A Unified Framework for Nonconvex Low-Rank plus Sparse Matrix Recovery

Xiao Zhang* Department of Computer Science University of Virginia

Lingxiao Wang* Department of Computer Science University of Virginia

Quanquan Gu Department of Computer Science University of Virginia

Abstract

We propose a unified framework to solve general low-rank plus sparse matrix recovery problems based on matrix factorization, which covers a broad family of objective functions satisfying the restricted strong convexity and smoothness conditions. Based on projected gradient descent and the double thresholding operator, our proposed generic algorithm is guaranteed to converge to the unknown low-rank and sparse matrices at a locally linear rate, while matching the best-known robustness guarantee (i.e., tolerance for sparsity). At the core of our theory is a novel structural Lipschitz gradient condition for low-rank plus sparse matrices, which is essential for proving the linear convergence rate of our algorithm, and we believe is of independent interest to prove fast rates for general superposition-structured models. We illustrate the application of our framework through two concrete examples: robust matrix sensing and robust PCA. Empirical experiments corroborate our theory.

1 INTRODUCTION

Low-rank matrix recovery has received considerable attention in machine learning and high-dimensional statistical inference in the past decades [9, 10, 45, 26, 25, 37, 1, 38, 55, 16, 27, 24, 23, 17, 22, 4, 49, 50, 54]. One important question is whether low-rank matrix estimation algorithms are robust to arbitrarily sparse corruptions, which motivates the problem of low-rank plus sparse matrix recovery, such as robust matrix sensing [51, 30], robust PCA [7, 52, 56], robust covariance matrix estimation [2] and robust multi-task regression [37, 53]. Following this line of research, we consider the general problem of low-rank plus sparse matrix recovery, where the objective is to recover an unknown model parameter matrix that can be decomposed as the sum of a low-rank matrix $\mathbf{X}^* \in \mathbb{R}^{d_1 \times d_2}$ and a sparse matrix $\mathbf{S}^* \in \mathbb{R}^{d_1 \times d_2}$, from a set of n observations generated from the model. More specifically, let $\mathcal{L}_n : \mathbb{R}^{d_1 \times d_2} \to \mathbb{R}$ be the sample loss function derived from some statistical model, which measures the goodness of fit to the observations with respect to any given low-rank matrix $\mathbf{X}$ and sparse matrix $\mathbf{S}$. Then the general low-rank plus sparse matrix recovery problem can be cast into the following nonconvex optimization problem

$$\begin{aligned} &\min_{\mathbf{X},\mathbf{S}\in\mathbb{R}^{d_1\times d_2}} \mathcal{L}_n(\mathbf{X}+\mathbf{S}), \\ &\text{subject to } \mathbf{X}\in\mathcal{C},\ \operatorname{rank}(\mathbf{X})\le r,\ \|\mathbf{S}\|_0\le s, \end{aligned} \tag{1.1}$$

where $\mathcal{C}$ is a constraint set such that $\mathbf{X}^* \in \mathcal{C}$ (see Section 3 for more details), r denotes the rank of $\mathbf{X}^*$, $\|\mathbf{S}\|_0$ denotes the number of nonzero entries in $\mathbf{S}$, and s denotes the number of nonzero entries in $\mathbf{S}^*$.

A long line of research has been proposed to study how to recover the unknown decomposition via convex relaxation [52, 7, 14, 25, 2, 16, 55, 29]. However, convex relaxation based algorithms usually involve a time-consuming singular value decomposition (SVD) step in each iteration, which is computationally very expensive for large scale high-dimensional data. In order to solve low-rank plus sparse matrix recovery problems more efficiently, recent studies [30, 41, 17, 21, 56] proposed to use nonconvex optimization algorithms such as alternating minimization and gradient descent. Although these nonconvex optimization based approaches improve the computational efficiency upon convex relaxation based methods, they still suffer from either unsatisfied robustness guarantee

*Equal Contribution

Proceedings of the 21^{st} International Conference on Artificial Intelligence and Statistics (AISTATS) 2018, Lanzarote, Spain. PMLR: Volume 84.

and/or limitations to specific models.

In this paper, we aim to develop a unified framework to recover both the low-rank and the sparse matrices from generic statistical models. Following [6], we reparameterize the low-rank matrix as the product of two small factor matrices, i.e., $\mathbf{X} = \mathbf{U}\mathbf{V}^\top$ where $\mathbf{U} \in \mathbb{R}^{d_1 \times r}$ and $\mathbf{V} \in \mathbb{R}^{d_2 \times r}$, and propose to solve the following nonconvex optimization problem

$$\begin{aligned} &\min_{\mathbf{U},\mathbf{V},\mathbf{S}} \mathcal{L}_n(\mathbf{U}\mathbf{V}^\top + \mathbf{S}), \\ &\text{subject to } \mathbf{U} \in \mathcal{C}_1,\ \mathbf{V} \in \mathcal{C}_2,\ \|\mathbf{S}\|_0 \le s, \end{aligned} \tag{1.2}$$

where $\mathcal{C}_1 \subseteq \mathbb{R}^{d_1 \times r}, \mathcal{C}_2 \subseteq \mathbb{R}^{d_2 \times r}$ are the corresponding rotation invariant constraint sets induced by $\mathcal{C}$ (see Section 3 for more details). Due to Burer-Monteiro factorization [6], i.e., the reformulation $\mathbf{X} = \mathbf{U}\mathbf{V}^\top$, the rank constraint is automatically satisfied in (1.2), which gets rid of the computationally inefficient SVD step. In order to solve (1.2), we propose a projected gradient descent algorithm, along with a unified theory that integrates both optimization-theoretic and statistical analyses. We further summarize our main contributions as follows:

1. Compared with existing work, our generic framework can be applied to a larger family of loss functions satisfying the restricted strong convexity and smoothness conditions [39, 37, 29]. We demonstrate the superiority of our framework through two concrete examples: robust matrix sensing and robust PCA.

2. The gradient descent phase of our proposed algorithm matches the best-known robustness guarantee $O(1/r)$ [25, 16]. Compared with existing robust PCA algorithms [56, 18], our algorithm achieves improved computational complexity $O\big(r^3 d \log d \log(1/\epsilon)\big)$, while matching the optimal sample complexity $O(r^2 d \log d)$ for Burer-Monteiro factorization-based low-rank matrix recovery [57] under the incoherence condition.

3. To ensure the linear convergence rate, from the algorithmic perspective, we construct a double thresholding operator, which integrates both hard thresholding [5] and truncation operators [56]; in terms of technical proof, we propose a novel *structural Lipschitz gradient condition* for low-rank plus sparse matrices. We believe both the double thresholding operator and the structural Lipschitz gradient condition are of independent interest for other superposition-structured models to prove faster convergence rates.

Notation Denote $[d]$ to be the index set $\{1, \ldots, d\}$. For any matrix $\mathbf{A} \in \mathbb{R}^{d_1 \times d_2}$, let $\mathbf{A}_{i,*}$, $\mathbf{A}_{*,j}$ be the i-th row and the j-th column of $\mathbf{A}$ respectively, and let A_{ij} be its (i,j)-th entry. Let the k-th largest singular value of $\mathbf{A}$ be $\sigma_k(\mathbf{A})$, and let $\mathrm{SVD}_r(\mathbf{A})$ be the rank-r SVD of matrix $\mathbf{A}$. For any d-dimensional vector $\mathbf{x}$, the ℓ_q vector norm of $\mathbf{x}$ is denoted by $\|\mathbf{x}\|_q = (\Sigma_{i=1}^d |x_i|^q)^{1/q}$, where $1 \le q < \infty$, and we use $\|\mathbf{x}\|_0$ to represent the number of nonzero entries of $\mathbf{x}$. For any d_1-by-d_2 matrix $\mathbf{A}$, we use $\|\mathbf{A}\|_2$ and $\|\mathbf{A}\|_F$ to denote the spectral norm and Frobenius norm respectively. And we use $\|\mathbf{A}\|_{\infty,\infty}$ to denote the elementwise infinity norm. In addition, we denote the number of nonzero entries in $\mathbf{A}$ by $\|\mathbf{A}\|_0$, and use $\|\mathbf{A}\|_{2,\infty}$ to represent the largest ℓ_2-norm of its rows. For any two sequences $\{a_n\}$ and $\{b_n\}$, if there exists a constant $0 < C < \infty$ such that $a_n \le C b_n$, then we write $a_n = O(b_n)$.

