A Nonconvex Projection Method for Robust PCA

Aritra Dutta,[*] Filip Hanzely,[†] Peter Richtárik[‡]

Abstract

Robust principal component analysis (RPCA) is a well-studied problem whose goal is to decompose a matrix into the sum of low-rank and sparse components. In this paper, we propose a nonconvex feasibility reformulation of RPCA problem and apply an alternating projection method to solve it. To the best of our knowledge, this is the first paper proposing a method that solves RPCA problem without considering any objective function, convex relaxation, or surrogate convex constraints. We demonstrate through extensive numerical experiments on a variety of applications, including shadow removal, background estimation, face detection, and galaxy evolution, that our approach matches and often significantly outperforms current state-of-the-art in various ways.

Principal component analysis (PCA) (Jolliffee 2002) addresses the problem of best approximation of a matrix $A \in \mathbb{R}^{m\times n}$ by a matrix of rank r:

$$X^* = \arg \min_{\substack{X\in\mathbb{R}^{m\times n} \\ \mathrm{rank}(X)\leq r}} \|A - X\|_F^2, \tag{1}$$

where $\|\cdot\|_F$ denotes the Frobenius norm of matrices. The solution to (1) is given by

$$X^* = H_r(A) \overset{\text{def}}{=} U\Sigma_r V^T, \tag{2}$$

where A has singular value decompositions $A = U\Sigma V^T$, and $\Sigma_r(A)$ is the diagonal matrix obtained from Σ by hard-thresholding that keeps the r largest singular values only and replaces the other singular values by 0. In many real-world problems, if sparse large errors or outliers are present in the data matrix, PCA fails to deal with it. Therefore, it is natural to consider a *robust* matrix decomposition model in which we wish to decompose A into the sum of a low-rank matrix L and an error matrix S: $A = L + S$. However, without further assumptions, the problem is ill-posed. We assume that the error matrix S is *sparse* and that it allows its entries to have arbitrarily large magnitudes. That is, given A, we consider the problem of finding a low rank matrix L and a sparse matrix S such that

$$A = L + S. \tag{3}$$

In this context, the celebrated principal component pursuit (PCP) formulation of the problem uses the ℓ_0 norm (cardinality) to address the sparsity constraint and (3). Therefore, PCP is the constrained minimization problem (Candès et al. 2011; Chandrasekaran et al. 2011):

$$\min_{L,S} \mathrm{rank}(L) + \lambda\|S\|_{\ell_0} \quad \text{subject to} \quad A = L + S, \tag{4}$$

where $\lambda > 0$ is a balancing parameter. Since both $\mathrm{rank}(L)$ and $\|S\|_{\ell_0}$ are non-convex, one often replaces the rank function by the (convex) nuclear norm and ℓ_0 by the (convex) ℓ_1 norm. This replacement leads to the immensely popular *robust principal component analysis (RPCA)* (Wright et al. 2009; Lin, Chen, and Ma 2010; Candès et al. 2011), which can be seen as a convex relaxation of (4):

$$\min_{L,S} \|L\|_* + \lambda\|S\|_{\ell_1} \quad \text{subject to} \quad A = L + S, \tag{5}$$

where $\|\cdot\|_*$ denotes the nuclear norm (sum of the singular values) of matrices. Under some reasonable assumptions on the low-rank and sparse components, (Chandrasekaran et al. 2011; Candès et al. 2011) showed that (4) can be provably solved via (5). A vast literature is dedicated to solving the RPCA problem, and among them the exact and inexact augmented Lagrangian method of multipliers (Lin, Chen, and Ma 2010), accelerated proximal gradient method (Wright et al. 2009), alternating direction method (Yuan and Yang 2013), alternating projection with intermediate denoising (Netrapalli et al. 2014), dual approach (Lin et al. 2009), and SpaRCS (Waters, Sankaranarayanan, and Baraniuk 2011) are a few popular ones. Recently, Yi et al. (Yi et al. 2016), Zhang and

[*]Aritra Dutta is with the Visual Computing Center, Division of Computer, Electrical and Mathematical Sciences and Engineering (CEMSE) at King Abdullah University of Science and Technology, Thuwal, Saudi Arabia-23955-6900, e-mail: aritra.dutta@kaust.edu.sa.

[†]Filip Hanzely is with the Visual Computing Center, Division of Computer, Electrical and Mathematical Sciences and Engineering (CEMSE) at King Abdullah University of Science and Technology, Thuwal, Saudi Arabia-23955-6900, e-mail: filip.hanzely@kaust.edu.sa.

[‡]Peter Richtárik is with the Visual Computing Center, Division of Computer, Electrical and Mathematical Sciences and Engineering (CEMSE) at King Abdullah University of Science and Technology, University of Edinburgh, and MIPT, e-mail: peter.richtarik@kaust.edu.sa.

Yang (Zhang and Yang 2017) proposed a manifold optimization to solve RPCA. We refer the reader to (Bouwmans and Zahzah 2014) for a comprehensive review of the RPCA algorithms. However, besides formulation (5), other tractable reformulations of (4) exist as well. For instance, by relaxing the equality constraint in (4) and moving it to the objective as a penalty, together with adding explicit constraints on the target rank r and target sparsity s leads to the following formulation (Zhou and Tao 2011):

$$\begin{aligned} &\min_{L,S} \|A - L - S\|_F^2 \\ &\text{subject to} \quad \operatorname{rank}(L) \leq r \ \text{ and } \ \|S\|_0 \leq s. \end{aligned} \tag{6}$$

One can extend the above model to the case of partially observed data that leads to the *robust matrix completion (RMC)* problem (Chen et al. 2011; Tao and Yang 2011; Cherapanamjeri, Gupta, and Jain 2017):

$$\begin{aligned} &\min_{L,S} \|\mathcal{P}_\Omega(A - L - S)\|_F^2 \\ &\text{subject to} \quad \operatorname{rank}(L) \leq r \ \text{ and } \ \|\mathcal{P}_\Omega(S)\|_0 \leq s', \end{aligned} \tag{7}$$

where $\Omega \subseteq [m] \times [n]$ is the set of observed data entries, and $\mathcal{P}_\Omega$ is the restriction operator defined by

$$(\mathcal{P}_\Omega[X])_{ij} = \begin{cases} X_{ij} & (i,j) \in \Omega \\ 0 & \text{otherwise.} \end{cases}$$

We note that with some modifications, problem (6) is contained in the larger class of problem presented by (7). We also refer to some recent work on RMC problem or outlier based PCA in (Cherapanamjeri, Jain, and Netrapalli 2017; Cherapanamjeri, Gupta, and Jain 2017). An extended model of (5) can also be referred to as a more general problem as in (Tao and Yang 2011) (see problem (1.2) in (Tao and Yang 2011)). More specifically, when $\Omega = [m] \times [n]$, that is, the whole matrix is observed, then (7) is (6). One can also think of the matrix completion (MC) problem as a special case of (7) (Candès and Plan 2009; Jain, Netrapalli, and Sanghavi 2013; Cai, Candès, and Shen 2010; Jain and Netrapalli 2015; Candès and Recht 2009; Keshavan, Montanari, and Oh 2010; Candès and Tao 2010; Mareček, Richtárik, and Takáč 2017). For MC problems, $S = 0$. Therefore, (7) is a generalization of two fundamental problems: RPCA and RMC.

