SPARSITY REGULARIZED PRINCIPAL COMPONENT PURSUIT

Jing Liu, Pamela C. Cosman, and Bhaskar D. Rao

Department of Electrical and Computer Engineering,
University of California at San Diego, La Jolla, CA 92093, USA

ABSTRACT

We study the problem of low-rank and sparse decomposition from possibly noisy observations. We propose a novel objective function with nuclear norm on the low-rank term and ℓ_0-'norm' on the sparse term, as well as ℓ_1-norm on the additive noise term. When there is no dense inlier noise, the proposed method shares the same theoretical guarantee as the Principal Component Pursuit (PCP), i.e., it can recover the low-rank component and sparse component exactly with high probability. Simulations in the noisy case demonstrate that the proposed method outperforms existing state-of-the-art methods. Results on a surveillance video application further verify the effectiveness of the proposed method.

Index Terms— ℓ_0 regularization, low-rank matrix, sparse matrix, Sparsity Regularized Principal Component Pursuit

1. INTRODUCTION

Recovering the low-rank matrix $\boldsymbol{L}_0$ and sparse matrix $\boldsymbol{S}_0$ from their composition $\boldsymbol{M}$ (possibly with additional dense noise) has been extensively studied recently. This problem was first studied in the noiseless case [1][2][3], known as Robust PCA. The formulation of this problem is [2]:

$$\min_{\boldsymbol{L},\boldsymbol{S}} \operatorname{rank}(\boldsymbol{L}) + \gamma\|\boldsymbol{S}\|_0 \quad s.t. \quad \boldsymbol{M} = \boldsymbol{L} + \boldsymbol{S}, \tag{1}$$

which is known to be NP-hard. However, [1][2][3][4] show that by relaxing rank to nuclear norm and ℓ_0-'norm' to ℓ_1-norm, i.e.,

$$\min_{\boldsymbol{L},\boldsymbol{S}} \|\boldsymbol{L}\|_* + \lambda\|\boldsymbol{S}\|_1 \quad s.t. \quad \boldsymbol{M} = \boldsymbol{L} + \boldsymbol{S}, \tag{2}$$

known as Principal Component Pursuit (PCP), one can recover both $\boldsymbol{L}_0$ and $\boldsymbol{S}_0$ exactly with high probability under certain conditions by solving this convex problem. In real world applications, besides the sparse 'corruptions' $\boldsymbol{S}_0$, there is also small magnitude dense inlier noise $\boldsymbol{G}$. The model becomes:

$$\boldsymbol{M} = \boldsymbol{L}_0 + \boldsymbol{S}_0 + \boldsymbol{G} \tag{3}$$

This research was supported by NSF Award IIS-1522125.

Zhou et al. [5] solved the following relaxed version of (2), known as Stable Principal Component Pursuit (SPCP):

$$\min_{\boldsymbol{L},\boldsymbol{S}} \|\boldsymbol{L}\|_* + \lambda\|\boldsymbol{S}\|_1 \quad s.t. \quad \|\boldsymbol{M} - \boldsymbol{L} - \boldsymbol{S}\|_F \le \delta. \tag{4}$$

It was shown that the estimation error can be bounded. Hsu et al. [6] proposed the Lagrange form of (4):

$$\min_{\boldsymbol{L},\boldsymbol{S}} \|\boldsymbol{L}\|_* + \lambda\|\boldsymbol{S}\|_1 + \frac{1}{2\mu}\|\boldsymbol{M} - \boldsymbol{L} - \boldsymbol{S}\|_F^2. \tag{5}$$

In light of M-estimators, He et al. [7] proposed to replace $\|\boldsymbol{S}\|_1$ with implicit regularizers of robust M-estimators, i.e., $\varphi(\boldsymbol{S})$, then solving the following problem:

$$\min_{\boldsymbol{L},\boldsymbol{S}} \mu\|\boldsymbol{L}\|_* + \varphi(\boldsymbol{S}) + \frac{1}{2}\|\boldsymbol{M} - \boldsymbol{L} - \boldsymbol{S}\|_F^2. \tag{6}$$