2 RELATED WORK

In recent years, there has been a large body of literature [13, 51, 15, 7, 14, 25, 30, 2, 16, 55, 29] focusing on the matrix recovery problems with low-rank plus sparse structures. For instance, [51, 30] studied the problem of robust matrix sensing, where they aim to recover both the low-rank matrix and the sparse matrix from compressive measurements. [15] analyzed the robust multi-task learning, where they characterize the task relationships using a low-rank structure, and simultaneously identify the outlier tasks using a sparse structure. The most widely studied low-rank plus sparse matrix recovery problem is robust PCA [7, 14, 25, 16, 29], where the goal is to recover the unknown low-rank matrix from corrupted observations. In the context of robust PCA, [7] proved that under random corruption model, their algorithm enables exact recovery with constant fraction of corruptions. Meanwhile, [14] considered the deterministic corruption model and showed that the tolerance of row/column sparsity for exact recovery is in the order of $O(1/r\sqrt{d})$, which was further improved to $O(1/r)$ by [25, 16]. Instead of considering specific models, unified analysis framework was proposed to cover more general low-rank plus sparse matrix recovery problems. In particular, [2] proposed to analyze a class of estimators for noisy matrix decomposition based on convex optimization with decomposable regualrizer. [55] considered a general class of M-estimators and provided a unified framework for superposition-structured statistical models.

However, most of the aforementioned work are based on convex relaxation, which involves a computationally expensive SVD step in each iteration. To address such computational barrier, various nonconvex optimization algorithms [41, 17, 21, 56, 18] have been carried out to solve low-rank plus sparse matrix recovery with provable guarantees. For example, [41] proposed alternating projection to simultaneously estimate the low-rank and sparse structure, while [17] showed that projected gradient descent based algorithm will linearly converge to the unknown matrix decomposition under suitable initialization procedure. The most re-

lated work to ours is [56], which proposed a fast gradient descent algorithm based on a novel truncation operator to recover the unknown low-rank matrix for robust PCA. Their approach allows for $O(1/r^{1.5})$ sparsity with improved computational efficiency upon previous work. Most recently, [18] further improved the existing work in terms of robustness guarantee from $O(1/r^{1.5})$ to $O(1/r)$. It is worth noting that these several pieces of work are limited to robust PCA, thus unable to deal with more general problem settings, such as robust matrix sensing.

3 THE PROPOSED ALGORITHM

Recall that our objective is to recover both unknown low-rank matrix $\mathbf{X}^* \in \mathbb{R}^{d_1 \times d_2}$ with rank-r and unknown sparse matrix $\mathbf{S}^* \in \mathbb{R}^{d_1 \times d_2}$ with s nonzero entries simultaneously. Let $\overline{\mathbf{U}}^* \mathbf{\Sigma}^* \overline{\mathbf{V}}^{*\top}$ be the SVD of $\mathbf{X}^*$, where $\overline{\mathbf{U}}^*, \overline{\mathbf{V}}^*$ are the left and right singular matrices respectively, and $\mathbf{\Sigma}^*$ denotes a r-by-r diagonal matrix with elements $\sigma_1 \geq \sigma_2 \geq \ldots \geq \sigma_r > 0$. Denote the condition number of $\mathbf{X}^*$ by $\kappa = \sigma_1/\sigma_r$.

Intuitively speaking, in order to distinguish between low-rank and sparse structures, the unknown low-rank matrix $\mathbf{X}^*$ cannot be too sparse. For instance, if $\mathbf{X}^*$ is equal to zero in nearly all elements, the recovery task is impossible unless all of the entries are sampled [19]. Therefore, we impose the following incoherence condition [9] on $\mathbf{X}^*$ to avoid such identifiability issue. More specifically, let the SVD of $\mathbf{X}^*$ be $\mathbf{X}^* = \overline{\mathbf{U}}^* \mathbf{\Sigma}^* \overline{\mathbf{V}}^{*\top}$, then we assume $\mathbf{X}^*$ is α-incoherent

$$\|\overline{\mathbf{U}}^*\|_{2,\infty} \leq \sqrt{\alpha r/d_1} \text{ and } \|\overline{\mathbf{V}}^*\|_{2,\infty} \leq \sqrt{\alpha r/d_2}, \quad (3.1)$$

where $\alpha \geq 1$ denotes the incoherence parameter. Thus, we let the constraint set $\mathcal{C}$ in (1.1) be the set of α-incoherent matrices. In addition, suppose $\mathbf{S}^*$ has at most β-fraction nonzero entries for each row and column [14], or in other words $\mathbf{S}^* \in \mathcal{K}$, where $\mathcal{K}$ is defined as follows

$$\mathcal{K} = \big\{\mathbf{S} \in \mathbb{R}^{d_1 \times d_2} \,\big|\, \|\mathbf{S}\|_0 \leq s,\ \|\mathbf{S}_{i,*}\|_0 \leq \beta d_2, \\ \forall i \in [d_1];\ \|\mathbf{S}_{*,j}\|_0 \leq \beta d_1, \forall j \in [d_2]\big\}.$$

Here, $\beta \in (0,1)$ represents the sparsity tolerance parameter. Recall (1.2), we define two constraint sets $\mathcal{C}_1, \mathcal{C}_2$ for $\mathbf{U}, \mathbf{V}$ respectively. Here, we provide the definitions of $\mathcal{C}_1, \mathcal{C}_2$ as follows

$$\mathcal{C}_1 = \big\{\mathbf{U} \in \mathbb{R}^{d_1 \times r} \big| \|\mathbf{U}\|_{2,\infty} \leq \sqrt{\alpha r/d_1}\|\mathbf{Z}^0\|_2\big\}, \\ \mathcal{C}_2 = \big\{\mathbf{V} \in \mathbb{R}^{d_2 \times r} \big| \|\mathbf{V}\|_{2,\infty} \leq \sqrt{\alpha r/d_2}\|\mathbf{Z}^0\|_2\big\}, \quad (3.2)$$

where $\mathbf{Z}^0 = [\mathbf{U}^0; \mathbf{V}^0]$ represents the initial solution of Algorithm 1, and we will further demonstrate in the theoretical analysis that $\mathbf{U}^* \in \mathcal{C}_1, \mathbf{V}^* \in \mathcal{C}_2$. Furthermore, in order to guarantee the uniqueness of the optimal solution to optimization problem (1.2), following [46, 57, 42], we impose an additional regularizer to penalize the scale difference between $\mathbf{U}$ and $\mathbf{V}$. In other words, we aim to estimate the unknown parameter set $(\mathbf{U}^*, \mathbf{V}^*, \mathbf{S}^*)$ by minimizing the following regularized objective function with constraints

$$F_n(\mathbf{U}, \mathbf{V}, \mathbf{S}) := \mathcal{L}_n(\mathbf{U}\mathbf{V}^\top + \mathbf{S}) + \|\mathbf{U}^\top\mathbf{U} - \mathbf{V}^\top\mathbf{V}\|_F^2/8, \\ \text{subject to } \mathbf{U} \in \mathcal{C}_1, \mathbf{V} \in \mathcal{C}_2, \mathbf{S} \in \mathcal{K}. \quad (3.3)$$

Next, we present our proposed generic gradient descent algorithm for solving (3.3), as displayed in Algorithm 1. For low-rank structure, we perform gradient descent on $\mathbf{U}$ and $\mathbf{V}$ respectively, followed by projection onto the corresponding constraint sets $\mathcal{C}_1$ and $\mathcal{C}_2$. For sparse structure, we perform double thresholding, which integrates both the hard thresholding operator in [5] and the truncation operator in [56], to ensure the output estimator $\mathbf{S}^t$ is sparse and has at most β-fraction nonzero entries per row and column as well.

Algorithm 1 Gradient Descent Phase

Input: Sample loss function $\mathcal{L}_n$; step size τ, η; total number of iterations T; parameters γ, γ'; initial solution $(\mathbf{U}^0, \mathbf{V}^0, \mathbf{S}^0)$.

