Robust Subspace Tracking with Missing Data and Outliers: Novel Algorithm and Performance Guarantee

Le Trung Thanh[1], Nguyen Viet Dung[1,2],
Nguyen Linh Trung[1], Karim Abed-Meraim[3,1]

July 2, 2020

Abstract

Subspace tracking, which is refered to online PCA, is a classical problem in signal processing with various applications in wireless communications, rada and image/video processing. Since outliers and missing data are ubiquitous and more common in big data regime, robust variants of subspace tracking (RST) are crucial. In this report, we propose a novel algorithm, namely PETRELS-ADMM, to improve RST performance in such scenario. The proposed approach consists of two main stages, including outlier rejection and subspace estimation. In the first stage, alternating direction method of multipliers (ADMM) solver is used to detect outliers residing in the observed data in an efficient way. In the second stage, we propose a modification of the parallel estimation and tracking by recursive least squares (PETRELS) algorithm to update the underlying subspace. A theoretical convegence analysis is provided, i.e., we prove that PETRELS-ADMM can generate a sequence of subspace solutions converging to the optimum of its batch counterpart. Performance studies show the superiority of our algorithms as compared to the state-of-the-art algorithms on both synthesis data and real data.

Index Terms

Robust subspace tracking, online robust PCA, robust matrix completion, missing data, outliers, alternating direction method of multipliers (ADMM).

[1] University of Engineering and Technology, Vietnam National University, Hanoi, Vietnam.
[2] National Institute of Advanced Technologies of Brittany, Brest, France.
[3] University of Orléans, Orléans, France.

Contents

Robust Subspace Tracking with Missing Data and Outliers: Novel Algorithm and Performance Guarantee

I. Introduction

Subspace estimation is a classical problem in signal processing with numerous applications in wireless communications, radar, navigation and image/video processing, to name a few [1]. It can be stated by a problem of estimating a r-dimensional subspace $\mathbf{U}$ of $\mathbb{R}^n$ where $r \ll n$, from a set of m observed data vectors $\{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_m\}$, or equivalently, a measurement data matrix $\mathbf{X}$ of size $n \times m$. Generally, it is also known as the principal component analysis (PCA) problem in machine learning. The standard approach is to solving an eigen-problem in batch manner where the underlying subspace is derived from taking either singular value decomposition of the data matrix or eigenvalue decomposition of its covariance matrix. In real-time or large-scale applications, batch algorithms are however not useful and become inefficient due to their high computational complexity $\mathcal{O}(nm\min(m,n))$ and memory cost $\mathcal{O}(nm)$. Subspace tracking or online (dynamic) PCA has been an excellent alternative with a much lower computational complexity as well as memory cost, e.g. being linear with respect to the size of data vectors n.

In the literature of signal processing, extensive surveys of the standard algorithms for subspace tracking are provided in [1], [2]. The algorithms can be categorized into three classes with respect to their complexity, including the class of high complexity $\mathcal{O}(n^2r)$, medium complexity $\mathcal{O}(nr^2)$ and low complexity $\mathcal{O}(nr)$. Note that, there usually exists a tradeoff among subspace estimation accuracy, convergence rate and computational complexity. It is stated that algorithms belonging to the medium complexity class can provide the best performance to cost ratio [2]. However, the standard algorithms are sensitive to the presence of the corruptions in a similar way as to PCA [3]. Their performance may be degraded significantly if the measurement data is corrupted by even a small outliers or missing observations. As mentioned in recent reviews [4]–[6], missing data and outliers are ubiquitous and more common in big data regime. This has led to attempts to define robust variants of subspace learning, namely robust subspace tracking (RST), or online robust PCA (ORPCA). In this work, we aim to investigate the RST problem when dealing with data in the presence of both outliers and missing observations.

A. Related Works

In recent years, there have been several studies to subspace tracking from missing data. Almost attempts are interpreted through geometric lens, i.e., the subspace tracking problem can be stated by an optimization with a certain objective function. Along the line, Grassmannian rank-one update subspace estimation (GROUSE) [7] is an incremental gradient subspace algorithm which performs the stochastic gradient descent on Grassmanian manifold of the r-dimensional subspace. It belongs to the class of low complexity and its convergence has recently been proved in [8]. A robust version of GROUSE for handling outliers residing in the data is Grassmannian robust adaptive subspace tracking (GRASTA) algorithm, which has been presented in [9]. GRASTA first uses a ℓ_1-norm cost function to reduce the effect of sparse outliers and then performs the incremental gradient on the Grassmanian manifold of subspace $\mathbf{U}$ in the similar way as to GROUSE. Although GRASTA is one of the fastest RST algorithms for handling missing data corrupted by outliers, no formal guarantees are obtained for the algorithm. Parallel estimation and tracking by recursive least squares (PETRELS) algorithm, proposed in [10], can be considered as an extension of the famous PAST algorithm [11], for handling missing data. Specifically, PETRELS is a recursive least squares-type algorithm applying the second order stochastic gradient descent to the cost function. The convergence results of PETRELS state that generated solutions can converge to global optima in the full observation setting. Inspired of PETRELS, various subspace tracking algorithms have been proposed to deal with missing data in the same line such as [12]–[14]. The subspace tracking algorithm in [12] is derived from minimizing the sum of squared residuals, but adding a regularization of the nuclear norm of subspace $\mathbf{U}$. The ROSETA algorithm [13] applies an adaptive step size at subspace estimate stage to enhance the convergence rate. While the core of PETRELS-CFAR algorithm [14] is to handle "outliers-removed" data, i.e., outliers are first removed before tracking subspace. However, convergence of the PETRELS-based algorithms have not been mathematically proved yet. Recursive projected compressive sensing (ReProCS)-based algorithms [15], [16] are also able to reconstruct a subspace from missing observations. The ReProCS-based algorithms provide not only a memory-efficient solution, but also a reasonable subspace estimation compared to the state-of-the-art algorithms. However, ReProCS-base algorithms require strong assumptions on subspace changes, outlier magnitudes and accurate initialization (i.e., knowledge of the underlying subspace must be available). In some applications, their assumptions are difficult to meet in data acquisition process or the inherent nature of data in practice. Other subspace tracking algorithms having ability to deal with missing data include pROST [17], APSM [18], POPCA [19] and OVBSL [20]. The algorithms either require to memorize previous observations and good initialization or do not provide a performance guarantee. Among algorithms mentioned above, only few of them can be capable to handle robust subspace tracking in the presence of both outliers and missing observations, including GRASTA [9], pROST [17], ROSETA [13], ReProCS-based algorithms [15], [16] and PETRELS-CFAR [14]. Adopting the approach of PETRELS-CFAR but aiming to improve RST performance, we are interested in looking for a method that can remove outliers more correctly.

B. Contributions

The main contributions of the report is two-fold:

- We propose a novel algorithm for the RST problem to deal with missing data corrupted by outliers. It consists of two main stages, including outlier rejection and subspace estimation. Specifically, outliers residing in the measurement data are detected and removed by our

TABLE I: A comparison of the state of the art algorithms for online robust PCA, robust subspace tracking

Algorithm	Missing Data	Mechanism	Convergence Rate	Convexity	Convergence Guarantee	Complexity
GRASTA (2012 [9])	✓	Grassmannian Manifold + ADMM	$O(1/k)$	✗	✓	✓
OR-PCA (2013 [21])	✗	ℓ_1-norm	✓	✓(global)	✓	$O((nr + r^2)(r + 1))$
pROST (2014 [17])	✓	ℓ_0-norm	✗	✗	-?	
ROSETA (2015 [13])	✓	Manifold + ADMM				
APSM (2015 [18])	✓	Robust Statistics	?	✓	?	$O(nr^2)$
RPCA-TNNR (2016 [22])	✗	?	?	?	?	?
OLP-PCA (2017 [23])	✗	l_p-norm	✗	✓	?	$O(n(r + c)^2)$ †
OMRMD (2017 [24])	✓	ℓ_0-norm	✗	✗	-?	
PETRELS-CFAR (2018 [14])	✓	Robust Statistics				
L1-PCA (2018 [25]–[27])	✗	?	✗	✓	?	$O(t^2nr)$ ‡
ReProCS (2019 [15], [16], [28])	✓	?	✗	✓	?	$O(nr\log(n)\log(1/\epsilon))$ *
OSTP (2019 [29])	✗	ℓ_1-norm + Schatten-p	✗	✓	?	$O(nr^2)$
PETRELS-ADMM	✓	ADMM	$O(1/k)$	✓	✓	

† c is

‡ t denotes the time instant. The subspace update is exponential in t and linear in n, r.

* ϵ is the constrained error between true subspace and estimated subspace: $\text{SE}(\hat{\mathbf{P}}, \mathbf{P}) := ||(\mathbf{I} - \hat{\mathbf{P}}\hat{\mathbf{P}}^T)\mathbf{P}||_2 \leq \epsilon$.

ADMM solver in an efficient way. A modification of PETRELS algorithm, where each row of the underlying subspace is updated in parallel, is then proposed to update subspace with a high accuracy.

- We are the first to provide a *strong guarantee* for the RST problem in the presence of both outliers and miss observations. Specifically, we show that a sequence of the objective values $\{f_t(\mathbf{U}_t)\}_{t=1}^{\infty}$ converges almost surely. The solutions $\{\mathbf{U}_t\}_{t=1}^{\infty}$ generated by PETRELS-ADMM converges to a stationary point of the expected loss function $f(\mathbf{U})$ asymptotically.

It is noted that a preliminary study has been presented in a conference version [30]. Compared to our earlier work, the problem formulation considered in Section II is more general. An upgrade of PETRELS-ADMM algorithm and its full *strong* convergence analysis are provided in this version. Furthermore, more extensive experiments on both synthesis data and real data are provided to illustrate the effectiveness of the proposed algorithm.

Compared to related works, there are several differences between PETRELS-ADMM and the state-of-the-arts RST algorithms which are what make our contributions significant. In particular, our mechanism for outlier rejection can facilitate the subspace estimation ability of RST algorithms where "clean" data involves the process only, thus improving overall performances. It is due to the fact that subspace learning or PCA is very sensitive to even a small fractions of outliers because of the quadratic norm minimization. The deviations from outliers will dominate the total norm and hence drive the basic components of the subspace. Excepting PETRELS-CFAR, the core of the state-of-the-art algorithms is "outlier-resistant", i.e., to have a "right" direction toward the true subspace, the algorithms have to require robust cost functions as well as additional adaptive parameter selection. For examples, GRASTA and ROSETA use the ℓ_1-norm robust estimator to reduce the effect of outliers while pROST applies the ℓ_0-norm one instead. Both the three algorithms then perform stochastic gradient decent on the Grassmannian of the underlying subspace. However, there is no guarantee that the ℓ_p-norm robust estimator (i.e., $p \in [0, 1]$) can provide an optimal solution because of non-convexity. Accordingly, the effect of outliers can not completely removed in tracking. This is why the algorithms fail when there are a large fractions of outliers in the measurement data or significant subspace changes in practice. By contrast, PETRELS-based algorithms can utilize advantages of the original PETRELS in missing observations and then treat outliers as missing data to facilitate the subspace tracking. Note that, PETRELS obtains the competitive performance in terms of subspace estimation accuracy in the case of "clean" data.

Our PETRELS-ADMM is more robust and efficient than PETRELS-CFAR. First, our ADMM solver may be efficient than CFAR in terms of memory cost and flexibility. The CFAR requires several previous observations to detect outliers[1], while our ADMM solver utilizes a new data vector and the previous estimated subspace and its memory cost is independent to the size of data samples. Moreover, performance of CFAR depends highly on predefined parameters such as the probability of false alarm P_{fa} and the size of training window N_ω [14]. By contrast, since the ADMM solver is a parameter free-type algorithm, our estimator is more flexible than PETRELS-CFAR. Secondly, PETRELS-CFAR may provide an unstable solution in the presence of a high corruption fraction. It is due to that PETRELS-CFAR uses the same line of the original PETRELS after removing outliers. Lack of regularization in the original PETRELS can result in an unstable solution when the fraction of missing data is large. Accordingly, convergence of PETRELS is confined to the full observation when it boils down to the PAST algorithm. In our work, an regularization of the $\ell_{2,\infty}$-norm of $\|\mathbf{U}\|_{2,\infty}^2$, which aims to control the maximum ℓ_2-norm of rows in $\mathbf{U}$, is therefore added in the objective function to avoid the scenario. In addition, an adaptive step size η_t is also applied to speed up the convergence rate as well as enhance the subspace estimation accuracy.

C. *Report organization*

The structure of the report is organized as follows. Section II states formulation for the RST problem dealing with data in the presence of both outliers and missing observations. Section III establishes our PETRELS-ADMM algorithm for RST and its theoretical convergence is analyzed in the Section IV. Section V presents extensive experiments to illustrate the effectiveness

[1] Recall that constant false alarm rate method (CFAR) [31] is a simple and efficient one for detecting target in radar systems

of PETRELS-ADMM compared to the state-of-the-art algorithms. We conclude the report in Section VI. Complete proofs are presented in the Appendix VIII.

D. Notations

In this report, we use lowercase (e.g. a), boldface lowercase (e.g. $\mathbf{a}$) letters to denote scalars, vectors, while capital boldface (e.g. $\mathbf{A}$) and calligraphic letters e.g. ($\mathcal{A}$) respectively denote matrices and sets. The i-th entry of a vector $\mathbf{a}$ is denoted by $\mathbf{a}(i)$. For a matrix $\mathbf{A}$, its (i,j)-th entry is denoted by $\mathbf{A}(i,j)$, while $\mathbf{A}_{:,k}$ and $\mathbf{A}_{l,:}$ are its k-th column and l-th row of $\mathbf{A}$ respectively. Operators $(.)^T, (.)^\dagger, \mathbb{E}[.], \mathrm{tr}(.)$ denote the transportation, pseudo-inverse, expectation, trace operator respectively. For $1 \leq p < \infty$, the ℓ_p-norm of a vector $\mathbf{a} \in \mathbb{R}^{n\times 1}$ is $\|\mathbf{a}\|_p \overset{\Delta}{=} \big(\sum_{i=1}^n |\mathbf{a}(i)|^p\big)^{1/p}$, meanwhile its ℓ_0-norm is $\|\mathbf{a}\|_0 \overset{\Delta}{=} \lim_{p\to 0}(\sum_{i=1}^n |\mathbf{a}(i)|^p)$ and its ℓ_∞-norm is $\|\mathbf{a}\|_\infty \overset{\Delta}{=} \max_i |a(i)|$. The $\ell_{2,\infty}$ of $\mathbf{A}$ is defined as the the maximum ℓ_2 row norm, i.e., $\|\mathbf{A}\|_{2,\infty} = \max_l \|\mathbf{A}_{l,:}\|_2$. The Frobenius norm of a matrix $\mathbf{A} \in \mathbb{R}^{n\times m}$ is $\|\mathbf{A}\|_F \overset{\Delta}{=} \big(\sum_{i=1}^n \sum_{j=1}^m \mathbf{A}(i,j)^2\big)^{1/2} = \sqrt{\mathrm{tr}(\mathbf{A}^T\mathbf{A})}$. The condition number of matrix $\mathbf{A}$ is $\kappa(\mathbf{A}) = \frac{\sigma_{\max(\mathbf{A})}}{\sigma_{\min(\mathbf{A})}}$, where $\sigma_{\max}(\mathbf{A})$ and $\sigma_{\min}(\mathbf{A})$ are maximal and minimal singular values of $\mathbf{A}$ respectively.

II. Problem Formulation

A. Robust Subspace Tracking

Assume that at each time instant t, we observe a signal $\mathbf{x}_t \in \mathbb{R}^{n\times 1}$ satisfying the following model:

$$\mathbf{x}_t = \mathbf{P}_t(\boldsymbol{\ell}_t + \mathbf{n}_t + \mathbf{s}_t), \tag{1}$$

where $\boldsymbol{\ell}_t \in \mathbb{R}^{n\times 1}$ is the true signal that lies in a low dimensional subspace of $\mathbf{U} \in \mathbb{R}^{n\times r}$ (i.e., $\boldsymbol{\ell}_t = \mathbf{U}\mathbf{w}_t$, $r \ll n$), $\mathbf{n}_t \in \mathbb{R}^{n\times 1}$ is a noise vector (e.g. $\mathcal{N}(0,\sigma)$), $\mathbf{s}_t \in \mathbb{R}^{n\times 1}$ is a sparse outlier vector which is somehow distributed in an ambient dimensional space, while the diagonal matrix $\mathbf{P}_t \in \mathbb{R}^{n\times n}$ is the observation mask showing whether the k-th entry of $\mathbf{x}_t$ is observed (i.e., $\mathbf{P}_t(k,k) = 1$) or not (i.e., $\mathbf{P}_t(k,k) = 0$).

Definition 1 (RST with Missing Data and Outliers): Given a set of observed signals, $\mathcal{X} = \{\mathbf{x}_i\}_{i=1}^t$, we wish to estimate a rank-r matrix $\mathbf{U}_t \in \mathbb{R}^{n\times r}$ such that it can cover the span of true signals $\{\boldsymbol{\ell}_i\}_{i=1}^t$.

The RST problem can be stated as the following minimization:

$$\begin{aligned} &\mathbf{U}_t = \underset{\mathbf{U}\in\mathbb{R}^{n\times r}}{\mathrm{argmin}}\, f_t(\mathbf{U}), \\ &\text{with } f_t(\mathbf{U}) \overset{\Delta}{=} \mathbb{E}_{\mathbf{x} \overset{\text{i.i.d}}{\sim} \mathbb{P}_{\text{empirical}}}[\ell(\mathbf{U},\mathbf{P},\mathbf{x})] = \frac{1}{t}\sum_{i=1}^{t} \lambda_i^{t-i}\ell(\mathbf{U},\mathbf{P}_i,\mathbf{x}_i), \end{aligned} \tag{2}$$

where $\mathbb{P}_{\text{empirical}}$ is the empirical data distribution and the loss function $\ell(\mathbf{U}, \mathbf{P}_i, \mathbf{x}_i)$ is given by

$$\ell(\mathbf{U}, \mathbf{P}_i, \mathbf{x}_i) \triangleq \min_{\mathbf{s},\mathbf{w}} \|\mathbf{P}_i(\mathbf{U}\mathbf{w} + \mathbf{s} - \mathbf{x}_i)\|_2^2 + \rho \|\mathbf{s}\|_1 , \tag{3}$$

where the ℓ_1-norm $\|\mathbf{s}\|_1$ associated with a regularization weight $\rho > 0$ is to control the outlier density (sparsity), and the forgetting factor $\lambda_i \in (0, 1]$ is to discount the effect of past observations. We usually prefer to minimize the expected cost $f(\mathbf{U})$ on signals distributed i.i.d from the true data-generating distribution $\mathbb{P}_{\text{data}}$, instead of the empirical cost $f_t(\mathbf{U})$ on $\mathbb{P}_{\text{empirical}}$ only. Thanks to the law of large numbers, we have average of the observations without discounting (i.e., $\lambda = 1$) converges to the expected value when t goes to infinity,

$$\begin{aligned} \mathbf{U} &= \operatorname*{argmin}_{\mathbf{U}\in\mathbb{R}^{n\times r}} f(\mathbf{U}) \\ \text{with } f(\mathbf{U}) &\triangleq \mathbb{E}_{\mathbf{x} \overset{\text{i.i.d}}{\sim} \mathbb{P}_{\text{data}}}[\ell(\mathbf{U}, \mathbf{P}, \mathbf{x})] = \lim_{t\to\infty} f_t(\mathbf{U}). \end{aligned} \tag{4}$$

From the past estimations $\{\mathbf{s}_i, \mathbf{w}_i\}_{i=1}^t$, instead of minimizing the empirical cost function $f_t(\mathbf{U})$ in (2), we propose to optimize the surrogate $g_t(\mathbf{U})$ of $f_t(\mathbf{U})$, which is defined by

$$g_t(\mathbf{U}) = \frac{1}{t} \sum_{i=1}^{t} \lambda^{t-i} \|\mathbf{P}_i(\mathbf{U}\mathbf{w}_i + \mathbf{s}_i - \mathbf{x}_i)\|_2^2 + \rho \|\mathbf{s}_i\|_1 . \tag{5}$$

Note that, the objective function provides an upper bound for $f_t(\mathbf{U})$ as $f_t(\mathbf{U}) \le g_t(\mathbf{U}),\ \forall t$. In our convergence analysis, we will prove that $f_t(\mathbf{U}_t)$ and $g_t(\mathbf{U}_t)$ converge almost surely to the same limit. As a result, the solution $\mathbf{U}_t$ obtained by minimizing $g_t(\mathbf{U})$ is exactly the solution of $f_t(\mathbf{U})$ can be when t tends to infinity.

B. *Discusses*

We have three remarks on the objective function statement above. First, the ℓ_0-norm regularization may be stronger, but more complicated than the ℓ_1-norm one. Since the ℓ_0-norm returns the number of nonzero entries in a vector, hence imposes on the entries in the same way. However, the ℓ_0-norm based sparsity control function is non-convex and the resulting RST will be a NP-hard problem [32]. While the ℓ_1-norm is the closest convex surrogate of the ℓ_0-norm, though the ℓ_1-norm relies highly on the magnitude of vector entries. There have been many extensions of ℓ_1-norm based (online) RPCA/RST approaches (see [6], [33], [34] for good reviews).

Second, the minimization (5) can be considered as a joint optimization of multiple variables including the subspace $\mathbf{U}$, coefficients $\mathbf{w}$ and outliers $\mathbf{s}$. Although the function is not jointly convex, it is (strongly) convex with respect to each of the variables while the others are fixed. As a result, the minimization (5) can be solved efficiently using the alternating minimization (AM) or alternating direction method of multipliers (ADMM) approaches [35]. Motivated by advantages of the ADMM framework in terms of convergence [**?**], [36], we derived an efficient algorithm, which will be presented in the next section, for handling the RST problem to handle missing data corrupted by outliers.

Third, performance of subspace tracking algorithms such as accuracy, stability, and complexity can depend on the forgetting factor λ. When the value of λ close to one, the algorithms can achieve good accuracy and stability, but their abilities in term of tracking and forgetting the past can be reduced. A smaller value of λ can improve the computational complexity and hence tracking but it may affect the accuracy of estimated subspaces. Generally, a constant forgetting factor λ is often used for this work such as the algorithm PETRELS (e.g. $\lambda = 0.98$) [4], [10] and ROBUSTA (e.g. $\lambda = 0.999$) [14], but variable forgetting factors can enhance the overall performance of the algorithms and control the memory in a more flexible way,

$$f_t(\mathbf{U}) = \lambda_t f_{t-1}(\mathbf{U}) + \ell(\mathbf{U}, \mathbf{P}_t, \mathbf{x}_t).$$

As a result, we modified the the original PETRELS in [10] by adding an adaptive step size $\eta_t \in (0, 1]$ at each time instant t (i.e., $\lambda_t = \eta_t \lambda$, see Section III-B). The modification provided a comparative performance in term of subspace estimation accuracy in practice. Also, we want to note that for the value of $\lambda_i \rightarrow 1$ and a large enough t (e.g. $t \rightarrow \infty$), it may be assumed that

$$f_t(\mathbf{U}) = \frac{1}{t} \sum_{i=1}^{t} \lambda_i^{t-i} \ell(\mathbf{U}, \mathbf{P}_i, \mathbf{x}_i) \cong \mathbb{E}[\ell(\mathbf{U}, \mathbf{P}, \mathbf{x})].$$

Therefore, our convergence analysis still holds for the variable forgetting factors of our algorithm.

C. Assumptions

We make the following assumptions for convenience of convergence analysis as well as helping deploy our optimization algorithm:

(A-1) The data-generation distribution $\mathbb{P}_{\text{data}}$ has a compact set $\mathcal{V}$, $\mathbf{x} \overset{\text{i.i.d}}{\sim} \mathbb{P}_{\text{data}}$. Real data are often bounded such as audio, image and video, hence the assumption (A-1) naturally holds.

(A-2) The constrained set $\mathcal{U} \subseteq \mathbb{R}^{n \times r}$ for the underlying subspace is $\mathcal{U} \triangleq \{\mathbf{U} \in \mathbb{R}^{n \times r}, \|\mathbf{U}_{k,:}\|_2 \leq 1, 1 \leq \kappa(\mathbf{U}) \leq \alpha\}$ with a constant α. The first constraint $\|\mathbf{U}_{k,:}\|_2 \leq 1$ is to bound the scale of basis vectors in $\mathbf{U}$ and hence prevent the arbitrarily very large values of $\mathbf{U}$. Therefore, there always exists a positive number $\epsilon > 0$ such that $\|\mathbf{U}_t\|_F \leq \epsilon, \forall t \geq 1$ (e.g. $\epsilon = \sqrt{n}$). While the low condition number of the subspace $\kappa(\mathbf{U})$ is to prevent the ill-conditioned computation. Furthermore, we also assume that the subspace change at two successive time instances is small, i.e., the largest principal angle between $\mathbf{U}_t$ and $\mathbf{U}_{t-1}$ is $0 \leq \theta_{\text{max}} \ll \pi/2$, or the distance between the two subspaces satisfies $0 \leq \text{SE}(\mathbf{U}_t, \mathbf{U}_{t-1}) = \sin(\theta_{\text{max}}) \ll 1$.

