

Design of Robust Face Recognition System Realized with the Aid of Automatic Pose Estimation-based Classification and Preprocessing Networks Structure

Eun-Hu Kim*, Bong-Youn Kim*, Sung-Kwun Oh[†] and Jin-Yul Kim*

Abstract – In this study, we propose a robust face recognition system to pose variations based on automatic pose estimation. Radial basis function neural network is applied as one of the functional components of the overall face recognition system. The proposed system consists of preprocessing and recognition modules to provide a solution to pose variation and high-dimensional pattern recognition problems. In the preprocessing part, principal component analysis (PCA) and 2-dimensional 2-directional PCA ((2D)²PCA) are applied. These functional modules are useful in reducing dimensionality of the feature space. The proposed RBFNNs architecture consists of three functional modules such as condition, conclusion and inference phase realized in terms of fuzzy “if-then” rules. In the condition phase of fuzzy rules, the input space is partitioned with the use of fuzzy clustering realized by the Fuzzy C-Means (FCM) algorithm. In conclusion phase of rules, the connections (weights) are realized through four types of polynomials such as constant, linear, quadratic and modified quadratic. The coefficients of the RBFNNs model are obtained by fuzzy inference method constituting the inference phase of fuzzy rules. The essential design parameters (such as the number of nodes, and fuzzification coefficient) of the networks are optimized with the aid of Particle Swarm Optimization (PSO). Experimental results completed on standard face database -Honda/UCSD, Cambridge Head pose, and IC&CI databases demonstrate the effectiveness and efficiency of face recognition system compared with other studies.

Keywords: Automatic pose estimation, Preprocessing networks structure, Particle swarm optimization, RBFNNs, 2-dimensional 2-directional PCA

1. Introduction

Biometric technology is about identifying individuals using physical characteristics such as face, iris, and finger scan. Among them, face recognition technology is a non-contact technique for identifying users, so it is less annoying than other biometric technologies [1]. This study is motivated, in particular, by considering the necessity of tracking and recognition of face in an unconstrained environment. In most of the existing face recognition systems, the frontal view of face is preferred to reduce the complexity of the recognition process. Thus individuals may be required to stare into the camera, or the camera should be located so that the frontal images are easily acquired. However, these constraints severely restrict the adoption of face recognition in a wide range of applications [2,3]. To address this problem, we consider the recognition system that recognizes in various pose and size variations for improving the performance of face recognition.

The main objective of this study is to develop robust face recognition to pose variations based on pose estimation and to show its applications to face recognition [4]. The key issues of this study can be summarized as follows:

(a) Novel pose classification step is taken into account. Unlike conventional face recognition system, the pose estimation before the face recognition step is carried out as an essential step in order to alleviate the performance degradation of face recognition caused by the pose change, and then face recognition is performed by using pose-classified face images. In the pose estimation step, the eigen-faces obtained from PCA or (2D)²PCA are used for the pose estimation. (b) Face recognition classifiers are designed corresponding to each facial pose. In this study, five poses (left 90°, left 45°, front, right 45°, and right 90°) are considered and each face recognition classifier is independently designed for each pose. As a result, pose estimation part is used to estimate the facial pose of input image, and then face recognition part is used to select a candidate from the built database based on the pose-classified face image. (c) Preprocessing algorithm serves as an important role in both parts. PCA or (2D)²PCA is used in the pose estimation and face recognition parts, but the

[†] Corresponding Author: School of Electrical and Electronic Engineering, The University of Suwon, Korea. (ohsk@suwon.ac.kr)

* School of Electrical and Electronic Engineering, The University of Suwon, Korea. ({wdkim, jykim}@suwon.ac.kr)

Received: February 16, 2017; Accepted: July 17, 2017

use effect is different in each part. In the pose estimation part, the preprocessing algorithm is employed for finding eigen-faces per each pose for pose estimation whereas, in the face recognition part, this algorithm is applied for the dimensional reduction of high dimensional images.

The RBFNNs is proposed as one of the recognition part of the overall face recognition system that consists of the two parts such as the preprocessing and recognition part[5]. The design methodology and the procedure of the proposed RBFNNs with PCA and (2D)²PCA as pre-processing algorithm are presented to construct solutions to high-dimensional pattern recognition problems and robust face recognition in various pose variations[6].

Two-dimensional face recognition is realized using Honda/UCSD, IC&CI and Cambridge head pose databases [8,9]. Poses of face in test image is estimated using PCA and (2D)²PCA, and then the estimated faces are passed as input data for comparing the face recognition performance of the conventional PCA-based classifiers and the proposed polynomial-based RBFNNs pattern classifier [10]. When performing face recognition based on PCA, (2D)²PCA is also used to recover the performance degradation of recognition by PCA[11].

Appearance difference caused by pose variation is typically more significant than the intrinsic differences between individuals, and thus it is not effective to directly compare face images of different pose. Therefore, many existing face recognition methods are not applicable to varying-pose cases since they assume that a person look at the front. To achieve face recognition robust to pose variation, we propose the adoption of pose classifier before face recognition. The proposed approach consists of the two main parts: the first part is to estimate and classify the pose of the test face image and the second part is to recognize a person registered in database based on the pose-classified face image. In the initial step of the pose estimation, gallery face images are classified according to the angle of face pose in order to construct a number of the pose database each containing only the images for specific pose and then PCA or (2D)²PCA is performed to compute major eigen-faces for each pose database. Then the pose of a test image can be classified by projecting the test image onto the PCA eigen-face space for each pose and identifying the index of the PCA space to which the Euclidean distance from the test image is minimum.

After the pose of the test image is classified, face recognition is performed to identify person against the pre-built face database with specific pose. In this work, a polynomial-based RBFNNs pattern classifier is proposed as a means to recognize face and its performance is compared with existing face recognition methods including PCA-based algorithms. Face recognition suffer from performance degradation due to the mismatch of pose angle between the test image and the classified faces in database. To solve this problem, in the process of pose classification, we present a novel method to measure the

degree of pose match between a test face and the classified faces in database with a specific pose. From the information on the degree of pose match, we can select a more suitable face image for face recognition among many candidate face images, e.g., from a sequence of face images from video.

