PAPER

Construction of Appearance Manifold with Embedded View-Dependent Covariance Matrix for 3D Object Recognition

Lina[†a)], ***Student Member***, **Tomokazu TAKAHASHI**[†], **Ichiro IDE**[†], ***and*** **Hiroshi MURASE**[†], ***Members***

SUMMARY We propose the construction of an appearance manifold with embedded view-dependent covariance matrix to recognize 3D objects which are influenced by geometric distortions and quality degradation effects. The appearance manifold is used to capture the pose variability, while the covariance matrix is used to learn the distribution of samples for gaining noise-invariance. However, since the appearance of an object in the captured image is different for every different pose, the covariance matrix value is also different for every pose position. Therefore, it is important to embed view-dependent covariance matrices in the manifold of an object. We propose two models of constructing an appearance manifold with view-dependent covariance matrix, called the View-dependent Covariance matrix by training-Point Interpolation (VCPI) and View-dependent Covariance matrix by Eigenvector Interpolation (VCEI) methods. Here, the embedded view-dependent covariance matrix of the VCPI method is obtained by interpolating every training-points from one pose to other training-points in a consecutive pose. Meanwhile, in the VCEI method, the embedded view-dependent covariance matrix is obtained by interpolating only the eigenvectors and eigenvalues without considering the correspondences of each training image. As it embeds the covariance matrix in manifold, our view-dependent covariance matrix methods are robust to any pose changes and are also noise invariant. Our main goal is to construct a robust and efficient manifold with embedded view-dependent covariance matrix for recognizing objects from images which are influenced with various degradation effects.

key words: *3D object recognition, appearance manifold, view-dependent covariance matrix, eigenvector interpolation, eigenvalue interpolation, eigenspace*

1. Introduction

The appearance-based framework has provided a convenient base for investigating issues in object recognition for decades. Combined with eigenspace concept using Principal Component Analysis (PCA), it has been widely used in many recognition tasks, such as in object recognition and face recognition. Some of the earlier works in this domain include the application of characterizing the human face using PCA method in eigenpictures by Kirby and Sirovich [1], [2] and eigenfaces by Turk and Pentland [3]. Later, Wiskott et al. [4] pointed out a major disadvantage of PCA, that it could not capture even the simplest invariance unless this information is explicitly provided in the training data. While in a non-controlled environment, some variances of pose, illumination, occlusion, shifting, rotation, and so on, might occur and change the appearance of objects in captured images. Therefore, earlier methods in the eigenspace domain tend to fail when there are significant variations.

Addressing these variations problems, researchers improved the previous eigenpoint model with the use of an appearance manifold in the eigenspace. For years, various types of appearance manifolds have been developed, such as the simple appearance manifold in [5]–[7] which could handle pose and illumination variations, the appearance manifold with probabilistic techniques in [8]–[12] for handling various facial changes, the layer-transparent manifold in [13] for recognizing occluded objects, and other types of appearance manifolds which address different problems.

First, we focus on the work of Murase and Nayar, called the Parametric Eigenspace method [7], due to its simplicity and applicability to the general pose variation problems. However, although it can deal with pose and illumination changes, this model only works well with the assumption that there are no degradation effects. Unfortunately, this assumption is not realistic in real-world applications. In an image capturing process or segmentation process, some degradation effects usually occur and influence the original image. When some significant variations exist, the position of a non-degraded image and the image which is influenced with some degradation effects might be placed far from each other in the eigenspace. Thus, making the learning process rely on a simple manifold to capture image variations is not sufficient.

To overcome these limitations, we propose a novel method to construct the appearance manifold with embedded view-dependent covariance matrix. Here, covariance matrix is used to learn the samples distribution of each pose for gaining noise-invariance. However, since the appearance of an object in the captured image will be different for every different pose, the covariance matrix value will also be changed. Thus, our idea is, in order to accurately capture the distribution information, it is important to embed a view-dependent covariance matrix in the appearance manifold.

In our view-dependent covariance matrix methods, the mean vectors and covariance matrices are analyzed and have different values for each training pose. Further, to cover the untrained poses, we construct the appearance manifold by interpolating every training-points from one pose to the training-points in a consecutive pose in the VCPI method. Meanwhile, in the VCEI method, the view-dependent co-

Manuscript received July 19, 2007.
Manuscript revised December 10, 2007.
[†]The authors are with the Department of Media Science, Graduate School of Information Science, Nagoya University, Nagoya-shi, 464–8601 Japan.
a) E-mail: lina@murase.m.is.nagoya-u.ac.jp
DOI: 10.1093/ietisy/e91–d.4.1091

variance matrix for the untrained poses were obtained by interpolating the eigenvectors and eigenvalues. Here, it is not necessary to critically control the correspondences between every training image of each pose to a consecutive pose. Thus, besides its noise-invariant characteristic, the construction model with our View-dependent Covariance matrix by Eigenvector Interpolation (VCEI) method is also computationally efficient.

The remainder of this paper is organized as follows: we describe the feature extraction process using eigenspace representation in Sect. 2. Section 3 covers the idea and construction types of an appearance manifold with embedded covariance matrix, and also the sufficient classification technique. Section 4 presents the experimental setup, results, and performance analysis in various degradation conditions and various training-pose intervals. Section 5 covers the discussion of our view-dependent covariance matrix methods. Finally, Sect. 6 presents our conclusions.

2. Eigenspace Representation

Appearance-based approaches use sets of training images in various poses. These images are usually represented in a very high dimensional space, thus, processing them directly in the image space are computationally expensive. Here, PCA provides a technique to efficiently represent a collection of images by reducing their dimensionality.

Generally, the captured images should be normalized in brightness and scaled in order to be invariant to image magnification and illumination intensity. These normalized images can be written as a vector $\mathbf{x}$ by reading the number of pixels (N) in an image:

$$\mathbf{x} = [x_1, x_2, \ldots, x_N]^T \tag{1}$$

Let M be the number of the images in a training set. By subtracting the average image $\mathbf{c}$ of all images, the learning set $\mathbf{Y}$ will be obtained.

$$\mathbf{Y} = [\mathbf{x}_1 - \mathbf{c}, \mathbf{x}_2 - \mathbf{c}, \ldots, \mathbf{x}_M - \mathbf{c}] \tag{2}$$

Next, the auto-correlation matrix is defined by

$$\mathbf{Q} = \mathbf{Y}\mathbf{Y}^T \tag{3}$$

and the eigenvalues λ_i with its corresponding eigenvectors $\mathbf{e}_i$ are determined by solving the following eigenvector decomposition problem:

$$\lambda_i \mathbf{e}_i = \mathbf{Q}\mathbf{e}_i \tag{4}$$

In order to obtain the dimension reduction, simply ignore small eigenvalues and use only k corresponding eigenvectors with a threshold value T:

$$\sum_{i=1}^{k} \lambda_i \Big/ \sum_{i=1}^{N} \lambda_i \leq T \tag{5}$$

where $k \ll N$.

