circles represent the columns of B and the asterisks represent the mapping $\mathcal{M}_p^T A_p$. Each of these four plots also shows the value of $\|\mathcal{M}_p^T A_p - B\|_F^2$ for a particular value of p . The bottom plot in Fig. 3 shows how the value $\|\mathcal{M}_p^T A_p - B\|_F^2$ decreases monotonically as p increases for this particular example.

The complete projection onto the circle in \mathbb{R}^2 encoding the p -dimensional eigenspace can now be defined by $\mathcal{P} = U_p A_p$. When a new input image arrives from the imaging system (\mathbf{f}_t), the projection through the eigenspace onto the circle is simply $\mathcal{P}^T \mathbf{f}_t$. Note that the projection $\mathcal{P}^T \mathbf{f}_t$ generates a point $\mathbf{p} = [p_1, p_2] \in \mathbb{R}^2$. Classification of the object can be done by computing $\|\mathbf{p}\|_2$ for identification and

$$\alpha = \text{atan}(p_2, p_1) \quad (6)$$

to estimate the pose.

IV. EMPIRICAL EVALUATION

To evaluate the accuracy of the proposed technique, several objects were imaged using the sampling procedure outlined in Section II. The angle of co-latitude β was set to 60° and an

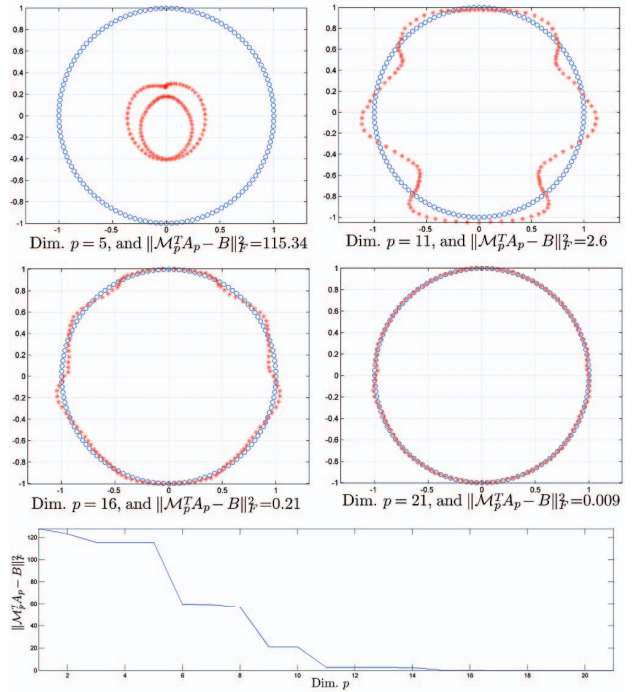


Fig. 3. Typical example of how the manifold \mathcal{M}_p is mapped onto the circle in \mathbb{R}^2 as a function of p . The top four plots show how the approximation $\mathcal{M}_p^T A_p$ becomes closer to B as p increases. The circles represent the columns of B and the asterisks represent the mapping $\mathcal{M}_p^T A_p$. The bottom plot shows how the norm $\|\mathcal{M}_p^T A_p - B\|_F^2$ decreases as a function of p .

image of each object was captured at a resolution of 128×128 . A total of 128 images were captured on S^1 resulting in an angular separation of 2.81° per sample. The images used in this paper were generated by ray-tracing high fidelity CAD models. An example image of each object is shown in Fig. 4 (the CAD models were provided by [32]). For each of the objects in Fig. 4, 64 test images were also captured on S^1 at random but known poses. None of the 64 test images were contained in the original training data matrix X . The proposed technique was then used to compute the matrix A_p mapping the original eigenspace onto a circle in \mathbb{R}^2 . Each of the 64 images for each of the 20 objects was then projected into this subspace using $\mathcal{P}^T \mathbf{f}_t$ and the pose in each image was estimated using (6). The absolute error in pose angle was then calculated for each test pose. To obtain a ground truth comparison, the pose of each of the 64 test images was also calculated using the standard eigendecomposition technique by finding the closest matching manifold point.² The dimension of the eigenspace p was the same for both techniques.

The top plot in Fig. 5 shows the distribution of the absolute pose estimation error using the proposed method for all objects in Fig. 4. The horizontal bars represent the average error, and the plus signs are outliers in the data. The circles represent the average absolute error as computed by the standard eigende-

²This error could be reduced by applying an interpolation technique.