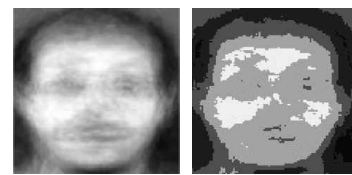
This paper is organized as follows. In Section 2, we introduce dimensional reduction algorithm PCA and (2D)²PCA for extraction the facial features. In Section 3, we propose the pose classification of face image using multi-space PCA and extraction of images with similar pose [12]. In Section 4, we propose a general architecture of RBFNNs. In Section 5, simulation and experimental results are presented. Finally, conclusions are covered in Section 6.

2. Dimensionality Reduction Algorithm

2.1 Principal Component Analysis (PCA)

Face images extracted from video typically constitute a very high-dimensional feature space. To improve the performance and speed of the learning, the dimensionality reduction of feature space is executed by using PCA (Principal Component Analysis) which is a widely adopted method to minimize loss of the information.

The main idea of the PCA is to find vectors that best account for the distribution of face images within the entire image space. Because these vectors are the eigenvectors of the covariance matrix corresponding to the original face images, and because they are face-like appearance, we refer to them as ‘eigenfaces’ (see Fig. 1).



(a) Average face (b) Eigen face

Fig. 1. Comparison of average and eigen faces

2.2 2-dimensional 2-directional PCA (2D)²PCA

Consider an m by n matrix A from randomly extracted image. Let $X \in R^{n \times d}$ be a matrix with orthonormal columns, $n \geq d$. Projecting A onto X yields an m by d matrix $Y=AX$. In 2DPCA, the total scattering of the projected samples was used to determine a sound projection matrix X . In other words, the following criterion is adopted:

$$\begin{aligned}
 J(X) &= \text{trace}\{E[(Y-EY)(Y-EY)^T]\} \\
 &= \text{trace}\{E[(AX-E(AX))(AX-E(AX))^T]\} \\
 &= \text{trace}\{E[(AX-E(AX))(AX-E(AX))^T]X\}
 \end{aligned} \tag{1}$$

where the last term in (1) results from the fact that $trace(AB)=trace(BA)$, for any two matrices. Define the image covariance matrix $G = E[(A-EA)^T(A-EA)]$, which is an n by n nonnegative definite matrix. Suppose that there are M training face images, denoted by m by n matrices A_k ($k=1,2,\dots,M$), and denote the average image as \bar{A} . Then G can be expressed as

$$G = \frac{1}{M} \sum_{k=1}^M (A_k - \bar{A})^T (A_k - \bar{A}) \quad (2)$$

It has been proven that the optimal value for the projection matrix X_{opt} is composed by the orthonormal eigenvectors X_1, \dots, X_d of G corresponding to the d largest eigenvalues, i.e. $X_{opt} = [X_1, \dots, X_d]$. Because the size of G is only n by n , computing its eigenvectors is very efficient. Also, like in PCA the value of d can be controlled by setting a threshold as follows

$$\sum_{i=1}^d \lambda_i / \sum_{i=1}^n \lambda_i \geq \theta \quad (3)$$

where $\lambda_1, \lambda_2, \dots, \lambda_n$ is the n biggest eigenvalues of and G and θ is a pre-set threshold.

However, the 2DPCA only works either in the row or column direction of images respectively. That is, 2DPCA learns an optimal matrix X from a set of training images reflecting information between rows of images, and then projects an m by n image A onto X , yielding an m by d matrix $Y=AX$. Similarly, the alternative 2DPCA learns optimal matrix Z reflecting information between columns of images, and then projects A onto Z , yielding a q by n matrix $B=Z^T A$.

The (2D)²PCA algorithm simultaneously use the projection matrices X and Z [6]. Consider that we have obtained the projection matrices X and Z , projecting the m by n image A onto X and Z simultaneously, yielding a q by d matrix C

$$C = Z^T AX \quad (4)$$

The matrix C is also called the coefficient matrix in image representation, which can be used to reconstruct the original image A in the following form

$$\hat{A} = ZCX^T \quad (5)$$

When used for face recognition, the matrix C is also called the feature matrix. After projecting each training image A_k ($k=1, 2, \dots, M$) onto X and Z , we obtain the training feature matrices C_k ($k=1, 2, \dots, M$). Given a test face image A , first use Eq. (6) to get the feature matrix C , then a nearest neighbor classifier is used for classification. Here the distance between C and C_k is defined by

$$d(C, C_k) = \|C - C_k\| = \sqrt{\sum_{i=1}^c \sum_{j=1}^d (C^{(i,j)} - C_k^{(i,j)})^2} \quad (6)$$

3. Pose Classification of Face Image

3.1 Pose classification using multi-space PCA

In order to alleviate the performance degradation of face recognition caused by the pose change, the pose estimation is considered as an essential step carried out prior to the face recognition, and then face recognition is performed by using pose-classified face images for improving face recognition performance. In order to solve the problem of pose variations, we use the preprocessing algorithm called PCA or (2D)²PCA. For the experiment, we collect the samples by considering the five poses (left 90°, left 45°, front, right 45°, and right 90°). After constructing database, the preprocessing algorithm is separately carried out per each pose to obtain the eigen-face of each angle. The eigen-face is used to classify facial pose(angle) of unknown image(i.e. testing dataset).

Face images are classified according to the angle of yaw($\pm 90^\circ, \pm 45^\circ, 0^\circ$) to construct image database for each pose as shown in Fig. 2. For each image database, PCA or (2D)²PCA are performed to compute the eigen-face vector of each pose.

After that, in order to classify the pose of a test image, we project the image onto the PCA space for each pose, and identify the PCA space to which the Euclidean distance from the test image is the minimum.

In the course of computing the distance to each PCA space, we found that the utilization of only the major principal components yields better results; that is, instead of utilizing all the eigenvectors from each PCA, only a few eigenvectors corresponding to the highest Eigenvalues are involved to calculate the distance to each PCA space.

The procedure of pose classification through pose estimation is outlined as follows.

[Step1] Generate Eigenvector for each pose using PCA or (2D)²PCA.

