Robust Principal Component Analysis with Complex Noise

Qian Zhao†
Deyu Meng†
Zongben Xu†
Wangmeng Zuo§
Lei Zhang‡

†School of Mathematics and Statistics, Xi’an Jiaotong University, Xi’an, China
§School of Computer Science and Technology, Harbin Institute of Technology, Harbin, China
‡Department of Computing, Hong Kong Polytechnic University, Hong Kong, China

Abstract

The research on robust principal component analysis (RPCA) has been attracting much attention recently. The original RPCA model assumes sparse noise, and use the $L_1$-norm to characterize the error term. In practice, however, the noise is much more complex and it is not appropriate to simply use a certain $L_p$-norm for noise modeling. We propose a generative RPCA model under the Bayesian framework by modeling data noise as a mixture of Gaussians (MoG). The MoG is a universal approximator to continuous distributions and thus our model is able to fit a wide range of noises such as Laplacian, Gaussian, sparse noises and any combinations of them. A variational Bayes algorithm is presented to infer the posterior of the proposed model. All involved parameters can be recursively updated in closed form. The advantage of our method is demonstrated by extensive experiments on synthetic data, face modeling and background subtraction.

1. Introduction

As a classical and popular tool for data analysis, principal component analysis (PCA) has a wide range of applications in science and engineering (Jolliffe, 2002). Essentially, PCA seeks the best $L_2$-norm low-rank approximation of the given data matrix. However, $L_2$-norm is sensitive to gross noises and outliers, which are often introduced in data acquisition. Therefore, how to make PCA robust has been attracting much attention in the last decade (De la Torre & Black, 2003; Ke & Kanade, 2005; Ding et al., 2006; Kwak, 2008).

Motivated by the recent advances in low-rank matrix analysis (Candès & Recht, 2009; Candès & Tao, 2010; Recht et al., 2010), the so-called robust principal component analysis (RPCA) (Wright et al., 2009) has been proposed to decompose a given data matrix into a low-rank matrix and a sparse matrix. Denote by $Y \in \mathbb{R}^{m \times n}$ the original data matrix, by $L \in \mathbb{R}^{m \times n}$ the low-rank component and by $E \in \mathbb{R}^{m \times n}$ the sparse component, RPCA can be mathematically described as the following convex optimization problem:

$$\min_{L,E} \|L\|_* + \lambda \|E\|_1 \quad s.t. \quad Y = L + E,$$

(1)

where $\|L\|_* = \sum_r \sigma_r(L)$ denotes the nuclear norm of $L$, $\sigma_r(L)$ ($r = 1, 2, \ldots, \min(m, n)$) is the $r$th singular value of $L$, $\|E\|_1 = \sum_{ij} |e_{ij}|$ denotes the $L_1$-norm of $E$ and $e_{ij}$ is the element in the $i^{th}$ row and $j^{th}$ column of $E$. Under certain noise sparsity and rank upper-bound assumptions, it has been proved that one can exactly recover $L$ and $E$ from $Y$ with high probability (Candès et al., 2011). RPCA has been successfully applied to many machine learning and computer vision problems, such as video surveillance (Wright et al., 2009), face modeling (Peng et al., 2010) and subspace clustering (Liu et al., 2010).

RPCA, however, still has clear limitations. As shown in Eq. (1), it utilizes $L_1$-norm to characterize $E$, which is only optimal for Laplacian noise. Although $L_1$-norm can better fit sparse noise than $L_2$-norm, the real noise is often neither Gaussian nor Laplacian, but has much more complex statistical structures. For example, the Bootstrap sequence (Li et al., 2004) shown in Figure 1 can be reasonably modeled as the sum of a low-rank part (background) and a noise part (fore-
background). However, such noise has a rather complex structure. As can be seen from Figure 1, the noise in the Bootstrap sequence can be decomposed into: moving people in the foreground, shadows alongside people, and background noise. Clearly, in this real scenario, it is not appropriate to simply use $L_1$- or $L_2$-norm to model the noise, which assumes Laplacian or Gaussian noise, respectively.

This paper presents a new RPCA approach, which can fit more complex noise. We formulate the problem as a generative model under the Bayesian framework, and model data noise as a mixture of Gaussians (MoG). We then employ the variational inference method to infer the posterior. Since MoG is a universal approximator to any continuous probability distribution (Bishop, 2006; Meng & De la Torre, 2013), the proposed MoG-RPCA approach is capable of adapting a much wider range of real noises than the current RPCA methods.

