Pattern Recognition 45 (2012) 487–497

Contents lists available at ScienceDirect

Pattern Recognition

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

Improve robustness of sparse PCA by L_1-norm maximization

Deyu Meng *, Qian Zhao, Zongben Xu

Institute for Information and System Sciences and Ministry of Education Key Lab for Intelligent Networks and Network Security, Xi'an Jiaotong University, Xi'an 710049, PR China

ARTICLE INFO

Article history:
Received 9 March 2010
Received in revised form
2 June 2011
Accepted 7 July 2011
Available online 19 July 2011

Keywords:
Face recognition
Noise
Outlier
Principal component analysis
Robust
Sparsity

ABSTRACT

Various sparse principal component analysis (PCA) methods have recently been proposed to enhance the interpretability of the classical PCA technique by extracting principal components (PCs) of the given data with sparse non-zero loadings. However, the performance of these methods is prone to be adversely affected by the presence of outliers and noises. To alleviate this problem, a new sparse PCA method is proposed in this paper. Instead of maximizing the L_2-norm variance of the input data as the conventional sparse PCA methods, the new method attempts to capture the maximal L_1-norm variance of the data, which is intrinsically less sensitive to noises and outliers. A simple algorithm for the method is specifically designed, which is easy to be implemented and converges to a local optimum of the problem. The efficiency and the robustness of the proposed method are theoretically analyzed and empirically verified by a series of experiments implemented on multiple synthetic and face reconstruction problems, as compared with the classical PCA method and other typical sparse PCA methods.

1. Introduction

Principal component analysis (PCA) is one of the most classical and popular techniques for data processing and dimensionality reduction, and has wide range of applications throughout science and engineering [1]. In essence, PCA seeks the so-called principal components (PCs) along which the data variance can be maximally preserved. By projecting the data into the low-dimensional linear subspace constituted by the PCs so extracted, the data structure in the original input space can be effectively captured.

Despite its many advantages, the traditional PCA suffers from the fact that each component is generally a linear combination of all the original variables and all weights in the linear combination, also known as loadings, are typically non-zeroes. In many applications, however, the original variables have meaningful physical interpretations. In biology for example, each involved variable might correspond to a specific gene. In these cases, the interpretation of the PCs will be facilitated if the derived PCs involve fewer non-zero loadings.

Accordingly, sparse PCA has been an active research topic for more than a decade, and a variety of methods for this topic have been developed [2–14]. For example, good results have been achieved by the SPCA algorithm of Zou et al., which is developed based on iterative elastic net regression [2]. D'Aspremont et al. proposed a method, called DSPCA, for finding sparse PCs by solving a sequence of semidefinite program relaxations of sparse PCA [3]. Journée et al. designed four algorithms (GPower$_{l_0}$, GPower$_{l_1}$, GPower$_{l_0,m}$, and GPower$_{l_1,m}$) for sparse PCA by formulating the issue as non-concave maximization problems with L_0- or L_1-norm sparsity-inducing penalties and extracting single unit sparse PC sequentially or block units ones simultaneously [4]. Based on expectation-maximization for probabilistic generative model of PCA, Sigg and Buhmann derived EMPCA for sparse and/or non-negative principal component analysis [5]. Very recently, Lu and Zhang developed an augmented Lagrangian method (ALSPCA briefly) for sparse PCA by solving a class of non-smooth constrained optimization problems [6]. Additionally, greedy methods were investigated for sparse PCA by Moghaddam et al. (GSPCA [7]) and d'Aspremont et al. (PathSPCA [8]). These methods have been successfully applied to many problems for extracting sparse and interpretable PCs from the given raw data.

However, the intrinsic principle underlying the current sparse PCA methods is to maximize the L_2-norm variance of the input data under certain sparsity constraint (which is to be introduced in detail toward the next section). This naturally conducts the problem that the methods are prone to the presence of outliers or noises due to the fact that the influence of outliers or noises with a large norm tends to be considerably exaggerated by the use of the L_2-norm. This robustness problem conducted by L_2-norm variance has been emphasized by multiple traditional PCA researchers [15–20], while has not been noted in sparse PCA area.

In this paper, instead of maximizing variance with intrinsic L_2-norm, a new optimization model that maximizes the L_1-norm variance is presented to achieve robust sparse PCA. A simple algorithm for solving the proposed L_1-norm optimization is correspondingly developed. The proposed algorithm is easy to

* Corresponding author. Tel.: +86 130 3290 4180; fax: +86 29 8266 8559.
E-mail address: dymeng@mail.xjtu.edu.cn (D. Meng).

0031-3203/$ - see front matter © 2011 Elsevier Ltd. All rights reserved.
doi:10.1016/j.patcog.2011.07.009

be implemented, and especially, it is theoretically evaluated that the computational speed of the new algorithm surprisingly exceeds many of the current sparse PCA methods. The algorithm is also proved to be able to converge to a reasonable local optimum of the original optimization model. By a series of experiments, it is verified that the proposed algorithm has an efficient and robust performance on data with intrinsic outliers and noises.

In what follows, the robustness problem of the current sparse PCA methods is first formulated in Section 2. The new robust sparse PCA algorithm is then proposed in Section 3. Also in this section the local optimality of the algorithm is proved and the computational complexity of the algorithm is evaluated. To verify the effectiveness of the proposed algorithm, results obtained from a series of empirical studies, as compared with those of other conventional methods, are analyzed and interpreted in Section 4. The paper is then concluded with a summary and outlook for future research.

2. Problem formulation

Denote the input data matrix as $X=[x_1,\ldots,x_n]\in R^{d\times n}$, where d and n are the dimensionality and the size of the given data, respectively. After a location transformation, we can assume all $\{x_i\}_{i=1}^n$ to have zero mean.

The classical PCA model tries to find an $m(<d)$ dimensional linear subspace where the variance of the input data X is maximized. Such a subspace can be achieved by solving the following optimization problem:

$$W^*=\arg\max_W \|W^TXX^TW\|_2=\|W^TX\|_2^2,\quad \text{subject to } W^TW=I_m, \quad (1)$$

where $W=[w_1,w_2,\ldots,w_m]\in R^{d\times m}$, where each column w_k of W corresponds to the k-th PC of the original data and $\|\cdot\|_2$ denotes the L_2-norm of a matrix or a vector. Under the constraint that $W^TW=I_m$, it is known that all $\{w_k\}_{k=1}^m$ constitute the regular orthogonal bases of the m-dimensional linear subspace where the maximal L_2-norm variance of X is captured.

Sparse PCA model aims at achieving sparse PCs on which maximal amount of data variance can be possibly obtained. This aim can be attained by solving the following optimization:

$$W^*=\arg\max_W \|W^TX\|_2^2,\quad \text{subject to } W^TW=I_m,\quad \|W\|_0<k. \quad (2)$$

Note that the only difference between the optimizations (1) and (2) for classical PCA and sparse PCA is that the latter involves an extra l_0 penalty, i.e., $\|W\|_0<k$, to enforce sparsity of the output PCs.

It is easy to see that the optimization (2) is a hard combinatorial problem and very difficult to solve. Hence a more generally employed sparse PCA formulation is to relax the non-convex l_0 penalty to a weaker but convex l_1 penalty, i.e., $\|W\|_1<t$. This leads to the following amended optimization:

$$W^*=\arg\max_W \|W^TX\|_2^2,\quad \text{subject to } W^TW=I_m,\quad \|W\|_1<t. \quad (3)$$

The formulation (3), as well as (2), constitutes the fundament of most of current sparse PCA methods [2–14].