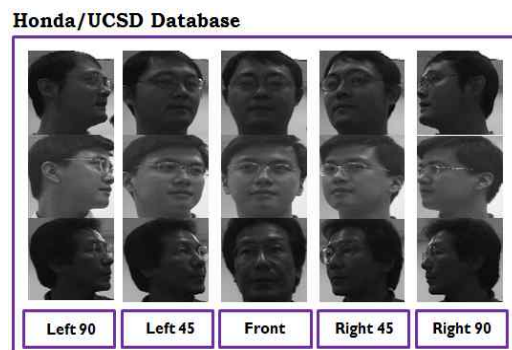
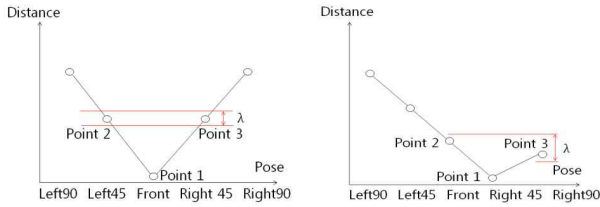


Fig. 2. Configuration of multi-space PCA



(a) Similar pose for database (b) Non-similar for database
Fig. 3. Determination of distance between adjacent poses

[Step 2] Extract the major principal components characteristic to each pose

[Step 3] Estimate the closest pose by projecting the candidate image via the major principal components of each pose and by identifying the PCA space in which the Euclidean distance is the minimum.

3.2 Similar pose estimation for database

Although the pose of the extracted face image can be classified as one of five possible poses using the method described in Section 3.1, when performing face recognition in the classified database of each pose, the performance degradation may occur by the mismatch of exact pose angle between the classified images in database and the candidate image. To solve this problem, the extracted image is only used for face recognition when it is considered to be similar to the face images in the classified database within a predefined bound. When a candidate face is in a certain pose, the distance to each PCA space shows quadratic curves shown in Fig. 3.

Suppose a candidate is in the Front pose. Then, as shown in Fig. 3(a), the distance to Front PCA space (Point 1) is minimum, and the distance to Left 45 PCA space (Point 2) and Right 45 PCA space (Point 3) are the same. In this case the value of λ , defined as the difference of distance between Point 2 and Point 3, become very small.

On the other hand, if the candidate has the pose located somewhere in the middle of Front and Right 45, as depicted in Fig. 3(b), then the value of λ , the difference of the distance to Left 45 PCA space (Point 2) and Right 45 PCA space(Point 3) becomes significant. In this case, the face image is recognized as non-similar image with database. Therefore by monitoring the value of λ against a predefined bound we decide whether the candidate face has the exactly same angle to the face images in the classified pose database.

4. Structure of Polynomial-based RBFNNs and Entire Face Recognition Architecture

4.1 Structure of polynomial-based RBFNNs

In Section 4, the proposed polynomial-based RBFNNs pattern classifier is considered as a means to identify

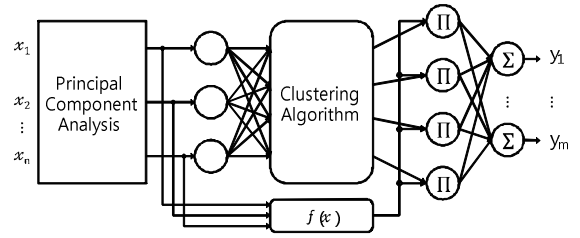


Fig. 4. Overall architecture of polynomial-based RBFNNs pattern classifier

recognition performance.

The proposed RBFNNs is implemented by realizing three processing phases that is, condition, conclusion, and aggregation phases. Condition and conclusion phases relate to the formation of the fuzzy rules and their following analysis. Aggregation phase is concerned with a fuzzy inference. The FCM clustering algorithm is used to form information granules. In the conclusion phase, connection weights are realized with the aid of four types of polynomials such as constant, linear, quadratic and modified quadratic.

The output produced by the model is expressed in the following form

$$\hat{y} = \sum_{i=1}^c u_i f_i(x_1, \dots, x_k) \quad (7)$$

In a nutshell, the rules are conditional (if-then) statements

$$IF \ x \text{ is } A_i \ \text{Then } f_{ij}(x) \quad (8)$$

4.2 Fuzzy C-Means (FCM) clustering

Consider a set X composed of N vectors located in a certain n -dimensional Euclidean space, that is, $X = \{x_1, x_2, \dots, x_N\}$, $x_k = \{x_{k1}, x_{k2}, \dots, x_{kn}\} \in R^n$, $1 \leq k \leq N$, $1 \leq j \leq n$, where N is the number of input data, n is the number of variables. Clustering results in the assignment of the input vectors $x_k \in X$ into c clusters where the clusters are represented by the prototypes (center values) $v_i = \{v_{i1}, v_{i2}, \dots, v_{in}\} \in R^n$, $1 \leq i \leq c$. Let R^c denote the set of real $c \times 1$ matrices with the entries in $[0, 1]$. Then $U = [u_{ik}] \in R^c$. The level of assignment of x_k [X to the i -th cluster] is expressed by the membership function $u_{ik} = u_i(x_k)$. Fuzzy partitions U of X satisfy the condition:

$$0 < \sum_{k=1}^N u_{ik} < N, \quad 1 \leq i \leq c, \quad \sum_{i=1}^c u_{ik} = 1, \quad 1 \leq k \leq N \quad (9)$$

The FCM algorithm develops the structure in data by minimizing the following objective function [10].

$$J_m = \sum_{i=1}^c \sum_{k=1}^N (u_{ik})^r d(x_k, v_i), \quad 1 < r < \infty \quad (10)$$

In (10), r is referred to as a fuzzification coefficient, $d(\mathbf{x}_k, \mathbf{v}_i)$ is a distance between the input vector $\mathbf{x}_k \in \mathbf{X}$ and the prototype (centroid) $\mathbf{v}_i \in \mathbf{R}^c$. Quite commonly it comes in the form of the weighed Euclidean distance computed between \mathbf{x}_k and \mathbf{v}_i and defined as follows:

$$d(\mathbf{x}_k, \mathbf{v}_i) = \|\mathbf{x}_k - \mathbf{v}_i\|^2 = \sum_{j=1}^n \frac{(x_{kj} - v_{ij})^2}{\sigma_j^2} \quad (11)$$

where σ_j^2 is the variance of the j -th input variable. This type of distance equipped with the weights allows us to deal with variables of different variability.