2. Related Work

Early attempts to solve the RPCA problem replace the $L_2$-norm error by some robust losses. De la Torre & Black (2003) utilized the Geman-McClure function in robust statistics to improve the robustness of PCA; Ding et al. (2006) used a smoothed $R_1$-norm to this end; Kwak (2008) introduced the $L_1$-norm variance and designed an efficient algorithm to optimize it. These methods, however, are sensitive to initialization, and only perform well on Laplacian-like noise.

In recent years, low-rank matrix analysis methods have been rapidly developed. Wright et al. (2009) initially formulated the RPCA model as shown in Eq. (1). Some variants have also been proposed, e.g., Xu et al. (2010) used the $L_{1,2}$-norm to handle data corrupted by column. The iterative thresholding method (Candès et al., 2011) was proposed to solve the RPCA model. This method, however, converges very slow. To speed up the computation, Lin et al. proposed the accelerated proximal gradient (APG) (Lin et al., 2009) and the augmented Lagrangian multiplier (ALM) (Lin et al., 2010) methods. ALM leads to state-of-the-art performance in terms of both speed and accuracy.

Bayesian approaches to RPCA have also been investigated. Ding et al. (2011) modeled the singular values of $L$ and the entries of $E$ with beta-Bernoulli priors, and used a Markov chain Monte Carlo (MCMC) sampling scheme to perform inference. This method needs many sampling iterations, always hampering its practical use. Babacan et al. (2012) adopted the automatic relevance determination (ARD) approach to model both $L$ and $E$, and utilized the variational Bayes (VB) method to do inference. This method is more computationally efficient. These Bayesian methods, however, assume a certain noise prior (a sparse noise plus a dense noise), which cannot always effectively model the diverse types of noises occurring in practice.

One problem closely related to RPCA is $L_1$-norm low-rank matrix factorization (LRMF), which aims to factorize a matrix into the product of two smaller matrices under the $L_1$ measure. Ke & Kanade (2005) proposed to solve it via alternative linear/quadratic programming (ALP/AQP). Eriksson & van den Hengel (2010) designed an $L_1$-Wiberg approach by extending the classical Wiberg method to $L_1$ minimization. Zheng et al. (2012) proposed a RegL1ALM method by using convex trace-norm regularization to improve convergence. Wang et al. (2012) considered the problem in a probabilistic framework, and solved it via conditional EM algorithm. Meng et al. (2013) proposed a novel cyclic weighted median method to efficiently solve the problem. The $L_1$-norm LRMF problem can be viewed as a fixed-rank variant of RPCA. However, the $L_1$-norm LRMF model is not convex and it can be trapped to local optima. Moreover, the rank needs to be pre-specified in this line of research, which is often unavailable in practice.

3. RPCA with MoG Noise

3.1. Model Formulation

Let’s consider RPCA as a generative model:

$$Y = L + E.$$  (2)

If we assume that the entries of $E$ are drawn independently from a Laplacian distribution, and the singular values of $L$ are drawn from another Laplacian distribution, we can obtain the RPCA model (1) by performing maximum a posteriori (MAP) estimation on $L$. Clearly, we can interpret RPCA as a MAP estimation problem with Laplacian noise. However, real noises are more complicated. To improve RPCA, a
natural idea is to use MoG to model noise since MoG is a universal approximator to any continuous distributions (Bishop, 2006). For example, a Gaussian is a special case of MoG and a Laplacian can be expressed as a scaled MoG (Andrews & Mallows, 1974). Similar noise modeling strategy was also adopted by Meng & De la Torre (2013) for LRMF problem.

