

# Use of Fourier and Karhunen-Loeve Decomposition for Fast Pattern Matching With a Large Set of Templates

Michihiro Uenohara, *Member, IEEE*, and Takeo Kanade, *Fellow, IEEE*

**Abstract**—We present a fast pattern matching algorithm with a large set of templates. The algorithm is based on the typical template matching speeded up by the dual decomposition; the Fourier transform and the Karhunen-Loeve transform. The proposed algorithm is appropriate for the search of an object with unknown distortion within a short period.

Patterns with different distortion differ slightly from each other and are highly correlated. The image vector subspace required for effective representation can be defined by a small number of eigenvectors derived by the Karhunen-Loeve transform. A vector subspace spanned by the eigenvectors is generated, and any image vector in the subspace is considered as a pattern to be recognized.

The pattern matching of objects with unknown distortion is formulated as the process to extract the portion of the input image, find the pattern most similar to the extracted portion in the subspace, compute normalized correlation between them at each location in the input image, and find the location with the best score. Searching for objects with unknown distortion requires vast computation. The formulation above makes it possible to decompose highly correlated reference images into eigenvectors, as well as to decompose images in frequency domain, and to speed up the process significantly.

**Index Terms**—Template matching, pattern matching, Karhunen-Loeve transform, Fourier transform, eigenvector.

## 1 INTRODUCTION

TEMPLATE matching has been a useful and familiar tool to detect an object in an image [1]. Template matching finds a pattern in the image that is similar to a given reference image using correlation or normalized correlation as the measurement of similarity.

The drawback of template matching is its high computation cost. It is not robust for rotation and other distortion of objects, either. Depending on the particular point of view, the object can appear as a number of different-looking images. Template matching can be applied to objects with unknown distortion by doing matching with many reference images from a number of points of view. The difficulty lies in its high computation cost. The more templates used for detecting the object from a wide range of views precisely, the higher the computation cost becomes.

A multiresolution image structure can reduce the search area, and, therefore, the computation cost. In the “coarse-to-fine” strategy, images and templates at different levels of resolution are generated, and the templates are searched at the lower resolution first, and the best-match location is found. The neighborhood of the best-match location is searched in an image using increasingly higher resolutions, up to the original resolution image. The “coarse-to-fine”

strategy works well for an object with significantly low spatial frequency components which are retained in a low resolution image. However, it does not work well for cluttered scenes and objects whose details need to be checked in order to distinguish one from another. If an object is missed in a lower resolution image, it cannot be recovered at later stages. To implement the “coarse-to-fine” strategy, it is also necessary to decide how many levels of resolution are to be used. It depends on the objects and scene. There is a trade-off between reducing computation and increasing the risk of missing objects.

Recently, fast template matching for multiple rotated templates has been proposed [2], [3]. The Karhunen-Loeve transform is first applied to a set of rotated images, and eigenvectors are extracted from them. Each template in this set is approximated by a linear combination of these eigenvectors. Since rotated templates differ slightly from each other and are highly correlated, templates can be approximated reasonably well by a smaller number of eigenvectors. Normalized correlation between rotated templates and the input image is efficiently computed by substituting the approximations for the templates when computing the normalized correlation. The computation cost for detecting targets from the whole image is still high. Multiresolution images are used to reduce computation cost, which leads to the same difficulty for cluttered scenes, as described before.

We will present a fast pattern matching algorithm with a large set of templates without multiresolution images. The algorithm is based on the typical template matching, which is the search for the given pattern in the

- M. Uenohara is with Toshiba R&D Center, 4-1, Ukishima-cho, Kawasaki-ku, Kawasaki, 210, Japan. E-mail: mue@eml.rdc.toshiba.co.jp.
- T. Kanade is with the Robotics Institute, Carnegie Mellon University, Smith Hall, Pittsburgh, PA 15213-3898. E-mail: tk@cs.cmu.edu.

Manuscript received 2 Apr. 1996. Recommended for acceptance by H.R. Keshavan. For information on obtaining reprints of this article, please send e-mail to: transpami@computer.org, and reference IEEECS Log Number 105024.

image, speeded up by the dual decomposition, the Fourier transform, and the Karhunen-Loeve transform. The matching criterion is normalized correlation.

Patterns with different distortion differ slightly from each other and are highly correlated. The image vector subspace required for effective representation can be defined by a small number of eigenvectors derived by the Karhunen-Loeve transform. Instead of approximating each template as a linear combination of the eigenvectors, a vector subspace spanned by the eigenvectors is generated. The vector subspace involves not only the discrete reference images with different distortion, but also their interpolation. Any image vector in the vector subspace is considered to be a pattern to be recognized.

The pattern matching of objects with unknown distortion is now formulated as the process to extract a portion of the input image, find the pattern most similar to the extracted portion in the vector subspace, compute the normalized correlation at each location in the whole input image, and find the location with the highest score.