The first k eigenvectors will be used to project S training samples of P objects with H poses. s sample image of object p with horizontal viewpoint θ_h, $\mathbf{x}_s^{(p)}(\theta_h)$, are projected onto the eigenspace:

$$\mathbf{g}_s^{(p)}(\theta_h) = [\mathbf{e}_1, \mathbf{e}_2, \cdots, \mathbf{e}_k]^T (\mathbf{x}_s^{(p)}(\theta_h) - \mathbf{c}) \tag{6}$$

By projecting all the training samples onto the eigenspace, training features are represented efficiently as a set of discrete points in a low dimensional space.

3. Appearance Manifold with Embedded Covariance Matrix in Eigenspace

As we have stated earlier, our idea to overcome the problem of recognizing objects from images which are influenced with degradation effects is by taking into account the correlation of the data sets for gaining noise invariant in the appearance manifold.

Figure 1 shows the scheme of construction process of

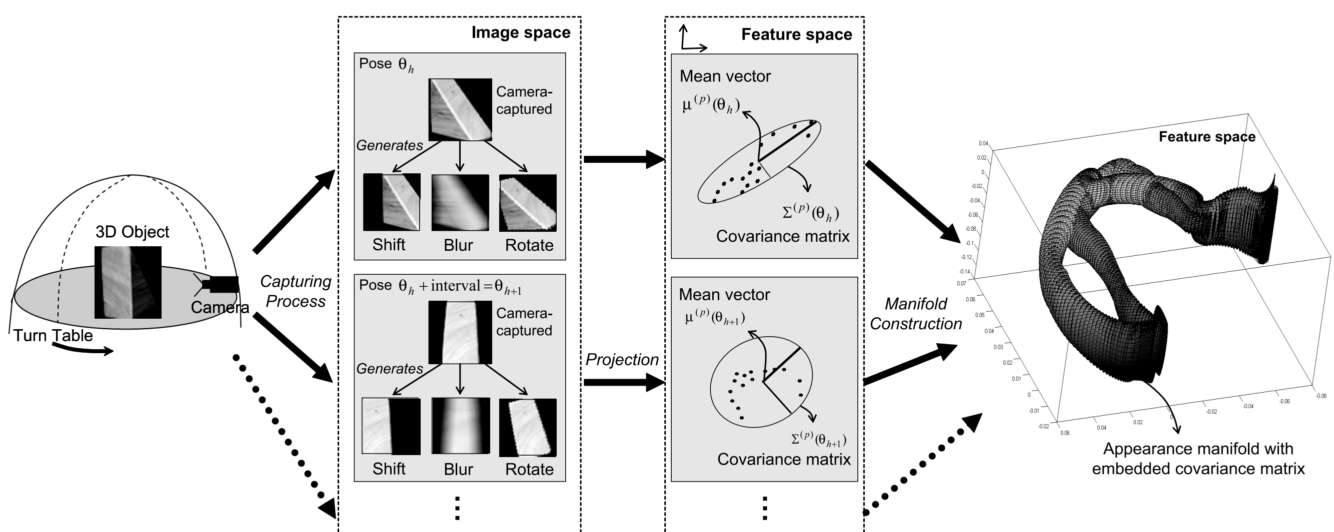


Fig. 1 Scheme of an appearance manifold with embedded covariance matrix.

an appearance manifold with embedded covariance matrix. In order to construct the appearance manifold with embedded covariance matrix, first, we train the system with images for each training pose. Note that for each pose, only a single camera-captured image is necessary. The other images could be obtained by generating artificial images from the original camera-captured image. Next, all the training samples are projected onto the feature space (eigenspace). Then, the mean vector and the covariance matrix are calculated to represent the center position of samples and sample-distribution for every training pose. Finally, to cover the unlearned poses, the appearance manifold is constructed by interpolating the mean vectors and covariance matrices. As it embeds the covariance matrix in the appearance manifold, the appearance manifold takes into account the correlations of the data set. Thus, it will be noise invariant.

In this section, we present various ways to construct the appearance manifold with embedded covariance matrix. We describe the idea of each construction type, explain the construction process for each method, and present the classification technique for the recognition step.

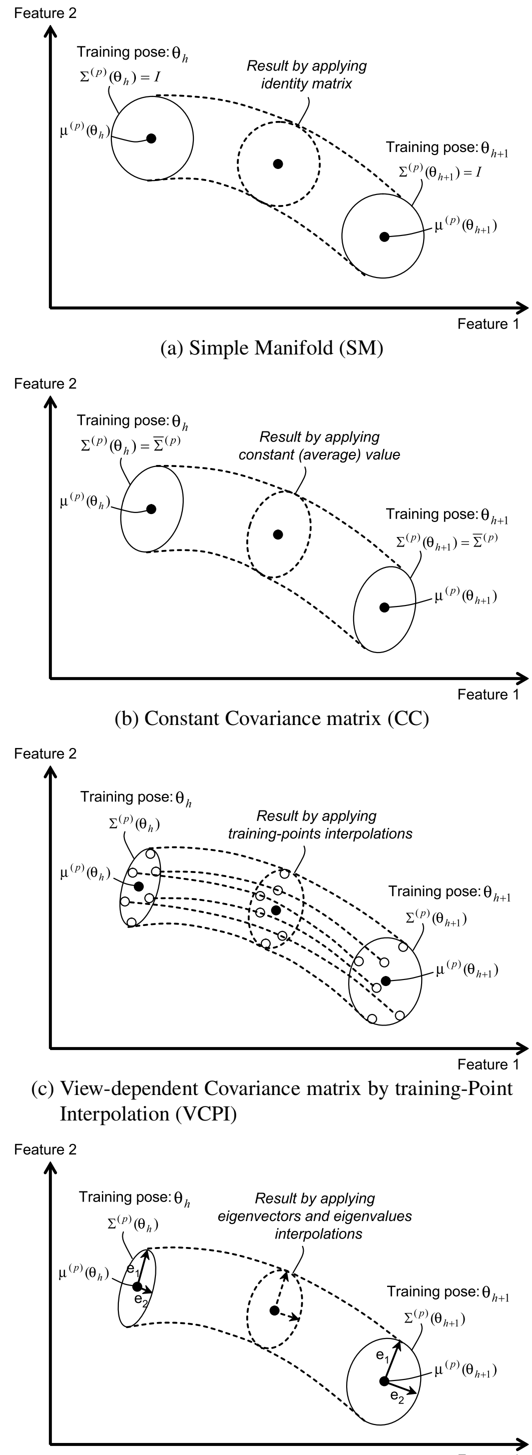
3.1 Various Types of Appearance Manifolds with Embedded Covariance Matrix

Here, we present the general idea for each construction type of an appearance manifold with embedded covariance matrix as shown in Fig. 2.

