An Appearance Based Fast Linear Pose Estimation

Toshiyuki Amano
Graduate School of Information Science,
Nara Institute of Science and Technology
amano@is.naist.jp

Toru Tamaki
Graduate School of Engineering
Hiroshima University
tamaki@hiroshima-u.ac.jp

Abstract

In this paper, we describe a high speed and multi degree of freedoms (DOFs) pose estimation method for a 3-D object that called Estimation-by-Completion (EbC) method. The most of employed processes are described with linear calculation, thus, whole procedure for each parameter estimation is expressed by a pair inner production, and it used only an arctangent calculation at the final part of the estimation. The accuracy evaluation by 3DOFs pose estimation that includes rotation around the object's vertical and horizontal translations is shown in experiment. We also describe its calculation cost in discussions.

1 Introduction

The posture and position estimation problem of 3-D object from 2-D monocular image are the most elementally but important computer vision problem. For this problem, many solutions were proposed and these methods are classified by two methodology that model based and the appearance based. The appearance based method that famous for parametric eigenspace method[6, 8] is convenient for application and easy to learn for recognition, because that method needs no geometrical models. Therefore, the appearance based method is a promising approach for complex shape object recognition. The parametric eigenspace method is the method that used the eigenface[12] of the computer vision problem of human face recognition for continuous variation image sequences by the rotation of the object and illumination direction. Since the parametric eigenspace method is convenient and well reproducible, it applied for moving object recognition[7], visual servo[4], illumination planning[6], shape from shading, etc. However, the appearance based method has a problem that vast learning samples are necessary by DOFs of the object movement and illumination variations.

Amano et al[2] proposed a solution to reduce learning samples by using range image but still range sensor is not popular in the general applications. However, at the applications for factory automation or visual navigation, only estimation or recognition process needs fast computation. Therefore, if it is acceptable computation cost, to learn in the short time is unnecessary, because the eigenspace method learns beforehand. The pose estimation method based on a linear model that was proposed by Okatani et al[10]. is the method which realized this idea. In this method, the parameters such as postures are estimated by the inner product of input image and coefficients that calculated by generalized inverse matrix and the computational cost of parameter estimation is dramatically reduced.

In this research, we describe the EbC (*Estimation-by-Completion*) method that is a posture estimation method based on eigenspace feature extraction and image interpolation by the new approach and different from Okatani's method. This method attaches information tracks at the bottom of learning sample image and generates eigenvectors. In the parameter estimation process, the EbC method estimates posture parameters by the decoding of the information tracks that attached at the bottom of an image. However, the information tracks are missing in the input image acquired with camera or given for parameter estimation. Therefore, BPLP (*Back Projection for Lost Pixels*) method that is image interpolation method based on learning is used for the information track restoration of an input image, and proposed method decodes parameters by the inner production of basis vectors.

Since all these processes are linear operation, so we can integrate and express these calculations to image pairs (*EbC Image Pairs*). Therefore, we can estimate parameters by correlation calculation of the image pair and arctangent calculation for each parameter.

2 Estimation by the restoration

The EbC method uses interpolation technique BPLP method[1] that based on the learning and restores posture information track that attached to the bottom lines of the learning image.

2.1 Overview of the BPLP method

The BPLP method uses eigenspace to perform interpolation. For the learning, we have learning sample images that expressed feature of the target for ready, and we express these learning samples as N-dimensional image vectors by the raster scanning of the image.

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

where $N = w \times h$ is the number of pixels of each sample image, w and h are the width and height of the sample image. Eigen vectors of the learning samples $\{\boldsymbol{\chi}_1, \boldsymbol{\chi}_2, \ldots, \boldsymbol{\chi}_M\}$ are given by SVD, etc., and sorted by the eigen value and used D largest eigen vectors:

$$E = [\boldsymbol{e}_1, \boldsymbol{e}_2, \ldots, \boldsymbol{e}_D]. \tag{2}$$

When an input image $\boldsymbol{\chi}'$ which some pixel values are missing was given, we express missing parts by $N \times N$ diagonal matrix Σ,

$$\Sigma = \mathrm{diag}(1, 1, \ldots, 0, \ldots, 1), \tag{3}$$

where that element's values are being set on 0 or 1, if the corresponding pixel values are missing or existing. The input image $\boldsymbol{\chi}'$ that includes missing elements can be expressed with

