Pattern Recognition 54 (2016) 116–127

Contents lists available at ScienceDirect

Pattern Recognition

journal homepage: www.elsevier.com/locate/pr

Generalized mean for robust principal component analysis

Jiyong Oh, Nojun Kwak*

Graduate School of Convergence Science and Technology, AICT, Seoul National University, 1 Gwanak-ro, Gwanak-gu, Seoul 08826, Republic of Korea

ARTICLE INFO

Article history:
Received 11 August 2014
Received in revised form
9 December 2015
Accepted 4 January 2016
Available online 16 January 2016

Keywords:
Generalized mean
Principal component analysis
Robust PCA
Dimensionality reduction

ABSTRACT

In this paper, we propose a robust principal component analysis (PCA) to overcome the problem that PCA is prone to outliers included in the training set. Different from the other alternatives which commonly replace L_2-norm by other distance measures, the proposed method alleviates the negative effect of outliers using the characteristic of the generalized mean keeping the use of the Euclidean distance. The optimization problem based on the generalized mean is solved by a novel method. We also present a generalized sample mean, which is a generalization of the sample mean, to estimate a robust mean in the presence of outliers. The proposed method shows better or equivalent performance than the conventional PCAs in various problems such as face reconstruction, clustering, and object categorization.

1. Introduction

Dimensionality reduction [1] is a classical problem in pattern recognition and machine learning societies, and numerous methods have been proposed to reduce the data dimensionality. Principal component analysis (PCA) [2] is one of the most popular unsupervised dimensionality reduction methods which tries to find a subspace where the average reconstruction error of the training data is minimized. It is useful in representation of input data in a low dimensional space and it has been successfully applied to face recognition [3,4], visual tracking [5], clustering [6,7], and so on.

When automatically collecting a large data set, outliers may be contained in the collected data since it is very difficult to examine whether each sample of data is outlier or not [8]. It is well known that, in this case, the conventional PCA is sensitive to outliers because it minimizes the reconstruction errors of training data in terms of the mean squared error and a few outliers with large errors dominate the objective function. This problem has been addressed in many studies [8–16]. Among them, some studies utilized L_1-norm instead of L_2-norm in the formulation of optimization problem to improve the robustness of PCA against outliers [9–11]. In [9], the cost function for optimization was constructed based on L_1-norm and a convex programming was employed to solve the problem. R_1-PCA [10] was presented to obtain a solution with the rotational invariance, which is a fundamental desirable property for learning algorithms [17]. In [11], PCA-L_1 was proposed, which maximizes an L_1 dispersion in the reduced space and an extension of PCA-L_1 using L_p-norm with arbitrary p was also proposed in [14]. Other method utilizing L_p-norm was also presented in [15]. On the other hand, some of robust PCAs were recently developed using information theoretic measures [12,13]. He et al. [12] proposed MaxEnt-PCA which finds a subspace where Renyi's quadratic entropy [18] is maximized. Renyi's entropy was estimated by a non-parametric Parzen window technique. In [13], HQ-PCA was developed based on the maximum correntropy criterion [19].

In this paper, we propose a new robust PCA method based on the power mean or the generalized mean [20], which can become the arithmetic, geometric, and harmonic means depending on the value of its parameter. The proposed method, *PCA-GM*, is a generalization of the conventional PCA by replacing the arithmetic mean with the generalized mean. The proposed method can effectively prevent outliers from dominating objective function by controlling the parameter in the generalized mean. Moreover, it is rotational invariant because it still uses the Euclidean distance as the distance measure between data samples. In doing so, we also propose a generalized sample mean, which is an enhancement of the conventional algebraic sample mean against outliers to address the problem that the sample mean is easily affected by outliers. It is used in the proposed PCA-GM instead of the arithmetic mean. The optimization problems based on the generalized mean are efficiently solved using a mathematical property of the generalized mean. Recently, Candés et al. proposed a robust PCA [21], which is sometimes referred to as RPCA in the literature, where data matrix is tried to be represented as a sum of a low rank matrix, which corresponds to reconstructions of data, and a sparse matrix, which corresponds to reconstruction errors different from the methods mentioned above. It can model pixel-wise noise effectively using the sparse matrix, thus it has been known that RPCA is useful in the applications such as background modeling from surveillance video and removing shadows and specularities from face images [21] by

* Corresponding author. Tel.: +82 31 888 9166; fax: +82 31 888 9148.
E-mail addresses: yong97@snu.ac.kr (J. Oh), nojunk@snu.ac.kr (N. Kwak).

http://dx.doi.org/10.1016/j.patcog.2016.01.002

using each element in the reconstruction error vector (the column of the sparse matrix). On the other hand, in this paper, we will utilize distance metric in removing the effect of outliers like the previously mentioned methods, and an entire sample is considered as an outlier if it has a large norm of the reconstruction error vector.

The remainder of this paper is organized as follows. Section 2 briefly introduces PCA and the state-of-the-art robust PCAs. The proposed method is described in Section 3. It is demonstrated in Section 4 that the proposed method gives better performances in face reconstruction and clustering problems than other variants of PCA. Finally, Section 5 concludes this paper.

2. PCA and robust PCAs

Let us consider a training set of N n-dimensional samples $\{lbracesupsub\mathbf{x}_i\}_{i=1}^N$. Assuming that the samples have zero-mean, PCA is to find an orthonormal projection matrix $\mathbf{W}\in\mathbb{R}^{n\times m}$ $(m\ll n)$ by which the projected samples $\left\{\mathbf{y}_i=\mathbf{W}^T\mathbf{x}_i\right\}_{i=1}^N$ have the maximum variance in the reduce space. It is formulated as follows:

$$\mathbf{W}_{PCA}=\arg\max_{\mathbf{W}}\ \mathrm{tr}(\mathbf{W}^T\mathbf{S}\mathbf{W}),$$

where $\mathbf{S}=\frac{1}{N}\sum_{i=1}^N\mathbf{x}_i\mathbf{x}_i^T$ is a sample covariance matrix and $\mathrm{tr}(\mathbf{A})$ is the trace of a square matrix $\mathbf{A}$. The projection matrix $\mathbf{W}_{PCA}$ can be also found from the viewpoint of projection errors, i.e., it minimizes the average of the squared projection errors or reconstruction errors. Mathematically, it is represented as the optimization problem minimizing the following cost function:

$$J_{L_2}(\mathbf{W})=\frac{1}{N}\sum_{i=1}^N\|\mathbf{x}_i-\mathbf{W}\mathbf{W}^T\mathbf{x}_i\|_2^2,$$

where $\|\mathbf{x}\|_2$ is the L_2-norm of a vector $\mathbf{x}$. The two optimization problems are equivalent and easily solved by obtaining the m eigenvectors associated with the m largest eigenvalues of $\mathbf{S}$. Although PCA is simple and powerful, it is prone to outliers [8,9] because $J_{L_2}(\mathbf{W})$ is based on the mean squared reconstruction error. To learn a subspace robust to outliers, Ke and Kanade [9] proposed to minimize an L_1-norm based objective function as follows:

$$J_{L_1}(\mathbf{W})=\frac{1}{N}\sum_{i=1}^N\|\mathbf{x}_i-\mathbf{W}\mathbf{W}^T\mathbf{x}_i\|_1,$$

where $\|\mathbf{x}\|_1$ is the L_1-norm of a vector $\mathbf{x}$. They also present an iterative method to obtain the solution for minimizing $J_{L_1}(\mathbf{W})$.

Although L_1-PCA minimizing $J_{L_1}(\mathbf{W})$ can relieve the negative effect of outliers, it is not invariant to rotations. In [10], Ding et al. proposed R_1-PCA, which is rotational invariant, at the same time is robust to outliers. It is to minimize the following objective function:

$$J_{R_1}(\mathbf{W})=\sum_{i=1}^N\rho\left(\sqrt{\mathbf{x}_i^T\mathbf{x}_i-\mathbf{x}_i^T\mathbf{W}\mathbf{W}^T\mathbf{x}_i}\right),$$

where $\rho(\cdot)$ is a generic loss function and the Cauchy function or Huber's M-estimator [22] was used for $\rho(\cdot)$ in [10]. Huber's M-estimator $\rho_H(s)$ is defined as

$$\rho_H(s)=\begin{cases}s^2 & \text{if } |s|\le c,\\ 2c|s|-c^2 & \text{otherwise}\end{cases}\tag{1}$$

where c is the cutoff parameter that controls the regularization effect of weights in a weighted covariance matrix. Note that $\rho_H(s)$ becomes a quadratic or a linear function of $|s|$ depending on the value of s. The solution for minimizing $J_{R_1}(\mathbf{W})$ was obtained by performing a subspace iteration algorithm [23].

On the other hand, PCA-L_1 was developed in [11] motivated by the duality between maximizing variance and minimizing reconstruction error. It maximizes an L_1 dispersion among the projected samples, $\sum_{i=1}^N\|\mathbf{W}^T\mathbf{x}_i\|_1$. A novel and efficient method for maximizing the L_1 dispersion was also presented in [11]. The method allows PCA-L_1 to be performed by much less computational effort than R_1-PCA.