It is well known that the computation of correlation can be reduced greatly by using the Fourier transform, especially when the image size is large. The formulation above makes it possible to apply the Fourier transform in an efficient way and speeds up the process significantly. It should be noted that the normalized correlation to multiple reference images represented as linear combinations of the eigenvectors is not speeded up by the Fourier transform. The number of Fourier transforms and inverse Fourier transforms is reduced by representing the reference images as the linear combinations of the eigenvectors; however, the computation for the linear combination of the Fourier transform of the eigenvectors is not negligible, and the whole computation cost is not reduced greatly. The alternative process is to generate the vector subspace, then find the most similar pattern in the vector subspace, and to compute the normalized correlation between them. This eliminates the computation for the linear combination.

The computation cost of the proposed pattern matching in an image of size  $N \times N$  is  $O((K+5)N^2 \log_2 N^2)$ , while the computation cost of the normalized correlation with the original  $P$  reference images is  $O((P+1)N^2 M^2)$  when the size of the reference images are  $M \times M$ .  $K$  is the number of eigenvectors used for the vector subspace.  $N$  is assumed to be a power of two. Its reduction rate is  $(\log_2 N^2)/M^2 \times K/P$ , which is the product of the reduction by the Karhunen-Loeve transform and that by the Fourier transform.

The paper is organized as follows. The vector subspace and normalized correlation in the vector subspace is explained in Section 2. In Section 3, the proposed pattern matching algorithm, which is the normalized correlation in the vector subspace using the Fourier transform, is presented, and experimental results are shown in Section 4.

## 2 NORMALIZED CORRELATION IN THE VECTOR SUBSPACE

### 2.1 The Karhunen-Loeve Transform

The Karhunen-Loeve (K-L) transform is a familiar technique for projecting a large amount of data onto a small dimensional subspace in pattern recognition and image compression [4], [5]. The K-L transform gives the orthogonal basis functions as the eigenvectors of the covariance matrix. This transform is optimal in that it is a canonical transform minimizing the mean square error between a truncated representation and the actual data.

Let the set of input data be  $\mathbf{x}_i, i = 1, 2, \dots, P$ ; vectors of dimension  $M^2$ , representing  $M \times M$  square images. The covariance matrix of the input data is

$$A = \frac{1}{P} \sum_{i=1}^P (\mathbf{x}_i - \mathbf{c})(\mathbf{x}_i - \mathbf{c})^T \quad (1)$$

where  $\mathbf{c}$  is the average image vector. The vectors  $\mathbf{e}_j$  and scalars  $\lambda_j$  are the eigenvectors and eigenvalues of the covariance matrix  $A$ , respectively. We obtain the optimal approximation of the input data by selecting eigenvectors in decreasing order of magnitude of the eigenvalues and representing each datum by a linear combination of major  $K$  eigenvectors as

$$\mathbf{x}_i \approx \mathbf{c} + \sum_{j=1}^K p_{ij} \mathbf{e}_j \quad (2)$$

where  $\mathbf{p}_i = [\mathbf{e}_1, \mathbf{e}_2, \dots, \mathbf{e}_K]^T (\mathbf{x}_i - \mathbf{c})$  and mean approximation error is

$$\epsilon^{(K)} = \sum_{i=K+1}^{P-1} \lambda_i.$$

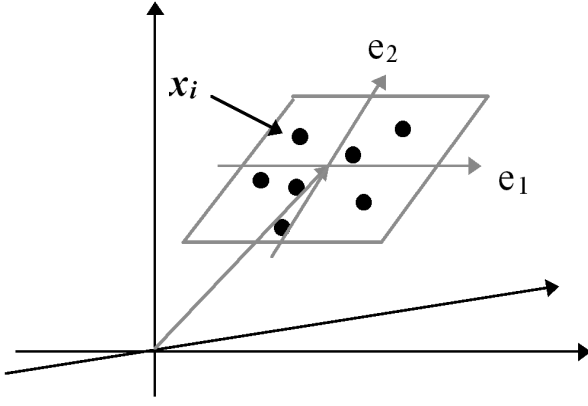
The cumulative proportion  $\mu^{(K)}$  is useful for deciding the number of eigenvectors

$$\mu^{(K)} = \left[ \sum_{i=1}^K \lambda_i \right] / \left[ \sum_{i=1}^{P-1} \lambda_i \right] \quad (3)$$

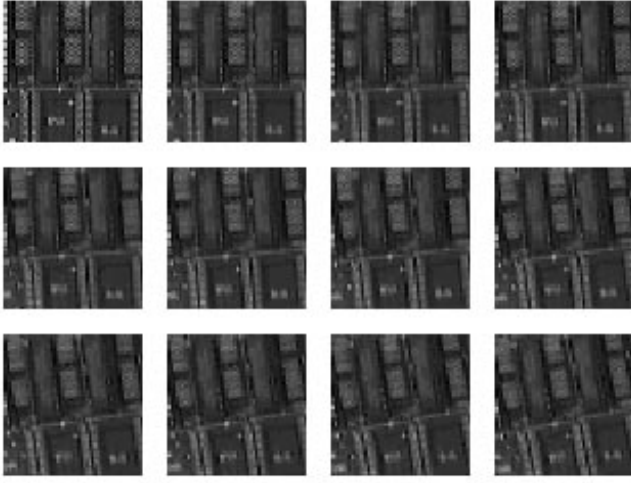
### 2.2 Vector Subspace

When we search an object with unknown distortion, the straightforward way is to do template matching with a large set of templates. Each template is the intensity pattern at a different point of view. Reference images with different points of view differ slightly from each other and are highly correlated. The image vector subspace required for effective representation can be defined by a small number of eigenvectors derived by the K-L transform. (See Fig. 1.)