$\mathbf{Z}^0 = [\mathbf{U}^0; \mathbf{V}^0]$; Let $\mathcal{C}_1, \mathcal{C}_2$ be defined in (3.2).

for: $t = 0, 1, 2, \ldots, T-1$ **do**

$\mathbf{S}^{t+1} = \mathcal{T}_{\gamma\beta} \circ \mathcal{H}_{\gamma' s}\big(\mathbf{S}^t - \tau\nabla_{\mathbf{S}}\mathcal{L}_n(\mathbf{U}^t\mathbf{V}^{t\top} + \mathbf{S}^t)\big)$

$\mathbf{U}^{t+1} = \mathcal{P}_{\mathcal{C}_1}\big(\mathbf{U}^t - \eta\nabla_{\mathbf{U}}\mathcal{L}_n(\mathbf{U}^t\mathbf{V}^{t\top} + \mathbf{S}^t) - \frac{1}{2}\eta\mathbf{U}^t(\mathbf{U}^{t\top}\mathbf{U}^t - \mathbf{V}^{t\top}\mathbf{V}^t)\big)$

$\mathbf{V}^{t+1} = \mathcal{P}_{\mathcal{C}_2}\big(\mathbf{V}^t - \eta\nabla_{\mathbf{V}}\mathcal{L}_n(\mathbf{U}^t\mathbf{V}^{t\top} + \mathbf{S}^t) - \frac{1}{2}\eta\mathbf{V}^t(\mathbf{V}^{t\top}\mathbf{V}^t - \mathbf{U}^{t\top}\mathbf{U}^t)\big)$

end for

Output: $(\mathbf{U}^T, \mathbf{V}^T, \mathbf{S}^T)$

In Algorithm 1, we let $\mathcal{P}_{\mathcal{C}_i}$ be the projection operator onto the constraint set $\mathcal{C}_i$, where $i = 1, 2$. We define $\mathcal{H}_k : \mathbb{R}^{d_1 \times d_2} \to \mathbb{R}^{d_1 \times d_2}$ as the hard thresholding operator, which keeps the largest k elements in terms of absolute value (i.e., magnitude) and sets the remaining entries as 0. In addition, we define $\mathcal{T}_\theta : \mathbb{R}^{d_1 \times d_2} \to \mathbb{R}^{d_1 \times d_2}$ as the truncation operator with parameter $\theta \in (0,1)$ as follows: for all $(i,j) \in [d_1] \times [d_2]$, we have

$$[\mathcal{T}_\theta(\mathbf{S})]_{ij} := \begin{cases} S_{ij}, & \text{if } |S_{ij}| \geq |S_{i,*}^{(\theta d_2)}| \text{ and } |S_{ij}| \geq |S_{*,j}^{(\theta d_1)}|, \\ 0, & \text{otherwise,} \end{cases}$$

where $S_{i,*}^{(k)}$ and $S_{*,j}^{(k)}$ denote the k-th largest magnitude entries of $\mathbf{S}_{i,*}$ and $\mathbf{S}_{*,j}$ respectively.

It will be shown in later analysis that Algorithm 1 is guaranteed to converge to the unknown true parameters $(\mathbf{U}^*, \mathbf{V}^*, \mathbf{S}^*)$, as long as the initial solution

$(\mathbf{U}^0, \mathbf{V}^0, \mathbf{S}^0)$ is close enough to $(\mathbf{U}^*, \mathbf{V}^*, \mathbf{S}^*)$. Therefore, motivated by gradient hard thresholding [5] and singular value projection [26], we propose a novel initialization algorithm in Algorithm 2 to ensure the condition on the initial solutions. Based on singular value projection operator, we add an additional infinity norm constraint for low-rank structure. Specifically, we use $\mathcal{P}_{\lambda',\zeta^*} : \mathbb{R}^{d_1\times d_2} \to \mathbb{R}^{d_1\times d_2}$ to denote the constrained projection operator such that

$$\mathcal{P}_{\lambda',\zeta^*}(\mathbf{X}) = \mathrm{argmin}_{\mathrm{rank}(\mathbf{Y})\le r,\, \|\mathbf{Y}\|_{\infty,\infty}\le\zeta^*} \|\mathbf{Y}-\mathbf{X}\|_F,$$

where ζ^* is defined as $\zeta^* = c_0\alpha r\kappa/\sqrt{d_1 d_2}$, with c_0 as a predetermined upper bound of $\sigma_r(\mathbf{X}^*)$. According to (3.1), we have $\|\mathbf{X}^*\|_{\infty,\infty} \le \|\overline{\mathbf{U}}^*\|_{2,\infty}\cdot\sigma_1(\mathbf{X}^*)\cdot\|\overline{\mathbf{V}}^*\|_{2,\infty} \le \zeta^*$. In practice, we can use Dykstra's alternating projection algorithm [3] to solve the projection operator $\mathcal{P}_{\lambda',\zeta^*}$. According to [33] and [32], the alternating projection achieves a local R-linear convergence rate. In our experiments, we only perform one step alternating projection, which is sufficient to derive the fast convergence rate of Algorithm 1. We believe this alternating projection step is efficient, and will further investigate it theoretically.

Algorithm 2 Initialization Phase

Input: Sample loss function $\mathcal{L}_n$; step size τ', η'; total number of iterations L; parameters λ, λ'.

$\mathbf{X}_0 = \mathbf{S}_0 = \mathbf{0}$

for: $\ell = 0, 1, 2, \ldots, L-1$ **do**

$\mathbf{S}_{\ell+1} = \mathcal{H}_{\lambda s}(\mathbf{S}_\ell - \tau'\nabla_{\mathbf{S}}\mathcal{L}_n(\mathbf{X}_\ell + \mathbf{S}_\ell))$

$\mathbf{X}_{\ell+1} = \mathcal{P}_{\lambda',\zeta^*}(\mathbf{X}_\ell - \eta'\nabla_{\mathbf{X}}\mathcal{L}_n(\mathbf{X}_\ell + \mathbf{S}_\ell))$

end for

$[\overline{\mathbf{U}}^0, \mathbf{\Sigma}^0, \overline{\mathbf{V}}^0] = \mathrm{SVD}_r(\mathbf{X}_L)$

$\mathbf{U}^0 = \overline{\mathbf{U}}^0(\mathbf{\Sigma}^0)^{1/2}, \mathbf{V}^0 = \overline{\mathbf{V}}^0(\mathbf{\Sigma}^0)^{1/2}, \mathbf{S}^0 = \mathbf{S}_L$

Output: $(\mathbf{U}^0, \mathbf{V}^0, \mathbf{S}^0)$

4 MAIN THEORY

Let $\mathbf{U}^* = \overline{\mathbf{U}}^*(\mathbf{\Sigma}^*)^{1/2}$, $\mathbf{V}^* = \overline{\mathbf{V}}^*(\mathbf{\Sigma}^*)^{1/2}$ and $\mathbf{Z}^* = [\mathbf{U}^*; \mathbf{V}^*]$ be the unknown matrix parameters we aim to estimate. Following [28, 46, 57], we introduce the following distance metric.

Definition 4.1. For any $\mathbf{Z} \in \mathbb{R}^{(d_1+d_2)\times r}$, define the distance metric between $\mathbf{Z}$ and $\mathbf{Z}^*$ with respect to the optimal rotation as $d(\mathbf{Z}, \mathbf{Z}^*)$ such that $d(\mathbf{Z}, \mathbf{Z}^*) = \min_{\mathbf{R}\in\mathbb{Q}_r}\|\mathbf{Z} - \mathbf{Z}^*\mathbf{R}\|_F$, where $\mathbb{Q}_r$ denotes the set of r-by-r orthonormal matrices.

Next, we lay out the restricted strong convexity (RSC) and restricted strong smoothness (RSS) conditions [39, 36] regarding $\mathcal{L}_n$. Note that our problem includes both low-rank and sparse structures, thus we assume the restricted strong smoothness and convexity conditions hold for one structure given the other.

Condition 4.2 (Low Rank Structure)**.** For all fixed sparse matrix $\mathbf{S} \in \mathbb{R}^{d_1\times d_2}$ with at most $\widetilde{s}$ nonzero entries, we assume $\mathcal{L}_n$ is restricted strongly convex with parameter μ_1 and restricted strongly smooth with parameter L_1 with respect to the low-rank structure, such that for all matrices $\mathbf{X}_1, \mathbf{X}_2 \in \mathbb{R}^{d_1\times d_2}$ with rank at most $\widetilde{r}$, we have

$$\begin{aligned}\mathcal{L}_n(\mathbf{X}_2+\mathbf{S}) \ge \mathcal{L}_n(\mathbf{X}_1+\mathbf{S}) + \langle\nabla_{\mathbf{X}}\mathcal{L}_n(\mathbf{X}_1+\mathbf{S}), \mathbf{X}_2-\mathbf{X}_1\rangle \\ + (\mu_1/2)\|\mathbf{X}_2-\mathbf{X}_1\|_F^2,\\ \mathcal{L}_n(\mathbf{X}_2+\mathbf{S}) \le \mathcal{L}_n(\mathbf{X}_1+\mathbf{S}) + \langle\nabla_{\mathbf{X}}\mathcal{L}_n(\mathbf{X}_1+\mathbf{S}), \mathbf{X}_2-\mathbf{X}_1\rangle \\ + (L_1/2)\|\mathbf{X}_2-\mathbf{X}_1\|_F^2.\end{aligned}$$

Here, $\widetilde{r}$, $\widetilde{s}$ satisfy $r \le \widetilde{r} \le Cr$ and $s \le \widetilde{s} \le Cs$, where $C \ge 1$ is a universal constant to be determined.