(A-3) The constrained set $\mathcal{W} \subseteq \mathbb{R}^{r \times 1}$ of coefficients is $\mathcal{W} = \{\mathbf{w} \in \mathbb{R}^{r \times 1}, \omega_1 \leq |w(i)| \leq \omega_2, i = 1, 2, \ldots, r\}$ with two constants $0 \leq \omega_1 < \omega_2$. Since the data $\mathbf{x}$ and subspace $\mathbf{U}$ are assumed to be bounded, it is natural that the subspace coefficient $\mathbf{w}$ is bounded.

(A-4) The constrained set $\mathcal{S}$ for outliers is $\mathcal{S} \triangleq \{\mathbf{s} \in \mathbb{R}^{n \times 1}, \|\mathbf{s}\|_\infty < C\}$. Theoretically, the ℓ_1-norm approximation can yield the sparse solution for $\mathbf{s}$, but it is not guaranteed that the solution is usually optimal with respect to the corresponding ℓ_0-norm one. In this work, we aim to estimate the locations of outliers correctly instead of their magnitude and then eliminate them. The constraint $\mathcal{S}$ imposed on the outlier magnitude can control the convergence rate of the proposed RST algorithm while still retaining the subspace estimation performance in practice. In some related works the bound C is often predefined, e.g. C can be chosen as $2^m - 1$, with m is the bit level for pixels in gray image/video applications. In addition, the assumption is also important to enhance the well-definedness as stated by the Proposition 1.

III. Proposed PETRELS-ADMM Algorithm

In this section, we present a novel algorithm, namely PETRELS-ADMM, for RST to handle missing data in the presence of outliers. The main idea is to develop a framework taken place in two sequential phases to minimize the empirical cost function $g_t(\mathbf{U})$ in Eq. (5). Specifically, outliers $\mathbf{s}_t$ and subspace $\mathbf{U}_t$ are alternatively updated at each time instant t.

Under the assumption (A-2) that the underlying subspace $\mathbf{U}$ changes slowly, we can detect outliers in $\mathbf{s}_t$ by projecting the new signal $\mathbf{x}_t$ into a space spanned by the previously estimated subspace $\mathbf{U}_{t-1}$ in the first phase. Specifically, we solve the following minimization problem:

$$\begin{aligned} &\mathbf{s}_t \triangleq \underset{\mathbf{s}\in\mathcal{S},\mathbf{w}\in\mathcal{W}}{\operatorname{argmin}} \,\tilde{\ell}(\mathbf{U}_{t-1}, \mathbf{P}_t, \mathbf{x}_t, \mathbf{w}, \mathbf{s}), \\ &\text{with } \tilde{\ell}_t(\mathbf{U}_{t-1}, \mathbf{P}_t, \mathbf{x}_t, \mathbf{w}, \mathbf{s}) = \|\mathbf{P}_t(\mathbf{U}_{t-1}\mathbf{w} + \mathbf{s} - \mathbf{x}_t)\|_2^2 + \rho\,\|\mathbf{s}\|_1 . \end{aligned} \tag{6}$$

Note that, the original loss function $\ell(\mathbf{U}_{t-1}, \mathbf{P}_t, \mathbf{x}_t)$ in Eq. (3) can be expressed w.r.t $(\mathbf{s}, \mathbf{w})$ as follows

$$\ell(\mathbf{U}_{t-1}, \mathbf{P}_t, \mathbf{x}_t) \triangleq \min_{\mathbf{s}\in\mathcal{S},\mathbf{w}\in\mathcal{W}} \tilde{\ell}_t(\mathbf{U}_{t-1}, \mathbf{P}_t, \mathbf{x}_t, \mathbf{w}, \mathbf{s}).$$

In the second phase, the subspace $\mathbf{U}_t$ can be estimated by minimizing the sum of squared residuals and account for outliers:

$$\mathbf{U}_t \triangleq \underset{\mathbf{U}\in\mathcal{U}}{\operatorname{argmin}} \frac{1}{t} \sum_{i=1}^{t} \lambda^{t-i} \|\mathbf{P}_i(\mathbf{U}\mathbf{w}_i + \mathbf{s}_i - \mathbf{x}_i)\|_2^2 \tag{7}$$

The following proposition help us to justify our assumptions in section II-C about the well-definedness of the RST problem which facilitates to derive several important results later in algorithm deployment and convergence analysis.

Proposition 1. *(Uniform bound of outliers and subspace): If* $\{\mathbf{U}_t, \mathbf{s}_t\}_{t=1}^{\infty}$ *be the sequence of optimal solutions of the minimization* (5)*, then they are bounded.*

Proof. Since $\mathbf{s}_t$ be the optimal solution of (6) defined as

$$\mathbf{s}_t \triangleq \underset{\mathbf{s}\in\mathcal{S},\mathbf{w}\in\mathcal{W}}{\operatorname{argmin}} \,\tilde{\ell}_t(\mathbf{U}_{t-1}, \mathbf{P}_t, \mathbf{x}_t, \mathbf{w}, \mathbf{s}),$$

we have the fact $\tilde{\ell}_t(\mathbf{U}_{t-1}, \mathbf{P}_t, \mathbf{x}_t, \mathbf{w}_t, \mathbf{s}_t) \leq \tilde{\ell}_t(\mathbf{U}_{t-1}, \mathbf{P}_t, \mathbf{x}_t, \mathbf{0}, \mathbf{0})$ and

$$\|\mathbf{s}_t\|_1 \leq \frac{1}{\rho}\|\mathbf{P}_t\mathbf{x}_t\|_2^2 = \frac{1}{\rho}\|\mathbf{x}_t\|_2^2$$

Moreover, under the assumption (A-1), it is natural that $\mathbf{s}_t$ is bounded.

Algorithm 1 Proposed PETRELS-ADMM

```
Input: A set of observed signal $\{\mathbf{x}_i\}_{i=1}^t, \mathbf{x}_i \in \mathbb{R}^{n \times 1}$, observation masks $\{\mathbf{P}_i\}_{i=1}^t, \mathbf{P}_i \in \mathbb{R}^{n \times n}$, true rank $r$.
procedure
    for $i = 1$ to $t$ do
    Estimate outliers $\mathbf{s}_i$ using Algorithm 2:
        $\mathbf{s}_i = \underset{\mathbf{s},\mathbf{w}}{\operatorname{argmin}} \|\mathbf{P}_i(\mathbf{U}_{i-1}\mathbf{w} + \mathbf{s} - \mathbf{x}_i)\|_2^2 + \rho \left\|\mathbf{s}\right\|_1 .$
    Recover signal $\mathbf{x}_i^{\text{re}}$: $\mathbf{x}_i^{\text{re}}(k) = \begin{cases} \frac{\|\mathbf{s}_i\|_0}{n}\mathbf{x}_i(k), & \text{if } \mathbf{s}_i(k) = 0, \\ 0, & \text{otherwise,} \end{cases}$
    Estimate subspace $\mathbf{U}_i$ using Algorithm 3:
        $\mathbf{U}_i = \underset{\mathbf{U},\mathbf{w}}{\operatorname{argmin}} \sum_{j=1}^{i} \lambda^{i-j} \|\mathbf{P}_j(\mathbf{x}_j^{\text{re}} - \mathbf{U}\mathbf{w})\|_2^2 + \frac{\alpha}{2i}\|\mathbf{U}\|_{2,\infty}^2 .$
    end for
return $\mathbf{U}_t$
```

To exam the bound for $\mathbf{U}_t$, we rewrite the problem (7) as

$$\mathbf{U}_t := \underset{\mathbf{U} \in \mathbb{R}^{n \times r}}{\operatorname{argmin}} \tilde{g}_t(\mathbf{U}) = \frac{1}{t}\sum_{i=1}^{t} \lambda^{t-i}\|\mathbf{P}_i(\mathbf{U}\mathbf{w}_i + \mathbf{s}_i - \mathbf{x}_i)\|_2^2 + \frac{\alpha}{2t}\|\mathbf{U}\|_{2,\infty}^2 .$$

We also exploit the fact that $\tilde{g}_t(\mathbf{U}_t) \leq \tilde{g}_t(\mathbf{U})|_{\mathbf{U}=\mathbf{0}}$ because of $\mathbf{U}_t = \operatorname{argmin}_{\mathbf{U}} \tilde{g}_t(\mathbf{U})$. It implies that $\|\mathbf{U}_t\|_{2,\infty}^2 \leq \frac{2}{\alpha}\sum_{i=1}^{t} \lambda^{t-i}\|\mathbf{P}_i(\mathbf{s}_i - \mathbf{x}_i)\|_2^2$ or $\mathbf{U}_t$ is bounded. □

Our algorithm first applies the ADMM framework in [35], which has been widely used in previous works for solving (6), and then propose a modification of PETRELS [10] to handle (7). In the outlier rejection stage, we emphasize here that we propose to focus on augmenting $\mathbf{s}$ (as shown in (9)) to further annihilate outlier effect, unlike GRASTA and ROSETA which focus on augmenting $\mathbf{w}$ only. While, we modify the subspace update step in PETRELS by adding an adaptive step size $\eta_t \in (0, 1]$ at each time instance t, instead of a constant as in the original version. The modification can be seen as an approximate interpretation of Newton's method.

A. Online ADMM for Outlier Detection

We show in the following how to solve (6) step-by-step:

Update $\mathbf{s}_t$*:* To estimate outlier $\mathbf{s}_t$ given $\mathbf{w}$, we exploit that the fact that (6) can be cast into the ADMM form as follows:

$$\min_{\mathbf{u},\mathbf{s}} h(\mathbf{u}) + q(\mathbf{s}), \quad \text{subject to } \mathbf{u} - \mathbf{s} = \mathbf{0}, \tag{8}$$

Algorithm 2 Remove outliers $\mathbf{s}_t$

```
Input: Observed signal $\mathbf{x}_t \in \mathbb{R}^{n\times 1}$, observation mask $\mathbf{P}_t \in \mathbb{R}^{n\times n}$, the previous estimate $\mathbf{U}_{t-1} \in \mathbb{R}^{n\times r}$, maximum iteration $K$, penalty parameters $\rho_1, \rho_2$, absolute and relative tolerances $\epsilon_{\text{abs}}$ and $\epsilon_{\text{rel}}$.
Initialization:
    Choose $\{\mathbf{u}^0, \mathbf{s}^0, \mathbf{w}^0, \mathbf{z}^0, \mathbf{e}^0\}$ randomly.
    $\{\mathbf{r}^0, \mathbf{e}^0\} \leftarrow \mathbf{0}^n$
procedure
    for $k = 0$ to $K$ do
    Update $\mathbf{s}$:
        $\mathbf{u}^{k+1} = \frac{1}{1+\rho_1}\big(\mathbf{P}_t(\mathbf{x}_t - \mathbf{U}_{t-1}\mathbf{w}^k) - \rho_1(\mathbf{s}^k - \mathbf{r}^k)\big)$
        $\mathbf{s}^{k+1} = S_{\rho/\rho_1}(\mathbf{u}^{k+1} + \mathbf{r}^k)$
        $\mathbf{r}^{k+1} = \mathbf{r}^k + \mathbf{u}^{k+1} - \mathbf{s}^{k+1}$
    Update $\mathbf{w}$:
        $\mathbf{w}^{k+1} = (\mathbf{U}_{t-1}^T\mathbf{P}_t\mathbf{U}_{t-1})^\dagger\mathbf{U}_{t-1}^T\mathbf{P}_t(\mathbf{x}_t - \mathbf{s}^{k+1} + \mathbf{e}^k)$
        $\mathbf{z}^{k+1} = \mathbf{P}_t(\mathbf{U}_{t-1}\mathbf{w}^{k+1} + \mathbf{s}^{k+1} - \mathbf{x}_t)$
        $\mathbf{e}^{k+1} = \frac{\rho_2}{1+\rho_2}\mathbf{z}^{k+1} + \frac{1}{1+\rho_2}S_{1+\frac{1}{\rho_2}}(\mathbf{z}^{k+1})$
    Stopping criteria:
        if $\left\|\mathbf{s}^{k+1} - \mathbf{s}^k\right\|_2 < \sqrt{n}\epsilon_{\text{abs}} + \epsilon_{\text{rel}}\left\|\rho_1\mathbf{r}^{k+1}\right\|_2$ then break
    end for
return $\mathbf{s}^{k+1}$
```

where $\mathbf{u}$ is the additional decision variable, $h(\mathbf{u}) = \frac{1}{2}||\mathbf{P}_t(\mathbf{U}_{t-1}\mathbf{w} + \mathbf{u} - \mathbf{x}_t)||_2^2$ and $q(\mathbf{s}) = \rho\|\mathbf{s}\|_1$. The corresponding augmented Lagrangian with the dual variable vector $\boldsymbol{\beta}$ is thus given by

$$\mathcal{L}(\mathbf{s}, \mathbf{u}, \boldsymbol{\beta}) = q(\mathbf{s}) + h(\mathbf{u}) + \boldsymbol{\beta}^T(\mathbf{u} - \mathbf{s}) + \frac{\rho_1}{2}\|\mathbf{u} - \mathbf{s}\|_2^2, \tag{9}$$

where $\rho_1 > 0$ is the regularization parameter.[2] Let $\mathbf{r} = \boldsymbol{\beta}/\rho_1$ be the scaled dual variable, we can rewrite the Lagrangian (9) as follows

$$\mathcal{L}(\mathbf{s}, \mathbf{u}, \mathbf{r}) = q(\mathbf{s}) + h(\mathbf{u}) + \rho_1\mathbf{r}^T(\mathbf{u} - \mathbf{s}) + \frac{\rho_1}{2}\|\mathbf{u} - \mathbf{s}\|_2^2. \tag{10}$$

Note that, in GRASTA and ROSETA focus on augmenting $\mathbf{w}$ only.[3]

[2] It is referred to as the penalty parameter. Although convergence rate of the proposed algorithm is dependent on the chosen value, the effect of the penalty parameter is little in practice. It is also shown that ADMM method can converge for all values of the parameter within a few tens of iterations [35].

[3] In GRASTA [9] and ROSETA [13], both the authors aimed to detect outliers $\mathbf{s}$ by solving the augmented Lagrangian of (6) as follows

$$\mathcal{L}(\mathbf{s}, \mathbf{y}, \mathbf{w}) = \|\mathbf{s}\|_1 + \frac{\rho}{2}\|\mathbf{P}_t(\mathbf{U}_{t-1}\mathbf{w} + \mathbf{s} - \mathbf{x}_t)\|_2^2 + \mathbf{y}^T(\mathbf{P}_t(\mathbf{U}_{t-1}\mathbf{w} + \mathbf{s} - \mathbf{x}_t)), \tag{11}$$

Therefore, we have the following update rule using the scaled dual variable at each k-th iteration, as

$$\mathbf{u}^{k+1}= \underset{\mathbf{u}}{\operatorname{argmin}} \big(h(\mathbf{u}) + \rho_1(\mathbf{r}^k)^T(\mathbf{u}-\mathbf{s}^k) + \frac{\rho_1}{2}\|\mathbf{u}-\mathbf{s}^k\|_2^2\big), \tag{12}$$

$$\mathbf{s}^{k+1}= \underset{\mathbf{s}}{\operatorname{argmin}} \big(q(\mathbf{s}) - \rho_1(\mathbf{r}^k)^T\mathbf{s} + \frac{\rho_1}{2}\|\mathbf{u}^{k+1}-\mathbf{s}\|_2^2\big), \tag{13}$$

$$\mathbf{r}^{k+1} = \mathbf{r}^k + \mathbf{s}^{k+1} - \mathbf{u}^{k+1}. \tag{14}$$

In particular, we first exploit that the minimization (12) can be formulated as a convex quadratic form:

$$\begin{aligned} \mathbf{u}^{k+1} &= \underset{\mathbf{u}}{\operatorname{argmin}} \left(\frac{1}{2}\left\|\mathbf{u}-\mathbf{P}_t(\mathbf{x}_t-\mathbf{U}_{t-1}\mathbf{w})\right\|_2^2 + \rho_1(\mathbf{r}^k)^T\mathbf{u} + \frac{\rho_1}{2}\|\mathbf{u}-\mathbf{s}^k\|_2^2\right) \\ &= \underset{\mathbf{u}}{\operatorname{argmin}} \left(\frac{1+\rho_1}{2}\left\|\mathbf{u}\right\|_2^2 - [\mathbf{P}_t(\mathbf{x}_t-\mathbf{U}_{t-1}\mathbf{w}) - \rho_1(\mathbf{s}^k-\mathbf{r}^k)]^T\mathbf{u}\right) \\ &= \frac{1}{1+\rho_1}\big(\mathbf{P}_t(\mathbf{x}_t-\mathbf{U}_{t-1}\mathbf{w}) - \rho_1(\mathbf{s}^k-\mathbf{r}^k)\big). \end{aligned} \tag{15}$$

While the problem (13) is truly a standard proximal minimization with the ℓ_1-norm [37] as

$$\begin{aligned} \mathbf{s}^{k+1} &:= \underset{\mathbf{s}}{\operatorname{argmin}} \ \big(\rho\left\|\mathbf{s}\right\|_1 + \frac{\rho_1}{2}\|\mathbf{s}-(\mathbf{u}^{k+1}+\mathbf{r}^k)\|_2^2\big) \\ &= S_{\rho/\rho_1}(\mathbf{u}^{k+1}+\mathbf{r}^k), \end{aligned} \tag{16}$$

where $S_\alpha(x)$ is the soft thresholding, defined as

$$S_\alpha(x) = \begin{cases} 0, & \text{if } |x| \le \alpha, \\ x-\alpha, & \text{if } x > \alpha, \\ x+\alpha, & \text{if } x < -\alpha, \end{cases}$$

which is a proximity operator of the ℓ_1-norm [37].

Finally, a simple update rule for the scaled dual variable $\mathbf{r}$ can be given by

$$\mathbf{r}^{k+1} = \mathbf{r}^k - \beta^k\nabla_{\mathcal{L}}(\mathbf{r}^k), \tag{17}$$

where the gradient $\nabla_{\mathcal{L}}(\mathbf{r}^k)$ is computed by $\nabla_{\mathcal{L}}(\mathbf{r}^k) = \rho_1(\mathbf{u}^{k+1}-\mathbf{s}^{k+1})$ and $\beta^k > 0$ is the step size controlling the convergence rate. For the method of multipliers in general and ADMM method in particular, the step size for the dual variable update can be chosen to be equal the penalty parameter [35]. Therefore, the step size β^k is here set to be $\beta^k = 1/\rho_1$ at the k-th iteration because of the scaled version.

Update $\mathbf{w}_t$*:* To estimate $\mathbf{w}_t$ given $\mathbf{s}$, (6) can be recast into the following ADMM form

$$\begin{aligned} &\min_{\mathbf{w}\in\mathcal{W},\mathbf{e}\in\mathbb{R}^{n\times1}} \frac{1}{2}\left\|\mathbf{P}_t(\mathbf{U}_{t-1}\mathbf{w}+\mathbf{s}-\mathbf{x}_t)\right\|_2^2 + y(\mathbf{e}) \\ &\text{subject to} \quad \mathbf{P}_t(\mathbf{U}_{t-1}\mathbf{w}+\mathbf{s}-\mathbf{x}_t) - \mathbf{e} = 0 \end{aligned} \tag{18}$$

where $y(\mathbf{e})$ is a convex regularizer function for the noise $\mathbf{e}$, (e.g. $y(\mathbf{e}) = \frac{\sigma}{2}\left\|\mathbf{e}\right\|_2^2$, with σ^{-1} can be chosen as the signal to noise ratio, SNR). However, the formulation (18) is still affected by outliers because s may not be completely rejected in each iteration. Therefore, (18) can be cast further into the ADMM form such that it can lie between least squares (LS) and least absolute deviations to reduce the impact of outliers. The Huber fitting can bring transition between the quadratic and absolute terms of $\mathcal{L}_{\mathbf{w},\mathbf{e}}(\mathbf{w},\mathbf{e},.)$, as

$$\mathcal{L}_{\mathbf{w},\mathbf{e}}(\mathbf{w},\mathbf{e},.) = f^{\text{Hub}}(\mathbf{e}) + \frac{\rho_2}{2}\left\|\mathbf{P}_t(\mathbf{U}_{t-1}\mathbf{w} + \mathbf{s} - \mathbf{x}_t) - \mathbf{e}\right\|_2^2), \tag{19}$$

where the Huber function can be given [35],

$$f^{\text{Hub}}(x) = \begin{cases} x^2/2, & |x| \le 1, \\ |x| - 1/2, & |x| > 1. \end{cases}$$

As a result, e-updates for estimating $\mathbf{w}$ involve the proximity operator of the Huber function, that is,

$$\mathbf{e}^{k+1} = \frac{\rho_2}{1+\rho_2}\mathbf{P}_t(\mathbf{U}_{t-1}\mathbf{w}^{k+1} + \mathbf{s} - \mathbf{x}_t) + \frac{1}{1+\rho_2}S_{1+\frac{1}{\rho_2}}(\mathbf{P}_t(\mathbf{U}_{t-1}\mathbf{w}^{k+1} + \mathbf{s} - \mathbf{x}_t)),$$

Hence, at the $(k+1)$-th iteration, $\mathbf{w}^{k+1}$ can be updated using the following closed-form solution of the convex quadratic function:

$$\mathbf{w}^{k+1} = (\mathbf{U}_{t-1}^T\mathbf{P}_t\mathbf{U}_{t-1})^{\dagger}\mathbf{U}_{t-1}^T\mathbf{P}_t(\mathbf{x}_t - \mathbf{s} + \mathbf{e}^k),$$

where $(.)^{\dagger}$ denotes the matrix pseudo-inversion operator. To sum up, the rule for updating $\mathbf{w}_t$ can be given by

$$\mathbf{w}^{k+1} = (\mathbf{U}_{t-1}^T\mathbf{P}_t\mathbf{U}_{t-1})^{\dagger}\mathbf{U}_{t-1}^T\mathbf{P}_t(\mathbf{x}_t - \mathbf{s} + \mathbf{e}^k), \tag{20}$$

$$\mathbf{z}^{k+1} = \mathbf{P}_t(\mathbf{U}_{t-1}\mathbf{w}^{k+1} + \mathbf{s} - \mathbf{x}_t), \tag{21}$$

$$\mathbf{e}^{k+1} = \frac{\rho_2}{1+\rho_2}\mathbf{z}^{k+1} + \frac{1}{1+\rho_2}S_{1+\frac{1}{\rho_2}}(\mathbf{z}^{k+1}). \tag{22}$$

We can use a parameter $\nu > 0$ is to ensure that the matrix $\mathbf{U}_{t-1}^T\mathbf{P}_t\mathbf{U}_{t-1} + \nu\mathbf{I}$ is invertible in Eq. (20) as well as to satisfy the constraint on subspace coefficients, i.e., $\mathbf{w} \in \mathcal{W}$. We note that, by using the Huber fitting operator, our algorithm is better in reducing the impact of outliers than GRASTA and ROSETA which use ℓ_2-norm regularization.

The procedure is stopped when the maximum iteration has reached or the accuracy tolerance for the primal residual and dual norm has met:

$$\left\|\mathbf{s}^{k+1} - \mathbf{s}^k\right\|_2 < \sqrt{n}\epsilon_{\text{abs}} + \epsilon_{\text{rel}}\left\|\rho_1\mathbf{r}^{k+1}\right\|_2,$$

where $\epsilon_{\text{abs}} > 0$ and $\epsilon_{\text{rel}} > 0$ are predefined tolerances for absolute and relative part respectively. A reasonable range for the absolute tolerance may be $[10^{-6}, 10^{-3}]$, while $[10^{-4}, 10^{-2}]$ is good for the relative tolerance, see [35] for further details of the stopping criterion.