Note that the objective of both of the above optimizations for sparse PCA is to maximize the data variance with intrinsic L_2-norm, i.e., $\|W^TX\|_2^2$. Yet it is known that the L_2-norm variance is sensitive to outliers and noises with large norms [15–23]. This phenomenon is graphically depicted in Fig. 1, which shows the L_2-norm variance curve $f(x)=\|x\|_2^2$ and L_1-norm one $f(x)=\|x\|_1$, respectively. From the figure, the exaggerative effect of L_2-norm variance at points with large norms, as compared with the L_1-norm one, is evident. This on one hand clarifies the robust problem of the traditional sparse PCA methods on the data with heavy outlier or noise pollution, and on the other hand implies a meliorative strategy to this problem by substituting the L_1-norm variance $\|W^TX\|_1$ for the L_2-norm one $\|W^TX\|_2^2$ in optimization problems (2) or (3).[1]

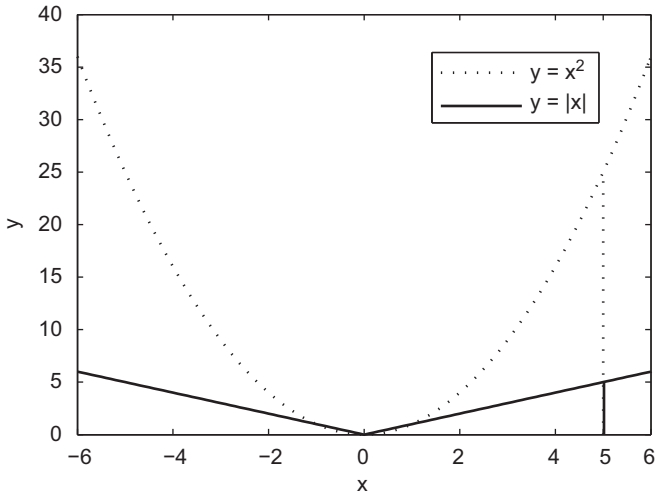

Fig. 1. Graphical presentation of the exaggerative effect of the L_2-norm variance curve, as compared with the L_1-norm curve. In particular, at point $x=5$, $\|x\|_2^2$ is dominantly (five times) larger than $\|x\|_1$.

Motivated by the above analysis, we formulate the following optimization to realize the robust sparse PCA:

$$W^*=\arg\max_W \|W^TX\|_1,\quad \text{subject to } W^TW=I_m,\quad \|W\|_1<t, \quad (4)$$

which is expected to be more robust to outliers and noises than the traditional sparse PCA techniques.

One downside of (4) is that the optimal i-th PC w_i yielded from (4) varies with different preset number m of PCs. Besides, finding a global solution of (4) for $m>1$ is very difficult. To ameliorate the problems, we simplify the problem (4) into a sequence of $m=1$ optimizations using a greedy search strategy. That is, (4) is simplified as the following optimization problem:

$$w^*=\arg\max_w \|X^Tw\|_1,\quad \text{subject to } w^Tw=1,\quad \|w\|_1<t. \quad (5)$$

Although the successive greedy solutions of (5) may differ from the optimal solution of (4), it is expected to provide a good approximation for (4). In the following, an efficient algorithm to solve (5) is first introduced and the greedy algorithm for searching $m>1$ PCs is then presented.

3. Robust sparse PCA

Even for the simplified problem (5), it is difficult to solve it by traditional optimization techniques due to its absolute value operations both on objective function and constraint. In this paper, a simple while efficient algorithm (called the robust sparse PCA algorithm, or simply RSPCA algorithm) is especially designed for (5), which is introduced in the following.

[1] It should be noted that the similar idea, i.e., substituting the L_1-norm objective for the L_2-norm one, has been employed by multiple machine learning algorithms to enhance the robustness of the related problems, such as the robust face recognition algorithm proposed in [22] and the robust PCA algorithm proposed in [23]. In this sense, these algorithms, including the proposed algorithm, are related to each other to a certain extent.

3.1. RSPCA algorithm for one sparse PC

The new algorithm for solving the optimization (5) is listed as follows. The initialization of the algorithm is to be discussed toward the end of this section. In the algorithm, $X=[x_1,\ldots,x_n]\in R^{d\times n}$ denotes the data matrix, $w(0)$ the initialized PC vector, $w(t)$ the sparse PC vector in t-th iteration, and w^* the sparse PC the algorithm finally converges to.

Algorithm 1. RSPCA algorithm for one sparse PC.

Input: data matrix X, sparsity k.

(1) Initialize $w(0)$; set $w(0)=\frac{w(0)}{\|w(0)\|_2}$ and $t=0$.

(2) Set $v=(v_1,\ldots,v_d)^T=\sum_{i=1}^n p_i(t)x_i$, where $p_i(t)=\begin{cases}1 & \text{if } w^T(t)x_i\geq 0\\ -1 & \text{if } w^T(t)x_i<0\end{cases}$; let γ be the $(k+1)$-th largest element of $|v|$.

(3) Let $\beta=(\beta_1,\ldots,\beta_d)^T$, where $\beta_i=\mathrm{sgn}(v_i)(|v_i|-\gamma)_+$ for $i=1,\ldots,d$. Here $(x)_+=\begin{cases}x, x>0\\ 0, x\leq 0\end{cases}$ and $\mathrm{sgn}(x)=\begin{cases}1, x>0\\ 0, x=0\\ -1, x<0\end{cases}$ denote the thresholding and sign functions, respectively. Set $w(t+1)=\frac{\beta}{\|\beta\|_2}$, and $t=t+1$.

(4) *Convergence check*:

(4.1) If $w(t)\neq w(t-1)$, go to step (2); otherwise, check (4.2).

(4.2) If thereexists i such that $w^T(t)x_i=0$ and $|\mathrm{sgn}(w^T(t))||\mathrm{sgn}(x_i)|\neq 0$, then let $\frac{w^T(t)+\Delta w}{\|w^T(t)+\Delta w\|_2}$ and go to step (2); otherwise, go to (4.3). Here Δw is a small non-zero random vector.

(4.3) Set $w^*=w(t)$ and stop iteration.

Output: The k sparse PC w^*.

The convergence of the above algorithm and the rationality of the obtained w^* are theoretically substantiated by the following theorem.

Theorem 1. *By implementing Algorithm 1, $w(t)$ converges to a k-sparse vector w^*, which is a local maximum point of $\|X^Tw\|_1$ in the k-dimensional subspace where the non-zero loadings of w^* are located (denoted as the k-subspace of w^* briefly in the following).*

We first present a lemma which is necessary to prove the above theorem.

Lemma 1. *Given the vector $v=(v_1,\ldots,v_d)^T$, the solution of the following optimization problem*

$$\max_w w^Tv,\quad \text{subject to } w^Tw=1,\quad \|w\|_1<t \tag{6}$$

is of the following form

$$w^*=\frac{\beta}{\|\beta\|_2} \tag{7}$$

where $\beta=(\beta_1,\ldots,\beta_d)^T$ and

$$\beta_i=\mathrm{sgn}(v_i)(|v_i|-\gamma)_+,\quad i=1,\ldots,d. \tag{8}$$

Furthermore, if the sparsity of the solution w^* is known to be k beforehand, then $\gamma=\theta_{k+1}$, where θ_k denotes the k-th largest element of $|v|$.

Proof of Lemma 1. We first prove that the solution of the optimization problem (6) can be expressed as the (7) form.

It is known that the Lagrangian formulation of (6) is

$$L(w)=w^Tv-\alpha(w^Tw-1)-\gamma(\|w\|_1-t),$$

where $\alpha\in R$ and $\gamma>0$ are Lagrangian multipliers. Since $\partial L(w)/\partial w=v-2\alpha w-\gamma\,\mathrm{sgn}(w)$, it follows that the optimal solution w^* of (6) corresponds to

$$w^*=\frac{1}{2\alpha}\beta=\frac{1}{2\alpha}(\beta_1,\ldots,\beta_d)^T,$$

where $\beta_i=\mathrm{sgn}(v_i)(|v_i|-\gamma)_+$, $\forall i=1,\ldots,d$. To further make the constraint $w^Tw=1$, we have that $\alpha=\|\beta\|_2/2$. That is, the optimal solution of (6) is of the form $w^*=\beta/\|\beta\|_2$.

Then we prove that if the sparsity of the solution w^* is known to be k, it holds that γ in (8) should be θ_{k+1}.

