Robust Multilinear Principal Component Analysis

Kohei Inoue, Kenji Hara, and Kiichi Urahama
Department of Visual Communication Design, Kyushu University
4-9-1, Shiobaru, Minami-ku, Fukuoka, 815-8540, Japan
{k-inoue,hara,urahama}@design.kyushu-u.ac.jp

Abstract

We propose two methods for robustifying multilinear principal component analysis (MPCA) which is an extension of the conventional PCA for reducing the dimensions of vectors to higher-order tensors. For two kinds of outliers, i.e., sample outliers and intra-sample outliers, we derive iterative algorithms on the basis of the Lagrange multipliers. We also demonstrate that the proposed methods outperform the original MPCA when datasets contain such outliers experimentally.

1. Introduction

Dimensionality reduction is of fundamental importance in image processing, pattern recognition and computer vision (CV) as it is necessary to avoid the curse of dimensionality [8]. Principal component analysis (PCA) [12] is one of the most fundamental and well-known techniques for dimensionality reduction. There is a broad community in CV and image analysis that makes use of PCA and its extensions, which includes many application areas, e.g., shape analysis [18], shape statistics [4], segmentation using deformable models [17] and handwritten digit classification [16].

Xu and Yuille [20] proposed self-organizing rules for robust PCA. Higuchi and Eguchi [11] generalized their approach [20] and adaptively selected a set of tuning parameters which control the degree of robustness. De la Torre and Black [5, 6] pointed out that the previous robust approaches to PCA are of limited application in CV problems as they reject entire images as outliers, i.e., they are robust to sample outliers, and proposed another robust PCA which is robust to intra-sample outliers, e.g., outlier pixels in an image sample. Although these methods achieve improved tolerance to outliers compared with the conventional PCA, it might be difficult to apply them to higher-order tensors because they are formulated for reducing the dimensions of vectors. Therefore, tensors have to be transformed into vectors in advance. However, this transformation might cause a high-dimensional situation in which the above robust PCA approaches fail.

In order to address the problem of high-dimensionality caused by the transformation of tensors into vectors, Yang et al. [21] proposed two-dimensional PCA (2DPCA) for reducing the dimensions of matrices rather than vectors, i.e., matrices do not need to be preliminarily transformed into vectors. However, the 2DPCA is approximately equivalent to the conventional PCA operated only on the row vectors of matrices [19, 9, 25]. Ye et al. [23] proposed generalized PCA (GPCA) which reduces the dimensions of both rows and columns of matrices, i.e., GPCA is a generalization of 2DPCA [21]. Cai et al. [2] proposed Tensor PCA and Tensor LDA; the former is essentially equivalent to GPCA [23]. Ye [22] also proposed generalized low rank approximations of matrices (GLRAM) which is highly similar to GPCA [23]. While GPCA [23] assumes that data are of zero mean, GLRAM [22] does not. Recently, Lu et al. [13] proposed multilinear PCA (MPCA) which reduces the dimensions of all modes of higher-order tensors, i.e., MPCA is a generalization of GPCA [23] and 2DPCA [21]. Although these extensions of PCA for vectors to that for higher-order tensors resolve the problem resulting from the high-dimensionality of vectors, their tolerance to outliers is limited because they are formulated as Frobenius norm minimization problems.

This work is motivated by the above observation of the state-of-the-art PCA techniques. That is, robustifying MPCA [13] will improve the tolerance of MPCA to outliers and avoid the problem of high-dimensionality simultaneously. Since the conventional PCA and its extensions including the above techniques are widely used in CV and other areas, the robust version of MPCA will increase the applicability of the PCA techniques in various areas.

In this paper, we propose two methods for robustifying the MPCA [13] to sample outliers and intra-sample outliers. For both of them, we derive iterative algorithms on the basis of the Lagrange multipliers. We experimentally compare our robust MPCA (RMPCA) with MPCA [13] and show the effectiveness of the RMPCA.