In [8] and [9], the authors proposed the following greedy approach, which was solved via alternating minimization.

$$\min_{\boldsymbol{L},\boldsymbol{S}} \|\boldsymbol{M} - \boldsymbol{L} - \boldsymbol{S}\|_F^2 \quad s.t. \quad \operatorname{rank}(\boldsymbol{L}) \le r, \|\boldsymbol{S}\|_0 \le k. \tag{7}$$

Recently, Netrapalli et al. [10] proposed a provable non-convex approach that alternates between projecting appropriate residuals onto the set of rank-k matrices, and the set of sparse matrices, where the rank k (as well as the sparsity level) is gradually increased towards the target rank.

Also in a recent work [11], the authors proposed to use an ℓ_0 penalty to enforce both sparsity and low rank:

$$\min_{\boldsymbol{A},\boldsymbol{B},\boldsymbol{S}} \|\boldsymbol{M} - \boldsymbol{A}\boldsymbol{B}^T - \boldsymbol{S}\|_F^2 + h^2\|\boldsymbol{S}\|_0 \quad s.t. \quad \boldsymbol{B}^T\boldsymbol{B} = \boldsymbol{I}_r. \tag{8}$$

However, these methods need to input rank (and sparsity), which are usually unknown in practice and hard to specify.

In this paper, we propose a novel objective function and the associated algorithm to recover the low-rank component $\boldsymbol{L}_0$ and sparse component $\boldsymbol{S}_0$ of a matrix, where we relax the rank to nuclear norm, but use ℓ_0 regularization to directly enforce the sparsity of $\boldsymbol{S}_0$ instead of relaxing it to the ℓ_1-norm. We call our proposed algorithm Sparsity Regularized Principal Component Pursuit (SRPCP). Comparing with [8-11], our method does not need any prior knowledge of rank and sparsity, which are recovered automatically.

We will introduce our objective function and the associated algorithm in Section 2. Theoretical guarantees for the proposed algorithm are studied in Section 3. Section 4 empirically studies the performance of the proposed method and demonstrates a real application. Conclusions and future work are discussed in Section 5.

Notation: Throughout this paper, bold capital letters denote matrices, e.g., $\boldsymbol{L}$, where $\boldsymbol{L}^{(k)}$ denotes the updated $\boldsymbol{L}$ in the kth iteration. $\|\boldsymbol{L}\|_\infty$, $\|\boldsymbol{L}\|_1$, and $\|\boldsymbol{L}\|_0$ denote the ℓ_∞-norm, ℓ_1-norm, and ℓ_0-'norm' of $\boldsymbol{L}$ seen as a long vector, respectively, while $\|\boldsymbol{L}\|_F$ and $\|\boldsymbol{L}\|_*$ denote the Frobenius norm and nuclear norm of the matrix $\boldsymbol{L}$, respectively.

2. SPARSITY REGULARIZED PRINCIPAL COMPONENT PURSUIT

We propose minimizing the following objective function to recover the low-rank component and sparse component:

$$J(\boldsymbol{L},\boldsymbol{S}) = \|\boldsymbol{L}\|_* + \beta\|\boldsymbol{S}\|_0 + \lambda\|\boldsymbol{M}-\boldsymbol{L}-\boldsymbol{S}\|_1 \tag{9}$$

We use the alternating minimization approach to minimize the nonconvex objective function in (9), which directly operates on the ℓ_0-'norm' and alternates between the following two steps. The procedure is summarized in Table 1.