Simple Manifold (SM): Figure 2 (a) shows the construction type of an appearance manifold using the Simple Manifold (SM) method. This method constructs the manifold based on interpolation of mean vectors of samples, and applies identity matrix as the covariance matrix for each pose. Therefore, this manifold model relies only on the mean vectors, without considering the information of sample distributions. In case of using only one image sample in each pose, this method will be the same as Murase and Nayar's Parametric Eigenspace method (see [5]–[7]).

Constant Covariance matrix (CC): The construction model of Constant Covariance matrix (CC) method is depicted in Fig. 2 (b). Here, the appearance manifold is constructed by interpolating mean vectors and applying the same (average) value to all covariance matrices of every pose. Thus, this model has a constant value of covariance matrix for every pose in the manifold.

View-dependent Covariance matrix by training-Point Interpolation (VCPI): Figure 2 (c) shows the appearance manifold with training-Point Interpolation (VCPI) method. In the VCPI method, the appearance manifold is obtained by interpolating every training-points in each pose class to the training-points in a consecutive pose class that has the same characteristics, such as same degradation effects. Next, the new eigenpoints for every untrained pose class could be generated and their mean vectors and covariance matrices could be calculated. Here, the appearance



(a) Simple Manifold (SM)

(b) Constant Covariance matrix (CC)

(c) View-dependent Covariance matrix by training-Point Interpolation (VCPI)

(d) View-dependent Covariance matrix by Eigenvector Interpolation (VCEI)

Fig. 2 Construction types of an appearance manifold with embedded covariance matrix.

manifold will have different values of mean vector and covariance matrix for each pose.

View-dependent Covariance matrix by Eigenvector Interpolation (VCEI): Figure 2 (d) shows the construction model of View-dependent Covariance matrix by the Eigenvector Interpolation (VCEI) method. Here, in order to capture the density model of each pose more accurately, we propose a manifold model which has different values of mean vector and covariance matrix for each pose. Further, we propose a manifold construction method by eigenvectors and eigenvalues interpolation. Besides its accurateness, this model provides a practicable way of efficiently constructing the appearance manifold without critically controlling the correspondence of every training-samples in each pose to a consecutive pose.

3.2 Simple Manifold (SM) Method

In the Simple Manifold (SM) method, after projecting all training images onto the eigenspace, the mean vector $\boldsymbol{\mu}^{(p)}(\theta_h)$ and the covariance matrix $\boldsymbol{\Sigma}^{(p)}(\theta_h)$ for each object p for horizontal viewpoint θ_h are calculated. The mean vector is typically estimated by:

$$\boldsymbol{\mu}^{(p)}(\theta_h) = \frac{1}{S}\sum_{s=1}^{S}\mathbf{g}_s^{(p)}(\theta_h) \tag{7}$$

where S is the number of training samples from each class and $\mathbf{g}_s^{(p)}(\theta_h)$ is the image sample s from viewpoint θ_h and object p in the eigenspace.

To construct the appearance manifold, any interpolation method to the mean vector of two-consecutive poses may be applied and the value of identity matrix for its covariance matrix is given by:

$$\boldsymbol{\Sigma}^{(p)}(\theta_h) = I \tag{8}$$

3.3 Constant Covariance Matrix (CC) Method

In the Constant Covariance matrix (CC) method, the mean vector is calculated as in Eq. (7) and then an interpolation method is applied to obtain mean vectors for the untrained poses. While for the covariance matrix, covariance matrix for every trained pose is calculated and then its average value is applied to every covariance matrix in every pose in the manifold. The covariance matrix is typically estimated by:

$$\boldsymbol{\Sigma}^{(p)}(\theta_h) = \frac{1}{S}\sum_{s=1}^{S}(\mathbf{g}_s^{(p)}(\theta_h) - \boldsymbol{\mu}^{(p)}(\theta_h)) (\mathbf{g}_s^{(p)}(\theta_h) - \boldsymbol{\mu}^{(p)}(\theta_h))^T \tag{9}$$

and the average value of covariance matrix for all poses is calculated through:

$$\bar{\boldsymbol{\Sigma}}^{(p)} = \frac{1}{H}\sum_{h=1}^{H}\boldsymbol{\Sigma}^{(p)}(\theta_h) \tag{10}$$

Thus, by applying the average value of covariance matrix to every pose in the manifold, we will obtain:

$$\boldsymbol{\Sigma}^{(p)}(\theta_h) = \bar{\boldsymbol{\Sigma}}^{(p)} \tag{11}$$

3.4 View-Dependent Covariance Matrix by Training-Point Interpolation (VCPI) Method

In the View-dependent Covariance matrix by training-Point Interpolation (VCPI) method, an interpolation technique is applied to every pair of training-points of two consecutive pose classes which has the same characteristic. The equation for linearly interpolating two training-points from two consecutive training-poses $\mathbf{g}^{(p)}(\theta_h)$ and $\mathbf{g}^{(p)}(\theta_{h+1})$ is given by:

$$\mathbf{g}^{(p)}(\theta_h + \eta) = (1 - \eta)\mathbf{g}^{(p)}(\theta_h) + (\eta)\mathbf{g}^{(p)}(\theta_{h+1}) \tag{12}$$

where η is the fractional part which indicates how far the pose value changes from the original θ_h value.

After the interpolation process, the mean vectors and covariance matrices could be calculated using Eqs. (7) and (9), respectively. In this VCPI method, the appearance manifold will have different values of mean vector and covariance matrix for each pose.

3.5 View-Dependent Covariance Matrix by Eigenvector Interpolation (VCEI) Method

Here, we present the steps to construct the appearance manifold with embedded view-dependent covariance matrix. The View-dependent Covariance matrix by Eigenvector Interpolation (VCEI) method consists of two stages: the interpolation of mean vector and the interpolation of eigenvectors and eigenvalues. The mean vector represents the center of samples in each learning pose, while the eigenvectors and eigenvalues represent the covariance matrix as distribution of samples in each pose.

To cover the information of untrained poses, the interpolation processes for the mean vectors can be done by simply selecting one of several existing algorithms. Meanwhile, the interpolation of eigenvectors and eigenvalues are done based on the high-dimensional rotation theory. Here, the eigenvectors and eigenvalues can be considered as axes directions and lengths of a hyper-ellipsoid in an eigenspace. Thus, we consider to obtain the covariance matrices of untrained poses by rotating hyper-ellipsoids from every two-consecutive trained poses. Buja et al. in [14] also developed a similar mathematical base of high-dimensional rotation for interactive data visualization.