Since the w^* is known to be k-sparse, based on (7) and (8), it is evident that γ should be evaluated in the interval $[\theta_{k+1},\theta_k)$. Let $\tilde v=(\tilde v_1,\ldots,\tilde v_d)^T$, where $\tilde v_i=v_i$ if $|v_i|\geq\theta_{k+1}$, otherwise $\tilde v_i=0$, and $e=\mathrm{sgn}(\tilde v)$. Based on (7) and (8), it is easy to obtain that when $\gamma\in[\theta_{k+1},\theta_k)$, it holds that

$$w^{*T}v=\frac{(\tilde v-\gamma e)^T\tilde v}{\|\tilde v-\gamma e\|_2}:=f(\gamma).$$

Differentiate $f(\gamma)$ w.r.t. γ, we then get

$$f'(\gamma)=-\tilde v^Te(\tilde v-\gamma e)^2+\frac{\tilde v^T(\tilde v-\gamma e)(\tilde v-\gamma e)^Te}{\|\tilde v-\gamma e\|_2^3}$$
$$=-\frac{\gamma\left(1-\left(\left(\frac{\tilde v}{\|\tilde v\|_2}\right)^T\left(\frac{e}{\|e\|_2}\right)\right)^2\right)}{\|\tilde v\|_2^2\|e\|_2^2\|\tilde v-\gamma e\|_2^3}\leq 0.$$

That is, $f(\gamma)$ monotonically decreases w.r.t. γ in the interval $[\theta_{k+1},\theta_k)$. Then it is easy to see that to maximize the objective $w^{*T}v$ in optimization problem (6), γ should be set as θ_{k+1}. The proof is then completed. □

Based on Lemma 1, Theorem 1 is then proved as follows.

Proof of Theorem 1. Inspired by the idea presented in [15], the convergence of $w(t)$ is proved by verifying the nondecreasing property of $\|X^Tw(t)\|$ w.r.t. t as follows:

$$\|X^Tw(t)\|_1=\sum_{i=1}^n|w^T(t)x_i|=w^T(t)\sum_{i=1}^n p_i(t)x_i\geq w^T(t)\sum_{i=1}^n p_i(t-1)x_i$$
$$\geq w^T(t-1)\sum_{i=1}^n p_i(t-1)x_i=\sum_{i=1}^n|w^T(t-1)x_i|=\|X^Tw(t-1)\|_1.$$

Because the objective function $\|X^Tw(t)\|_1$ is obviously bounded and nondecreasing w.r.t. t, the convergence of Algorithm 1 is then naturally conducted. In the above deduction, due to the fact that $p_i(t)w^T(t)x_i\geq 0$ for all i, the first inequality is evident. The second inequality holds since for any t, $w(t)$ is the k-sparse unit vector which maximizes the inner product of $w(t)^Tv(t-1)=\sum_{i=1}^n w(t)^T(p_i(t-1)x_i)$ according to Lemma 1.

Then we prove the local optimality of the k-sparse point w^* yielded by the algorithm in its located k-subspace.

For all $i=1,2,\ldots,n$, let $p_i=-1$ if $w^{*T}x_i<0$; otherwise let $p_i=1$. Due to the convergence condition (4.2) of the algorithm, it is evident that $w^{*T}p_ix_i>0$ for any x_i satisfying $|\mathrm{sgn}(w^{*T})||x_i|\neq 0$. For such x_i, it is easy to conduct that in a small neighborhood $N(w^*)$ of w^* in its k-subspace, it holds that for any $w\in N(w^*)$, $w^Tp_ix_i\geq 0$, i.e., $w^Tp_ix_i=|w^Tx_i|$. Besides, if x_i satisfies $|\mathrm{sgn}(w^{*T})||x_i|=0$, it is easy to conduct that the k loadings of such x_i in the k-subspace of w^* are all zeroes. This implies that for any w in the k-subspace of w^*, it holds that $w^Tx_i=w^Tp_ix_i=|w^Tx_i|=0$. Accordingly, it follows that

for any $w \in N(w^*)$, $w^T p_i x_i = |w^T x_i|$. Then according to Lemma 1 and the convergence of our algorithm as proved above, w^* is the optimal k-sparse unit vector to maximize $w^T p_i x_i$, and it naturally follows that $\|X^T w^*\|_1 = w^{*T} \sum_{i=1}^n p_i x_i \geq w^T \sum_{i=1}^n p_i x_i = \|X^T w\|_1$ for all $w \in N(w^*)$.

Thus, w^* yielded by Algorithm 1 corresponds to a local maximum of $\|X^T w\|_1$ in its k-subspace. □

So far, we have clarified that Algorithm 1 tends to attain a reasonable k-sparse PC of the given data. In the following we further extend this algorithm to a heuristic greedy strategy for finding an arbitrary number of k-sparse PCs for RSPCA model (4).

3.2. RSPCA algorithm for m sparse PCs

The RSPCA algorithm for $m(>1)$ sparse PCs is constructed by applying Algorithm 1 greedily to the remainder of the projected samples X^j. The procedure is listed as follows.

Algorithm 2. RSPCA algorithm for m sparse PCs.

Input: data matrix X, sparsity k, desired PC number $m>1$;

(1) Set $w_0 = \vec{0} \in R^d$, where $\vec{0}$ is the all-zero vector; denote $X^0 = \{x_i^0 = x_i\}_{i=1}^n$.

(2) For $j = 1, \ldots, m$, do the following iteration:

(2.1) Let $X^j = \{x_i^j = x_i^{j-1} - w_{j-1}(w_{j-1}^T x_i^{j-1})\}_{i=1}^n$.

(2.2) Apply Algorithm 1 to the projected data X^j to get the k-sparse PC vector w_j.

End for

Output: m k-sparse PCs $\{w_i\}_{i=1}^m$.

It should be noted that the sparse PCs yielded by the proposed heuristic algorithm only offer an approximate solution to (4). On one hand, only a local maximum of the L_1-norm optimization (4) can be achieved by Algorithm 1; and on the other hand, the orthonormality of the projection vectors $\{w_i\}_{i=1}^m$ generated by Algorithm 2 cannot be theoretically guaranteed. However, since in each iteration of the algorithm, the projected samples X^j are in fact located in the subspace orthogonal to the $(j-1)$-dimensional space spanned by $\{w_i\}_{i=1}^{j-1}$, i.e., $w_k^T x_i^j = 0$ for all $i = 1, \ldots, n$ and $k = 1, \ldots, j-1$, the k-sparse PC w_j obtained from the projected data X^j also inclines to be approximately orthogonal to all w_is $(i = 1, \ldots, j-1)$. Besides, despite the heuristic approximation of Algorithm 1, the proposed algorithm is expected to provide good projections that can possibly capture a large L_1 dispersion of the original data, and hence offer good robust sparse PCs. All the aforementioned is further to be verified by experiments depicted in the next section.

An important issue still remains in the implementation of the proposed algorithm: the initial w_0 in Algorithm 1 needs to be properly specified to guarantee that the algorithm can converge to a good local optimal solution. Here two strategies are suggested. The first is to specify w_0 as the solution of the classical PCA. Since PCA precisely attains the global optimal solution where the L_2-norm variance of the original data is maximized, it is expected that the proposed algorithm could also converge to a good sparse PC by starting the iteration from the PCA solution. Yet the downside is that the supplemental implementation of PCA in step (1) might materially increase the computational complexity of the proposed algorithm, especially for large data set. The second strategy is to run the proposed algorithm multiple times with different initial w_0 (which can be easily specified as the random vector or simple all-0 or all-1 vector) and output the solution that gives the maximal L_1 dispersion. This strategy is simple and easy to be implemented, and hence was employed for specification of initial w_0 in our experiments.

The computation of the proposed algorithm is mainly costed on its iterative process, i.e., steps (2)–(4) of Algorithm 1. Evidently, only simple vector computation is involved in these steps, and it is easy to obtain that the computational complexity of the whole algorithm is around $O(nd \log d) \times n_{it}$, where n_{it} is the number of the

Table 1
Performance comparison of 11 methods, including the classical PCA method, nine current sparse PCA methods, and the proposed RSPCA method, by applying them to the toy data with intrinsic two outliers. The iteration time t_1+t_2 of a method denotes that it needs t_1 and t_2 iterations to compute PC1 and PC2, respectively.