Step 1: Fix $\boldsymbol{S}^{(k)}$, update

$$\boldsymbol{L}^{(k+1)} = \underset{\boldsymbol{L}}{\operatorname{argmin}}\, \|\boldsymbol{L}\|_* + \lambda\|\boldsymbol{M}-\boldsymbol{S}^{(k)}-\boldsymbol{L}\|_1$$

Step 2: Fix $\boldsymbol{L}^{(k+1)}$, update

$$\boldsymbol{S}^{(k+1)} = \underset{\boldsymbol{S}}{\operatorname{argmin}}\, \beta\|\boldsymbol{S}\|_0 + \lambda\|\boldsymbol{M}-\boldsymbol{L}^{(k+1)}-\boldsymbol{S}\|_1$$

At first glance, it seems more reasonable to use the Frobenius norm rather than the ℓ_1-norm in the third term of the objective function (9) and in Step 1, especially for Gaussian noise. We want to point out that, in Step 1 of each iteration, we do not expect that all the outliers are identified by the previous iteration. It is likely some outliers are not identified. Even for the identified outliers, their magnitudes are usually not correct before the algorithm converges. So it is safer to use the ℓ_1-norm in Step 1 than the Frobenius norm, which is very sensitive to large residuals. A similar spirit can be found in our robust linear regression work.

At the beginning, we have no information about outliers except sparsity, so we simply initialize $\boldsymbol{S}^{(0)} = \boldsymbol{0}$. Then in Step 1 of the first iteration, SRPCP solves the following:

$$\min_{\boldsymbol{L}} \|\boldsymbol{L}\|_* + \lambda\|\boldsymbol{M}-\boldsymbol{L}\|_1, \tag{10}$$

which is equivalent to PCP in (2).

Solutions for Each Step:

In Step 1, the subproblem is convex and can be recast as PCP with the data matrix $(\boldsymbol{M}-\boldsymbol{S}^{(k)})$. Various algorithms have been proposed to solve that, e.g., Augmented Lagrange Multiplier (ALM) Method [12], Accelerated Proximal Gradient [13], and Singular Value Thresholding [14].

In Step 2, though the subproblem is not convex, we can directly derive its global solution through entrywise thresholding, which is detailed in Table 1.

Table 1. Sparsity Regularized Principal Component Pursuit

Input: $\boldsymbol{M}$, β, λ
Initialization: $k=0$, $\boldsymbol{S}^{(0)} = \boldsymbol{0}$
While $J(\boldsymbol{L},\boldsymbol{S})$ **not converged DO:**
  Iteration $k+1$
  Step 1: fix $\boldsymbol{S}^{(k)}$, update
    $\boldsymbol{L}^{(k+1)} = \operatorname{argmin}_{\boldsymbol{L}} \|\boldsymbol{L}\|_* + \lambda\|\boldsymbol{M}-\boldsymbol{S}^{(k)}-\boldsymbol{L}\|_1$
  Step 2: fix $\boldsymbol{L}^{(k+1)}$, update
    $\boldsymbol{S}_{i,j}^{(k+1)} = \begin{cases} 0, & |(\boldsymbol{M}-\boldsymbol{L}^{(k+1)})_{i,j}| \le \frac{\beta}{\lambda} \\ (\boldsymbol{M}-\boldsymbol{L}^{(k+1)})_{i,j}, & \textit{otherwise} \end{cases}$
End While
Output: $\hat{\boldsymbol{L}}$ and $\hat{\boldsymbol{S}}$

From Step 2 of SRPCP, if any entry of $|\boldsymbol{M}-\boldsymbol{L}^{(k+1)}|$ is larger than $\frac{\beta}{\lambda}$, this entry will be considered as an outlier corrupted entry. In general, $\frac{\beta}{\lambda}$ should be set at least larger than the inlier noise level, e.g., could be set as several times the standard deviation of inlier noise. In practice the parameter λ is fixed, and adaptation to the noise level is transferred to parameter β.