Fig. 4. Example images of ray-traced CAD models courtesy of Kator Legaz [32]. Each object is sampled as discussed in Section II at a resolution of 128×128 . The objects are ordered from left to right, then top to bottom.

composition search technique. As can be seen from the figure, on average, the proposed technique is able to estimate the pose of the object to a higher degree of accuracy than the standard technique. It is important to note that the objects in Fig. 5 are sorted according to their energy ratio (E.R.), defined as

$$\text{E.R.} = \frac{\|(U_p \mathbf{a}_1)^T X\|_F^2 + \|(U_p \mathbf{a}_2)^T X\|_F^2}{\|\mathbf{u}_2^T X\|_F^2 + \|\mathbf{u}_3^T X\|_F^2}, \quad (7)$$

where \mathbf{a}_i is the i^{th} column of A_p . The E.R. is a measure of how aligned the vectors $U_p A_p$ are with the dominant eigenimages, and achieves a maximum value of one. The first eigenimage is excluded in this calculation because it represents the mean image of the data set and has no significance in classifying the object. The bottom plot in Fig. 5 shows the E.R. for each of the objects in Fig. 4. The data used for both of these plots was computed using $\epsilon = 0.01$.

To determine an appropriate value of ϵ to use in the calculation of (2), the average absolute error for all 1280 test images was calculated for different values of ϵ . The top plot in Fig. 6 shows that for $\epsilon \approx 0.1$, the pose estimation error is very close to that using the standard eigendecomposition search technique. The bottom plot shows the average dimension p required to achieve a desirable value of ϵ . As can be seen in this plot, the average dimension p remains relatively small for $\epsilon > 0.1$. This shows that there is a trade-off between accuracy of pose estimation and the required dimension p . This has a significant impact on the off-line computation because higher accuracy also requires more eigenimages to be computed.

Finally, Fig. 7 shows the relationship between the average error in pose estimation and the number of training images used. As can be seen in the figure, accurate pose estimation can be achieved for training sets of 64 poses or more, resulting in an angular separation of 5.6° per sample. This plot also shows that for a similar number of training samples, the proposed

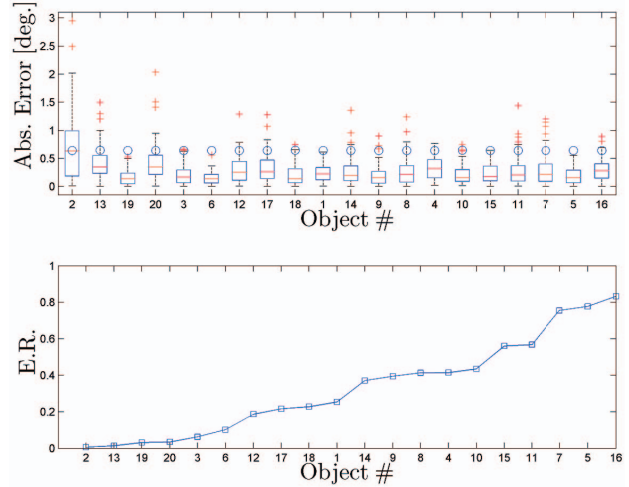


Fig. 5. The top plot shows the distribution of the absolute pose estimation error using the proposed method for all objects in Fig. 4 sorted by their E.R. The horizontal bars represent the average error, and the plus signs are outliers in the data. The circles represent the mean error as computed by the standard eigendecomposition search technique. The bottom plot shows the E.R. for each of the 20 objects.

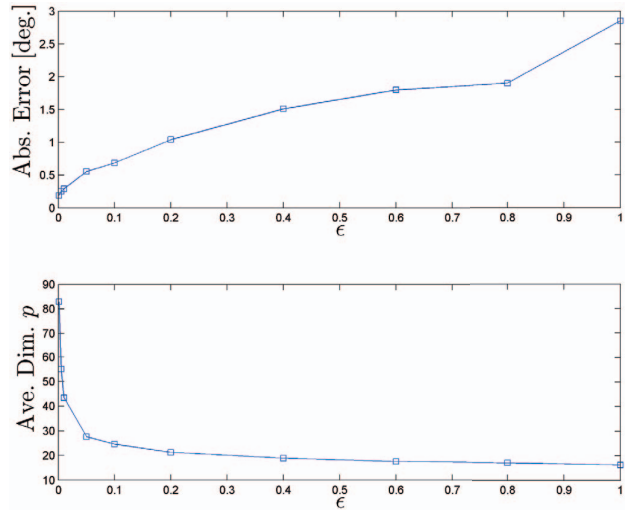


Fig. 6. The top plot shows the average absolute pose estimation error as a function of ϵ for 1280 test poses. The bottom plot shows the average dimension p required to achieve a desired value of ϵ .

technique outperforms the standard eigendecomposition search technique in terms of the accuracy of the estimated pose of the object.