HQ-PCA is formulated based on the maximum correntropy criterion in terms of information theoretic learning. Without the zero-mean assumption, which is necessary in other variants of PCA, HQ-PCA maximizes the correntropy estimated between a set of training samples $\{\mathbf{x}_i\}_{i=1}^N$ and the set of their reconstructed samples $\{\mathbf{W}\mathbf{y}_i+\mathbf{m}\}_{i=1}^N$, where $\mathbf{m}$ is a data mean. Mathematically, HQ-PCA tries to maximize the following objective function:

$$\arg\max_{\mathbf{W},\mathbf{m}}\sum_{i=1}^N g\left(\sqrt{\overline{\mathbf{x}}_i^T\overline{\mathbf{x}}_i-\overline{\mathbf{x}}_i^T\mathbf{W}\mathbf{W}^T\overline{\mathbf{x}}_i}\right),\tag{2}$$

where $g(x)=\exp(-x^2/2\sigma^2)$ is the Gaussian kernel and $\overline{\mathbf{x}}_i=\mathbf{x}_i-\mathbf{m}$. Note that HQ-PCA finds a data mean as well as a projection matrix. Using the Welsch M-estimator $\rho_W(x)=1-g(x)$, HQ-PCA is regarded as a robust M-estimator formulation because it is equivalent to finding $\mathbf{W}_H$ and $\mathbf{m}_H$ that minimize the following objective function:

$$J_{HQ}(\mathbf{W},\mathbf{m})=\sum_{i=1}^N\rho_W\left(\sqrt{\overline{\mathbf{x}}_i^T\overline{\mathbf{x}}_i-\overline{\mathbf{x}}_i^T\mathbf{W}\mathbf{W}^T\overline{\mathbf{x}}_i}\right).\tag{3}$$

In [13], the optimization problem in (2) was effectively solved in the half-quadratic optimization framework, which is often used to address nonlinear optimization problems in information theoretic learning.

3. Robust principal component analysis based on generalized mean

3.1. Generalized mean for positive numbers

For a $p\neq 0$, the generalized mean or power mean $\mathcal{M}_p$ of $\{a_i>0,\ i=1,\ldots,N\}$ [20] is defined as

$$\mathcal{M}_p\{a_1,\ldots,a_N\}=\left(\frac{1}{N}\sum_{i=1}^N a_i^p\right)^{1/p}.$$

The arithmetic mean, the geometric mean, and the harmonic mean are special cases of the generalized mean when $p=1, p\to 0$, and $p=-1$, respectively. Furthermore, the maximum and the minimum values of the numbers can also be obtained from the generalized mean by making $p\to\infty$ and $p\to-\infty$, respectively. Note that as p decreases (increases), the generalized mean is more affected by the smaller (larger) numbers than the larger (smaller) ones, i.e., controlling p makes it possible to adjust the contribution of each number to the generalized mean. This characteristic is useful in the situation where data samples should be differently handled according to their importance, for example, when outliers are contained in the training set.

In [24], it was shown that the generalized mean of a set of positive numbers can be expressed by a nonnegative linear combination of the elements in the set and, in this paper, it is further simplified as follows:

$$\sum_{i=1}^K a_i^p=b_1a_1+\cdots+b_Ka_K$$
$$b_i=a_i^{p-1},\quad i=1,\ldots,K.\tag{4}$$

Note that each weight b_i has the same value of 1 if $p=1$, where the generalized mean becomes the arithmetic mean. It is also noted that, if p is less than one, the weight b_i increases as a_i decreases. This means that, when $p<1$, the generalized mean is more influenced by the small numbers in $\{a_i\}_{i=1}^K$, and the extent of the influence increases as p decreases. This equation plays an

important role in solving the optimization problems using the generalized mean.

3.2. Generalized sample mean

Most conventional PCAs commonly assume that training samples have zero-mean. To satisfy this assumption, all of the samples are subtracted by the sample mean, i.e., $\mathbf{x}_i - \mathbf{m}_S$ for $i = 1, \ldots, N$, where $\mathbf{m}_S = \frac{1}{N}\sum_{i=1}^{N} \mathbf{x}_i$. The conventional sample mean can be considered as the center of the samples in the sense of the least square, i.e.,

$$\mathbf{m}_S = \arg\min_{\mathbf{m}} \frac{1}{N}\sum_{i=1}^{N} \|\mathbf{x}_i - \mathbf{m}\|_2^2. \tag{5}$$

In (5), a small number of outliers in the training samples dominate the objective function because the objective function in (5) is constructed based on the squared distances. To obtain a robust sample mean in the presence of outliers, a new optimization problem is formulated by replacing the arithmetic mean in (5) with the generalized mean as

$$\mathbf{m}_G = \arg\min_{\mathbf{m}} \left(\frac{1}{N}\sum_{i=1}^{N} \left(\|\mathbf{x}_i - \mathbf{m}\|_2^2 \right)^p \right)^{1/p}.$$

This problem is equivalent to (5) if $p = 1$. As mentioned in the previous subsection, the contribution of a large number to the objective function decreases as p decreases. Thus, the negative effect of outliers can be alleviated if $p < 1$. From now on, we will call $\mathbf{m}_G$ as the *generalized sample mean*. Using the fact that x^p with $p > 0$ being a monotonic increasing function of x for $x > 0$, this problem can be converted to

$$\mathbf{m}_G = \arg\min_{\mathbf{m}} \sum_{i=1}^{N} \left(\|\mathbf{x}_i - \mathbf{m}\|_2^2 \right)^p. \tag{6}$$

Although the minimization in (6) should be changed into the maximization when $p < 0$, we only consider positive values of p in this paper.

The necessary condition for $\mathbf{m}_G$ to be a local minimum is that the gradient of the objective function in (6) with respect to $\mathbf{m}$ is equal to zero, i.e.,

$$\frac{\partial}{\partial \mathbf{m}} \sum_{i=1}^{N} \left(\|\mathbf{x}_i - \mathbf{m}\|_2^2 \right)^p = 0.$$

However, it is hard to find a closed-form solution of the above equation. Although any gradient-based iterative algorithms can be applied to obtain $\mathbf{m}_G$, they usually have slow convergence speed. Alternatively, we develop a novel method based on (4), which is more efficient than gradient-based iterative methods. Our method for solving the problem in (6) is an iterative one, similar to the expectation–maximization algorithm [25].

In the derivation, we decompose (6) into the form of (4) and consider the weight b_i in (4) as a constant. Then, (6) can be approximated by a quadratic function of $\|\mathbf{x}_i - \mathbf{m}\|_2$ which can easily be optimized. The details are as follows. Let us denote the value of $\mathbf{m}$ after t iterations as $\mathbf{m}^{(t)}$. The first step of the update rule is, for $\mathbf{m}$ close to a fixed $\mathbf{m}^{(t)}$, to represent the objective function in (6) as a linear combination of $\|\mathbf{x}_i - \mathbf{m}^{(t)}\|_2^2$ using (4), i.e.,

$$\sum_{i=1}^{N} \left(\|\mathbf{x}_i - \mathbf{m}\|_2^2 \right)^p \approx \sum_{i=1}^{N} \alpha_i^{(t)} \|\mathbf{x}_i - \mathbf{m}\|_2^2,$$

where

$$\alpha_i^{(t)} = \left(\|\mathbf{x}_i - \mathbf{m}^{(t)}\|_2^2 \right)^{p-1}. \tag{7}$$

Here, the approximation becomes exact when $\mathbf{m} = \mathbf{m}^{(t)}$. Note that the objective function near $\mathbf{m}^{(t)}$ can be approximated as a quadratic function of $\mathbf{m}$ without computing the Hessian matrix of the objective function. The next step is to find $\mathbf{m}^{(t+1)}$ that minimizes the approximated function based on the computed $\alpha_i^{(t)}$, i.e.,

$$\frac{\partial}{\partial \mathbf{m}} \sum_{i=1}^{N} \alpha_i^{(t)} \|\mathbf{x}_i - \mathbf{m}\|_2^2 = 0.$$

The solution of this equation is just the weighted average of the samples as follows:

$$\mathbf{m}^{(t+1)} = \frac{1}{\sum_{j=1}^{N} \alpha_j^{(t)}} \sum_{i=1}^{N} \alpha_i^{(t)} \mathbf{x}_i. \tag{8}$$

This update rule with the two steps is repeated until a convergence condition is satisfied. This procedure is summarized in Algorithm 1.

Algorithm 1. Generalized sample mean.

1: **Input:** $\{\mathbf{x}_1, \ldots, \mathbf{x}_N\}$, $p > 0$.
2: $t \longleftarrow 0$.
3: $\mathbf{m}^{(t)} \longleftarrow \mathbf{m}_S$.
4: **repeat**
5: **Approximation**: For fixed $\mathbf{m}^{(t)}$, compute $\alpha_1^{(t)}, \ldots, \alpha_N^{(t)}$ according to (7).
6: **Minimization**: Using the computed $\alpha_1^{(t)}, \ldots, \alpha_N^{(t)}$, update $\mathbf{m}^{(t+1)}$ according to (8).
7: $t \longleftarrow t+1$.
8: **until** A stop criterion is satisfied
9: **Output:** $\mathbf{m}_G = \mathbf{m}^{(t)}$.