Under the assumption of the form of the weighted Euclidean distance, the necessary conditions for solutions (\mathbf{U}, \mathbf{V}) of $\min \{J_m(\mathbf{U}, \mathbf{V})\}$ are specified as:

$$u_{ik} = w_{ik} \frac{1}{\sum_{j=1}^c \left(\frac{d(\mathbf{x}_k, \mathbf{v}_j)}{d(\mathbf{x}_k, \mathbf{v}_i)} \right)^{2/(r-1)}} \quad (12)$$

4.3 Weighted Least Square Estimation (WLSE)

In the existing fuzzy inference system, Least Square Estimation (LSE) as global training estimation is mainly used to estimate the values of the parameters standing in the conclusion part of the rules. But LSE is known to cause over-fitting with the increase of the number of rules and input. It is because parameters of each rule are estimated at the same time. To achieve this problem, we use Weighted Least Square Estimation (WLSE) as local training estimation to obtain parameters of each rule respectively.

The WLSE criterion comes in the form

$$Q_w = \sum_{i=1}^c (\mathbf{Y} - \mathbf{X}\mathbf{A}_i)^T \mathbf{U}_i (\mathbf{Y} - \mathbf{X}\mathbf{A}_i) \quad (13)$$

where \mathbf{A}_i is polynomial parameter of i -th, \mathbf{Y} is output data and \mathbf{U}_i is membership value of input-data for i -th cluster. \mathbf{X}_j is matrix of input data for estimating parameter of i -th local model. The linear model is defined in the form.

$$\mathbf{X} = \begin{bmatrix} 1 & x_{11} & \cdots & x_{1n} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & x_{N1} & \cdots & x_{Nn} \end{bmatrix}, \quad \mathbf{U}_i = \begin{bmatrix} u_{i1} & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & u_{iNi} \end{bmatrix} \quad (14)$$

Where, N stands for the number of data and parameters of the polynomial for rule of i -th is as follows:

$$\mathbf{A}_i = (\mathbf{X}^T \mathbf{U}_i \mathbf{X})^{-1} \mathbf{X} \mathbf{U}_i \mathbf{Y} \quad (15)$$

Polynomial-based RBFNNs pattern classifier has been shown to demonstrate many tangible advantages with regard to learning abilities, generalization aspects, and robustness.

4.4 Optimization process of the RBFNN classifier

PSO is originally proposed by Kennedy and Eberhart and was first intended for simulating social behavior as a representation of the movement of organisms in a bird flock or fish school. The algorithm was simplified and it was observed to be performing optimization. PSO comes as a sound solution to realize search in complex problems [13].

The design framework of the proposed classifier consists of the following steps.

[Step 1] Determine system's input variables

Define input variables x_1, x_2, \dots, x_n related to the output variable y .

[Step 2] Form training, validation and testing data.

The input-output data $(\mathbf{x}_i, y_i) = (x_{1i}, x_{2i}, \dots, x_{ni}, y_i)$, $i=1, 2, \dots, N$ (with N being the total number of data points) is split into three subsets that is, a training, validation and testing dataset. The training data is used to construct the classifier. Next, the validation and testing data is used to evaluate the quality of the classifier.

[Step 3] Determine the parameters of RBFNN structure using PSO algorithm

The optimal architecture of the RBFNN is determined by using the PSO algorithm. The PSO is available in the RBFNN structure by using a particle of PSO. As shown there, the design of optimal parameters available within the RBFNN at last leads to a structurally and parametrically optimized classifier, which is more flexible as well as simpler than the underlying RPNN formed without any optimization.

[Step 4] Construction and evaluation of classifier

To evaluate the performance of the RBFNN classifier, the training, validation dataset and testing data are considered. The classification rate is calculated and used as the fitness value of objective function of PSO. The classification rate and objective function of PSO are expressed in the form

$$ClassificationRate[\%] = \left(1 - \frac{F}{N} \right) \times 100 \quad (16)$$

$$Objective\ function = \frac{(TR + VA)}{2} \quad (17)$$

Where F is the total number of false classification cases and, N is the number of the data. In the experiments, TR denotes the classification rate for the training data, VA means the classification rate for the validation data, and TE stands for the classification rate for the testing data.

[Step 5] Check the termination criterion

Once the termination criterion has been satisfied, the optimization process becomes completed. At this point, the particle with the highest fitness is selected (the best particle), and its content is used to form the classifier. Otherwise, the sequences [step 3] - [step 4] are repeated until the termination criterion has been met.

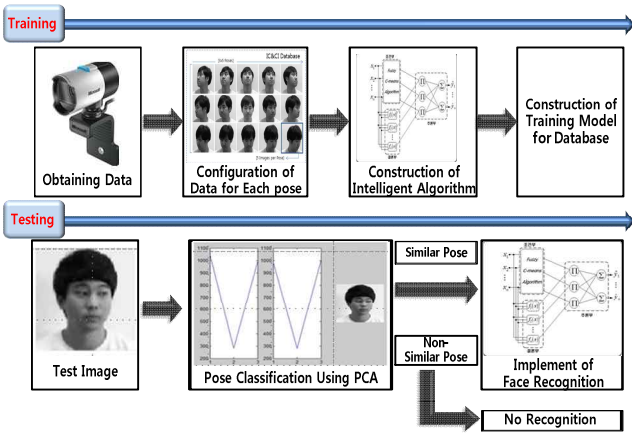


Fig. 5. Overall flow of the 2-dimensional RBFNN based face recognition by using automatic pose classification-based classification

4.5 Architecture of entire face recognition system

In this study, the recognition performance of RBFNNs-based pattern classifier is compared with that of PCA based classifiers for designing robust 2-dimensional face recognition system to pose variation. Firstly, face images with various pose are extracted from Honda/UCSD, Cambridge head pose and IC&CI database. A part of them is used to construct and to evaluate the database for each pose. In the course of constructing the database for each pose, the pose of remaining face images is classified by the method that is based on the PCA or $(2D)^2PCA$ after the pose of some initial faces has been classified manually.

In the evaluation phase, the pose of the test image is classified based on the PCAs or $(2D)^2PCA$, and then the face recognition is performed for each pose, to compare the performance of each pattern classifier based on PCA, $(2D)^2PCA$ and RBFNNs. A flowchart as shown in Fig. 5 is overall flow of the 2-dimensional RBFNNs based face recognition by using automatic pose estimation-based classification.

