Supplemental Material for “Bayesian Robust Matrix Factorization for Image and Video Processing”

Anonymous ICCV submission

Paper ID 306

1. More Details on PRMF Model

PRMF [3] places a Laplace distribution on the residual error as likelihood and Gaussian distributions on the factors as priors. Mathematically, the probabilistic generative process for PRMF is given as follows:

1. Draw each basis element u_{ij} from $\mathcal{N}(0, \lambda_u^{-1})$.
2. Draw each coefficient v_{ij} from $\mathcal{N}(0, \lambda_v^{-1})$.
3. Draw each observation y_{ij} from $\mathcal{L}(\mathbf{u}_i^T \mathbf{v}_j, \lambda)$.

Model inference seeks to find the model parameters $\mathbf{U}$ and $\mathbf{V}$ that maximize the joint posterior probability. From Bayes' rule, the joint posterior distribution can be expressed as

$$p(\mathbf{U}, \mathbf{V} \mid \mathbf{Y}, \lambda, \lambda_u, \lambda_v) \propto p(\mathbf{Y} \mid \mathbf{U}, \mathbf{V}, \lambda)\, p(\mathbf{U} \mid \lambda_u)\, p(\mathbf{V} \mid \lambda_v). \tag{1}$$

By taking logarithm, we get

$$\log p(\mathbf{U}, \mathbf{V} \mid \mathbf{Y}, \lambda, \lambda_u, \lambda_v) = -\lambda \|\mathbf{Y} - \mathbf{U}\mathbf{V}^T\|_1 - \frac{\lambda_u}{2}\|\mathbf{U}\|_2^2 - \frac{\lambda_v}{2}\|\mathbf{V}\|_2^2 + C, \tag{2}$$

where C is a constant independent of $\mathbf{U}$ and $\mathbf{V}$. It is easy to see that maximizing the log posterior probability in Eqn. (2) is equivalent to the following minimization problem:

$$\min_{\mathbf{U}, \mathbf{V}} \|\mathbf{Y} - \mathbf{U}\mathbf{V}^T\|_1 + \frac{\lambda'_u}{2}\|\mathbf{U}\|_2^2 + \frac{\lambda'_v}{2}\|\mathbf{V}\|_2^2, \tag{3}$$

where $\lambda'_u = \lambda_u/\lambda$ and $\lambda'_v = \lambda_v/\lambda$. This shows that conventional robust matrix factorization based on the l_1 loss can be derived from a probabilistic formulation. Moreover, the authors make a connection of this formulation with the well-known *principal component pursuit* (PCP) [1] by exploiting the relationship between the summation of the Frobenius norms of $\mathbf{U}$ and $\mathbf{V}$ and the nuclear norm of $\mathbf{U}\mathbf{V}^T$. Further details can be found in the original paper [3].

A major difficulty with model inference is attributed to the fact that the l_1 norm is non-smooth at zero, making traditional gradient methods unsuitable. To avoid this problem and to make the inference efficient, the authors exploit a useful hierarchical view of the Laplace distribution by expressing it as an infinite Gaussian mixture with the exponential distribution as mixing distribution:

$$\mathcal{L}(z \mid u, \alpha) = \int_0^\infty \mathcal{N}(z \mid u, \tau)\, \mathrm{Exp}(\tau \mid \frac{\alpha}{2})\, d\tau. \tag{4}$$

With this, the model can be reformulated as

$$\begin{aligned} y_{ij} \mid \mathbf{U}, \mathbf{V}, \mathbf{T} &\sim \mathcal{N}(y_{ij} \mid \mathbf{u}_i^T \mathbf{v}_j, \tau_{ij}) \\ u_{ij} \mid \lambda_u &\sim \mathcal{N}(u_{ij} \mid 0,\ \lambda_u^{-1}) \\ v_{ij} \mid \lambda_v &\sim \mathcal{N}(v_{ij} \mid 0,\ \lambda_v^{-1}) \\ \tau_{ij} \mid \lambda &\sim \mathrm{Exp}(\tau_{ij} \mid \lambda/2), \end{aligned} \tag{5}$$

where $\mathbf{T} = [\tau_{ij}] \in \mathbb{R}^{m\times n}$ and each element τ_{ij} is a latent variable for the corresponding y_{ij} with an exponential prior.