Algorithm 3 Modified PETRELS for updating $\mathbf{U}_t$

```
Input: Recovered signals $\{\mathbf{x}_i^{\text{fre}}\}_{i=1}^t$, observation mask $\mathbf{P}_t$, the previous estimate $\mathbf{U}_{t-1}$,
    forgetting factor $\lambda$, the step size $\eta$, the previous Hessian $\mathbf{H}_{t-1}$.
procedure
    $\mathbf{w}_t = (\mathbf{P}_t^{\text{re}}\mathbf{U}_{t-1}) \setminus \mathbf{x}_t^{\text{fre}}$
    $x_t = \dfrac{\big\|\mathbf{x}_t^{\text{fre}} - \mathbf{P}_t^{\text{re}}\mathbf{U}_{t-1}\mathbf{w}_t\big\|_2}{\|\mathbf{w}_t\|_2}$
    $\eta_t = \dfrac{x_t}{\sqrt{x_t^2+1}}$
    if $\eta_t > \eta$ then $\eta_t = 1$ end if
    for $m = 1$ to $n$ do
        $\mathbf{H}_t^m = \lambda\mathbf{H}_{t-1}^m + \mathbf{P}_t^{\text{re}}(m,m)\mathbf{w}_t\mathbf{w}_t^T$
        $\mathbf{R}_t^m = \mathbf{H}_t^m + \alpha\big(\frac{1}{2t} - \frac{\lambda_t}{2(t-1)}\big)\mathbf{I}$
        $\mathbf{a}_t = \mathbf{R}_{t-1}^m \setminus \mathbf{w}_t$
        $\mathbf{u}_t^m = \mathbf{u}_{t-1}^m + \eta_t\ \mathbf{P}_t^{\text{re}}(m,m)(\mathbf{x}_t^{\text{re}}(m) - \mathbf{w}_t^T\mathbf{u}_{t-1}^m)\mathbf{a}_t$
    end for
return $\mathbf{U}_t$
```

B. Modified PETRELS for Subspace Estimation

Having estimated $\mathbf{s}_t$, we can rewrite the optimization (7) as

$$\mathbf{U}_t := \underset{\mathbf{U}}{\operatorname{argmin}} \frac{1}{t}\sum_{i=1}^{t}\lambda^{t-i}\|\mathbf{P}_i^{\text{re}}(\mathbf{x}_i^{\text{re}} - \mathbf{U}\mathbf{w}_i)\|_2^2 + \frac{\alpha}{2t}\|\mathbf{U}\|_{2,\infty}^2, \tag{23}$$

where the recovered signal $\mathbf{x}_i^{\text{re}}$ and the new observation $\mathbf{P}_i^{\text{re}}$ are determined by the following rule:

- if $\mathbf{s}_i(k) = 0$, then $\mathbf{x}_i^{\text{re}}(k) = \frac{\|\mathbf{s}_i\|_0}{n}\mathbf{x}_i(k)$,
- if $\mathbf{s}_i(k) \neq 0$, then $\mathbf{P}_i^{\text{re}}(k,k) = 0$,

and the outliers $\mathbf{s}_i$ can be eliminated.

Thanks to the parallel scheme of PETRELS [10], the optimal solution of the problem (23) can be obtained by solving its subproblems at each row $\mathbf{u}^m$ of $\mathbf{U}$, $m = 1, 2, \dots, n$, that is,

$$\mathbf{u}^m = \underset{\mathbf{u}^m \in \mathbb{R}^{r\times 1}}{\operatorname{argmin}} \sum_{i=1}^{t}\lambda^{t-i}\mathbf{P}_i^{\text{re}}(m,m)(\mathbf{x}_i^{\text{re}}(m) - \mathbf{w}_i^T\mathbf{u}^m)^2 + \frac{\alpha}{2t}\left\|\mathbf{u}^m\right\|_2^2. \tag{24}$$

In this way, we can speed up the subspace update by ignoring the $\mathbf{u}^m$ if the m-th entry of $\mathbf{x}_t^{\text{re}}$ is labeled as missing observation or outlier. The update is summarized in Algorithm 3. Note that, we relax the recursive update rule for each row $\mathbf{u}^m$ by adding an adaptive step size $\eta_t \in (0,1]$ at each time instance t, instead of a constant as in the original PETRELS [10] and the simplified PETRELS [4], i.e.,

$$\mathbf{u}_t^m = \mathbf{u}_{t-1}^m + \eta_t\mathbf{P}_t^{\text{re}}(m,m)(\mathbf{x}_t^{\text{re}}(m) - \mathbf{w}_t^T\mathbf{u}_{t-1}^m)\mathbf{a}_t \tag{25}$$

where $\mathbf{R}_t^m = \sum_{i=1}^t \lambda^{t-i}\mathbf{P}_t(m,m)\mathbf{w}_i\mathbf{w}_i^T + \alpha\big(\frac{1}{2t} - \frac{\eta_t\lambda}{2(t-1)}\big)\mathbf{I}$, and $\mathbf{a}_t = (\mathbf{R}_t^m)^\dagger\mathbf{w}_t$ and the adaptive step size η_t is given by

$$\eta_t = \frac{x_t}{\sqrt{x_t^2+1}}, \text{ with } x_t = \frac{\|\mathbf{e}_t\|_2}{\|\mathbf{w}_t\|_2}, \tag{26}$$

where the residual error $\mathbf{e}_t$ is computed by $\mathbf{e}_t = \mathbf{x}_t^{\text{re}} - \mathbf{P}_t^{\text{re}}\mathbf{U}_{t-1}\mathbf{w}_t$. Specifically, the adaptive step size η_t can be expressed by $\eta_t = \sin(\theta_t)$, see Fig. 1. The smaller angle θ_t is, we are closer to the true subspace, the smaller step size is needed.

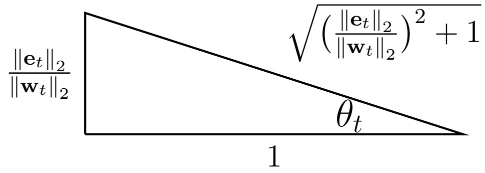

$\frac{\|\mathbf{e}_t\|_2}{\|\mathbf{w}_t\|_2}$
$\sqrt{\big(\frac{\|\mathbf{e}_t\|_2}{\|\mathbf{w}_t\|_2}\big)^2+1}$
θ_t
1

Fig. 1: Adaptive step size η_t.

The modification can be seen as an approximate interpretation of Newton's method which guarantee that the solution can converge to stationary point. The convergence analysis will be provided in the next section.

IV. Theoretical Analysis

In this section, we provide a convergence analysis of the proposed PETRELS-ADMM algorithm for RST problem dealing with missing data corrupted by outliers. Motivated by the results of convergence of empirical processes for online sparse coding in [38] and online robust PCA in [21], [24], we derive a theoretical approach to analyze the convergence of values of the objective function $\{f_t(\mathbf{U}_t)\}_{t=1}^\infty$ as well as the solutions $\{\mathbf{U}_t\}_{t=1}^\infty$ generated by PETRELS-ADMM. We note that, there are several differences between our work and the previous works. In particular, the work of [38] is to dedicate to sparse coding with a different mechanism where behaviors of its objective function and surrogate are to augment on sparse coefficients, while we focuses on controlling outlier sparsity and the underlying subspace. The work of [21], [24] can only guarantee in the case of data with full observations. In addition, the authors in [21], [24] assumed that the surrogate functions $g_t(\mathbf{U})$ are strongly convex and the minimizers of the s-update are unique (see Assumptions (A2-A3) therein), while we can prove them (see Proposition 3 and Lemma 1). Furthermore, there is a forgetting factor λ_t to enable the ability of forgetting the past and observation masks $\{\mathbf{P}_i\}_{i=1}^t$ in our objective function which lead to more challenges in convergence analysis. In addition, differences in term of optimization algorithms, constraints and assumptions induce many other differences in our proof compared with the previous works.

Given assumptions defined in Section II-C, our main theoretical result can be stated by the following theorem:

Theorem 1. *(Convergence of PETRELS-ADMM): Let $\{\mathbf{U}_t\}_{t=1}^{\infty}$ be the sequence of solutions generated by PETRELS-ADMM, then the sequence converges to a stationary point of the expected loss function $f(\mathbf{U})$ when $t \to \infty$.*

Proof Sketch. Our proof is derived similarly the line in [21], [38] which is divided into main stages as follows: We first prove that the solutions $\{\mathbf{U}_t, \mathbf{s}_t\}_{t\geq 1}$ generated by the PETRELS-ADMM algorithm are optimal and uniformly bounded. We then prove that a nonnegative sequence $\{g_t(\mathbf{U}_t)\}_{t=1}^{\infty}$ converges almost surely when $\{\mathbf{U}_t\}_{t=1}^{\infty}$ be the sequence of optimal solutions generated by the PETRELS-ADMM algorithm. After that, we prove that the surrogate $\{g_t(\mathbf{U}_t)\}_{t=1}^{\infty}$ converges almost surely to the that of the empirical loss function $\{f_t(\mathbf{U}_t)\}_{t=1}^{\infty}$ as well as the true loss function, i.e., $g_t(\mathbf{U}_t) \overset{a.s.}{\to} f_t(\mathbf{U}_t) \overset{a.s.}{\to} f(\mathbf{U}_t)$, thanks to the central limit theorem.

Due to space limit, we here present key results and report their proof sketch only, while the details of their proofs are provided in the Appendix VIII. □

Lemma 1. *(Convergence of Algorithm 2): At each time instant t, let $\{\mathbf{s}^k, \mathbf{u}^k, \mathbf{r}^k, \mathbf{w}^k, \mathbf{e}^k\}_{k=1}^{\infty}$ be a sequence generated by Algorithm 2 for outlier detection, there always exists a set of positive numbers $\{c_u, c_s, c_r, c_w, c_e\}$ at each iteration such that the minimizers satisfy*

$$\begin{aligned}\mathcal{L}(\mathbf{s}^{k+1}, \mathbf{u}^{k+1}, \mathbf{r}^{k+1}, \mathbf{w}^{k+1}, \mathbf{e}^{k+1}) \leq \; & \mathcal{L}(\mathbf{s}^k, \mathbf{u}^k, \mathbf{r}^k, \mathbf{w}^k, \mathbf{e}^k) - c_u\|\mathbf{u}^k - \mathbf{u}^{k+1}\|_2^2 \\ & - c_s\|\mathbf{s}^k - \mathbf{s}^{k+1}\|_2^2 - c_r\|\mathbf{r}^k - \mathbf{r}^{k+1}\|_2^2 \\ & - c_w\|\mathbf{w}^k - \mathbf{w}^{k+1}\|_2^2 - c_e\|\mathbf{e}^k - \mathbf{e}^{k+1}\|_2^2, \\ & \leq \mathcal{L}(\mathbf{s}^k, \mathbf{u}^k, \mathbf{r}^k, \mathbf{w}^k, \mathbf{e}^k),\end{aligned} \tag{27}$$

and the asymptotic variation of $\mathbf{s}^k$ *(i.e., outliers) is given by*

$$\lim_{k\to\infty} \left\|\mathbf{s}^{k+1} - \mathbf{s}^k\right\|_2^2 = 0. \tag{28}$$

Proof Sketch. We state the following proposition, which is the same line as in previous convergence analysis of ADMM algorithms [39], [40], to prove the first part of lemma 1 as follows

Proposition 2. *Let $\{\mathbf{s}^k, \mathbf{u}^k, \mathbf{r}^k, \mathbf{w}^k, \mathbf{e}^k\}_{k=1}^{\infty}$ be a sequence generated by Algorithm 2, then*

1) *The minimizer* $\mathbf{u}^{k+1}$ *defined in* (13) *satisfies*

$$\mathcal{L}(\mathbf{s}^k, \mathbf{u}^{k+1}, \mathbf{r}^k, \mathbf{w}^k, \mathbf{e}^k) \leq \mathcal{L}(\mathbf{s}^k, \mathbf{u}^k, \mathbf{r}^k, \mathbf{w}^k, \mathbf{e}^k) - c_u\|\mathbf{u}^k - \mathbf{u}^{k+1}\|_2^2.$$

2) *The minimizer* $\mathbf{s}^{k+1}$ *defined in* (16) *satisfies*

$$\mathcal{L}(\mathbf{s}^{k+1}, \mathbf{u}^{k+1}, \mathbf{r}^k, \mathbf{w}^k, \mathbf{e}^k) \leq \mathcal{L}(\mathbf{s}^k, \mathbf{u}^{k+1}, \mathbf{r}^k, \mathbf{w}^k, \mathbf{e}^k) - c_s\|\mathbf{s}^k - \mathbf{s}^{k+1}\|_2^2.$$

3) *The minimizer* $\mathbf{r}^{k+1}$ *defined in* (14) *satisfies*

$$\mathcal{L}(\mathbf{s}^{k+1}, \mathbf{u}^{k+1}, \mathbf{r}^{k+1}, \mathbf{w}^k, \mathbf{e}^k) \leq \mathcal{L}(\mathbf{s}^{k+1}, \mathbf{u}^{k+1}, \mathbf{r}^k, \mathbf{w}^k, \mathbf{e}^k) - c_r\|\mathbf{r}^k - \mathbf{r}^{k+1}\|_2^2.$$

4) *The minimizer* $\mathbf{w}^{k+1}$ *defined in* (20) *satisfies*

$$\mathcal{L}(\mathbf{s}^{k+1}, \mathbf{u}^{k+1}, \mathbf{r}^{k+1}, \mathbf{w}^{k+1}, \mathbf{e}^k) \leq \mathcal{L}(\mathbf{s}^{k+1}, \mathbf{u}^{k+1}, \mathbf{r}^{k+1}, \mathbf{w}^k, \mathbf{e}^k) - c_w\|\mathbf{w}^k - \mathbf{w}^{k+1}\|_2^2.$$

5) *The minimizer* $\mathbf{e}^{k+1}$ *defined in* (22) *satisfies*

$$\mathcal{L}(\mathbf{s}^{k+1},\mathbf{u}^{k+1},\mathbf{r}^{k+1},\mathbf{w}^{k+1},\mathbf{e}^{k+1}) \leq \mathcal{L}(\mathbf{s}^{k+1},\mathbf{u}^{k+1},\mathbf{r}^{k+1},\mathbf{w}^{k+1},\mathbf{e}^{k}) - c_e\|\mathbf{e}^k - \mathbf{e}^{k+1}\|_2^2.$$

As a result, the cluster $\{\mathbf{s}^k,\mathbf{u}^k,\mathbf{r}^k,\mathbf{w}^k,\mathbf{e}^k\}$ converges to stationary point of $\mathcal{L}(\mathbf{s},\mathbf{u},\mathbf{r},\mathbf{w},\mathbf{e})$ when $k\to\infty$ and it also implies that the sequence $\{\mathbf{s}_k\}_{k=0}^{\infty}$ is convergent, i.e.,

$$\sum_{k=0}^{\infty}\|\mathbf{s}^{k+1}-\mathbf{s}^k\|_2^2 < \infty \text{ or } \lim_{k\to\infty}\left\|\mathbf{s}^{k+1}-\mathbf{s}^k\right\|_2^2 = 0.$$

□

Corollary 1. *Let the sequence* $\{\mathbf{s}_t\}_{t\geq 1}$ *be solutions generated by the Algorithm 2, then it is uniformly bounded.*

Proposition 3. *(Convexity of the surrogate functions* $g_t(\mathbf{U})$*) : Given assumptions in Section II-C, the surrogate function* $g_t(\mathbf{U})$ *defined in Eq.* (5) *is not only strongly convex, but also Lipschitz function, i.e., there always exists two positive numbers* m_1 *and* m_2 *such that*

$$m_1\left\|\mathbf{U}_{t+1}-\mathbf{U}_t\right\|_F^2 \leq |g_t(\mathbf{U}_{t+1})-g_t(\mathbf{U}_t)| \leq m_2\left\|\mathbf{U}_{t+1}-\mathbf{U}_t\right\|_F. \tag{29}$$

Proof Sketch. To prove that $g_t(\mathbf{U})$ is strongly convex, we state the following facts: $g_t(\mathbf{U})$ is continuous and differentiable; its second derivative is a positive semi-definite matrix (i.e., $\nabla^2_{\mathbf{U}}g_t(\mathbf{U}) \geq m\mathbf{I}$); and the domain of $g_t(\mathbf{U})$ is convex. In order to satisfy the Lipschitz condition, we show that the first derivative of $g_t(\mathbf{U})$ is bounded. □

Lemma 2. *(Convergence of Algorithm 3): Given an outlier vector* $\mathbf{s}_t$ *generated by Algorithm 2 at each time instant t, Algorithm 3 can provide an local optimal solution* $\mathbf{U}_t$ *for minimizing* $g_t(\mathbf{U})$*. Moreover, the asymptotic variation of estimated subspaces* $\{\mathbf{U}_t\}_{t\geq 1}$ *is given by*

$$\left\|\mathbf{U}_t-\mathbf{U}_{t+1}\right\|_F \overset{a.s.}{\to} \mathcal{O}\left(\frac{1}{t}\right) \tag{30}$$

Proof Sketch. To establish the convergence, we exploit that our modification can be seen as an approximate interpretation of Newton's method,

$$\mathbf{U}_t \cong \mathbf{U}_{t-1} - \eta_t\big[\mathbf{H}\tilde{g}_t(\mathbf{U}_{t-1})\big]^{-1}\nabla\tilde{g}_t(\mathbf{U}_{t-1}) + \mathcal{O}\left(\frac{1}{t}\right),$$

where $\mathbf{H}\tilde{g}_t(\mathbf{U}_{t-1})$ and $\nabla\tilde{g}_t(\mathbf{U}_{t-1})$ are the Hessian matrix and gradient of the function $\tilde{g}_t(\mathbf{U})$ at $\mathbf{U}_{t-1}$. It implies that the estimated $\mathbf{U}_t$ converges to the stationary point of $g_t(\mathbf{U})$.

Furthermore, since $g_t(\mathbf{U})$ is strongly convex and Lipschitz function w.r.t the variable $\mathbf{U}$ as shown in Proposition 3, we have the following inequality

$$\begin{aligned} & m_1\left\|\mathbf{U}_{t+1}-\mathbf{U}_t\right\|_F^2 \leq |g_t(\mathbf{U}_{t+1})-g_t(\mathbf{U}_t)| \leq m_2\left\|\mathbf{U}_{t+1}-\mathbf{U}_t\right\|_F \\ & \Leftrightarrow \left\|\mathbf{U}_t-\mathbf{U}_{t+1}\right\|_F \leq \frac{m_2}{m_1} = \mathcal{O}\left(\frac{1}{t}\right). \end{aligned}$$

Note that the positive number $m_2 = \mathcal{O}(1/t)$ is already given in the proof of Proposition 3 in Appendix VIII-C, while m_1 is a constant. □

Corollary 2. *If $\{\mathbf{U}_t\}_{t\geq 1}$ be the sequence generated by PETRELS-ADMM, the sequence $\{\mathbf{U}_t\}_{t\geq 1}$ is uniformly bounded.*

Lemma 3. *(Convergence of the surrogate function $g_t(\mathbf{U})$): Let $\{\mathbf{U}_t\}_{t=1}^{\infty}$ be a sequence of solutions generated by Algorithm 1 at each time instant t, the sequence $\{g_t(\mathbf{U}_t)\}_{t=1}^{\infty}$converges almost surely, i.e.,*

$$\sum_{t=1}^{\infty} \big|\mathbb{E}[g_{t+1}(\mathbf{U}_{t+1}) - g_t(\mathbf{U}_t)]\big| < \infty \quad a.s. \tag{31}$$

Proof Sketch. We denote the stochastic process $\{u_t\}_{t\geq 1}, u_t \triangleq g_t(\mathbf{U}_t) \geq 0$ and prove that the sum of the positive difference of $\{u_t\}_{t\geq 1}$ is bounded, i.e.,

$$\sum_{t=1}^{\infty} \big|\mathbb{E}[u_{t+1} - u_t]\big| < \infty \quad a.s.$$

In particular, we have the following inequality

$$\mathbb{E}[u_{t+1} - u_t] \leq \underbrace{\mathbb{E}[\sqrt{t}(f(\mathbf{U}_t) - f_t(\mathbf{U}_t))]}_{\mathbb{E}[G_t(\mathbf{U}_t)]} \underbrace{\frac{1}{\sqrt{t}(t+1)}}_{a_t}.$$

In parallel, we exploit that $G_t(\mathbf{U}_t) = \sqrt{t}(f(\mathbf{U}_t) - f_t(\mathbf{U}_t))$ is the scaled and centered version of the empirical measure, which converges in distribution to a normal random variable, thanks to the center limit theorem. Therefore $\mathbb{E}[\sqrt{t}(f(\mathbf{U}_t) - f_t(\mathbf{U}_t))]$ is bounded. Furthermore, we also indicate that the sum $\sum_{t=1}^{\infty} a_t$ converges. The two facts result in the Lemma 3. □

Lemma 4. *(Convergence of the empirical loss function $f_t(\mathbf{U})$): The empirical loss function $\{f_t(\mathbf{U}_t)\}_{t=1}^{\infty}$ converges almost surely when $t \to \infty$, or*

$$g_t(\mathbf{U}_t) - f_t(\mathbf{U}_t) \xrightarrow{a.s.} 0. \tag{32}$$

Proof Sketch. We begin the proof with providing the following inequality:

$$\frac{g_t(\mathbf{U}_t) - f_t(\mathbf{U}_t)}{t+1} =\leq \underbrace{u_t - u_{t+1}}_{\text{(S-1)}} + \underbrace{\frac{\ell(\mathbf{U}_t, \mathbf{P}_{t+1}, \mathbf{x}_{t+1}) - f_t(\mathbf{U}_t)}{t+1}}_{\text{(S-2)}}$$

We then prove that two sequences (S-1)-(S-2) converge almost surely. As a result, the sequence $\big\{(g_t(\mathbf{U}_t) - f_t(\mathbf{U}_t))\frac{1}{t+1}\big\}$ also convergence almost surely, i.e.,

$$\sum_{t=0}^{\infty} \big(g_t(\mathbf{U}_t) - f_t(\mathbf{U}_t)\big)\frac{1}{t+1} < \infty.$$

In parallel, we exploit that the real sequence $\{\frac{1}{t+1}\}_{t\geq 1}$ diverges, i.e., $\sum_{t=1}^{\infty} \frac{1}{t+1} = \infty$. It implies that $g_t(\mathbf{U}_t) - f_t(\mathbf{U}_t)$ converges.

□

Corollary 3. *The expected loss function $\{f(\mathbf{U}_t)\}_{t=1}^{\infty}$ converges almost surely when $t \to \infty$.*

$$g_t(\mathbf{U}_t) - f(\mathbf{U}_t) \xrightarrow{a.s.} 0. \tag{33}$$

Proof. Since $f_t(\mathbf{U}_t) \overset{a.s.}{\to} f(\mathbf{U}_t)$ and $g_t(\mathbf{U}_t) \overset{a.s.}{\to} f_t(\mathbf{U}_t)$, then $g_t(\mathbf{U}_t) \overset{a.s.}{\to} f(\mathbf{U}_t)$. Since $g_t(\mathbf{U}_t)$ converges almost surely, $f(\mathbf{U}_t)$ also converges almost surely when $t \to \infty$. □

V. Experiments

In this section, we evaluate performance of the proposed algorithm by comparing to state-of-the-arts in three scenarios: robust subspace tracking, robust matrix completion and video background-foreground separation. In particular, extensive experiments on synthesis data are carried on to demonstrate the convergence and robustness of our PETRELS-ADMM algorithm as well as the state of the art algorithms for subspace tracking and matrix completion. While four real video sequences are used to illustrate the effectiveness of PETRELS-ADMM for background-foreground separation.

A. Robust Subspace Tracking

In the following experiments, data $\mathbf{x}_t$ at each time t is generated randomly using the standard signal model as in Eq. (1)

$$\mathbf{x}_t = \mathbf{P}_t(\mathbf{A}\boldsymbol{\omega}_t + \mathbf{n}_t + \mathbf{s}_t),$$

where $\mathbf{A} \in \mathbb{R}^{n \times r}$ denotes a mixing matrix, $\boldsymbol{\omega}_t$ is a random vector living on $\mathbb{R}^r$ space (i.e., $\boldsymbol{\ell} = \mathbf{A}\boldsymbol{\omega}_t$) and both of them are Gaussian i.i.d. $\mathcal{N}(0,1)$; $\mathbf{n}_t$ presents the white Gaussian noise $\mathcal{N}(0,\sigma^2)$, with $\mathrm{SNR} = -10\log_{10}(\sigma^2)$ is the signal-to-noise ratio to control the impact of noise on algorithm performance; $\mathbf{P}_t$ is the observation mask showing the percentage of observed entries in $\mathbf{x}_t$; and $\mathbf{s}_t$ is uniform i.i.d. over $[0, 1.(\text{fac-outlier})]$ given the magnitude fac-outlier of outliers that aim to create a space for outliers.

