Eigenshapes for 3D Object Recognition in Range Data

R. J. Campbell and P. J. Flynn

Department of Electrical Engineering
The Ohio State University
Columbus, OH. 43210-1272
{campbelr,flynn}@ee.eng.ohio-state.edu

Abstract

Much of the recent research in object recognition has adopted an appearance-based scheme, wherein objects to be recognized are represented as a collection of prototypes in a multidimensional space spanned by a number of characteristic vectors (eigen-images) obtained from training views. In this paper, we extend the appearance-based recognition scheme to handle range (shape) data. The result of training is a set of 'eigensurfaces' that capture the gross shape of the objects. These techniques are used to form a system that recognizes objects under an arbitrary rotational pose transformation. The system has been tested on a 20 object database including free-form objects and a 54 object database of manufactured parts. Experiments with the system point out advantages and also highlight challenges that must be studied in future research.

1 Introduction

Appearance-based (or 'eigenface') approaches to object recognition have demonstrated the ability to recognize large numbers of general objects quickly [3],[4],[5]. These methods encode the variations of an object shape and reflectance with respect to its pose and the illumination conditions. This technique has been applied successfully in the tasks of face recognition [1], tracking of objects in image sequences [10], illumination planning [11], and object recognition in the presence of occlusion [12],[6],[7].

To our knowledge, no work with the exception of the indexing-oriented work of Johnson and Hebert [9] has employed eigendecompositions for a space based on training images generated only from shape. The application of appearance-based techniques has intuitive appeal (the absence of lighting artifacts in range data offers potential robustness) but implementation of the technique revealed some challenges. This paper describes a first-generation system for 'appearance-based' recognition of 3D objects in range images, employing two object databases. Experiments with the system show the potential power of the technique and highlight areas to be improved in future research.

*This work was supported by the National Science Foundation under grants XXX-xxxxxxx and XXX-xxxxxxx.

2 Notation and Fundamental Concepts

Let $\mathbf{Y}$ be a range image with r rows and c columns. $\mathbf{Y}$ can be viewed as a vector of length $n = r \cdot c$ by concatenating rows, yielding the n-vector $\vec{\mathbf{T}} = [y_{1,1} \cdots y_{1,c}\, y_{2,1} \cdots y_{r,1} \cdots y_{r,c}]^T$. $\vec{\mathbf{T}}$ lies in a vector space $\mathcal{I}$ with dimension n. We assume that the images $\mathbf{Y}$ contain range ($2\frac{1}{2}$D) data; that is, the pixel coordinates (i, j) and the corresponding measurements y_{ij} represent a set of points in 3D. The vector space $\mathcal{I} \subset \mathbf{R}^{\mathcal{N}}$ contains all possible range images $\mathbf{Y}$.

A eigenspace [2] is constructed from a set of m training views. Each training view $\vec{\mathbf{T}}_i$ is viewed as a column in the training matrix $\mathbf{X}$.

The $n \times n$ covariance matrix $\mathbf{Q} = \mathbf{X}\mathbf{X}^T$ of can, in principle, be decomposed into its eigenvalues and corresponding eigenvectors $\{(\lambda_i, \vec{\mathbf{e}}_i) | i = 1, \ldots n\}$ (in our case the image vectors $\vec{\mathbf{T}}$ are not subtracted by the mean so $\mathbf{Q}$ is the scatter matrix). Murakami and Kumar [4] observed that the rank of $\mathbf{Q}$ is the minimum of n and m and described a technique to obtain the eigenvectors of $\mathbf{Q}$ from the scatter matrix of $\mathbf{X}^T$. In either case the $(\lambda_i, \vec{\mathbf{e}}_i)$ can be obtained, The eigenspace $\mathcal{E}$ corresponding to $\mathbf{X}$ is simply the span of the $\vec{\mathbf{e}}_i$.