Contributions. We solved the RPCA and RMC problems by addressing the original decomposition problem (3) directly, without introducing any optimization objective or surrogate constraints. This is a novel approach because we aim to find a point at the intersection of three sets, two of which are non-convex. We formulate both RPCA and RMC as set feasibility problems and propose alternating projection algorithm to solve them. This leads to Algorithm 2 and 3. Our approach is described in the next section. We also propose a convergence analysis of our algorithm.

Our feasibility approach does not require one to use the hard to interpret parameters (such as λ) and surrogate functions (such as the nuclear norm, or ℓ_1 norm) which makes our approach unique compared to existing models. Instead, we rely on two direct parameters: the target rank r and the desired level of sparsity s. By performing extensive numerical experiments on both synthetic and real datasets, we show that our approach can match or outperform state-of-the-art methods in solving the RPCA and RMC problems. More precisely, when the sparsity level is low, our feasibility approach can viably reconstruct any target low rank, which the RPCA algorithms can not. Moreover, our approach can tolerate denser outliers than can the state-of-the-art RPCA algorithms when the original matrix has a low-rank structure (see details in the experiment section). These attributes make our approach attractive to solve many real-world problems because our performance matches or outperforms that of state-of-the-art RPCA algorithms in solution quality, and do this in comparable or less time.

Nonconvex Feasibility and Alternating Projections

Set feasibility problem aims to find a point in the intersection of a collection of closed sets, that is:

$$\text{Find} \ \ x \in \mathcal{X} \qquad \text{where} \qquad \mathcal{X} \stackrel{\text{def}}{=} \cap_i^m \mathcal{X}_i \neq \emptyset, \tag{8}$$

for closed sets $\mathcal{X}_i$. Usually, sets $\mathcal{X}_i$s are assumed to be simple and easy to project on. A special case of the above setting for convex sets $\mathcal{X}_i$ is the convex feasibility problem and is already well studied. In particular, a very efficient convex feasibility algorithm is known as the alternating projection algorithm (Kaczmarz 1937; Bauschke and Borwein 1996), in which each iteration picks one set $\mathcal{X}_i$ and projects the current iterate on it. There are two main methods to choosing the sets $\mathcal{X}_i$ – traditional cyclic method and randomized method (Strohmer and Vershynin 2009; Gower and Richtárik 2015; Necoara, Richtárik, and Patrascu 2018), and in general, randomized method is faster and not vulnerable to adversarial set order.

We also note that the alternating projection algorithm for convex feasibility problem does not converge in general to the projection of the starting point onto $\mathcal{X}$, but rather finds a close-to feasible point in $\mathcal{X}$, except the case when $\mathcal{X}_i$s are affine spaces. However, once an exact projection onto $\mathcal{X}$ is desired, Dykstra's algorithm (Boyle and Dykstra 1986) should be applied.

Algorithm 1: Alternating projection method for set feasibility

Input : $\Pi_i(\cdot)$ – Projector onto $\mathcal{X}_i$ for each $i \in \{1, \dots m\}$, starting point x_0
for $k = 0, 1, \dots$ **do**
  Choose via some rule i (e.g., cyclically or randomly)
  $x_{k+1} = \Pi_i(x_k)$
end
Output : x_{k+1}

On the other hand, for general nonconvex sets $\mathcal{X}_i$, projection algorithms might not converge. In some special settings, some forms of convergence (for example, local convergence) can be guaranteed even without convexity (Lewis and Malick 2008; Lewis, Luke, and Malick 2009; Hesse and Luke 2013; Drusvyatskiy, Ioffe, and Lewis 2015; Pang 2015).

Set feasibility for RPCA

In this scope, we define α-sparsity as it appears in the last convex constraint $\mathcal{X}_3$. We do it so that our approach is directly comparable to the approaches from (Yi et al. 2016; Zhang and Yang 2017). However, we note that the ℓ_0-ball constraint can be applied as well.

Definition 1 (α-sparsity)**.** A matrix $S \in \mathbb{R}^{m\times n}$ is considered to be α-sparse if each row and column of S contains at most αn and αm nonzero entries, respectively. That is, the cardinality of the support set of each row and column of the matrix S do not exceed αn and αm, respectively. Formally, we write

$$\|S_{(i,.)}\|_0 \le \alpha n \;\text{ and }\; \|S_{(.,j)}\|_0 \le \alpha m \;\text{ for all } i \in [m], j \in [n],$$

where i^{th} row and j^{th} column of S are $S_{(i,.)}$ and $S_{(.,j)}$, respectively.

Now we consider the following reformulation of RPCA:

$$\text{Find} \quad M \overset{\text{def}}{=} [L, S] \in \mathcal{X} \overset{\text{def}}{=} \cap_{i=1}^{3} \mathcal{X}_i \neq \emptyset, \tag{9}$$

where

$$\begin{aligned} \mathcal{X}_1 &\overset{\text{def}}{=} \{M \,|\, L + S = A\} \\ \mathcal{X}_2 &\overset{\text{def}}{=} \{M \mid \operatorname{rank}(L) \le r\} \\ \mathcal{X}_3 &\overset{\text{def}}{=} \{M \,|\, \|S_{(i,.)}\|_0 \le \alpha n \;\text{ and }\; \|S_{(.,j)}\|_0 \le \alpha m \\ &\qquad \text{for all } i \in [m], j \in [n].\} \end{aligned} \tag{10}$$

Clearly, $\mathcal{X}_1$ is convex, but $\mathcal{X}_2$ and $\mathcal{X}_3$ are not. Nevertheless, the algorithm we propose – alternating Frobenius norm projection on $\mathcal{X}_i$ performs well to solve RPCA in practice. To validate the robustness of our algorithm, we compare our method to other state-of-the-art RPCA approaches on various practical problems. We also study the local convergence properties and show that despite the non-convex nature of the problem, the algorithms we propose often behave surprisingly well.