Note that a weighted average is computed at each iteration in Algorithm 1. Thus, it can be said that Algorithm 1 is a special case of the mean shift algorithm [26]. It is also noted that the number of initial points is only one, which is set to $\mathbf{m}_S$. Since non-convex optimization methods depend on initial points, they are generally conducted several times started from different initial points and the solution is selected as the one providing the best performance. However, we have empirically found that Algorithm 1 started from $\mathbf{m}_S$ converges to a local optimum point that is enough robust to outliers.

To demonstrate the robustness of the generalized sample mean obtained by Algorithm 1, we randomly generated 100 samples from a two-dimensional Gaussian distribution with the mean $\mathbf{m}_i = 0$ and covariance matrix $\Sigma_i = diag[0.5, 0.5]$ for inliers and also generated 10 samples from another two-dimensional Gaussian distribution with the mean $\mathbf{m}_o = [5, 5]^T$ and covariance matrix $\mathbf{\Sigma}_o = diag[0.3, 0.3]$ for outliers. Using the generated samples, the sample mean was computed and two generalized sample means were also obtained by Algorithm 1 with $p = 0.1$ and $p = 0.2$, respectively. Fig. 1 shows the arithmetic sample mean and the two generalized sample means together with the generated samples. It is obvious that the generalized sample means are located close to the mean of the inliers, $[0, 0]^T$, whereas the arithmetic sample mean is much more biased by the ten outliers. This illustrates that the generalized sample mean with an appropriate value of p is more robust to outliers than the arithmetic sample mean.

3.3. Principal component analysis using generalized mean

For a projected sample $\mathbf{W}^T\mathbf{x}$, the squared reconstruction error $e(\mathbf{W})$ can be computed as

$$e(\mathbf{W}) = \tilde{\mathbf{x}}^T\tilde{\mathbf{x}} - \tilde{\mathbf{x}}^T\mathbf{W}\mathbf{W}^T\tilde{\mathbf{x}},$$

where $\tilde{\mathbf{x}} = \mathbf{x} - \mathbf{m}$. We use the generalized sample mean $\mathbf{m}_G$ for $\mathbf{m}$. To prevent outliers corresponding to large $e(\mathbf{W})$ from dominating the objective function, we propose to minimize the following objective function:

$$J_G(\mathbf{W}) = \left(\frac{1}{N}\sum_{i=1}^{N} [e_i(\mathbf{W})]^p \right)^{1/p}, \tag{9}$$

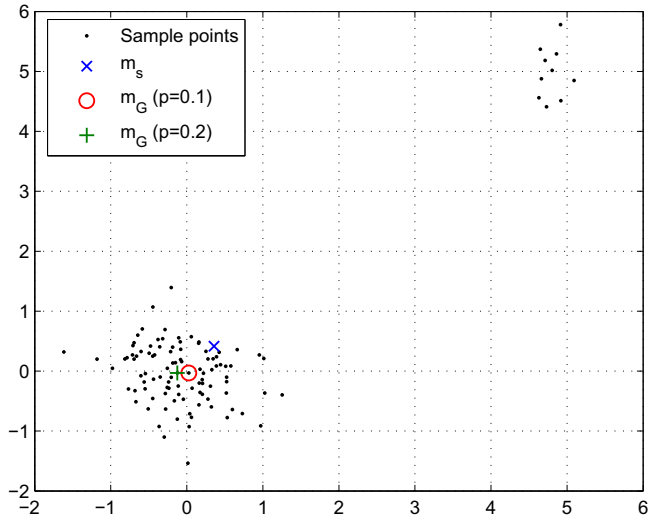

Fig. 1. 2D toy example with 100 inliers and 10 outliers. The arithmetic mean ($\mathbf{m}_S$) and the generalized sample mean ($\mathbf{m}_G$) are marked.

where $e_i(\mathbf{W}) = \tilde{\mathbf{x}}_i^T\tilde{\mathbf{x}}_i - \tilde{\mathbf{x}}_i^T\mathbf{W}\mathbf{W}^T\tilde{\mathbf{x}}_i$ is the squared reconstruction error of $\mathbf{x}_i$ with respect to $\mathbf{W}$. Note that $J_G(\mathbf{W})$ is formulated by replacing the arithmetic mean in $J_{L_2}(\mathbf{W})$ with the generalized mean keeping the use of the Euclidean distance and it is equivalent to $J_{L_2}(\mathbf{W})$ if $p=1$. The negative effect raised by outliers is suppressed in the same way as in (6). Also, the solution that minimizes $J_G(\mathbf{W})$ is rotationally invariant because each $e_i(\mathbf{W})$ is measured based on the Euclidean distance. To obtain $\mathbf{W}_G$, we develop an iterative optimization method similar to Algorithm 1.

Like the optimization problem for $\mathbf{m}_G$ in the previous subsection, under the assumption that $p>0$, the optimization problem based on (9) is firstly converted as follows:

$$\mathbf{W}_G = \arg\min_{\mathbf{W}^T\mathbf{W}=\mathbf{I}} \left(\frac{1}{N}\sum_{i=1}^{N}[e_i(\mathbf{W})]^p\right)^{1/p} = \arg\min_{\mathbf{W}^T\mathbf{W}=\mathbf{I}} \sum_{i=1}^{N}[e_i(\mathbf{W})]^p, \tag{10}$$

Next, let us denote $\mathbf{W}^{(t)}$ as the value of $\mathbf{W}\in\mathbb{R}^{n\times m}$ after the t-th iteration. Near a fixed $\mathbf{W}^{(t)}$, the converted objective function in (10) can be approximated as a quadratic function of $\mathbf{W}$ according to (4) as

$$\sum_{k=1}^{N}[e_i(\mathbf{W})]^p \approx \sum_{i=1}^{N}\beta_i^{(t)}e_i(\mathbf{W}),$$

where

$$\beta_i^{(t)} = \left[e_i(\mathbf{W}^{(t)})\right]^{p-1}. \tag{11}$$

Here, the approximation becomes exact if $\mathbf{W}=\mathbf{W}^{(t)}$. After calculating each $\beta_i^{(t)}$, $\mathbf{W}^{(t+1)}$ can be computed by minimizing the approximated function as

$$\mathbf{W}^{(t+1)} = \arg\min_{\mathbf{W}} \sum_{i=1}^{N}\beta_i^{(t)}e_i(\mathbf{W}) = \arg\max_{\mathbf{W}} \mathrm{tr}\left(\mathbf{W}^T\mathbf{S}_\beta^{(t)}\mathbf{W}\right), \tag{12}$$

where $\mathbf{S}_\beta^{(t)} = \sum_{i=1}^{N}\beta_i^{(t)}\tilde{\mathbf{x}}_i\tilde{\mathbf{x}}_i^T$.

Algorithm 2. PCA-GM.

1: **Input:** $\{\mathbf{x}_1,\ldots,\mathbf{x}_N\}$, $\mathbf{m}_G$, m, p.
2: $t\longleftarrow 0$.
3: $\mathbf{W}^{(t)}\longleftarrow\mathbf{W}_{PCA}\in\mathbb{R}^{n\times m}$.

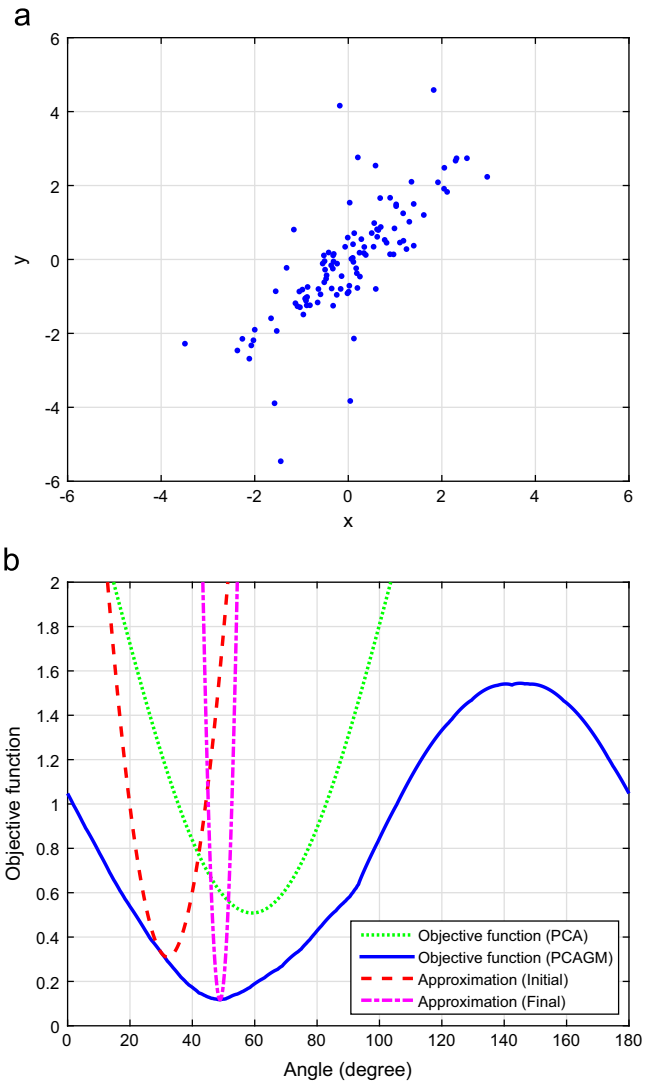

Fig. 2. (a) A toy example to illustrate Algorithm 2. (b) The values of the objective functions $J_{L_2}(\mathbf{W})$, $J_G(\mathbf{W})$ are plotted. The quadratic approximations of $J_G(\mathbf{W})$ at the initial point $\mathbf{W}^{(0)}=[\cos 30^\circ \ \sin 30^\circ]^T$ and the final point $\mathbf{W}^{(t)}=[\cos 48.9^\circ \ \sin 48.9^\circ]^T$ are also plotted. (For interpretation of the references to color in this figure caption, the reader is referred to the web version of this paper.)