The major  $K$  eigenvectors, in addition to the average image vector  $\mathbf{c}$ , span a  $(K+1)$ -dimensional subspace of all possible images, and a set of images in the subspace is considered as a template to be recognized [6]. A set of reference images in the vector subspace is therefore expressed in terms of a linear combination of a finite set of orthonormal basis:



(a)



(b)

Fig. 1. (a) Distribution of reference images. (b) Example of reference images.

$$\mathbf{x} = \sum_{j=0}^K p_j \mathbf{e}_j \quad (4)$$

where

$$\mathbf{e}_0 = \left( \mathbf{c} - \sum_{j=1}^K p_j^c \mathbf{e}_j \right) / \left\| \mathbf{c} - \sum_{j=1}^K p_j^c \mathbf{e}_j \right\|$$

is the normalized average image vector from which the projection into the subspace spanned by the  $K$  eigenvectors is subtracted, and  $\mathbf{p}^c = [\mathbf{e}_1, \mathbf{e}_2, \dots, \mathbf{e}_K]^T \mathbf{c}$ .

### 2.3 Normalized Correlation in the Vector Subspace

The input image is evaluated at each location as to how it fits the template by extracting the region and finding the pattern most similar in the vector subspace and computing normalized correlation between them. When we extract a portion of image  $\mathbf{y}$ , we normalize it so that the average intensity of the whole pixels is zero, i.e.,

$$\sum_{i=1}^{M^2} \tilde{y}_i = 0.$$

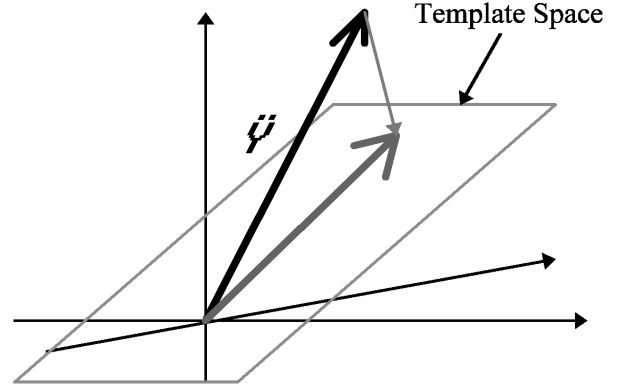


Fig. 2. Template space and the projection of input image.

$M^2$  is the number of pixels in the reference images. This normalization makes matching insensitive to the variation of the background intensity.

The most similar pattern in the vector subspace is the projection of the normalized extracted region vector  $\tilde{\mathbf{y}}$  into the vector subspace (Fig. 2). Its normalized correlation to the vector  $\tilde{\mathbf{y}}$  is the largest. The normalized correlation between the vector  $\tilde{\mathbf{y}}$  and a reference vector  $\mathbf{x}$  is given by

$$\bar{C}(\mathbf{x}, \tilde{\mathbf{y}}) = \frac{\mathbf{x} \tilde{\mathbf{y}}}{\|\tilde{\mathbf{y}}\|}.$$

Replacing the reference image vector  $\mathbf{x}$  with the projection

$$\tilde{\mathbf{x}} = \sum_{j=1}^K (\mathbf{e}_j^T \tilde{\mathbf{y}}) \mathbf{e}_j$$

yields

$$\bar{C}(\tilde{\mathbf{y}}, \tilde{\mathbf{x}}) = \frac{\sqrt{\sum_{j=0}^K (\mathbf{e}_j^T \tilde{\mathbf{y}})^2}}{\|\tilde{\mathbf{y}}\|} \quad (5)$$

The coefficient vector  $\tilde{\mathbf{p}}$  for the projection  $\tilde{\mathbf{x}}$  is

$$\tilde{\mathbf{p}} = [\mathbf{e}_0^T \tilde{\mathbf{y}}, \mathbf{e}_1^T \tilde{\mathbf{y}}, \dots, \mathbf{e}_K^T \tilde{\mathbf{y}}]^T.$$

The normalized correlation score above is the measure of similarity considering not only the prestored discrete  $P$  reference images but also their interpolation. This makes the system robust against the variation of illumination. The computation cost is greatly reduced compared to the normalized correlation to the original  $P$  reference images. The original normalized correlation requires  $M^2(P+1)$  operations for  $P$  reference images, while (5) requires only  $M^2(K+2)$  operations, where  $K$  can be much smaller than  $P$ .