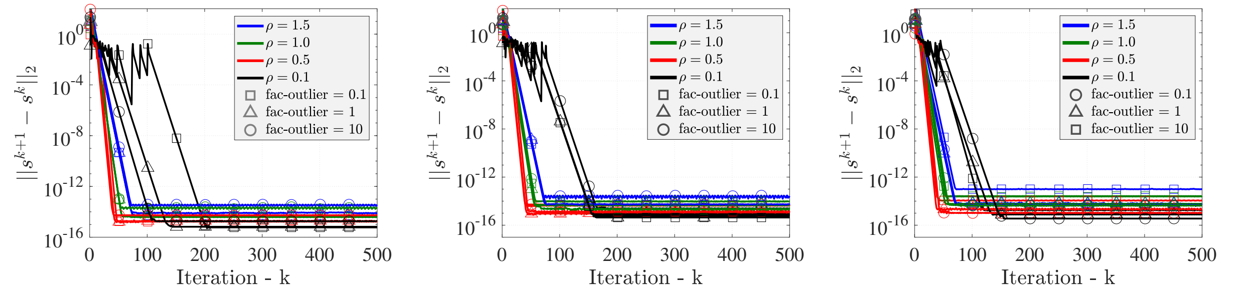
In order to evaluate the subspace estimation accuracy, we use the subspace estimation performance (SEP) [41], [42] metric

$$\mathrm{SEP} = \frac{1}{L}\sum_{i=1}^{L} \frac{\mathrm{tr}\{\mathbf{U}_{\text{es-i}}^T(\mathbf{I} - \mathbf{U}_{\text{ex}}\mathbf{U}_{\text{ex}}^T)\mathbf{U}_{\text{es-i}}\}}{\mathrm{tr}\{\mathbf{U}_{\text{es-i}}^T(\mathbf{U}_{\text{ex}}\mathbf{U}_{\text{ex}}^T)\mathbf{U}_{\text{es-i}}\}},$$

and subspace error (SE) metric, also referred to as the distance between two subspaces [43],

$$\mathrm{SE} = \frac{1}{L}\sum_{i=1}^{L}\sqrt{1-\cos^2(\theta_i)}, \text{ with } \cos(\theta_i) = \max_{\mathbf{u}\in\mathbf{U}_{\text{ex}}}\max_{\mathbf{v}\in\mathbf{U}_{\text{es-i}}}\frac{\mathbf{u}^T\mathbf{v}}{\|\mathbf{u}\|\,\|\mathbf{v}\|},$$

where L is the number of independent runs, $\mathbf{U}_{\text{ex}}$ and $\mathbf{U}_{\text{es-i}}$ are the true and the estimated subspaces at the i-th run respectively. The lower SEP and SE are, the better performance of the algorithm achieves.



(a) SNR = 0 dB (b) SNR = 10 dB (c) SNR = 20 dB

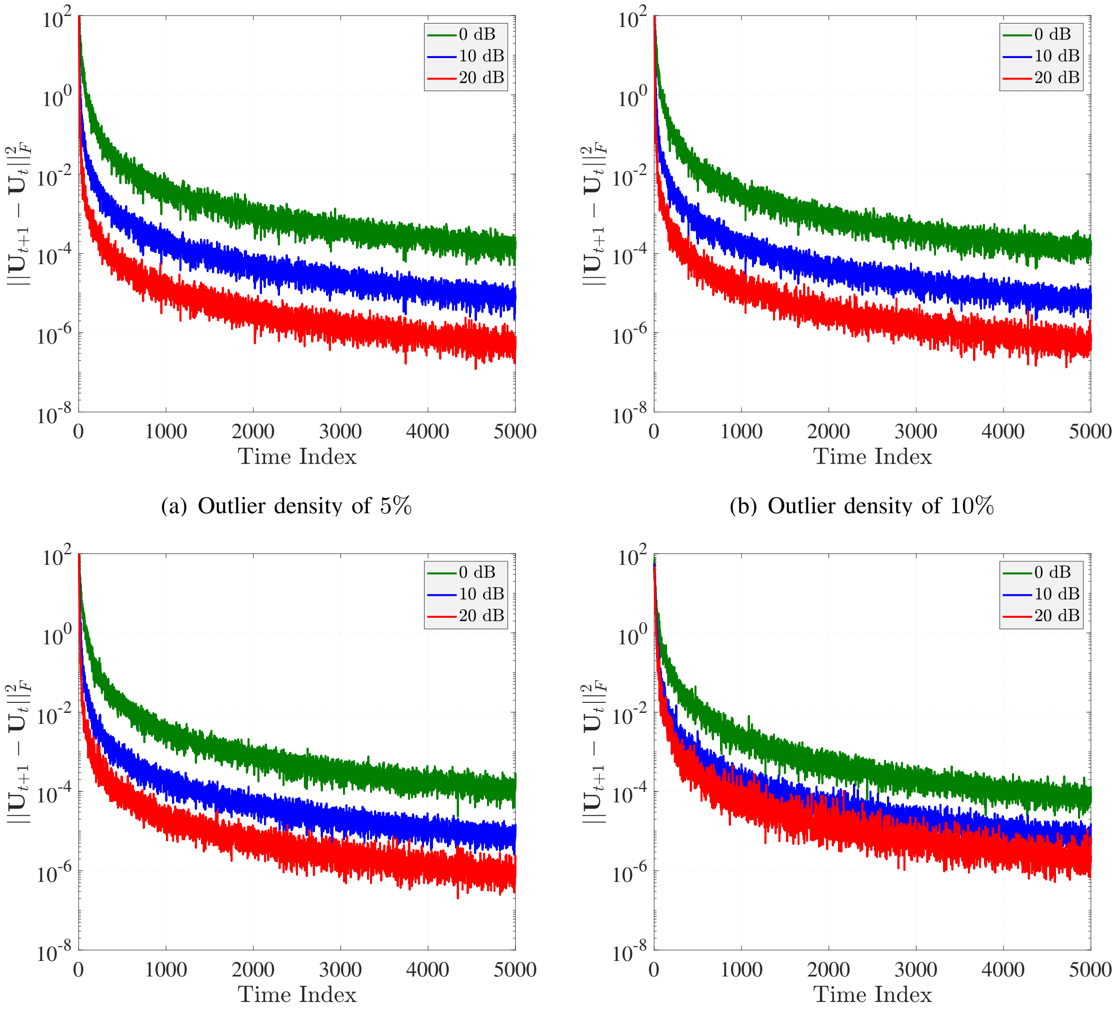
Fig. 2: Convergence of PETRELS-ADMM in terms of the variation $\left\|\mathbf{s}^{k+1} - \mathbf{s}^{k}\right\|_2$: $n = 50, r = 2$, 90% entries observed and outlier density of 5%.

State-of-the-art algorithms for comparison are: GRASTA [9], ROSETA [13] and PETRELS-CFAR [14], ReProCS [15]. To have a fair comparison, the parameters of these algorithms are set default and these codes are available online[4]. The experimental results are averaged over 100 independent runs. The experiments are conducted in following investigations:

1) Convergence of PETRELS-ADMM: To demonstrate the convergence of our algorithm, we use a synthesis data whose number of row $n = 50$, rank $r = 2$, and 5000 observations with 90% entries observed on average. Specifically, the outlier density is varied from 5% to 40%, while the outlier intensity is set at three values of low, medium and high level (i.e., $\text{fac-outlier} = 0.1, 1$ and 10 respectively). The regularization weight ρ varies in the range $[0.1, 1.5]$. Also, three noise levels are considered, with $\text{SNR} \in \{0, 10, 20\}$ dB. The results are shown as in Fig. 2, Fig. 3 and Fig. 4.

Fig. 2 shows the typical convergence behavior of PETRELS-ADMM w.r.t the two variables: fac-outlier and the weight ρ. We can see that, the variation of $\{\mathbf{s}^k\}_{k\geq 1}$ always converges in all testing cases (i.e., approximate 10^{-14} on average). When the regularization weight $\rho \geq 0.5$, the convergence rate is fast which the variation $\left\|\mathbf{s}^{k+1} - \mathbf{s}^{k}\right\|_2$ can converge in 50 iterations in both low- and high-noise cases. The results are practical evidences of the Lemma 1. Similarly, variations of the sequence $\{\mathbf{U}_t\}_{t\geq 0}$ generated by PETRELS-ADMM also have asymptotic converged behavior as shown in Fig. 3. The convergences of $\{\mathbf{U}_t\}_{t\geq 0}$ are also verified by comparing to the original PETRELS [10] with perfect reconstructions as in Fig. 4. Clearly, our PETRELS-ADMM outperforms the original PETRELS with full observation in terms of both SEN and SE metrics, albeit the convergence rate of the original PETRELS is faster than that of our algorithm. Note that, the original PETRELS can converge to the global optima in the full observation regime given a deterministic noise level [10]. Therefore, it testifies the robustness of PETRELS-ADMM in high-noise cases.

[4] GRASTA: https://sites.google.com/site/hejunzz/grasta
ROSETA: http://www.merl.com/research/license#ROSETA
ReProCS: https://github.com/praneethmurthy/ReProCS
Our codes: https://github.com/thanhtbt/RST



(a) Outlier density of 5%

(b) Outlier density of 10%

(c) Outlier density of 20%

(d) Outlier density of 40%

Fig. 3: Convergence of PETRELS-ADMM in terms of the variation $\|\mathbf{U}_{t+1} - \mathbf{U}_t\|_F$: $n = 50, r = 2$, 90% entries observed and outlier intensity $\text{fac-outlier} = 10$.

2) Outlier Detection: Follow the experiment above, we verify the ability of PETRELS-ADMM for outlier detection on the same synthesis data. Outliers $\{\mathbf{s}_t\}_{t\geq 0}$ are uniform i.i.d. over $[0, 1.(\text{fac-outlier})]$ given the magnitude fac-outlier of outliers which is fixed at 5. The results are shown as in Fig. 5. We can see that, the location of outliers $\mathbf{s}_t$ are detected completely even when the measurement data is mixed by noise with a high SNR value (e.g. 10 dB). Also, amplitude of the outliers is recovered nearly correctly with a small relative error $(\text{RE} = \frac{\|\mathbf{s}_t - \tilde{\mathbf{s}}_t\|_2}{\|\mathbf{s}_t\|_2})$ in both cases (e.g. $\text{RE} = 0.0635$ at the 20 dB noise level). As a result, the corrupted signals are also well reconstructed, see Fig. 5(b) and (d).

A performance comparison of PETRELS-ADMM and GRASTA for outlier detection task is carried out to show the effectiveness of the proposed method. The synthesis data is also generated for the number of row, $n = 50$, rank $r = 2$ and 5000 observations. Outlier density

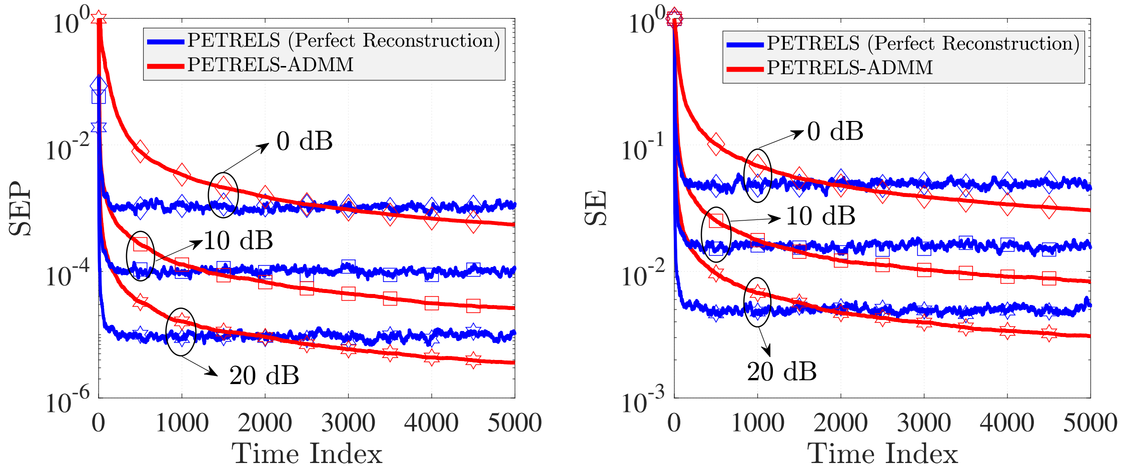
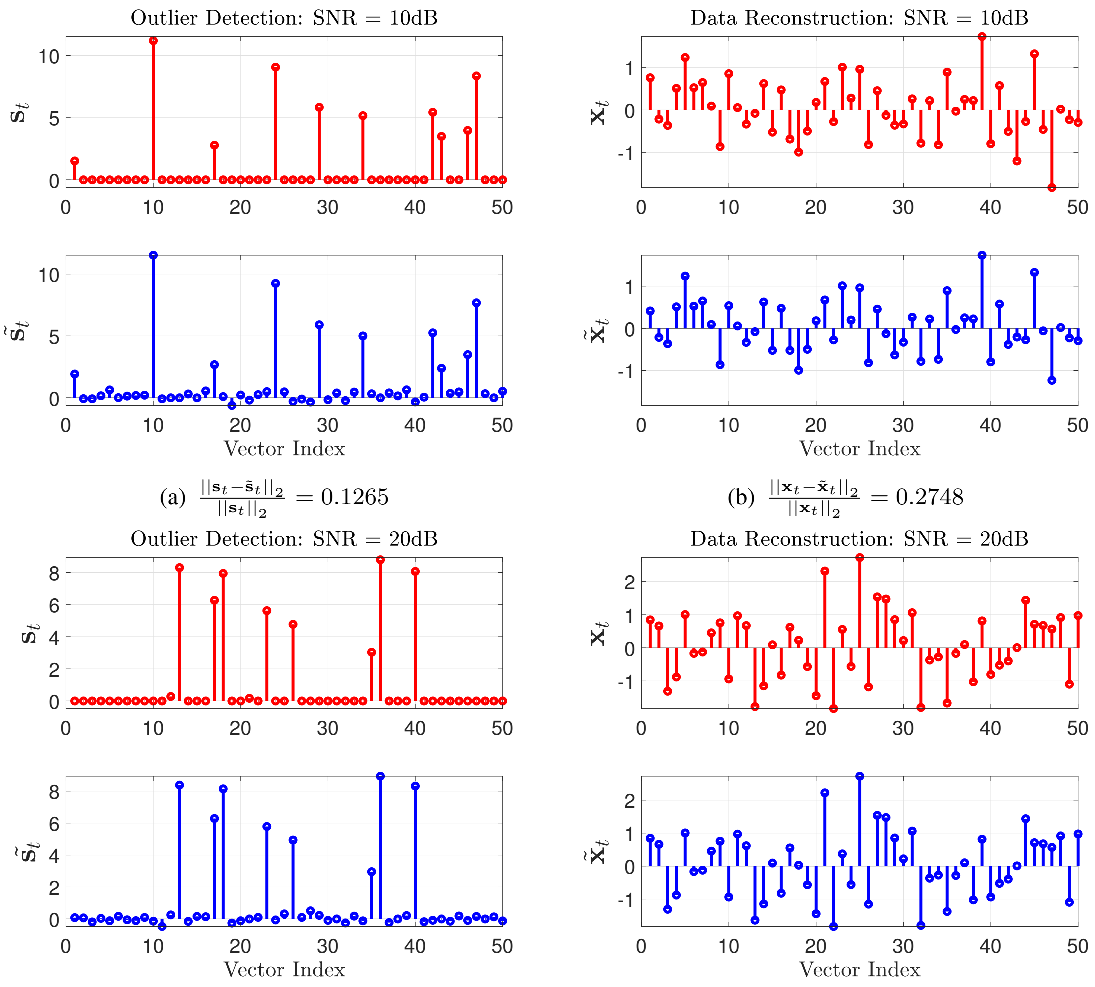


Fig. 4: Convergence of PETRELS-ADMM compared to that of original PETRELS with full observation and perfect reconstructions: $n = 50, r = 2, 90\%$ entries observed, outlier intensity $\text{fac-outlier} = 10$ and outlier density of 5%.

and intensity are varied in the range $[5\% - 40\%]$ and $[0.1, 1, 5, 10]$ respectively while SNR is set at 4 noise levels, $[5 - 40]$ dB. The results are shown as in Fig. 6, 8 and 7. In particular, when the density of outliers is low (e.g. 20%), both methods can detect outliers effectively at the high values of SNR. Their detection performance may be degraded when the effect of random noise is increased. Although the location of outliers can be identified correctly, PERTRELS-ADMM provides better results than GRASTA in term of sparsity, see Fig. 6(b) and (c). The effect of outlier intensity and density on their outlier detection performance are illustrated in Fig. 7 and Fig. 8 respectively. Similarly, our method also outperforms GRASTA. When the data is corrupted by strong outliers, both methods are able to detect them efficiently, but results of our method are more sparse than that of GRASTA, see Fig. 7(c) and (d). Moreover, in spire of the low SNR value, outliers are localized accurately by PETRELS-ADMM even in the presence of high corruptions, while GRASTA may yield many locations labeled as outliers, see Fig. 7(a) and Fig. 8(b) for examples. Besides, GRASTA fails to detect outliers in the case of a low outlier intensity (e.g. $\text{fac-outlier} = 0.1$), see Fig. 7(a).

3) Missing Scenarios: In order to illustrate the improvement of our iPETRELS for subspace update and tracking in the case of incomplete observations, a performance comparison of our method against the original PETRELS [10] and a well-known GROUSE algorithm [7] is conducted. For a fair comparison, the effect of outliers is ignored in this task. We consider the two system models, including large ($n = 500$, rank $r = 10$) and small ($n = 30$, rank $r = 2$). The subspace in the two scenarios will be corrupted at the time index 3000 over a total of 5000 observations. The noise level SNR is fixed at two values of 10 dB and 20 dB. The number of miss entries are very high, i.e., $[70 - 90]\%$ the total number of data are not observed.

The results are shown as in Fig. 9 and 10. We can see that, three algorithms can track the subspace for all tests, but iPETRELS outperforms the original PETRELS and GROUSE.



(a) $\frac{||\mathbf{s}_t-\tilde{\mathbf{s}}_t||_2}{||\mathbf{s}_t||_2} = 0.1265$ (b) $\frac{||\mathbf{x}_t-\tilde{\mathbf{x}}_t||_2}{||\mathbf{x}_t||_2} = 0.2748$

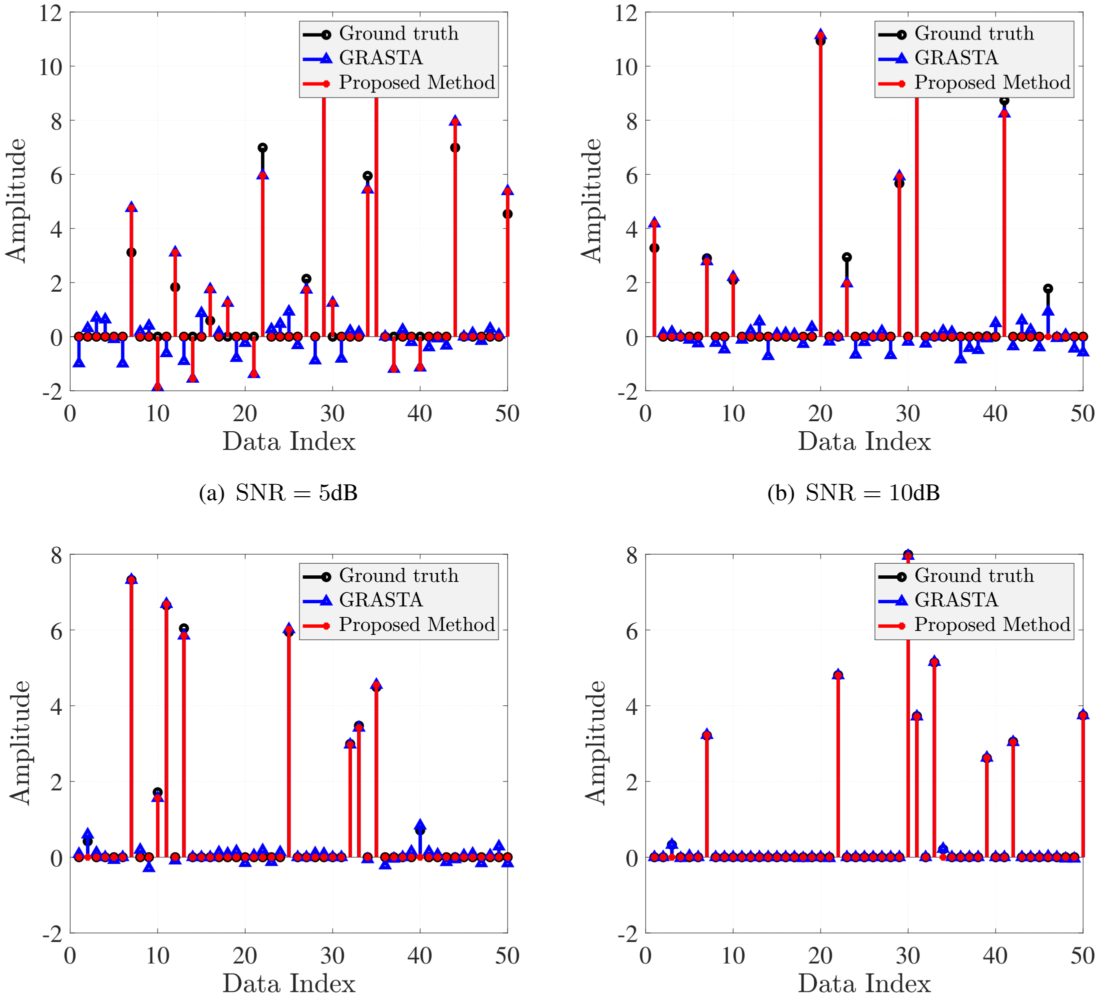
(c) $\frac{||\mathbf{s}_t-\tilde{\mathbf{s}}_t||_2}{||\mathbf{s}_t||_2} = 0.0635$ (d) $\frac{||\mathbf{x}_t-\tilde{\mathbf{x}}_t||_2}{||\mathbf{x}_t||_2} = 0.0636$

Fig. 5: Outlier detection and data reconstruction: $n = 50, r = 2$, 90% entries observed, outlier intensity $\text{fac-outlier} = 5$, and outlier density of 20%.

Particularly, for the large system, PETRELS-based algorithms converge faster than GROUSE even with a small number of entries observed each time. Moreover, the iPETRELS yields a much better subspace estimation performance than the original PETRELS in terms of SEP metric, see Fig. 9. For the small system, GROUSE provides a very good convergence rate compared to that of PETRELS, but no better than our method.

4) Robustness of PETRELS-ADMM: To investigate the robustness of PETRELS-ADMM, we vary the outlier intensity, corruption fraction (i.e., outlier and missing density) and then measure the SEP metric.