The rest of this paper is organized as follows. Section
order tensor and a component analysis (MPCA) [13] and rewrite the objective formulated as follows:

\[
\Psi_A - \Psi_B = \sum_{m=1}^{M} \| \tilde{A}_m - \tilde{B}_m \|_F^2 - \sum_{m=1}^{M} \| \tilde{A}_m \times \{U^T\} \|_F^2
\]  

(6)

(7)

\[
= \sum_{m=1}^{M} \| \tilde{A}_m \|_F^2 - \sum_{m=1}^{M} \| \tilde{A}_m \times \{U^T\} \|_F^2
\]  

(8)

(9)

\[
= \sum_{m=1}^{M} \left( \| \tilde{A}_m \|_F^2 - 2 \| \tilde{A}_m \times \{U^T\} \|_F^2 \right)
\]  

\[
+ \| \tilde{A}_m \times \{U^T\} \|_F^2
\]  

(10)

(11)

\[
= \sum_{m=1}^{M} \| \tilde{A}_m - \tilde{B}_m \times \{U\} \|_F^2 \ 
\]  

(12)

where \( \tilde{A}_m = [\tilde{a}_{m,i_1 \ldots i_N}] = A_m - \bar{A} \) and \( \tilde{B}_m = [\tilde{b}_{m,j_1 \ldots j_N}] = B_m - \bar{B} \).

Unlike in the above Frobenius norm minimization formulation where the mean tensor \( \bar{A} \) is obtained analytically, in the following robust MPCA (RMPCA), \( \bar{A} \), i.e., the robust mean, must be estimated as well as \( \{\tilde{B}\} \) and \( U \).

3. Robust MPCA

The above MPCA is formulated as a minimization of the sum of the Frobenius norms. Therefore, the MPCA is not robust to outliers. In this section, we propose two methods for robustifying the MPCA to sample outliers and intra-sample outliers.

3.1. RMPCA for Sample Outliers

First, we modify the minimization of (12) as

\[
\min_{\bar{A}, \{\tilde{B}_m\}, U} \sum_{m=1}^{M} \rho \left( \| A_m - \bar{A} - \tilde{B}_m \times \{U\} \|_F \right),
\]  

(13)

where \( \rho(x) \) is a \( \rho \)-function for robust \( M \)-estimation. Among several \( M \)-estimators including Andrews, bisquare, Cauchy, and Huber estimators [10], we select the Welsch estimator for which the \( \rho \)-function is given by

\[
\rho(x) = 1 - e^{-\alpha x^2},
\]  

(14)

because it worked well in our experiments described later. Let \( F(\bar{A}, \{\tilde{B}_m\}, U) \) be the objective function in (13) with
where $M$ is a constant, we may rewrite (13) as
\[
\max_{\tilde{A}, \{\tilde{B}_m\}} F(\tilde{A}, \{\tilde{B}_m\}, U),
\]
where
\[
\tilde{F}(\tilde{A}, \{\tilde{B}_m\}) = M - F(\tilde{A}, \{\tilde{B}_m\}, U)
= \sum_{m=1}^{M} e^{-\alpha \|A_m - \tilde{A} - B_m \times \{U\}\|_F^2}.
\]

The Lagrange function for (16) with (4) is given by
\[
L(\tilde{A}, \{\tilde{B}_m\}, U, \{\Lambda^{(n)}\}) = \tilde{F}(\tilde{A}, \{\tilde{B}_m\}, U)
+ \alpha \sum_{n=1}^{N} \text{tr} \left[ \Lambda^{(n)} \left( \sum_{n=1}^{N} \text{tr} \left( U^{(n)} U^{(n)} - I_{J_n} \right) \right) \right],
\]
where $\text{tr}$ denotes the matrix trace and $\Lambda^{(n)} \in \mathbb{R}^{J_n \times J_n}$ for $n = 1, \ldots, N$ is a symmetric matrix of which the elements are the Lagrange multipliers. Then we have the following necessary conditions for optimality:
\[
\frac{\partial L}{\partial \tilde{A}} = 2\alpha \sum_{m=1}^{M} (C_m - \tilde{A}) e^{-\alpha \|C_m - \tilde{A}\|_F^2} = 0,
\]
where $C_m = A_m - \tilde{B}_m \times \{U\}$, and
\[
\frac{\partial L}{\partial \tilde{B}_m} = 2\alpha \left( \tilde{A}_m - \tilde{B}_m \times \{U\} \right) + \alpha \sum_{n=1}^{N} \text{tr} \left[ \Lambda^{(n)} \left( \sum_{n=1}^{N} \text{tr} \left( U^{(n)} U^{(n)} - I_{J_n} \right) \right) \right] = 0.
\]