$$\boldsymbol{\chi}' = \Sigma\boldsymbol{\chi} \tag{4}$$

by the matrix Σ, if we set the missing element value as 0. From the projection points relation of the lacked image and non-lacked image, missing elements are estimated and we get an estimated image

$$\hat{\boldsymbol{\chi}} = E(E^T\Sigma E)^{-1}E^T\boldsymbol{\chi}'. \tag{5}$$

2.2 Information Track Restoration

In this paper, we assume the object that forms a smooth manifold by the continuous object's parameter change on the image vector space as well as parametric eigenspace method. Let $\boldsymbol{\chi}_i$ denote learning samples that sampled images densely to approximates manifold shape on the image vector space. Also, we assume the object's parameter with posture or displacement is not limit to one, and we give a description of three parameters $\theta_j, (j = 1, 2, 3)$ below. In learning, we attach the information track

$$\boldsymbol{\eta}_j = [y_{j1}, y_{j2}, \ldots, y_{jw}]^T \tag{6}$$

of the object's parameters (e.g. postures, displacement) at the bottom one line (Actually, length of $\boldsymbol{\eta}_j$ is enough with a few pixels, if the sampling error due to integer description is negligible.) of the image for each parameter, and we get expanded leaning sample vector

$$\boldsymbol{\zeta} = [\boldsymbol{\chi}^T, \boldsymbol{\eta}_1^T, \boldsymbol{\eta}_2^T, \boldsymbol{\eta}_3^T]^T. \tag{7}$$

With this description, we compute an eigen space E for M pieces of learning samples $\{\boldsymbol{\zeta}_1, \boldsymbol{\zeta}_2, \ldots, \boldsymbol{\zeta}_M\}$ as well as BPLP method.

In a process of parameters estimation, we apply BPLP method to expanded image vector

$$\boldsymbol{\zeta}' = [\boldsymbol{\chi}'^T, \mathbf{0}^T, \mathbf{0}^T, \mathbf{0}^T]^T \tag{8}$$

of an input image $\boldsymbol{\chi}'$, that lacked information tracks because the object's parameters of input image are unknown. An estimation result that is restored information tracks of expanded input image vector are written as:

$$\hat{\boldsymbol{\zeta}} = E(E^T\Sigma E)^{-1}E^T\boldsymbol{\zeta}' \tag{9}$$

where

$$\Sigma = \mathrm{diag}(\overbrace{1, \ldots, 1}^{M}, \overbrace{0, \ldots, 0}^{3w}) \tag{10}$$

is a diagonal matrix shown lacking element's position. If we describe the eigen vectors as block matrix

$$E = [E_I^T, E_{O_1}^T, E_{O_2}^T, E_{O_3}^T]^T, \tag{11}$$

information track of parameter θ_j is directly calculated by

$$\hat{\boldsymbol{\eta}}_j = E_{Oj}(E_I^T E_I)^{-1}E_I^T\boldsymbol{\chi}'. \tag{12}$$
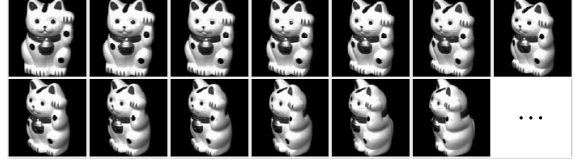

Figure 1: Learning samples.

2.3 Parameter Estimation

We can consider many kind of description formats of information tracks. However, as the information track description, we adopted sine wave

$$y_{ji} = K\cos\left(\frac{2\pi}{w}(i-1) - \theta_j\right) + C, \quad i = 1, \ldots, w \tag{13}$$

where K, C are constants. The values θ_j were expressed with the phase of the each sine wave because it can express periodicity such as rotation angle. Furthermore, the information track that restored by the BPLP method becomes a sine wave in this description at any input image, and we can detect a phase by the inner products of cos and sin vectors

$$\boldsymbol{\omega_c} = \left[\cos 0, \cos(\tfrac{2\pi}{w}), \ldots, \cos(\tfrac{2(w-1)\pi}{w})\right] \tag{14}$$

$$\boldsymbol{\omega_s} = \left[\sin 0, \sin(\tfrac{2\pi}{w}), \ldots, \sin(\tfrac{2(w-1)\pi}{w})\right] \tag{15}$$