## 3 NORMALIZED CORRELATION USING THE FOURIER TRANSFORM

It is well known that the Fourier transform of the correlation of the two functions is the product of the Fourier transform of the one function and the complex conjugate of another function [7].

$$F[C(\mathbf{x}, \mathbf{y})] = F[\mathbf{x}] F^*[\mathbf{y}] \quad (6)$$

The inverse Fourier transform of the product above gives the values of the correlation at different lags. The computation using the Fourier transform is much more computationally efficient than the correlation in spatial domain, especially in the case of large size images.

### 3.1 Normalized Correlation in the Vector Subspace Using the Fourier Transform

Normalized correlation in the vector subspace is computed as below:

$$\bar{C}(\tilde{\mathbf{y}}, \tilde{\mathbf{x}}) = \frac{\sqrt{\sum_{j=0}^K (\mathbf{e}_j^T \tilde{\mathbf{y}})^2}}{\|\tilde{\mathbf{y}}\|} \quad (7)$$

We compute eigenvectors  $\mathbf{e}_l$  from normalized reference images  $\sum_{l=1}^{M^2} \mathbf{x}_{li} = 0$ . The summation of all pixels in each eigenvector is, therefore, zero.

$$\sum_{l=1}^{M^2} \mathbf{e}_{li} = 0 \quad (8)$$

The correlation between the eigenvectors  $\mathbf{e}_l$  and the normalized portion of the input image  $\tilde{\mathbf{y}}$  is the same as the correlation between the eigenvectors  $\mathbf{e}_l$  and the portion of the input image  $\mathbf{y}$

$$\mathbf{e}_l^T \tilde{\mathbf{y}} = \sum_{i=1}^{M^2} \mathbf{e}_{li} (y_i - \bar{y}) = \mathbf{e}_l^T \mathbf{y} \quad (9)$$

$\bar{y}$  is the average of the portion of the input image  $\mathbf{y}$ . The correlation above can be computed efficiently using the Fourier transform as: FFT the two data sets  $\mathbf{e}_l$  and  $\mathbf{y}$ , multiply the one resulting transform by the complex conjugate of the other, and inversely transform the product.

The norm of the normalized portion of the image  $\|\tilde{\mathbf{y}}\|$  at location  $(i, j)$  can also be computed using the Fourier transform:

$$\|\tilde{\mathbf{y}}\|_{ij} = \sqrt{\sum_{l=1}^{M^2} (y_l - \bar{y})^2} = \sqrt{\left( \sum_{l=1}^{M^2} y_l^2 \right) - M^2 \bar{y}^2} \quad (10)$$

$$\left[ \sum_{l=1}^{M^2} y_l^2 \right]_{ij} = \mathbf{I}_M \otimes [\mathbf{y}^2] = F^{-1} [F[\mathbf{I}_M] \bullet F^*[\mathbf{y}^2]] \quad (11)$$

$$[M^2 \bar{y}]_{ij} = \left[ \sum_{l=1}^{M^2} y_l \right]_{ij} = \mathbf{I}_M \otimes [\mathbf{y}] = F^{-1} [F[\mathbf{I}_M] \bullet F^*[\mathbf{y}]] \quad (12)$$

where  $\mathbf{I}_M$  is the matrix of the size  $M \times M$  with all the elements of unity.

We obtain the normalized correlation (7) by computing the summation of the correlation between the eigenvectors  $\mathbf{e}_l$  and the portion of the input image  $\mathbf{y}$ , computing the square root of it, and then dividing it by the norm of the normalized portion of the input image (10).

### 3.2 Off-Line and On-Line Processing of the Proposed Pattern Matching

We will show the fast pattern matching algorithm with a

large set of templates to detect the location of the object and the best matched template which indicates the geometric distortion parameters of the object. It involves off-line processing and on-line processing.

In off-line processing, we first gather or generate reference images of the object with different distortion parameters.

$$\mathbf{x}_0 \rightarrow \mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_p \quad (13)$$

We compute the average intensity of each reference image and subtract them from the reference images, and normalize the reference images to unit energy:

$$\sum_{i=1}^{M^2} \mathbf{x}_{ij} = 0 \quad (14)$$

$$\mathbf{x}_i^T \mathbf{x}_i = 1 \quad (15)$$

We compute the average image vector  $\mathbf{c}$  and the first  $K$  eigenvectors by the Karhunen-Loeve transform. We select  $K$  eigenvectors whose corresponding eigenvalues are the largest. We use a cumulative proportion

$$\mu^{(K)} = \left[ \sum_{i=1}^K \lambda_i \right] / \left[ \sum_{i=1}^{P-1} \lambda_i \right]$$

for deciding the number of eigenvectors  $K$ . This measurement shows well how many eigenvectors contribute to approximate reference images. We subtract the projection of the average image vector  $\mathbf{c}$  into the subspace spanned by the  $K$  eigenvectors from the average image vector  $\mathbf{c}$ , and normalize it to unit energy, and call it the No.0 eigenvector  $\mathbf{e}_0$ . We then calculate coefficients by projecting reference images onto the vector subspace spanned by these  $(K+1)$  eigenvectors. The coefficient vector  $\mathbf{p}_i$  is the representation of reference images in the vector subspace corresponding to distortion parameters.