Condition 4.3 (Sparse Structure)**.** For all fixed rank-$\widetilde{r}$ matrix $\mathbf{X} \in \mathbb{R}^{d_1\times d_2}$, we assume $\mathcal{L}_n$ is restricted strongly convex with parameter μ_2 and restricted strongly smooth with parameter L_2 in terms of the sparse structure, such that for all matrices $\mathbf{S}_1, \mathbf{S}_2 \in \mathbb{R}^{d_1\times d_2}$ with at most $\widetilde{s}$ nonzero entries, we have

$$\begin{aligned}\mathcal{L}_n(\mathbf{X}+\mathbf{S}_2) \ge \mathcal{L}_n(\mathbf{X}+\mathbf{S}_1) + \langle\nabla_{\mathbf{S}}\mathcal{L}_n(\mathbf{X}+\mathbf{S}_1), \mathbf{S}_2-\mathbf{S}_1\rangle \\ + (\mu_2/2)\|\mathbf{S}_2-\mathbf{S}_1\|_F^2\\ \mathcal{L}_n(\mathbf{X}+\mathbf{S}_2) \le \mathcal{L}_n(\mathbf{X}+\mathbf{S}_1) + \langle\nabla_{\mathbf{S}}\mathcal{L}_n(\mathbf{X}+\mathbf{S}_1), \mathbf{S}_2-\mathbf{S}_1\rangle \\ + (L_2/2)\|\mathbf{S}_2-\mathbf{S}_1\|_F^2.\end{aligned}$$

Moreover, we propose the following novel structural Lipschitz gradient condition on the interaction term between low-rank and sparse structures.

Condition 4.4 (Structural Lipschitz Gradient)**.** Let $\mathbf{X}^*$,$\mathbf{S}^*$ be the unknown low-rank and sparse matrices respectively. For all low-rank matrices $\mathbf{X} \in \mathbb{R}^{d_1\times d_2}$ with rank at most $\widetilde{r}$ and sparse matrices $\mathbf{S}$ with at most $\widetilde{s}$ nonzero entries, we assume

$$\begin{aligned}|\langle\nabla_{\mathbf{X}}\mathcal{L}_n(\mathbf{X}^*+\mathbf{S}) - \nabla_{\mathbf{X}}\mathcal{L}_n(\mathbf{X}^*+\mathbf{S}^*), \mathbf{X}\rangle - \langle\mathbf{S}-\mathbf{S}^*, \mathbf{X}\rangle| \\ \le K\|\mathbf{X}\|_F\cdot\|\mathbf{S}-\mathbf{S}^*\|_F,\\ |\langle\nabla_{\mathbf{S}}\mathcal{L}_n(\mathbf{X}+\mathbf{S}^*) - \nabla_{\mathbf{S}}\mathcal{L}_n(\mathbf{X}^*+\mathbf{S}^*), \mathbf{S}\rangle - \langle\mathbf{X}-\mathbf{X}^*, \mathbf{S}\rangle| \\ \le K\|\mathbf{X}-\mathbf{X}^*\|_F\cdot\|\mathbf{S}\|_F,\end{aligned}$$

where $K \in (0, 1)$ is the structural Lipschitz gradient parameter depending on r, s, d_1, d_2 and n, which can be a sufficiently small constant, as long as sample size n is large enough.

Roughly speaking, Condition 4.4 defines a variant of Lipschitz continuity on $\nabla\mathcal{L}_n$. Take the first inequality for example, the gradient is taken with respect to the

low-rank structure, while the Lipschitz continuity is with respect to any $\widetilde{s}$-sparse matrix $\mathbf{S}$ and $\mathbf{S}^*$.

Finally, we assume that at $\mathbf{X}^*+\mathbf{S}^*$, the gradient of the sample loss function $\nabla\mathcal{L}_n$ is upper bounded in terms of both matrix spectral and infinity norms.

Condition 4.5. For a given sample size n and tolerance parameter $\delta \in (0,1)$, we let $\epsilon_1(n,\delta)$ and $\epsilon_2(n,\delta)$ be the smallest scalars such that $\|\nabla_{\mathbf{X}}\mathcal{L}_n(\mathbf{X}^*+\mathbf{S}^*)\|_2 \leq \epsilon_1(n,\delta)$ and $\|\nabla_{\mathbf{S}}\mathcal{L}_n(\mathbf{X}^*+\mathbf{S}^*)\|_{\infty,\infty} \leq \epsilon_2(n,\delta)$.

4.1 Results For The Generic Model

Now we provide main results for our proposed algorithms. The following theorem guarantees the linear convergence rate of Algorithm 1 under proper conditions. We introduce the following distance metric to measure the estimation error of the output

$$D(\mathbf{Z},\mathbf{S}) = d^2(\mathbf{Z},\mathbf{Z}^*) + \|\mathbf{S}-\mathbf{S}^*\|_F^2/\sigma_1. \tag{4.1}$$

The parameter $1/\sigma_1$ comes from the scale difference between $\mathbf{X} = \mathbf{U}\mathbf{V}^\top$ and $\mathbf{Z} = [\mathbf{U};\mathbf{V}]$, or specifically, $\|\mathbf{X}-\mathbf{X}^*\|_F^2 \leq c\sigma_1 d^2(\mathbf{Z},\mathbf{Z}^*)$ for some constant c.

Theorem 4.6. Let $\mathbf{X}^* = \mathbf{U}^*\mathbf{V}^{*\top}$ be the unknown rank-r matrix that satisfies (3.1) and $\mathbf{S}^*$ be the unknown s-sparse matrix with at most β-fraction nonzero entries per row/column. Suppose the sample loss function $\mathcal{L}_n$ satisfies Conditions 4.2 - 4.5. There exist constants c_1, c_2, c_3, c_4 such that if set step size $\eta = c_1/\sigma_1, \tau = c_2/L_2$ and γ, γ' large enough, under condition $\beta \leq c_3/(\alpha r\kappa)$, for any initial estimator $(\mathbf{Z}^0, \mathbf{S}^0)$ satisfying $D(\mathbf{Z}^0,\mathbf{S}^0) \leq c_4^2\sigma_r$, with probability at least $1-\delta$, the t-th iterate of Algorithm 1 satisfies

$$D(\mathbf{Z}^t,\mathbf{S}^t) \leq \rho^t D(\mathbf{Z}^0,\mathbf{S}^0) + \frac{\Gamma_1' r\epsilon_1^2(n,\delta) + \Gamma_2' s\epsilon_2^2(\epsilon,\delta)}{(1-\rho)\sigma_1},$$

where $D(\mathbf{Z},\mathbf{S})$ is defined in (4.1), and $\rho = \max\{1-\eta\mu_1\sigma_r/80, 1-\mu_2\tau/32\} \in (0,1)$ denotes the contraction parameter, provided that the sample size n is large enough such that the structural Lipschitz parameter K is sufficiently small and $\Gamma_1' r\epsilon_1^2(n,\delta) + \Gamma_2' s\epsilon_2^2(n,\delta) \leq (1-\rho)c_2^2\sigma_1\sigma_r$. Here, Γ_1', Γ_2' are absolute constants depending on μ_1, μ_2, L_1, L_2, γ and γ'.

Remark 4.7. Theorem 4.6 establishes the linear convergence rate of Algorithm 1. The right hand side of the contraction inequality consists of two terms: The first term corresponds to the optimization error, while the other term represents the statistical error. When considering the noiseless case, only the optimization error term exists. It is worth noting that our robustness guarantee required for the gradient descent phase matches the best-known results $O(1/r)$ in [25, 16, 18].

The next theorem provides the theoretical guarantee of Algorithm 2 regarding the initialization.