without fourier transformation easily. Inner products with information track that restored by BPLP method are

$$\begin{bmatrix} c_j \\ s_j \end{bmatrix} = \begin{bmatrix} \boldsymbol{\omega_c}^T \\ \boldsymbol{\omega_s}^T \end{bmatrix}\hat{\boldsymbol{\eta}}_j \equiv \begin{bmatrix} \boldsymbol{\Omega}_{cj}^T \\ \boldsymbol{\Omega}_{sj}^T \end{bmatrix}\boldsymbol{\chi} \tag{16}$$

where

$$\begin{bmatrix} \boldsymbol{\Omega}_{cj}^T \\ \boldsymbol{\Omega}_{sj}^T \end{bmatrix} = \begin{bmatrix} \boldsymbol{\omega_c}^T \\ \boldsymbol{\omega_s}^T \end{bmatrix} E_{Oj}(E_I^T E_I)^{-1}E_I. \tag{17}$$

Therefore, c_j and s_j are able to calculate from only each inner product of $\boldsymbol{\Omega}_{cj}^T\boldsymbol{\chi}$ or $\boldsymbol{\Omega}_{sj}^T\boldsymbol{\chi}$. Then, we get the phase of the information tracks that mean the parameters of the object by

$$\hat{\theta}_j = \tan^{-1}\left(\frac{s_j}{c_j}\right). \tag{18}$$

These vectors $\boldsymbol{\Omega}_{cj}$ and $\boldsymbol{\Omega}_{sj}$ can be expressed also as images, and we call these vectors EbC image pair.

3 Experimental Results

3.1 EbC Image Pair Generation

The EbC method has a learning process and generates EbC image pair before parameter estimation process. Columbia Object Image Library (COIL-20)[9] that is 128×128 pixels gray scale image library of 20 objects and we used one object of this library shown in figure 1 for the experiment. For experiment, we set up parameters of estimation for 3DOFs of rotation vertical axis of the object, translation of vertical
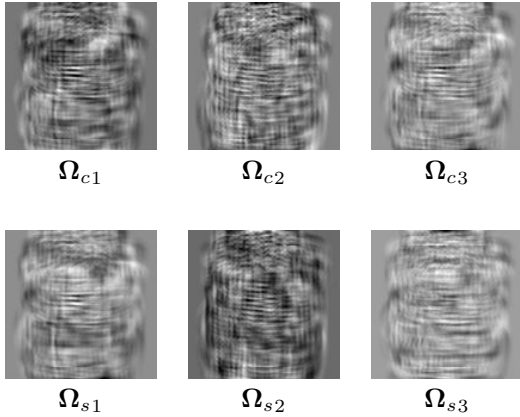
Ω_{c1} Ω_{c2} Ω_{c3}

Ω_{s1} Ω_{s2} Ω_{s3}

Figure 2: EbC image pairs ($\Delta v = 4$).

Δv	M	D	Comp. time
12	324	131	19sec.
6	900	252	2min. 18sec.
4	1764	337	8min. 46sec.
3	2916	388	23min. 14sec.

Table 1: The results of computational times at number of samples M, Eigenspace dimension D.

Figure 3: Set up values of translation

(a) In case of $\Delta v = 12$ (b) In case of $\Delta v = 6$

(c) In case of $\Delta v = 4$ (d) In case of $\Delta v = 3$

Figure 4: Pose estimation results

axis and horizontal axis of the image plane. The image variations of this parameter are given by original COIL-20 library, and we generate images for the variation of translation beforehand. Furthermore, the image variations of the vertical axis rotation are given by 72 pieces and we used 36 samples of the rotation angle $R_\theta = 0, 10, \ldots, 350$[deg] for learning and used the other samples $R_\theta = 5, 15, \ldots, 355$[deg] for accuracy evaluation of proposed method. That means we used separate samples for the evaluation. The EbC image pairs for R_θ and v_x, v_y were computed at the translation unit width $\Delta v = 12, 6, 4, 3$ [pixel], with the range of $-12 \leq v_x \leq 12$ and $-12 \leq v_y \leq 12$. Therefore, the number of learning samples $M = 36 \times 3 \times 3 = 324$ in $\Delta v = 12$, because of $v_x = -12, 0, 12$ and $v_y = -12, 0, 12$. Figure 2 shows a sample of the EbC image pairs where $\Delta v = 4$, $K = 127, C = 127$, and we encoded phases θ_j of information tracks by

$$\theta_1 = R_\theta, \quad \theta_2 = \pi \frac{v_x}{24}, \quad \theta_3 = \pi \frac{v_y}{24}. \quad (19)$$

The dimension of eigen space was set by cumulative proportion ratio of 99.0%. The table is 1 shows the number of sample images M, eigenspace dimension E and computational time with the linux computer 3.0 GHz CPU.