The Algorithm

Denote Π_i to be projector onto $\mathcal{X}_i$. Note that Π_2 does not include S and projection onto Π_3 does not include L. Consequently, $\Pi_2\Pi_3$ is a projector onto $\mathcal{X}_2 \cap \mathcal{X}_3$. Because our goal is to find a point at the intersection of two sets, we shall employ a cyclic projection method (note that randomized method does not make sense). Indeed, steps 4 and 5 of Algorithm 2 perform projection onto $\mathcal{X}_1$, step 6 performs projection onto $\mathcal{X}_2$, and finally, step 7 performs projection onto $\mathcal{X}_3$. Later in this section we describe the exact implementation and prove correctness the steps mentioned above. Lastly, we propose a similar algorithm to solve the RMC problem (7) in Appendix (Algorithm 3). Finally, we provide a local convergence of Algorithm 2, which depends on the local geometry of the optimal point, and is mostly linear, which we prove later.

Projection on the linear constraint. The next lemma provides an explicit formula for the projection onto $\mathcal{X}_1$, which corresponds to steps 4 and 5 of Algorithm 2.

Algorithm 2: Alternating projection method for RPCA

1 **Input** : $A \in \mathbb{R}^{m\times n}$ (the given matrix), rank r, sparsity level $\alpha \in (0, 1]$
2 **Initialize** : L_0, S_0
3 **for** $k = 0, 1, \ldots$ **do**
4 $\quad \tilde{L} = \frac{1}{2}(L_k - S_k + A)$
5 $\quad \tilde{S} = \frac{1}{2}(S_k - L_k + A)$
6 $\quad L_{k+1} = H_r(\tilde{L})$
7 $\quad S_{k+1} = \mathcal{T}_\alpha(\tilde{S})$
end
8 **Output** : L_{k+1}, S_{k+1}

Lemma 1. *Solutions to*

$$\min_{L,S} \|L - L_0\|_F^2 + \|S - S_0\|_F^2 \quad \textit{subject to} \quad L + S = A$$

is $L^* = \frac{1}{2}(L_0 - S_0 + A)$ *and* $S^* = \frac{1}{2}(S_0 - L_0 + A)$.

Projection on the low rank constraint. Consider $L^{(r)}$ to be the projection of L onto the rank r constraint, that is,

$$L^{(r)} = \arg\min_{L'} \|L' - L\|_F \quad \text{subject to} \quad \operatorname{rank}(L') \le r.$$

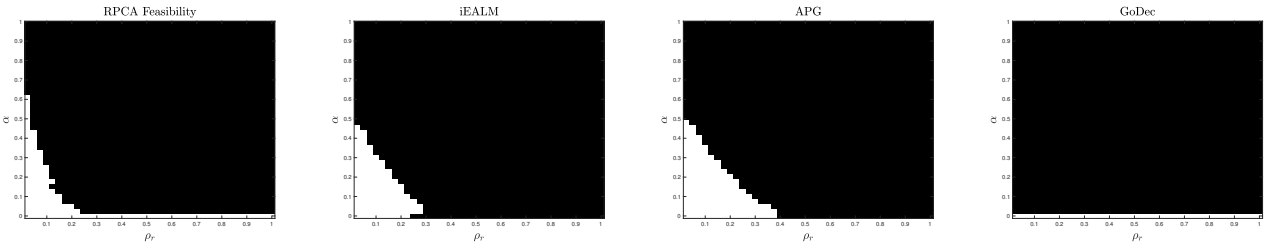
It is known that $L^{(r)}$ can be computed as r-SVD of L. Fast r-SVD solvers has improved greatly in recent years (Halko, Martinsson, and Tropp 2011; Musco and Musco 2015; Shamir 2015; A.-Zhu and Li 2016). Unfortunately, the most recent approaches (Shamir 2015; A.-Zhu and Li 2016) were not applied in our setting because they are inefficient; they need to compute $LL^\top$ (or $L^\top L$), which is expensive. Instead, we use block Krylov approach from (Musco and Musco 2015). For completeness, we quote the algorithm in Appendix. Regarding the computational complexity, it was shown that block Krylov SVD outputs Z satisfying $\|L - ZZ^\top L\|_F \le (1+\tilde{\epsilon})\|L - L^{(r)}\|_F$ in

$$\mathcal{O}\left(\|L\|_0 \frac{r \log n}{\sqrt{\tilde{\epsilon}}} + \frac{mr^2 \log^2 n}{\tilde{\epsilon}} + \frac{r^3 \log^3 n}{\tilde{\epsilon}^{3/2}}\right)$$

flops. Therefore, projection on the low-rank constraint is not an issue for relatively small rank r.

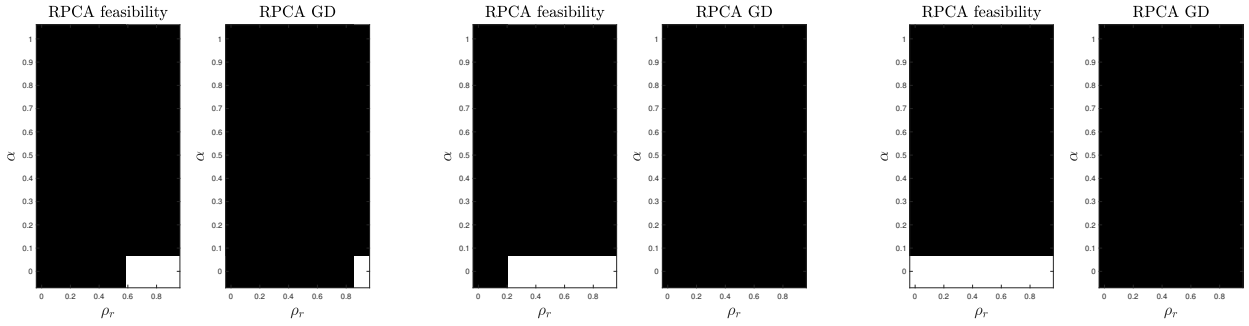
Projection on sparsity constraint. Projection onto $\mathcal{X}_3$ simply keeps the α-fraction of the largest elements in absolute value in each row and column and set the rest to zero. One can use a global hard-thresholding operator that considers ℓ_0 constraint on the entire matrix. Instead, we proposed an operator $\mathcal{T}_\alpha(\cdot)$. Indeed, $\mathcal{T}_\alpha(\cdot)$ does not perform an explicit Euclidean projection onto $\mathcal{X}_3$. Instead, it performs a projection onto a certain subset of $\mathcal{X}_3$ and this is clear from the definition (11) (the subset is defined through support Ω_α). Formally, we define:

$$\begin{aligned} \mathcal{T}_\alpha[S] &\overset{\text{def}}{=} \mathcal{P}_{\Omega_\alpha}(S) \in \mathbb{R}^{m\times n} : (i,j) \in \Omega_\alpha \text{if} \\ &\qquad |S_{ij}| \ge |S^{(\alpha n)}_{(i,.)}| \;\text{ and }\; |S_{ij}| \ge |S^{(\alpha m)}_{(.,j)}|, \end{aligned} \tag{11}$$



(a) (b) (c) (d)

Figure 1: Phase transition diagram for RPCA F, iEALM, APG, and GoDec with respect to rank and error sparsity. Here, $\rho_r = \mathrm{rank}(L)/m$ and α is the sparsity measure. We have $(\rho_r, \alpha) \in (0.025, 1] \times (0, 1)$ with $r = 5 : 5 : 200$ and $\alpha = \texttt{linspace}(0, 0.99, 40)$. We perform 10 runs of each algorithm.