By regarding $\mathbf{T}$ as a hidden variable, the EM algorithm can be applied. In the E-step, the expectation of the posterior of τ_{ij}^{-1} is computed as follows:

$$E[\tau_{ij}^{-1}|\mathbf{Y},\mathbf{U},\mathbf{V}] = \frac{\sqrt{\lambda}}{|r_{ij}|} \triangleq \langle\tau_{ij}^{-1}\rangle, \tag{6}$$

where $r_{ij} = y_{ij} - \mathbf{u}_i^T\mathbf{v}_j$. In the M-step, the expectation is maximized by computing its derivatives with respect to $\mathbf{U}$ and $\mathbf{V}$, respectively, to obtain the following update rules:

$$\begin{aligned}\mathbf{v}_j &= (\mathbf{U}^T\Omega_j\mathbf{U} + \lambda_v\mathbf{I})^{-1}\mathbf{U}^T\Omega_j\mathbf{y}_{\cdot j}\\ \mathbf{u}_i &= (\mathbf{V}^T\Lambda_i\mathbf{V} + \lambda_u\mathbf{I})^{-1}\mathbf{V}^T\Lambda_i\mathbf{y}_{i\cdot},\end{aligned} \tag{7}$$

where $\mathbf{y}_{i\cdot}$ and $\mathbf{y}_{\cdot j}$ denote the ith row and jth column, respectively, of $\mathbf{Y}$, $\Omega_j \triangleq \mathrm{diag}(\langle\tau_{1j}^{-1}\rangle,\ldots,\langle\tau_{mj}^{-1}\rangle)$, and $\Lambda_i \triangleq \mathrm{diag}(\langle\tau_{i1}^{-1}\rangle,\ldots,\langle\tau_{in}^{-1}\rangle)$.

2. Generalized Inverse Gaussian Distribution

The pdf of the GIG distribution is defined as follows:

$$f(x \mid p,a,b) = \frac{(a/b)^{p/2}}{2K_p(\sqrt{ab})}x^{(p-1)}e^{-(ax+b/x)/2}, \quad x>0, \tag{8}$$

where $a>0$, $b>0$ and p are three parameters that control the shape of the distribution, and $K_p(\cdot)$ is a modified Bessel function of the second kind. GIG can be regarded as a generalization of many well-known distributions in the exponential family, which include the gamma distribution ($b=0$), inverse gamma distribution ($a=0$), inverse Gaussian distribution ($p=-1/2$), etc. The general form for GIG makes it very suitable as prior distributions for Bayesian methods [4]. For more properties of GIG, please refer to [2].

3. More Details on BRMF Model Inference

This section provides more details beyond those in section 5.2 of the paper.

Sample $\boldsymbol{\mu}_u$ and $\boldsymbol{\Lambda}_u$: The joint posterior distribution of these two variables is a standard result of the exponential family of distributions. See `http://en.wikipedia.org/wiki/Conjugate_prior` for more details.

Sample $\mathbf{u}_i$: We extract all terms related to $\mathbf{u}_i$ and then apply Bayes' rule:

$$\begin{aligned}p(\mathbf{u}_i\,|\,\mathbf{Y},\mathbf{V},\boldsymbol{\mu}_u,\boldsymbol{\Lambda}_u,\mathbf{T}) &\propto \prod_{j=1}^{n}\mathcal{N}(y_{ij}\,|\,\mathbf{u}_i^T\mathbf{v}_j,\tau_{ij})\mathcal{N}(\mathbf{u}_i\,|\,\boldsymbol{\mu}_u,\boldsymbol{\Lambda}_u^{-1})\\ &\propto \exp\left\{-\frac{1}{2}\left[\sum_{j=1}^{n}\frac{1}{\tau_{ij}}(y_{ij}-\mathbf{u}_i^T\mathbf{v}_j)^2 + (\mathbf{u}_i-\boldsymbol{\mu}_u)^T\boldsymbol{\Lambda}_u(\mathbf{u}_i-\boldsymbol{\mu}_u)\right]\right\}\\ &\propto \exp\left\{-\frac{1}{2}\left[\sum_{j=1}^{n}\frac{1}{\tau_{ij}}(\mathbf{u}_i^T\mathbf{v}_j\mathbf{v}_j^T\mathbf{u}_i - 2\mathbf{u}_i^T\mathbf{v}_jy_{ij}) + \mathbf{u}_i^T\boldsymbol{\Lambda}_u\mathbf{u}_i - 2\mathbf{u}_i^T\boldsymbol{\Lambda}_u\boldsymbol{\mu}_u\right]\right\}\\ &= \exp\left\{-\frac{1}{2}\left[\mathbf{u}_i^T\Big(\sum_{j=1}^{n}\frac{\mathbf{v}_j\mathbf{v}_j^T}{\tau_{ij}} + \boldsymbol{\Lambda}_u\Big)\mathbf{u}_i - 2\mathbf{u}_i^T\Big(\boldsymbol{\Lambda}_u\boldsymbol{\mu}_u + \sum_{j=1}^{n}\frac{\mathbf{v}_jy_{ij}}{\tau_{ij}}\Big)\right]\right\}.\end{aligned} \tag{9}$$