5. Simulation and Experimental Results

5.1 Experiment

In this part, we experiment RBFNNs based face recognition system by using Honda/UCSD, Cambridge head pose, and Intelligent Control & Computational Intelligence (IC&CI) databases. As depicted in Fig. 6, 5-fold cross validation(5-kcv) is used to obtain quantitative results and the performance is represented as the mean and its standard deviation.

Also, setting of initial optimization parameters and search range of parameters for efficient training is summarized as shown in Table 2.

Table 2. Setting of initial optimization parameters and search range of parameters

	Parameters	Values
PSO	No. of generations	100
	Swarm size	60
	$[W_{min}, W_{max}]$	$[0.4 \ 0.9]$
	c_1, c_2	2.0
Search Range	Fuzzification coefficient	$[1.1 \sim 3.0]$
	Polynomial type	$[1 \sim 4]$
	No. of nodes	$[2 \sim 10]$

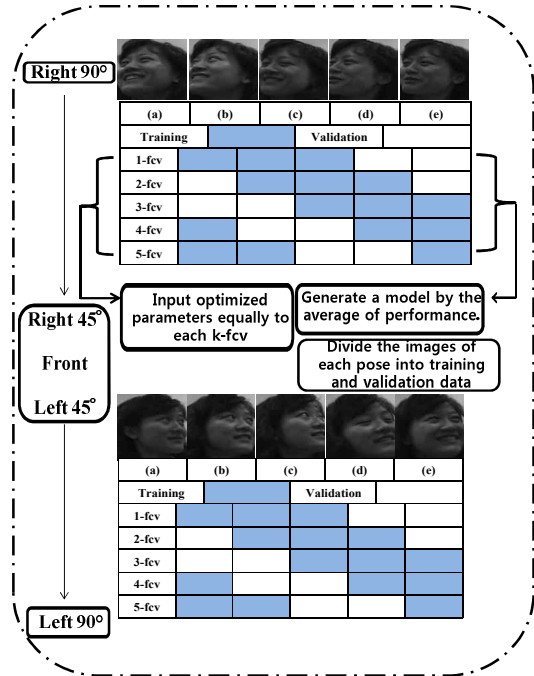


Fig. 6. Generation of the proposed model by k-fold cross validation

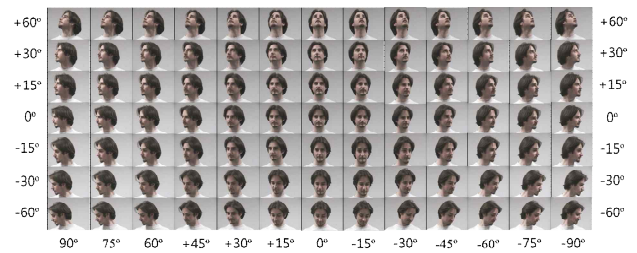


Fig. 9. Samples of face images in the Cambridge Head Pose database

5.2 Face recognition system using honda/UCSD database

5.2.1 Configuration of face recognition system

First, we have conducted face recognition experiment using Honda/UCSD database [8]. Training data consist of 20 people with 5 poses (left 90°, left 45°, front, right 45°, right 90°) per person and 5 face images per pose, which

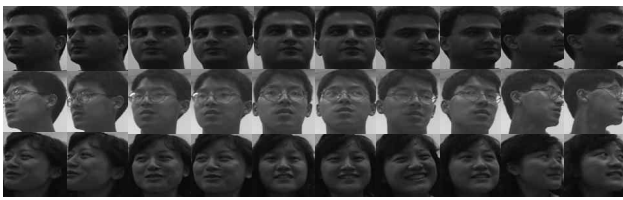


Fig. 7. Samples of face images in the Honda/UCSD database

1 st Candidate	2 nd Candidate	3 rd Candidate	4 th Candidate	5 th Candidate	6 th Candidate	7 th Candidate	
							<input type="radio"/>
<input type="radio"/>	<input checked="" type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
8 th Candidate	9 th Candidate	10 th Candidate	11 th Candidate	12 th Candidate	13 th Candidate	14 th Candidate	
							<input type="radio"/>
<input type="radio"/>	<input checked="" type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
15 th Candidate	16 th Candidate	17 th Candidate	18 th Candidate	19 th Candidate	20 th Candidate	Success	
						50 / 60	<input type="radio"/>
<input checked="" type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	10 / 60	<input type="radio"/>
<input type="radio"/>	<input checked="" type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Failure	<input type="radio"/>
[Ex.]							Classification rate(%)
							83.33%

(a) Pose estimation of test data using PCA

1 st Candidate	2 nd Candidate	3 rd Candidate	4 th Candidate	5 th Candidate	6 th Candidate	7 th Candidate	
							<input type="radio"/>
<input type="radio"/>	<input checked="" type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
8 th Candidate	9 th Candidate	10 th Candidate	11 th Candidate	12 th Candidate	13 th Candidate	14 th Candidate	
							<input type="radio"/>
<input checked="" type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
15 th Candidate	16 th Candidate	17 th Candidate	18 th Candidate	19 th Candidate	20 th Candidate	Success	
						52 / 60	<input type="radio"/>
<input type="radio"/>	<input checked="" type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	8 / 60	<input type="radio"/>
<input type="radio"/>	<input checked="" type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Failure	<input type="radio"/>
[Ex.]							Classification rate(%)
							86.67%

(b) Pose estimation of test data using (2D)²PCA

Fig. 8. Results of pose estimation using both PCA and (2D)²PCA

adds up to a total of 500 face images. For 5-fold cross validation, each of 5 face images per pose is assigned for training and validation as explained previously in Fig. 7. Testing data consist of 20 people with 3 poses per person, thus a total of 60 face images. In preparing the testing data, the face images are deliberately chosen whose pose is slightly different from that of the training data. We refer to these images as ‘Random Right 45, Random Front, and random Left 45.