Noise Component Modeling. We assume that each $e_{ij}$ in $E$ follows a MoG distribution:

$$e_{ij} \sim \sum_{k=1}^{K} \pi_k N(e_{ij} | \mu_k, \tau_k^{-1}), \quad (3)$$

where $\pi_k$ is the mixing proportion with $\pi_k \geq 0$ and $\sum_{k=1}^{K} \pi_k = 1$, $K$ is the Gaussian components number and $N(e | \mu, \tau^{-1})$ denotes the Gaussian distribution with mean $\mu$ and precision $\tau$. Eq. (3) can be equivalently expressed as a two-level generative model by introducing the indicator variables $z_{ijk}$ (Bishop, 2006):

$$e_{ij} \sim \prod_{k=1}^{K} N(e_{ij} | \mu_k, \tau_k^{-1})^{z_{ijk}}, \quad (4)$$

$$z_{ij} \sim \text{Multinomial}(z_{ij} | \pi),$$

where $z_{ij} = (z_{ij1}, \ldots, z_{ijk}) \in \{0,1\}^K$, $\sum_{k=1}^{K} z_{ijk} = 1$ and $z_{ij}$ follows a multinomial distribution parameterized by $\pi = (\pi_1, \ldots, \pi_K)$. To complete the Bayesian model, we introduce conjugate priors over the parameters of Gaussian components, $\mu_k$s, $\tau_k$s, and the mixing proportions, $\pi$, as:

$$\mu_k, \tau_k \sim N(\mu_0, (\beta_0 \tau_0)^{-1}) \text{Gam}(\tau_k | c_0, d_0), \quad (5)$$

$$\pi \sim \text{Dir}(\pi | a_0),$$

where $\text{Gam}(\tau_0 | c_0, d_0)$ is the Gamma distribution with parameters $c_0$ and $d_0$, and $\text{Dir}(\pi | a_0)$ denotes the Dirichlet distribution parameterized by $a_0 = (a_{01}, \ldots, a_{0K})$.

Low-rank Component Modeling. One simple way to model the low-rank component $L$ is to impose a Laplacian prior over the singular values of $L$. Another way is to incorporate the beta-Bernoulli priors on the singular values, resulting in exact zeros on most singular values (Ding et al., 2011). In this paper, we adopt the ARD for low-rank component modeling (Babacan et al., 2012) due to its fast speed and good scalability.

We formulate $L \in \mathbb{R}^{m \times n}$ with rank $l \leq \min(m, n)$ as the product of $U \in \mathbb{R}^{n \times R}$ and $V \in \mathbb{R}^{n \times R}$:

$$L = UV^T = \sum_{r=1}^{R} u_{r}v_{r}^T, \quad (6)$$

where $R > l$, and $u_{r}$ ($v_{r}$) is the $r$th column of $U$ ($V$). Our goal is to achieve column sparsity in $U$ and $V$, such that some columns in $U$ and $V$ will approach zeros. The low-rank nature of $L$ can then be guaranteed.

This goal can be achieved by imposing the following priors on $U$ and $V$:

$$u_r \sim N(u_r | 0, \gamma_r^{-1}I_m), \quad v_r \sim N(v_r | 0, \gamma_r^{-1}I_n), \quad (7)$$

where $I_m$ denotes the $m \times m$ identity matrix. The conjugate prior on each precision variable $\gamma_r$ is:

$$\gamma_r \sim \text{Gam}(\gamma_r | a_0, b_0). \quad (8)$$

Note that each column pair $u_r$, $v_r$ of $U$, $V$ has the same sparsity profile characterized by the common precision variable $\gamma_r$. It has been validated that such a modeling could lead to large precision values of some $\gamma_r$s, and hence result in a good low-rank estimate of $L$ (Babacan et al., 2012).

Combining Eqs. (2), (4)-(8) together, we can construct the full Bayesian model of RPCA with MoG noise, denoted by MoG-RPCA. The goal turns to infer the posterior of all involved variables:

$$p(U, V, Z, \mu, \tau, \pi, \gamma | Y), \quad (9)$$

where $Z = \{z_{ij}\}$, $\mu = (\mu_1, \ldots, \mu_K)$, $\tau = (\tau_1, \ldots, \tau_K)$, and $\gamma = (\gamma_1, \ldots, \gamma_R)$.