V. SUMMARY AND FUTURE DIRECTIONS

This paper has presented a method to significantly reduce the on-line computation required for object classification using eigendecomposition. It has been shown that by choosing an appropriate linear combination of the principle eigenimages of an image data set, the eigenspace manifold can be forced to have a desirable geometric structure. This geometric structure

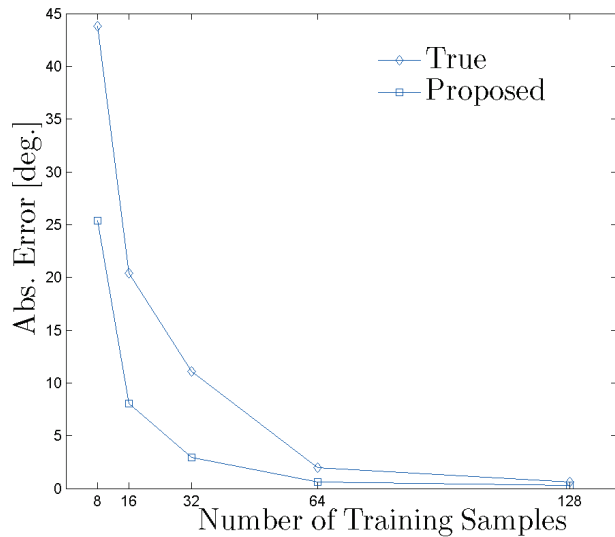


Fig. 7. The relationship between the average absolute error in pose estimation and the number of training images used for both the proposed and true, i.e., standard eigendecomposition search technique.

allows for object classification to be performed by simply computing a norm and arctangent as opposed to searching a manifold in a high-dimensional space. An empirical investigation was performed to validate the proposed technique, as well as assess the accuracy of pose estimation as compared to standard eigenspace search techniques. Future work will focus on extending this technique to data sets correlated in higher dimensions, as well as evaluating its robustness.