4: **repeat**
5: **Approximation**: For fixed $\mathbf{W}^{(t)}$, compute $\beta_1^{(t)},\ldots,\beta_N^{(t)}$ using (11).
6: **Minimization**: Using the computed $\beta_1^{(t)},\ldots,\beta_N^{(t)}$, find $\mathbf{W}^{(t+1)}$ by solving the eigenvalue problem in (12).
7: $t\longleftarrow t+1$.
8: **until** A stop criterion is satisfied
9: **Output:** $\mathbf{W}_G=\mathbf{W}^{(t)}$.

Note that $\mathbf{S}_\beta^{(t)}$ is a weighted covariance matrix and the columns of $\mathbf{W}^{(t+1)}$ are the m orthonormal eigenvectors associated with the largest m eigenvalues of $\mathbf{S}_\beta^{(t)}$. These two steps are repeated until a convergence criterion is satisfied. Algorithm 2 summarizes this iterative procedure of computing $\mathbf{W}_G$. Unfortunately, the update rule in Algorithm 2 does not guarantee that $J_G\left(\mathbf{W}^{(t+1)}\right) < J_G\left(\mathbf{W}^{(t)}\right)$. Nonetheless, the experimental results show that $\mathbf{W}_G$ obtained by the algorithm is good enough.

Fig. 3. Examples of original face images (upper row) and the corresponding images (lower row) occluded by rectangular noise.

To help understanding of Algorithm 2, we made another toy example as shown in Fig. 2(a) where 110 two dimensional samples are plotted. Among the samples, 100 samples are regarded as inliers and the others are regarded as outliers. The samples were generated as the following rule:

$$x_i \sim N(0,1),$$
$$y_i = x_i + \epsilon_i,$$

where the random noise ϵ_i is sampled from $N(0, 0.5^2)$ for inliers and $N(0, 3^2)$ for outliers, respectively. Fig. 2(b) shows the objective function of PCA-GM in (9) with $p = 0.3$ for the samples as shown in Fig. 2(a). We can see from Fig. 2(b) that the conventional PCA is prone to the ten outliers because its objective function is minimized around $\mathbf{W} = [\cos 60^\circ \ \sin 60^\circ]^T$. However, PCA-GM is robust to the outliers because its objective function is minimized at $\mathbf{W} = [\cos 48.9^\circ \ \sin 48.9^\circ]^T$, which is close to the solution without the outliers $\mathbf{W}^* = [\cos 45^\circ \ \sin 45^\circ]^T$. Given an initial projection vector $\mathbf{W}^{(0)} = [\cos 30^\circ \ \sin 30^\circ]^T$, the approximation step in Algorithm 2 gives a quadratic function corresponding to the red dashed line in Fig. 2(b). In the second step, the next iteration $\mathbf{W}^{(1)}$ is determined as $[\cos 32.1^\circ \ \sin 32.1^\circ]^T$ by minimizing the approximate function. Interestingly, it can be said that the approximate function plays a similar role of an upper bound of the objective function around $\mathbf{W}^{(0)}$ in this update rule. It is also noted that the approximated function at the local optimal point $\mathbf{W} = [\cos 48.9^\circ \ \sin 48.9^\circ]^T$ has its minimum as the same location, which is denoted as the magenta dashed dotted line in Fig. 2(b). This means that Algorithm 2 converges to the local minimum point of the objective function for the problem shown in Fig. 2(a).

In practice, when $e_i(\mathbf{W}^{(t)})$ is zero or very small for any i, $\left[e_i(\mathbf{W}^{(t)})\right]^{p-1}$ is numerically unstable if $p < 1$, and Algorithm 2 cannot proceed anymore. This problem can also occur in Algorithm 1. It can be overcome by adding a small constant δ into each $e_i(\mathbf{W})$ as

$$e_i(\mathbf{W})' = \tilde{\mathbf{x}}_i^T \tilde{\mathbf{x}}_i - \tilde{\mathbf{x}}_i^T \mathbf{W}\mathbf{W}^T \tilde{\mathbf{x}}_i + \delta, \tag{13}$$

where δ should be small enough that the modified objective function is not affected too much. This perturbation also changes $\mathbf{S}_\beta^{(t)}$ in (12) into $\widehat{\mathbf{S}}_\beta^{(t)}$ as

$$\widehat{\mathbf{S}}_\beta^{(t)} = \sum_{i=1}^{N} \beta_i^{(t)} \left(\tilde{\mathbf{x}}_i \tilde{\mathbf{x}}_i^T + \frac{\delta}{n} \right),$$

where n is the original dimensionality of data.

4. Experiments

To evaluate the proposed method, we considered face reconstruction, digit clustering, and object categorization problems, the first two of which were addressed in [11,13], respectively. The proposed method was compared with PCA, PCA-L_1, R_1-PCA, and HQ-PCA. Except for the conventional PCA, they have the parameters to be predetermined and we determined the values of the parameters according to the recommendations in [10,11,13]. Also, in PCA-GM, the generalized sample mean was used instead of the sample mean, and the perturbation parameter δ in (13) was set to 0.01 times the minimum of $e_i(\mathbf{W}_{PCA})$ for $i = 1, \ldots, N$. For the iterative algorithms as R_1-PCA, HQ-PCA, PCA-GM, the number of iterations was limited to 100.

4.1. Face reconstruction

We collected 800 facial images from the subset 'fa' of the Color FERET database [27] for the face reconstruction problem. Each face image was normalized to a size of 40×50 pixels using the eye coordinates, which were obtained in the database. We simulated two types of outliers. For the first type of outliers, some of the facial images were randomly selected, and each of the selected images was occluded by a rectangular area, each pixel in which was randomly set to 0 (black) or 255 (white). The size and location of the rectangular area were randomly determined. Fig. 3 shows examples of original normalized faces in the upper row and their corresponding faces occluded by the rectangular noise in the lower row. To evaluate the proposed method with different noise levels, we selected 80, 160, and 240 images from the 800 facial images and occluded them by rectangular noise, so that we made three training sets including 80, 160, and 240 occluded images. For the second type of outliers, other three training sets were constructed by adding 80, 160, and 240 dummy images (outlier) with the same size to the original 800 face images (inlier), so that the numbers of inliers and outliers in the three training sets are (800,80), (800,160), and (800,240) respectively. Each pixel in the dummy images was also randomly set to 0 or 255.

After applying different versions of PCAs to the training sets with the various numbers of extracted features m from 5 to 100, we compared the average reconstruction errors as in [11] defined as

$$\frac{1}{N} \sum_{i=1}^{N} \left\| \left(\mathbf{x}_i^{ori} - \mathbf{m} \right) - \mathbf{W}\mathbf{W}^T (\mathbf{x}_i - \mathbf{m}) \right\|_2, \tag{14}$$

where $\mathbf{x}_i^{ori}$ and $\mathbf{x}_i$ are the i-th original unoccluded image and the corresponding training image, respectively, N is the number of the face images, and $\mathbf{m}$ is the mean of the original normalized faces. For the training sets related to the second type of outliers, the dummy images were excluded when measuring the average reconstruction errors, and $\mathbf{x}_i^{ori}$ and $\mathbf{x}_i$ were identical. Note that $\mathbf{W}$ is the projection matrix obtained from PCA, PCA-L_1, R_1-PCA, HQ-PCA, and PCA-GM for the various values of m. Moreover, PCA-GM was performed using 0.1, 0.2, 0.3, and 0.4 for the value of p to figure out the effect of it.