The procedure of the face recognition experiment is as follows; In PCA based face recognition method (Expt.1), PCA is used for pose estimation and then face recognition is performed using PCA, (2D)²PCA, and the proposed RBFNNs pattern classifier with input feature obtained by PCA and (2D)²PCA. In the proposed method (Expt.2), face recognition is also performed using PCA, (2D)²PCA, RBFNNs pattern classifier with PCA and (2D)²PCA, however, (2D)²PCA is adopted for pose estimation.

5.2.2 Experimental Results

In this experiment, PCA and (2D)²PCA are used for pose estimation of the proposed face recognition system. Fig. 8 show the application of pose estimation for the test data.

Table 3. Comparison of face recognition experiments

Recognition algorithm		RR(%)				
Nearest neighbor(NN) matching in Fisherface [14]		74.2				
NN matching in LLE + k-means clustering [15]		88.3				
Mutual Subspace Method(MSM) [16]		89.8				
Manifold to Manifold Distance (MMD) [17]		93.6				
Proposed recognition system	Pose estimation	Recognition algorithm	Time(sec)		RR(%)	
			TR	TE		
	Expt.1	PCA	PCA	N/A	15.37	74.1
			(2D) ² PCA	N/A	7.458	76.9
	Expt.2	(2D) ² PCA	PCA	N/A	15.04	74.2
			(2D) ² PCA	N/A	6.766	78.7
Expt.2	PCA	PCA+ RBFNNs	903.2	10.13	96.1	
		(2D) ² PCA + RBFNNs	895.6	9.684	96.8	
	(2D) ² PCA	PCA+ RBFNNs	880.1	9.212	98.3	
		(2D) ² PCA + RBFNNs	738.1	8.046	98.5	

RR: recognition rate(%)

The test data consists of 20 people, with 3 poses per person for a total of 60 face images.

The first three face images for a candidate person are input in order of Random Right 45, Random Front, and Random Left 45 and as shown in Fig. 8, and the pose of each image is classified. Red “×” mark denotes that the image is classified incorrectly by pose estimation and 50 images and 52 images are successfully classified as shown in Fig. 8.

The comparison of processing time of each experiment is as shown in Table 3. Firstly, we can see the processing time of the face recognition system with (2D)²PCA is faster than PCA in both the Expt.1 and Expt.2. Also, although Expt.2 requires much more processing time than Expt.1 for training and validation steps in RBFNNs pattern classifier, the processing time for testing is comparable to Expt. 2.

As for the recognition rate, we observe the face recognition performance of the proposed method, Expt.2, is much better than Expt.1 with the aid of robust property of RBFNNs pattern classifier. Also, we observe that the recognition rate with (2D)²PCA is slightly better than with PCA in both cases.

5.3 Face recognition system using cambridge head pose database

5.3.1 Configuration of face recognition system

Cambridge head pose database is configured for 15 candidates. Training and validation data consists of 3 poses including upper and lower poses per person with 182 face images. Also, experimental data is divided into 3 sections such as training, validation, and test data using 5-fold cross validation. The test data is acquired from the video with the exception of training and validation data and a total data consists of 91 poses per each candidate for a total of 182 face images[9].

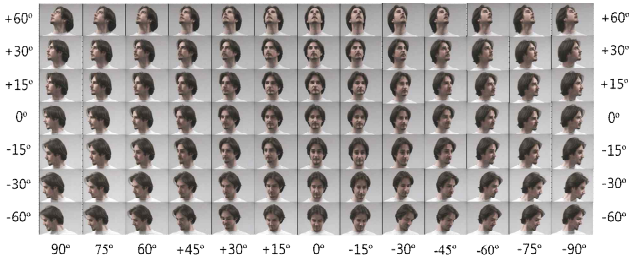


Fig. 9. Samples of face images in the Cambridge Head Pose database

Table 4. Description of face image database used for experiment

(a) Successful results for pose estimation

Candidate	1-fcv	2-fcv	3-fcv	4-fcv	5-fcv
1 st	35/37	36/37	36/37	35/37	36/37
2 nd	32/37	28/37	28/37	25/37	28/37
3 rd	34/37	35/37	36/37	32/37	35/37
4 th	36/37	37/37	35/37	37/37	36/37
5 th	36/37	36/37	35/37	35/37	36/37
6 th	34/37	37/37	36/37	36/37	37/37
7 th	34/37	35/37	35/37	36/37	37/37
8 th	37/37	37/37	34/37	36/37	33/37
9 th	35/37	37/37	36/37	37/37	37/37
10 th	35/37	35/37	32/37	36/37	37/37
11 th	35/37	36/37	34/37	36/37	33/37
12 th	37/37	37/37	37/37	35/37	37/37
13 th	36/37	37/37	37/37	36/37	37/37
14 th	35/37	36/37	32/37	34/37	35/37
15 th	36/37	36/37	36/37	36/37	36/37

Successful mean pose estimation rate : 94.88 %

(b) Pose classification results for successful pose estimation

1-fcv	2-fcv	3-fcv	4-fcv	5-fcv	Average pose classification rate(%)
28/35	35/36	34/36	29/35	36/36	88.47
29/32	29/28	26/28	25/25	26/28	94.52
29/34	34/35	32/36	30/32	33/35	90.80
32/36	34/37	33/35	33/37	35/36	91.34
31/36	34/36	31/35	29/35	33/36	87.30
28/34	33/37	35/36	29/36	32/37	85.55
31/34	34/35	34/35	32/36	36/37	93.60
30/37	33/37	29/34	33/36	30/33	86.06
31/35	35/37	32/36	32/37	35/37	89.46
32/35	33/35	32/32	31/36	35/37	92.44
32/35	35/36	31/34	32/36	32/33	92.25
35/37	34/37	36/37	32/35	36/37	93.82
33/36	33/37	36/37	33/36	36/37	92.61
31/35	34/36	30/32	30/34	33/35	90.78
35/36	35/36	32/36	32/36	34/36	92.50

Mean pose classification rate : 90.77%

The procedure of the experiment using Cambridge head pose database is as follows; Images in this dataset include large variations in out of plane (left/light and up/down) head movement as well as in facial expression. We apply this dataset by using PCA as the preprocessing algorithm in order to perform pose estimation. And we show recognition

rate for several competing methods: our model with using entire test image and using successful pose image for pose classification. Our proposed approach with using successful test image for pose classification outperforms other methods.