3.2 Parameter Estimation Results

For the pose estimation, we gave a displacement to 36 sample images that not used for the learning by pixel-by-pixel movement in the range of $-12 \leq v_x \leq 12$ and $-12 \leq v_y \leq 12$ as shown in Figure 3. Therefore, when we generate EbC image pair by $\Delta v = 12$, if we set the displacements to $(v_x, v_y) = (-12, -12), (-11, -12), \ldots, (0, 0), \ldots, (+12, +12)$,
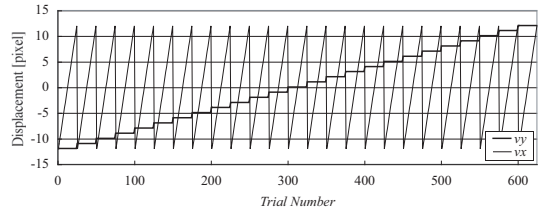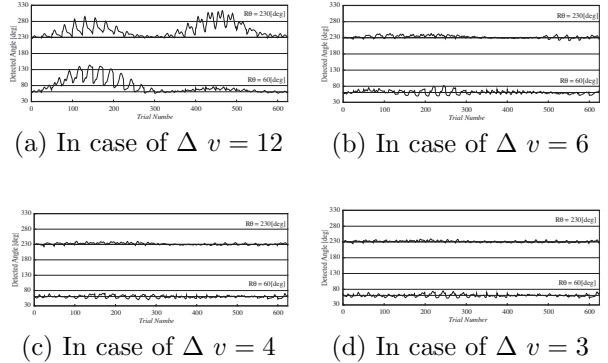
these displacements are matching with learning samples. However, the parameters are estimated from the interpolation of learning samples at all conditions because we use separate test images. The examples of pose parameter estimation are shown in figure 4 where the given rotational angles for the input images are $R_\theta = 60, 230$[deg]. The computational time of all parameters (R_θ and v_x, v_y) was 533[μsec] for each input image with the linux computer 3.0 GHz CPU. Horizontal axes of figure 4 are trial numbers of input image for parameter estimation and displacement shown in figure 3 was given to each input image. From these results, we can confirm the estimation results of pose angle are approaching to the true value by the smaller step width as a general trend. However, there is no difference between $\Delta v = 4$ and $\Delta v = 3$ because the densities of learning samples in these conditions were almost saturated. Therefore, we estimate it is a limit of proposed method at given samples.

The parameter estimation error in other objects in the COIL-20 is shown in figure 5. These results show the estimation error of posture angle (horizontal axis of the figure.) and average of displacements (vertical axis of the figure.) of each object of all estimation at $\Delta v = 3$. From these results, it is supposed to the estimation error of objects near cylinder shape are rather small, and the other objects those change contour shapes significantly with rotation are big. We would like to make clear about this relation in the future works.

4 Discussions

4.1 Dimensionality of the eigenspace

Generally, the accuracy is in trend of drop with lower dimension eigenspace. However, it is not always true.

Δv	12	6	4	3
R_θ [deg]	24.8	7.76	5.86	5.31
v_x [pixel]	1.11	0.833	0.692	0.696
v_y [pixel]	1.53	0.724	0.490	0.450

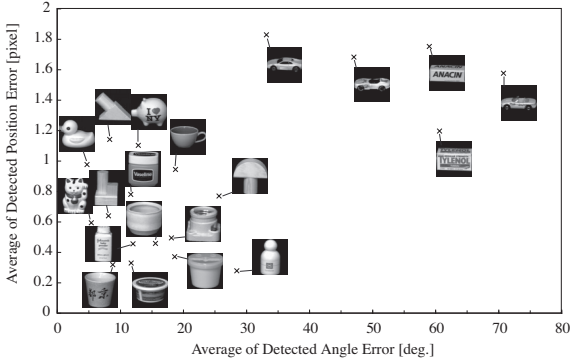Table 2: The average of pose angle and translation estimation accuracy.
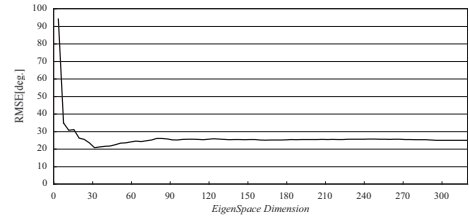

Figure 5: The estimation error of other objects.