Impact of outlier intensity on algorithm performance: We fix $n = 50$, $r = 2$, 90% entries observed, outlier density of 5%, SNR = 20 dB while varying fac-outlier in the range of $[0.1, 10]$. We can see from the Fig. 11 that PETRELS-ADMM always outperforms other state-of-the-art algorithms in all testing cases with different fac-outlier values. At low outlier intensity (i.e.,



(a) SNR = 5dB

(b) SNR = 10dB

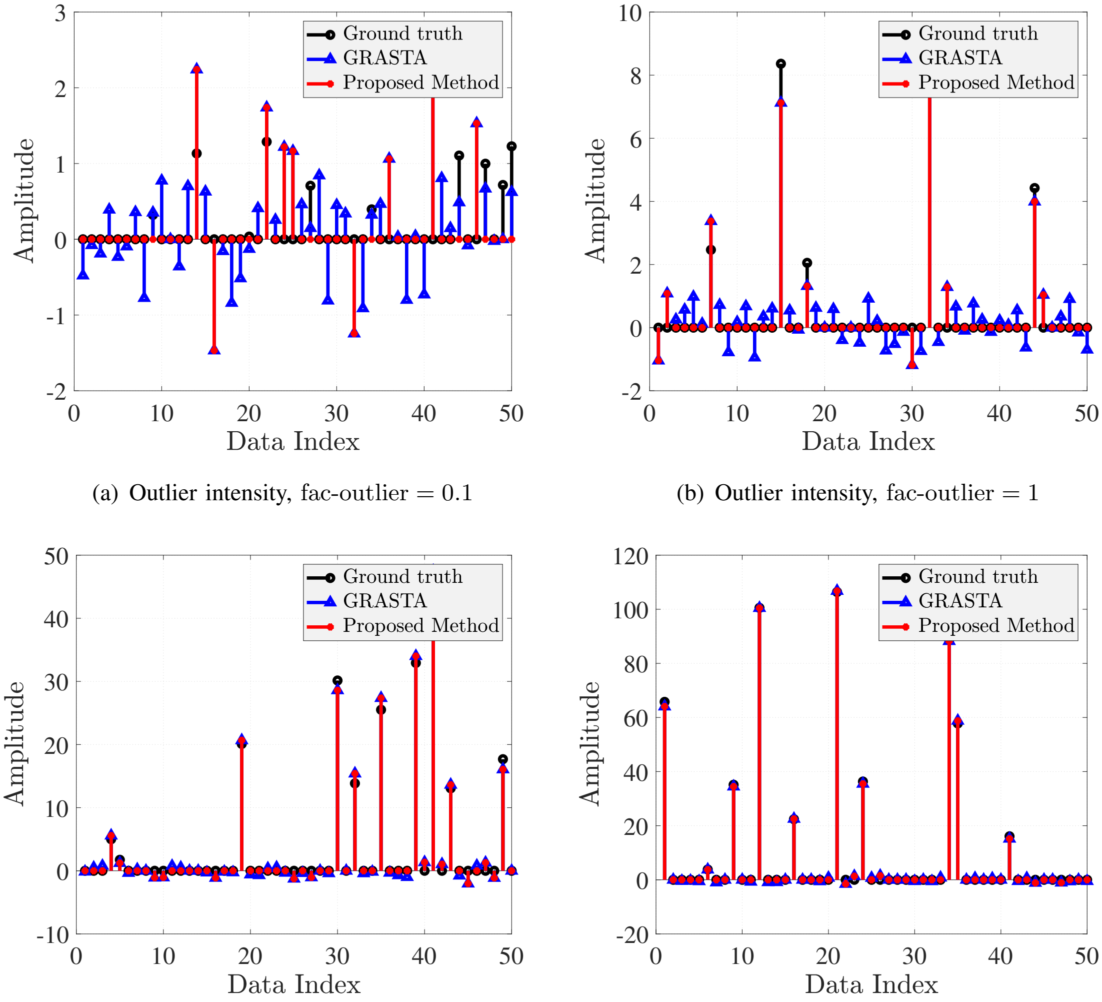
(c) SNR = 20dB

(d) SNR = 40dB

Fig. 6: Effect of noise on outlier detection performance: $n = 50, r = 2$, outlier density of 20% and outlier intensity $\text{fac-outlier} = 1$.

$\text{fac-outlier} \leq 1$), all algorithms yield good acuracy with fast convergences, though ROSETA obtains the higher SEP (i.e., $\approx 10^{-3}$) as compared to that of the four remaining algorithms. In particular, PETRELS-ADMM provides the best subspace estimation accuracy, i.e., $\text{SEP} \approx 10^{-5}$ in the both cases (see Fig. 11(a)-(b)). At a high intensity level (e.g. $\text{fac-outlier} = 5$ or 10), PETRELS-ADMM again provides the best performance in terms of both convergence rate and accuracy. GRASTA performs similarly to ReProCS and slightly worse than PETRELS-CFAR (i.e., their SEP values are around 10^{-4}). While ROSETA fails to recover the underlying subspace in the presence of strong outliers. Remark that, in all four experiments above, PETRELS-ADMM always obtains the best SEP value of around 10^{-5} and hence is robust to outlier intensity.

Impact of outlier density on algorithm performance: We fix $n = 50$, $r = 2$, 90% entries observed, outlier intensity $\text{fac-outlier} = 5$, $\text{SNR} = 20$ dB while varying the outlier density from



(a) Outlier intensity, fac-outlier = 0.1

(b) Outlier intensity, fac-outlier = 1

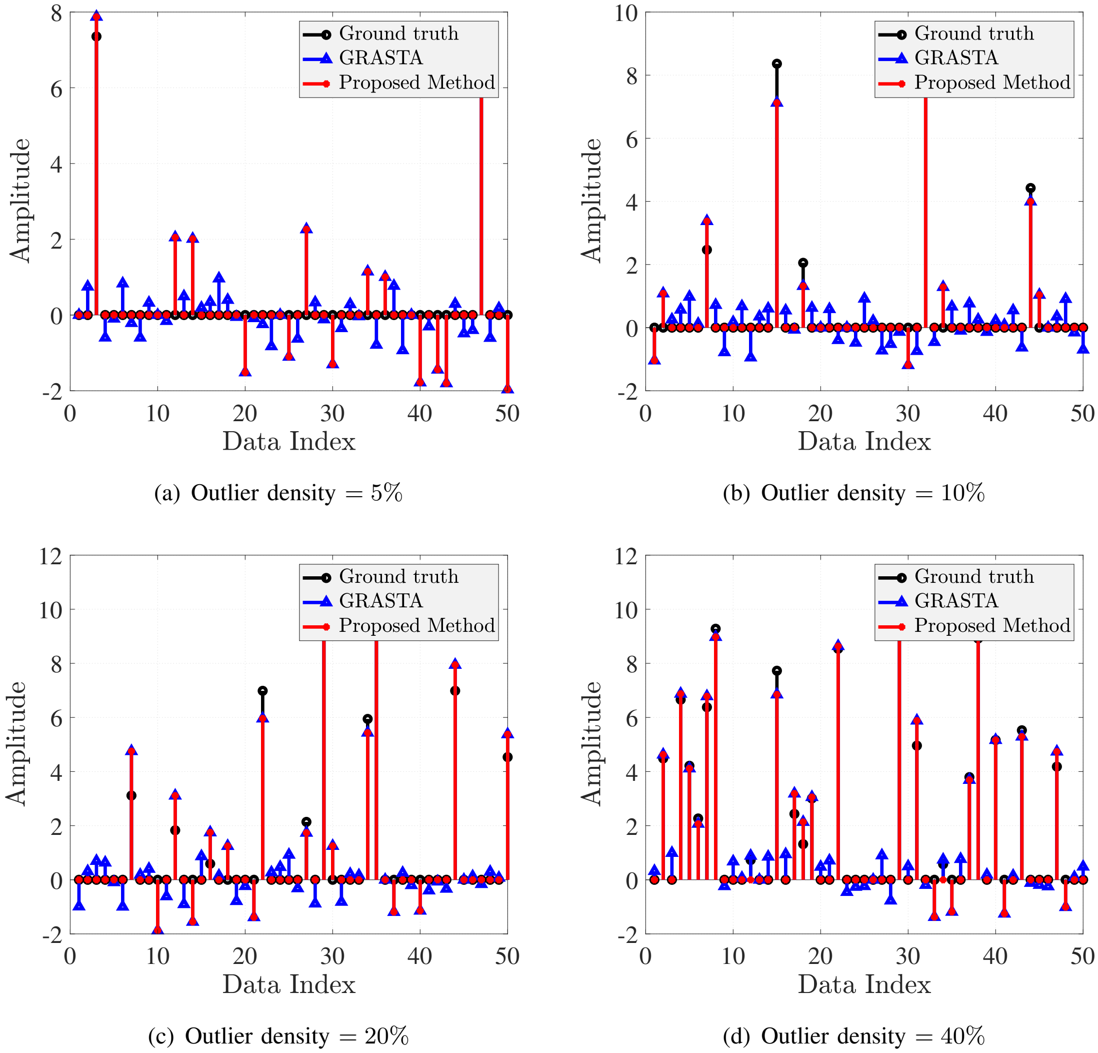
(c) Outlier intensity, fac-outlier = 5

(d) Outlier intensity, fac-outlier = 10

Fig. 7: Effect of outlier intensity on outlier detection performance: $n = 50, r = 2, \mathrm{SNR} = 5$ dB and outlier density of 20%.

5% to 40%. The results are shown as in Fig. 12. Similar to the first investigation, PETRELS-ADMM outperforms the four remaining algorithms in this task. In particular, our algorithm performs very well even when the fraction of outliers is high (e.g. 40%). By contrast, three algorithms including GRASTA, ROSETA and ReProCS may fail to track subspace in the case of a high outlier density (see Fig. 12(d)). The PETRELS-CFAR works well but has a lower convergence rate and accuracy in term of SEP metric than that of PETRELS-ADMM in this case. When the measurement data is corrupted by a smaller number of outliers, PETRELS-ADMM still provides better performance than the others, as shown in Fig. 12 (a)-(c).

Impact of corruption fraction on algorithm performance: Follow experiments above, we change the fraction of corruptions in the measurement data while fixing the other attributes. The results are reported in Fig. 13 and Fig. 14. In particular, the effect of missing density



(a) Outlier density = 5%

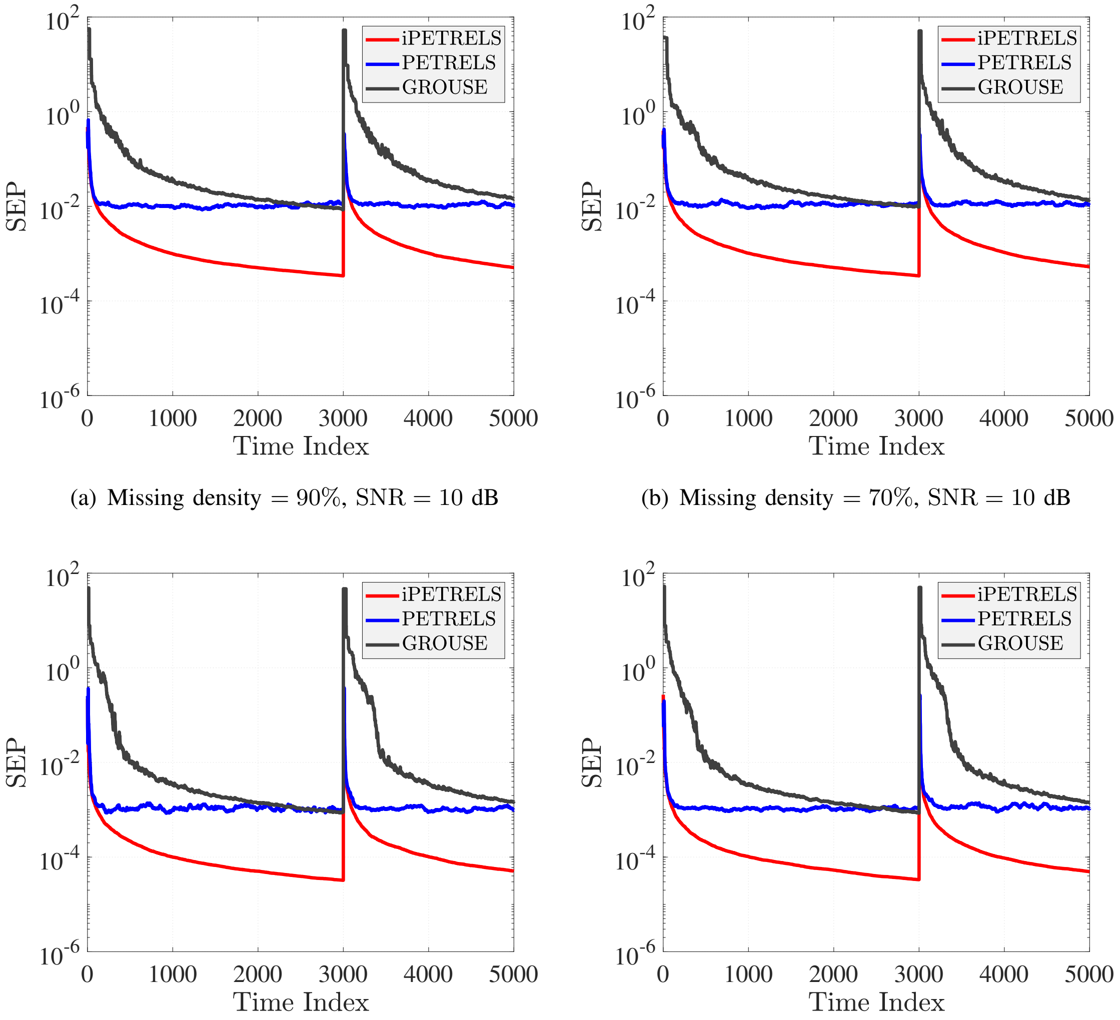
(b) Outlier density = 10%

(c) Outlier density = 20%

(d) Outlier density = 40%

Fig. 8: Effect of outlier density on outlier detection performance: $n = 50, r = 2, \mathrm{SNR} = 5$ dB and $\text{fac-outlier} = 1$.

on algorithm performance is presented in the Fig. 13. Similarly, PETRELS-ADMM yields the best performance in four cases of missing observations. Three algorithms including PETRELS-CFAR, GRASTA and ReProCS provides good performance but with slower convergence rate and accuracy, while ROSETA has failed again in this task due to the high outlier intensity (i.e., $\text{fac-outlier} = 5$). We continue to investigate deeper the impact of high corruption fractions on algorithm performance. As can be seen from Fig. 14(a)-(c) that the state-of-the-art algorithms only perform well when the number of corruptions is smaller than half the number of entries in the data measurement. While PETRELS-ADMM still obtains the reasonable subspace estimation performance in terms of SEP (i.e., $\approx 10^{-3}$) in the case of very high corruptions, see Fig. 14(d).



(a) Missing density = 90%, SNR = 10 dB

(b) Missing density = 70%, SNR = 10 dB

(c) Missing density = 90%, SNR = 20 dB

(d) Missing density = 70%, SNR = 20 dB

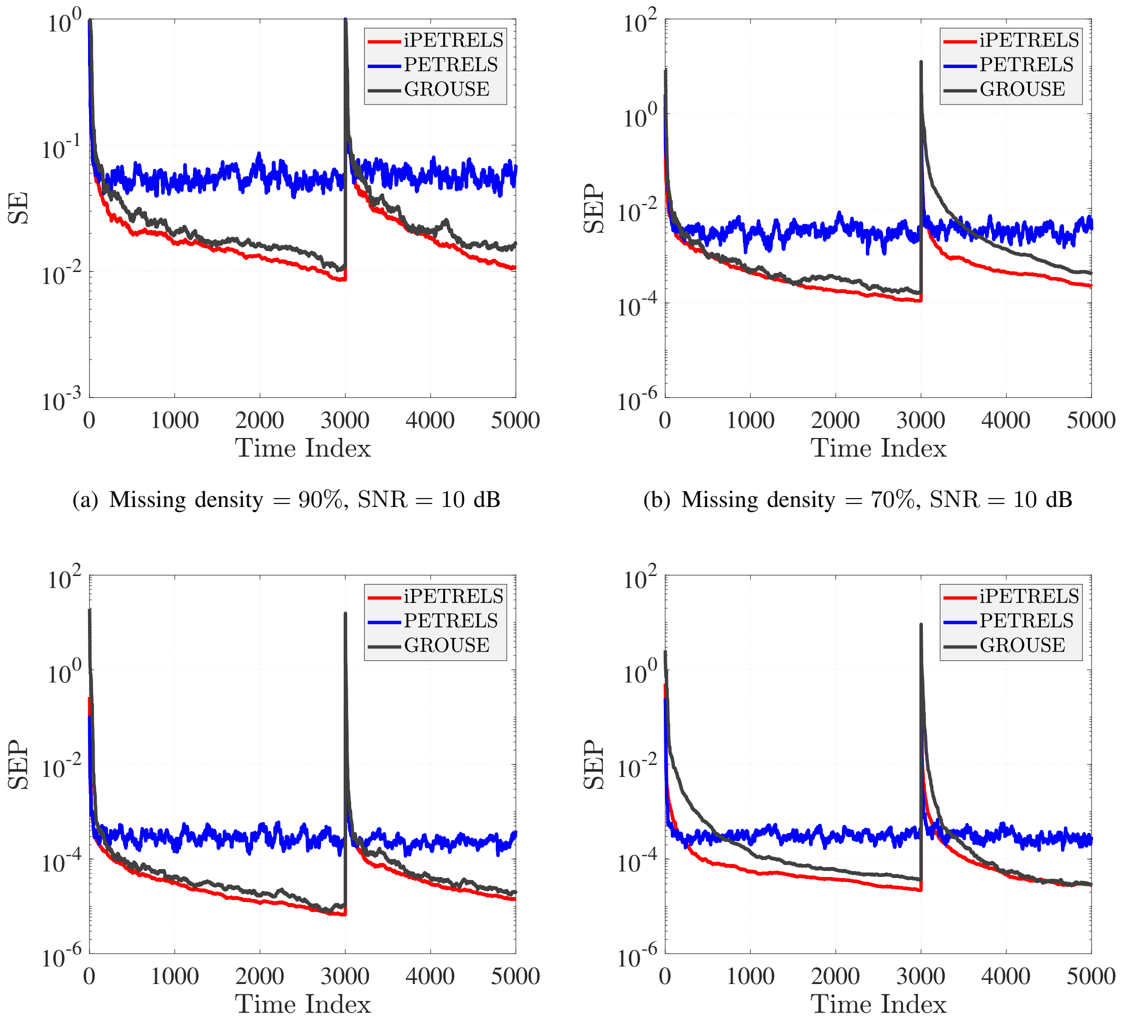
Fig. 9: Performance comparison between the subspace tracking algorithms for handling missing data: a large system with $n = 500, r = 10$.

B. Robust Matrix Completion

We compare the proposed algorithm of PETRELS-ADMM based robust matrix completion (RMC) with GRASTA [9], LRGeomGC [44] and RPCA-GD [45].

The measurement data $\mathbf{X} = \mathbf{AS}$ used for this task was the rank-2 matrices with size of 400×400. We generated the mixing matrix $\mathbf{A} \in \mathbb{R}^{400 \times 2}$ and the signal matrix $\mathbf{S} \in \mathbb{R}^{2 \times 400}$ at random. The entries were Gaussian i.i.d. of $\mathcal{N}(0, 1)$. The measurement data $\mathbf{X}$ was added with white Gaussian noise $\mathbf{N} \in \mathbb{R}^{400 \times 400}$ whose SNR is fixed at 40 dB. The matrix was corrupted by different percentages of missing and outliers from $0\% - 90\%$. The location and value of corrupted entries (including missing and outliers) were uniformly distributed.

Fig. 15 shows that the proposed algorithm of PETRELS-ADMM based RMC outperformed



(a) Missing density $= 90\%$, SNR $= 10$ dB

(b) Missing density $= 70\%$, SNR $= 10$ dB

(c) Missing density $= 90\%$, SNR $= 20$ dB

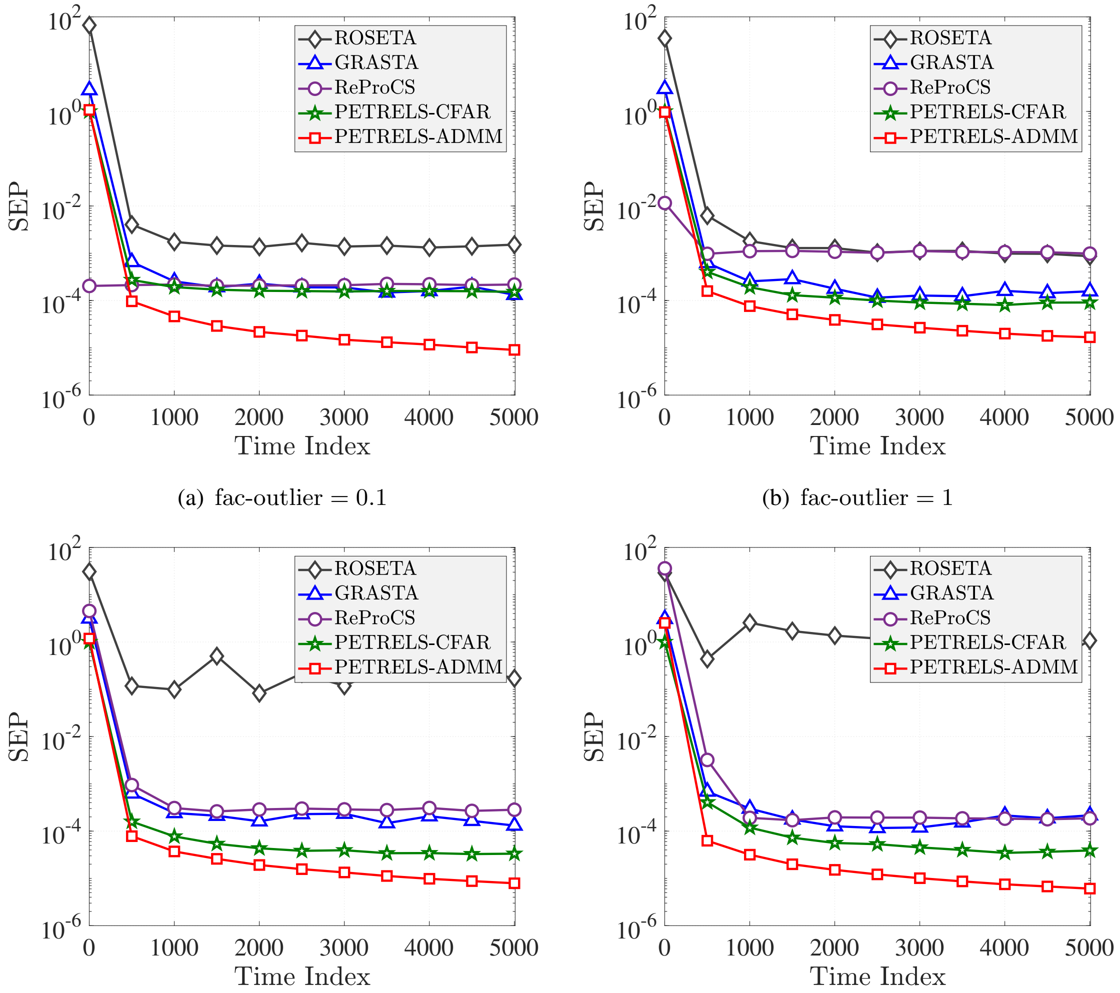
(d) Missing density $= 70\%$, SNR $= 20$ dB

Fig. 10: Performance comparison between the subspace tracking algorithms for handling missing data: a small system with $n = 30, r = 2$.

GRASTA and LRGeomGC and RPCA-GD. At low outlier intensity (i.e., $\text{fac-outlier} = 0.1$), PETRELS-ADMM based RMC, LRGeomGC and RCPA-GD provide excellent performance even when the data is corrupted by a very high corruption fraction. At high outlier intensity (i.e., $\text{fac-outlier} \geq 1$), PETRELS-ADMM based RMC provided the best matrix reconstruction error performance, GRASTA still retained good performance, while RPCA-GD and LRGeomGC failed to recover corrupted entries.