The m-dimensional prototype $\vec{\mathbf{g}}_i$ of a training image $\vec{\mathbf{T}}_i$ its image under the the linear transformation defined by the eigenvectors $\vec{\mathbf{e}}_i$:

$$\vec{\mathbf{g}}_i = \begin{pmatrix} \vec{\mathbf{e}}_1^T \\ \vdots \\ \vec{\mathbf{e}}_m^T \end{pmatrix} \vec{\mathbf{T}}_i$$

An m-dimensional prototype (or arbitrary m-

vector) $\vec{g}$ corresponds to a reconstructed image $\tilde{\mathbf{T}}$:

$$\tilde{\mathbf{T}} = \left(\vec{\mathbf{e}}_1, \ldots, \vec{\mathbf{e}}_M \right) \vec{\mathbf{g}}$$

The eigenvectors $\vec{\mathbf{e}}_i$ are the principal components of the subspace spanned by the training images $\{\mathbf{T}_j\}$, and the corresponding eigenvalues λ_i measure the variation along that direction in the eigenspace. In most situations the bulk of the total variation is captured in a small number of principal components. For that reason, as well as a desire for computational efficiency, it is common to select the $k \ll \min(m, n)$ eigenvectors corresponding to the k largest eigenvalues to form a k-dimensional eigenspace. This reduced-dimensionality eigenspace still admits reconstructions from arbitrary k-dimensional prototypes $\vec{\mathbf{g}}$.

In a multiple-object recognition context, training views of each object to be recognized must be available. Let $\vec{\mathbf{T}}_i^j$ be the ith training image (out of a total of m_j) of the j'th model. Murase and Nayar [2]. proposed the use of two types of eigenspaces for multiple-object recognition.

1. The universal eigenspace $\mathcal{U}$ is constructed from $\{\mathbf{T}_i^j\}$ for all i and j.

2. The object eigenspaces $\mathcal{O}_\mathrm{j}$, one per model, are constructed from the training images $\{\mathbf{T}_i^j\}$ of the specific model j.

Where the universal eigenspace $\mathcal{U}$ is used to discriminate between objects during recognition and the objects eigenspace $\mathcal{O}_\mathrm{j}$ is used to determine the pose of the object.

Appearance-based recognition techniques operate by projecting the input image (of an unidentified object known to be represented in the system via training views) into the k-dimensional eigenspace and identifying the most similar model. This identification is often performed using a nearest-neighbor search, although closest-manifold search techniques have also been developed [2],[3].

Trucco and Verri [8] have established the equivalence between this approach and squared-error template matching or correlation maximization. The key to success in this approach, then, is to employ a training set that captures the expected range of variations in imaging conditions.

The genesis of our project was an observation that imaging conditions for range sensors relate primarily to the rotational component of pose; by contrast, lighting is a major factor in intensity-based techniques. The number of training views (as well as their size) affects speed and memory requirements in the computation of eigenspaces and we saw an opportunity to capitalize on the lack of illumination variation by sampling the space of rotational 3D object poses more densely. Figure 1 shows the first eight eigenshape and the 100th eigenshape calculated from a database of 54 objects.

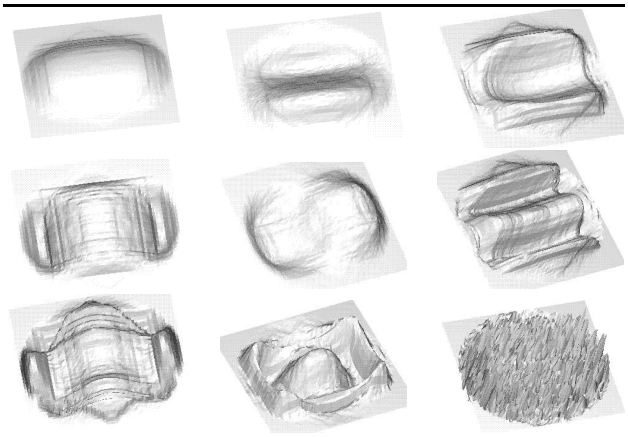

Figure 1: Examples of 'eigenshapes'

3 View Generation

The range images $\mathbf{Y}$ in this paper where synthetically generated from polyhedral mesh descriptions obtained from WSU [1] and rendered using an OpenGL based custom renderer. A set of rigid rotations was applied to the canonical model to obtain coverage of the 2D pose space (coverage of rotation about the 'optical' axis of the range sensor may not be necessary, as discussed below). For testing purposes we scaled the object to fit within a unit cube prior to rendering. The output of this step, for an input model j, is a set of m_j training views $\{\mathbf{T}_i^j, i = 1 \ldots m_j\}$.