3. THEORETICAL ANALYSIS

In this section, we study the main properties of SRPCP. It is shown to share the same exact recovery guarantee as PCP when there is no dense inlier noise. The analysis benefits greatly from the analysis of PCP [3]. We first quote the incoherence condition with parameter μ from [3]:

The singular value decomposition of $\boldsymbol{L}_0 \in R^{n_1\times n_2}$ is

$$\boldsymbol{L}_0 = \boldsymbol{U}diag(\sigma_1,...,\sigma_r)\boldsymbol{V}^* = \sum_{i=1}^{r}\sigma_i u_i v_i^*,$$

where *r* is the rank of $\boldsymbol{L}_0$, $\sigma_1,...,\sigma_r$ are the positive singular values, and $\boldsymbol{U}=[u_1,...,u_r], \boldsymbol{V}=[v_1,...,v_r]$ are the matrices of left- and right-singular vectors. Then, the incoherence condition with parameter μ states that

$$\max_i \|\boldsymbol{U}^* e_i\|^2 \le \frac{\mu r}{n_1}, \quad \max_i \|\boldsymbol{V}^* e_i\|^2 \le \frac{\mu r}{n_2} \tag{11}$$

and

$$\|\boldsymbol{U}\boldsymbol{V}^*\|_\infty \le \sqrt{\frac{\mu r}{n_1 n_2}}. \tag{12}$$

Theorem 1. (Exact recovery in noiseless case) Suppose $\boldsymbol{L}_0 \in R^{n\times n}$ obeys the incoherence condition with parameter

μ. Fix any $n \times n$ matrix Σ of signs. Suppose that the support set Ω of $\boldsymbol{S}_0$ is uniformly distributed among all sets of cardinality m, and that $\text{sgn}((\boldsymbol{S}_0)_{i,j}) = \Sigma_{i,j}$ for all $(i,j) \in \Omega$. If

$$\text{rank}(\boldsymbol{L}_0) \leq \rho_r n \mu^{-1} (\log n)^{-2} \quad and \quad m \leq \rho_s n^2, \qquad (13)$$

where ρ_r and ρ_s are positive numerical constants, then there is a numerical constant c such that with probability at least $1 - cn^{-10}$ (over the choice of support of $\boldsymbol{S}_0$), SRPCP with $\lambda = \frac{1}{\sqrt{n}}$ and any $\beta > 0$ recovers $\boldsymbol{L}_0$ exactly in two iterations.

If additionally $\beta < \frac{1}{\sqrt{n}} \min\{|(\boldsymbol{S}_0)_{i,j}| : (\boldsymbol{S}_0)_{i,j} \neq 0\}$, then SRPCP recovers both $\boldsymbol{L}_0$ and $\boldsymbol{S}_0$ exactly.

In the general rectangular case, where $\boldsymbol{L}_0$ is $n_1 \times n_2$, define $n_{(1)} = \max(n_1, n_2), n_{(2)} = \min(n_1, n_2)$. SRPCP with $\lambda = \frac{1}{\sqrt{n_{(1)}}}$ and $0 < \beta < \frac{1}{\sqrt{n_{(1)}}} \min\{|(\boldsymbol{S}_0)_{i,j}| : (\boldsymbol{S}_0)_{i,j} \neq 0\}$ recovers both $\boldsymbol{L}_0$ and $\boldsymbol{S}_0$ exactly with probability at least $1 - cn_{(1)}^{-10}$, provided that $\text{rank}(\boldsymbol{L}_0) \leq \rho_r n_{(2)} \mu^{-1} (\log n_{(1)})^{-2}$ and $m \leq \rho_s n_1 n_2$.

Proof. In Step 1 of the first iteration, SRPCP solves the Robust PCA problem via (10) with $\lambda = 1/\sqrt{n}$. From Theorem 1.1 of [3], we know there is a numerical constant c such that with probability at least $1 - cn^{-10}$, $\boldsymbol{L}^{(1)} = \boldsymbol{L}_0$.

In the following, we prove that if $\boldsymbol{L}^{(1)} = \boldsymbol{L}_0$, we must have $\boldsymbol{L}^{(2)} = \boldsymbol{L}_0$, thus SRPCP converges. Then with probability at least $1 - cn^{-10}$, SRPCP recovers $\boldsymbol{L}_0$ exactly.