Here, we define three steps to interpolate the eigenvectors and eigenvalues:

[Step1] Correspond axes directions The process of checking axes directions between two eigenvectors from two-consecutive poses is necessary, since these axes directions will be used to define the rotation angle of hyper-ellipsoids. Let $\mathbf{E}_0$ and $\mathbf{E}_1$ be matrices formed by aligning each pair of eigenvectors $\mathbf{e}_{0j}$ and $\mathbf{e}_{1j}$ $(j = 1, 2, \cdots, k)$ of covariance matrices $\boldsymbol{\Sigma}_0$ and $\boldsymbol{\Sigma}_1$. The same process on eigenvalues should be done also by aligning eigenvalues λ_{0j} and λ_{1j} $(j = 1, 2, \cdots, k)$ to form k-dimensional vectors $\boldsymbol{\lambda}_0$ and $\boldsymbol{\lambda}_1$. To obtain the correspondences of axes, sort the eigenvectors of $\mathbf{E}_0$ and $\mathbf{E}_1$ based on their eigenvalues $\boldsymbol{\lambda}_0$ and $\boldsymbol{\lambda}_1$ to form $\mathbf{E}'_0$ and $\mathbf{E}'_1$, respectively. Next, perform the same task for the eigenvalues to form $\boldsymbol{\lambda}'_0$ and $\boldsymbol{\lambda}'_1$ from $\boldsymbol{\lambda}_0$ and $\boldsymbol{\lambda}_1$. Then, check the angle between the corresponded axes and invert the eigenvector $\mathbf{e}'_{1j}$ $(j = 1, 2, \cdots, k)$ if ${\mathbf{e}'_{0j}}^T \mathbf{e}'_{1j} < 0$ so that the angle between corresponded axes is less than or equal to $\pi/2$. For covariance matrix $\boldsymbol{\Sigma}_x$, the eigenvalues λ_{xj} $(j = 1, 2, \cdots, k)$ is simply calculated by

$$\lambda_{xj} = \left((1-x)\sqrt{\lambda'_{0j}} + x\sqrt{\lambda'_{1j}}\right)^2 \tag{13}$$

while, its eigenvectors $\mathbf{E}_x$ is calculated using a $k \times k$ rotation matrix by

$$\mathbf{E}_x = \mathbf{R}(x\boldsymbol{\phi})\mathbf{E}'_0 \tag{14}$$

$\mathbf{R}$ represents an interpolated rotation when $0 \leq x \leq 1$. Here, $\boldsymbol{\phi} = [\phi_1, \cdots, \phi_m]$ where $\boldsymbol{\phi}$ is the parameter vector from m numbers of rotation angles to define the rotation matrix $\mathbf{R}$.

[Step2] Determine the rotation matrix As the elements of $\mathbf{E}'_0$ and $\mathbf{E}'_1$ are orthogonal, thus, a rotation matrix could be defined by

$$\mathbf{R}(\boldsymbol{\phi}) = \mathbf{E}'_1 {\mathbf{E}'_0}^T \tag{15}$$

Refering to the Special Orthogonal (SO) rule, $\mathbf{R}(\boldsymbol{\phi})$ can be diagonalized with a $k \times k$ unitary matrix $\mathbf{U}$ and a diagonal matrix $\mathbf{D}$ including complex elements as

$$\mathbf{R}(\boldsymbol{\phi}) = \mathbf{U}\mathbf{D}(\boldsymbol{\phi})\mathbf{U}^\dagger \tag{16}$$

where $\mathbf{U}^\dagger$ represents a complex conjugate transpose matrix of $\mathbf{U}$. Furthermore, the result of the diagonalization process is a k-dimensional diagonal matrix $\mathbf{D}(\boldsymbol{\phi})$ which always has m pairs of complex conjugate roots $\mathbf{e}^{i\phi}$. Since each pair of complex conjugate root has a rotation angle, in order to check how many rotation angles we have in k-dimensional $\mathbf{D}(\boldsymbol{\phi})$, we calculate $m = \lfloor k/2 \rfloor$. If k is an even number $(k = 2m)$, then $\mathbf{D}(\boldsymbol{\phi}) = diag(\mathbf{e}^{i\phi_1}, \mathbf{e}^{-i\phi_1}, \cdots, \mathbf{e}^{i\phi_m}, \mathbf{e}^{-i\phi_m})$. However, if k is an odd number $(k = 2m + 1)$, then the first diagonal element of the complex conjugate roots is always 1. Thus, $\mathbf{D}(\boldsymbol{\phi}) = diag(1, \mathbf{e}^{i\phi_1}, \mathbf{e}^{-i\phi_1}, \cdots, \mathbf{e}^{i\phi_m}, \mathbf{e}^{-i\phi_m})$, where $\mathbf{e}^{i\phi} = \cos\phi + i\sin\phi$.

[Step3] Interpolate eigenvectors and eigenvalues Finally, $\mathbf{R}(x\boldsymbol{\theta})$ can be obtained simply by applying linear interpolation on the vectors. Next, $\boldsymbol{\Sigma}_x$ is calculated by

$$\boldsymbol{\Sigma}_x = \mathbf{E}_x \boldsymbol{\Lambda}_x {\mathbf{E}_x}^T \tag{17}$$

where $\boldsymbol{\Lambda}_x$ represents a diagonal matrix with λ_{xj} $(j = 1, 2, \cdots, n)$ as the diagonal elements.

3.6 Classification in Eigenspace

The Mahalanobis metric provides a sufficient way to classify images in terms of considering their related characteristics and likelihood in each pose class. Based on the correlations between variables, it becomes a useful way of determining similarity of an unknown sample to known sets.

In order to recognize an input image $\mathbf{u}$, it is first projected onto the eigenspace

$$\mathbf{z} = [\mathbf{e}_1, \mathbf{e}_2, \ldots, \mathbf{e}_k]^T(\mathbf{u} - \mathbf{c}) \tag{18}$$

Then, the distance $d^{(p)}(\mathbf{z})$ between the projected-image in the eigenspace $\mathbf{z}$ and the manifold of an object p is calculated as follows:

$$d^{(p)}(\mathbf{z}) = (\mathbf{z} - \boldsymbol{\mu}^{(p)}(\theta))^T \boldsymbol{\Sigma}^{(p)}(\theta)^{-1}(\mathbf{z} - \boldsymbol{\mu}^{(p)}(\theta)) \tag{19}$$

Finally, the input image $\mathbf{u}$ will be recognized as object p which has the minimum $d^{(p)}(\mathbf{z})$.