Methods	PC1	PC2	ARSE	Iteration times
PCA	(0.6811,0.7322)	(−0.7322,0.6811)	1.0155	0
SPCA	(0,−1)	(1,0)	1.2500	4
DSPCA	(0,1)	(−1,0)	1.2500	8+4
PathSPCA	(0,1)	(1,0)	1.2500	1+1
EMPCA	(0,1)	(1,0)	1.2500	2+1
GPower$_{l_1}$	(0,1)	(1,0)	1.2500	3+3
GPower$_{l_0}$	(0,1)	(1,0)	1.2500	3+3
GPower$_{l_1,m}$	(0,1)	(0.9945,0.1048)	1.2500	5
GPower$_{l_0,m}$	(0.6766,0.7363)	(0,0)	1.0188	11
ALSPCA	(0,1)	(1,−0.003)	1.2500	3
RSPCA	(1,0)	(0,1)	0.5720	2+2

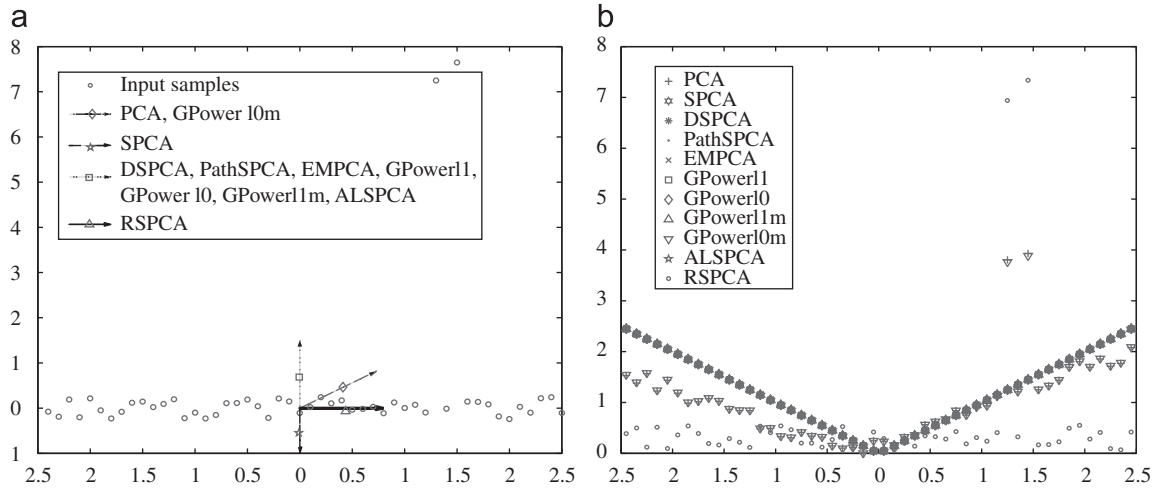

Fig. 2. (a) The toy data points with two intrinsic outliers, and the first PCs yielded by applying the PCA, SPCA, DSPCA, PathSPCA, EMPCA, GPower$_{l_1}$, GPower$_{l_0}$, GPower$_{l_1,m}$, GPower$_{l_0,m}$, ALSPCA, and RSPCA methods to this data set. (b) Residual errors of the data by projecting them to the first PC calculated by the 11 methods.

iterations for convergence. As compared with the computational complexities of most of the current sparse PCA methods, such as $O(nd^3)\times n_{it}$ of SPCA, $O(nd^4\log d)\times n_{it}$ of DSPCA, $O(nd\log d)\times n_{it}$ of EMPCA, $O(nd^2)\times n_{it}$ of ALSPCA, $O(nd)\times n_{it}$ of GPowerl_0, and $O(n^3d)\times n_{it}$ of PathSPCA (where n_{it} is the iteration time of the corresponding method), the proposed algorithm does not substantially increase (for EMPCA and GPower$_{l_0}$ methods), or even decrease (for other methods) the computational time for sparse PCA calculation. Besides, the iteration number n_{it} of the proposed algorithm is generally very small, further conducting the efficiency of the proposed algorithm. All of the aforementioned will be further verified by the simulation results given in the next section.

4. Experiment results

To evaluate the performance, especially the robustness, of the proposed method, it was applied to problems with intrinsic noises and outliers to different extents. For comparison, the classical PCA and nine of the current sparse PCA methods, including SPCA [2], DSPCA [3], PathSPCA [8], EMPCA [5], GPower$_{l_1}$, GPower$_{l_0}$, GPower$_{l_1,m}$, GPower$_{l_0,m}$ [4], and ALSPCA [6] methods, have also been utilized. The results are summarized in the following discussion. All programs were implemented under Matlab 7.0 platform. The implementation environment was the personal computer with Intel Core(TM)2 Q9300@2.50 G (CPU), 3.25 GB (memory), and Windows XP (OS).

4.1. A toy problem with two outliers

The performance of the proposed RSPCA method was first evaluated on the 2D toy data $\{x_i,y_i\}_{i=1}^{50}$ as depicted in Fig. 2(a). The data were generated by picking x_i from -2.4 to 2.5 with the similar interval 0.1, and yielding y_i from the uniform distribution on $[-0.25,0.25]$, except that at $x_i=1.3$ and $x_i=1.5$, y_is were set values around 7. Evidently, the data contain two intrinsic outliers, and if we discard the outliers, the first principal component of the data should be the sparse vector (1,0).

The first PC vectors obtained by applying the classical PCA method, nine existing sparse PCA methods, and the RSPCA method to the toy data are depicted in Fig. 2(a) and also listed in Table 1. Fig. 2(b) shows the residual error of each x_i $(i=1,\ldots,n)$ conducted by each of the 11 employed methods. Here the residual error of x_i is calculated by $e_i=|x_i-ww^Tx_i|$, where w is the first PC vector obtained from the corresponding method. Furthermore, the average residual errors (denoted as ARSE in brief) of 11 utilized methods and the iteration times of these methods on calculating the PCs of the toy data are listed in Table 1 for further comparison.

By observing Fig. 2 and Table 1, it is evident that the RSPCA outperforms the other methods in the toy problem. First, the RSPCA attains the accurate sparse PC (1,0) of the original data set, while all of the other 10 methods do not. Second, RSPCA achieves the smallest ARSE of all of the 11 utilized methods. These results show that the other methods are much influenced by the outliers than the proposed method. Besides, from Table 1, it is impressive that the RSPCA only needs four iterations to attain such a robust result, no larger than most of the other sparse PCA methods. This further verifies the efficiency of the proposed method in the outlier case.

4.2. Tests on benchmark data with intrinsic noises and outliers

In this section we consider the data first proposed by [2]. The data set contains a collection of 10D data points $(x_1,\ldots,x_{10})^T$

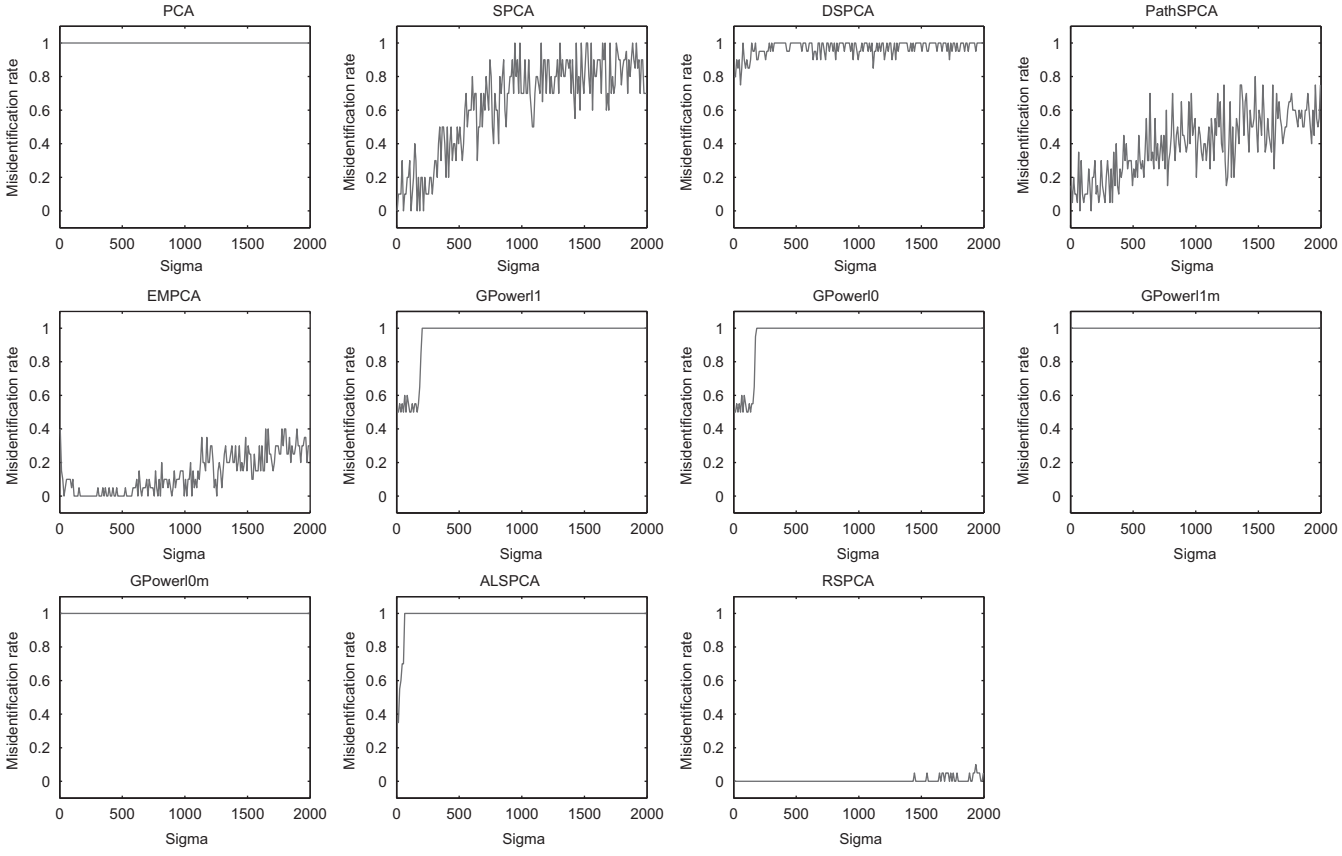

Fig. 3. The tendency curves of the misidentification rate w.r.t. the noise extent σ corresponding to the classical PCA method, nine current sparse PCA methods, and the RSPCA method, respectively.