Most work in appearance-based recognition has assumed a one-dimensional pose space, typically allowing objects to be rotated on a turntable in the sensor's field of view. We tessellated the surface of the 3D viewsphere to identify view angles with reasonably even angular spacing. The vertices (normalized to lie on the unit sphere) of the l-frequency subdivision of the icosahedron were used to define viewpoints around the object in its canonical position. Table 1 enumerates the number of viewpoints and the angle between view directions for a range of values of the subdivision parameter l. The choice of l allows the number of viewpoints (hence the angular spacing between view-

[1] `http://www.eecs.wsu.edu/ flynn/3DDB/Models/`

Frequency	Num of Viewpoints	Angle between
1	12	64.3°
2	42	$\approx 32.1^\circ$
3	92	$\approx 21.4^\circ$
4	162	$\approx 16.1^\circ$
5	252	$\approx 12.8^\circ$
6	362	$\approx 10.7^\circ$

Table 1: Icosahedron Subdivision Parameters

points) to be tuned by the user. A training image was obtained for each of these pose coordinates.

The rotation of an object to an arbitrary viewpoint (as accomplished above) will fix two degrees of freedom in its rotational pose. There still is an additional degree of freedom left in the rotation of the object about the viewpoint (or the 'optical axis' of the sensor). Rather than generate a set of views for each viewpoint, we developed a canonicalizing transformation (a planar rotation) to align the major and minor axis of the 2D 'footprint' of each view with the coordinate axes. This produces a canonical training image for that viewpoint. This approach works well if the object has significant elongation. Since zenith and nadir views of the object will produce mirror image footprints, we actually generate two training views per viewpoint, one being a mirror image of the other.

We will denote the number of viewpoints for a particular experiment as $N_{\rm vp}$. The value of l, hence the value of $N_{\rm vp}$, is fixed on an experiment by experiment basis, and is used to generate training views $\{\mathbf{T}_i^j : i = 1, \ldots, 2 \cdot N_{\rm vp}\}$ for each model j.

4 Object Recognition and Pose Determination

In appearance-based object recognition the training images $\mathbf{T}_i$ are projected into the subspace $\mathcal{E}$ along the eigenvectors $\vec{\mathbf{e}}_j$ to produce k-dimensional prototypes $\vec{\mathbf{g}}_i$. These coordinates can be used to generate a parametric manifold in the subspace (in our case, the relevant parameters are the two free pose parameters), and recognition would be implemented as identification of the closest manifold in the subspace to the prototype $\vec{g}^*$ generated by projection of an input image $\mathbf{T}^*$. In our initial matching procedure, however, we employ nearest-neighbor search to identify the object (Figure 2).

5 Experimental Results

The system has been tested on two object databases of 3D models. The first object database (database A) contains twenty 3D models (Figure 3). The database contains objects from human bones, mechanical parts (designed on a mechanical CAD package), to reverse-engineered 3D models constructed from range imagery.[2] All models are stored in a polyhedral mesh format approximating the 3D smooth surfaces. The second database (database B) contains 54 mechanical parts designed using CAD packages (Figure 4).