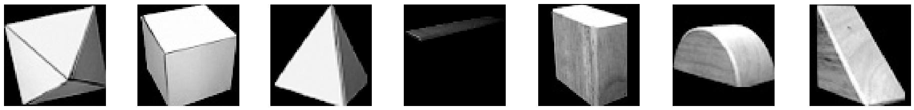
4. Experiments and Analysis

We implemented the view-dependent covariance matrix methods for an object recognition application. To evaluate the performance of our methods, we developed 3D object recognition systems in various conditions. The first condition was to recognize objects from non-degraded images, while other conditions were recognizing objects from images influenced with various degradation effects. In the experiments, the degradation types were motion blur, translation, and rotation effects. The system performance was also examined for various training-pose interval. The verification of the modelling result is given at the end of Sect. 5.

4.1 Experimental Setup

We design a series of experiments to evaluate the performance of the proposed methods. First, images of objects were captured using CCD camera, taken at pose intervals of one degree along the horizontal axis. This corresponds to 360 images per object. The images were then cropped and rescaled to 32×32 pixels of grayscale image with a uniform black background. Here, two datasets of objects were used in the experiments. Dataset 1 contains seven objects with block shapes, while Dataset 2 consists of ten objects with toy figures. The samples of objects used in the experiments are shown in Fig. 3.

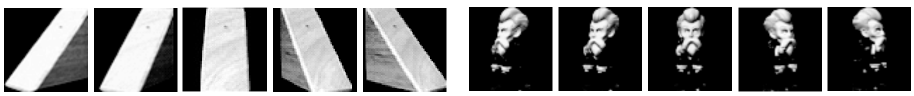
Further, there are three pose-interval sets used in the experiments. In the first set, the system was trained with a total of 6,552 images. Each object consists of 36 poses with 10 degree intervals of horizontal positions (0°, 10°, 20°, $\cdots$, 350°), and each pose consists of 26 training images with an original camera-captured image and 25 generated images with various degradation effects. The generated images were obtained by composing artificial noises, such as left and right translations (3, 6, 9, 12, and 15 pixels), clockwise and counter-clockwise rotations (5°, 10°, 15°, 20°, and



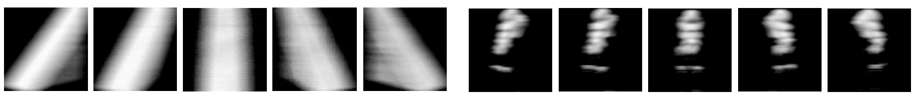
(a) Dataset 1

(b) Dataset 2

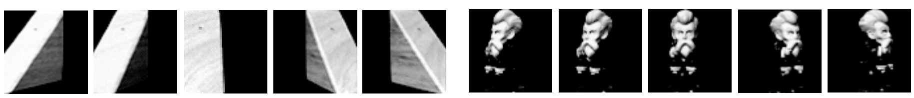
Fig. 3 Samples of objects used in the experiments.



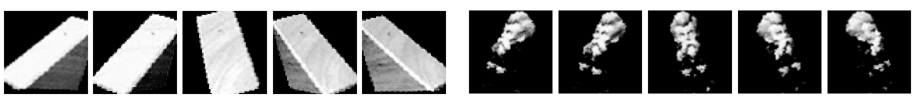
(a) Camera-captured images



(b) Images with blur effects



(c) Images with translation effects



(d) Images with rotation effects

Fig. 4 Samples of training images of objects.

25°), and motion blur (5%, 10%, 15%, 20%, 25%). These artificial noises were generated using MATLAB functions. As the second set, for 30 degree training-pose intervals (0°, 30°, 60°, ···, 330°), each object has 12 training poses and 2,184 images were used as training images. For the third set, the 60 degree training-pose interval (0°, 60°, 120°, ···, 300°), there were 6 poses trained for each object. Thus, the total number of training images in this dataset was 1,092 images.

The samples of training images of an object in each dataset are shown in Fig. 4. The camera-captured images are shown in Fig. 4 (a). While Fig. 4 (b) shows the images with blur effects, Fig. 4 (c) shows the images which are influenced with translation (shift) effects, and Fig. 4 (d) are images with rotation effects.

Next, features were extracted using PCA and were projected onto the eigenspace. The appearance manifolds were then created based on each construction method. The interpolation method was uniformed by using the linear interpolation method for interpolating mean vectors, while the covariance matrices were obtained according to the construction methods explained in Sect. 3.

Here, we evaluated two methods for constructing the appearance manifold based on the Simple Manifold (SM) model: the Simple Manifold with Non-Degraded center (SMND) and Simple Manifold with Mean center (SMM) methods. In the SMND method, the center of the samples distribution is based on the non-degraded (original camera-captured) image. Meanwhile, in the SMM method, the center of the sample distribution is based on the mean vector of image samples in each pose. For both the SMND and SMM methods, an identity matrix was applied to all the covariance matrices in the manifold. For the Constant Covariance matrix (CC) method, the mean vector of image samples become the center of the samples distribution for each pose and the average value of all covariance matrices was applied to every covariance matrices along the manifold. For the View-dependent Covariance matrix by training-Point Interpolation (VCPI) and the View-dependent Covariance matrix by Eigenvector Interpolation (VCEI) methods, the center of the samples distribution is the mean vector of image samples of each pose. Each pose in the appearance manifold also has a different covariance matrix. However, in the VCPI method, the covariance matrices for the untrained poses were obtained by interpolating each training-point of each pose to a consecutive training-pose. Meanwhile, in the VCEI method, the covariance matrices of untrained poses were obtained by interpolating only the eigenvectors and eigenvalues of two consecutive training-poses, as explained in Sect. 3.5.

Finally, we tested the system with input images which were different from the learning images in horizontal poses

(5°, 15°, 25°, ···, 355°). We also conducted experiments to observe the degradation in performance of the proposed approaches when the images were influenced with various types of degradation effects. For classification, we applied the Mahalanobis distance measurement.

We compared the performance of the proposed view-dependent covariance matrix methods (the VCPI and VCEI methods) with that of the Simple Manifold with Non-Degraded center (SMND) method, the Simple Manifold with Mean center (SMM) method, and the Constant Covariance matrix (CC) method. The performances were evaluated on non-degraded input-image condition, various degradation conditions, and various training-pose interval.

4.2 Overall Performance in Various Degradation Conditions

Table 1 presents the average recognition accuracies for the SMND, SMM, CC, VCPI, and VCEI methods for two datasets in two ranges of degradation conditions. The ranges of the degradation conditions are (0-10)% and (0-25)%.