generated via the following two processes: first three hidden factors were created:

$$V_1 \sim N(0,290), \quad V_2 \sim N(0,300), \quad V_3 = -0.3V_1 + 0.925V_2 + \varepsilon,$$

where $\varepsilon \sim N(0,1)$, and V_1,V_2 and ε are independent; afterwards, 10 observed variables were generated as

$$x_i = V_j + \varepsilon_i^j, \quad \varepsilon_i^j \sim N(0,\sigma) \tag{9}$$

with $j=1$ for $i=1,2,3,4$, $j=2$ for $i=5,6,7,8$, $j=3$ for $i=9, 10$, $\sigma=1$ and all ε_i^j s independent. It has been clarified that the data so generated are of intrinsic sparse PCs [2]. In particular, the first PC should recover the factor V2 only using (x_5,x_6,x_7,x_8), and the second should recover V1 only using (x_1,x_2,x_3,x_4). This type of data is one of the most frequently utilized benchmark examples to evaluate the performance of the sparse PCA method [3,6], and hence employed here to verify the robustness of the proposed method. Specifically, two sequences of data were constructed by blending such benchmark data with noises and outliers respectively as follows:

- *Noise data sequence*: Contain 2000 collections of 10D data sets, each with size 10,000. Each data set in the sequence was generated from the benchmark distribution as formulated in (9) with noise extent σ varying from 1 to 2000 at regular interval 1.
- *Outlier data sequence*: Contain 3000 collections of 10D data sets, each with size 10,000. In each data, 9500 points were generated via the aforementioned benchmark process, and 500 ones were obtained by letting $x_i=0$, for $i=1,\ldots,8$, and $x_i=\xi_i$ for $i=9, 10$, where $\xi_i \sim N(0,\sigma)$ $(i=9, 10)$ and ξ_9 and ξ_{10} are independent. By varying the outlier extent σ from 1 to 3000 with the fixed interval 1, the sequence of outlier data sets was then yielded. Evidently, 5% outliers are intrinsically mixed in each of the data set so generated.

For each data of the above cases, the classical PCA, the nine current sparse PCA methods, and the RSPCA method were, respectively, employed to calculate the first two PCs of the data. By virtue of the oracle information of the ideal sparse PCs (i.e., the positions of the intrinsic non-zero loadings of the first two PCs), the misidentification rate (MR in brief) of each method corresponding to the data can thus be obtained. By calculating the average of every 10 successive MR values so obtained, the tendency curves of MR w.r.t. the noise extent and the outlier extent were then attained (with lengths 200 and 300), as depicted in Figs. 3 and 4, respectively. To make a clearer clarification, Tables 2 and 3 summarize performance of the 11 employed methods when they were applied to the data sets with the largest noise ($\sigma=2000$) and outlier ($\sigma=3000$) extents, respectively.

From Figs. 3 and 4, it is easy to observe that both MR tendency curves of the RSPCA w.r.t. the noise and outlier extents are always located at or very close to 0. Combined with Tables 2 and 3, it is apparent that the RSPCA method robustly delivers the ideal sparse representations of the first two PCs underlying the data, and has a stable performance w.r.t. different extents of noises and outliers. As compared with the tendency curves of the other 10 methods, it is evident that the proposed method significantly improves the robustness of the current sparse PCA methods. Furthermore, it is seen from Tables 2 and 3 that for each of the listed outlier and noise cases, the iteration time of the RSPCA method is the smallest of all of the utilized 10 sparse PCA methods. This further implies the efficiency of the proposed method.

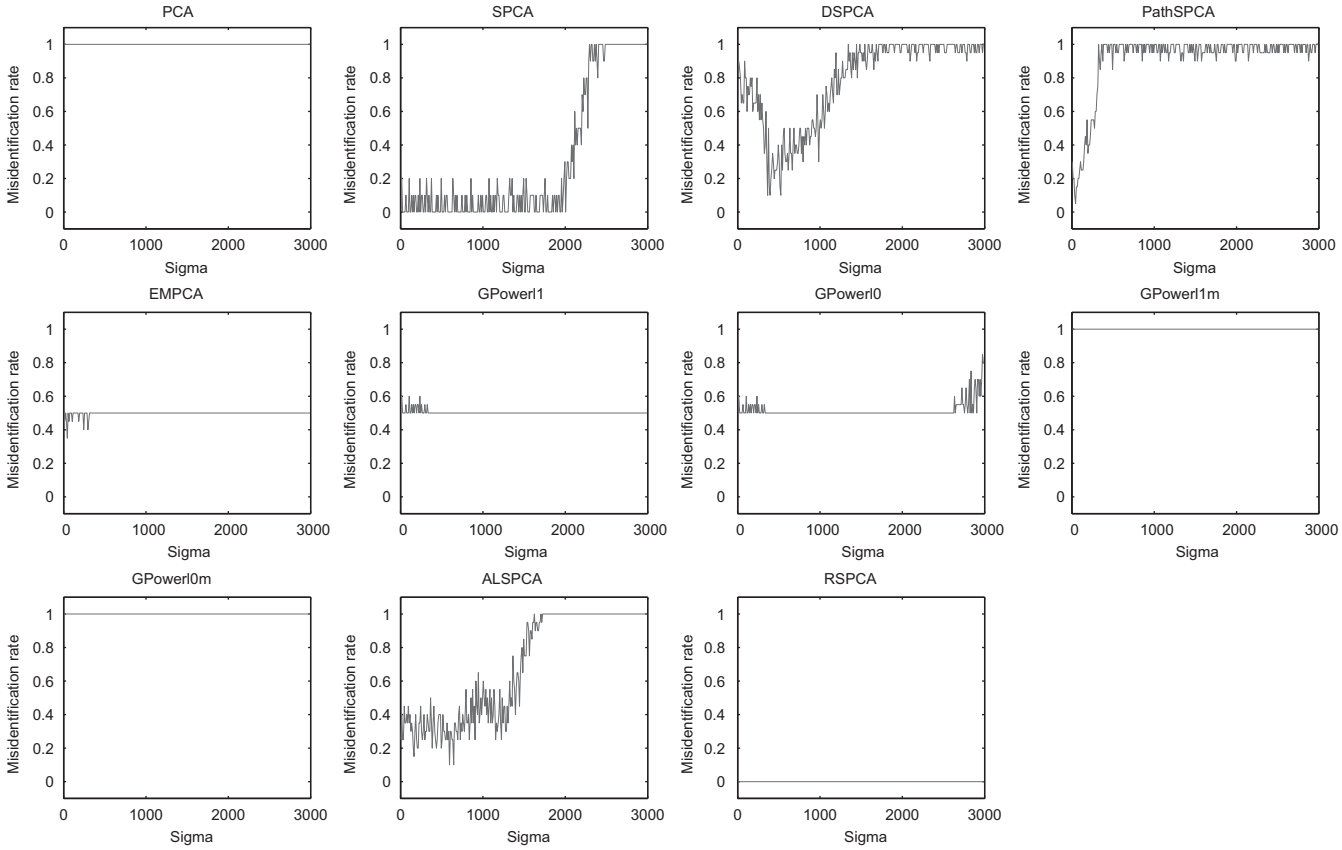

Fig. 4. The tendency curves of the misidentification rate w.r.t. the outlier extent σ corresponding to the classical PCA method, nine current sparse PCA methods, and the RSPCA method, respectively.