Figs. 4 and 5 show the average reconstruction errors measured as in (14) for the training sets constructed to simulate two types of

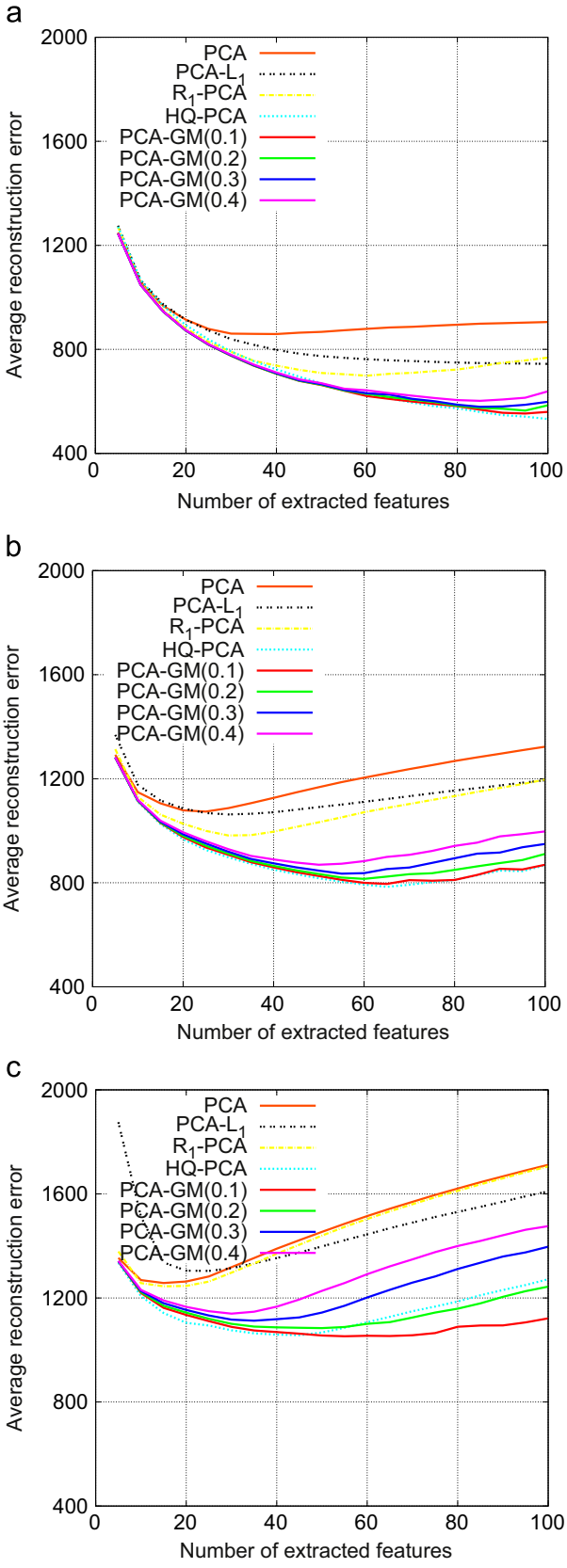

Fig. 4. Average reconstruction errors of different PCA methods for the data sets where the numbers of inliers and outliers (occlusion) are (a) (720, 80), (b) (640,160), and (c) (560, 240). (For interpretation of the references to color in this figure caption, the reader is referred to the web version of this paper.)

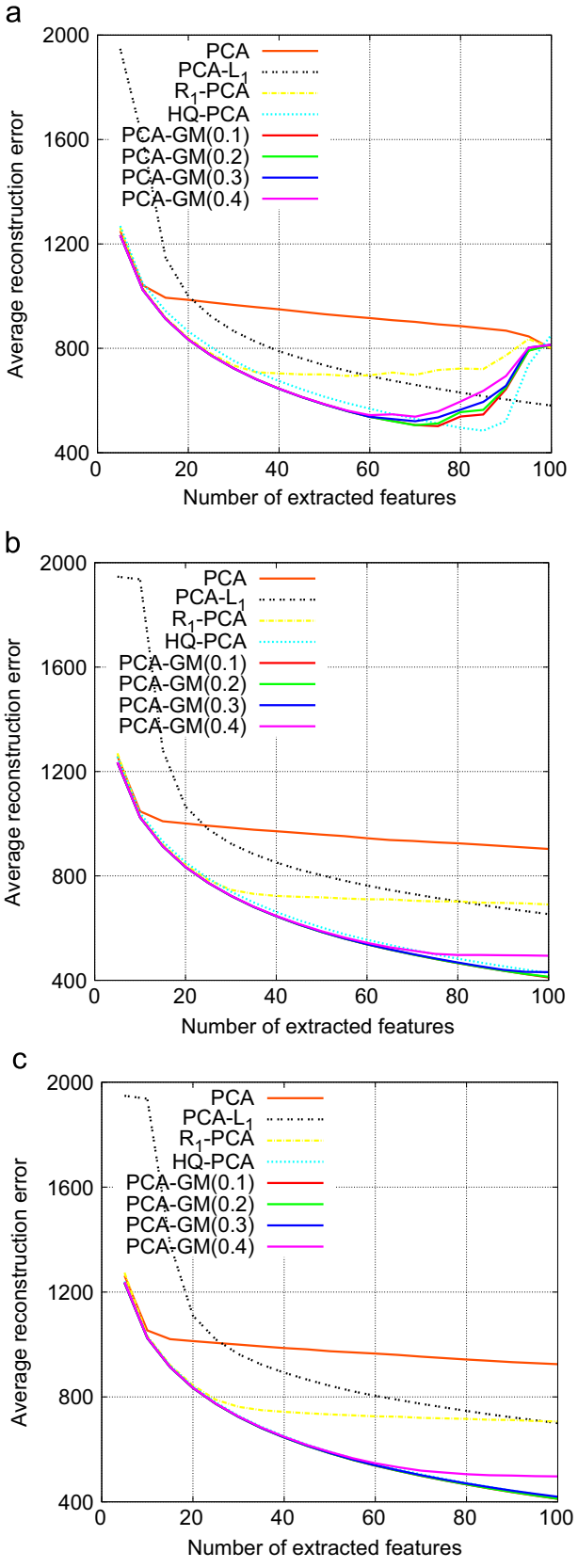

Fig. 5. Average reconstruction errors of different PCA methods for the data sets where the numbers of inliers and outliers (dummy images) are (a) (800, 80), (b) (800, 160), and (c) (800, 240). (For interpretation of the references to color in this figure caption, the reader is referred to the web version of this paper.)

outliers when $5 \leq m \leq 100$. As shown in the figures, PCA-GM and HQ-PCA generally gave better performances than PCA, PCA-L_1, and R_1-PCA regardless of the types of outliers and the level of noise, and they yielded competitive results to each other. When the number of the occluded images is 240, which corresponds to Fig. 4(c), HQ-PCA provided lower average reconstruction errors than PCA-GM for $m \leq 40$ while PCA-GM with $p = 0.1$ and $p = 0.2$ gave better performances than HQ-PCA for $m \geq 60$. When the