The (0-10)% range means that the testing images were influenced with various degradation effects (translation, rotation, blur) within the range of 0% up to 10% of image size. Meanwhile, the second range, (0-25)% range means that the testing images were influenced with translation, rotation, and blur effects within the range of 0% up to 25% of image size. Thus, the testing images in the second range were influenced with 0 pixel (no translation) up to 9 pixels of translation effects, 0° (no rotation) up to 25° of rotation effects, and 0% (no blur) up to 25% of blur effects.

For recognizing objects in Dataset 1 which consists of various block shapes, for (0-10)% range, both the VCPI and VCEI methods achieved higher recognition rates compared with that of the CC method, and the simple manifold methods (the SMND and SMM methods). The VCPI method achieved 90.04% and the VCEI method achieved 89.20%, while the CC, SMND, and SMM methods only achieved 76.41%, 73.68%, and 73.54%, respectively. Next, in the (0-25)% range case, where the degradation effects became more severe, the recognition accuracies of all methods decreased along with the increment level of the degradation effects. However, both the VCPI and VCEI methods could maintain their superiority.

In the next experiment, for recognizing toy figure objects in Dataset 2 with (0-10)% range, the VCPI and VCEI methods also achieved higher recognition rates compared with that of the CC, SMND, and SMM methods. The VCPI method achieved 98.93% and the VCEI method achieved 98.77%. Meanwhile, the CC, SMND, and SMM methods only achieved 77.22%, 58.70%, and 69.35%, respectively. The VCPI and VCEI methods also successfully performed their robustness upon the increment level of the degradation effects. It is shown in the next (0-25)% range case, where the VCPI method still achieved 98.92% and the VCEI method achieved 98.62% of recognition accuracy.

In overall, as shown in Table 1, the proposed view-dependent covariance matrix methods (the VCPI and VCEI methods) could outperform the CC method and simple manifold methods (the SMND and SMM methods) in recognizing different types of objects in various degradation conditions for both datasets. In the next sections, we take the case of Dataset 1 in order to present more detailed analysis on the recognition results of the appearance manifold methods.

4.3 Performance in Non-degraded Input-Image Condition

Table 2 shows a comparison of recognition accuracies of the simple manifold methods (the SMND and SMM methods), Constant Covariance matrix (CC), and view-dependent covariance matrix methods (the VCPI and VCEI methods) when recognizing objects in Dataset 1 from images with non-degraded effects. These results show that for all training-pose intervals, shown in Table 2, the SMND, VCPI, and VCEI methods gave better performances compared with the other two methods in a non-degraded input-image condition. In this experiment, since the input images were not influenced with the degradation effects, the input images are

Table 1 Results for recognizing objects of two datasets from images with various ranges of degradation effects.

Methods	Dataset 1 (0-10)%	Dataset 2 (0-25)%	Dataset 1 (0-10)%	Dataset 2 (0-25)%
Simple Manifold with Non-Degraded center (SMND)	73.68	58.70	55.25	41.50
Simple Manifold with Mean center (SMM)	73.54	69.35	56.78	53.50
Constant Covariance matrix (CC)	76.41	77.22	60.91	64.32
View-dependent Covariance matrix by training-Point Interpolation (VCPI)	90.04	98.93	82.23	98.92
View-dependent Covariance matrix by Eigenvector Interpolation (VCEI)	89.20	98.77	79.28	98.62

Table 2 Results for recognizing objects in Dataset 1 from images with non-degraded effects.

Methods	10° pose-interval	30° pose-interval	60° pose-interval
Simple Manifold with Non-Degraded center (SMND)	99.60	94.05	81.75
Simple Manifold with Mean center (SMM)	76.19	67.86	66.67
Constant Covariance matrix (CC)	79.37	75.40	77.38
View-dependent Covariance matrix by training-Point Interpolation (VCPI)	94.05	88.89	82.54
View-dependent Covariance matrix by Eigenvector Interpolation (VCEI)	92.06	85.71	82.54

likely to be similar to the original camera-captured images. Thus, the SMND method which use the non-degraded image as the center of sample distribution could achieve good performance in recognizing objects from non-degraded images, especially in 10° and 30° training-pose interval. However, when a wider training-pose interval is used (60°), the proposed view-dependent covariance matrix methods (the VCPI and VCEI methods) could outperform the SMND method.

4.4 Performance in Various Degradation Conditions

To evaluate the performance of the proposed view-dependent covariance matrix methods in various degradation conditions, we conducted several experiments. As shown in Fig. 5, the appearance manifold methods were applied to recognize images in Dataset 1 which are influenced with motion blur effects, translation effects, and rotation effects. For motion blur effects, the VCPI and VCEI methods always give higher recognition accuracies compared with that of the CC and SMM methods. Although the SMND method showed the highest recognition accuracies for 5% and 10% blur effects, when we increased the level of the blur effects to 15%, 20% and 25%, the proposed VCPI and VCEI methods could outperform the SMND method. Next, for recognizing objects with various left and right translation effects, the proposed VCPI and VCEI methods always give higher recognition accuracies than the SMND, SMM, and CC methods. Finally, the same trend also appeared in recognizing objects with various rotation effects, where the proposed VCPI and VCEI methods could outperform the SMND, SMM, and CC methods.

Figure 5 indicates that the proposed VCPI and VCEI methods always achieve higher recognition accuracies compared with the SMND, SMM, and CC methods for various degradation conditions. It also shows the robustness of the VCPI and VCEI methods to various degradation conditions, especially for motion blur effects and rotation effects.

4.5 Performance in Various Training-Pose Intervals

We also conducted experiments to observe the degradation performance of the proposed view-dependent covariance matrix methods in various training-pose intervals and influenced with various degradation effects. The system was trained with 10°, 30°, and 60° intervals of horizontal positions with the same conditions as explained in the experimental setup in Sect. 4.1.

Figure 6 shows the recognition accuracies of the proposed VCEI method in various training-pose intervals with various motion blur, translations, and rotation effects. For the motion blur case, the decrement value of recognition accuracies from 10° training-pose interval to 30° training-pose interval was 5.32% in average. While higher decrement value of recognition accuracies of 7.06% occurred when the training-pose interval was changed from 30° to 60°. In the case of translation effects, the decrement value of recognition accuracies from 10° training-pose interval to 30° training-pose interval was 5.52% in average and 4.80% for changing the training-pose interval from 30° to 60°. Finally, for rotation effects, the decrement value of recognition accuracies from 10° to 30° training-pose interval and from 30° to 60° training-pose interval were relatively small; 4.56% and 2.74%, respectively. In overall, the recognition accuracy of the proposed VCEI method only decreased 5.13% when 30° wider training-pose interval was used.