In Step 2 of the first iteration, as $\boldsymbol{L}^{(1)} = \boldsymbol{L}_0$, for any $\beta > 0$, $\boldsymbol{S}^{(1)}$ will be a trimmed version [1] of $\boldsymbol{S}_0$.

In Step 1 of the second iteration, SRPCP solves

$$\boldsymbol{L}^{(2)} = \underset{\boldsymbol{L}}{\text{argmin}} \, \|\boldsymbol{L}\|_* + \lambda \|\boldsymbol{M} - \boldsymbol{S}^{(1)} - \boldsymbol{L}\|_1 \qquad (14)$$

$$= \underset{\boldsymbol{L}}{\text{argmin}} \, \|\boldsymbol{L}\|_* + \lambda \|(\boldsymbol{L}_0 + \boldsymbol{S}_0) - \boldsymbol{S}^{(1)} - \boldsymbol{L}\|_1 \qquad (15)$$

$$= \underset{\boldsymbol{L}}{\text{argmin}} \, \|\boldsymbol{L}\|_* + \lambda \|\boldsymbol{L}_0 + (\boldsymbol{S}_0 - \boldsymbol{S}^{(1)}) - \boldsymbol{L}\|_1 \qquad (16)$$

As $\boldsymbol{S}^{(1)}$ is a trimmed version of $\boldsymbol{S}_0$, $(\boldsymbol{S}_0 - \boldsymbol{S}^{(1)})$ must also be a trimmed version of $\boldsymbol{S}_0$. From Theorem 2.2 of [3], we know that Robust PCA with data matrix $\boldsymbol{L}_0 + (\boldsymbol{S}_0 - \boldsymbol{S}^{(1)})$ is exact as well, i.e., $\boldsymbol{L}^{(2)} = \boldsymbol{L}_0$.

$\boldsymbol{L}^{(2)} = \boldsymbol{L}^{(1)}$ implies $\boldsymbol{S}^{(2)} = \boldsymbol{S}^{(1)}$, so SRPCP converges in two iterations and recovers $\boldsymbol{L}_0$ exactly. If additionally $\beta < \frac{1}{\sqrt{n}} \min\{|(\boldsymbol{S}_0)_{i,j}| : (\boldsymbol{S}_0)_{i,j} \neq 0\}$, we must have $\boldsymbol{S}^{(2)} = \boldsymbol{S}^{(1)} = \boldsymbol{S}_0$ according to Step 2 in Table 1. Thus SRPCP recovers both $\boldsymbol{L}_0$ and $\boldsymbol{S}_0$ exactly. □

[1] $\boldsymbol{S}'$ is said [3] to be a trimmed version of $\boldsymbol{S}$ if $supp(\boldsymbol{S}') \subset supp(\boldsymbol{S})$ and $\boldsymbol{S}'_{i,j} = \boldsymbol{S}_{i,j}$ whenever $\boldsymbol{S}'_{i,j} \neq 0$.

4. EMPIRICAL STUDIES

In this section, we first empirically study and compare the proposed method with SPCP and He's implicit regularizer (GAPG_Welsch) [7] that corresponds to Welsch M-estimation on the noisy simulated data. Then we demonstrate the effectiveness of the proposed method on a real application.

Comparison on simulated data: For SPCP, it is solved via Alternating Direction Method with Increasing Penalty (ADMIP) [15]. For GAPG_Welsch, we use the author's algorithm, where the parameter is tuned as the author did to give better results. For our Step 1, we use Augmented Lagrange Multipliers (ALM) [12] to solve it. Our experimental setup is similar to [15], which is as follows:

1. Given the rank r, the low-rank component $\boldsymbol{L}_0$ is built as $\boldsymbol{L}_0 = \boldsymbol{A}\boldsymbol{B}^T$, where $\boldsymbol{A}$ and $\boldsymbol{B}$ are randomly generated $n \times r$ standard Gaussian matrices;

2. Given the fraction ρ (corruption rate) of non-zero entries in $\boldsymbol{S}_0$, the support of $\boldsymbol{S}_0$ is chosen uniformly at random with size $k = \text{round}(\rho n^2)$, and the value of each non-zero entry is independently drawn from a uniform distribution over the interval $[-\sqrt{\frac{8r}{\pi}}, \sqrt{\frac{8r}{\pi}}]$;

3. Each entry of the noise $\boldsymbol{G}$ is independently drawn from a Gaussian distribution with mean 0 and variance σ^2.