3.2. Remarks

Capability of MoG-RPCA in Fitting Sparse Noise: Since the original RPCA model was designed to deal with sparse noise, we need to evaluate if the proposed MoG-RPCA can also handle sparse noise. Consider a MoG distribution with two components:

$$p(x) = \pi N(x | \mu_1, \tau_1^{-1}) + (1 - \pi) N(x | \mu_2, \tau_2^{-1}). \quad (10)$$

Let $\mu_1 = 0$ and $\tau_1^{-1} = 0$. Eq. (10) degenerates to:

$$p(x) = \pi \delta(0) + (1 - \pi) N(x | \mu_2, \tau_2^{-1}), \quad (11)$$

where $\delta(0)$ is the Dirac delta distribution concentrated at 0. This distribution can be equivalently described by the generative model as:

$$x \sim z \delta(0) + (1 - z) N(x | \mu_2, \tau_2^{-1}), \quad z \sim \text{Bernoulli}(\pi), \quad (12)$$

where $\text{Bernoulli}(\pi)$ is the Bernoulli distribution with parameter $\pi$, implying that the variable $x$ is zero with probability $\pi$. This distribution is actually the spike-and-slab prior, from which sparsity can be naturally confirmed (Ishwaran & Rao, 2005). The spike-and-slab prior can thus be viewed as a special case of MoG, that is, the MoG-RPCA is also capable of fitting sparse noise, as can be easily observed from Figure 2.

Merits of MoG-RPCA: The first merit of MoG-RPCA is that it is capable of fitting a much wider range of noises than the traditional RPCA models.
3.3. Variational Inference

We use the variational Bayes (VB) (Bishop, 2006) method to infer the posterior of MoG-RPCA. VB seeks an approximation distribution \( q(x) \) to the true posterior \( p(x|D) \) (\( D \) denotes the observed data) by solving the following variational optimization:

\[
\min_{q \in \mathcal{C}} \text{KL}(q||p) = - \int q(x) \ln \left( \frac{p(x|D)}{q(x)} \right) \, dx, \tag{13}
\]

where \( \text{KL}(q||p) \) denotes the KL divergence between \( q(x) \) and \( p(x|D) \), and \( C \) denotes the set of probability densities with certain restrictions to make the minimization tractable. Taking \( q(x) = \prod_i q_j(x_j) \), the closed-form solution to \( q_j(x_j) \), with other factors fixed, can be attained by:

\[
q_j^*(x_j) = \frac{\exp \left\{ \langle p(x|D) \rangle |x_i=x_j \rangle \right\}}{\int \exp \left\{ \langle p(x|D) \rangle |x_i=x_j \rangle \right\} \, dx_j}, \tag{14}
\]

where \( \langle \cdot \rangle \) denotes the expectation, and \( x_i|x_j \) denotes the set of \( x \) with \( x_j \) removed. Eq. (13) can be solved by alternatively calculating (14).

Let’s approximate the posterior distribution (9) with the following factorized form:

\[
q(U, V, Z, \mu, \tau, \pi, \gamma) = \prod_i q(u_i) \prod_j q(v_j) \prod_{ij} q(z_{ij}) \prod_k q(\mu_k) \prod_k q(\tau_k) \prod_r q(\pi_r), \tag{15}
\]

where \( u_i \) (or \( v_j \)) is the \( i \)th (\( j \)th) row of \( U \) (or \( V \)). Then we can analytically infer all the factorized distributions involved in Eq. (15) as below. The computational details are given in the supplementary material.

**Estimation of Noise Component:** The parameters involved in the noise component are \( \mu, \tau, Z \) and \( \pi \). Based on the prior imposed in Eq. (5) and its conjugate property, we can get the following update equation for each \( \mu_k, \tau_k \) (\( k = 1, \ldots, K \)):

\[
q(\mu_k, \tau_k) = \mathcal{N}(\mu_k | m_k, (\beta_k \tau_k)^{-1}) \text{Gam}(\tau_k | c_k, d_k), \tag{16}
\]

where

\[
\beta_k = 1 + \sum_{ij} \langle z_{ijk} \rangle,
\]

\[
m_k = \frac{1}{\beta_k} \left( \beta_0 \mu_0 + \sum_{ij} \langle z_{ijk} \rangle (y_{ij} - \langle u_i \rangle \langle v_j \rangle^T) \right),
\]

\[
c_k = c_0 + \frac{1}{2} \sum_{ij} \langle z_{ijk} \rangle,
\]

\[
d_k = d_0 + \frac{1}{2} \sum_{ij} \langle z_{ijk} \rangle \left( (y_{ij} - \langle u_i \rangle \langle v_j \rangle^T)^2 + \beta_0 \mu_0^2 \right) - \frac{1}{\beta_k} \sum_{ij} \langle z_{ijk} \rangle (y_{ij} - \langle u_i \rangle \langle v_j \rangle^T + \beta_0 \mu_0)^2.
\]

Similarly, it is easy to obtain the update equation for mixing proportions \( \pi \):

\[
q(\pi) = \text{Dir}(\pi | \alpha), \tag{17}
\]

where \( \alpha = (\alpha_1, \ldots, \alpha_k) \), \( \alpha_k = \alpha_0 + \sum_{ij} \langle z_{ijk} \rangle \).