From (20) and (21), we have
\[
\tilde{A} = \frac{\sum_{m=1}^{M} C_m e^{-\alpha \|C_m - \tilde{A}\|_F^2}}{\sum_{m=1}^{M} e^{-\alpha \|C_m - \tilde{A}\|_F^2}}
\]
and
\[
\tilde{B}_m = \tilde{A}_m \times \{U^{(1)}\},
\]
respectively. From (22), we have
\[
U^{(n)} = P^{(n)} \left( Q^{(n)} - \Lambda^{(n)} \right)^{-1},
\]
where
\[
P^{(n)} = \sum_{m=1}^{M} \tilde{A}_m \left( \tilde{B}_m \right)^{T} e^{-\alpha \|A_m - \tilde{B}_m \times \{U\}\|_F^2},
\]
\[
Q^{(n)} = \sum_{m=1}^{M} \tilde{B}_m \left( \tilde{B}_m \right)^{T} e^{-\alpha \|A_m - \tilde{B}_m \times \{U\}\|_F^2}.
\]

By substituting (26) into (23) we find that
\[
P^{(n)}^{T} P^{(n)} = \left( Q^{(n)} - \Lambda^{(n)} \right)^{T} \left( Q^{(n)} - \Lambda^{(n)} \right).
\]

Let
\[
P^{(n)}^{T} P^{(n)} = V^{(n)} \Sigma^{(n)} V^{(n)^{T}}
\]
be a spectral decomposition of $P^{(n)}^{T} P^{(n)}$, where $\Sigma^{(n)}$ is a diagonal matrix of which the diagonal elements are the eigenvalues of $P^{(n)}^{T} P^{(n)}$ and $V^{(n)}$ is an orthogonal matrix of which the columns are the corresponding eigenvectors of $P^{(n)}^{T} P^{(n)}$. Then it follows from (29) and (30) that
\[
\Lambda^{(n)} = Q^{(n)} - \Sigma^{(n)} 1/2 V^{(n)^{T}}.
\]

Substituting this into (26), we have
\[
U^{(n)} = P^{(n)} V^{(n)} \Sigma^{(n)}^{-1/2}.
\]

Consequently, we obtain the following algorithm:

**Algorithm 1.**

**Step 0:** Initialize $\tilde{A}$ and $U$ as $\tilde{A}^{(0)} = \frac{1}{M} \sum_{m=1}^{M} A_m$ and $U^{(0)} = \{U^{(1,0)}, \ldots, U^{(N,0)}\}$, where $U^{(n,0)}$ is an orthonormal matrix of which the columns are the eigenvectors of $\sum_{m=1}^{M} \tilde{A}_m \left( \tilde{B}_m \right)^{T}$ corresponding to the largest $J_n$ eigenvalues, where $\tilde{A}_m$ is the model matricizing $\{U^{(n)}\}$ of $\tilde{A}_m^0 = A_m - \tilde{A}^{(0)}$. Then initialize $\tilde{B}_m$ as $\tilde{B}_m^{(0)} = \tilde{A}_m^0 \times \{U^{(1)}\}$ for $m = 1, \ldots, M$. Initialize the number of iterations as $t = 0$.

**Step 1:** Compute $U^{(t+1)} = \{U^{(1,t+1)}, \ldots, U^{(N,t+1)}\}$, where $U^{(n,t+1)}$ is given by (32) into which $\tilde{A}^{(t)}$, $U^{(t)}$ and $\{\tilde{B}_m^{(t)}\}$ are substituted.
Step 2: For \( m = 1, \ldots, M \), compute \( \tilde{B}^{(t+1)}_m \) by (25) into which \( \tilde{A}^{(t)} \) and \( U^{(t+1)} \) are substituted.

Step 3: Compute \( \tilde{A}^{(t+1)} \) by (24) into which \( \tilde{A}^{(t)} \), \( \{\tilde{B}^{(t+1)}_m\} \) and \( U^{(t+1)} \) are substituted.