By completing the square, we can show that

$$\mathbf{u}_i \mid \mathbf{Y},\mathbf{V},\boldsymbol{\mu}_u,\boldsymbol{\Lambda}_u,\mathbf{T} \sim \mathcal{N}(\mathbf{u}_i \mid \mathbf{u}_i', (\boldsymbol{\Lambda}_i')^{-1}), \tag{10}$$

where

$$\begin{aligned} \mathbf{\Lambda}'_i &= \mathbf{\Lambda}_u + \sum_{j=1}^{n} \frac{\mathbf{v}_j \mathbf{v}_j^T}{\tau_{ij}} \\ \mathbf{u}'_i &= (\mathbf{\Lambda}'_i)^{-1} \left(\sum_{j=1}^{n} \frac{y_{ij} \mathbf{v}_j}{\tau_{ij}} + \mathbf{\Lambda}_u \boldsymbol{\mu}_u \right). \end{aligned} \tag{11}$$

Sample τ_{ij}: We note that

$$p(\tau_{ij} \mid y_{ij}, r_{ij}, \eta_{ij}) \propto \tau_{ij}^{-\frac{1}{2}} \exp\Big[-\frac{1}{2}(\eta_{ij}\tau_{ij} + \frac{r_{ij}^2}{\tau_{ij}}) \Big]. \tag{12}$$

By a change of variables, we have

$$\begin{aligned} p(\frac{1}{\tau_{ij}} \mid y_{ij}, r_{ij}, \eta_{ij}) &\propto \tau_{ij}^{-\frac{1}{2}} \exp\Big[-\frac{1}{2}(\eta_{ij}\tau_{ij} + \frac{r_{ij}^2}{\tau_{ij}}) \Big] \Big| \frac{d\tau_{ij}}{d\tau_{ij}^{-1}} \Big| \\ &\propto \tau_{ij}^{\frac{3}{2}} \exp\Big[-\frac{1}{2}(\eta_{ij}\tau_{ij} + \frac{r_{ij}^2}{\tau_{ij}}) \Big]. \end{aligned} \tag{13}$$

This implies that

$$\frac{1}{\tau_{ij}} \mid y_{ij}, \mathbf{u}_i, \mathbf{v}_j, \eta_{ij} \sim \mathrm{IG}\Big(\frac{\sqrt{\eta_{ij}}}{|r_{ij}|}, \eta_{ij} \Big). \tag{14}$$

Sample η_{ij} :

$$p(\eta_{ij} \mid \tau_{ij}, p, a, b) \;\propto\; \eta_{ij}^{p} \exp\Big[-\frac{1}{2}\Big((a + \tau_{ij})\eta_{ij} + \frac{b}{\eta_{ij}} \Big) \Big]. \tag{15}$$

Thus,

$$\eta_{ij} \mid \tau_{ij}, p, a, b \sim \mathrm{GIG}(p + 1, \tau_{ij} + a, b). \tag{16}$$

4. Contribution of Residue to Objective for SPCP and DECOLOR

In this section, we show how to derive the results in Table 1 of the paper.

To eliminate $\mathbf{E}$, we consider the optimality condition for each case:

SPCP: The optimality condition of $\mathbf{E}$ is obtained by applying the elementwise soft-thresholding operator:

$$e_{ij} = \mathrm{sgn}(r_{ij}) \max(0, |r_{ij}| - \lambda). \tag{17}$$

DECOLOR: The optimality condition of $\mathbf{E}$ is obtained by applying the elementwise hard-thresholding operator:

$$e_{ij} = \begin{cases} r_{ij}, & r_{ij}^2 > 2\lambda \\ 0, & r_{ij}^2 \leq 2\lambda. \end{cases} \tag{18}$$

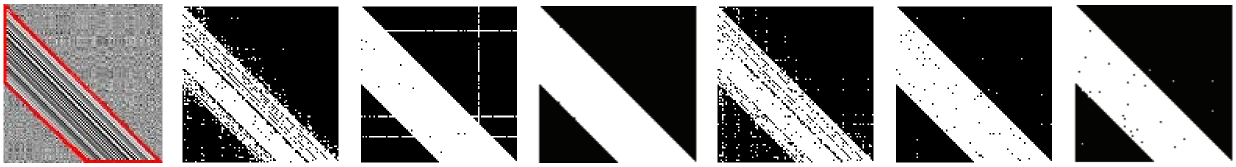
We then substitute the corresponding $\mathbf{E}$ back into the objective function to obtain the results in Table 1 of the paper.