Theorem 4.8 (Initialization)**.** Under the same condition as in Theorem 4.6, suppose $L_1/\mu_1 \in (1,6)$, $L_2/\mu_2 \in (1,4/3)$, $\mu_1 \geq 1/3$ and $K \leq c\cdot\min\{\mu_1,\mu_2\}$, where c is a small constant. For any $\ell \geq 0$, with step size $\eta' = 1/(6\mu_1)$, $\tau' = 3/(4\mu_2)$ and λ, λ' sufficient large, the ℓ-th iterate of Algorithm 2 satisfies

$$\begin{aligned}\|\mathbf{X}_\ell - \mathbf{X}^*\|_F + \|\mathbf{S}_\ell - \mathbf{S}^*\|_F &\leq \rho'^{\ell}(\|\mathbf{X}^*\|_F + \|\mathbf{S}^*\|_F) \\ &+ \Gamma_1\sqrt{r}\epsilon_1(n,\delta) + \Gamma_2\sqrt{s}\epsilon_2(n,\delta) + \Gamma_3\frac{c_0\alpha r\kappa\sqrt{s}}{\sqrt{d_1 d_2}}\end{aligned} \tag{4.2}$$

with probability at least $1-\delta$, where $\rho' = \max\{\rho_1', \rho_2'\} \in (0, 19/20)$ with $\rho_1' = \big(1+2/\sqrt{\lambda'-1}\big)\cdot\big(\sqrt{1-\mu_1\eta'} + \tau' K\big)$ and $\rho_2' = \big(1+2/\sqrt{\lambda-1}\big)\cdot\big(\sqrt{1-\mu_2\tau'} + \eta'(1+K)\big)$. Here, Γ_1, Γ_2 and Γ_3 are absolute constants depending on $\mu_1, \mu_2, \lambda, \lambda', r$ and s.

Combined both Theorem 4.6 and Theorem 4.8, we arrive at the following main result regarding our method.

Theorem 4.9. Suppose the rank-r matrix $\mathbf{X}^*$ satisfies (3.1) and the s-sparse matrix $\mathbf{S}^*$ has at most β-fraction nonzero entries per row/column. Assume the sample loss function $\mathcal{L}_n$ satisfies Conditions 4.2 - 4.5. There exist constants c_1, c_2, c_3, c_4, c_5, provided that $\beta \leq c_1/(\alpha r\kappa)$, $s \leq c_2 d_1 d_2/(\alpha^2 r^2\kappa^2)$ and the sample size n large enough, if perform $L = O(1)$ iterations in Algorithm 2 with step size $\eta' = 1/(6\mu_1), \tau' = 3/(4\mu_2)$ and parameters λ, λ' large enough, the output of Algorithm 1, with step size $\eta = c_3/\sigma_1, \tau = c_4/L_2$ and parameters γ, γ' large enough, satisfies

$$D(\mathbf{Z}^T,\mathbf{S}^T) \leq \rho^T\cdot c_5\sigma_r + \Gamma\cdot\frac{r\epsilon_1^2(n,\delta) + s\epsilon_2^2(\epsilon,\delta)}{(1-\rho)\sigma_1}$$

with probability at least $1-\delta$, where $\mathbf{Z}^T = [\mathbf{U}^T;\mathbf{V}^T]$, ρ denotes the contraction parameter defined in Theorem 4.6, and Γ is an absolute constant depending on μ_1, μ_2, L_1, L_2, γ and γ'.

Remark 4.10. In Theorem 4.9, we require the tolerance of overall sparsity for $\mathbf{S}^*$ is in the order of $O(d_1 d_2/r^2)$, which is near optimal compared with existing work regarding robust PCA. This suboptimality is due to the more general settings we considered in this work. Specifically, we aim to derive the recovery results for both low-rank and sparse structures, which is applicable for more general loss function beyond robust PCA, such as robust matrix sensing.

4.2 Results For Specific Models

Our main result for the generic model can be readily applied to specific models. In the following discussions, we assume $d_1 = d_2 = d$ for simplicity.

Robust Matrix Sensing. The problem of robust matrix sensing [51, 30] has a broad range of applications in video recovery [11] and hyperspectral imaging [12]. Specifically, we observe $\boldsymbol{y} = \mathcal{A}(\mathbf{X}^*+\mathbf{S}^*)+\boldsymbol{\epsilon}$, where $\mathbf{X}^*, \mathbf{S}^*$ are the unknown low-rank and sparse matrices respectively, and $\boldsymbol{\epsilon}$ denotes the noise vector. Let $\mathcal{A} : \mathbb{R}^{d_1\times d_2} \to \mathbb{R}^n$ be a linear measurement operator such that $\mathcal{A}(\mathbf{X}^*+\mathbf{S}^*) = (\langle \mathbf{A}_1, \mathbf{X}^*+\mathbf{S}^*\rangle, \ldots, \langle \mathbf{A}_n, \mathbf{X}^*+\mathbf{S}^*\rangle)^\top$, where each random matrix $\mathbf{A}_i \in \mathbb{R}^{d_1\times d_2}$ is called sensing matrix, whose entries follow i.i.d. standard normal distribution. In the following discussions, we call $\mathcal{A}$ the standard normal linear operator for simplicity. Thus the sample loss function derived from robust matrix sensing is

$$\mathcal{L}_n(\mathbf{U}\mathbf{V}^\top + \mathbf{S}) := (2n)^{-1}\|\mathbf{y} - \mathcal{A}_n(\mathbf{U}\mathbf{V}^\top + \mathbf{S})\|_2^2.$$

Next, we present the theoretical guarantee of our proposed algorithm for robust matrix sensing.

Corollary 4.11. Suppose $\mathbf{X}^*$, $\mathbf{S}^*$ and $\mathcal{L}_n$ satisfy the same conditions as in Theorem 4.9. Consider robust matrix sensing with standard normal linear operator $\mathcal{A}$ and noise vector $\boldsymbol{\epsilon}$, whose entries follow i.i.d. sub-Gaussian distribution with parameter ν. There exist constants $\{c_i\}_{i=1}^{10}$ such that under condition that sample size $n \geq c_1(rd + s)\log d$, robustness guarantee $\beta \leq 1/(c_2 r\kappa)$ and $s \leq c_3 d_1 d_2/(\alpha^2 r^2\kappa^2)$, if we perform $L = O(1)$ iterations in Algorithm 2 with appropriate step size η', τ' and parameters λ, λ' large enough, then with probability at least $1 - c_4/d$, the output of Algorithm 1, with $\eta = c_5/\sigma_1$, $\tau = c_6$ and γ, γ' large enough, satisfies

$$D(\mathbf{Z}^T, \mathbf{S}^T) \leq \rho^T D(\mathbf{Z}^0, \mathbf{S}^0) + c_7\nu^2\frac{rd}{n} + c_8\nu^2\frac{s\log d}{n},$$

where $\rho = \max\{1 - c_9\eta\sigma_r, 1 - c_{10}\tau\}$.

Remark 4.12. According to Corollary 4.11, in the noiseless setting, our algorithm can achieve exactly recovery for both low-rank and sparse matrices. In addition, to establish the structural Lipschitz gradient condition, we require the sample size $n = O\big((rd+s)\log d\big)$. If $s \leq rd$, it achieves the optimal sample complexity as that of standard matrix sensing [45, 46, 49] up to a logarithmic term. In the noisy setting, after $O\big(\kappa\log\big(n/(rd + s\log d)\big)\big)$ number of iterations, our estimator achieves $O\big((rd + s\log d)/n\big)$ statistical error. The term $O(rd/n)$ corresponds to the statistical error for the low-rank matrix recovery, which matches the minimax lower bound of standard noisy matrix sensing [37]. The other term $O(s\log d/n)$ corresponds to the statistical error for the sparse matrix recovery, which also matches the minimax lower bound of sparse matrix regression [44]. We notice that [51] studied the same problem using a greedy algorithm. However, there is no theoretical guarantee of their algorithm.