5.3.2 Experimental results

We first evaluate the proposed approach on benchmark datasets specifically designed for evaluating the performance of pose classifications (Table 4, (a) and (b)), optimization (Table 5), and recognition (Table 6). Pose classification with Cambridge head pose database is performed by using PCA. Test data consists of 15 people with 37 images for each candidate is classified.

Table 4 shows pose estimation and classification rates on standard video datasets. The pose estimation per each candidate is performed using training dataset as mentioned in Section 3.1. After that, the pose classification is carried out through a comparison between testing dataset and eigen-face obtained from training dataset.

Table 5 shows the optimal parameters for each classifier (Model) optimized by PSO. Interestingly, the fuzzification coefficient and the number of rules are selected differently through optimization, but the type of polynomial is selected as linear function for all classifiers.

Table 5. Parameters for optimized each model of Particle Swarm Optimization

Parameters for optimized each model of particle swarm optimization			
	Model 1 (Left)	Model 2 (Front)	Model 3 (Right)
Fuzzification coefficient	1.594	2.673	1.480
Polynomial type	Linear	Linear	Linear
No. of rules	8	7	9

Table 6 shows the face recognition rate of the RBFNNs classifier optimized by the PSO. The parameters selected by PSO are represented in Table 5.

Table 6. Recognition rate of face recognition system

Recognition rate(%)				
Training and validation			Testing	
Model for each pose	Training AVG±STD	Validation AVG±STD	Testing data type	Testing AVG±STD
Model 1 (Left)	100±0.0	99.08±0.54	Case I	98.38±0.74
Model 2 (Front)	100±0.0	97.92±1.24		
Model 3 (Right)	100±0.0	98.33±0.88	Case II	99.23±0.44

Case I: Using entire testing data without pose estimation;

Case II: Using testing data completed successful pose estimation

The training dataset is used to construct the classifier and the validation dataset is employed to evaluate the generalization ability as well as to alleviate overfitting



Fig. 10 Samples of face images in the IC&CI database

caused by excessive learning during PSO. The generalization ability of the final classifier completed by the optimization is recorded by using testing data. As a result, the performance of Case II after the pose estimation is superb than that of Case I without pose estimation.

5.4 Face recognition system using IC&CI database

5.4.1 Configuration of face recognition system

IC&CI database was made by Intelligent Control and Computational Intelligence laboratory, at the University of Suwon and it is configured for 15 candidates. Training and validation data consists of 5 poses (left 90°, left 45°, front, right 45°, right 90°) including upper and lower poses per person for a total of 1125 face images. Also, experimental data is divided into training, validation, and test data using 5-fold cross validation. Test data is randomly extracted from the video for 15 candidates and it consists of 15 poses per each candidate for a total of 450 face images.

The procedure of the face recognition experiment using IC&CI database is as follows. PCA and (2D)²PCA are adopted for pose estimation and face recognition is performed using PCA, (2D)²PCA and the proposed RBFNNs pattern classifier with input feature obtained by PCA and (2D)²PCA.

5.4.2 Experimental results

In this study, there are two steps composed of pose estimation and face recognition. Moreover, preprocessing algorithm is used in both parts. In the estimation part, PCA or (2D)²PCA is employed for pose estimation based on the eigen-faces obtained from the preprocessing, whereas in the face recognition, PCA or (2D)²PCA serves as the dimensional reduction on pose-classified face images. In this experiment, testing datasets consist of 15 people with 450 images per each candidate. Experiment 1(Expt. 1) means that only 5 poses images without using the upper and lower images are used in the experiment, whereas, Experiment 2(Expt. 2) means that 5 poses including the upper and lower images are employed in the experiment. As a result, the pose classification using (2D)²PCA outperforms PCA on all experiments, especially in Expt. 1, the pose classification is improved over 8%. In addition, when using the datasets including the upper and lower images, the pose classification is higher than in the case of

Table 7. Performance results of face recognition system

	Pose estimation algorithm	CR(%)	Face recognition algorithm	RR(%)
Expt.1	PCA	76.35	PCA + RBFNNs	97.84
			(2D) ² PCA + RBFNNs	98.36
	(2D) ² PCA	84.61	PCA + RBFNNs	97.86
			(2D) ² PCA + RBFNNs	98.32
Expt.2	PCA	98.15	PCA + RBFNNs	98.38
			(2D) ² PCA + RBFNNs	98.57
	(2D) ² PCA	98.26	PCA + RBFNNs	98.14
			(2D) ² PCA + RBFNNs	98.46

CR: Classification rate(%); RR: Recognition rate(%)

simple dataset, but interestingly, the face recognition rate is similar to each other.

6. Conclusion

In this study, we have developed the face recognition system robust to pose variation. To do this, pose estimation part was introduced before face recognition part. Moreover, we compared the performance of both PCA and (2D)²PCA used for pose estimation and face recognition. The (2D)²PCA used in the preprocessing part exhibits several tangible advantages over the PCA. Since (2D)²PCA is simple and straightforward from aspect of feature extraction as well as it is better than PCA in terms of the resulting recognition rate. In the recognition part, the proposed RBFNNs pattern classifier exhibit useful characteristics. The RBFNNs involves a partition module formed by the FCM clustering and used here as an activation function of the neurons located in the hidden layer. The resulting model is endowed with polynomial weights. Given this, the network is capable of generating more complex nonlinear discriminant functions. The estimation of some parameters of the RBFNNs such as fuzzification coefficient, the number of rules and polynomial type by means of particle swarm optimization (PSO) contributes to the effectiveness of the design process. Owing to this form of optimization, we can produce a classifier that can effectively cope with the nonlinearity of the classifier. The RBFNNs could be of interest as effective constructs for handling pattern classification problems of high-dimensionality and involving various pose variations. To handle these tasks, it would be advantageous to consider feature aspects including the PCA and (2D)²PCA using particle swarm optimization.