C. Video Background/Foreground Separation

We further illustrate the effectiveness of the proposed PETRELS-ADMM algorithm in the application of RST for video background/foreground separation, and compare with GRASTA



(a) fac-outlier = 0.1

(b) fac-outlier = 1

(c) fac-outlier = 5

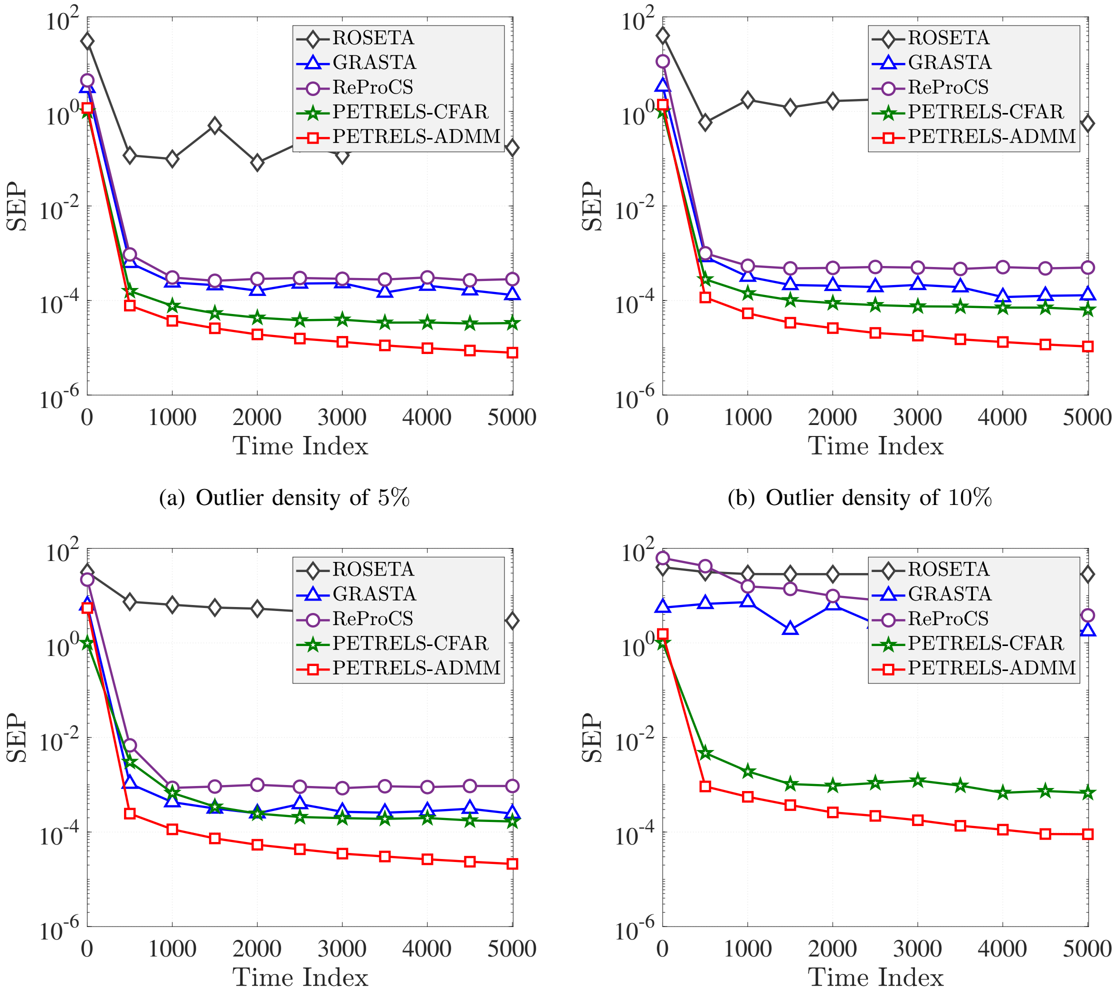
(d) fac-outlier = 10

Fig. 11: Impact of outlier intensity on algorithm performance: $n = 50$, $r = 2$, 90% entries observed, outlier density of 5% and SNR = 20 dB.

and PETRELS-CFAR. We use four real video sequences for this task, including `Hall`, `Lobby`, `Sidewalk` and `Highway` datasets. In particular, the two former datasets are from GRASTA's homepage[5], while the two latter dataset are from CD.net2012[6] [46]. The `Hall` dataset consists of 3584 frames of size 174×144 pixels, while the `Lobby` dataset has 1546 frames of size 144×176 pixels. The `Sidewalk` dataset includes 1200 frames of size 240×352 pixels. `Highway dataset` has 1700 frames of size 240×320 pixels. We can see from Fig. 16, PETRELS-ADMM is capable of detecting objects in video and provided competitive performance to GRASTA and PETRELS-CFAR.

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

[6] http://jacarini.dinf.usherbrooke.ca/dataset2012



(a) Outlier density of 5%

(b) Outlier density of 10%

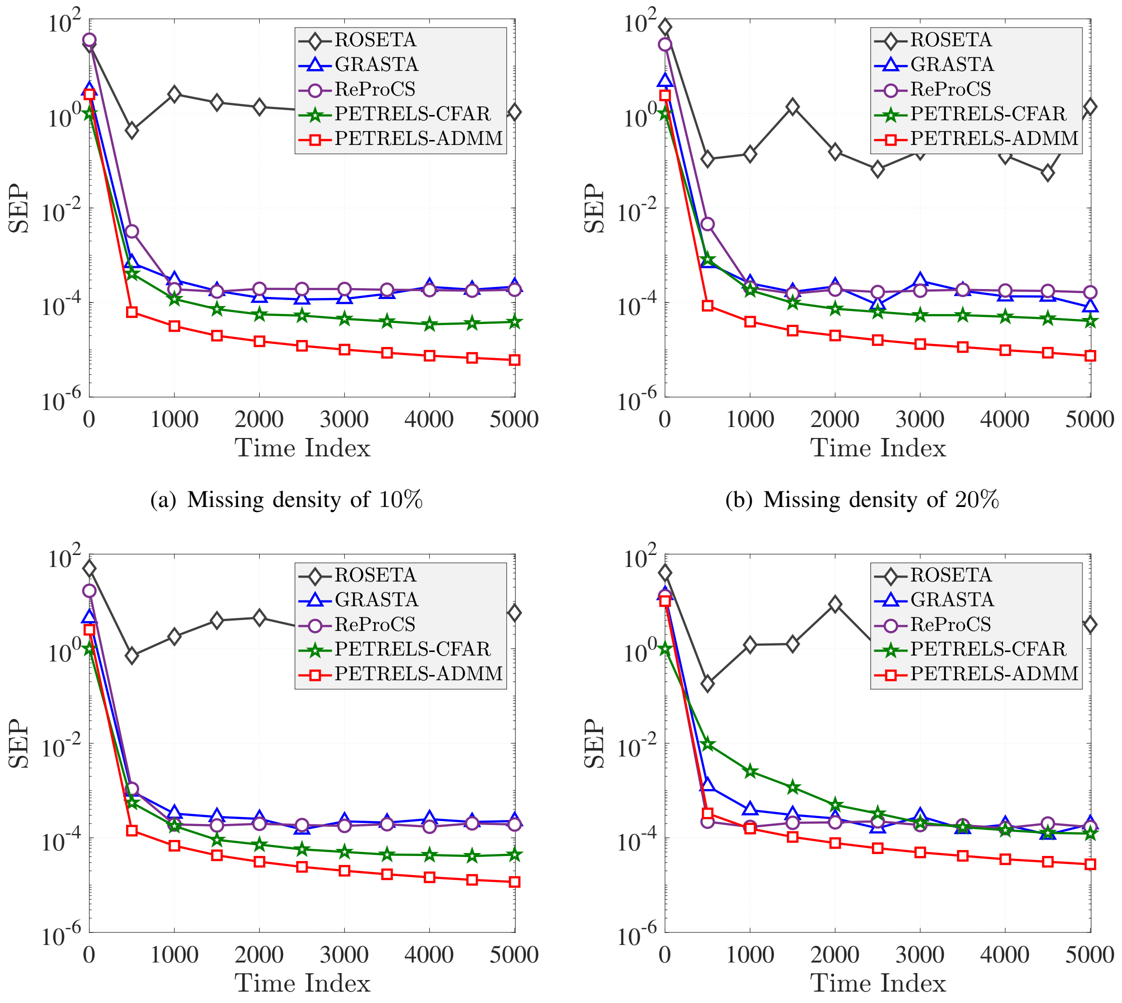
(c) Outlier density of 20%

(d) Outlier density of 40%

Fig. 12: Impact of outlier density on algorithm performance: $n = 50$, $r = 2$, 90% entries observed, outlier intensity $\text{fac-outlier} = 5$ and $\text{SNR} = 20$ dB.

VI. Conclusions

In this report, we proposed an efficient algorithm, namely PETRELS-ADMM, for the robust subspace tracking problem to handle missing data in the presence of outliers. By converting the original RST problem to a surrogate ones which facilitates the tracking ability, we derive an online implementation for outlier rejection with a low computational complexity and a fast convergence rate while still retaining a high subspace estimation performance. We established a theoretical convergence which guarantees that the solutions generated by PETRELS-ADMM will converge to a stationary point asymptotically. Experiments were conducted to evaluate the effectiveness of PETRELS-ADMM in terms of both quantity and quality. The results have suggested that our algorithm is more robust than state-of-the-art algorithm, e.g. GRASTA, ReProCS and PETRELS-CFAR in robust subspace tracking task; GRASTA, PRCA-GD and LRGeomGC in robust matrix



(a) Missing density of 10%

(b) Missing density of 20%

(c) Missing density of 30%

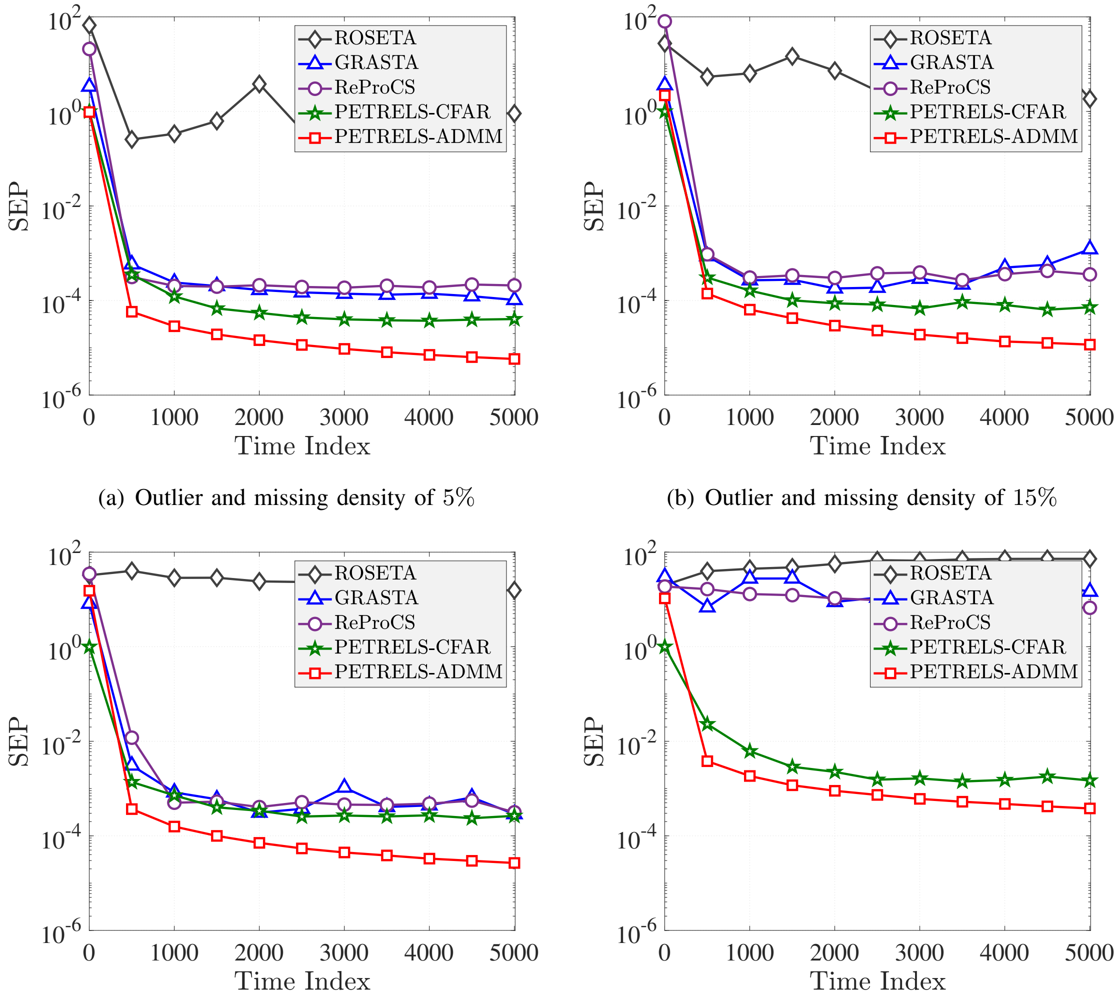
(d) Missing density of 40%

Fig. 13: Impact of missing density on algorithm performance: $n = 50, r = 2$, outlier density of 5%, outlier intensity $\text{fac-outlier} = 10$ and $\text{SNR} = 20$ dB.

completion task. The effectiveness of PETRELS-ADMM was also verified for background-foreground separation.

VII. Acknowledgment

This work was supported by the National Foundation for Science and Technology Development of Vietnam under Grant No. 102.04-2019.14.



(a) Outlier and missing density of 5%

(b) Outlier and missing density of 15%

(c) Outlier and missing density of 25%

(d) Outlier and missing density of 40%

Fig. 14: Impact of corruption fraction on algorithm performance: $n = 50, r = 2$ and $\text{fac-outlier} = 10$ and $\text{SNR} = 20$ dB.

VIII. Appendix

A. Technical Propositions

Before providing full proofs of the propositions, lemmas and theorems in the main report, we first give the following propositions which help us to derive several important results in the proofs.

Proposition 4. *([47, Lemma 13]): The function f is strongly convex if and only if for all $\mathbf{u}, \mathbf{v} \in \boldsymbol{dom}(f)$ we always have*

$$f(\mathbf{v}) - f(\mathbf{u}) - \frac{1}{2}\|\mathbf{v}-\mathbf{u}\|_2^2 \geq \langle \mathbf{v}-\mathbf{u}, \boldsymbol{\theta} \rangle, \quad \forall \boldsymbol{\theta} \in \partial f(\mathbf{u}).$$

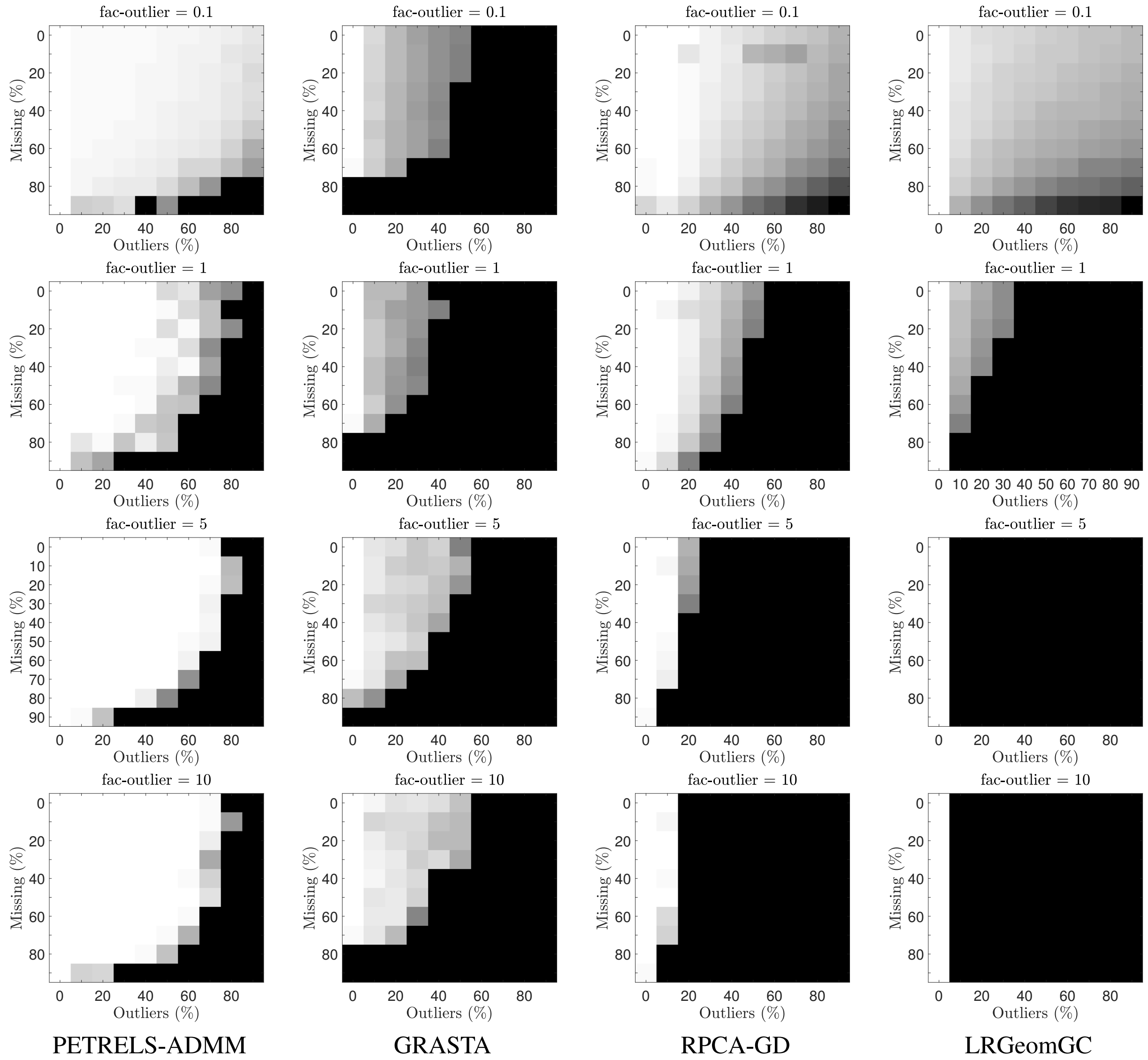


Fig. 15: Effect of outlier intensity on robust matrix completion performance. White colour denotes perfect recovery, black colour denotes failure and gray colour is in between.

Proposition 5. *([48]): The function f is m-strongly convex, with a constant m if and only if for all $\mathbf{u}, \mathbf{v} \in$ **dom**(f) we always have*

$$|f(\mathbf{v}) - f(\mathbf{u})| \geq \frac{m}{2} \|\mathbf{v} - \mathbf{u}\|_2^2 .$$

Proposition 6. *([48]): Every norm on $\mathbb{R}^n$ is convex and the sum of convex functions is convex.*

Proposition 7. *([49]): The Huber penalty function replaces the ℓ_1-norm $\|\mathbf{x}\|_1, \mathbf{x} \in \mathbb{R}^n$ is given by the sum $\sum_{i=1}^n f_\mu^{Hub}(x(i))$, where*

$$f_\mu^{Hub}(x(i)) = \begin{cases} \frac{x(i)^2}{2\mu}, & |x(i)| \leq \mu, \\ |x(i)| - \mu/2, & |x| > \mu. \end{cases}$$

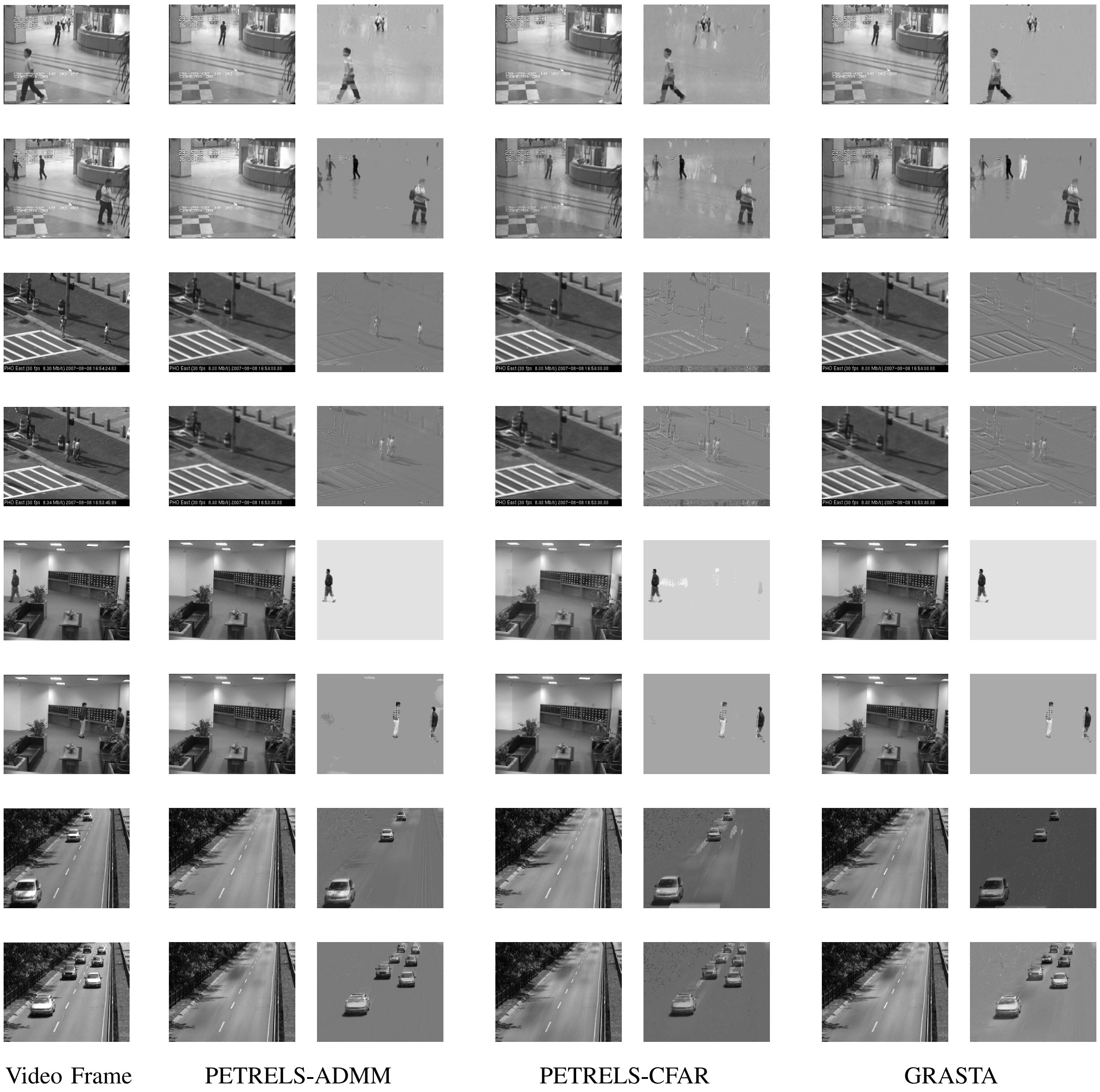


Fig. 16: Results of background-foreground separation

There exists a smooth version of the Huber function f_μ^{Hub}, which has derivatives of all degrees, i.e.,

$$\psi_\mu(\mathbf{x}) = \sum_{i=1}^{n} \big((x(i)^2 + \mu^2)^{1/2} - \mu\big).$$

and the first derivative of the pseudo-Huber function ψ_μ is defined by

$$\nabla\psi_\mu(\mathbf{x}) = \big[x(1)(x(1)^2 + \mu^2)^{-1/2}, \ldots, x(n)(x(n)^2 + \mu^2)^{-1/2}\big]^T.$$

Proposition 8. *([50, Proposition 1.2.4]): Let $\{a_t\}_{t=1}^{\infty}$ and $\{b_t\}_{t=1}^{\infty}$ be two nonnegative sequences such that $\sum_{i=1}^{\infty} a_i = \infty$ and $\sum_{i=1}^{\infty} a_i b_i < \infty$, $|b_{t+1} - b_t| < K a_t$ with some constant K, then $\lim\limits_{t\to\infty} b_t = 0$ or $\sum_{i=1}^{\infty} b_i < \infty$.*

Proposition 9. *If $\{f_t\}_{t\geq 1}$ and $\{g_t\}_{t\geq 1}$ are sequences of bounded functions which converge uniformly on a set $\mathcal{E}$, then $\{f_t + g_t\}_{t\geq 1}$ and $\{f_t g_t\}_{t\geq 1}$ converge uniformly on $\mathcal{E}$.*

B. Proof of Lemma 1

Follow the line as in previous convergence analysis of ADMM algorithms [39], [40], we can derive the proof of Lemma 1 as follows

1) Proof of Proposition (P-1): The minimizer $\mathbf{u}^{k+1}$ defined in (13) satisfies

$$\mathcal{L}(\mathbf{s}^k, \mathbf{u}^{k+1}, \mathbf{r}^k, \mathbf{w}^k, \mathbf{e}^k) \leq \mathcal{L}(\mathbf{s}^k, \mathbf{u}^k, \mathbf{r}^k, \mathbf{w}^k, \mathbf{e}^k) - c_u \|\mathbf{u}^k - \mathbf{u}^{k+1}\|_2^2. \tag{P-1}$$

At the k-th iteration, the $\mathbf{u}$-update in fact minimizes the objective function in Eq. (12), as

$$\mathbf{u}^{k+1} = \underset{\mathbf{u}}{\operatorname{argmin}}\, \mathcal{L}_{\mathbf{u},k}(\mathbf{u}, .) = \left(\frac{1+\rho_1}{2} \|\mathbf{u}\|_2^2 - [\mathbf{P}_t(\mathbf{x}_t - \mathbf{U}_{t-1}\mathbf{w}) - \rho_1(\mathbf{s}^k - \mathbf{r}^k)]^T \mathbf{u} \right).$$

The function $\mathcal{L}_{\mathbf{u},k}(\mathbf{u}, .)$ is strongly convex with a positive constant $(1+\rho_1)$, i.e., the Hessian of $\mathcal{L}_{\mathbf{u},k}(\mathbf{u}, .)$ is given by

$$\nabla^2 \mathcal{L}_{\mathbf{u},k}(\mathbf{u}, .) = (1+\rho_1)\mathbf{I}.$$

Since $\mathbf{u}^{k+1} = \operatorname{argmin}_{\mathbf{u}} \mathcal{L}_{\mathbf{u},k}(\mathbf{u}, .)$, we have the fact $\mathcal{L}_{\mathbf{u},k}(\mathbf{u}^{k+1}, .) \leq \mathcal{L}_{\mathbf{u},k}(\mathbf{u}^k, .)$,. Therefore, we obtain the following inequality

$$\mathcal{L}_{\mathbf{u},k}(\mathbf{u}^k, .) - \mathcal{L}_{\mathbf{u},k}(\mathbf{u}^{k+1}, .) \geq \frac{1+\rho_1}{2} \|\mathbf{u}^{k+1} - \mathbf{u}^k\|_2^2,$$

thanks to Proposition 5. It results in the Proposition (P-1).