**Estimation of Low-rank Component:** The parameters involved in the low-rank component are \( U, V \) and \( \gamma \). For each row \( u_i \) of \( U \), using the factorization (15), we can get

\[
q(u_i) = \mathcal{N}(u_i | \mu_u, \Sigma_u), \tag{19}
\]

with mean \( \mu_u \) and covariance \( \Sigma_u \), given by

\[
\mu_{u_i} = \Sigma_u \left\{ \sum_k \langle \tau_k \rangle \sum_j \langle z_{ijk} \rangle (y_{ij} - \langle \mu_k \rangle \langle v_j \rangle)^T \right\},
\]

\[
\Sigma_{u_i} = \left\{ \sum_k \langle \tau_k \rangle \sum_j \langle z_{ijk} \rangle (\langle v_j^T v_j \rangle + \Gamma)^{-1} \right\},
\]

where \( \Gamma = \text{diag}(\langle \gamma \rangle) \). Similarly, for each row \( v_j \) of \( V \), we have

\[
q(v_j) = \mathcal{N}(v_j | \mu_v, \Sigma_v), \tag{20}
\]

where

\[
\mu_{v_j} = \Sigma_v \left\{ \sum_k \langle \tau_k \rangle \sum_i \langle z_{ijk} \rangle (y_{ij} - \langle \mu_k \rangle \langle u_i \rangle) \right\}^T,
\]

\[
\Sigma_{v_j} = \left\{ \sum_k \langle \tau_k \rangle \sum_i \langle z_{ijk} \rangle (\langle u_i^T u_i \rangle + \Gamma)^{-1} \right\}.
\]

For \( \gamma \) which controls the rank of \( L \), we have

\[
q(\gamma) = \text{Gam}(\gamma | a_r, b_r), \tag{21}
\]
where
\[ a_i = a_0 + \frac{m + n}{2}, \quad b_i = b_0 + \frac{1}{2} \left( \langle u_i^T u_r \rangle + \langle v_i^T v_r \rangle \right). \]

As discussed in Babacan et al. (2012), some \( \gamma_i \)'s tend to be very large during the inference process and the corresponding \( u_i \) and \( v_r \) will be removed from \( U \) and \( V \). The low-rank property of \( L \) can thus be achieved.

**Setting of the Hyperparameters:** We set all the hyperparameters involved in our model in a non-informative manner to make them influence as less as possible the inference of posterior distributions (Bishop, 2006). Throughout our experiments, we set \( \mu_0 = 0 \), and \( \alpha_0, \ldots, \alpha_K, \beta_0, b_0, c_0, d_0 \) a small value \( 10^{-6} \). Our method performs stably well on all experiments with these easy settings.

**Tuning of the Number of Gaussians:** We use a simple but effective method to automatically tune the number of Gaussians \( K \). We first run the proposed MoG-RPCA method with a relatively large \( K \). Then we check if there exist two analogous noise components (with means \( \mu_i, \mu_j \) and variances \( \tau_i^{-1}, \tau_j^{-1} \), respectively) which satisfy that both \( |\mu_i - \mu_j|/(|\mu_i| + |\mu_j|) \) and \( |\tau_i^{-1} - \tau_j^{-1}|/(\tau_i^{-1} + \tau_j^{-1}) \) are smaller than a preset threshold. If yes, we set \( K \) to \( K - 1 \) and re-run our method by taking current parameters as initialization- and combining the two analogous Gaussian components into one. Otherwise we terminate the iteration and output the result. Such a strategy is efficient since the information obtained in previous step is fully utilized as initialization to the next step. In all our experiments, we simply set the maximum \( K \) as 6, and the experimental results showed that it is flexible enough to fit the noises in all synthetic and real data we used.