$$\mathbf{p}_i = [\mathbf{e}_0, \mathbf{e}_1, \dots, \mathbf{e}_K]^T \mathbf{x}_i \quad (16)$$

We compute the Fourier transform of the  $(K+1)$  eigenvectors:

$$F[\mathbf{e}_0], F[\mathbf{e}_1], \dots, F[\mathbf{e}_K] \quad (17)$$

We generate the matrix of the size  $M \times M$  with all the elements of unity and compute the Fourier transform of it:

$$F[\mathbf{I}_M] = F \left[ \begin{bmatrix} 1 & 1 & \dots & 1 \\ 1 & 1 & \dots & 1 \\ \dots & \dots & \dots & 1 \\ 1 & 1 & \dots & 1 \end{bmatrix} \right] \quad (18)$$

In on-line processing, we obtain image  $\mathbf{y}$ , and compute the squared image

$$[\mathbf{y}^2]_{ij} = [y_{ij}^2] \quad (19)$$

We then compute the complex conjugate of the Fourier transform of the input image  $[\mathbf{y}]_{ij}$  and the squared image  $[\mathbf{y}^2]_{ij}$

$$F^*[\mathbf{y}], F^*[\mathbf{y}^2] \quad (20)$$

We have the Fourier transform of the eigenvectors and the matrix  $\mathbf{I}_M$  that are precomputed off-line. We calculate the

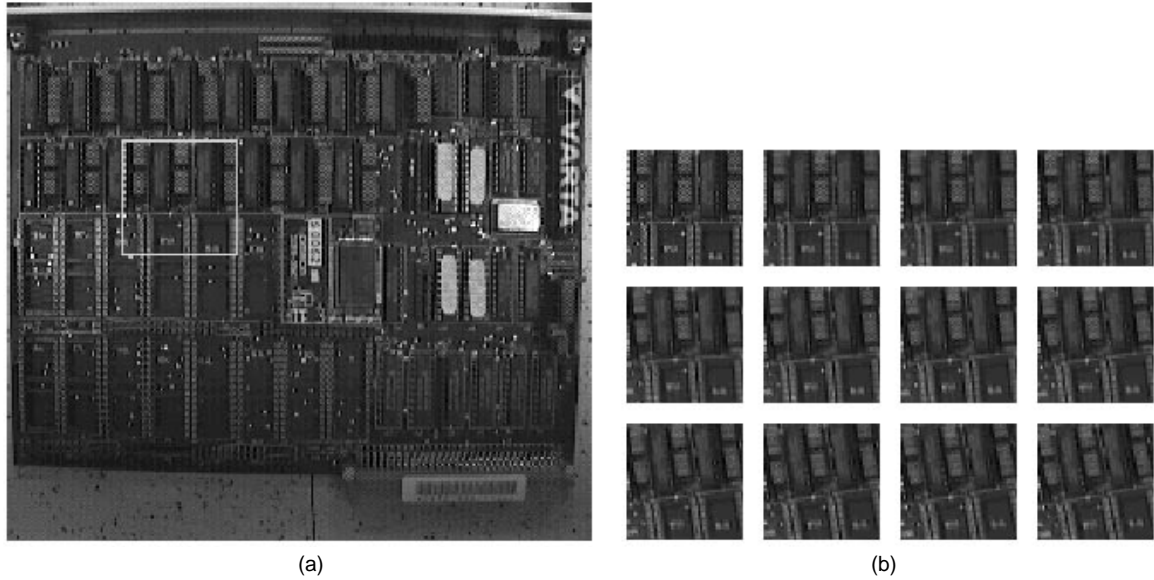


Fig. 3. (a) Printed circuit board, (b) rotated reference images (12 of 101 are shown).