Robust PCA We proceed to consider robust PCA. More specifically, we observe a data matrix $\mathbf{Y} \in \mathbb{R}^{d_1\times d_2}$ such that $\mathbf{Y} = \mathbf{X}^*+\mathbf{S}^*$, where $\mathbf{X}^*, \mathbf{S}^* \in \mathbb{R}^{d_1\times d_2}$ are the unknown low-rank and sparse matrices. We consider the uniform observation model

$$Y_{jk} := \begin{cases} X^*_{jk} + S^*_{jk} + E_{jk}, & \text{for any}(j,k) \in \Omega \\ *, & \text{otherwise,} \end{cases}$$

where $\Omega \subseteq [d_1] \times [d_2]$ denotes the observed index set such that for any $(j,k) \in \Omega$, $j \sim \text{uniform}([d_1])$ and $k \sim \text{uniform}([d_2])$. Here $\mathbf{E} \in \mathbb{R}^{d_1\times d_2}$ is the noise matrix, where each entry of $\mathbf{E}$ follows i.i.d. normal distribution with variance $\nu^2/(d_1 d_2)$, resulting in a dimension-free signal-to-noise ratio. In addition, we assume that $\mathbf{S}^*$ is not restrictive to Ω. Therefore, for robust PCA, we have the following objective loss function

$$\mathcal{L}_\Omega(\mathbf{U}\mathbf{V}^\top + \mathbf{S}) := (2p)^{-1}\textstyle\sum_{(j,k)\in\Omega}(\mathbf{U}_{j*}\mathbf{V}_{k*}^\top + S_{jk} - Y_{jk})^2.$$

In the following discussions, we are going to consider both full observation model ($p = 1$) and partial observation model ($0 < p < 1$) for robust PCA.

Corollary 4.13 (Fully Observed RPCA)**.** Suppose $\mathbf{X}^*$, $\mathbf{S}^*$ and $\mathcal{L}_n$ satisfy the same conditions as in Theorem 4.9. There exist constants $\{c_i\}_{i=1}^{9}$ such that under the robustness guarantee $\beta \leq 1/(c_1 r\kappa)$ and $s \leq c_2 d_1 d_2/(\alpha^2 r^2\kappa^2)$, if we perform $L = O(1)$ iterations in Algorithm 2 with appropriate step size η', τ' and λ, λ' large enough, then with probability at least $1 - c_3/d$, the output of Algorithm 1, with step size $\eta = c_4/\sigma_1$,$\tau = c_5$ and γ, γ' large enough, satisfies

$$D(\mathbf{Z}^T, \mathbf{S}^T) \leq \rho^T D(\mathbf{Z}^0, \mathbf{S}^0) + c_6\nu^2\frac{rd}{d_1 d_2} + c_7\nu^2\frac{s\log d}{d_1 d_2},$$

where $\rho = \max\{1 - c_8\eta\sigma_r, 1 - c_9\tau\}$.

Remark 4.14. Corollary 4.13 suggests that in the noiseless setting, the statistical error terms equal to zero. Therefore, our algorithm can exactly recover both low-rank and sparse matrices. Note that [2] also analyzed this model using M-estimators. However, their results include an additional standardized error term $\widetilde{\alpha}^2 s/(d_1 d_2)$, where $\widetilde{\alpha}$ is the maximum magnitude among entries of $\mathbf{X}^*$.

For the partially observed robust PCA, we further impose an infinity norm constraint for $\mathbf{S}^*$ such that $\|\mathbf{S}^*\|_{\infty,\infty} \leq \alpha_1/\sqrt{d_1 d_2}$, to ensure the statistical guarantee for the sparse structure. Note that this condition is essential for sparse recovery as illustrated in [29].

Corollary 4.15 (Partially Observed RPCA)**.** Consider partially observed robust PCA under uniform sampling model. Suppose $\mathbf{X}^*$, $\mathbf{S}^*$ and $\mathcal{L}_n$ satisfy the same conditions as in Theorem 4.9. There exist constants $\{c_i\}_{i=1}^{10}$ such that under the robustness

Table 1: Complexity comparisons among different algorithms for robust PCA under partially observed model.

Algorithm	Sample Complexity	Computational Complexity
Fast RPCA [56]	$O\Big(r^2 d \log d\Big)$	$O\Big(r^4 d \log d \log(1/\epsilon)\Big)$
PG-RMC [18]	$O\Big(r^2 d \log^2 d \log^2(\sigma_1/\epsilon)\Big)$	$O\Big(r^3 d \log^2 d \log^2(\sigma_1/\epsilon)\Big)$
This paper	$O\Big(r^2 d \log d\Big)$	$O\Big(r^3 d \log d \log(1/\epsilon)\Big)$

$\beta \leq 1/(c_1 r\kappa)$, $s \leq c_2 d_1 d_2/(\alpha^2 r^2 \kappa^2)$, and sample size $n \geq c_3(r^2 d + s)\log d$, if we perform $L = O(1)$ number of iterations in Algorithm 2 with appropriate step size η', τ' and λ, λ' large enough, then with probability at least $1 - c_4 d$, the output of Algorithm 1, with step size $\eta = c_5/\sigma_1$,$\tau = c_6$ and γ, γ' large enough, satisfies

$$D(\mathbf{Z}^T, \mathbf{S}^T) \leq \rho^T D(\mathbf{Z}^0, \mathbf{S}^0) + c_7 \max\{\nu^2, \alpha^2 r\}\frac{rd\log d}{n} + c_8 \max\{\alpha_1^2, \nu^2\}\frac{s\log d}{n} + \frac{\alpha_1^2 s}{d_1 d_2}, \quad (4.3)$$

where $\rho = \max\{1 - c_9\eta\sigma_r, 1 - c_{10}\tau\}$, and $\alpha_1 = \sqrt{d_1 d_2}\|\mathbf{S}^*\|_{\infty,\infty}$.

Remark 4.16. Note that the extra fourth term $\alpha_1^2 s/(d_1 d_2)$, on the right hand side of (4.3), is due to the unobserved corruption entries but is in fact dominated by the third term. Corollary 4.15 suggests that, after $O\big(\kappa \log\big(n/\big((r^2 d + s)\log d\big)\big)\big)$ number of iterations, the output of our algorithm achieves $O\big((r^2 d + s)\log d/n\big)$ statistical error, and the term $O(r^2 d \log d/n)$ denotes the statistical error for the low-rank matrix. The term $O(s \log d/n)$ corresponds to the statistical error for the sparse matrix, which matches the minimax lower bound [44]. Moreover, compared with existing nonconvex robust PCA algorithms [56, 18], our algorithm achieves better computational complexity while matching the best-known sample complexity provided that $s \leq r^2 d$. The detailed comparisons are summarized in Table 1.

5 EXPERIMENTS

In this section, we illustrate our experimental results to further demonstrate the performance of our proposed algorithm. In particular, we investigate robust matrix sensing and robust PCA on synthetic data. For robust matrix sensing, we compare our algorithm with SpaRCS [51]. For robust PCA, we compare our algorithm with several state-of-the-art algorithms, including NcRPCA [41], Fast RPCA [56], and PG-RMC [18]. Note that all the experimental results are based on the optimal parameters, which are selected by cross validation, and averaged over 30 trials. In addition, we also compare our algorithm with several existing robust PCA algorithms, including GoDec [58], Alt RPCA [20], and Fast RPCA [56], on real-world data.

Robust Matrix Sensing. Our data are generated from the model $\mathbf{y} = \mathcal{A}(\mathbf{X}^* + \mathbf{S}^*) + \boldsymbol{\epsilon}$. We generate $\mathbf{X}^* \in \mathbb{R}^{d_1 \times d_2}$ via $\mathbf{X}^* = \mathbf{U}^* \mathbf{V}^{*\top}$, where each entry of $\mathbf{U}^* \in \mathbb{R}^{d_1 \times r}$ and $\mathbf{V}^* \in \mathbb{R}^{d_2 \times r}$ is generated independently from standard Gaussian distribution. Besides, we generate the unknown sparse matrix $\mathbf{S}^*$ with each element sampled from Bernoulli distribution with parameter $1 - \beta$, where β is the corruption parameter. The value of each nonzero element of $\mathbf{S}^*$ is drawn uniformly from $[-\alpha, \alpha]$. And each element of the sensing matrix $\mathbf{A}_i$ is drawn from i.i.d. standard normal distribution. For the noisy setting, we consider ϵ_i follows i.i.d. zero mean normal distribution with variance ν^2.

For robust matrix sensing, we study the following experimental settings: (i) $d_1 = d_2 = 100, r = 3$; (ii) $d_1 = d_2 = 150, r = 4$; (iii) $d_1 = d_2 = 200, r = 5$. Furthermore, we consider the noiseless case, choose $\alpha = r, \beta = 0.1$, and set the the number of observation $n = 0.2 * d_1 d_2$. We report the relative error and its standard deviation of low-rank structure ($\|\widehat{\mathbf{X}} - \mathbf{X}^*\|_F / \|\mathbf{X}^*\|_F$) as well as CPU time for different algorithms in Table 2. Note that we didn't show the results of sparse structure since it has similar performance to low-rank structure. The results show that our proposed algorithm outperforms the baseline algorithms in terms of relative error and CPU time.