(a) (b) (c)

Figure 2: Phase transition diagram for Relative error for RMC problems: (a) $|\Omega^C| = 0.5(m.n)$, (b) $|\Omega^C| = 0.75(m.n)$, (c) $|\Omega^C| = 0.9(m.n)$. Here, $\rho_r = \mathrm{rank}(L)/m$ and α is the sparsity measure. We have $(\rho_r, \alpha) \in (0.025, 1] \times (0, 1)$ with $r = 5 : 25 : 200$ and $\alpha = \texttt{linspace}(0, 0.99, 8)$.

where $S^{(\alpha n)}_{(i,.)}$ and $S^{(\alpha m)}_{(.,j)}$ denote the α fraction of largest entries of S along the i^{th} row and j^{th} column, respectively. This allows us to inexpensively compute an approximate projection onto $\mathcal{X}_3$ which works well in practice. Remarkably, this does not affect our theoretical results (which are formulated for exact projection onto $\mathcal{X}_3$) in any way. We note that the operator $\mathcal{T}_\alpha(\cdot)$ is similar to that defined in (Yi et al. 2016; Zhang and Yang 2017). Projection on sparsity constraint (11) can be implemented in $\mathcal{O}(nd)$ time: for each row and each column we find αn-th largest element (or αd) and simultaneously (for other rows/columns) mask the rest. In our experiments, we use fast implementation of n-th element computation from (Li 2013).

Remark 1. One may use the Douglas-Rachford operator splitting method (Artacho, Borwein, and Tam 2016) as an alternative to the nonconvex projections. We leave this for future research.

Convergence Analysis

In this section, we establish a local convergence analysis of our algorithm by using the basic properties of the alternating projection algorithm (Lewis and Malick 2008). A similar analysis was done for GoDec (Zhou and Tao 2011), although Algorithm 2 is vastly different compare to GoDec (we report detailed comparison of these algorithms in Appendix).

Recall that Algorithm 1 performs an alternating projection of $[L, S]$ on the sets $\mathcal{X}_1$ and $\mathcal{X}_\cap \overset{\text{def}}{=} \mathcal{X}_2 \cap \mathcal{X}_3$ defined in (10). Before stating the convergence theorem, let us define a (local) angle between sets.

Definition 2. Let a point p be in the intersection of set boundaries ∂K and ∂L. Define $c(K, L, p)$ be the cosine of an angle between sets K and L at point a p as:

$$c(K, L, p) \overset{\text{def}}{=} \cos \angle(\partial K^\top|_p, \partial L^\top|_p),$$

where $\partial K^\top|_p$ denotes a tangent space of set boundary ∂K at p and $\angle$ returns the angle between subspaces given as arguments. As a consequence, we have $0 \leq c(K, L, p) \leq 1$.

Let us also define $d_{\mathcal{X}_1 \cap \mathcal{X}_\cap}(x)$ to be an Euclidean distance of a point x to the set $\mathcal{X}_1 \cap \mathcal{X}_\cap$.

Theorem 3. *(Lewis and Malick 2008) Suppose that* $[\bar{L} \;\; \bar{S}] \in \partial(\mathcal{X}_1 \cap \mathcal{X}_\cap)$. *Given any constant* $c \in \mathbb{R}$ *such that* $c > c(\mathcal{X}_1, \mathcal{X}_\cap, [\bar{L} \;\; \bar{S}])$ *there is a starting point* $[L_0 \;\; S_0]$ *close to* $[\bar{L} \;\; \bar{S}]$ *such that the iterates* L_k, S_k *of Algorithm 2 satisfy*

$$d_{\mathcal{X}_1 \cap \mathcal{X}_\cap}([L_k \;\; S_k]) < c^k d_{\mathcal{X}_1 \cap \mathcal{X}_\cap}([L_0 \;\; S_0]).$$

Remark 2. From Theorem 3 it is clear that the smaller $c(\mathcal{X}_1, \mathcal{X}_\cap, [\bar{L} \;\; \bar{S}])$ produces a faster convergence, while $c(\mathcal{X}_1, \mathcal{X}_\cap, [\bar{L} \;\; \bar{S}]) = 1$ can stop the convergence, as described in Example 2 in Appendix.

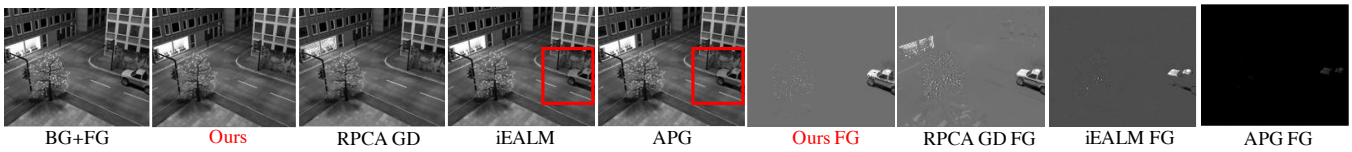


Figure 3: Background and foreground separation on Stuttgart dataset `Basic` video. Except RPCA GD and our method, all other methods fail to remove the static foreground object.

Figure 4: Background and foreground separation on Stuttgart dataset `Basic` video. We used 90% sample. GRASTA forms a fragmentary background and exhausts around 540 frames to form a stable video. We also note that RPCA GD has more false positives in the foreground.

Remark 3. Theorem 3 is stated for Algorithm 2, however, one can easily obtain an equivalent result for Algorithm 3 as well.

Remark 4. Considering the nuclear norm relaxation instead of low rank constraint and ℓ_1 norm relaxation instead of sparsity constraint, the set $\mathcal{X}_\cap$ becomes convex, and thus the whole problem becomes convex as well. Therefore, Algorithms 2 and 3 converge globally.