**Complexity:** It is easy to see that only simple computations are involved in the variational inference of parameters, except that inferring each of \( u_i \) and \( v_j \) needs to invert a \( R \times R \) matrix, leading to \( \mathcal{O}\left((m+n)R^3\right) \) costs in total. Altogether, the complexity of MoG-RPCA is \( \mathcal{O}((m+n)R^3 + KmnR + mnR^2) \) per iteration, where \( m, n, K, R \) are the dimensionality and size of the input data, the MoG number, and the rank p-resetting, respectively. The cost of our method is thus linear in both data dimensionality and size, which is comparable to the existing RPCA algorithms.

**4. Experiments**

We evaluate the performance of the proposed MoG-RPCA method on synthetic, face and video data. The competing methods include classic PCA and representative RPCA and LRMF methods: RPCA (Wright et al., 2009), BRPCA (Ding et al., 2011), VBRPCA (Babacan et al., 2012), ALP (Ke & Kanade, 2005), RegL1ALM (Zheng et al., 2012) and RPMF (Wang et al., 2012). We used the publicly available codes downloaded from the authors’ websites, except for ALP written by ourselves and PCA by the “svd” function in Matlab. All experiments were implemented in Matlab on a PC with 2.60GHz CPU and 8GB RAM.

**4.1. Synthetic Simulations**

Ten sets of synthetic data were generated to evaluate the performance of MoG-RPCA with different types of noises. In the first five sets of simulations, we randomly generated 20 matrices with size \( 100 \times 100 \) and rank \( r = 5 \). Each of these matrices was generated by the product of two smaller matrices as \( UV^T \). Both \( U \) and \( V \) are of sizes \( 100 \times r \), and their entries were independently drawn from \( \mathcal{N}(0,1) \). We further added certain types of noise to the ground truth matrix as follows. (1) No noise added. (2) Sparse noise: 10% entries mixed with uniform noise within \([-25,25]\). (3) Gaussian noise: all entries mixed with Gaussian noise \( \mathcal{N}(0,0.05) \). (4) Mixture noise with zero mean: 10% entries mixed with uniform noise between \([-25,25]\), 20% with Gaussian noise \( \mathcal{N}(0,1) \), and the other 70% with Gaussian noise \( \mathcal{N}(0,0.01) \). (5) Mixture noise with nonzero mean: 10% entries mixed with uniform noise within \([-15,35]\), 30% with Gaussian noise \( \mathcal{N}(0.1,1) \), and the other 60% with Gaussian noise \( \mathcal{N}(-0.1,0.01) \).

The other five sets of simulations were similarly constructed except for setting rank \( r = 10 \).

Two criteria were utilized for performance assessment. (1) Relative reconstruction error (RRE): \( \|\hat{L} - L\|_F/\|L\|_F \), where \( L \) and \( \hat{L} \) denote the ground truth and reconstructed low-rank matrices, respectively. (2) Estimated rank (ER): the rank of \( \hat{L}^2 \). The performance of each competing method on each simulation was evaluated as the average over the 20 matrices in terms of RRE and ER, as listed in Table 1.

We can see from Table 1 that, unsurprisingly, PCA has the best performance in the no noise and Gaussian noise cases, and the \( L_1 \)-norm based methods perform better in the case of sparse noise. While our method always has comparable performance in these situations, its advantage tends to be significant in the case of more complex noise, which is demonstrated by the following Mann-Whitney-Wilcoxon test (Mann & Whitney, 1947). In Gaussian noise experiments, the performance of MoG-RPCA is not significantly differ-

---

1See supplementary material for links.

2Since in the LRMF methods, PCA, ALP, RegL1ALM and RPMF, the rank is fixed as a pre-specified parameter, this criterion is not applicable to this line of methods.
Robust Principal Component Analysis with Complex Noise

Table 1. Performance evaluation on synthetic data. The best results in terms of RRE are highlighted in bold.