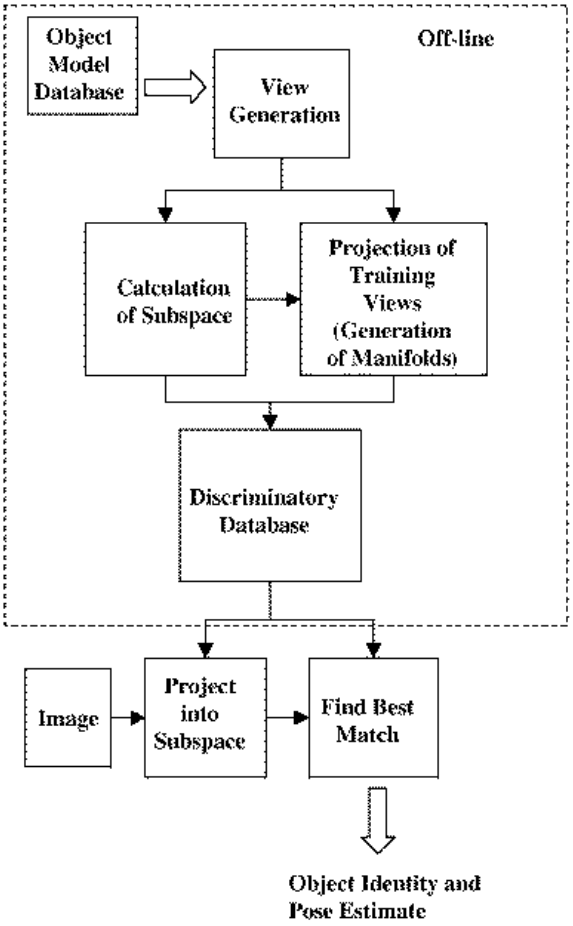

Figure 2: System Diagram for Paper

For each database a series of tests was run to determine the dependence of recognition accuracy on the number of viewpoints $N_{\rm vp}$, the size of the image n, and the dimension of the subspace k. A structured approach was used to generate test views. Since the training views were generated at the vertices of the

[2]We are grateful to Marc Soucy of Innovmetric Corp. and Marc Levoy of Stanford University for making their models available.

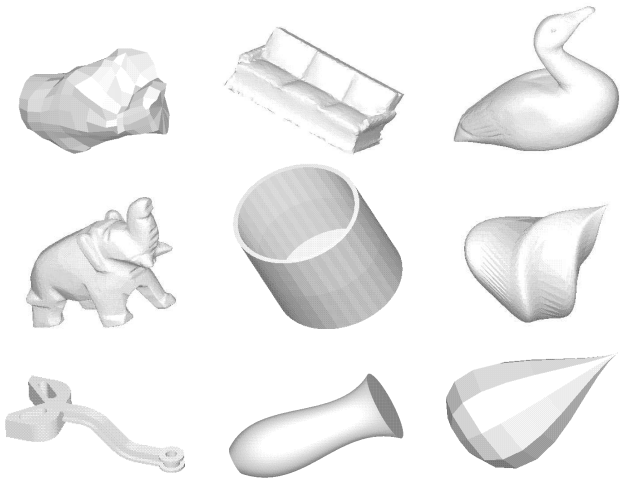

Figure 3: Some Examples of Objects From Database A

Figure 4: Some Examples of Objects From Database B

subdivided dodecahedron, we felt that viewpoints chosen as far as possible from the vertices would provide a worst-case test set. Therefore, test views were chosen from viewpoints corresponding to the centers of the triangular faces in the l-frequency subdivision of the icosahedron.

For database **A** eleven subspaces where generated for testing, corresponding to different values of N_{vp}, l, and n. The results of recognition rate *vs.* k are shown in Figure 5 for the various eigenspaces defined in Table 2.

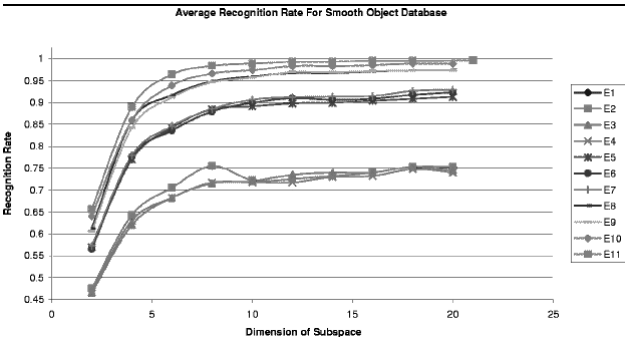

Figure 5: Recognition Rates for Database A

SubSpace	N_{vp}	n (image size)	Best Rate
$\mathcal{E}_1$	12	1024 (32x32)	75%
$\mathcal{E}_2$	12	4096 (64x64)	74.5%
$\mathcal{E}_3$	12	16384 (128x128)	75%
$\mathcal{E}_4$	12	65536 (256x256)	74.7%
$\mathcal{E}_5$	42	1024	91%
$\mathcal{E}_6$	42	4096	92.3%
$\mathcal{E}_7$	42	16384	92.8%
$\mathcal{E}_8$	92	1024	97.4%
$\mathcal{E}_9$	92	4096	97.3%
$\mathcal{E}_{10}$	162	1024	99%
$\mathcal{E}_{11}$	252	1024	99.7%

Table 2: Table of parameters used in generation of subspaces for Database **A**

For database **B** nine subpsaces where generated, but based on the results obtained from database **A** we violated the contition placed by most 'appearance-based' methods and allowed $n < m$. This allowed us to calculate 92, 162, and 252 viewpoint trials. The results of the trials are shown in Figure 6 for the eigenspaces defined in Table 3.