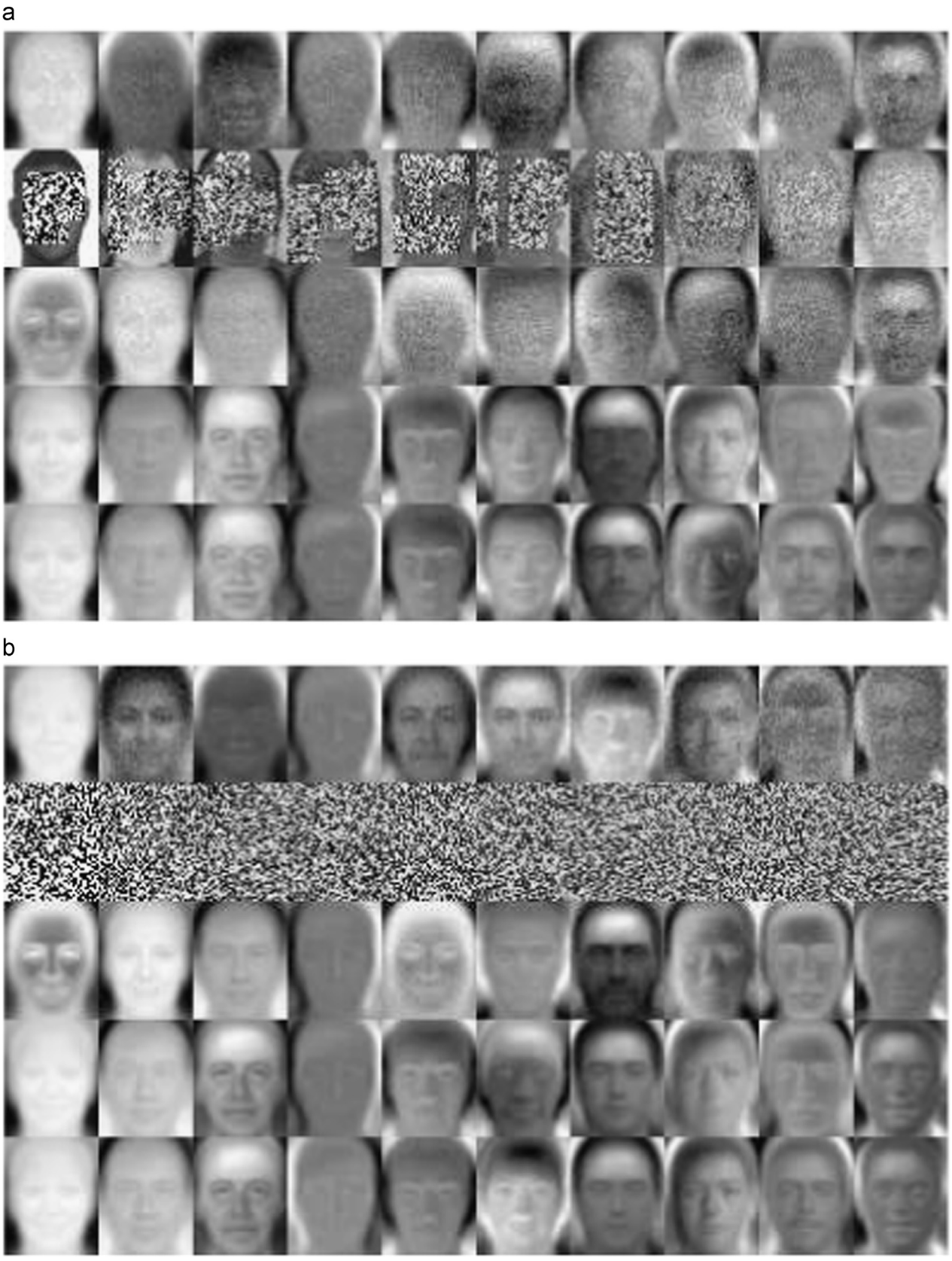

Fig. 6. Eigenfaces obtained by PCA, PCA-L_1, R_1-PCA, HQ-PCA, and PCA-GM with $p = 0.1$ in order of row. (a) Occlusion noise. (b) Dummy image noise.

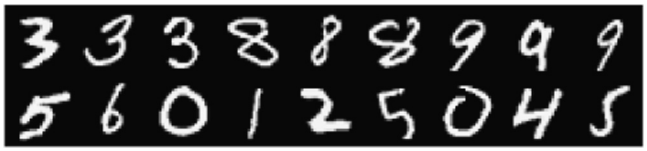

Fig. 7. Examples of MNIST handwritten digit images used as inliers (first row; 3, 8, 9) and outliers (second row; other digits).

number of the dummy images is 80, which corresponds to Fig. 5 (a), the lower reconstruction errors could be obtained by PCA-GM rather than HQ-PCA when $m \leq 60$ while HQ-PCA preformed better than PCA-GM for $80 \leq m < 100$.

The effectiveness of the proposed method can also be found by visualizing projection matrices in terms of the *Eigenfaces* [3]. Fig. 6 shows the first ten of Eigenfaces obtained by different PCA methods when $m = 40$ and the number of outliers is 240 for both types of outliers. We can see that the Eigenfaces of HQ-PCA and PCA-GM are less contaminated from the outliers than PCA, PCA-L_1, and R_1-PCA. Also, it can be seen from the figure that PCA-L_1 yielded projection matrix with a different property. This may be a reason of the fact that PCA-L_1 provided relatively large reconstruction errors when m is small as shown in Figs. 4 and 5(c).

The effect of p in PCA-GM was as expected. For the occlusion noise as shown in Fig. 4, the lower values of p gave better performances and the performance differences are more distinct as m and noise level increase. For the dummy images as shown in Fig. 5, PCA-GM showed almost similar performances for all the values of p except for 0.4 when $m \geq 70$. These results agree with the fact that the generalized mean of a set of positive numbers depends on small numbers more and more as p gets smaller.

4.2. Clustering

The clustering problem was dealt with using a subset of the MNIST handwritten digit database,[2] which includes a training set of 60,000 examples and a test set of 10,000 examples. We randomly gathered 100 examples per the digits '3', '8', and '9' from the first 10,000 examples in the training set. To simulate outliers, we also randomly gathered 60 examples corresponding to the other digits from the same 10,000 examples. Thus, our training set for the clustering problem consists of 300 inliers and 60 outliers, which were normalized to unit norm. Fig. 7 shows nine images of the inliers in the upper row and nine images of the outliers in the lower row.

After obtaining projection matrices by applying various versions of PCAs to the training set, K-means clustering with $K = 3$ was performed using the projected inlier examples. For the initial means of the K-means clustering, we selected the two examples with the largest distance and then selected another example which had the largest sum of the distances from the previously selected two examples. The clustering accuracy was computed based on the class labels assigned to the examples in the database. Table 1 shows the clustering accuracy for various numbers of extracted features. As the previous experiments, we conducted PCA-GM using the settings of $p \in \{0.1, 0.2, 0.3, 0.4\}$. The best performance was achieved when $p = 0.3$ which is reported in Table 1. Considering the clustering accuracy without the dimensionality reduction was 70%, PCA-GM improved the clustering accuracy by 5%. Different from the results of the face reconstruction problem in the previous subsection, R_1-PCA and PCA-L_1 gave similar highest clustering accuracy as PCA-GM, while HQ-PCA performed poorly than PCA-GM. However, R_1-PCA and PCA-L_1 provided the highest accuracy when $m = 300$ whereas PCA-GM yielded the best performance when $m = 200$.

[2] http://yann.lecun.com/exdb/mnist/

Table 1
Clustering accuracy (%) of the digit images corresponding to '3', '8', and '9' in the reduced spaces which are obtained from the training set containing the other digit images as outliers.

m	PCA	PCA-L_1	R_1-PCA	HQ-PCA	PCA-GM
50	70.00	69.00	69.67	70.00	70.00
100	70.00	72.00	70.00	69.33	69.67
150	70.67	74.00	70.67	70.00	70.00
200	70.33	73.67	70.33	73.67	**75.00**
250	70.33	73.67	73.67	74.00	74.00
300	70.33	**75.00**	**75.00**	73.67	73.67

4.3. Object categorization

We evaluated the proposed method by performing object categorization on Small NORB data set [28]. The NORB data set consists of images of 50 different objects belonging to 5 categories each of which contains 10 objects. For each category, the images of 5 objects shown in Fig. 8(a) belong to its training set and those of 5 objects shown in Fig. 8(b) belong to its test set. The Small NORB data set is a subset of the NORB data set comprising 24,300 images for training and 24,300 images for testing, which are normalized with the size of 96×96 pixels on uniform background. Each object in the data set was captured under 18 azimuths, 9 elevations, and 6 light conditions. To evaluate the proposed method for different numbers of training samples, we uniformly sampled the three image capture variables to construct three training sets with 3375, 12,150, and 24,300 samples. We also resized the images in the data set to 48×48 and 64×64 pixels for computational efficiency. Consequently, we have six training sets with different number of samples (N) and dimensionality of input samples (n).

Although there are various approaches to categorize an arbitrary sample $\mathbf{z}$ corresponding to an image of an object, we performed the categorization by the *nearest-to-subspace*, i.e., $\mathbf{z}$ is determined to belong to the category minimizing the distance from $\mathbf{z}$ to the subspace spanned by the training samples in the category. For the distance from $\mathbf{z}$ to the subspace of the i-th category, we employed the squared residual error of $\mathbf{z}$ to the subspace computed as $\tilde{\mathbf{z}}_i^T\tilde{\mathbf{z}}_i - \tilde{\mathbf{z}}_i^T\mathbf{W}_i\mathbf{W}_i^T\tilde{\mathbf{z}}_i$, where $\tilde{\mathbf{z}}_i = \mathbf{z} - \mathbf{m}_i$ and $\mathbf{W}_i$ is the orthonormal basis of the subspace, which corresponds to the projection matrix and can be obtained by one of the PCA methods aforementioned. Also, $\mathbf{m}_i$ is the mean of the training samples in the i-th category. We used the sample mean $\mathbf{m}_S$ for $\mathbf{m}_i$ in PCA, PCA-L_1, and R_1-PCA while we used $\mathbf{m}_H$ and $\mathbf{m}_G$ instead of $\mathbf{m}_S$ in HQ-PCA and PCA-GM, respectively. For the purpose of comparison, the categorization accuracy was evaluated varying the dimensionality of subspaces (m) from 5 to 50.

Fig. 9 shows the categorization accuracy measured on the 24,300 test images in Small NORB data set. It is necessary to note that artificial outliers were not used in this experiments different from the previous ones. We can see that PCA-GM with an appropriate value of p is competitive with the conventional PCA when $N = 3375$ and the proposed method provides higher categorization accuracies than PCA as N increases. Especially when $N = 24,300$, the proposed method achieves the best performance for all the cases of m. This trend appears in both cases of $n = 48 \times 48$ and $n = 64 \times 64$. However, the other variants of PCA did not gave higher accuracies than the conventional PCA for most cases. In particular, HQ-PCA, which showed competitive performance in the face reconstruction experiments, resulted in the lowest categorization accuracy. This means that the proposed method can be an effective alternative to PCA in object categorization using the nearest-to-subspace when training data is enough.