Step 4: If, for \( \epsilon > 0 \), \( |\tilde{F}(\tilde{A}^{(t+1)}, \{\tilde{B}^{(t+1)}_m\}, U^{(t+1)}) - \tilde{F}(\tilde{A}^{(t)}, \{\tilde{B}^{(t)}_m\}, U^{(t)})|/M < \epsilon \), then stop, otherwise increase \( t \) by 1 and go to Step 1.

3.2. RMPCA for Intra-Sample Outliers

Next, we modify the minimization of (12) as

\[
\min_{\tilde{A}, \{\tilde{B}_m\}} \sum_{m=1}^{M} \sum_{i_1=1}^{I_1} \cdots \sum_{i_N=1}^{I_N} \rho(d_{m,i_1 \cdots i_N}),
\]

where \( d_{m,i_1 \cdots i_N} = a_{m,i_1 \cdots i_N} - \tilde{a}_{m,i_1 \cdots i_N} - \sum_{j_1=1}^{J_1} \cdots \sum_{j_N=1}^{J_N} b_{mj_1 \cdots j_N} \). Let \( G(\tilde{A}, \{\tilde{B}_m\}, U) \) be the objective function in (33) with (14). Then we have

\[
G(\tilde{A}, \{\tilde{B}_m\}, U) = \Theta - \sum_{m=1}^{M} \sum_{i_1=1}^{I_1} \cdots \sum_{i_N=1}^{I_N} e^{-\alpha d_{m,i_1 \cdots i_N}^2},
\]

where \( \Theta = M \prod_{n=1}^{N} I_n \). Since \( \Theta \) is a constant, we may rewrite (33) as

\[
\max_{\tilde{A}, \{\tilde{B}_m\}} \tilde{G}(\tilde{A}, \{\tilde{B}_m\}, U),
\]

where

\[
\tilde{G}(\tilde{A}, \{\tilde{B}_m\}, U) = G(\tilde{A}, \{\tilde{B}_m\}, U)
\]

\[
- \sum_{m=1}^{M} I_1 \cdots I_N \sum_{i_N=1}^{I_N} e^{-\alpha d_{m,i_1 \cdots i_N}^2}.
\]

The Lagrange function for (35) with (4) is given by

\[
\tilde{L}(\tilde{A}, \{\tilde{B}_m\}, U, \{\tilde{\Lambda}_n\}) = \tilde{G}(\tilde{A}, \{\tilde{B}_m\}, U) + \alpha \sum_{n=1}^{N} \text{tr}\left[ \tilde{\Lambda}_n \left( U^{(n)} U^{(n)T} - I_{J_n} \right) \right],
\]

where \( \tilde{\Lambda}_n \in \mathbb{R}^{J_n \times J_n} \) for \( n = 1, \ldots, N \) is a symmetric matrix of which the elements are the Lagrange multipliers. Then we have the following necessary conditions for optimality:

\[
\frac{\partial \tilde{L}}{\partial \tilde{A}} = 2\alpha \sum_{m=1}^{M} (C_m - \tilde{A}) \odot E_m = 0,
\]

where \( E_m = [e^{-\alpha d_{m,x_1 \cdots x_N}^2}] \in \mathbb{R}^{I_1 \times \cdots \times I_N} \) for \( m = 1, \ldots, M \) and \( \odot \) denotes the Hadamard product or element-wise product [14] of tensors, and

\[
\frac{\partial \tilde{L}}{\partial \tilde{B}_m} = 2\alpha \left( \tilde{A}_m - \tilde{B}_m \times \{U\} \odot E_m \right) \times \{U^T\} = 0
\]

for \( m = 1, \ldots, M \), and

\[
\frac{1}{2\alpha} \frac{\partial \tilde{L}}{\partial U^{(n)}} = \sum_{m=1}^{M} \left( \left( \tilde{A}_m - \tilde{B}_m \times \{U\} \odot E_m, \tilde{B}_m^{(-n)} \right)_{\{K;K\}} + U^{(n)} \tilde{\Lambda}^{(n)} \right) = 0
\]

for \( n = 1, \ldots, N \).