 TABLE 1  
 ORIENTATION OF THE SMALL PART

orientation(deg)	4.1	8.2	12.3	16.4	20.5	24.6	28.7	32.8	36.9	41.0	45.1
detected orientation (deg)	4	8	12	16	20	25	29	32	37	41	45

correlation between the eigenvectors and input image, the matrix  $I_M$  and the input image, and the matrix  $I_M$  and the squared input image.

$$e_i \otimes y = F^{-1}[F[e_i] \cdot F^*[y]] \quad (21)$$

$$I_M \otimes y = F^{-1}[F[I_M] \cdot F^*[y]] \quad (22)$$

$$I_M \otimes y^2 = F^{-1}[F[I_M] \cdot F^*[y^2]] \quad (23)$$

We obtain the correlation between the input image and the pattern most similar in the vector subspace at each location in the image by

$$C(\tilde{y}, \tilde{x})_{ij} = \sum_{j=0}^K [e_j \otimes y]_{ij}^2 \quad (24)$$

The norm of the normalized portion of the input image  $\tilde{y}$  at each location is computed:

$$\|\tilde{y}\|_{ij} = \sqrt{[I_M \otimes y^2]_{ij} - ([I_M \otimes y]_{ij})^2 / M^2} \quad (25)$$

The normalized correlation at each location is then obtained:

$$\bar{C}(\tilde{x}, \tilde{y})_{ij} = \frac{\sqrt{C(\tilde{x}, \tilde{y})_{ij}}}{\|\tilde{y}\|_{ij}} \quad (26)$$

We obtain the location of the object by searching the location with the maximum score.

The vector  $[e_0^T \tilde{y}, e_1^T \tilde{y}, \dots, e_K^T \tilde{y}]^T$  that is already computed in (21) represents the coefficient vectors  $\tilde{p}$  for the projection of the vector  $\tilde{y}$  into the vector subspace.  $\tilde{x}$  is represented as

$$\tilde{x} = \sum_{i=0}^K \tilde{p}_i e_i \quad (27)$$

while each reference image is represented as

$$x_i = \sum_{j=0}^K \tilde{p}_{ij} e_j \quad (28)$$

The distance  $e_i$  between the coefficient vectors  $\tilde{p}$  and the coefficient vectors of the reference image vectors  $p_i$  is computed as:

$$e_i = \sqrt{\sum_{j=0}^K (\tilde{p}_{ij} - \tilde{p}_j)^2} \quad (29)$$

We find the reference image  $x_i$  with the minimum distance and obtain the distortion parameter of the object.

## 4 EXPERIMENTAL RESULTS

We have conducted experiments to verify the accuracy and computational efficiency of the proposed algorithm.

The target object in the experiments is a small part on the printed circuit board shown in Fig. 3a. The size of the image is  $256 \times 236$ . Fig. 3b shows the rotated reference images obtained by rotating the original reference image at the upper-left corner in Fig. 3b. The size of the reference images is  $50 \times 50$ , and the number of reference images is 101 ( $-50$  degrees to  $+50$  degrees, every one degree). They are normalized so that the average of pixel intensities is zero and the energy is unit.

Our first step is to compress the reference image sets into the low-dimensional subspace that captures the most appearance characteristics of objects by the Karhunen-Loeve transform. We use  $K=20$  in the experiments. The cumulative proportion of it is 0.85.

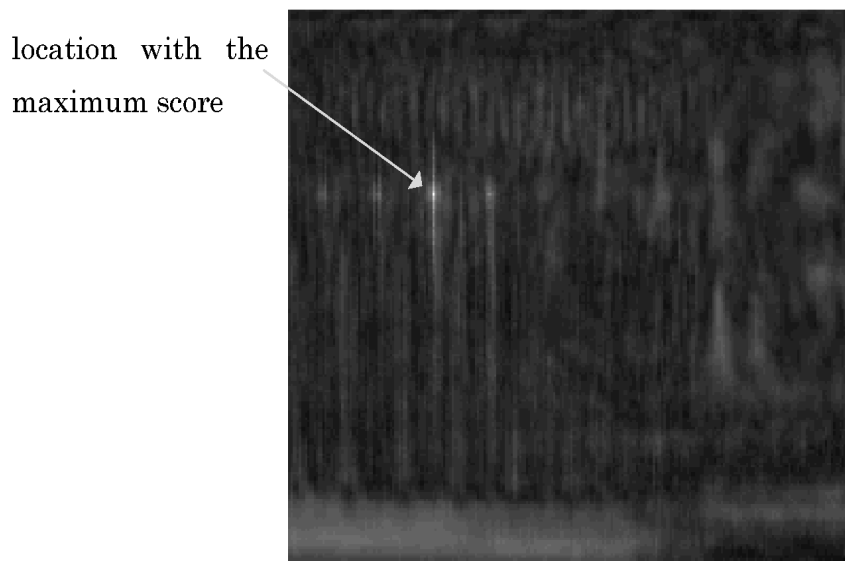


Fig. 4. Normalized correlation at each location.