Together with the categorization accuracy, we measured number of iterations in PCA-GM and running time of the proposed method to obtain projection matrices from the above six training

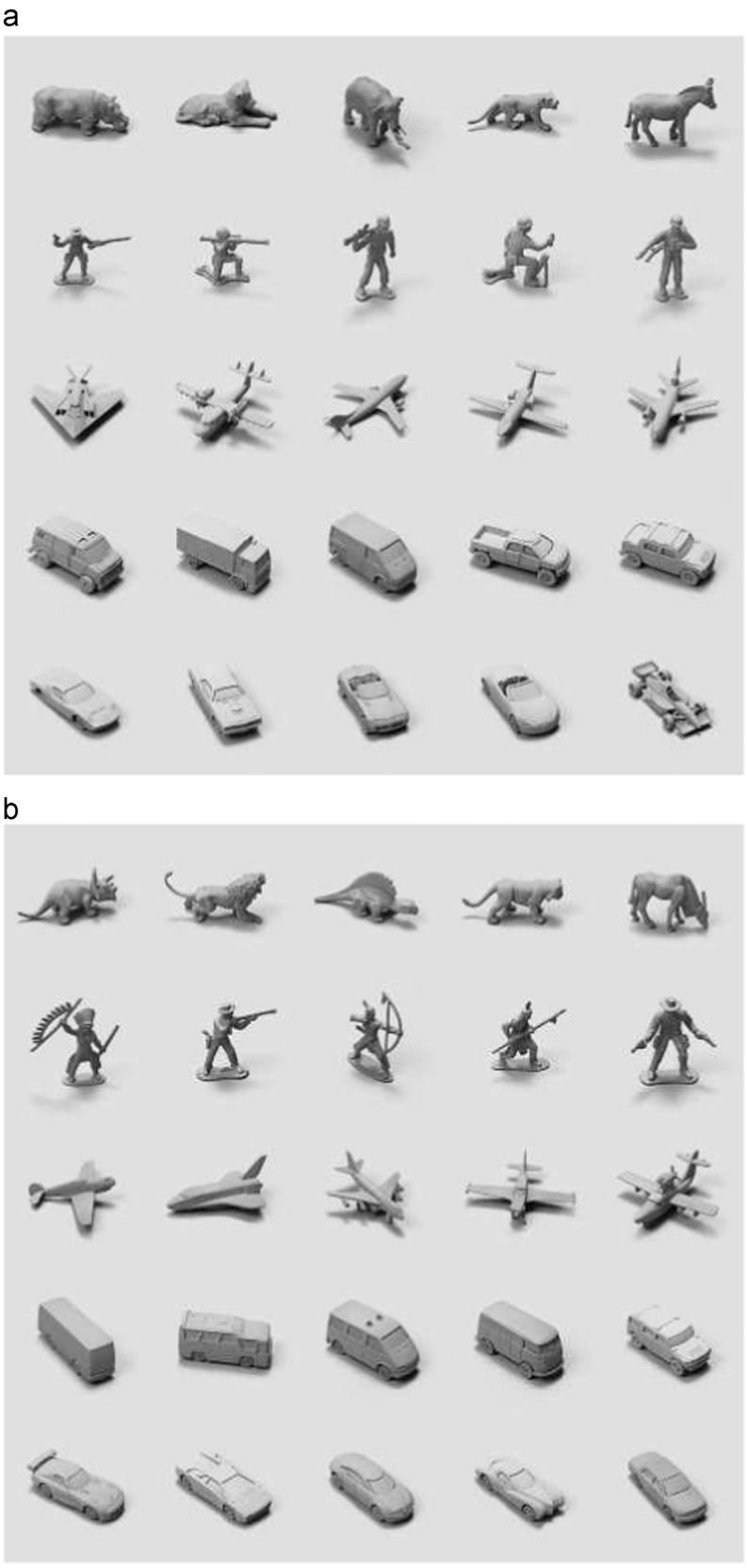

Fig. 8. Images of objects in (a) training and (b) test sets of NORB data set.

sets of the Small NORM data set. Table 2 shows the average numbers of iterations performed in the proposed method. From this table, we can find that PCA-GM converges in less than 50 iterations on average. Also, the average number of iterations decreases as the value of p increases from 0.1 to 0.9. This may have been resulted from the fact that the objective function of PCA-GM has many fluctuations when the value of p is close to zero whereas it is similar to one of the conventional PCA, which is quadratic,

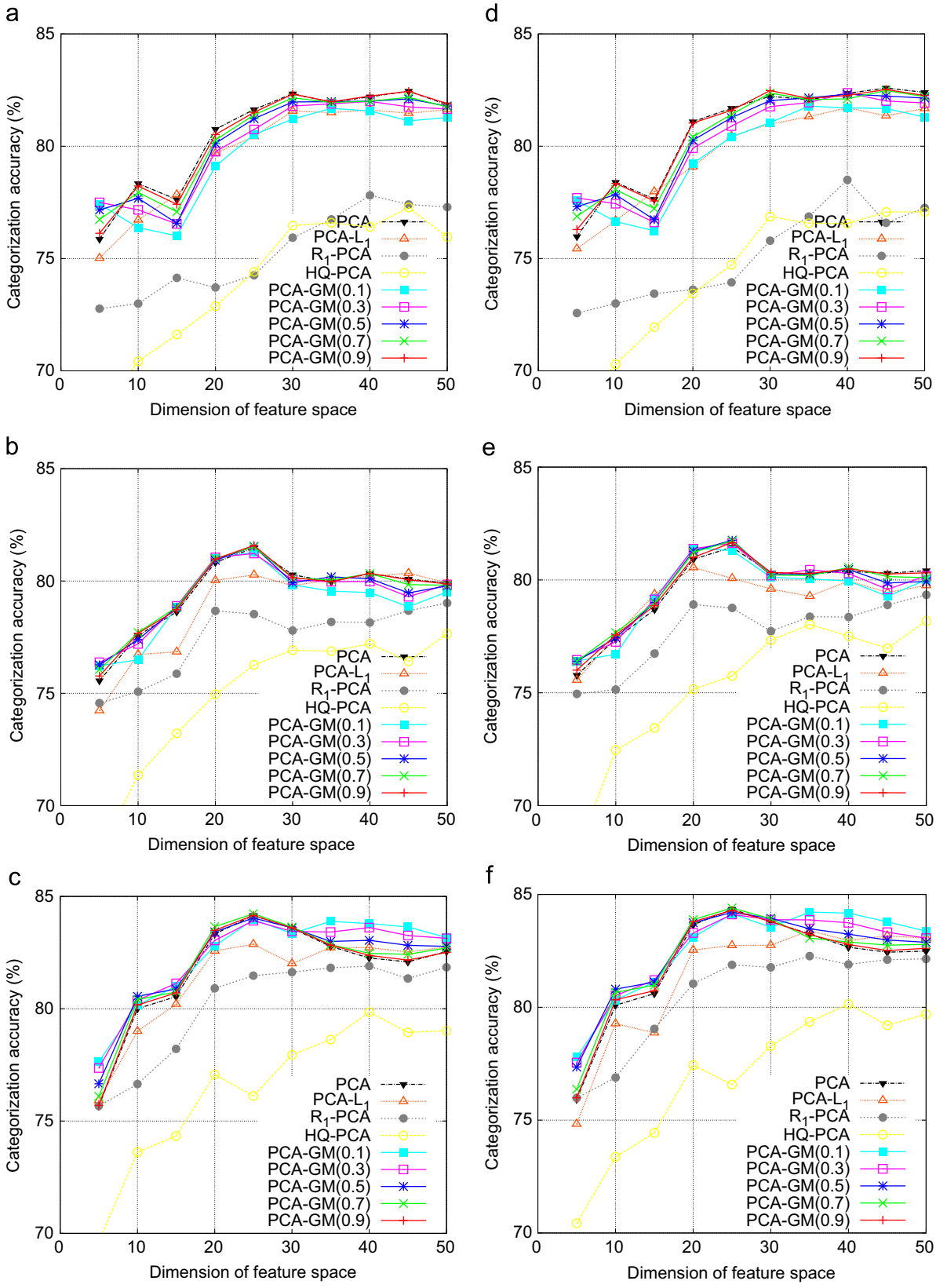

Fig. 9. Categorization accuracy of different PCA methods for the training sets with different sizes of training images (n) and different numbers of training samples (N); (a) (48×48, 3375), (b) (48×48, 12,150), (c) (48×48, 24,300), (d) (64×64, 3375), (e) (64×64, 12,150), (c) (64×64, 24,300). (For interpretation of the references to color in this figure caption, the reader is referred to the web version of this paper.)

when the value of p is close to one. The overall running time of the proposed method described in Algorithm 2 varies depending on the number of iterations needed until a stop criterion is satisfied. Thus, we divided the overall running times by the average numbers of iterations performed in computing five projection matrices with respect to five categories for every combination of m, n, and

Table 2
Average numbers of iterations needed in PCA-GM on small NORB data set.