Table 2
Performance comparison of the classical PCA method, nine current sparse PCA methods, and the proposed RSPCA method by applying them to the benchmark data with noise extent $\sigma = 2000$. Here IT denotes the iteration time of the corresponding method on calculating the PCs.

PCA		SPCA		DSPCA		PathSPCA		EMPCA	
PC1	PC2	PC1	PC2	PC1	PC2	PC1	PC2	PC1	PC2
−0.1314	−0.4858	0.5290	0	0	−0.8513	0	−0.5176	0	−0.5013
−0.0761	−0.4850	0.4878	0	0	−0.4320	0	−0.4908	0	−0.5081
−0.0853	−0.4824	0.5014	0	0	−0.2979	0	−0.5020	0	−0.5174
−0.0849	−0.4676	0.4805	0	0	0	0	−0.4891	0	−0.4720
0.4011	−0.1450	0	0.5522	0.6121	0	−0.5049	0	0	0
0.3968	−0.1094	0	0.3149	0.1301	0	−0.4889	0	−0.5017	0
0.4146	−0.1430	0	0.7386	0.7776	0	−0.5253	0	−0.5422	0
0.3809	−0.1432	0	0.2242	0.0613	0	−0.4797	0	−0.4768	0
0.4070	0.0482	0	0	0	0	0	0	0	0
0.4020	0.0351	0	0	0	0	0	0	−0.4764	0
		IT: 110		IT: 15+12		IT: 4+4		IT: 341+48	

$GPower_{l_1}$		$GPower_{l_0}$		$GPower_{l_1,m}$		$GPower_{l_0,m}$		ALSPCA		RSPCA	
PC1	PC2	PC1	PC2	PC1	PC2	PC1	PC2	PC1	PC2	PC1	PC2
0	0	0	0	0.1231	0.4883	0.0487	0.4854	0	0.9949	0	0.4952
0	0	0	0	0.0679	0.4864	0.0539	0.4933	0	0	0	0.4662
0	0	0	0	0.0771	0.4840	0.0641	0.4803	0	0	0	0.5383
0	0	0	0	0.0769	0.4692	0.0289	0.5019	0	0	0	0.4977
0	1	0	1	−0.4035	0.1368	−0.4147	0.1098	0	0	0.5504	0
0	0	0	0	−0.3986	0.1013	−0.3967	0.0899	0	0	0.3476	0
1	0	1	0	−0.4169	0.1345	−0.4165	0.0598	−1	0	0.6621	0
0	0	0	0	−0.3832	0.1354	−0.4023	0.0851	0	0.1008	0.3712	0
0	0	0	0	−0.4061	−0.0564	−0.4155	−0.0527	0	0	0	0
0	0	0	0	−0.4014	−0.0433	−0.3905	−0.0680	0	0	0	0
IT: 3+3		IT: 3+3		IT: 17		IT: 16		IT: 7		IT: 2+2	

Table 3
Performance comparison of the classical PCA method, nine current sparse PCA methods, and the proposed RSPCA method by applying them to the benchmark data with outlier extent $\sigma = 3000$.

PCA		SPCA		DSPCA		PathSPCA		EMPCA	
PC1	PC2	PC1	PC2	PC1	PC2	PC1	PC2	PC1	PC2
−0.1120	−0.4786	0.5003	0	0	−0.6777	−0.0311	0	0	−0.5002
−0.1122	−0.4785	0.5003	0	0	−0.7092	0	0	0	−0.5002
−0.1121	−0.4783	0.4999	0	0	−0.1942	0	0	0	−0.4999
−0.1120	−0.4782	0.4995	0	0	0	0	−0.0267	0	−0.4997
0.3841	−0.1453	0	0.0155	0.0773	0	0.4574	0	−0.4881	0
0.3843	−0.1452	0	0.0223	0.0784	0	0.3574	0.5477	−0.4883	0
0.3843	−0.1448	0	0.0185	0.0785	0	0	−0.7999	−0.4882	0
0.3837	−0.1455	0	0	0.0748	0	−0.8137	0.2439	0	0
0.4254	0.0101	0	0.9995	0.9250	0	0	0	−0.5339	0
0.4227	0.0102	0	0	0.3471	0	0	0	0	0
		IT: 10		IT: 17+13		IT: 4+4		IT: 48+4	

$GPower_{l_1}$		$GPower_{l_0}$		$GPower_{l_1,m}$		$GPower_{l_0,m}$		ALSPCA		RSPCA	
PC1	PC2	PC1	PC2	PC1	PC2	PC1	PC2	PC1	PC2	PC1	PC2
0	0	0	0	−0.3717	−0.3070	0	0.0001	0	0	0	0.4998
0	0	0	0	0.2652	0.3032	0.0004	0	0	0	0	0.5006
0	0	0	0	−0.3988	−0.4566	−0.0005	0	0	0.8986	0	0.5000
0	0	0	0	0.2663	0.3039	0.0001	0	0	0	0	0.4995
0	0	0	0	0.2090	−0.2508	−0.0001	0.0001	0	0	0.5013	0
0	0	0	0	−0.1148	0.3606	0.0001	0	0	0	0.5036	0
0	0	0	0	0.2083	−0.2512	−0.0001	−0.0002	0	0	0.5016	0
0	0	0	0	0.2085	−0.2511	0.0001	0	0	0.4387	0.4934	0
1	0	1	0	−0.6331	0.4264	0.7137	0.6985	1	0	0	0
0	1	0	1	−0.1309	−0.1097	0.7004	−0.7156	0	0	0	0
IT: 3+3		IT: 3+3		IT: 4		IT: 4		IT: 5		IT: 2+2	

4.3. Face reconstruction problems with occluded and dummy images

The proposed method was also applied to face reconstruction problems and the performance was compared with those of the other methods. The employed data set is the well-known Yale face database [15,21]. The database contains 165 gray-scale images of 15 individuals, and there are 11 images per subject, one per different facial expression or configuration. The data can be downloaded from

the website "http://cvc.yale.edu/projects/yalefaces/yale-faces.html". The original pixel size of the face image is 320×243. In our experiments, preprocessing was first performed to crop the original face images: the facial areas were cropped into the final images for matching, and the size of each cropped image is 171×215. Some typical cropped faces are depicted in Figs. 5 and 6. Each pixel was regarded as an input variable, and hence the input space of the database is of intrinsic 36,765 dimensionality.

Based on the database so generated, the occluded and dummy databases were then generated as follows, with intrinsic noises and outliers, respectively.

- *Occluded face data*: Randomly pick two images from each of the 15 individuals' face images, and then occlude them with a rectangular noise consisting of random black and white dots whose size was 80×160 or 150×70, located at a random position of the image. Fig. 5 shows typical images so occluded.
- *Dummy face data*: Add 30 dummy images which consist of random black and white dots to the original 165 cropped face images to constitute the dummy data set.

We have performed the classical PCA, nine existing sparse PCA methods, and the RSPCA method on the occluded and dummy face databases, respectively, while each of the SPCA, DSPCA, PathSPCA, GPower$_{l_1,m}$, GPower$_{l_0,m}$, ALSPCA methods encountered the "out of memory" problem and could not be executed. Consequently, only the performance of PCA, EMPCA, GPower$_{l_1}$, GPower$_{l_0}$, and RSPCA is involved for the following substantiation.

Our aim is to detect how well the images could be reconstructed by utilizing only a small number of PCs extracted from the employed methods. By taking the average reconstruction error (ARCE in brief [15]) as the criterion, the quality of the reconstruction can be quantitatively assessed. The ARCE of each method with the first m PCs is calculated as [15]: $e(m) = (1/n)\sum_{i=1}^{n} \|x_i^{org} - \sum_{j=1}^{m} w_j w_j^T x_i\|_2$, where $n=165$ is the number of original face samples, x_i^{org} and x_i are the i-th original image and the corresponding one in the occluded or dummy data sets, respectively, and m is the number of the involved PCs. Fig. 7(a) and (b) show the ARCE tendency curves of the employed five methods with various numbers of extracted PCs in occluded and dummy cases, respectively. Besides, Figs. 5 and 6 demonstrate some of the original and the reconstructed images by projecting the occluded and the dummy images into the subspaces constituted by the 20 and the 30 PCs calculated from the five employed methods, respectively. Tables 4 and 5 further list the summarizations of the performance of the five methods on the two utilized databases, respectively.