5. Synthetic Experiments

In this experiment, we generate both the low-rank matrix and outliers randomly. Specifically, we follow the same settings as in [5]. We first generate the low-rank background $\mathbf{B} \in \mathbb{R}^{m \times m}$ as $\mathbf{B} = \mathbf{U}\mathbf{V}^T$, where both $\mathbf{U} \in \mathbb{R}^{m \times r}$ and $\mathbf{V} \in \mathbb{R}^{m \times r}$ are generated i.i.d. from Gaussian distributions. Then a region of contiguous outliers is added to the data matrix. A sample data matrix is shown in Figure 1(a) with the outlier pixels surrounded by a red boundary. The recovered masks obtained by different methods are shown in Figure 1(b) to 1(g). VBLR is excluded from this experiment because it often fails to converge, probably because there are a large number of outliers in this experiment.

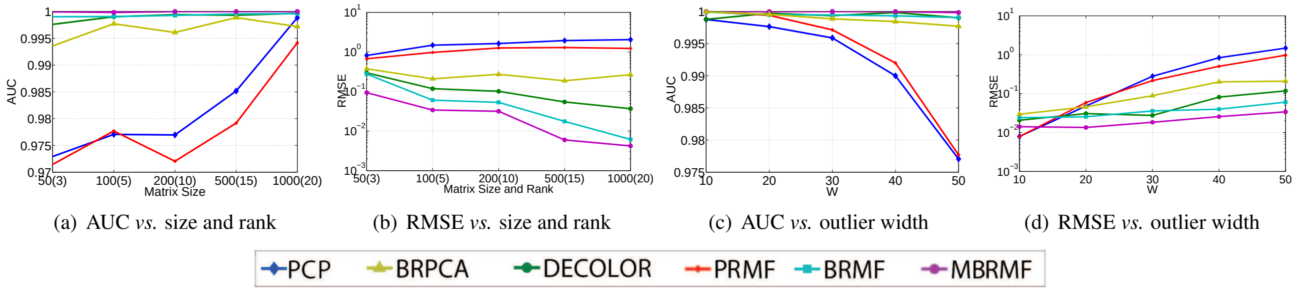
From Figure 1, we can see that DECOLOR, BRMF and MBRMF give similar results and are the best in detecting the outliers. BRPCA gives reasonable results except for the four lines of major errors. Compared with these methods, PCP and PRMF give the worst results.

We also show quantitative results in Figure 2. Each point plotted represents the average of five runs with each run corresponding to a different data matrix. We evaluate the performance of detecting outliers and reconstructing the low-rank matrix using the *area under curve* (AUC) and *root mean squared error* (RMSE) measures separately. For each case, we also vary the matrix size and rank and the width of the outlier region to illustrate how they affect the performance. We fix the outliers width W at $m/2$ in the former case and keep the size and rank of the matrix at 100 and 5, respectively, in the latter case. For outlier detection, the quantitative results agree with the visual results shown in Figure 1. However, as far as recovering the underlying matrix is concerned, MBRMF has significant advantage over other algorithms. In fact, even the basic BRMF model that does not consider the clustering effect of outliers outperforms DECOLOR.



(a) Data (b) PCP (c) BRPCA (d) DECOLOR (e) PRMF (f) BRMF (g) MBRMF

Figure 1. Illustration of synthetic experiment.



(a) AUC *vs.* size and rank (b) RMSE *vs.* size and rank (c) AUC *vs.* outlier width (d) RMSE *vs.* outlier width

Figure 2. Quantitative results for synthetic experiment.

6. More Details on Video Sequences in Real Dataset

We summarize the statistics of the video sequences used in Section 7.3.2 of the paper in Table 1 below. All the test sequences are from `https://sites.google.com/site/backgroundsubtraction/test-sequences`. Note that applying background modeling to short videos is a rather challenging task due to the lack of sufficient information, causing overfitting to occur particularly in optimization-based methods even though some regularizers are introduced.

	Resolution	#Frames	Challenges
fountain	128×160	50	Short length and dynamic background
hall	144×176	24	Extremely short length
waterSurface	128×160	48	Short length and dynamic background
wavingTrees	120×160	287	Outliers with large area and dynamic background

Table 1. Summary of video sequences used in background modeling experiments.