For completeness, we also derive the exact form of tangent spaces of $\mathcal{X}_1, \mathcal{X}_\cap$ mentioned in Definition 2. Suppose that $\mathrm{rank}(\bar{L}) = r$, and $\bar{S}$ is a matrix of maximal sparsity, that is, $\bar{S} \in \mathcal{X}_3$ while $\bar{S} + S' \notin \mathcal{X}_3$ s.t. $\|S'\| = 1$ and $\|\bar{S} + S'\|_0 = \|\bar{S}\|_0 + 1$. The tangent spaces of $\partial\mathcal{X}_1$ and $\partial\mathcal{X}_\cap$ at point $[\bar{L} \;\; \bar{S}]$ are given by

$$\begin{aligned} \partial\mathcal{X}_1^\top|_{[\bar{L}\;\;\bar{S}]} &= \mathcal{X}_1 \\ \partial\mathcal{X}_\cap^\top|_{[\bar{L}\;\;\bar{S}]} &= \partial\mathcal{X}_2^\top|_{\bar{L}} \times \partial\mathcal{X}_3^\top|_{\bar{S}}, \end{aligned}$$

where

$$\begin{aligned} \partial\mathcal{X}_2^\top|_{\bar{L}} &= \{\tilde{L} |\, \tilde{L} = \bar{L} + \tilde{U}\tilde{\Sigma}\tilde{V}^\top, \tilde{U}^\top\tilde{U} = \tilde{V}^\top\tilde{V} = I, \\ &\quad \bar{U}^\top\tilde{U} = 0, \bar{V}^\top\tilde{V} = 0, \tilde{\Sigma} = \mathrm{diag}\,\tilde{\Sigma}\} \\ \partial\mathcal{X}_3^\top|_{\bar{S}} &= \{\tilde{S} |\, \tilde{S} = \bar{S} + S', S'_{i,j} = 0 \;\forall (i,j) \text{ s.t. } \bar{S}_{i,j} = 0\}. \end{aligned}$$

Later in Appendix, we also empirically show that:

1. Convergence speed is not significantly influenced by starting point.
2. Convergence is usually fastest for small true sparsity level α and small true rank r, which is the situation in many practical applications.
3. Convergence of Algorithm 3 is slower for medium sized number of observable entries, that is, when $|\Omega| \approx 0.5(m \cdot n)$, and faster for smaller and bigger sizes.
4. If sparsity and rank levels (α and r) are set to be smaller than their true values at the optimum incorrectly, Algorithm 2 does not converge (as in this case, $\cap\mathcal{X}_i$ might not exist). Moreover, the performance of the algorithm is sensitive to the choice of r, and this is particularly so if we underestimate the true value (see Figure 8 in Appendix).

Finally, in Appendix we give two examples for the convex version of the problem (9) with the same block structure; in them, the alternating projection algorithm either converges extremely fast or does not even converge linearly.

Numerical experiments

To explore the strengths and flexibility of our feasibility approach, we performed numerical experiments. First, we work with synthetic data and subsequently apply our method to four real-world problems.

Results on synthetic data

To perform our numerical simulations, first, we construct the test matrix A. We follow the seminal work of Wright et al. (Wright et al. 2009) to design our experiment. To this end, we construct A as a low-rank matrix, L, corrupted by sparse large noise, S, with arbitrary large entries such that $A = L + S$. We generate L as a product of two independent full-rank matrices of size $m \times r$ whose elements are independent and identically distributed (i.i.d.) $\mathcal{N}(0,1)$ random variables and $\mathrm{rank}(L) = r$. We generate S as a noise matrix whose elements are sparsely supported by using the operator (11) and lie in the range $[-500, 500]$. We fix $m = 200$ and define $\rho_r = \mathrm{rank}(L)/m$ where $\mathrm{rank}(L)$ varies. We choose the sparsity level $\alpha \in (0,1)$. For each pair of (ρ_r, α) we apply iEALM, APG, GoDec, and our algorithm to recover the pair $(\hat{L}, \hat{S})$ such that $\hat{A} = \hat{L} + \hat{S}$ be the recovered matrix. For both APG and iEALM, we set $\lambda = 1/\sqrt{m}$ and for iEALM we use $\mu = 1.25/\|A\|_2$ and $\rho = 1.5$, where $\|A\|_2$ is the

spectral norm (maximum singular value) of A. For GoDec we set $q = 2$. If the recovered matrix pair $(\hat{L}, \hat{S})$ satisfies the relative error $\frac{\|L-\hat{L}\|_F+\|S-\hat{S}\|_F}{\|A\|_F} < 0.01$ then we consider the construction is viable. In Figure 1 we show the fraction of perfect recovery, where white denotes *success* and black denotes *failure*. As mentioned in (Wright et al. 2009), the success of APG is approximately below the line $\rho_r + \alpha = 0.35$. as that of APG; and GoDec only recovers the matrices with low sparsity level. To conclude, when the sparsity level α is low, our feasibility approach can provide a feasible reconstruction for any ρ_r. We note that for low sparsity level, the RPCA algorithms can only provide a feasible reconstruction for $\rho_r \leq 0.4$. On the other hand, for low ρ_r, our feasibility approach can tolerate sparsity level approximately up to 63%. In contrast, RPCA algorithms can tolerate sparsity up to 50% for low ρ_r. Therefore, taken together, we can argue that our method can be proved useful to solve real-world problems when one wants to recover a moderately sparse matrix having *any* inherent low-rank structure present in it or in case of a low-rank matrix corrupted by dense outliers of arbitrary large magnitudes.

Results on synthetic data: RMC problem

We used a similar technique as that used in the previous section to generate the test matrix A. We fixed $m = 200$ and denote ρ_r and α same as previously. We randomly select the set of observable entries in A. We compare our method against the RPCA gradient descent (RPCA GD) by Yi et al. (Yi et al. 2016) and use the relative error for the low-rank component recovered as performance measure, that is, if $\|L - \hat{L}\|_F/\|L\|_F < \tilde{\epsilon}$ then we consider the construction is viable. Note that L is the original low-rank matrix and that $\hat{L}$ is the low-rank matrix recovered. For $|\Omega^C| = 0.5(m.n), 0.75(m.n)$, and $0.9(m.n)$ we consider $\tilde{\epsilon} = 0.2, 0.6$, and 1, respectively. In Figure 2, for the phase transition diagram white denotes *success* and black denotes *failure*. From Figure 2 we observe that irrespective of the cardinalities of the set of the observed entries, our feasibility approach outperforms RPCA GD. Interestingly, as the cardinality of the set of the observable entries, that is, $|\Omega|$ decreases, the performance of our feasibility approach improves (see Figure 2c). We compare these two methods with respect to the root mean square error (RMSE) (reported in Appendix).

Applications to real-world problem

In this section we demonstrate the robustness of our feasibility approach to solving four classic real-world problems: (i) background and foreground estimation from fully and partially observed data, (ii) shadow removal from face images captured under varying illumination and camera position, (iii) inlier subspace detection, (iv) processing astronomical data.