$p=0.1$	$p=0.3$	$p=0.5$	$p=0.7$	$p=0.9$
22.89	23.43	19.27	14.80	8.42

Table 3
Average running time in seconds per each iteration in PCA-GM with $p=0.1$ on small NORB data set.

m	$n=48\times 48$			$n=64\times 64$		
	$N=3375$	$N=12,150$	$N=24,300$	$N=3375$	$N=12,150$	$N=24,300$
5	1.05	22.55	40.28	1.62	47.37	146.31
10	0.95	25.45	42.90	1.59	51.36	150.91
15	0.98	25.69	37.78	1.59	52.55	134.89
20	0.94	23.74	39.14	1.49	46.61	149.55
25	0.89	17.99	35.50	1.44	36.83	130.75
30	0.88	19.60	38.86	1.42	42.74	98.59
35	0.89	18.34	39.40	1.46	43.60	136.14
40	0.84	20.23	32.62	1.42	42.51	106.12
45	0.85	20.21	33.70	1.38	39.35	126.85
50	0.84	18.35	32.03	1.43	41.43	111.99

N, which are summarized in Table 3 in the setting of $p=0.1$. From the other values of p, we could see the similar tendencies. The running times were measured on a 3.4 GHz Intel Xeon workstation with 12 cores using MATLAB. Each iteration in the algorithm consists of two processes, the approximation and the minimization. Compared to the approximation, the minimization requires much more computations. It corresponds to the weighted eigenvalue decomposition, which was implemented by applying the singular value decomposition (SVD) to the weighted data matrix instead of computing the weighted covariance matrix and applying the eigenvalue decomposition to it for efficiency. Thus, the running times reported in Table 3 can be regarded as the running time of the SVD approximately. Considering the average numbers of iterations shown in Table 2, it can be said that the proposed method is feasible enough until $N=25,000$ and $n=5000$ roughly.

5. Conclusion and discussion

We proposed a robust PCA using the generalized mean to mitigate the negative effect of outliers belonging to the training set. Considering the fact that the sample mean is prone to the outliers, a generalized sample mean was proposed based on the generalized mean as an alternative to the sample mean in the framework of the proposed method. The efficient iterative methods were also developed to solve the optimization problems formulated using the generalized mean. Experiments on the face reconstruction, clustering, and object categorization problems demonstrated that the proposed method performs better than or equal to the other robust PCAs depending on the problems tackled. We expect that the proposed methods can be used in various applications. For example, a trimmed average, which is one of the robust first-order statistics, was used in a scalable robust PCA method [29]. We think that the generalized sample mean can be an effective alternative to the trimmed average.

Conflict of interest

None declared.

Acknowledgments

This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (NRF-2013R1A1A2007461, NRF-2013R1A1A1006599).

References

[1] A. Jain, R. Duin, M. Jianchang, Statistical pattern recognition: a review, IEEE Trans. Pattern Anal. Mach. Intell. 22 (1) (2000) 4–37.
[2] I. Jolliffe, Principal Component Analysis, 2nd edition, Springer-Verlag, New York, 2002.
[3] M. Turk, A. Pentland, Eigenfaces for recognition, J. Cognit. Neurosci. 3 (1) (1991) 71–86.
[4] P. Belhumeur, J. Hespanha, D. Kriegman, Eigenfaces vs. Fisherfaces: recognition using class specific linear projection, IEEE Trans. Pattern Anal. Mach. Intell. 19 (7) (1997) 711–720.
[5] D.A. Ross, J. Lim, R.-S. Lin, M.-H. Yang, Incremental learning for robust visual tracking, Int. J. Comput. Vis. 77 (1–3) (2008) 125–141.
[6] K.Y. Yeung, W.L. Ruzzo, Principal component analysis for clustering gene expression data, Bioinformatics 17 (9) (2001) 763–774.
[7] C. Ding, X. He, K-means clustering via principal component analysis, in: Proceedings of the 21st International Conference on Machine Learning, ICML '04, 2004.
[8] F. de la Torre, M.J. Black, A framework for robust subspace learning, Int. J. Comput. Vis. 54 (1–3) (2003) 117–142.
[9] Q. Ke, T. Kanade, Robust L_1 norm factorization in the presence of outliers and missing data by alternative convex programming, in: IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2005, vol. 1, 2005, pp. 739–746.
[10] C. Ding, D. Zhou, X. He, H. Zha, R_1-PCA: rotational invariant L_1-norm principal component analysis for robust subspace factorization, in: Proceedings of the 23rd International Conference on Machine Learning, ICML '06, 2006, pp. 281–288.
[11] N. Kwak, Principal component analysis based on L1-norm maximization, IEEE Trans. Pattern Anal. Mach. Intell. 30 (9) (2008) 1672–1680.
[12] R. He, B. Hu, X. Yuan, W.-S. Zheng, Principal component analysis based on non-parametric maximum entropy, Neurocomputing 73 (10–12) (2010) 1840–1852.
[13] R. He, B.-G. Hu, W.-S. Zheng, X.-W. Kong, Robust principal component analysis based on maximum correntropy criterion, IEEE Trans. Image Process. 20 (6) (2011) 1485–1494.
[14] N. Kwak, Principal component analysis by l_p-norm maximization, IEEE Trans. Cybern. 44 (5) (2014) 594–609.
[15] Z. Liang, S. Xia, Y. Zhou, L. Zhang, Y. Li, Feature extraction based on L_p-norm generalized principal component analysis, Pattern Recognit. Lett. 34 (9) (2013) 1037–1045.
[16] J. Brooks, J. Dulá, E. Boone, A pure L_1-norm principal component analysis, Comput. Stat. Data Anal. 61 (2013) 83–98.

[17] A.Y. Ng, Feature selection, L_1 vs. L_2 regularization, and rotational invariance, in: Proceedings of the 21st International Conference on Machine Learning, ICML '04, 2004.
[18] T.M. Cover, J.A. Thomas, Elements of Information Theory, John Wiley & Sons, New York, 1981.
[19] W. Liu, P.P. Pokharel, J.C. Principe, Correntropy: properties and applications in non-Gaussian signal processing, IEEE Trans. Signal Process. 55 (11) (2007) 5286–5298.
[20] P. Bullen, Handbook of Means and Their Inequalities, 2nd edition, Springer, Netherlands, 2003.
[21] E.J. Candès, X. Li, Y. Ma, J. Wright, Robust principal component analysis? J. ACM 58 (3) (2011) 11:1–11:37.
[22] P.J. Huber, E.M. Ronchetti, Robust Statistics, 2nd edition, Wiley, New Jersey, 2009.
[23] G.H. Golub, C.F.V. Loan, Matrix Computations, 3rd edition, Johns Hopkins, Baltimore, 1996.
[24] J. Oh, N. Kwak, M. Lee, C.-H. Choi, Generalized mean for feature extraction in one-class classification problems, Pattern Recognit. 46 (12) (2013) 3328–3340.
[25] C.M. Bishop, Pattern Recognition and Machine Learning, Springer-Verlag, New York, 2006.
[26] D. Comaniciu, P. Meer, Mean shift: a robust approach toward feature space analysis, IEEE Trans. Pattern Anal. Mach. Intell. 24 (5) (2002) 603–619.
[27] P. Phillips, H. Moon, S. Rizvi, P. Rauss, The FERET evaluation methodology for face-recognition algorithms, IEEE Trans. Pattern Anal. Mach. Intell. 22 (10) (2000) 1090–1104.
[28] Y. LeCun, F.J. Huang, L. Bottou, Learning methods for generic object recognition with invariance to pose and lighting, in: Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol. 2, 2004, pp. II-97–104.
[29] S. Hauberg, A. Feragen, M. Black, Grassmann averages for scalable robust pca, in: Proceedings of the 2014 IEEE Conference on Computer Vision and Pattern Recognition, 2014, pp. 3810–3817.

Jiyong Oh received the B.S. degree from the School of Electronic Engineering, Ajou University, Korea in 2004 and the M.S. and Ph.D. degrees from the School of Electrical Engineering and Computer Science, Seoul National University, Korea in 2006 and 2012, respectively. He was a postdoctoral researcher in Sungkyunkwan and Ajou University, Korea in 2012 and 2013, respectively. Since Sept. 2013, he has been a research fellow in the graduate school of convergence science and technology, Seoul National University, Korea, where he currently holds a position of BK assistant professor. His research interests include feature extraction, machine learning, pattern recognition, computer vision, and their applications.

Nojun Kwak was born in Seoul, Korea in 1974. He received the BS, MS, and PhD degrees from the School of Electrical Engineering and Computer Science, Seoul National University, Seoul, Korea, in 1997, 1999 and 2003 respectively. From 2003 to 2006, he was with Samsung Electronics. In 2006, he joined Seoul National University as a BK21 Assistant Professor. From 2007 to 2013, he was a Faculty Member of the Department of Electrical and Computer Engineering, Ajou University, Suwon, Korea. Since 2013, he has been with the Graduate School of Convergence Science and Technology, Seoul National University, Seoul, Korea, where he is currently an Associate Professor. His current research interests include pattern recognition, machine learning, computer vision, data mining, image processing, and their applications.