REFERENCES

- [1] K. Fukunaga, *Introduction to Statistical Pattern Recognition*. London, U.K.: Academic, 1990.
- [2] J. J. Gerbrands, "On the relationships between SVD, KLT and PCA," *Pattern Recognition*, vol. 14, no. 1-6, pp. 375-381, Dec. 1981.
- [3] R. T. Chin and C. R. Dyer, "Model-based recognition in robot vision," *Computing Surveys*, vol. 18, no. 1, pp. 67-108, Mar. 1986.
- [4] S. K. Nayar, S. A. Nene, and H. Murase, "Subspace methods for robot vision," *IEEE Trans. Robot. Automat.*, vol. 12, no. 5, pp. 750-758, Oct. 1996.
- [5] L. Sirovich and M. Kirby, "Low-dimensional procedure for the characterization of human faces," *J. Opt. Soc. Amer.*, vol. 4, no. 3, pp. 519-524, Mar. 1987.
- [6] M. Kirby and L. Sirovich, "Application of the Karhunen-Loeve procedure for the characterization of human faces," *IEEE Trans. PAMI*, vol. 12, no. 1, pp. 103-108, Jan. 1990.
- [7] M. Turk and A. Pentland, "Eigenfaces for recognition," *J. Cogn. Neurosci.*, vol. 3, no. 1, pp. 71-86, Mar. 1991.
- [8] R. Epstein, P. Hallinan, and A. Yuille, " 5 ± 2 eigenimages suffice: An empirical investigation of low-dimensional lighting models," in *Proc. IEEE Workshop on Physics-Based Vision*, Cambridge, MA, June 1995, pp. 108-116.
- [9] R. Ramamoorthi, "Analytic PCA construction for theoretical analysis of lighting variability in images of a Lambertian object," *IEEE Trans. PAMI*, vol. 24, no. 10, pp. 1322-1333, Oct. 2002.
- [10] R. Basri and D. Jacobs, "Lambertian reflectance and linear subspaces," *IEEE Trans. PAMI*, vol. 25, pp. 383-390, Feb. 2003.
- [11] H. Murase and S. K. Nayar, "Visual learning and recognition of 3-D objects from appearance," *Int. J. Comp. Vis.*, vol. 14, no. 1, pp. 5-24, Jan. 1995.
- [12] S. K. Nayar and H. Murase, "Dimensionality of illumination in appearance matching," in *IEEE Int. Conf. Robot. Automat.*, Minneapolis, MN, Apr. 1996, pp. 1326-1332.
- [13] B. Shekar, D. Guru, and P. Nagabhushan, "Object recognition through the principal component analysis of spatial relationship amongst lines," in *Asian Conf. on Comp. Vis.*, Jan. 2006, pp. 170-179.
- [14] J. Winkeler, B. S. Manjunath, and S. Chandrasekaran, "Subset selection for active object recognition," in *IEEE Conf. Comp. Vis. and Patt. Rec.*, Fort Collins, CO, June 1999, pp. 511-516.
- [15] K. Saitwal, A. A. Maciejewski, and R. G. Roberts, "Computationally efficient eigenspace decomposition of correlated images characterized by three parameters," accepted to appear in *Patt. Anal. and Apps.*, 2009.
- [16] R. C. Hoover, A. A. Maciejewski, and R. G. Roberts, "Aerial pose detection of 3-D objects using hemispherical harmonics," in *IEEE SSIAI*, Santa Fe, NM, Mar. 2008, pp. 157-160.
- [17] R. C. Hoover, A. A. Maciejewski, and R. G. Roberts, "Pose detection of 3-D objects using S^2 -correlated images and discrete spherical harmonic transforms," in *IEEE Int. Conf. Robot. Automat.*, Pasadena, CA, May. 2008, pp. 993-998.
- [18] R. C. Hoover, A. A. Maciejewski, and R. G. Roberts, "Pose detection of 3-D objects using images sampled on $SO(3)$, spherical harmonics, and Wigner-D matrices," in *IEEE Conf. on Automat. Sci. and Engr.*, Washington DC, Aug 2008, pp. 47-52.
- [19] S. K. Nayar, H. Murase, and S. A. Nene, "Learning, positioning, and tracking visual appearance," in *Proc. IEEE Int. Conf. Robot. Automat.*, San Diego, CA, May 1994, pp. 3237-3246.
- [20] H. Murase and R. Sakai, "Moving object recognition in eigenspace representation: Gait analysis and lip reading," *Patt. Rec. Lett.*, vol. 17, no. 2, pp. 155-162, Feb. 1996.
- [21] C. Y. Huang, O. I. Camps, and T. Kanungo, "Object recognition using appearance-based parts and relations," in *Proc. IEEE Comp. Soc. Conf. Comp. Vis. and Patt. Rec.*, San Juan, PR, June 1997, pp. 877-883.
- [22] 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.
- [23] C. Y. Chang, A. A. Maciejewski, and V. Balakrishnan, "Fast eigenspace decomposition of correlated images," *IEEE Trans. Image Proc.*, vol. 9, no. 11, pp. 1937-1949, Nov. 2000.
- [24] C.-Y. Chang, A. A. Maciejewski, V. Balakrishnan, R. G. Roberts, and K. Saitwal, "Quadtree-based eigendecomposition for pose estimation in the presence of occlusion and background clutter," *Patt. Anal. and App.*, vol. 10, no. 1, pp. 15-31, Feb. 2007.
- [25] K. Fukunaga and P. M. Narendra, "A branch and bound algorithm for computing k-nearest neighbors," *IEEE Trans. Comput.*, vol. 24, no. 7, pp. 750-753, July 1975.
- [26] J. L. Bentley, "Multidimensional binary search trees in database applications," *IEEE Trans. Softw. Eng.*, vol. 5, no. 4, pp. 333-340, July 1979.
- [27] A. Guttman, "R-trees: A dynamic index structure for spatial searching," in *SIGMOD, Proceedings of Annual Meeting*, Boston, MA, June 1984, pp. 47-57.
- [28] F. Aurenhammer, "Voronoi diagrams: A survey of a fundamental geometric data structure," *ACM Computing Surveys*, vol. 23, no. 3, pp. 345-405, Sept. 1991.
- [29] A. Califano and R. Mohan, "Multidimensional indexing for recognizing visual shapes," *IEEE Trans. PAMI*, vol. 16, no. 4, pp. 373-392, Apr. 1994.
- [30] K. L. Clarkson, "A randomized algorithm for closest-point queries," *SIAM J. Comput.*, vol. 17, no. 4, pp. 830-847, Aug. 1988.
- [31] S. Nene and S. Nayar, "A simple algorithm for nearest neighbor search in high dimensions," *IEEE Trans. PAMI*, vol. 19, no. 9, pp. 989-1003, Sept. 1997.
- [32] K. Legaz. (2007) Kator Legaz: 3-D model database for Blender. [Online]. Available: http://www.katorlegaz.com/3d_models/index.php