Background and foreground estimation from fully observed data. In the past decade, one of the most prevalent approaches used to solve background estimation problem has been treated by most approaches as a low-rank and sparse matrix decomposition problem (Bouwmans et al. 2017; Sobral and Vacavant 2014; Dutta et al. 2017; Mateos and Giannakis 2012; Wang et al. 2012; He, Balzano, and Szlam 2012; Xu et al. 2013; Dutta 2016; Dutta, Li, and Richtárik 2017; Dutta, Li, and Richtárik 2018; Dutta and Li 2017). In the case of a sequence of n video frames whose each frame is mapped into a vector $a_i \in \mathbb{R}^m$, $i = 1, 2, ..., n$, the data matrix $A \in \mathbb{R}^{m\times n}$ in the collection of all the frame vectors is expected to be split into $L+S$. This idea led researchers to introduce RPCA (Candès et al. 2011; Lin, Chen, and Ma 2010; Wright et al. 2009) in which they consider the background frames, L, to have a low-rank structure and the foreground, S, to be sparse. The convex relaxation of the problem is (5). For simulations, we used the `Basic` sequence of the Stuttgart artificial dataset (Brutzer, Höferlin, and Heidemann 2012). Also, we compare our methods against inexact augmented Lagrange methods of multiplier (iEALM) of Lin et al. (Lin, Chen, and Ma 2010), accelerated proximal gradient (APG) of Wright et al. (Wright et al. 2009), and RPCA GD. We downsampled the video frames to a resolution of 144×176 and for iEALM we use $\mu = 1.25/\|A\|_2$ and $\rho = 1.5$. For both APG and iEALM we set $\lambda = 1/\sqrt{\max\{m, n\}}$. For RPCA GD and our method we use target rank $r = 2$, sparsity $\alpha = 0.1$. The threshold ϵ for all methods are kept to 2×10^{-4}. The qualitative analysis of the background and foreground recovered on the sample frame of the `Basic` sequence in Figure 3 suggests that our method and RPC GD recover a visually better quality background and foreground than can the other methods. We also note that RPCA GD recovers a foreground with more false positives than can our method and iEALM and APG cannot remove the static foreground object.

Background and foreground estimation from partially observed data. We randomly select the set of observable entries in the data matrix A and tested our algorithm against Grassmannian Robust Adaptive Subspace Tracking Algorithm (GRASTA) (He, Balzano, and Szlam 2012) and RPCA GD. In Figure 4, we demonstrate the performance on the `Basic` sequence of the Stuttgart dataset with $|\Omega| = 0.9(m.n)$. The parameters for our algorithm and RPCA GD are set as the same as those used in the previous section. For GRATSA we set the parameters the same as those mentioned in the authors' website[1]. Our comparison of the performance by different algorithms on a subsampled frame in Figure 4 shows that RPCA GD and our approach can reconstruct the background the best. However, when compared with the foreground ground truth, our method has a better quantitative measure because RPCA GD has a higher number of false positives in the foreground (see Figure 6). Next, we define the metric ϵ–proximity, that is, $d^{\epsilon}(X, Y)$.

ϵ-proximity metric. Let $X = (X_1, \ldots, X_n)$ and $y = (Y_1, \ldots, Y_n)$ be two video sequences (reconstructed foreground and ground truth foreground), where $X_i, Y_i \in \mathbb{R}^m$ are vectors corresponding to frame i, each containing m pixels. We scale all pixel values to $[0, 1]$. To compare the video

[1] https://sites.google.com/site/hejunzz/grasta

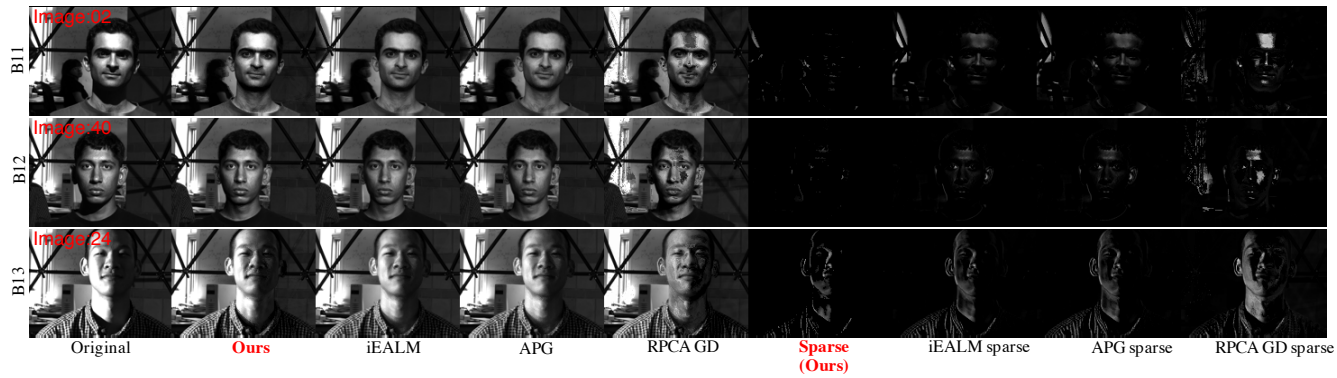
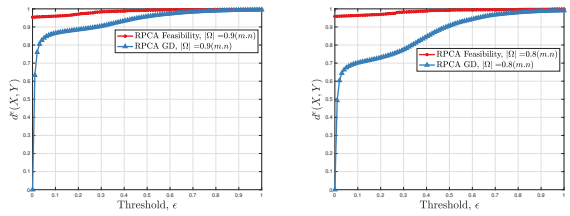


Figure 5: Shadow and specularities removal from face images captured under varying illumination and camera position. Our feasibility approach provides comparable reconstruction to that of iEALM and APG.



(a) (b)

Figure 6: Quantitative comparison of foreground recovered by RPCA F and RPCA GD on `Basic` video, frame size 144×176 with observable entries: (a) $|\Omega| = 0.9(m.n)$, (b) $|\Omega| = 0.8(m.n)$. The performance of RPCA GD drops significantly as $|\Omega|$ decreases. In contrast, the performance of RPCA F stays stable irrespective of the size of $|\Omega|$. See Appendix for description of the metric $d^{\epsilon}(X, Y)$.

sequences, we define an ϵ–proximity of X and Y as

$$d^{\epsilon}(X, Y) \stackrel{\text{def}}{=} \frac{1}{nm} \sum_{i=1}^{n} \sum_{k=1}^{m} d^{\epsilon}(X_{ik}, Y_{ik}),$$

where

$$d^{\epsilon}(u, v) \stackrel{\text{def}}{=} \begin{cases} 1 & |u - v| \leq \epsilon, \\ 0 & \text{otherwise,} \end{cases}$$

and $\epsilon \in [0, 1]$ is a threshold. Clearly, $0 \leq d^{\epsilon}(X, Y) \leq 1$, $\epsilon \mapsto d^{\epsilon}(X, Y)$ is increasing, and $d^{1}(X, Y) = 1$. If $d^{\epsilon}(X, Y) = \alpha$, then $\alpha \times 100\%$ of pixels in the recovered video are within ϵ distance, in absolute value, from the ground truth. We used ϵ–proximity to plot results in Figure 6 and more quantitative results by using different $|\Omega|$ in Appendix.