For occluded case, it is seen from Fig. 7(a) that when the number m of the extracted PCs is small, the ARCE value of the RSPCA is non-substantially larger than those of the other four methods. Yet from around $m=20$, it is apparent that the RSPCA starts to be better than the other methods. From Fig. 5, we can observe that when projected into the subspace constituted by the first 20 PCs yielded from different methods, the images reconstructed by the RSPCA eliminate the largest extent of noises of the original occluded images. Considering that ARCE curve of the RSPCA tends to be decreasing while those of the other methods incline to be increasing from $m=20$, the advantage of the RSPCA is evident.

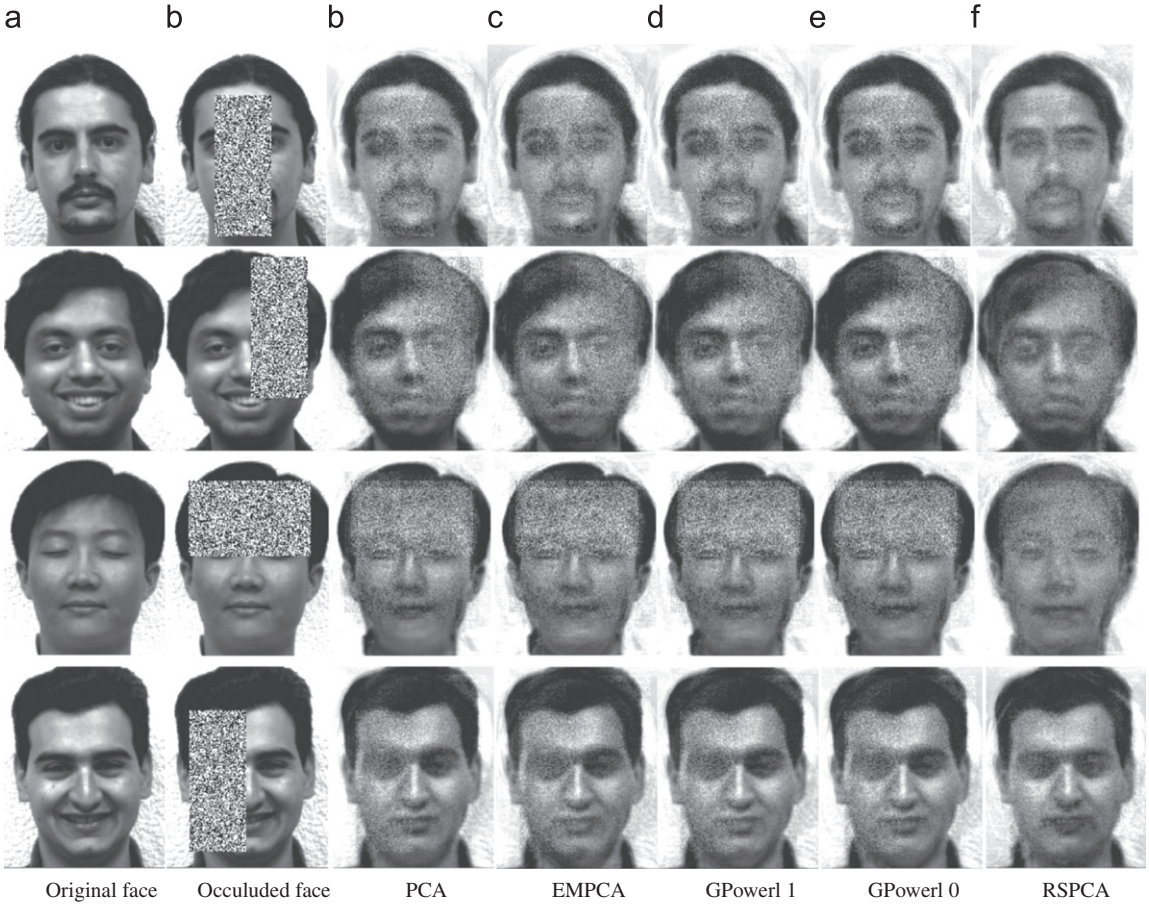

Fig. 5. The original face images, the corresponding images with occlusion, and the faces reconstructed by the classical PCA, EMPCA, GPower$_{l_1}$, GPower$_{l_0}$, and RSPCA methods with 20 corresponding projection PCs, respectively.

Fig. 6. Face images trained with dummy images and the faces reconstructed by the classical PCA, EMPCA, GPower$_{l_1}$, GPower$_{l_0}$, and RSPCA methods with 30 corresponding projection PCs, respectively.

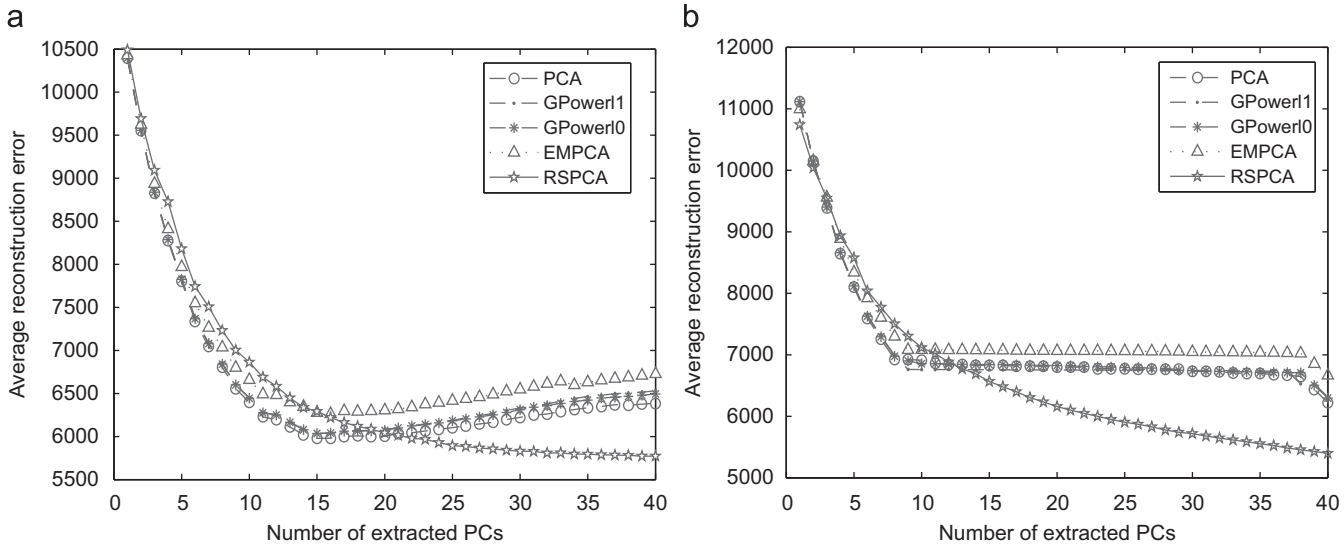

Fig. 7. Tendency curves of the average reconstruction errors of the classical PCA, EMPCA, GPower$_{l_1}$, GPower$_{l_0}$, and RSPCA methods w.r.t. the different extracted PC numbers for the Yale data set. (a) Occluded case; (b) dummy case.

For dummy case, the similar phenomenon as the occluded case is observed from Fig. 7(b): as m is small, the ARCE value of the RSPCA is a little larger than those of the other four methods, while from around $m=10$, the better reconstruction capability of the RSPCA becomes apparent. This can be further substantiated by Fig. 6: by projecting the images to the subspace spanned by the

Table 4
Performance comparison of the classical PCA, EMPCA, GPower$_{l_1}$, GPower$_{l_0}$, and RSPCA methods by applying them to the occluded face database. The ARCE and computation time of each method are evaluated and recorded at $m=40$.

Methods	Average sparsity	ARCE	Computation time	IT (the first three PCs)
PCA	36,765	6384.9971	0.417860	0
EMPCA	18,000	6726.0090	1415.837026	110+30+32=172
GPower$_{l_1}$	22,836.125	6531.1172	58.448949	24+10+19=53
GPower$_{l_0}$	22,781.85	6494.5514	56.339795	15+11+22=48
RSPCA	18,000	5829.2719	34.795060	10+17+12=39

Table 5
Performance comparison of the classical PCA, EMPCA, GPower$_{l_1}$, GPower$_{l_0}$, and RSPCA methods by applying them to the dummy face database. The ARCE and computation time of each method are evaluated and recorded at $m=40$.