<table>
<thead>
<tr>
<th>Method</th>
<th>No Noise</th>
<th>Sparse Noise</th>
<th>Gaussian Noise</th>
<th>Mixture Noise (zero mean)</th>
<th>Mixture Noise (nonzero mean)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>rank(L) = 5</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>RRE</td>
<td>Time(s)</td>
<td>RRE</td>
<td>Time(s)</td>
<td>RRE</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PCA</td>
<td>1.04e-15</td>
<td>0.123</td>
<td>1.050</td>
<td>0.121</td>
<td>1.050</td>
</tr>
<tr>
<td>RPCA</td>
<td>3.43e-8</td>
<td>0.232</td>
<td>5.11e-4</td>
<td>3.96e-4</td>
<td>1.96e-4</td>
</tr>
<tr>
<td>BRPCA</td>
<td>2.68e-2</td>
<td>0.997</td>
<td>2.41e-4</td>
<td>2.68e-2</td>
<td>2.41e-4</td>
</tr>
<tr>
<td>VBRPCA</td>
<td>0.193</td>
<td>0.117</td>
<td>4.21e-4</td>
<td>0.117</td>
<td>4.21e-4</td>
</tr>
<tr>
<td>ALP</td>
<td>58</td>
<td>0.076</td>
<td>1.11</td>
<td>0.076</td>
<td>1.11</td>
</tr>
<tr>
<td>RegL1ALM</td>
<td>3.11e-2</td>
<td>0.316</td>
<td>3.11e-2</td>
<td>0.316</td>
<td>3.11e-2</td>
</tr>
<tr>
<td>RPMF</td>
<td>3.11e-2</td>
<td>0.316</td>
<td>3.11e-2</td>
<td>0.316</td>
<td>3.11e-2</td>
</tr>
<tr>
<td>MoG-RPCA</td>
<td>3.11e-2</td>
<td>0.316</td>
<td>3.11e-2</td>
<td>0.316</td>
<td>3.11e-2</td>
</tr>
</tbody>
</table>

Table 2. Quantitative comparison of the ground truth (denote by “True”) noise probability density functions and those estimated (denote by “Est.”) by the MoG-RPCA method in the synthetic experiments.

<table>
<thead>
<tr>
<th>Method</th>
<th>No Noise</th>
<th>Sparse Noise</th>
<th>Gaussian Noise</th>
<th>Mixture Noise (zero mean)</th>
<th>Mixture Noise (nonzero mean)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>rank(L) = 5</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>σ_k</td>
<td>Comp. 1</td>
<td>Comp. 2</td>
<td>Comp. 1</td>
<td>Comp. 2</td>
</tr>
<tr>
<td></td>
<td>μ_k</td>
<td>Comp. 1</td>
<td>Comp. 2</td>
<td>Comp. 1</td>
<td>Comp. 2</td>
</tr>
<tr>
<td></td>
<td>v_k</td>
<td>Comp. 1</td>
<td>Comp. 2</td>
<td>Comp. 1</td>
<td>Comp. 2</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PCA</td>
<td>True</td>
<td>0.003</td>
<td>0.003</td>
<td>0.003</td>
<td>0.003</td>
</tr>
<tr>
<td>RPCA</td>
<td>Est</td>
<td>0.057</td>
<td>0.057</td>
<td>0.057</td>
<td>0.057</td>
</tr>
<tr>
<td>BRPCA</td>
<td>Est</td>
<td>0.057</td>
<td>0.057</td>
<td>0.057</td>
<td>0.057</td>
</tr>
<tr>
<td>VBRPCA</td>
<td>Est</td>
<td>0.057</td>
<td>0.057</td>
<td>0.057</td>
<td>0.057</td>
</tr>
<tr>
<td>ALP</td>
<td>True</td>
<td>0.003</td>
<td>0.003</td>
<td>0.003</td>
<td>0.003</td>
</tr>
<tr>
<td>RegL1ALM</td>
<td>Est</td>
<td>0.057</td>
<td>0.057</td>
<td>0.057</td>
<td>0.057</td>
</tr>
<tr>
<td>RPMF</td>
<td>Est</td>
<td>0.057</td>
<td>0.057</td>
<td>0.057</td>
<td>0.057</td>
</tr>
<tr>
<td>MoG-RPCA</td>
<td>Est</td>
<td>0.057</td>
<td>0.057</td>
<td>0.057</td>
<td>0.057</td>
</tr>
</tbody>
</table>

The good performance of the proposed MoG-RPCA method in complex noise cases can be easily explained by Table 2 and Figure 2, which compare the ground truth noises and those estimated by our method. It is easy to see that the estimated noise distributions comply with the real ones very well. Especially, in the cases of complicated mixture noise (the 3rd and 4th columns of Figure 2), our method very faithfully resolves the real noise configurations.