Acknowledgements

This work was supported by the GRRC program of Gyeonggi province [GRRC Suwon 2017-B2, Center for U-city Security & Surveillance Technology and also partly supported by Basic Science Research Program through the

National Research Foundation of Korea(NRF) funded by the Ministry of Science, ICT & Future Planning (NRF-2015R1A2A1A15055365) and the Ministry of Education (NRF-2017R1D1A1B03032333)

References

- [1] M. Balasubramanian, S. Palanivel and V. Ramalingam, "Real Time Face and Mouth Recognition Using Radial Basis Function Neural Networks," *Expert Systems with Applications*, vol. 36, pp. 6879-6888, April 2009.
- [2] Wei Huang, Jinsong Wang, Jiping Liao, A granular classifier by means of context-based similarity clustering," *Journal of Electrical Engineering & Technology*, vol. 11, no. 5, pp. 1383-1394, 2016.
- [3] Wen-zhun Huang, and Shan-wen Zhang, "A Novel Face Recognition Algorithm based on the Deep Convolution Neural Network and Key Points Detection Jointed Local Binary Pattern Methodology," *Journal of Electrical Engineering & Technology*, vol. 12, no. 1, pp. 363-372, 2017.
- [4] Erik Murphy-Chutorian and Mohan Manubhai Trivedi "Head Pose Estimation in Computer Vision: A Survey," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 31, no. 4, pp. 607-626, April 2009.
- [5] Meng Joo Er, Shiqian Wu, Juwei Lu and Hock Lye, "Face Recognition with Radial Basis Function (RBF) Neural Networks," *IEEE Trans. on Neural Networks*, vol. 13, no. 3, pp. 697-710, May 2002.
- [6] Daoqiang Zhang and Zhi-Hua Zhou, "Two-directional Two-dimensional PCA for Efficient Face Representation and Recognition," *Neurocomputing*, vol. 69, no. 1-3, pp. 224-231, Dec. 2005.
- [7] Minyoung Kim, Sanjiv Kumar, Vladimir Pavlovic and Henry Rowley, "Face Tracking and Recognition with Visual Constraints in Real-World Videos", *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference*, June 2008.
- [8] K. C. Lee, J. Ho, M. H. Yang and D. Kriegman, "Honda UCSD Video Database", Available: <http://vision.ucsd.edu/content/honda-ucsd-video-database>, 2005, [Accessed: Oct 22, 2013]
- [9] Sujith Srinivasan and Kim L. Boyer "Head Pose Estimation Using View Based Eigenspaces," in *Proceedings of 16th International Conference on Pattern Recognition*, Aug. 2002.
- [10] Sung-Kwun Oh, Sung-Hoon Yoo and Witold Pedrycz, "Design of Face Recognition Algorithm Using PCA-LDA Combined for Hybrid Data Pre-processing and Polynomial-based RBF Neural Networks: Design and Its Application" *Expert Systems with Applicants*, vol. 40, no. 5, pp. 1451-1466, April 2013.
- [11] Ergun Gumus, Niyazi Kilic, Ahmet Sertbas and Osman N. Ucan, "Evaluation of Face Recognition Technique Using PCA, Wavelets and SVM," *Expert Systems with Applications*, vol. 37, no. 9, pp. 6404-6408, Sept. 2010.
- [12] Sung-Hoon Yoo, Sung-Kwun Oh and Witold Pedrycz, "Optimized Face Recognition Algorithm Using Radial Basis Function Neural Networks and Its Practical Applications," *Neural Networks*, vol. 69, pp. 111-125, Sept. 2015.
- [13] James Kennedy and Russell Eberhart, "Particle Swarm Optimization," in *Proceedings of IEEE International Conference on Neural Networks*, Dec. 1995.
- [14] Peter N. Belhumeur, Joao P. Hespanha and David J. Kriegman, "Eigenfaces vs. Fisherfaces: Recognition Using Class Specific Linear Projection," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 19, no. 7, pp. 711-720, July 1997.
- [15] A. Hadid and M. Pietikäinen, "From Still Image to Video-Based Face Recognition: An Experimental Analysis," in *Proceedings of Sixth IEEE International Conference on Automatic Face and Gesture Recognition*, May 2004.
- [16] Osamu Yamaguchi, Kazuhiro Fukui and Ken-ichi Maeda, "Face Recognition Using Temporal Image Sequence," in *Proceedings Third IEEE International Conference on Automatic Face and Gesture Recognition*, April 1998.
- [17] Ruiping Wang, Shiguang Shan, Xilin Chen and Wen Gao, "Manifold-Manifold Distance with Application to Face Recognition based on Image Set" *IEEE Conference on Computer Vision and Pattern Recognition*, June 2008.



Eun-Hu Kim He received his B.Sc., M.Sc., and Ph.D. degrees in electrical engineering from Suwon University, Hwaseong-si, Korea, in 2009, 2011, and 2016, respectively. He is currently a postdoctoral fellow with the Department of electrical engineering, Suwon University, Korea. His research interests include fuzzy set, neural networks, evolutionary algorithm, computational intelligence and statistical learning.



Bong-Youn Kim He received his B.Sc., and M.Sc. degrees in electrical engineering from Suwon University, Hwaseong-si, Korea, in 2014, and 2016 respectively. His research interests include fuzzy inference system, neural networks, pattern recognition, face recognition, and evolutionary algorithm.



Sung-Kwun Oh He received his B.Sc., M.Sc., and Ph.D. degrees in electrical engineering from Yonsei University, Seoul, Korea, in 1981, 1983, and 1993, respectively. During 1983-1989, he was a Senior Researcher of R&D Lab. of Lucky-Goldstar Industrial Systems Co., Ltd. From 1996 to 1997, he was a

Postdoctoral Fellow with the Department of Electrical and Computer Engineering, University of Manitoba, Winnipeg, MB, Canada. He is currently a Professor with the Department of Electrical Engineering, University of Suwon, Suwon, South Korea. His research interests include fuzzy system, fuzzy-neural networks, automation systems, advanced computational intelligence, and intelligent control. He currently serves as an Associate Editor of the KIEE Transactions on Systems and Control, International Journal of Fuzzy Logic and Intelligent Systems of the KFIS, and Information Sciences.



Jin-Yul Kim He received his BA degree in Electronics Eng. from Seoul National University in 1986, and MA degree and Ph.D. degree in Electrical Eng. from Korea Advanced Institute of Science and Technology, in 1988 and 1993, respectively. He is a Professor with the Dept. of Electronic Eng.,

University of Suwon, South Korea. His research interests include image processing, visual object tracking and object recognition.