(a) posture estimation error.

(b) displacement estimation error.

Figure 6: RMSE of estimated translations

Because it has a property of curse-of-dimension, so it is not a better choice to use eigenspace of the higher dimension. The parameter estimation process of EbC method was integrated to EbC image pairs, but these image pairs are meaning eigenspace projection and parameter space projection. Therefore, the EbC method has a problem the decision of the eigenspace dimension and it affects to the accuracy of parameter estimation.

Figure 6 shows a relation between posture or displacement estimation error vs dimension of the eigenspace. These vertical axes are parameter estimation error of posture angle and displacement, horizontal axes are dimension of the eigen space, and these results were calculated from 625 test images that generated by pixel-by-pixel movement in the range of $-12 \leq v_x \leq 12$ and $-12 \leq v_y \leq 12$. In these results, we used odd-number images of COIL-20 object 4 for learning, and used even-number images for test to evaluate the generalization performance.
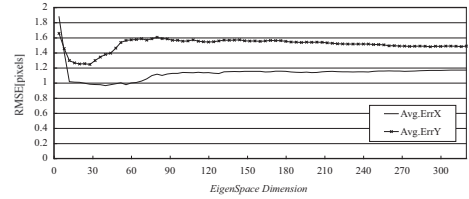
When we generate EbC image pair from all eigen vectors, the calculation by proposed method is interchangeable with Okatani's method, but it is apparent that its case is not best from the figure 6. The eigenspace of 30 to 40 dimension are better suited for parameter estimations and estimation errors are small. Because the generalization by the eigenspace is effective for this parameter estimation problem. We expect that to use of higher dimensional eigenspace make big estimation errors when irrelevance components to estimate parameters are included to learning samples because these samples make effect on higher dimensional eigenvectors. We wish to solve about this relation and consider the decision method of the eigenspace dimension in the future work.

4.2 Number of images for learning

The number of samples M is important for practical situations because taking many images for learning is very difficult in general. As a simulation, we show the performance of estimation when the number of images increases more than 10,000[3]. However, usually we have to perform estimation with relatively small number of sample images.

Figure 7 show errors in estimates as M changes. This experiments estimate just 1DOF rotation angle R_θ, therefore the maximum number of images is 72. The horizontal axis is $M = 72, 36, 24, 18, 12, 9, 8, 6, 4, 3$ where the samples are taken at evenly spaced view angles, and the vertical axis is RMSE of estimated angles. No dimensionality reduction of the eigenspace is applied. From the figures, at least 12 samples are needed for all of these cases to achieve reasonable error less than 5 degrees.

Clearly, we can see three types of objects. Figure 7(a) shows the first type. As we expected, small training images lead to large errors. Figure 7(b) shows the second type. Objects in this type keep errors less than 15 degrees even for only three training images. The last type is shown in Figure 7(c). This type behaves strangely that errors for more training images show larger error than for small training images. This types are further investigated as future work.

4.3 Accuracy for estimating 3DOF rotation

It is interesting to apply the proposed method to estimation of 3DOF rotation. This is the case usually called 3DOF pose estimation, and it requires to represent 3D pose with a 3×3 rotation matrix, whereas the experiment above used an angle and two translations.

3DOF rotation requires a large number of samples much more than 1DOF rotation. Therefore, we used 2000 synthesized images of a 3D model [11] as training images. Then, we used 100 image for evaluation. 3DOF poses of both training and test images were restricted to be less than 30 degrees as a rotation angle of the exponential map[5] (rotation axes were randomly chosen). We had the average error of estimate angles of 2.26 ± 2.07 degrees (again this is a rotation angle).