Methods	Average sparsity	ARCE	Computation time	IT (the first three PCs)
PCA	36,765	6223.0231	0.488258	0
EMPCA	18,000	6665.7251	7114.826249	52+38+47=137
GPower$_{l_1}$	24,961.775	6315.7237	82.327144	16+14+18=48
GPower$_{l_0}$	24,946.675	6295.4676	60.607400	14+12+16=42
RSPCA	18,000	5507.8358	39.602456	9+7+5=21

first 30 PCs calculated by the employed methods, it is clear that the RSPCA best reconstructs the original images.

Furthermore, from Tables 4 and 5, it is easy to see that in both cases, by extracting sparser PCs, RSPCA achieves smaller ARCE values and spends less computational cost (naturally conducted by less iteration steps) than the other utilized sparse PCA methods. These results further demonstrate the robustness and efficiency of the proposed RSPCA method.

5. Conclusion

In this paper we have proposed a new sparse PCA method, called RSPCA method, to enforce robustness of the sparse PCA calculation. The most distinguished characteristic of the new method is that it intends to find the PC directions of the feature space where the L_1 dispersion, instead of the L_2-norm variance generally employed by the current sparse PCA methods, of the input data can be maximally captured. A simple algorithm to implement the RSPCA has also been developed, which has been proved to be able to converge to a local optimum of the problem. The robustness of the proposed method to outliers and noises has been supported by a series of experiments performed on the synthetic and face reconstruction problems. The efficiency of the method has also been theoretically and empirically substantiated.

There are, however, limitations of the proposed method. For example, the proposed RSPCA method only attains an approximation while not the rigorous solution to the original optimization problem (4), as aforementioned in Section 3.2. Endeavors still need to be made to design an effective and efficient algorithm to get the exact solution of (4) and hence to further enhance the performance of the robust sparse PCA. Besides, in our future research, the proposed method will be further evaluated by more practical applications.

Acknowledgments

This research was supported by the China NSFC project under Contract 60905003 and the National Grand Fundamental Research 973 Program of China under Grant no. 2007CB311002.

Appendix A. Supplementary data

Supplementary data associated with this article can be found in the online version at doi:10.1016/j.patcog.2011.07.009.

References

[1] I. Jollife, Principal Component Analysis, Springer Verlag, New York, 1986.
[2] H. Zou, T. Hastie, R. Tibshirani, Sparse principal component analysis, Journal of Computational and Graphical Statistics 15 (2006) 265–286.
[3] A. d'Aspremont, L. El Ghaoui, M.I. Jordan, G.R. Lanckriet, A direct formulation for sparse PCA using semidefinite programming, in: Advances in Neural Information Processing Systems, vol. 17, MIT Press, Cambridge, MA, 2005.
[4] M. Journée, Y. Nesterov, P. Richtárik, R. Sepulchre, Generalized power method for sparse principal component analysis, Journal of Machine Learning Research 11 (2010) 451–487.
[5] C.D. Sigg, J.M. Buhmann, Expectation maximization for sparse and non-negative PCA, in: Proceedings of the 25th International Conference on Machine Learning, ACM, Helsinki, Finland, 2008, pp. 960–967.
[6] Z.S. Lu, Y. Zhang, An augmented lagrangian approach for sparse principal component analysis. Mathematical Programming, 2009, doi:10.1007/s10107-011-0452-4.
[7] B. Moghaddam, Y. Weiss, S. Avidan, Spectral bounds for sparse PCA: exact and greedy algorithms, in: Advances in Neural Information Processing Systems, vol. 18, MIT Press, Cambridge, MA, 2006, pp. 915–922.
[8] A. d'Aspremont, F.R. Bach, L.E. Ghaoui, Optimal solutions for sparse principal component analysis, Journal of Machine Learning Research 9 (2008) 1269–1294.
[9] I.T. Jollife, M. Uddin, A modified principal component technique based on the lasso, Journal of Computational and Graphical Statistics 12 (2003) 531–547.
[10] I. Jollife, Rotation of principal components: choice of normalization constraints, Journal of Applied Statistics 22 (1995) 29–35.
[11] J. Cadima, I. Jollife, Loadings and correlations in the interpretation of principal components, Journal of Applied Statistics 22 (1995) 203–214.
[12] H.P. Shen, J.Z. Huang, Sparse principal component analysis via regularized low rank matrix approximation, Journal of Multivariate Analysis 99 (2008) 1015–1034.
[13] B.K. Sriperumbudur, D.A. Torres, G.R.G. Lanckriet, Sparse eigen methods by D.C. programming, in: Proceedings of the 24th International Conference on Machine learning, Corvallis, OR, USA, 2007, pp. 831–838.
[14] R. Zass, A. Shashua, Nonnegative sparse PCA, in: Advances in Neural Information Processing Systems, vol. 19, MIT Press, Cambridge, MA, 2007, pp. 1561–1568.
[15] N. Kwak, Principal component analysis based on L1-norm maximization, IEEE Transactions on Pattern Analysis and Machine Intelligence 30 (2008) 1672–1680.
[16] F.D. la Torre, M.J. Black, A framework for robust subspace learning, International Journal of Computer Vision 54 (1–3) (2003) 117–142.
[17] H. Aanas, R. Fisker, K. Astrom, J. Carstensen, Robust factorization, IEEE Transactions on Pattern Analysis and Machine Intelligence 24 (2002) 1215–1225.
[18] C. Ding, D. Zhou, X. He, H. Zha, R1-PCA: rotational invariant L1-norm principal component analysis for robust subspace factorization, in: Proceedings of the 23rd International Conference on Machine Learning, ACM, Pittsburgh, PA, 2006, pp. 281–288.
[19] A. Baccini, P. Besse, A.D. Falguerolles, A L1-norm PCA and a heuristic approach, Ordinal and Symbolic Data Analysis (1996) 359–368.
[20] Q. Ke, T. Kanade, Robust L1 norm factorization in the presence of outliers and missing data by alternative convex programming, in: Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, IEEE, San Diego, USA, 2005, pp. 739–746.
[21] P.N. Belhumeur, J.P. Hespanha, D.J. Kriegman, Eigenfaces versus Fisherfaces: recognition using class specific linear projection, IEEE Transactions on Pattern Analysis and Machine Intelligence 19 (1997) 711–720.
[22] J. Wright, A. Yang, A. Ganesh, S. Sastry, Y. Ma, Robust face recognition via sparse representation, IEEE Transactions on Pattern Analysis and Machine Intelligence 30 (2009) 210–217.
[23] E. Candes, X.D. Li, Y. Ma, J. Wright, Robust principal component analysis? Journal of the ACM 58(3) (2011), doi:10.1145/1970392.1970395.

Deyu Meng received his B.Sc., M.Sc., and Ph.D. degrees in 2001, 2004, and 2008, respectively, from Xi'an Jiaotong University, Xi'an, China. He is currently a lecturer with the Institute for Information and System Sciences, Faculty of Science, Xi'an Jiaotong University. His current research interests include principal component analysis, nonlinear dimensionality reduction, feature extraction and selection, compressed sensing, and sparse machine learning methods.

Qian Zhao received his B.Sc. in 2009 from Xi'an Jiaotong University, Xi'an, China, where he is currently working toward the M.Sc. degree. His current research interests include feature extraction and selection, dimensionality reduction, and compressed sensing.

Zongben Xu received his M.Sc. degree in mathematics and the Ph.D. degree in applied mathematics from Xi'an Jiaotong University, Xi'an, China, in 1981 and 1987, respectively. In 1988, he was a Postdoctoral Researcher with the Department of Mathematics, The University of Strathclyde, Glasgow, UK. He was a research fellow with the Information Engineering Department from February 1992 to March 1994, the center for environmental studies from April 1995 to August 1995, and the mechanical engineering and automation department from September 1996 to October 1996, The Chinese University of Hong Kong, Shatin, Hong Kong. From January 1995 to April 1995, he was a research fellow with the Department of Computing, The Hong Kong Polytechnic University, Kowloon, Hong Kong. He is currently a professor with the Institute for Information and System Sciences, Faculty of Science, Xi'an Jiaotong University. His current research interests include manifold learning, neural networks, evolutionary computation, and multiple-objective decision-making theory.