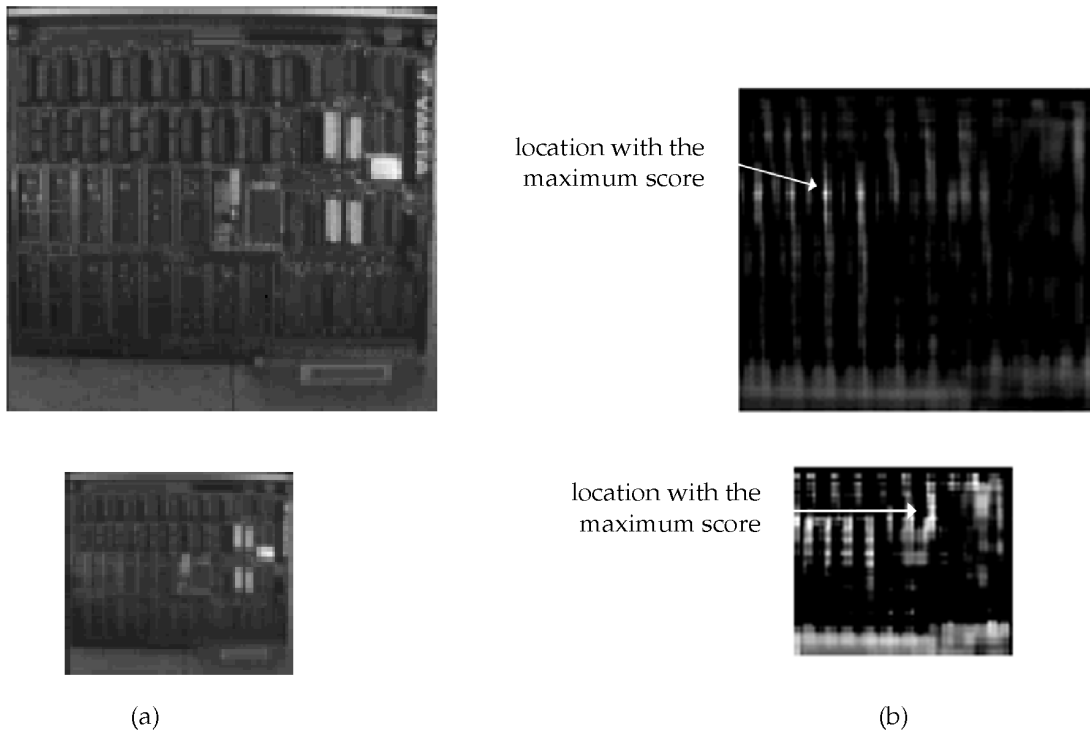


Fig. 5. (a) Lower resolution images. (b) Normalized correlation at each location.

The Fourier transform of the eigenvectors, as well as the Fourier transform of the matrix  $I_M$ , are computed and stored in the file. At the beginning of the on-line operation, those data are loaded from the file and used for computation.

#### 4.1 Detection of the Rotated Printed Circuit

In the first experiment, we generate rotated images of the printed circuit board synthetically and use them for input. We detect the location and orientation of the small part by the proposed algorithm and compare with the location and orientation of the small part in the images we generate.

Table 1 shows the rotation angle of the image (which is the rotation angle of the small part also) and the rotation

angle of the small part detected by the program. The program computes the coefficient vector  $\tilde{p}$  and the normalized correlation score at each location in the image. The program then searches the location with the highest score, which is the location of the small part, and finds the coefficient parameter of the reference image with the minimum distance from the coefficient parameter  $\tilde{p}$  at the location with the highest score. In the experiment, we use the reference images rotated every one degree. The resolution of the detected rotation angle is one degree. We could obtain subdegree accuracy by applying a quadratic fitting function to the correlation score.

Table 1 shows that the proposed algorithm detects the orientation of the small part reasonably well, considering that the size of the reference images in the experiment is  $50 \times 50$ , and one degree rotation moves each pixel in the reference images less than one pixel location. The program works well even if the small part is located to the orientation between reference images.

#### 4.2 Comparison with the "Coarse-to-Fine" Strategy

The second experiment shows that the proposed algorithm works well for an object with high spatial frequency components which cause troubles when we use the "coarse-to-fine" strategy. Fig. 4 shows the normalized correlation score at each location detected by the proposed algorithm. The reference images are the same as in the first experiment and the Fourier transform of the eigenvectors and the matrix  $I_M$  are generated. The input image is the printed circuit board shown in Fig. 3a. Fig. 4 shows that the small part is well distinguished from others. The same type of chips are located horizontally. The cue to distinguish one location from another is the characters on the chips and the pattern of the printed circuit that are high spatial frequency components.

Fig. 5 shows that the "coarse-to-fine" search does not work well for the objects like this. Fig. 5a shows the low resolution images of the printed board. We generate half resolution images and reference images just by averaging four neighborhood pixels. Fig. 5b shows the normalized correlation results between the printed circuit board image and the reference image at each resolution. The small part is detected successfully in the half resolution image. However, it is missed in the one-fourth resolution image.

The computation time for the proposed algorithm in the experiments is 23.9 seconds on the Sparc 10/30. The number of reference images  $P = 101$ , the number of the eigenvectors  $K = 20$ , the size of the input image is  $256 \times 236$ , and the size of the reference image is  $50 \times 50$ . The input image is extended to  $256 \times 256$  by padding the last 20 rows with zeros. We use Fourier transform to compute correlation in the proposed algorithm so that the size of the reference image does not affect the computation time. The computation time for the normalized correlation between the printed circuit board image shown in Fig. 3a and the reference image at the upper-left corner in Fig. 3b is 218.3 seconds. We need  $P = 101$  times computation for computing normalized correlation between 101 reference images, which leads to a computation time of 22,048.3 seconds; 922.5 times the computation time in the case of using the proposed algorithm.