From (39) and (40), we have

\[
\tilde{A} = \left( \sum_{m=1}^{M} C_m \odot E_m \right) \odot \mathcal{H},
\]

where \( \mathcal{H} = [1/\sum_{m=1}^{M} e^{-\alpha d_{m,i_1 \cdots i_N}^2}] \in \mathbb{R}^{I_1 \times \cdots \times I_N} \), and (25), respectively. From (41), we have

\[
U^{(n)} = \tilde{P}^{(n)} \left( \tilde{Q}^{(n)} - \tilde{\Lambda}^{(n)} \right)^{-1},
\]

where

\[
\tilde{P}^{(n)} = \sum_{m=1}^{M} \left( \tilde{A}_m \odot E_m, \tilde{B}_m^{(-n)} \right)_{\{K;K\}},
\]

\[
\tilde{Q}^{(n)} = \sum_{m=1}^{M} \left( \tilde{B}_m^{(-n)} \odot \tilde{B}_m^{(-n)}, E_m \right)_{\{K;K\}}.
\]

Since (44) is analogous to (26), we can derive the following equation in the same manner as the derivation of (32):

\[
U^{(n)} = \tilde{P}^{(n)} \tilde{V}^{(n)} \left( \tilde{S}^{(n)} \right)^{-1/2},
\]

where \( \tilde{S}^{(n)} \) is a diagonal matrix of which the diagonal elements are the eigenvalues of \( (\tilde{P}^{(n)})^T \tilde{P}^{(n)} \) and \( \tilde{V}^{(n)} \) is an orthogonal matrix of which the columns are the corresponding eigenvectors of \( (\tilde{P}^{(n)})^T \tilde{P}^{(n)} \).

Consequently, we obtain the following algorithm:

**Algorithm 2.**
Step 0: Initialize $\bar{A}$ and $U$ as $\bar{A}^{(0)} = \frac{1}{M} \sum_{m=1}^{M} A_m$ and $U^{(0)} = \{U^{(1,0)}, \ldots, U^{(N,0)}\}$, where $U^{(n,0)}$ is an orthogonal matrix of which the columns are the eigenvectors of $\sum_{m=1}^{M} \bar{A}_{m(n)} (\bar{A}_{m(n)})^T$ corresponding to the largest $J_n$ eigenvalues, where $\bar{A}_{m(n)}$ is the mode-$n$ matricizing of $A_m = \bar{A}^{(0)}$. Then initialize $\bar{B}_m$ as $\bar{B}^{(0)} = \bar{A}^{(0)} \times \{U^{(0)}\}^T$ for $m = 1, \ldots, M$. Initialize the number of iterations as $t = 0$.

Step 1: Compute $U^{(t+1)} = \{U^{(1,t+1)}, \ldots, U^{(N,t+1)}\}$, where $U^{(n,t+1)}$ is given by (47) into which $\bar{A}^{(t)}$, $U^{(t)}$ and $\{\bar{B}_m^{(t)}\}$ are substituted.

Step 2: For $m = 1, \ldots, M$, compute $\bar{B}_m^{(t+1)}$ by (25) into which $\bar{A}^{(0)}$ and $U^{(t+1)}$ are substituted.

Step 3: Compute $\bar{A}^{(t+1)}$ by (43) into which $\bar{A}^{(t)}$, $\{\bar{B}_m^{(t+1)}\}$ and $U^{(t+1)}$ are substituted.

Step 4: If, for $\epsilon > 0$, $|G(\bar{A}^{(t+1)}, \{\bar{B}_m^{(t+1)}\}, U^{(t+1)}) - G(\bar{A}^{(t)}, \{\bar{B}_m^{(t)}\}, U^{(t)})|/\Theta < \epsilon$, then stop, otherwise increase $t$ by 1 and go to Step 1.