Removal of shadows. The images of a face exposed to a wide variety of lighting conditions can be approximated accurately by a low-dimensional linear subspace. More specifically, the images under distant, isotropic lighting lie close to a 9-dimensional linear subspace which is known as the harmonic plane (Basri and Jacobs 2003). We used the `Extended Yale Face Database` for our experiments (Georghiades, Belhumeur, and Kriegman 2001). We compared iEALM, APG, and RPCA GD against our algorithm. We downsampled each image to a resolution of 120×160 and use 63 images of a subject in each test. For APG and iEALM, the same parameters as those used previously were set. For RPCA GD and our method, we set the target rank $r = 9$ and sparsity level $\alpha = 0.1$. The qualitative analysis on the recovered images shows that image reconstruction obtained by our feasibility approach is comparable to that by iEALM and APG (see Figure 5). In contrast, the visual quality of the images of a face reconstructed by RPCA GD are poor.

Further experiments. Due to the limitation of the space, we show many of our experimental results in Appendix. First, on synthetic data, we empirically validate the sensitivity of Algorithm 2 with respect to the initialization, the choices of r, and sparsity level α; and the effect of the cardinality of Ω for Algorithm 3. Next, for quantitative experiments on partially observed background estimation, we incorporate more results for different values of $|\Omega|$ by using the ϵ–proximity metric. Finally, we perform experiments on inlier face detection and galaxy evolution and show the qualitative and quantitative results.

Conclusion

In this paper, we proposed a simplistic and novel approach to solving the classic RPCP and RMC problems. We considered an alternating projection algorithm based on the set feasibility approach to solve these problems in their crude form, without considering any further heuristics, such as loss functions, convex and surrogate constraints. Also, we proposed a convergence analysis of our method; we also investigated the convergence through numerical simulations on synthetic and real-world data and extensively compared with the current state-of-the-art methods. Our feasibility approach can open a new direction of potential research on online algorithms based on RPCA framework (Rodriguez and Wohlberg 2016; He, Balzano, and Szlam 2012; Xu et al. 2013) that are vastly used in key areas, including video analysis, segmentation, subspace detection.

References

A.-Zhu, Z., and Li, Y. 2016. LazySVD: Even faster SVD decomposition yet without agonizing pain. In *Advances in Neural Information Processing Systems*, 974–982.

Artacho, F. J. A.; Borwein, J. M.; and Tam, M. K. 2016. Global behavior of the Douglas–Rachford method for a nonconvex feasibility problem. *Journal of Global Optimization* 65(2):309–327.

Basri, R., and Jacobs, D. 2003. Lambertian reflection and linear subspaces. *IEEE Transaction on Pattern Analysis and Machine Intelligence* 25(3):218–233.

Bauschke, H. H., and Borwein, J. M. 1996. On projection algorithms for solving convex feasibility problems. *SIAM review* 38(3):367–426.

Bouwmans, T., and Zahzah, E.-H. 2014. Robust PCA via principal component pursuit: A review for a comparative evaluation in video surveillance. *Computer Vision and Image Understanding* 122:22–34.

Bouwmans, T.; Sobral, A.; Javed, S.; Jung, S. K.; and Zahzah, E.-H. 2017. Decomposition into low-rank plus additive matrices for background/foreground separation: A review for a comparative evaluation with a large-scale dataset. *Computer Science Review* 23:1–71.

Boyle, J. P., and Dykstra, R. L. 1986. A method for finding projections onto the intersection of convex sets in Hilbert spaces. In *Advances in order restricted statistical inference*. Springer. 28–47.

Brutzer, S.; Höferlin, B.; and Heidemann, G. 2012. Evaluation of background subtraction techniques for video surveillance. *IEEE Computer Vision and Pattern Recognition* 1568–1575.

Cai, J. F.; Candès, E. J.; and Shen, Z. 2010. A singular value thresholding algorithm for matrix completion. *SIAM Journal on Optimization* 20(4):1956–1982.

Candès, E. J., and Plan, Y. 2009. Matrix completion with noise. *Proceedings of the IEEE* 98(6):925–936.

Candès, E. J., and Recht, B. 2009. Exact matrix completion via convex optimization. *Foundations of Computational Mathematics* 9(6):717–772.

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

Candès, E. J.; Li, X.; Ma, Y.; and Wright, J. 2011. Robust principal component analysis? *Journal of the Association for Computing Machinery* 58(3):11:1–11:37.

Chandrasekaran, V.; Sanghavi, S.; Parrilo, P. A.; and Willsky, A. S. 2011. Rank-sparsity incoherence for matrix decomposition. *SIAM Journal on Optimization* 21(2):572–596.

Chen, Y.; Xu, H.; Caramanis, C.; and Sanghavi, S. 2011. Robust matrix completion and corrupted columns. In *Proceedings of the 28th International Conference on International Conference on Machine Learning*, 873–880.

Cherapanamjeri, Y.; Gupta, K.; and Jain, P. 2017. Nearly optimal robust matrix completion. In *Proceedings of the 34th International Conference on Machine Learning (ICML)*, 797–805.

Cherapanamjeri, Y.; Jain, P.; and Netrapalli, P. 2017. Thresholding based outlier robust PCA. In *Proceedings of the 30th Conference on Learning Theory (COLT)*, 593–628.

Drusvyatskiy, D.; Ioffe, A. D.; and Lewis, A. S. 2015. Transversality and alternating projections for nonconvex sets. *Foundations of Computational Mathematics* 15(6):1637–1651.

Dutta, A., and Li, X. 2017. Weighted low rank approximation for background estimation problems. In *The IEEE International Conference on Computer Vision Workshops (ICCVW)*, 1853–1861.

Dutta, A.; Gong, B.; Li, X.; and Shah, M. 2017. Weighted singular value thresholding and its application to background estimation. arXiv:1707.00133.

Dutta, A.; Li, X.; and Richtárik, P. 2017. A batch-incremental video background estimation model using weighted low-rank approximation of matrices. In *The IEEE International Conference on Computer Vision Workshops (ICCVW)*, 1835–1843.