We also compared the average CPU times for all methods in Table 1. As can be seen, our method costs comparable CPU time with other methods in most situations, which complies with the complexity analysis in Section 3.3. Considering its superiority in complex noise adaption and automatic rank estimation, it is reasonable to say that our method is efficient.

4.2. Face Modeling

This experiment aims to test the effectiveness of MoG-RPCA in face modeling applications. The second subset of the Extended Yale B database (Georghiades et al., 2001), consisting of 64 faces of one subject with size 192 × 168, was used to generate the data matrix with size 3256 × 64. Typical images are depicted in Figure 3. All competing methods were implemented,
Robust Principal Component Analysis with Complex Noise

Figure 2. Visual comparison of the ground truth (denote by “True”) noise probability density functions and those estimated (denote by “Est.”) by the MoG-RPCA method in the synthetic experiments. The embedded sub-figures depict the zoom-in of the indicated portions.

except for ALP which encounters the “out of memory” problem. We set the rank as 4 (Basri & Jacobs, 2003) for the factorization-based methods, including PCA, RegL1ALM and RPMF. For the other competing methods, the rank was automatically learned from data. The reconstructed faces and the extracted noises by all methods are compared in Figure 3.

The proposed method, as well as the other competing methods, is able to remove the cast shadows and saturations in faces. Our method, however, performs better on faces with a large dark region. Such face images contain both significant cast shadow and saturation noises, which correspond to the highly dark and bright areas in face, and camera/read noise (Nakamura, 2005) which is much amplified in the dark areas. It is very interesting that the proposed method is capable of accurately extracting these two kinds of noises, as clearly depicted in Figure 3. The better noise fitting capability of the proposed method thus leads to better face reconstruction performance.

4.3. Background Subtraction

Background subraction from video sequences captured by a static camera can be modeled as a low-rank matrix analysis problem (Wright et al., 2009; Candès et al., 2011). Four commonly utilized video sequences, including two indoor scenes (Bootstrap and Hall) and two outdoor scenes (Fountain and Campus), provided by Li et al. (2004)\(^3\), were adopted in our experiments. We extracted 400 frames from the Fountain sequence and 600 frames from each of the other three sequences. All the competing methods except ALP were implemented for comparison. For the factorization-based methods, including PCA, RegL1ALM and RPMF, several rank parameters were tried and the best one was

\(^3\)http://perception.i2r.a-star.edu.sg/bk_model/bk_index
recorded. For the other methods, the rank was learned from data. The results for typical sample frames are shown in Figures 1 and 4.

It can be seen that all the competing methods can extract the background from videos with slight differences in visualization. Our method, however, can extract more elaborate foreground (noise) information. More specifically, our method can discover three levels of foreground information with different variations from the Hall and Bootstrap videos: moving people, shadows alongside the foreground objects and background noise, and from the Fountain and Campus videos: moving people/bus, variations of fountain/tree leaves, and background noise.

We also tested our method on a real traffic video sequence acquired by a static surveillance camera\(^4\), which is more challenging due to diverse variations of its foreground cars. Typical background frames extracted by competing methods are shown in Figure 5. It is easy to see that our method more clearly removes foreground information and attains a better subtraction of background. This further substantiates the effectiveness of the proposed method in real applications.

5. Conclusion

We proposed a new RPCA method by modeling noise as a MoG distribution under the Bayesian framework. Compared with the current RPCA methods, which assume certain noise distribution (e.g., Gaussian or sparse noise) on data, our method can perform the RPCA task under more complex noises. The effectiveness of our method was demonstrated by synthetic data with artificial noises and by face modeling and background subtraction problems with real noises. The proposed method shows clear advantages over previous methods on its capability in accurately recovering the low-rank structure and elaborately extracting the multimodal noise configuration from observed data.

Acknowledgments

This research was supported by 973 Program of China with No. 2013CB329404, the NSFC projects with No. 61131006, 61075054, 61271093, and HK RGC GRF grant (PolyU 5313/13E).

\(^4\)http://www.eecs.qmul.ac.uk/~andrea/avss2007_d.html
References