2) Proof of Proposition (P-2): The minimizer s^{k+1} defined in (16) satisfies

$$\mathcal{L}(\mathbf{s}^{k+1}, \mathbf{u}^{k+1}, \mathbf{r}^k, \mathbf{w}^k, \mathbf{e}^k) \leq \mathcal{L}(\mathbf{s}^k, \mathbf{u}^{k+1}, \mathbf{r}^k, \mathbf{w}^k, \mathbf{e}^k) - c_s \|\mathbf{s}^k - \mathbf{s}^{k+1}\|_2^2. \tag{P-2}$$

At the k-th iteration, the variable s is updated by minimizing the objective function $\mathcal{L}_{\mathrm{s},k}(\mathrm{s}, .)$ in Eq. (13), as

$$\mathrm{s}^{k+1} = \underset{\mathbf{s}}{\operatorname{argmin}}\, \mathcal{L}_{\mathbf{s},k}(\mathrm{s}, .) = \rho \|\mathrm{s}\|_1 + \frac{\rho_1}{2} \|\mathrm{s} - (\mathbf{u}^{k+1} + \mathbf{r}^k)\|_2^2.$$

We exploit that if given $\mathbf{u}^{k+1}$ and $\mathbf{r}^k$, then both functions of the ℓ_1-norm $\|\mathrm{s}\|_1$ and ℓ_2-norm $\|\mathrm{s} - (\mathbf{u}^{k+1} + \mathbf{r}^k)\|_2^2$ are convex, so the $\mathcal{L}_{\mathbf{s},k}(\mathrm{s}, .)$ w.r.t. s is also convex. It is therefore that for any $\mathrm{s}^k, \mathrm{s}^{k+1} \in \mathcal{S}$, we always have

$$\mathcal{L}_{\mathbf{s},k}(\mathrm{s}^k, .) \geq \mathcal{L}_{\mathbf{s},k}(\mathrm{s}^{k+1}, .) + \langle \mathrm{s}^k - \mathrm{s}^{k+1}, \nabla \mathcal{L}_{\mathbf{s},k}(\mathrm{s}^{k+1}, .)\rangle + \frac{1}{2}\|\mathrm{s}^{k+1} - \mathrm{s}^k\|_2^2,$$

thanks to the Proposition 4.

Since $\mathbf{s}^{k+1} = \operatorname{argmin}_{\mathbf{s}} \mathcal{L}_{\mathbf{s},k}(\mathbf{s}, .)$, the first derivative $\nabla \mathcal{L}_{\mathbf{s},k}(\mathbf{s}^{k+1}, .) = \mathbf{0}$ and hence

$$\mathcal{L}_{\mathbf{s},k}(\mathbf{s}^k, .) \geq \mathcal{L}_{\mathbf{s},k}(\mathbf{s}^{k+1}, .).$$

In other word, there always exists a nonnegative number $c_s \geq 0$ such that

$$\mathcal{L}_{\mathbf{s},k}(\mathbf{s}^k, .) \geq \mathcal{L}_{\mathbf{s},k}(\mathbf{s}^{k+1}, .) + \frac{1}{2}\|\mathbf{s}^{k+1} - \mathbf{s}^k\|_2^2.$$

As a result, we have

$$\sum_{k=1}^{K} \frac{1}{2}\|\mathbf{s}^{k+1} - \mathbf{s}^k\|_2^2 \leq \sum_{i=1}^{K} \mathcal{L}_{\mathbf{s},k}(\mathbf{s}^k, .) - \mathcal{L}_{\mathbf{s},k}(\mathbf{s}^{k+1}, .) = \mathcal{L}_{\mathbf{s},k}(\mathbf{s}^1, .) - \mathcal{L}_{\mathbf{s},k}(\mathbf{s}^{K+1}, .)$$

Let $K \to \infty$, we then have

$$\sum_{k=1}^{\infty} \|\mathbf{s}^{k+1} - \mathbf{s}^k\|_2^2 < \infty.$$

It ends the proof of (P-2) and the second part of Lemma 1.

3) Proof of Proposition (P-3): The minimizer $\mathbf{r}^{k+1}$ defined in (14) satisfies

$$\mathcal{L}(\mathbf{s}^{k+1}, \mathbf{u}^{k+1}, \mathbf{r}^{k+1}, \mathbf{w}^k, \mathbf{e}^k) \leq \mathcal{L}(\mathbf{s}^{k+1}, \mathbf{u}^{k+1}, \mathbf{r}^k, \mathbf{w}^k, \mathbf{e}^k) - c_r\|\mathbf{r}^k - \mathbf{r}^{k+1}\|_2^2. \tag{P-3}$$

Follow the $\mathbf{r}$-update in Eq. (14), it is easy to verify that

$$\begin{aligned}\mathcal{L}(\mathbf{s}^{k+1}, \mathbf{u}^{k+1}, \mathbf{r}^{k+1}, \mathbf{w}^k, \mathbf{e}^k) &= \rho_1(\mathbf{r}^k + \mathbf{s}^{k+1} - \mathbf{u}^{k+1})^T(\mathbf{u}^{k+1} - \mathbf{s}^{k+1}) + A \\ &= \rho_1(\mathbf{r}^k)^T(\mathbf{u}^{k+1} - \mathbf{s}^{k+1}) - \rho_1\|\mathbf{u}^{k+1} - \mathbf{s}^{k+1}\|_2^2 + A \\ &= \mathcal{L}(\mathbf{s}^{k+1}, \mathbf{u}^{k+1}, \mathbf{r}^k, \mathbf{w}^k, \mathbf{e}^k) - \rho_1\|\mathbf{r}^{k+1} - \mathbf{r}^k\|_2^2,\end{aligned}$$

where $A = g(\mathbf{s}^{k+1}) + h(\mathbf{u}^{k+1}) + \frac{\rho_1}{2}\|\mathbf{u}^{k+1} - \mathbf{s}^{k+1}\|$. It implies the proposition (P-3).

4) Proof of Proposition (P-4): The minimizer $\mathbf{w}^{k+1}$ defined in (20) satisfies

$$\mathcal{L}(\mathbf{s}^{k+1}, \mathbf{u}^{k+1}, \mathbf{r}^{k+1}, \mathbf{w}^{k+1}, \mathbf{e}^k) \leq \mathcal{L}(\mathbf{s}^{k+1}, \mathbf{u}^{k+1}, \mathbf{r}^{k+1}, \mathbf{w}^k, \mathbf{e}^k) - c_w\|\mathbf{w}^k - \mathbf{w}^{k+1}\|_2^2. \tag{P-4}$$

Denote $\mathbf{z} = \mathbf{P}_t(\mathbf{U}_t\mathbf{w} + \mathbf{s}^{k+1} - \mathbf{x}_t)$. In fact, the $\mathbf{w}$-update minimizes the smooth version of the objective function (19), as follows

$$\mathcal{L}_{\mathbf{z},k}(\mathbf{z}, .) = \sum_{i=1}^{n} \Big(\big((\mathbf{z}(i)^2 + 1)^{1/2} - 1\big) + \frac{\rho_2}{2}\big((\mathbf{z}(i) - \mathbf{e}^k(i))^2 + 1)^{1/2} - 1\big) \Big)$$

The first two derivatives of $\mathcal{L}_{\mathbf{z},k}(\mathbf{z}, .)$ are given by

$$\begin{aligned}\nabla \mathcal{L}_{\mathbf{z},k}(\mathbf{z}, .) =& \big[\mathbf{z}(1)(\mathbf{z}(1)^2 + 1)^{-1/2}, \ldots, \mathbf{z}(n)(\mathbf{z}(n)^2 + 1)^{-1/2}\big]^T \\ & + \rho_2\big[(\mathbf{z}(1) - \mathbf{e}^k(1))((\mathbf{z}(1) - \mathbf{e}^k(1))^2 + 1)^{-1/2}, \ldots, (\mathbf{z}(n) - \mathbf{e}^k(n))((\mathbf{z}(1) - \mathbf{e}^k(1))^2 + 1)^{-1/2}\big]^T,\end{aligned}$$

and

$$\begin{aligned}\nabla^2 \mathcal{L}_{\mathbf{z},k}(\mathbf{z}, .) =& \operatorname{diag}\Big(\Big[(\mathbf{z}(1)^2 + 1)^{-3/2}, \ldots, (\mathbf{z}(n)^2 + 1)^{-3/2}\Big]\Big) \\ & + \rho_2 \operatorname{diag}\Big(\Big[((\mathbf{z}(1) - \mathbf{e}^k(1))^2 + 1)^{-3/2}, \ldots, (\mathbf{z}(n) - \mathbf{e}^k(n))^2 + 1)^{-3/2}\Big]\Big).\end{aligned}$$

The Hessian matrix $\nabla^2 \mathcal{L}_{\mathbf{z},k}(\mathbf{z},.)$ then satisfies

$$\rho_2 \mathbf{I} < \nabla^2 \mathcal{L}_{\mathbf{z},k}(\mathbf{z},.) \leq (\rho_2 + 1)\mathbf{I}.$$

It is therefore that $\mathcal{L}_{\mathbf{z},k}(\mathbf{w},.)$ is strongly convex and Lipschitz continuous. In other word, it implies that

$$\mathcal{L}(\mathbf{s}^{k+1}, \mathbf{u}^{k+1}, \mathbf{r}^{k+1}, \mathbf{w}^{k}, \mathbf{e}^{k}) - \mathcal{L}(\mathbf{s}^{k+1}, \mathbf{u}^{k+1}, \mathbf{r}^{k+1}, \mathbf{w}^{k+1}, \mathbf{e}^{k}) > \frac{\rho_2}{2} \|\mathbf{w}^{k} - \mathbf{w}^{k+1}\|_2^2.$$

which results in the Proposition (P-4), thanks to Proposition 5.

5) Proof of Proposition (P-5): The minimizer $\mathbf{e}^{k+1}$ defined in (22) satisfies

$$\mathcal{L}(\mathbf{s}^{k+1}, \mathbf{u}^{k+1}, \mathbf{r}^{k+1}, \mathbf{w}^{k+1}, \mathbf{e}^{k+1}) \leq \mathcal{L}(\mathbf{s}^{k+1}, \mathbf{u}^{k+1}, \mathbf{r}^{k+1}, \mathbf{w}^{k+1}, \mathbf{e}^{k}) - c_e \|\mathbf{e}^{k} - \mathbf{e}^{k+1}\|_2^2. \tag{P-5}$$

Similarly, we also have $\mathcal{L}_{\mathbf{e},k}(\mathbf{e},.)$ is strongly convex, i.e.,

$$\nabla^2 \mathcal{L}_{\mathbf{e},k}(\mathbf{e},.) = \rho_2 \operatorname{diag}\left(\left[((\mathbf{z}^k(1) - \mathbf{e}(1))^2 + 1)^{-3/2}, \ldots, (\mathbf{z}^k(n) - \mathbf{e}(n))^2 + 1)^{-3/2}\right]\right).$$

Therefore we have

$$\mathcal{L}_{\mathbf{e},k}(\mathbf{e}^k,.) - \mathcal{L}_{\mathbf{e},k}(\mathbf{e}^{k+1},.) \geq \frac{\rho_2}{2} \left\|\mathbf{e}^{k+1} - \mathbf{e}^{k}\right\|_2^2.$$

It ends the proof.

C. Proof of Proposition 3

To prove that $g_t(\mathbf{U})$ is strongly convex, we state the following facts: $g_t(\mathbf{U})$ is continuous and differentiable; its second derivative is a positive semi-definite matrix (i.e., $\nabla^2_{\mathbf{U}} g_t(\mathbf{U}) \geq m\mathbf{I}$); and the domain of $g_t(\mathbf{U})$ is convex. In order to satisfy the Lipschitz condition, we show that the first derivative of $g_t(\mathbf{U})$ is bounded.

Stage I: Prove that g_t is a strong convex function: We show that there exists a positive number m such that

$$|g_t(\mathbf{U}_{t+1}) - g_t(\mathbf{U}_t)| \geq m_1 \left\|\mathbf{U}_{t+1} - \mathbf{U}_t\right\|_F^2.$$

In particular, we state the two claims as follows:

(C-1) $g_t(\mathbf{U})$ is continuous and differentiable.

Proof. Given two variables $\mathbf{A}, \mathbf{B} \in \mathcal{U}$ such that $\|\mathbf{A} - \mathbf{B}\|_F^2 < \gamma$ for some positive constant γ. It is easy to verify that there exists a positive number θ such that $|g_t(\mathbf{A}) - g_t(\mathbf{B})| < \theta$. Thanks to the triangle inequality, we have the following inequality:

$$\begin{aligned} |g_t(\mathbf{A}) - g_t(\mathbf{B})| =& \frac{1}{t}\left| \sum_{i=1}^{t} \lambda^{t-i} \left\|\mathbf{P}_i(\mathbf{A}\mathbf{w}_i + \mathbf{s}_i - \mathbf{x}_i)\right\|_2^2 - \sum_{i=1}^{t} \lambda^{t-i} \left\|\mathbf{P}_i(\mathbf{B}\mathbf{w}_i + \mathbf{s}_i - \mathbf{x}_i)\right\|_2^2 \right| \\ &\leq \frac{1}{t} \sum_{i=1}^{t} \lambda^{t-i} \left\|\mathbf{P}_i(\mathbf{A} - \mathbf{B})\mathbf{w}_i\right\|_2^2 \leq \frac{1}{t} \sum_{i=1}^{t} \lambda^{t-i} \left\|\mathbf{P}_i(\mathbf{A} - \mathbf{B})\right\|_F^2 \left\|\mathbf{w}_i\right\|_2^2 \\ &\leq \frac{1}{t} \sum_{i=1}^{t} \lambda^{t-i} \left\|\mathbf{A} - \mathbf{B}\right\|_F^2 \left\|\mathbf{w}_i\right\|_2^2 = \frac{\gamma}{t} \sum_{i=1}^{t} \lambda^{t-i} \left\|\mathbf{w}_i\right\|_2^2 = \theta, \end{aligned}$$

It is therefore that the set of functions $\{g_t(\mathbf{U})\}_{t=1}^{\infty}$ is equicontinuous on $\mathcal{U}$. Furthermore, for any $\mathbf{U}^*, \mathbf{H} \in \mathcal{U}$, we show that the following limit exists:

$$\lim_{a \to 0} \frac{g_t(\mathbf{U}^* + a\mathbf{H}) - g_t(\mathbf{U}^*)}{a} = \lim_{a \to 0} \frac{1}{ta} \sum_{i=1}^{t} \lambda^{t-i} \bigg(\|\mathbf{P}_i((\mathbf{U}^* + a\mathbf{H})\mathbf{w}_i + \mathbf{s}_i - \mathbf{x}_i)\|_2^2 - \|\mathbf{P}_i(\mathbf{U}^*\mathbf{w}_i + \mathbf{s}_i - \mathbf{x}_i)\|_2^2 \bigg).$$

Specifically, let us denote $\mathbf{y}_i = \mathbf{P}_i(\mathbf{U}^*\mathbf{w}_i + \mathbf{s}_i - \mathbf{x}_i)$, the limit can be written as follows:

$$\begin{aligned}\lim_{a \to 0} \frac{g_t(\mathbf{U}^* + a\mathbf{H}) - g_t(\mathbf{U}^*)}{a} &= \lim_{a \to 0} \frac{1}{ta} \sum_{i=1}^{t} \lambda^{t-i} \big(\|\mathbf{y}_i - a\mathbf{P}_i\mathbf{H}\mathbf{w}_i\|_2^2 - \|\mathbf{y}_i\|_2^2 \big) \\ &= \lim_{a \to 0} \frac{1}{ta} \sum_{i=1}^{t} \lambda^{t-i} \big(\|a\mathbf{P}_i\mathbf{H}\mathbf{w}_i\|_2^2 - 2a\langle \mathbf{u}_i, \mathbf{P}_i\mathbf{H}\mathbf{w}_i \rangle \big) \\ &= \frac{-2}{t} \sum_{i=1}^{t} \lambda^{t-i} \langle \mathbf{y}_i, \mathbf{P}_i\mathbf{H}\mathbf{w}_i \rangle < \infty.\end{aligned}$$

As a result, the function $g_t(\mathbf{U})$ is differentiable and its first derivative $\nabla_{\mathbf{U}} g_t(\mathbf{U})$ can be given by

$$\nabla_{\mathbf{U}} g_t(\mathbf{U}) = \frac{2}{t} \sum_{i=1}^{t} \lambda^{t-i} \mathbf{P}_i(\mathbf{U}\mathbf{w}_i + \mathbf{s}_i - \mathbf{x}_i)\mathbf{w}_i^T.$$

In the similar way, it is easy to verify that $\nabla_{\mathbf{U}} g_t(\mathbf{U})$ is also continuous and the second derivative $\nabla_{\mathbf{U}}^2 g_t(\mathbf{U})$ is given by

$$\nabla_{\mathbf{U}}^2 g_t(\mathbf{U}) = \frac{2}{t} \sum_{i=1}^{t} \lambda^{t-i} \mathbf{P}_i \mathbf{w}_i \mathbf{w}_i^T.$$

□

(C-2) The second derivative $\nabla_{\mathbf{U}}^2 g_t(\mathbf{U})$ is a positive-define matrix. For all $\mathbf{x} \in \mathbb{R}^{p \times 1}$, we have

$$\mathbf{x}^T \nabla_{\mathbf{U}}^2 g_t(\mathbf{U}) \mathbf{x} = \frac{2}{t} \sum_{i=1}^{t} \lambda^{t-i} \mathbf{P}_i (\mathbf{w}_i^T \mathbf{x})^T (\mathbf{w}_i^T \mathbf{x}) = \frac{2}{t} \sum_{i=1}^{t} \lambda^{t-i} \mathbf{P}_i (\mathbf{w}_i^T \mathbf{x})^2 > 0, \quad \forall \lambda, t > 0.$$

It implies that there always exist a positive constant m such that $\nabla_{\mathbf{U}}^2 g_t(\mathbf{U}) \geq m\mathbf{I}$.

It follows to the claims (C-1), (C-2) and the assumptions showing that the domain of $g_t(\mathbf{U})$ is a convex set that $g_t(\mathbf{U}_t)$ is strongly convex [48, Section 3.1.4].

Stage II: Prove that g_t is a Lipschitz function:

$$|g_t(\mathbf{U}_{t+1}) - g_t(\mathbf{U}_t)| \leq m_2 \left\|\mathbf{U}_{t+1} - \mathbf{U}_t\right\|_F.$$

Let us denote $d_t(\mathbf{U}) = g_t(\mathbf{U}) - g_{t+1}(\mathbf{U})$. Since $\mathbf{U}_t = \underset{\mathbf{U}\in\mathcal{U}}{\operatorname{argmin}}\, g_t(\mathbf{U})$, we exploit that $g_{t+1}(\mathbf{U}_{t+1}) \leq g_{t+1}(\mathbf{U}_t)$ and hence

$$\begin{aligned} g_t(\mathbf{U}_{t+1}) - g_t(\mathbf{U}_t) &= g_t(\mathbf{U}_{t+1}) - g_{t+1}(\mathbf{U}_t) + g_{t+1}(\mathbf{U}_t) - g_t(\mathbf{U}_t) \\ &\leq \underbrace{\big(g_t(\mathbf{U}_{t+1}) - g_{t+1}(\mathbf{U}_{t+1})\big)}_{d_t(\mathbf{U}_{t+1})} - \underbrace{\big(g_t(\mathbf{U}_t) - g_{t+1}(\mathbf{U}_t)\big)}_{d_t(\mathbf{U}_t)}. \end{aligned}$$

The first derivative of $d_t(\mathbf{U}) = g_t(\mathbf{U}) - g_{t+1}(\mathbf{U})$ is given by

$$\begin{aligned} \nabla_{\mathbf{U}} d_t(\mathbf{U}) &= \nabla_{\mathbf{U}} g_t(\mathbf{U}) - \nabla_{\mathbf{U}} g_{t+1}(\mathbf{U}) \\ &= \frac{1}{t}\sum_{i=1}^{t} \lambda^{t-i}\mathbf{P}_i(\mathbf{U}\mathbf{w}_i + \mathbf{s}_i - \mathbf{x}_i)\mathbf{w}_i^T - \frac{1}{t+1}\sum_{i=1}^{t+1}\lambda^{t+1-i}\mathbf{P}_i(\mathbf{U}\mathbf{w}_i + \mathbf{s}_i - \mathbf{x}_i)\mathbf{w}_i^T. \end{aligned}$$

Let $\mathbf{A}_t = \sum_{i=1}^t \lambda^{t-i}\mathbf{P}_i\mathbf{U}\mathbf{w}_i\mathbf{w}_i^T$ and $\mathbf{B}_t = \sum_{i=1}^t \lambda^{t-i}\mathbf{P}_i(\mathbf{s}_i - \mathbf{x}_i)$, we can rewrite $\nabla_{\mathbf{U}} d_t(\mathbf{U})$ as

$$\nabla_{\mathbf{U}} d_t(\mathbf{U}) = \left(\frac{\mathbf{A}_t}{t} - \frac{\mathbf{A}_{t+1}}{t+1}\right) + \left(\frac{\mathbf{B}_t}{t} - \frac{\mathbf{B}_{t+1}}{t+1}\right).$$

Under the assumptions in Section II-C, the subspace $\mathbf{U}$, outlier $\{\mathbf{s}_t\}$, signal $\{\mathbf{x}_t\}$ and subspace coefficients $\{\mathbf{w}_t\}$ are bounded, then both $\mathbf{A}_t$ and $\mathbf{B}_t$ are bounded. It is therefore that

$$\left\|\nabla_{\mathbf{U}} d_t(\mathbf{U})\right\|_F \leq \left\|\frac{\mathbf{A}_t}{t} - \frac{\mathbf{A}_{t+1}}{t+1}\right\|_F + \left\|\frac{\mathbf{B}_t}{t} - \frac{\mathbf{B}_{t+1}}{t+1}\right\|_F \leq m_2 = \mathcal{O}(1/t).$$

Therefore $d_t(\mathbf{U})$ is Lipschiz with the constant m_2,

$$\frac{|d_t(\mathbf{U}_{t+1}) - d_t(\mathbf{U}_t)|}{\left\|\mathbf{U}_{t+1} - \mathbf{U}_t\right\|_F} \leq m_2, \text{ hence } \frac{|g_t(\mathbf{U}_{t+1}) - g_t(\mathbf{U}_t)|}{\left\|\mathbf{U}_{t+1} - \mathbf{U}_t\right\|_F} \leq m_2.$$

This ends the proof.

D. Proof of Lemma 2

We prove that our update rule is an approximate interpretation of Newton's method. Since the objective function g_t is strongly convex with respect to the variable $\mathbf{U}$, our algorithm can guarantee that the solution converges to the stationary point of the problem.