Dutta, A.; Li, X.; and Richtárik, P. 2018. Weighted low-rank approximation of matrices and background modeling. arXiv:1804.06252.

Dutta, A. 2016. *Weighted Low-Rank Approximation of Matrices:Some Analytical and Numerical Aspects*. Ph.D. Dissertation, University of Central Florida.

Georghiades, A.; Belhumeur, P.; and Kriegman, D. 2001. From few to many: Illumination cone models for face recognition under variable lighting and pose. *IEEE Transactions on PAMI* 23(6):643–660.

Gower, R. M., and Richtárik, P. 2015. Randomized iterative methods for linear systems. *SIAM Journal on Matrix Analysis and Applications* 36(4):1660–1690.

Halko, N.; Martinsson, P.-G.; and Tropp, J. A. 2011. Finding structure with randomness: Probabilistic algorithms for constructing approximate matrix decompositions. *SIAM review* 53(2):217–288.

He, J.; Balzano, L.; and Szlam, A. 2012. Incremental gradient on the Grassmannian for online foreground and background separation in subsampled video. *IEEE Computer Vision and Pattern Recognition* 1937–1944.

Hesse, R., and Luke, D. R. 2013. Nonconvex notions of regularity and convergence of fundamental algorithms for feasibility problems. *SIAM Journal on Optimization* 23(4):2397–2419.

Jain, P., and Netrapalli, P. 2015. Fast exact matrix completion with finite samples. In *Proceedings of The 28th Conference on Learning Theory (COLT)*, 1007–1034.

Jain, P.; Netrapalli, P.; and Sanghavi, S. 2013. Low-rank matrix completion using alternating minimization. In *Proceedings of the Forty-fifth Annual ACM Symposium on Theory of Computing*, 665–674.

Jolliffee, I. T. 2002. Principal component analysis. Second edition.

Kaczmarz, S. 1937. Angenaherte auflosung von systemen linearer gleichungen. *Bulletin International de l'Académie Polonaise des Sciences et des Lettres, A* 35:355–357.

Keshavan, R.; Montanari, A.; and Oh, S. 2010. Matrix completion from a few entries. *IEEE Transactions on Information Theory* 56(6):2980–2998.

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

Lewis, A. S.; Luke, R.; and Malick, J. 2009. Local linear convergence for alternating and averaged non-convex projections. *Foundations of Computational Mathematics* 9(4):485–513.

Li, P. 2013. Nth element. https://www.mathworks.com/matlabcentral/fileexchange/29453-nth-element.

Lin, Z.; Ganesh, A.; Wright, J.; Wu, L.; Chen, M.; and Ma, Y. 2009. Fast convex optimization algorithms for exact recovery of a corrupted low-rank matrix. *UIUC Technical Report UILU-ENG-09-2214*.

Lin, Z.; Chen, M.; and Ma, Y. 2010. The augmented Lagrange multiplier method for exact recovery of corrupted low-rank matrices. arXiv1009.5055.

Mareček, J.; Richtárik, P.; and Takáč, M. 2017. Matrix completion under interval uncertainty. *European Journal of Operational Research* 256(1):35 – 43.

Mateos, G., and Giannakis, G. 2012. Robust PCA as bilinear decomposition with outlier-sparsity regularization. *IEEE Transaction on Signal Processing* 60(10):5176–5190.

Musco, C., and Musco, C. 2015. Randomized block Krylov methods for stronger and faster approximate singular value decomposition. In *Advances in Neural Information Processing Systems*, 1396–1404.

Necoara, I.; Richtárik, P.; and Patrascu, A. 2018. Randomized projection methods for convex feasibility problems: conditioning and convergence rates. *arXiv preprint arXiv:1801.04873*.

Netrapalli, P.; Niranjan, U. N.; Sanghavi, S.; Anandkumar, A.; and Jain, P. 2014. Non-convex robust PCA. In *Advances in Neural Information Processing Systems 27*. 1107–1115.

Pang, C. H. J. 2015. Nonconvex set intersection problems: From projection methods to the newton method for super-regular sets. *arXiv:1506.08246*.

Rodriguez, P., and Wohlberg, B. 2016. Incremental principal component pursuit for video background modeling. *Journal of Mathematical Imaging and Vision* 55(1):1–18.

Shamir, O. 2015. A stochastic PCA and SVD algorithm with an exponential convergence rate. In *International Conference on Machine Learning*, 144–152.

Sobral, A., and Vacavant, A. 2014. A comprehensive review of background subtraction algorithms evaluated with synthetic and real videos. *Computer Vision and Image Understanding* 122:4 – 21.

Strohmer, T., and Vershynin, R. 2009. A randomized Kaczmarz algorithm with exponential convergence. *Journal of Fourier Analysis and Applications* 15(2):262.

Tao, M., and Yang, J. 2011. Recovering low-rank and sparse components of matrices from incomplete and noisy observations. *SIAM Journal on Optimization* 21(1):57–81.

Wang, N.; Yao, T.; Wang, J.; and Yeung, D.-Y. 2012. A probabilistic approach to robust matrix factorization. In *Proceedings of 12th European Conference on Computer Vision*, 126–139.

Waters, A. E.; Sankaranarayanan, A. C.; and Baraniuk, R. 2011. SpaRCS: Recovering low-rank and sparse matrices from compressive measurements. *Proceedings of 24nd Advances in Neural Information Processing systems* 1089–1097.

Wright, J.; Peng, Y.; Ma, Y.; Ganseh, A.; and Rao, S. 2009. Robust principal component analysis: Exact recovery of corrupted low-rank matrices by convex optimization. *Proceedings of 22nd Advances in Neural Information Processing systems* 2080–2088.

Xu, J.; Ithapu, V. K.; Mukherjee, L.; Rehg, J. M.; and Singh, V. 2013. Gosus: Grassmannian online subspace updates with structured– sparsity. In *In Proceedings of IEEE International Conference on Computer Vision*, 3376–3383.

Yi, X.; Park, D.; Chen, Y.; and Caramanis, C. 2016. Fast algorithms for robust PCA via gradient descent. *Advances in Neural Information Processing systems* 361–369.

Yuan, X., and Yang, J. 2013. Sparse and low-rank matrix decomposition via alternating direction methods. *Pacific Journal of Optimization* 9(1):167–180.

Zhang, T., and Yang, Y. 2017. Robust PCA by manifold optimization. arXiv:1708.00257v3.

Zhou, T., and Tao, D. 2011. Godec: Randomized low-rank and sparse matrix decomposition in noisy case. In *Proceedings of the 28th International Conference on Machine Learning (ICML)*, 33–40.