4. Finally, generate $\boldsymbol{M} = \boldsymbol{L}_0 + \boldsymbol{S}_0 + \boldsymbol{G}$. Estimate $\boldsymbol{L}_0$ and $\boldsymbol{S}_0$ from $\boldsymbol{M}$ using different methods.

We set $n = 50$, $\sigma = 0.01$, $\lambda = 1/\sqrt{n}$, and $\beta = 5\sigma \times \lambda$ in the experiment. For each $r \in \{10\%n, 16\%n\}$, and each $\rho \in \{0.01 : 0.01 : 0.20\}$, we repeat the above procedure 100 times. For evaluation, the estimated $\hat{\boldsymbol{L}}$ and $\hat{\boldsymbol{S}}$ are compared with ground truth. Besides measuring the Relative Error $\frac{\|\hat{\boldsymbol{L}} - \boldsymbol{L}_0\|_F}{\|\boldsymbol{L}_0\|_F}$ and $\frac{\|\hat{\boldsymbol{S}} - \boldsymbol{S}_0\|_F}{\|\boldsymbol{S}_0\|_F}$, we also compute the distance between the supports of $\hat{\boldsymbol{S}}$ and $\boldsymbol{S}_0$. Denoting the two supports as $\hat{\Omega}$ and Ω, the distance is defined as follows [16]:

$$\text{dist}(\hat{\Omega}, \Omega) = \frac{\max\{|\hat{\Omega}|, |\Omega|\} - |\hat{\Omega} \cap \Omega|}{\max\{|\hat{\Omega}|, |\Omega|\}} \qquad (17)$$

We denote the average of $\text{dist}(\hat{\Omega}, \Omega)$ over Monte Carlo runs as the Probability of Error in Support.

We also compare the rank of $\hat{\boldsymbol{L}}$ with the ground truth, where we automatically truncate the rank of $\hat{\boldsymbol{L}}$ before comparison. More specifically, denoting the singular values of $\hat{\boldsymbol{L}}$ as $\sigma_1, \sigma_2, ..., \sigma_n$ in descending order, for this experimental setup, we find the minimal j such that $\sigma_j / \sigma_{j+1} > 2.5$ and set it as the truncated rank of $\hat{\boldsymbol{L}}$. We denote the empirical probability of not recovering the true rank over Monte Carlo runs as the Probability of Error in Rank.

Fig. 1 shows the comparison between SPCP, GAPG_Welsch and SRPCP. SRPCP, the proposed method, often achieves the best results in terms of Relative Error, support recovery, and rank recovery. GAPG_Welsch fails to recover the rank at higher corruption rates, while SPCP and SRPCP always recover the rank correctly in the experiments.