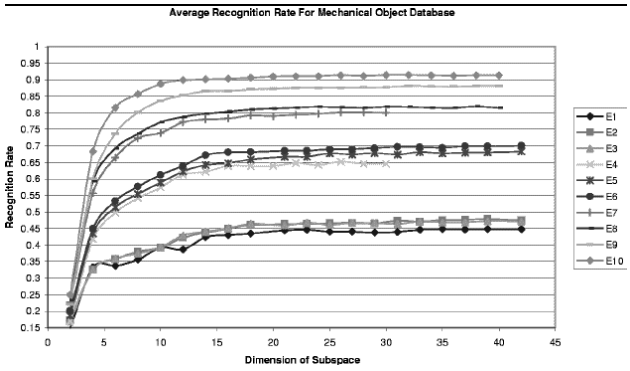

Figure 6: Recognition Rates for Database B

SubSpace	$N_{\rm vp}$	n (image size)	Best Rate
$\mathcal{E}_1$	12	1024 (32x32)	44.8 %
$\mathcal{E}_2$	12	4096 (64x64)	47.5%
$\mathcal{E}_3$	12	16384 (128x128)	47.1%
$\mathcal{E}_4$	42	256 (16x16)	65%
$\mathcal{E}_5$	42	1024	68.3%
$\mathcal{E}_6$	42	4096	70%
$\mathcal{E}_7$	92	246	80%
$\mathcal{E}_8$	92	1024	81%
$\mathcal{E}_9$	162	1024	88%
$\mathcal{E}_9$	252	1024	91%

Table 3: Table of parameters used in generation of subspaces for Database **B**

From the Figures 5,6 the most important parameters are the number of views $N_{\rm vp}$ and the dimension of the subspace K. For database **A** a subspace $\mathcal{E}$ with dimension greater than 13 does not improve the recognition rate. While the larger database **B** requires more than 20 eigensurfaces before the recognition rate approaches its max. A significant result found in both Figures 5 and 6, is that the size of the images n does not significantly change the recognition rate, even when an 16x16 image template is use the recognition rate only drops a few percent. Apparently, coarse shape matching is all that is needed to distinguish the objects in these databases.

The number of training views does have a dramatic effect on the recognition rate. For database A, increasing the number of training views from 12 to 92 increases the recognition rate from 75% to 97%. For database B, the rate increases from 47% to 80% when this change is made. For 252 views the eigenspace calculated for database **A** almost obtains perfect recognition, while database **B** only obtains a 91% recognition rate.

The larger database of mechanical objects is much harder to obtain high recognition rates. This is in part due to some of the objects similarity under certian viewing positions. In cases where a particular view of an object changes in shape appearance enough from its neighboring training views the view may be matched with another object of similar shape that is close to the gross shape of the object in that view. Figure 7 shows an example where the view to be recognized $\vec{\mathbf{T}}^*$ from object bigwye is incorrectly identified as another cylindrical object with handles (part 331c). In these particular views of the bigwye and 331c objects, the large discrimitory features have dissapeared and the neighboring views of the bigwye object are more dissimmalar than that of the 331c training view. This is due to the change of aspect caused by the appearance of the wye feature when the object is rotated 10 degrees to the neighboring training views where prototypes $\vec{\mathbf{g}}$ where generated.