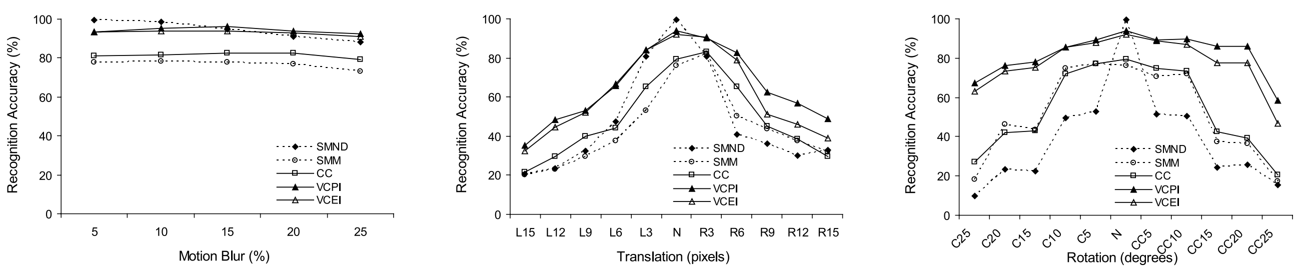


Fig. 5 Results for recognizing objects in Dataset 1 from images with various degradation effects.

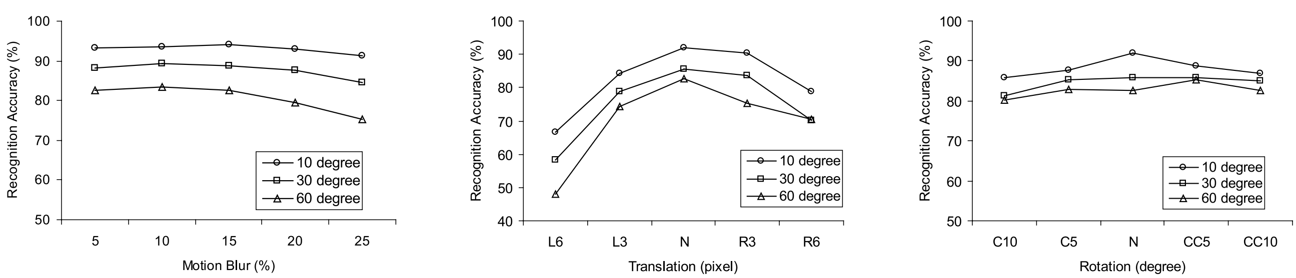
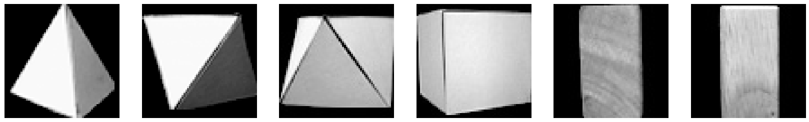
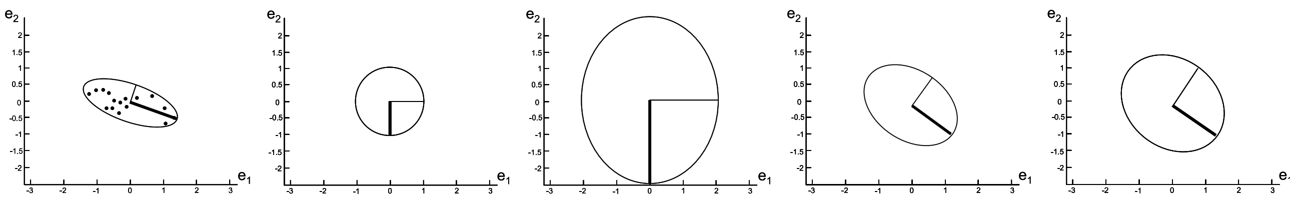


Fig. 6 Recognition results of Dataset 1 of View-dependent Covariance matrix by the Eigenvector Interpolation (VCEI) method in various training-pose intervals and degradation conditions.



(a) (b) (c) (d) (e) (f)

Fig. 7 Samples of cases where the Simple Manifold with Mean center (SMM) method failed but View-dependent Covariance matrix by Eigenvector Interpolation (VCEI) method succeed.



(a) Ground truth (b) SMND and SMM (c) CC (d) VCPI (e) VCEI

Fig. 8 Verification results of various methods.

5. Discussion

From the experimental results, we have proved that the idea to embed view-dependent covariance matrix in an appearance manifold worked well to overcome the problem of recognizing objects from images which are influenced with degradation effects. Here, we have presented novel manifold construction models, called the View-dependent Covariance matrix by training-Point Interpolation (VCPI) and View-dependent Covariance matrix by Eigenvector Interpolation (VCEI) methods. The VCEI method constructs the manifold by using only the eigenvectors and eigenvalues without preserving the information on how to build the correspondence of every training image in each pose. This method outperformed the efficiency of the VCPI method which needs to interpolate each training image in every trained-pose to its consecutive poses.

We also observed that the SMND method only worked well in recognizing objects from non-degraded images (as shown in Table 2). Meanwhile, the SMM method gave good recognition results only for objects having distinct appearance and major differences in shape characteristics. However, the proposed view-dependent covariance matrix methods could recognize objects with a similar appearance. Figure 7 shows the set of objects for which the SMM method failed but the VCEI method succeeded. Using the SMM method, the object in Fig. 7 (a) was recognized as the object in Fig. 7 (b), since it had similar color appearance. Another failed case in the SMM method, but succeed in the VCEI method is that the object in Fig. 7 (c) was recognized as the object in Fig. 7 (d) due to the similar size. Also, the object in Fig. 7 (e) was recognized incorrectly as the object in Fig. 7 (f) since it had a similar shape. These results show that the proposed VCEI method, which considers sample distribution information captured in the view-dependent covariance matrix, has a high capability in recognizing similar objects both in non-degraded condition and when influenced with degradation effects.

Finally, verification results are shown in Fig. 8, which shows the construction of covariance matrix along with its first and second eigenvectors directions. Here, in order to simplify the visualization, we draw the covariance matrix construction in 2D figures. The covariance matrix constructions in Fig. 8 were obtained by slicing the appearance manifold of each method on an untrained 45° viewpoint, where each appearance manifold was constructed from 0° and 90° viewpoints. We intentionally set this condition in order to emphasize the difficulty in modelling of interpolation results from two extremely different learning viewpoints.