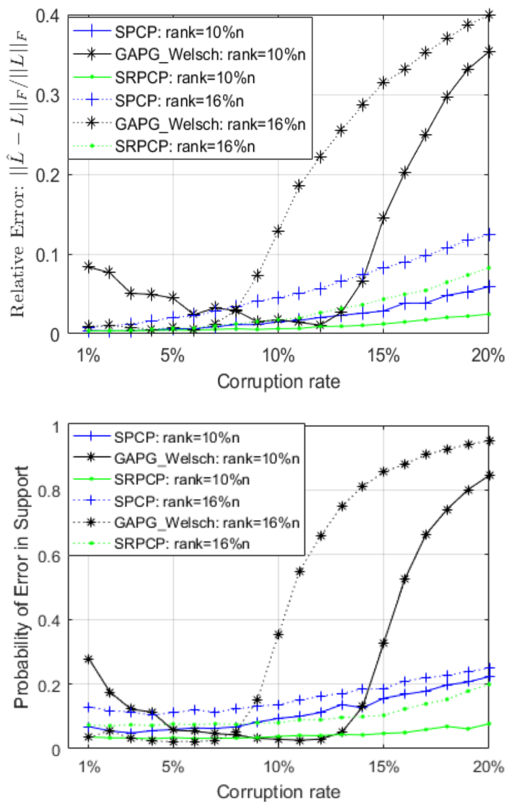


Fig. 1. Average Relative Error of $\hat{\boldsymbol{L}}$ (top) and Probability of Error in Support (bottom).

Parameter sensitivity: Both SPCP and SRPCP need the knowledge of inlier noise level. In the above experiments, we assume we know the standard deviation σ of the inlier noise, and set $\beta = 5\sigma \times \lambda$ for SRPCP. However, in practice, the estimated $\hat{\sigma}$ may be slightly greater or less than the true σ, which is equivalent to setting β slightly greater or less than $5\sigma\lambda$. We repeat the above experiments with rank $r = 10\%n$, setting β to values ranging from $2\sigma\lambda$ to $8\sigma\lambda$ for SRPCP. Fig. 2 shows the Relative Error in $\hat{\boldsymbol{L}}$ for different values of β. SRPCP performance is not sensitive to small variations of β, and is better than SPCP. SRPCP with different β always recovers the true rank in the experiments.

Experiments on real data: We now demonstrate SRPCP on a restaurant surveillance video[2], where each frame is converted to a column vector. The background over the frames is the low-rank component and the moving objects over the frames can be considered to be the sparse component. We run our algorithm with the inexact ALM [12] on the first 200 frames. The recovered rank is one (after truncation), which is reasonable. Fig. 3 shows the recovered background and moving objects as well as noise in a frame. We can see that SRPCP successfully separated the foreground from the background.

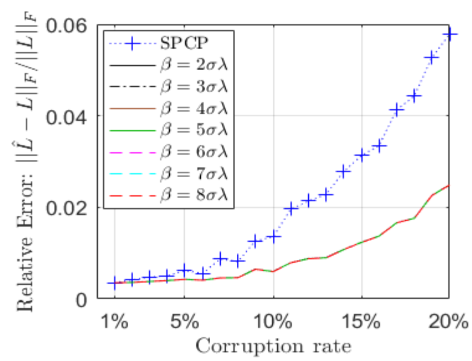
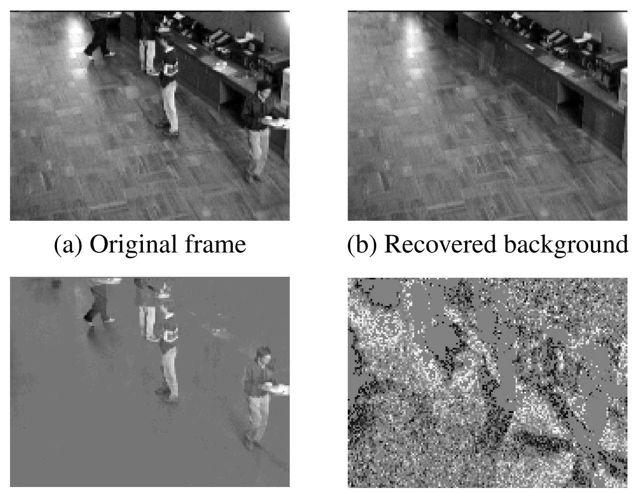


Fig. 2. Average Relative Error of $\hat{\boldsymbol{L}}$.



(a) Original frame (b) Recovered background

(c) Recovered foreground (d) Noise

Fig. 3. SRPCP background and foreground recovery.