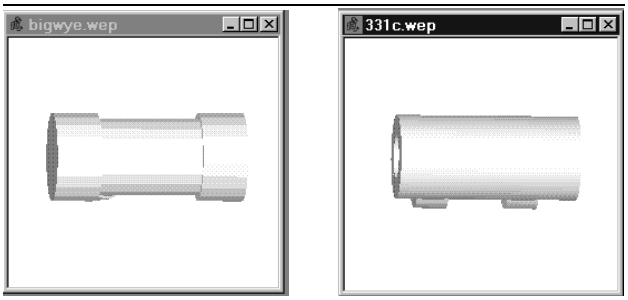

Figure 7: One Case of Mistaken Identity

In all these experiments, the angular sampling of the possible views is coarser than that used in current appearance-based methods. In Murase and Nayar's seminal work on 3-D object recognition [2] an angular separation of 4° was used in the 1D orientation space. They used a coarser sampling of 7.5° in their work on recognition of a 100 object database[3]. Since our system samples a 2D pose manifold embedded in 3D, a larger anguar spacing was needed because of memory and computation requirements. There are undoubtedly optimizations that can be made to increase the sampling frequency in the 2D pose space; such optimizations are the topic of current research.

6 Summary and Conclusions

In this paper we have shown the usefullness of using appearance-based approach to represent, recognize and determine pose of $2\frac{1}{2}$D shape data. Althought

this approach only used synthetic data it has shown some interesting properties in using appearance-based methods for range image object recognition under ideal conditions.

1. Image size does not dramaticly effect the ability to differentate between objects in the eigen subspace.
2. A Subspace with dimension $k \approx 20$ are sufficent to be close to the best recognition rates.
3. The smaller the angle between training views the greater the recognition rate.
4. The change of some aspects of an object can cause missclassification during recognition.
5. Representation is usefull for discriminating free-form objects as well as manufactured parts.
6. Using a simple 2D 'footprint' to align each view with the coordinate axes is a usefull way to resolve the third degree of freedom in the pose.

The results suggest that best sampling the views of an object will have to be adaptive if the number of views taken for each object is to be minimized. As a focus of future research a density of sampling needs to be determined that will produce accurate pose results, improve discrimitory power in regions of similarity and handle critical aspects where large surface changes occur over relativily small changes in view direction.

References

[1] M. Turk and A. Pentland, "Face Recognition Using Eigenfaces," *IEEE Conference on Computer Vision and Pattern Recognition,*, pp. 586-591, June 1991.

[2] H. Murase and S. K. Nayar, "Visual Learning and Recognition of 3-D Objects from Appearance," *Int'l Jour. of Computer Vision*, (14): 5-24, 1995.

[3] S. K. Nayar, S. A. Nene, and H. Murase, "Real-Time 100 Object Recognition System," *Image Understanding Workshop*, 1223-1227, 1996.

[4] H. Murakami and V. Kumar, "Efficient Calculation of Primary Images from a Set of Images," *IEEE Trans. Pattern Analysis and Machine Intelligence*, (4)5: 511-515, 1982.

[5] P. N. Belhumeur, J. P. Hespanha and D. J. Kriegman, "Eigenfaces vs. Fisherfaces: Recognition Using Class Specific Linear Projection," *IEEE Trans. Pattern Analysis and Machine Intelligence*, (19)7: 711-720, 1997.

[6] H. Bischof and A. Leonardis, "Robust Recognition of Scaled Eigenimages Throught a Hierarchical Approach," *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 664-670, June 1998.

[7] O. I. Camps, C. Huang and T. Kanungo, "Hierarchical Organization of Appearance-based Parts and Relations for Object Recognition," *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 685-691, June 1998.

[8] E. Trucco and A. Verri, *Introductory Techniques for 3-D Computer Vision,* Prentice Hall, 1998.

[9] A. E. Johnson and M. Hebert, "Efficient Multiple Model Recognition in Cluttered 3-D Scenes," *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 671-677, June 1998.

[10] M. J. Black and A. D. Jepson, "EigenTracking: Robust Matching and Tracking of Articulated Objects Using a View-Based Representation," *Int'l Jour. of Computer Vision*,(26)1:63-84, 1998.

[11] H. Murase and S. K. Nayar, "Illumination Planning for Object Recognition Using Parametric Eigenspaces," *IEEE Trans. Pattern Analysis and Machine Intelligence*, (16)12: 1219-1227, 1994.

[12] K. Ohba and K. Ikeuchi, "Detectability, Uniqueness, and Reliability of Eigen Windows for Stable Verification of Partially Occluded Objects," *IEEE Trans. Pattern Analysis and Machine Intelligence*, (19)9: 1043-1051, 1997.