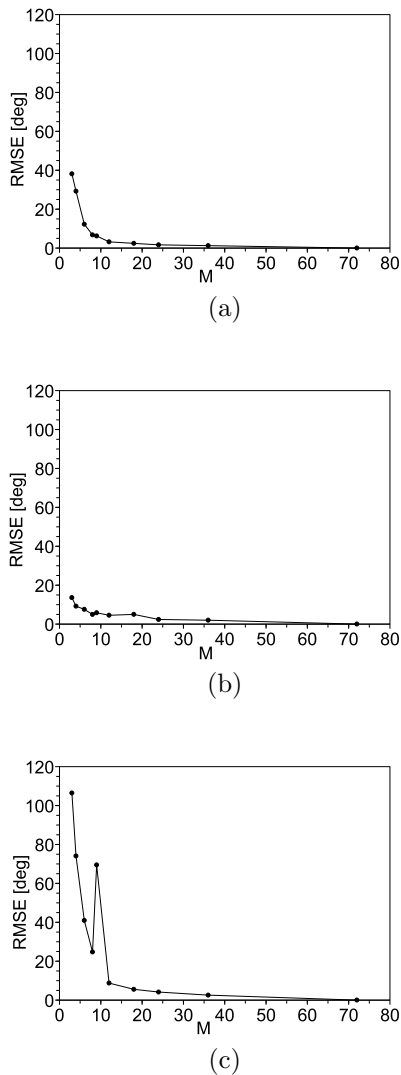However, estimation of fully random 3DOF pose is difficult to achieve a reasonable estimation less than

(a)

(b)

(c)

Figure 7: Results for different M. RMSEs are shown for (a) object 4, (b) object 15, and (c) object 5 in COIL-20.

20 degrees. We think that the use of kernelized BPLP [1] would overcome this limitation of linear property of the proposed method.

5 Conclusion

In this paper, the EbC method for estimating the postures and displacements parameters of a three-dimensional object at high speed from a input image was described. The EbC method introduced the concept of the information track to an image into the parameter estimation method of the appearance base, and parameter estimation was realized from the information track restoration by using the image interpolation by the BPLP method. All these information track restoration processes were expressed by the EbC image pair, and parameter estimation was realized by inner product operation and arctangent calculation. As a future work, we continue to research a relationship about the number of learning samples, degree of freedom and dimension of information tracks. And also, we consider a modest calculation of the EbC image pair for multiple degree of freedom.

References

[1] T. Amano. Image interpolation by high dimensional projection based on subspace method. *Proc.of 17th ICPR*, pages 665–668, 2004.

[2] T. Amano, H. Shinsaku, K. Sato, and S. Inokuchi. Object search and pose detection based on eigenspace approach. *Proc. of the 2000 JapanUSA Flexible Automation Conference*, CD-ROM Proceedings:F–04 SIGNAL PROCESSING 2000JUSFA–13104, 2000.

[3] T. Amano and T. Tamaki. Linear 3-d object pose estimation with dense sample images. *MVA2009*, 2009.

[4] K. Deguchi and T. Noguchi. Visual servoing using eigen-space method and dynamic calculation of interaction matrices. *Proc. of the 13th International Conference on Pattern Recgnition*, 1:302–306, 1996.

[5] Y. Ma, S. Soatto, J. Košecká, and S. S. Sastry. *An Invitation To 3-D Vision*. Springer, 2004.

[6] H. Murase and S. K. Nayar. Illumination planning for object recognition using parametric eigenspaces. *PAMI*, 16(12):1219–1227, 1994.

[7] H. Murase and R. Sakai. Moving object recognition in eigenspace representation: Gait analysis and lip reading. *Pattern recognition letters*, 17:155–162, 1996.

[8] S. K. Nayar, H. Murase, and S. A. Nene. *Parametric Appearance Representation* Early Visual Learning-*Chapter 6*. Oxford University Press, 1996.

[9] S. A. Nene, S. K. Nayar, and H. Murase. Columbia object image library (coil-20). Research note, Columbia University, 1996.

[10] T. Okatani and K. Deguchi. Yet another appearance based method for pose estimation based on a linear model. *MVA2000*, pages 258–261, 2000.

[11] P. Shilane, P. Min, M. Kazhdan, and T. Funkhouser. The princeton shape benchmark. *Shape Modeling International*, 2004.

[12] M. Turk and A. P. Pentland. Face recognition using eigenface. *CVPR91*, pages 586–591, 1991.