Figure 8 (a) shows the ground truth of a covariance matrix construction, obtained from real image projections, while Figs. 8 (b), (c), (d), and (e) show the construction results of covariance matrix from the SM, CC, VCPI, and VCEI methods, respectively. Figure 8 (d) of the proposed VCPI method presents the most similar construction result of the covariance matrix with the result of real image projections depicted in Fig. 8 (a). The VCEI method then followed in the second most similar construction result of the covariance matrix with Fig. 8 (e). Meanwhile, the CC method with its construction result depicted in Fig. 8 (c) and the SM method in Fig. 8 (b) were less similar in shape and direction from the ground truth in Fig. 8 (a). These results confirm their weak recognition capability compared with the proposed view-dependent covariance matrix methods (the VCPI and VCEI methods).

6. Conclusion

We have presented novel methods for constructing an appearance manifold with embedded view-dependent covariance matrix. First, the View-dependent Covariance matrix by training-Point Interpolation (VCPI) method constructs the appearance manifold by interpolating every training-point from one pose class to the training-points in a consecutive pose class. Meanwhile, the View-dependent Covariance matrix by Eigenvector Interpolation (VCEI) method is based on the eigenvalues and eigenvectors interpolations to form a view-dependent covariance matrix model. The proposed view-dependent covariance matrix methods have been implemented in a 3D object recognition system to recognize 3D objects from images that are influenced by geometric distortions such as translation and rotation, and quality-degradation (motion blur) effects. Experimental results proved that the VCPI and VCEI methods are superior to the other manifold construction types, such as the Constant Covariance matrix (CC) and simple manifold (the SMND and SMM) methods. Their performances also seemed to be consistent even when some degradation factors exist, such as images influenced with different types of degradation effects and when lesser numbers of training images were used. Here, the advantage of the VCEI method is that, since it obtains the view-dependent covariance matrix by interpolating only the eigenvectors and eigenvalues, it is computationally more efficient compared with the VCPI method which has to correspond every training-image in each pose to their consecutive poses. Our future work will concentrate on recognizing objects under more severe conditions, such as more complex poses and more similar shapes.

References

[1] L. Sirovich and M. Kirby, "A low-dimensional procedure for the characterization of human faces," J. Optical Society of America A, vol.4, no.3, pp.519–524, March 1987.
[2] M. Kirby and L. Sirovich, "Application of the Karhunen-Loeve procedure for the characterization of human face," IEEE Trans. Pattern Anal. Mach. Intell., vol.12, no.1, pp.103–108, Dec. 1990.
[3] M. Turk and A. Pentland, "Face recognition using eigenfaces," Proc. IEEE Conf. on Computer Vision and Pattern Recognition, pp.586–591, Maui HI, USA, June 1991.
[4] L. Wiskott, J.M. Fellous, N. Kruger, and C. von der Malsburg, "Face recognition by elastic bunch graph matching," IEEE Trans. Pattern Anal. Mach. Intell., vol.19, no.7, pp.775–779, July 1997.
[5] H. Murase and S.K. Nayar, "Visual learning and recognition of 3-D objects from appearance," Int. J. Comput. Vis., vol.14, no.1, pp.5–24, Jan. 1995.
[6] S.K. Nayar, S.A. Nene, and H. Murase, "Real-time 100 object recognition system," Proc. IEEE Conf. on Robotics and Automation, pp.2321–2325, Minneapolis MN, USA, April 1996.
[7] H. Murase and S.K. Nayar, "Illumination planning for object recognition using parametric eigenspaces," IEEE Trans. Pattern Anal. Mach. Intell., vol.16, no.12, pp.1219–1227, Dec. 1994.
[8] B. Moghaddam and A. Pentland, "Probabilistic visual learning for object representation," IEEE Trans. Pattern Anal. Mach. Intell., vol.19, no.7, pp.696–710, July 1997.
[9] Y. Chang, C. Hu, and M. Turk, "Manifold of facial expression," Proc. IEEE Int. Workshop on Analysis and Modeling of Faces and Gestures, pp.28–35, Nice, France, Oct. 2003.
[10] A.M. Martinez, "Recognition of partially occluded and/or imprecisely localized faces using a probabilistic approach," Proc. IEEE Conf. on Computer Vision and Pattern Recognition, vol.1, pp.712–717, Hilton Head Island SC, USA, June 2000.
[11] Y. Chang, C. Hu, and M. Turk, "Probabilistic expression analysis on manifolds," Proc. IEEE Conf. on Computer Vision and Pattern Recognition, vol.2, pp.520–527, Washington DC, USA, July 2004.
[12] C.M. Christoudias and T. Darrel, "On modeling nonlinear shape-and-texture appearance manifolds," Proc. IEEE Conf. on Computer Vision and Pattern Recognition, vol.2, pp.1067–1074, San Diego CA, USA, June 2005.
[13] B.J. Frey, M. Jojic, and A. Kannan, "Learning appearance and transparency manifolds of occluded objects in layers," Proc. IEEE Conf. on Computer Vision and Pattern Recognition, vol.1, pp.45–52, Madison WI, USA, June 2003.
[14] A. Buja, D. Cook, D. Asimov, and C. Hurley, "Computational methods for high-dimensional rotations in data visualization," in Handbook of Statistics: Data Mining and Data Visualization, ed. C.R. Rao, E.J. Wegman, and J.L. Solka, pp.391–412, North Holland, Amsterdam, 2005.
[15] Lina, T. Takahashi, I. Ide, and H. Murase, "Appearance manifold with embedded covariance matrix for robust 3D object recognition," Proc. IAPR Conf. on Machine Vision Applications 2007, pp.504–507, Tokyo, Japan, May 2007.

Lina received her B.S. degree (2001) from the Department of Computer Science, Tarumanagara University and a M.S. degree (2004) from the Department of Computer Science, University of Indonesia. Currently, she is pursuing a Ph.D. in the Department of Media Science at Nagoya University, Japan. Her research interests include 3D object and face recognition.

Tomokazu Takahashi received his B.S. degree from the Department of Information Engineering at Ibaraki University, and a M.S. and Ph.D. from the Graduate School of Science and Engineering at Ibaraki University. His research interests include computer graphics and image recognition.

Ichiro Ide received his B.S. degree from the Deparment of Electronic Engineering, a M.Eng. degree from the Department of Information Engineering, and a Ph.D. from the Department of Electrical Engineering at the University of Tokyo. He is currently an Associate Professor in the Graduate School of Information Science at Nagoya University. His research interests include integrated media processing and video processing.

Hiroshi Murase received his B.S., M.S., and Ph.D. degrees from the Graduate School of Electrical Engineering at Nagoya University. He is currently a Professor in the Graduate School of Information Science at Nagoya University. He received the Ministry Award from the Ministry of Education, Culture, Sports, Science and Technology in Japan in 2003. He is a Fellow of the IEEE. His research interests include image recognition, intelligent vehicle, and computer vision.