5. CONCLUSIONS AND FUTURE WORK

A novel objective function is proposed to recover the low-rank component and sparse component of a possibly noisy matrix. For the case of no dense inlier noise, the proposed method can guarantee the exact recovery with high probability under certain conditions. For the noisy case, simulations and results on real data demonstrate the advantage and effectiveness of the proposed method. The theoretical guarantee for the noisy case is our ongoing work. By analogy with our robust linear regression work, it may be possible to show that SRPCP can handle more non-zero entries in the sparse matrix $\boldsymbol{S}_0$ and have a smaller error bound than SPCP.

[2] http://perception.i2r.a-star.edu.sg/bk_model/bk_index.html

6. REFERENCES

[1] V. Chandrasekaran, S. Sanghavi, P. A. Parrilo, and A. S. Willsky, "Rank-sparsity incoherence for matrix decomposition," *SIAM Journal on Optimization*, vol. 21, no. 2, pp. 572–596, 2011.

[2] J. Wright, Y. Peng, Y. Ma, A. Ganesh, and S. Rao, "Robust principal component analysis: Exact recovery of corrupted low-rank matrices via convex optimization," in *Advances in Neural Information Processing Systems (NIPS)*, 2009, pp. 1–9.

[3] E. J. Candès, X. Li, Y. Ma, and J. Wright, "Robust principal component analysis?," *J. ACM*, vol. 58, no. 3, pp. 11:1–11:37, June 2011.

[4] H. Zhang, Y. Zhou, and Y. Liang, "Analysis of robust PCA via local incoherence," in *Advances in Neural Information Processing Systems (NIPS)*, 2015, pp. 1819–1827.

[5] Z. Zhou, X. Li, J. Wright, E. J. Candès, and Y. Ma, "Stable principal component pursuit," in *2010 IEEE International Symposium on Information Theory*, June 2010, pp. 1518–1522.

[6] D. Hsu, S. M. Kakade, and T. Zhang, "Robust matrix decomposition with sparse corruptions," *IEEE Transactions on Information Theory*, vol. 57, no. 11, pp. 7221–7234, Nov 2011.

[7] R. He, T. Tan, and L. Wang, "Robust recovery of corrupted low-rankmatrix by implicit regularizers," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 36, no. 4, pp. 770–783, April 2014.

[8] T. Zhou and D. Tao, "Godec: Randomized low-rank & sparse matrix decomposition in noisy case," in *International Conference on Machine Learning*, 2011, pp. 33–40.

[9] L. Xiong, X. Chen, and J. Schneider, "Direct robust matrix factorizatoin for anomaly detection," in *2011 IEEE 11th International Conference on Data Mining*, Dec 2011, pp. 844–853.

[10] P. Netrapalli, N. U N, S. Sanghavi, A. Anandkumar, and P. Jain, "Non-convex robust PCA," in *Advances in Neural Information Processing Systems (NIPS)*, 2014, pp. 1107–1115.

[11] M. O. Ulfarsson, V. Solo, and G. Marjanovic, "Sparse and low rank decomposition using ℓ_0 penalty," in *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, April 2015, pp. 3312–3316.

[12] Z. Lin, M. Chen, and Y. Ma, "The Augmented Lagrange Multiplier Method for Exact Recovery of Corrupted Low-Rank Matrices," *arXiv:1009.5055*, p. 23, 2013.

[13] K. C. Toh and S. Yun, "An accelerated proximal gradient algorithm for nuclear norm regularized linear least squares problems," *Pacific Journal of Optimization*, vol. 6, no. 3, pp. 615–640, 2010.

[14] J. Cai, E. J. Candès, and Z. Shen, "A singular value thresholding algorithm for matrix completion," *SIAM Journal on Optimization*, vol. 20, no. 4, pp. 1956–1982, 2010.

[15] N. S. Aybat and G. Iyengar, "An alternating direction method with increasing penalty for stable principal component pursuit," *Computational Optimization and Applications*, vol. 61, no. 3, pp. 635–668, 2015.

[16] M. Elad, *Sparse and Redundant Representations: From Theory to Applications in Signal and Image Processing*, Springer, 1st edition, 2010.