7. More Background Modeling Results with Different Noise Types

Recall that for the noiseless case, all algorithms except PCP give satisfactory results on the *fountain* sequence. Figure 3 shows the performance of different methods on the *fountain* sequence when different types of noise (Speckle, Poisson, Salt, Gaussian) are added. For the salt and pepper noise, all methods show robustness to it. By treating the salt and pepper noise as a collection of singleton outliers, it can be removed successfully. For the other three types of noise, however, only MBRMF and VBLR give satisfactory results while the ghosting effect in various degrees is observed in the other methods.

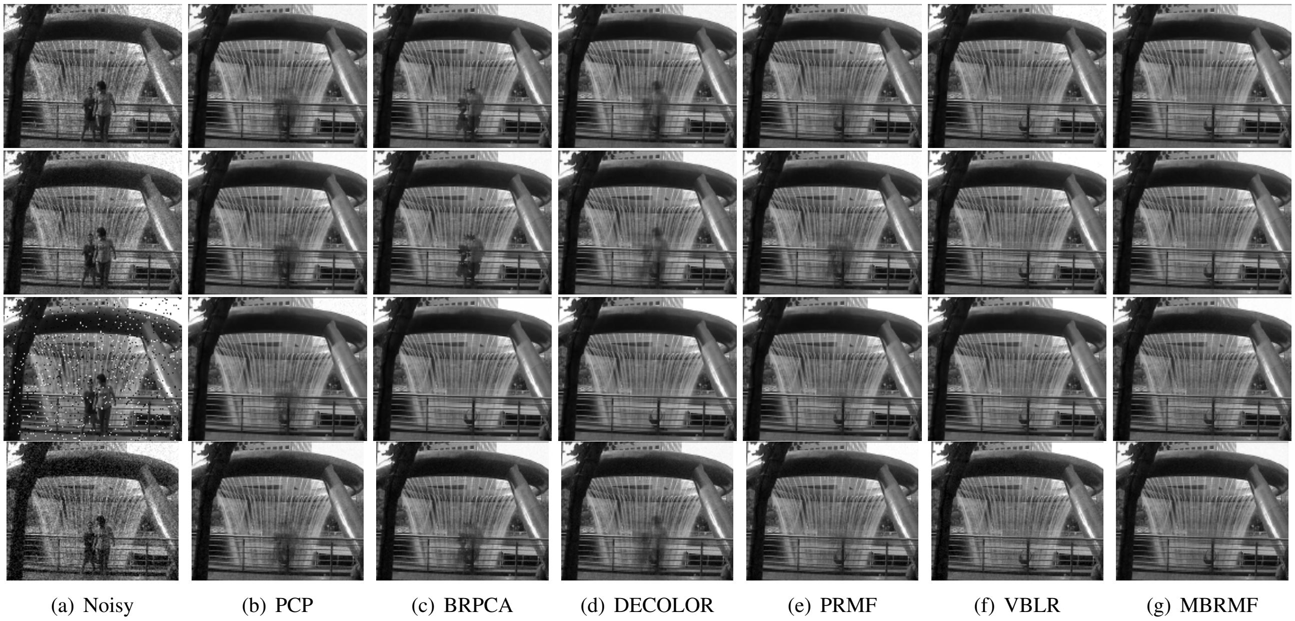

(a) Noisy (b) PCP (c) BRPCA (d) DECOLOR (e) PRMF (f) VBLR (g) MBRMF

Figure 3. One frame in the *fountain* video sequence. Each row corresponds to one type of noise (from top to bottom: Speckle, Poisson, Salt, Gaussian).

References

[1] E. Candes, X. Li, Y. Ma, and J. Wright. Robust principal component analysis? *Journal of the Association for Computing Machinery*, 58(3), 2011. 1

[2] B. Jørgensen. *Statistical Properties of the Generalized Inverse Gaussian Distribution*, volume 21. Springer, New York, 1982. 2

[3] N. Wang, T. Yao, J. Wang, and D.-Y. Yeung. A probabilistic approach to robust matrix factorization. In *ECCV*, pages 126–139, 2012. 1

[4] Z. Zhang, S. Wang, D. Liu, and M. Jordan. EP-GIG priors and applications in Bayesian sparse learning. *Journal of Machine Learning Research*, 13:2031–2061, 2012. 2

[5] X. Zhou, C. Yang, and W. Yu. Moving object detection by detecting contiguous outliers in the low-rank representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(3):597–610, 2013. 3