Robust PCA. We generate the data according to $\mathbf{Y} = \mathbf{X}^* + \mathbf{S}^* + \mathbf{E}$, where the matrices $\mathbf{X}^*, \mathbf{S}^* \in \mathbb{R}^{d_1 \times d_2}$ are generated by the same procedures as in robust matrix sensing. In the noisy setting, each element of the noisy matrix $\mathbf{E} \in \mathbb{R}^{d_1 \times d_2}$ is drawn from i.i.d. zero mean Gaussian distribution with variance ν^2.

For robust PCA, we study the following experimental settings: (i) $d_1 = d_2 = 100, r = 3$; (ii) $d_1 = d_2 = 1000, r = 20$; (iii) $d_1 = d_2 = 5000, r = 50$. In addition, we consider the noiseless case and choose $\alpha = r, \beta = 0.1$. Note that all the experimental results are based on the optimal parameters, which are selected by cross validation, and averaged over 30 trials. We report the averaged root mean square error

Table 2: Experimental results for robust matrix sensing in terms of relative error ($\times 10^{-3}$) and CPU time.

	$d_1 = d_2 = 100, r = 3$		$d_1 = d_2 = 150, r = 4$		$d_1 = d_2 = 200, r = 5$	
Methods	Error	Time (s)	Error	Time (s)	Error	Time (s)
SpaRCS	28.6 (1.24)	30.21	26.83 (1.18)	107.71	25.73 (1.47)	275.40
Ours	5.51 (0.60)	24.31	5.33 (0.57)	63.05	4.75 (0.62)	177.24

Table 3: Experimental results for robust PCA in terms of RMSE ($\times 10^{-3}$) and CPU time.

	$d_1 = d_2 = 100, r = 5$		$d_1 = d_2 = 1000, r = 20$		$d_1 = d_2 = 5000, r = 50$	
Methods	RMSE	Time (s)	RMSE	Time (s)	RMSE	Time (s)
NcRPCA	5.12 (1.84)	0.164	4.39 (2.27)	2.12	5.48 (1.44)	61.78
Fast RPCA	4.67 (0.22)	0.179	4.25 (0.47)	1.86	4.78 (0.39)	43.16
PG-RMC	5.45 (2.15)	0.185	3.97 (1.27)	3.23	6.88 (1.06)	89.29
Ours	3.97 (0.16)	0.121	3.74 (0.15)	1.54	3.67 (0.17)	35.72

(a) Original (b) GoDec (c) Alt RPCA (d) Fast RPCA (e) Ours

Figure 1: Background reconstruction of *Hall of a business building* video. (a) The original frame. (b)-(e) Background frames estimated by GoDec [58], Alt RPCA [20], Fast RPCA [56], and our algorithm respectively.

(RMSE) and its standard deviation of low-rank structure ($\|\widehat{\mathbf{X}} - \mathbf{X}^*\|_F / \sqrt{d_1 d_2}$) as well as CPU time for different algorithms in Table 3. Note that we didn't show the RMSE of sparse structure since it has similar results to low-rank structure. The results show that all the algorithms perform well in terms of RMSE. However, our algorithm outperforms the baseline algorithms in terms of CPU time, especially when the dimension is large, which aligns well with our theory.

Real-world Data. We evaluate our proposed method through the problem of background modeling [34]. The goal of background modeling is to reveal the correlation between video frames, reconstruct the static background and detect moving objects in foreground. More specifically, a video sequence has a low-rank plus sparse structure, because backgrounds of all frames are related, while the moving objects in foregrounds are sparse and independent. Due to this superstructure property, robust PCA has been widely used for background modeling [58, 20, 56]. We apply our proposed method to one surveillance video [34], which includes 200 frames with the resolution 144×176. In particular, we convert each frame to a vector and form a 25344×200 data matrix $\mathbf{Y}$. Figure 1 illustrates the estimated background frames (i.e., low-rank structure) by different methods. The background frames estimated by our method are comparable to others. However, compared with GoDec (taking about 32 seconds), Alt RPCA (taking about 22 seconds), and Fast RPCA (taking about 26 seconds), our proposed method only takes around 18 seconds to process the video sequence. All of these experimental results demonstrate the superiority of our proposed method.

6 CONCLUSIONS

We proposed a nonconvex optimization framework for low-rank plus sparse matrix recovery, which integrates both optimization-theoretic and statistical analyses. However, there still exist some open problems along this line of research, e.g., (1) How to achieve $O(1/r)$ robustness guarantee for the initialization phase targeted for general loss functions? (2) How to improve the sample complexity from $O(r^2 d \log d)$ to $O(rd \log d)$ for robust PCA based on nonconvex optimization?

Acknowledgments

We would like to thank the anonymous reviewers for their helpful comments. This research was sponsored in part by the National Science Foundation IIS-1618948 and IIS-1652539. The views and conclusions contained in this paper are those of the authors and should not be interpreted as representing any funding agencies.

References

[1] Alekh Agarwal, Sahand Negahban, and Martin J Wainwright. Fast global convergence of gradient methods for high-dimensional statistical recovery. *The Annals of Statistics*, 40(5):2452–2482, 2012.

[2] Alekh Agarwal, Sahand Negahban, and Martin J Wainwright. Noisy matrix decomposition via convex relaxation: Optimal rates in high dimensions. *The Annals of Statistics*, pages 1171–1197, 2012.

[3] Heinz H Bauschke and Jonathan M Borwein. Dykstra s alternating projection algorithm for two sets. *Journal of Approximation Theory*, 79(3):418–443, 1994.

[4] Srinadh Bhojanapalli, Behnam Neyshabur, and Nathan Srebro. Global optimality of local search for low rank matrix recovery. *arXiv preprint arXiv:1605.07221*, 2016.

[5] Thomas Blumensath and Mike E Davies. Iterative hard thresholding for compressed sensing. *Applied and Computational Harmonic Analysis*, 27(3):265–274, 2009.

[6] Samuel Burer and Renato DC Monteiro. A nonlinear programming algorithm for solving semidefinite programs via low-rank factorization. *Mathematical Programming*, 95(2):329–357, 2003.

[7] Emmanuel J Candès, Xiaodong Li, Yi Ma, and John Wright. Robust principal component analysis? *Journal of the ACM (JACM)*, 58(3):11, 2011.

[8] Emmanuel J Candes and Yaniv Plan. Tight oracle inequalities for low-rank matrix recovery from a minimal number of noisy random measurements. *IEEE Transactions on Information Theory*, 57(4):2342–2359, 2011.

[9] Emmanuel J Candès and Benjamin Recht. Exact matrix completion via convex optimization. *Foundations of Computational mathematics*, 9(6):717–772, 2009.

[10] Emmanuel J Candès and Terence Tao. The power of convex relaxation: Near-optimal matrix completion. *Information Theory, IEEE Transactions on*, 56(5):2053–2080, 2010.

[11] Volkan Cevher, Aswin Sankaranarayanan, Marco Duarte, Dikpal Reddy, Richard Baraniuk, and Rama Chellappa. Compressive sensing for background subtraction. *Computer Vision–ECCV 2008*, pages 155–168, 2008.

[12] Ayan Chakrabarti and Todd Zickler. Statistics of real-world hyperspectral images. In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pages 193–200. IEEE, 2011.

[13] Venkat Chandrasekaran, Pablo A Parrilo, and Alan S Willsky. Latent variable graphical model selection via convex optimization. In *Communication, Control, and Computing (Allerton), 2010 48th Annual Allerton Conference on*, pages 1610–1613. IEEE, 2010.

[14] Venkat Chandrasekaran, Sujay Sanghavi, Pablo A Parrilo, and Alan S Willsky. Rank-sparsity incoherence for matrix decomposition. *SIAM Journal on Optimization*, 21(2):572–596, 2011.

[15] Jianhui Chen, Jiayu Zhou, and Jieping Ye. Integrating low-rank and group-sparse structures for robust multi-task learning. In *KDD*, pages 42–50. ACM, 2011.