In order to estimate subspace, at each time instant t, we optimize the following minimization

$$\mathbf{u}^m = \underset{\mathbf{u}^m\in\mathbb{R}^{r\times 1}}{\operatorname{argmin}}\, \tilde{f}_t(\mathbf{u}^m), \text{ with } \tilde{f}_t = \sum_{i=1}^{t}\lambda^{t-i}\mathbf{P}_i(m,m)(\mathbf{x}_i^{\text{re}}(m) - \mathbf{w}_i^T\mathbf{u}^m)^2 + \frac{\alpha}{2t}\left\|\mathbf{u}^m\right\|_2^2.$$

The first derivative of the objective function $\tilde{f}_t(\mathbf{u}^m)$ can be determined by

$$\begin{aligned} \nabla\tilde{f}_t(\mathbf{u}_{t-1}^m) &= -2\sum_{i=1}^{t}\lambda^{t-i}\mathbf{P}_i(m,m)(\mathbf{x}_i^{\text{re}}(m) - \mathbf{w}_i^T\mathbf{u}_{t-1}^m)\mathbf{w}_i^T + \frac{\alpha}{t}\mathbf{u}_{t-1}^m \\ &= \nabla\tilde{f}_{t-1}(\mathbf{u}_{t-1}^m) - 2\mathbf{P}_t(m,m)(\mathbf{x}_t^{\text{re}}(m) - \mathbf{w}_t^T\mathbf{u}_{t-1}^m)\mathbf{w}_t^T + \frac{\alpha}{t}(\mathbf{u}_{t-1}^m - \mathbf{u}_{t-2}^m). \end{aligned}$$

Since $\mathbf{u}_{t-1}^m = \text{argmin}_{\mathbf{u}^m} \tilde{f}_{t-1}(\mathbf{u}^m)$, the derivative $\nabla \tilde{f}_{t-1}(\mathbf{u}_{t-1}^m) = \mathbf{0}$ and the Hessian at $\mathbf{u}_{t-1}^m$ is then given by

$$\mathbf{H}\tilde{f}_t(\mathbf{u}_{t-1}^m) = \nabla^2 \tilde{f}_t(\mathbf{u}_{t-1}^m) = 2\sum_{i=1}^{t} \lambda^{t-i}\mathbf{P}_i(m,m)\mathbf{w}_i\mathbf{w}_i^T + \frac{\alpha}{t}\mathbf{I}.$$

Thanks to Newton's method [48], a rule for subspace update can be obtained as

$$\mathbf{u}_t^m = \mathbf{u}_{t-1}^m - \eta_t \big(\mathbf{H}\tilde{f}_t(\mathbf{u}_{t-1}^m)\big)^{-1} \nabla \tilde{f}_t(\mathbf{u}_{t-1}^m).$$

Let us denote $\mathbf{R}_t^m = \sum_{i=1}^{t} \lambda^{t-i}\mathbf{P}_t(m,m)\mathbf{w}_i\mathbf{w}_i^T + \alpha\big(\frac{1}{2t} - \frac{\lambda_t}{2(t-1)}\big)\mathbf{I}$, we have

$$\mathbf{H}\tilde{f}_t(\mathbf{u}_{t-1}^m) = 2\mathbf{R}_t^m + \alpha\bigg(\frac{\lambda_t}{2(t-1)} - \frac{1}{2t}\bigg)\mathbf{I}$$

. As a result, we can derive the inverse Hessian matrix easily as follows

$$\big(\mathbf{H}\tilde{f}_t(\mathbf{u}_{t-1}^m)\big)^{-1} = \frac{1}{2}(\mathbf{R}_t^m)^{-1}\bigg(\frac{\mathcal{O}(1/t)}{2}(\mathbf{R}_t^m)^{-1} + \mathbf{I}\bigg)^{-1}.$$

When t is large enough, the term $\big(\frac{\mathcal{O}(1/t)}{2}(\mathbf{R}_t^m)^{-1} + \mathbf{I}\big)^{-1} \approx \mathbf{I} + \mathcal{O}\big(\frac{1}{t}\big)$. It is therefore that the step size can be approximated by

$$\big(\mathbf{H}\tilde{f}_t(\mathbf{u}_{t-1}^m)\big)^{-1}\nabla \tilde{f}_t(\mathbf{u}_{t-1}^m) = -\mathbf{P}_t(m,m)(\mathbf{x}_t^{\text{re}}(m) - \mathbf{w}_t^T\mathbf{u}_{t-1}^m)(\mathbf{R}_t^m)^{-1}\mathbf{w}_t + \mathcal{O}\big(\frac{1}{t}\big).$$

It implies that $\mathbf{u}_t^m$ can be updated by the following recursive update rule

$$\mathbf{u}_t^m = \mathbf{u}_{t-1}^m + \eta_t \mathbf{P}_t(m,m)(\mathbf{x}_t^{\text{re}}(m) - \mathbf{w}_t^T\mathbf{u}_{t-1}^m)(\mathbf{R}_t^m)^{\dagger}\mathbf{w}_t,$$

which is already defined in Eq. (25). In other word, the $\mathbf{u}_t^m$ generated by our algorithm can converge to the stationary point of $\tilde{f}_t(\mathbf{u}^m)$.

Note that, the properties of the objective functions and assumptions we made in Section II-C can guarantee the method will converge in practice. In particular, the objective functions $\tilde{g}_t(\mathbf{U})$ as well as $\tilde{f}_t(\mathbf{u})$ and their first derivatives are continuously differentiable which can avoid derivative issues in Newton's method. In addition, the starting points in our algorithm are always chosen at random. Further, since the objective functions $\{\tilde{g}_t(\mathbf{U})\}_{t=1}^{\infty}$ are always positive, PETRELS-ADMM can ignore the cases when their roots approach to zero asymptotically. To sum up, the solution $\mathbf{U}_t$ generated by PETRELS-ADMM will converge to the stationary point of the function $\tilde{g}_t(\mathbf{U})$.

The second part of the Lemma VIII-D can be easy to verify. Since $g_t(\mathbf{U}_t)$ is strongly convex and Lipschitz function as proved in Proposition 3, we have the following inequality

$$\begin{aligned}
&m_1 \|\mathbf{U}_{t+1} - \mathbf{U}_t\|_F^2 \leq |g_t(\mathbf{U}_{t+1}) - g_t(\mathbf{U}_t)| \leq m_2 \|\mathbf{U}_{t+1} - \mathbf{U}_t\|_F \\
&\Leftrightarrow \|\mathbf{U}_{t+1} - \mathbf{U}_t\|_F \big(\|\mathbf{U}_{t+1} - \mathbf{U}_t\|_F - \frac{m_2}{m_1}\big) \leq 0 \\
&\Leftrightarrow \|\mathbf{U}_t - \mathbf{U}_{t+1}\|_F \leq \frac{m_2}{m_1}.
\end{aligned}$$

Note that the positive number $m_2 = \mathcal{O}(1/t)$ is already given in the Appendix VIII-C, so it ends the proof .

E. Proof of the Lemma 3

Inspired of the result of convergence analysis for online sparse coding framework in [38, Proposition 2], we derive the convergence of $g_t(\mathbf{U}_t)$ in the similar way. In particular, we first denote the nonnegative stochastic process $\{u_t\}$ as follows

$$u_t \triangleq g_t(\mathbf{U}_t) \geq 0,$$

and then prove that it is a quasi-martingale, i.e., we have to prove the sum of the positive difference of $\{u_t\}_{t=1}^{\infty}$ is bounded,

$$\sum_{t=1}^{\infty} \big|\mathbb{E}[u_{t+1} - u_t]\big| < +\infty \quad a.s.$$

We can express $g_{t+1}(\mathbf{U}_t)$ with respect to $g_t(\mathbf{U}_t)$ as follows

$$\begin{aligned} g_{t+1}(\mathbf{U}_t) &= \frac{1}{t+1}\sum_{i=1}^{t+1} \lambda^{t+1-i} \left\|\mathbf{P}_i(\mathbf{U}_t\mathbf{w}_i + \mathbf{s}_i - \mathbf{x}_i)\right\|_2^2 + \rho\left\|\mathbf{s}_i\right\|_1 \\ &= \left(\frac{\lambda}{t+1}\sum_{i=1}^{t} \lambda^{t-i} \left\|\mathbf{P}_i(\mathbf{U}_t\mathbf{w}_i + \mathbf{s}_i - \mathbf{x}_i)\right\|_2^2 + \rho\left\|\mathbf{s}_i\right\|_1\right) \\ &\qquad\qquad + \left(\frac{1}{t+1}\big(\left\|\mathbf{P}_{t+1}\mathbf{U}_t + \mathbf{s}_{t+1} - \mathbf{x}_{t+1}\right\|_2^2 + \rho\left\|\mathbf{s}_{t+1}\right\|_1\big)\right) \\ &= \frac{\lambda t}{t+1} g_t(\mathbf{U}_t) + \frac{1}{t+1}\ell(\mathbf{U}_t, \mathbf{P}_{t+1}, \mathbf{x}_{t+1}). \end{aligned}$$

Since $\mathbf{U}_{t+1} = \text{argmin}_{\mathbf{U}}\, g_{t+1}(\mathbf{U})$, we have the fact $g_{t+1}(\mathbf{U}_{t+1}) - g_{t+1}(\mathbf{U}_t) \leq 0$, $f_t(\mathbf{U}_t) \leq g_t(\mathbf{U}_t)$, and hence

$$\begin{aligned} u_{t+1} - u_t = g_{t+1}(\mathbf{U}_{t+1}) - g_t(\mathbf{U}_t) &= \underbrace{g_{t+1}(\mathbf{U}_{t+1}) - g_{t+1}(\mathbf{U}_t)}_{\leq 0} + g_{t+1}(\mathbf{U}_t) - g_t(\mathbf{U}_t) \\ &\leq g_{t+1}(\mathbf{U}_t) - g_t(\mathbf{U}_t) = \frac{1}{t+1}\ell(\mathbf{U}_t, \mathbf{P}_{t+1}, \mathbf{x}_{t+1}) - \frac{t(1-\lambda)+1}{t+1} g_t(\mathbf{U}_t). \end{aligned}$$

It is therefore that

$$\begin{aligned} \mathbb{E}[u_{t+1} - u_t] &\leq \frac{\mathbb{E}[\ell(\mathbf{U}_t, \mathbf{P}_{t+1}, \mathbf{x}_{t+1}) - (t(1-\lambda)+1)g_t(\mathbf{U}_t)]}{t+1} \\ &\leq \frac{\mathbb{E}[\ell(\mathbf{U}_t, \mathbf{P}_{t+1}, \mathbf{x}_{t+1}) - g_t(\mathbf{U}_t)]}{t+1} \leq \frac{\mathbb{E}[\ell(\mathbf{U}_t, \mathbf{P}_{t+1}, \mathbf{x}_{t+1})] - f_t(\mathbf{U}_t)}{t+1} \\ &= \frac{\mathbb{E}[f(\mathbf{U}_t) - f_t(\mathbf{U}_t)]}{t+1} = \underbrace{\Big(\mathbb{E}[\sqrt{t}(f(\mathbf{U}_t) - f_t(\mathbf{U}_t))]\Big)}_{\mathbb{E}[G_t(\mathbf{U}_t)]}\underbrace{\left(\frac{1}{\sqrt{t}(t+1)}\right)}_{a_t}, \end{aligned}$$

because of $f_t(\mathbf{U}_t) \leq g_t(\mathbf{U}_t)$ and $\mathbb{E}[\ell(\mathbf{U}_t, \mathbf{P}_{t+1}, \mathbf{x}_t)] = f(\mathbf{U}_t)$. In parallel, we exploit that $G_t(\mathbf{U}_t) = \sqrt{t}(f(\mathbf{U}_t) - f_t(\mathbf{U}_t))$ is the scaled and centered version of the empirical measure, which converges in distribution to a normal random variable, thanks to the center limit theorem. Hence

$\mathbb{E}[\sqrt{t}(f(\mathbf{U}_t) - f_t(\mathbf{U}_t))]$ is bounded with a constant α. Then, the sum of the positive difference of $\mathbf{u}_t$ becomes

$$\sum_{t=1}^{\infty} |\mathbb{E}[u_{t+1} - u_t]| < \sum_{t=1}^{\infty} \frac{\alpha}{\sqrt{t}(t+1)}.$$

Furthermore, let us consider the convergence of the sum $\sum\limits_{t=1}^{\infty} \dfrac{\alpha}{\sqrt{t}(t+1)}$. We use the Cauchy-MacLaurin integral test [51] for convergence, as

$$\begin{aligned}\int_{t=1}^{+\infty} \frac{\alpha}{\sqrt{t}(t+1)} dt = \int_{x=1}^{\infty} \frac{\alpha}{(x^2+1)} dx &= \alpha.\text{arctan}(x)|_1^{+\infty} \\ &= \alpha.(\text{arctan}(\infty) - \text{arctan}(1)) = \alpha\frac{\pi}{4} < \infty.\end{aligned}$$

In other words, since the sum of $\mathbf{a}_t$ convergences, hence

$$\sum_{t=1}^{\infty} \mathbb{E}[u_{t+1} - u_t] < \infty.$$

We complete the proof.

F. Proof of Lemma 4

We investigate the convergence of a surrogate sequence $\big\{(g_t(\mathbf{U}_t) - f_t(\mathbf{U}_t))\frac{1}{t+1}\big\}$ as follows

$$\begin{aligned}\frac{g_t(\mathbf{U}_t) - f_t(\mathbf{U}_t)}{t+1} =& u_t - u_{t+1} + \underbrace{g_{t+1}(\mathbf{U}_{t+1}) - g_{t+1}(\mathbf{U}_t)}_{\leq 0} + \underbrace{\frac{t(\lambda-1)}{t+1} g_t(\mathbf{U}_t)}_{\leq 0} \\ &+ \frac{\ell(\mathbf{U}_t, \mathbf{P}_{t+1}, \mathbf{x}_{t+1}) - f_t(\mathbf{U}_t)}{t+1} \\ \leq& \underbrace{u_t - u_{t+1}}_{\text{(S-1)}} + \underbrace{\frac{\ell(\mathbf{U}_t, \mathbf{P}_{t+1}, \mathbf{x}_{t+1}) - f_t(\mathbf{U}_t)}{t+1}}_{\text{(S-2)}}\end{aligned}$$

because of $u_t = g_t(\mathbf{U}_t)$ and $\lambda \leq 1$. Note that, (S-1) – (S-2) converge almost surely:

- The sequence $\mathbb{E}[u_t - u_{t+1}]$ converges almost surely as proved in Lemma 3.
- The sequence (S-2) also converges, thanks to the fact $\mathbb{E}[\ell(\mathbf{U}_t, \mathbf{P}_{t+1}, \mathbf{x}_{t+1})] = f(\mathbf{U}_t)$ and the convergence of $\frac{\mathbb{E}[f(\mathbf{U}_t) - f_t(\mathbf{U}_t)]}{t+1}$ as mentioned in the appendix VIII-E.

It is therefore that the sequence $\big\{(g_t(\mathbf{U}_t) - f_t(\mathbf{U}_t))\frac{1}{t+1}\big\}$ converges almost surely, i.e.,

$$\sum_{t=0}^{\infty} \big(g_t(\mathbf{U}_t) - f_t(\mathbf{U}_t)\big)\frac{1}{t+1} < \infty.$$

On the other hand, the real sequence $\{\frac{1}{t+1}\}$ diverges, $\sum_{t=0}^{\infty} \frac{1}{t+1} = \infty$. It implies that $g_t(\mathbf{U}_t) - f_t(\mathbf{U}_t)$ convergences, thanks to the Proposition 8.

References

[1] A. Tulay and H. Simon, *Adaptive signal processing: next generation solutions*. John Wiley & Sons, Mar. 2010.

[2] P. Comon and G. H. Golub, "Tracking a few extreme singular values and vectors in signal processing," *Proceedings of the IEEE*, vol. 78, no. 8, pp. 1327–1343, 1990.

[3] I. T. Jolliffe and J. Cadima, "Principal component analysis: a review and recent developments," *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, vol. 374, no. 2065, p. 20150202, 2016.

[4] C. Wang, Y. C. Eldar, and Y. M. Lu, "Subspace estimation from incomplete observations: A high-dimensional analysis," *IEEE Journal of Selected Topics in Signal Processing*, vol. 12, no. 6, pp. 1240–1252, 2018.

[5] L. Balzano, Y. Chi, and Y. M. Lu, "Streaming pca and subspace tracking: The missing data case," *Proceedings of the IEEE*, vol. 106, no. 8, pp. 1293–1310, Aug 2018.

[6] N. Vaswani, T. Bouwmans, S. Javed, and P. Narayanamurthy, "Robust subspace learning: Robust pca, robust subspace tracking, and robust subspace recovery," *IEEE Signal Processing Magazine*, vol. 35, no. 4, pp. 32–55, July 2018.

[7] L. Balzano, R. Nowak, and B. Recht, "Online identification and tracking of subspaces from highly incomplete information," in *Communication, Control, and Computing (Allerton), 2010 48th Annual Allerton Conference on*. IEEE, 2010, pp. 704–711.

[8] D. Zhang and L. Balzano, "Global convergence of a grassmannian gradient descent algorithm for subspace estimation." in *International Conference on Artificial Intelligence and Statistics, AISTATS*, Cadiz, Spain, May 2016, pp. 1460–1468.

[9] J. He, L. Balzano, and A. Szlam, "Incremental gradient on the grassmannian for online foreground and background separation in subsampled video," in *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*. IEEE, 2012, pp. 1568–1575.

[10] Y. Chi, Y. C. Eldar, and R. Calderbank, "Petrels: Parallel subspace estimation and tracking by recursive least squares from partial observations," *IEEE Transactions on Signal Processing*, vol. 61, no. 23, pp. 5947–5959, 2013.

[11] B. Yang, "Projection approximation subspace tracking," *IEEE Transactions on Signal processing*, vol. 43, no. 1, pp. 95–107, 1995.

[12] M. Mardani, G. Mateos, and G. B. Giannakis, "Subspace learning and imputation for streaming big data matrices and tensors," *IEEE Transactions on Signal Processing*, vol. 63, no. 10, pp. 2663–2677, May 2015.

[13] H. Mansour and X. Jiang, "A robust online subspace estimation and tracking algorithm," in *Acoustics, Speech and Signal Processing (ICASSP), 2015 IEEE International Conference on*. IEEE, 2015, pp. 4065–4069.

[14] N. Linh-Trung, V. D. Nguyen, M. Thameri, T. Minh-Chinh, and K. Abed-Meraim, "Low-complexity adaptive algorithms for robust subspace tracking," *IEEE Journal of Selected Topics in Signal Processing*, vol. 12, no. 6, pp. 1197–1212, 2018.

[15] P. Narayanamurthy and N. Vaswani, "Provable dynamic robust pca or robust subspace tracking," *IEEE Transactions on Information Theory*, vol. 65, no. 3, pp. 1547–1577, March 2019.

[16] P. Narayanamurthy, V. Daneshpajooh, and N. Vaswani, "Subspace tracking from missing and outlier corrupted data," *arXiv preprint arXiv:1810.03051*, 2018.

[17] C. Hage and M. Kleinsteuber, "Robust pca and subspace tracking from incomplete observations using l_0-surrogates," *Computational Statistics*, vol. 29, no. 3-4, pp. 467–487, 2014.

[18] S. Chouvardas, Y. Kopsinis, and S. Theodoridis, "Robust subspace tracking with missing entries: The set-theoretic approach," *IEEE Transactions on Signal Processing*, vol. 63, no. 19, pp. 5060–5070, 2015.

[19] A. Gonen, D. Rosenbaum, Y. C. Eldar, and S. Shalev-Shwartz, "Subspace learning with partial information," *The Journal of Machine Learning Research*, vol. 17, no. 1, pp. 1821–1841, 2016.

[20] P. V. Giampouras, A. A. Rontogiannis, K. E. Themelis, and K. D. Koutroumbas, "Online sparse and low-rank subspace learning from incomplete data: A bayesian view," *Signal Processing*, vol. 137, pp. 199–212, 2017.

[21] J. Feng, H. Xu, and S. Yan, "Online robust pca via stochastic optimization," in *Advances in Neural Information Processing Systems*, 2013, pp. 404–412.

[22] B. Hong, L. Wei, Y. Hu, D. Cai, and X. He, "Online robust principal component analysis via truncated nuclear norm regularization," *Neurocomputing*, vol. 175, pp. 216–222, 2016.

[23] K. G. Quach, C. N. Duong, K. Luu, and T. D. Bui, "Non-convex online robust pca: Enhance sparsity via l_p-norm minimization," *Computer Vision and Image Understanding*, vol. 158, pp. 126–140, 2017.

[24] J. Shen, H. Xu, and P. Li, "Online optimization for max-norm regularization," *Machine Learning*, vol. 106, no. 3, pp. 419–457, 2017.

[25] P. P. Markopoulos, G. N. Karystinos, and D. A. Pados, "Optimal algorithms for l_1-subspace signal processing," *IEEE Transactions on Signal Processing*, vol. 62, no. 19, pp. 5046–5058, Oct 2014.

[26] P. P. Markopoulos, S. Kundu, S. Chamadia, and D. A. Pados, "Efficient l1-norm principal-component analysis via bit flipping," *IEEE Transactions on Signal Processing*, vol. 65, no. 16, pp. 4252–4264, Aug 2017.

[27] P. P. Markopoulos, M. Dhanaraj, and A. Savakis, "Adaptive l1-norm principal-component analysis with online outlier rejection," *IEEE Journal of Selected Topics in Signal Processing*, pp. 1–1, 2018.

[28] P. Narayanamurthy and N. Vaswani, "Nearly optimal robust subspace tracking," in *International Conference on Machine Learning*, 2018, pp. 3698–3706.

[29] X. Jia, X. Feng, W. Wang, H. Huang, and C. Xu, "Online schatten quasi-norm minimization for robust principal component analysis," *Information Sciences*, vol. 476, pp. 83–94, 2019.

[30] L. T. Thanh, V.-D. Nguyen, N. Linh-Trung, and K. Abed-Meraim, "Robust subspace tracking with missing data and outliers via admm," in *27th European Signal Processing Conference*, 2019.

[31] M. Shor and N. Levanon, "Performances of order statistics cfar," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 27, no. 2, pp. 214–224, 1991.

[32] J. A. Tropp, "Just relax: Convex programming methods for identifying sparse signals in noise," *IEEE transactions on information theory*, vol. 52, no. 3, pp. 1030–1051, 2006.

[33] N. Vaswani and P. Narayanamurthy, "Static and dynamic robust pca and matrix completion: A review," *Proceedings of the IEEE*, vol. 106, no. 8, pp. 1359–1379, Aug 2018.

[34] S. Ma and N. S. Aybat, "Efficient optimization algorithms for robust principal component analysis and its variants," *Proceedings of the IEEE*, vol. 106, no. 8, pp. 1411–1426, Aug 2018.

[35] S. Boyd, N. Parikh, E. Chu, B. Peleato, J. Eckstein *et al.*, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Foundations and Trends® in Machine learning*, vol. 3, no. 1, pp. 1–122, 2011.

[36] W. Deng and W. Yin, "On the global and linear convergence of the generalized alternating direction method of multipliers," *Journal of Scientific Computing*, vol. 66, no. 3, pp. 889–916, 2016.

[37] N. Parikh, S. Boyd *et al.*, "Proximal algorithms," *Foundations and Trends® in Optimization*, vol. 1, no. 3, pp. 127–239, 2014.

[38] J. Mairal, F. Bach, J. Ponce, and G. Sapiro, "Online learning for matrix factorization and sparse coding," *Journal of Machine Learning Research*, vol. 11, no. Jan, pp. 19–60, 2010.

[39] G. Li and T. K. Pong, "Global convergence of splitting methods for nonconvex composite optimization," *SIAM Journal on Optimization*, vol. 25, no. 4, pp. 2434–2460, 2015.

[40] Y. Wang, W. Yin, and J. Zeng, "Global convergence of admm in nonconvex nonsmooth optimization," *Journal of Scientific Computing*, vol. 78, no. 1, pp. 29–63, 2019.

[41] V. D. Nguyen, K. Abed-Meraim, N. Linh-Trung, and R. Weber, "Generalized minimum noise subspace for array processing," *IEEE Transactions on Signal Processing*, vol. 65, no. 14, pp. 3789–3802, July 2017.

[42] L. T. Thanh, V.-D. Nguyen, N. Linh-Trung, and K. Abed-Meraim, "Three-way tensor decompositions: A generalized minimum noise subspace based approach," *REV Journal on Electronics and Communications*, vol. 8, no. 1-2, pp. 28–45, Jun. 2018.

[43] G. H. Golub and C. F. Van Loan, *Matrix computations*. JHU press, 2012, vol. 3.

[44] B. Vandereycken, "Low-rank matrix completion by riemannian optimization," *SIAM Journal on Optimization*, vol. 23, no. 2, pp. 1214–1236, 2013.

[45] X. Yi, D. Park, Y. Chen, and C. Caramanis, "Fast algorithms for robust pca via gradient descent," in *Advances in neural information processing systems*, 2016, pp. 4152–4160.

[46] N. Goyette, P. Jodoin, F. Porikli, J. Konrad, and P. Ishwar, "Changedetection.net: A new change detection benchmark dataset," in *2012 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, June 2012, pp. 1–8.

[47] S. Shalev-Shwartz and Y. Singer, "Online learning: Theory, algorithms, and applications," 2007.

[48] S. Boyd and L. Vandenberghe, *Convex optimization*. Cambridge university press, 2004.

[49] K. Fountoulakis and J. Gondzio, "A second-order method for strongly convex ℓ_1-regularization problems," *Mathematical Programming*, vol. 156, no. 1-2, pp. 189–219, 2016.

[50] D. P. Bertsekas, "Nonlinear programming," *Journal of the Operational Research Society*, vol. 48, no. 3, pp. 334–334, 1997.

[51] K. Knopp, *Theory and application of infinite series*. Courier Corporation, 2013.