The search in the one-fourth resolution images requires  $1/256$  computation. The computation time in the experiment shows that the proposed pattern matching algorithm reduces computation more than the search in the one-fourth resolution image and still leads to the correct result.

## 5 CONCLUSION

We have presented a novel pattern matching technique with a large set of templates. The object to be recognized is given as multiple intensity patterns with different distor-

tion parameters such as rotation angle, scaling factor. The proposed algorithm decomposes the given pattern by the Karhunen-Loeve transform, and generates the vector subspace spanned by the major eigenvectors. The algorithm then decomposes the input image and the derived eigenvectors by the Fourier transform, finds the pattern most similar in the vector subspace, and computes normalized correlation between the most similar pattern and the input image.

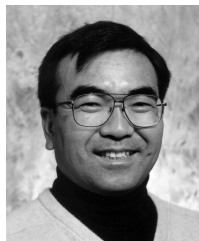
The dominant part of the proposed algorithm in computation is the Fourier transform. The computation cost of the proposed pattern matching in the image of the size  $N \times N$  is  $O((K+5)N^2 \log_2 N^2)$ , while the computation cost of the normalized correlation with the original  $P$  reference images used for generating the vector subspace is  $O((P+1)N^2 M^2)$ .  $K$  is the number of eigenvectors. The size of the reference images is  $M \times M$ . The Karhunen-Loeve transform speeds up the process by  $K/P$ , and the Fourier transform speeds up the process by  $(\log_2 N^2)/M^2$ . In the experiment,  $P = 101$ ,  $K = 20$ ,  $M = 50$ ,  $N = 256$ , and the proposed algorithm speeds up the process by almost 1,000 times.

## REFERENCES

- [1] A. Rosenfeld and A. Ka, *Digital Picture Processing*. New York: Academic Press, 1982.
- [2] S. Yoshimura and T. Kanade, "Fast Template Matching Based on the Normalized Correlation by Using Multiresolution Eigenimages," *Proc. IROS '94*, Munich, Germany, 1994.
- [3] O. Amidi, Y. Mesaki, T. Kanade, and M. Uenohara, "Research on an Autonomous Vision-Guided Helicopter," *Proc. Fifth RI/SME World Conf. Robotics Research*, Cambridge, Mass., 1994.
- [4] K. Fukunaga, *Introduction to Statistical Pattern Recognition*. Boston: Academic Press, 1990.
- [5] H. Murase and S. Nayar, "Visual Learning and Recognition of 3-D Objects from Appearance," *Int'l J. Computer Vision*, vol. 14, no. 1, pp. 5-24, 1995.
- [6] M. Uenohara and T. Kanade, "Vision-Based Object Registration for Real-Time Image Overlay," *Proc. 1995 Conf. Computer Vision, Virtual Reality, and Robotics in Medicine*, Nice, France, 1995.
- [7] W. Press, S. Teukolsky, W. Vetterling, and B. Flannery, *Numerical Recipes in C*. Cambridge Univ. Press, 1992.
- [8] M. Uenohara and T. Kanade, "Vision-Based Object Recognition for Real-Time Image Overlay," *Int'l J. Computers in Biology and Medicine*, vol. 25, no. 2, pp. 249-260, 1995.
- [9] M. Turk and A. Pentland, "Eigenfaces for Recognition," *J. Cognitive Neuroscience*, vol. 3, no. 1, 1991.



**Michihiro Uenohara** received the BS degree in electrical engineering from the University of Tokyo, Tokyo, Japan, in 1985. He joined Toshiba R&D Center, Kawasaki, Japan, in 1985, and currently is a research scientist. He was a visiting research scientist at Carnegie Mellon University, Pittsburgh, Pennsylvania, from 1993 to 1995. His research areas of interest are computer vision and robotics. He is a member of the IEEE.



**Takeo Kanade** received his PhD degree in electrical engineering from Kyoto University, Japan, in 1974. After holding a faculty position at the Department of Information Science, Kyoto University, he joined Carnegie Mellon University in 1980, where he is currently the U.A. Helen Whitaker Professor of Computer Science and director of the Robotics Institute. He was a founding chairperson of CMU's robotics PhD program, probably the first of its kind.

Dr. Kanade has worked in multiple areas of robotics: vision, manipulators, autonomous mobile robots, and sensors. He has written more than 200 technical papers and reports in these areas. He has been the principal investigator of several major vision and robotics projects at Carnegie Mellon.

Dr. Kanade is a fellow of the IEEE, a founding fellow of the American Association of Artificial Intelligence, and the founding editor of the *International Journal of Computer Vision*. Dr. Kanade has been elected to the National Academy of Engineering as a foreign associate.