[16] Yudong Chen, Ali Jalali, Sujay Sanghavi, and Constantine Caramanis. Low-rank matrix recovery from errors and erasures. *IEEE Transactions on Information Theory*, 59(7):4324–4337, 2013.

[17] Yudong Chen and Martin J Wainwright. Fast low-rank estimation by projected gradient descent: General statistical and algorithmic guarantees. *arXiv preprint arXiv:1509.03025*, 2015.

[18] Yeshwanth Cherapanamjeri, Kartik Gupta, and Prateek Jain. Nearly-optimal robust matrix completion. *arXiv preprint arXiv:1606.07315*, 2016.

[19] David Gross. Recovering low-rank matrices from few coefficients in any basis. *IEEE Transactions on Information Theory*, 57(3):1548–1566, 2011.

[20] Qilong Gu and Arindam Banerjee. High dimensional structured superposition models. In *Advances In Neural Information Processing Systems*, pages 3684–3692, 2016.

[21] Quanquan Gu, Zhaoran Wang, and Han Liu. Low-rank and sparse structure pursuit via alternating minimization. In *Proceedings of the 19th International Conference on Artificial Intelligence and Statistics*, pages 600–609, 2016.

[22] Huan Gui and Quanquan Gu. Towards faster rates and oracle property for low-rank matrix estimation. *arXiv preprint arXiv:1505.04780*, 2015.

[23] Moritz Hardt, Raghu Meka, Prasad Raghavendra, and Benjamin Weitz. Computational limits for matrix completion. In *COLT*, pages 703–725, 2014.

[24] Moritz Hardt and Mary Wootters. Fast matrix completion without the condition number. In *COLT*, pages 638–678, 2014.

[25] Daniel Hsu, Sham M Kakade, and Tong Zhang. Robust matrix decomposition with sparse corruptions. *IEEE Transactions on Information Theory*, 57(11):7221–7234, 2011.

[26] Prateek Jain, Raghu Meka, and Inderjit S Dhillon. Guaranteed rank minimization via singular value projection. In *Advances in Neural Information Processing Systems*, pages 937–945, 2010.

[27] Prateek Jain and Praneeth Netrapalli. Fast exact matrix completion with finite samples. *arXiv preprint*, 2014.

[28] Prateek Jain, Praneeth Netrapalli, and Sujay Sanghavi. Low-rank matrix completion using alternating minimization. In *STOC*, pages 665–674, 2013.

[29] Olga Klopp, Karim Lounici, and Alexandre B Tsybakov. Robust matrix completion. *Probability Theory and Related Fields*, pages 1–42, 2014.

[30] Anastasios Kyrillidis and Volkan Cevher. Matrix alps: Accelerated low rank and sparse matrix reconstruction. In *Statistical Signal Processing Workshop (SSP), 2012 IEEE*, pages 185–188. IEEE, 2012.

[31] Michel Ledoux. *The concentration of measure phenomenon*. Number 89. American Mathematical Soc., 2005.

[32] Adrian S Lewis, David Russel Luke, and Jérôme Malick. Local linear convergence for alternating and averaged nonconvex projections. *Foundations of Computational Mathematics*, 9(4):485–513, 2009.

[33] Adrian S Lewis and Jérôme Malick. Alternating projections on manifolds. *Mathematics of Operations Research*, 33(1):216–234, 2008.

[34] Liyuan Li, Weimin Huang, Irene Yu-Hua Gu, and Qi Tian. Statistical modeling of complex backgrounds for foreground object detection. *IEEE Transactions on Image Processing*, 13(11):1459–1472, 2004.

[35] Xingguo Li, Raman Arora, Han Liu, Jarvis Haupt, and Tuo Zhao. Nonconvex sparse learning via stochastic optimization with progressive variance reduction. *arXiv preprint arXiv:1605.02711*, 2016.

[36] Po-Ling Loh and Martin J Wainwright. Regularized m-estimators with nonconvexity: Statistical and algorithmic theory for local optima. In *Advances in Neural Information Processing Systems*, pages 476–484, 2013.

[37] Sahand Negahban and Martin J Wainwright. Estimation of (near) low-rank matrices with noise and high-dimensional scaling. *The Annals of Statistics*, pages 1069–1097, 2011.

[38] Sahand Negahban and Martin J Wainwright. Restricted strong convexity and weighted matrix completion: Optimal bounds with noise. *Journal of Machine Learning Research*, 13(May):1665–1697, 2012.

[39] Sahand Negahban, Bin Yu, Martin J Wainwright, and Pradeep K Ravikumar. A unified framework for high-dimensional analysis of m-estimators with decomposable regularizers. In *Advances in Neural Information Processing Systems*, pages 1348–1356, 2009.

[40] Yurii Nesterov. *Introductory lectures on convex optimization: A Basic Course*. Springer Science & Business Media, 2004.

[41] Praneeth Netrapalli, UN Niranjan, Sujay Sanghavi, Animashree Anandkumar, and Prateek Jain. Non-convex robust pca. In *NIPS*, pages 1107–1115, 2014.

[42] Dohyung Park, Anastasios Kyrillidis, Constantine Caramanis, and Sujay Sanghavi. Finding low-rank solutions via non-convex matrix factorization, efficiently and provably. *arXiv preprint arXiv:1606.03168*, 2016.

[43] Garvesh Raskutti, Martin J. Wainwright, and Bin Yu. Restricted eigenvalue properties for correlated gaussian designs. *Journal of Machine Learning Research*, 99:2241–2259, August 2010.

[44] Garvesh Raskutti, Martin J Wainwright, and Bin Yu. Minimax rates of estimation for high-dimensional linear regression over-balls. *Information Theory, IEEE Transactions on*, 57(10):6976–6994, 2011.

[45] Benjamin Recht, Maryam Fazel, and Pablo A Parrilo. Guaranteed minimum-rank solutions of linear matrix equations via nuclear norm minimization. *SIAM review*, 52(3):471–501, 2010.

[46] Stephen Tu, Ross Boczar, Mahdi Soltanolkotabi, and Benjamin Recht. Low-rank solutions of linear matrix equations via procrustes flow. *arXiv preprint arXiv:1507.03566*, 2015.

[47] R. Vershynin. Introduction to the non-asymptotic analysis of random matrices. *arXiv preprint arXiv:1011.3027*, 2010.

[48] Roman Vershynin. On the role of sparsity in compressed sensing and random matrix theory. In *Computational Advances in Multi-Sensor Adaptive Processing (CAMSAP), 2009 3rd IEEE International Workshop on*, pages 189–192. IEEE, 2009.

[49] Lingxiao Wang, Xiao Zhang, and Quanquan Gu. A unified computational and statistical framework for nonconvex low-rank matrix estimation. *arXiv preprint arXiv:1610.05275*, 2016.

[50] Lingxiao Wang, Xiao Zhang, and Quanquan Gu. A universal variance reduction-based catalyst for nonconvex low-rank matrix recovery. *arXiv preprint arXiv:1701.02301*, 2017.

[51] Andrew E Waters, Aswin C Sankaranarayanan, and Richard Baraniuk. Sparcs: Recovering low-rank and sparse matrices from compressive measurements. In *NIPS*, pages 1089–1097, 2011.

[52] Huan Xu, Constantine Caramanis, and Sujay Sanghavi. Robust pca via outlier pursuit. In *Advances in Neural Information Processing Systems*, pages 2496–2504, 2010.

[53] Huan Xu and Chenlei Leng. Robust multi-task regression with grossly corrupted observations. In *AISTATS*, pages 1341–1349, 2012.

[54] Pan Xu, Jian Ma, and Quanquan Gu. Speeding up latent variable gaussian graphical model estimation via nonconvex optimization. In *Advances in Neural Information Processing Systems*, pages 1930–1941, 2017.

[55] Eunho Yang and Pradeep K Ravikumar. Dirty statistical models. In *Advances in Neural Information Processing Systems*, pages 611–619, 2013.

[56] Xinyang Yi, Dohyung Park, Yudong Chen, and Constantine Caramanis. Fast algorithms for robust pca via gradient descent. In *Advances in neural information processing systems*, pages 4152–4160, 2016.

[57] Qinqing Zheng and John Lafferty. Convergence analysis for rectangular matrix completion using burer-monteiro factorization and gradient descent. *arXiv preprint arXiv:1605.07051*, 2016.

[58] Tianyi Zhou and Dacheng Tao. Godec: Randomized low-rank & sparse matrix decomposition in noisy case. In *International conference on machine learning*. Omnipress, 2011.