4. Experimental Results

In this section, we experimentally evaluate the performance of the proposed algorithms on the ORL face image database [15]. The ORL database [15] contains face images of 40 persons. For each person, there are 10 different face images. That is, the total number of the images in the database is 400. The size of each image is 112 × 92 pixels, i.e., $I_1 = 112$, $I_2 = 92$ and $N = 2$. An example of face images in the database is shown in Fig. 1, where five images of a person are arranged along with an example of sample outliers (the rightmost image). Reconstructed images with MPCA and RMPCA for sample outliers are shown in Fig. 2. Five reconstructed images from outlier-free data are shown at the top row in Fig. 2(a), i.e., the clean reconstruction, where the rightmost image is the mean image of 10 face images of the person shown in Fig. 1. The corresponding zoomed parts of the forehead of the person are aligned at the bottom row in Fig. 2(a). Reconstructed images with MPCA and RMPCA for data including sample outliers are shown in Figs. 2 (b) and (c), respectively. In Fig. 2(b), the left five reconstructed images and the rightmost mean image are distorted by the sample outlier. On the other hand, in Fig. 2(c), the left five reconstructed images are less sensitive to the outlier and the rightmost robust mean image is close to the outlier-free mean image in Fig. 2(a). In this example, we set $J_1 = J_2 = 30$ for both MPCA and RMPCA, and $\alpha = 10^{-6}$ and $\epsilon = 10^{-6}$ for RMPCA. Although we select the value of $\alpha$ manually, some self-tuning algorithms [3, 24] might work well. Let $M$ be the number of sample outliers per person. Then we set $M = 10 + M$ for each person. Reconstructed images are computed as $A_m = \bar{B}_m \times \{U\} + \bar{A}$ for $m = 1, \ldots, 10$. Sample outliers are numbered from 11 to $M$. Variation in the objective function $\tilde{F}$ until the convergence is shown in Fig. 3, in which the horizontal axis denotes the number of iterations and the vertical axis denotes the value of $\tilde{F}$ at each iteration. Reconstruction errors calculated with all face images in the ORL database [15] are shown in Fig. 4, in which we evaluated the errors by the root mean squared error (RMSE) defined as

$$\text{RMSE} = \sqrt{\frac{1}{M} \sum_{m=1}^{M} \|A_m - \hat{A}_m\|^2_F}. \quad (48)$$

In this figure, the horizontal axis denotes the number of
sample outliers per person and the vertical axis denotes the RMSE. MPCA and RMPCA are denoted by solid and broken lines, respectively. The RMSE for RMPCA is lower than that for MPCA. Although the RMSE for MPCA increases with the number of sample outliers, that for RMPCA is almost constant.

An example of face images including intra-sample outliers is shown in Fig. 5. In each image, 2% of pixels are outliers. In this experiment, we use the MATLAB function ‘imnoise’ with ‘salt & pepper’ option to generate intra-sample outliers. Reconstructed images with MPCA and RMPCA for intra-sample outliers are shown in Figs. 6 (a) and (b), respectively. At the top row in Fig. 6(a), the left five reconstructed images and the rightmost mean image are disturbed by the intra-sample outliers. The corresponding zoomed parts are aligned at the bottom row. On the other hand, in Fig. 6(b), the left five reconstructed images and the rightmost robust mean image are less sensitive to the outliers. In this example, we set $\alpha = 10^{-3}$ and $M = 10$. Variation in the objective function $\tilde{G}$ until convergence is shown in Fig. 7, in which the horizontal axis denotes the number of iterations and the vertical axis denotes the value of $\tilde{G}$. Reconstruction errors evaluated by the RMSE in (48) are shown in Fig. 8, in which the horizontal axis denotes the ratio of intra-sample outliers in all pixels and the vertical axis denotes the RMSE. In this figure, MPCA and RMPCA are denoted by solid and broken lines, respectively. The RMSE for RMPCA is lower than that for MPCA. In contrast to the RMSE for RMPCA in Fig. 4, which is almost constant, that in Fig. 8 increases with the ratio of intra-sample outliers. This result demonstrates the difficulty in detecting and removing the intra-sample outliers compared to the sample outliers.

5. Conclusion

We have proposed two robust methods for multilinear principal component analysis (MPCA): one for sample out-
liers and another for intra-sample outliers. For both of them, we derived iterative algorithms on the basis of the Lagrange multipliers. Experimental results show that the proposed methods outperform the original MPCA when datasets contain outliers. Future work will include the robustification of multilinear discriminant analysis.

Acknowledgments

This work was partially supported by the Ministry of Education, Culture, Sports, Science and Technology under the Grant-in-Aid for Scientific Research (20700165).

References