

LERE: Learning-Based Low-Rank Matrix Recovery with Rank Estimation

Zhengqin Xu¹, Yulun Zhang², Chao Ma¹, Yichao Yan¹, Zelin Peng¹,
Shoulie Xie³, Shiqian Wu⁴, Xiaokang Yang^{1,*}

¹AMoE Key Lab of Artificial Intelligence, AI Institute, Shanghai Jiao Tong University, China

²ETH Zürich, Switzerland

³Signal Processing, RF & Optical Dept. Institute for Infocomm Research A*STAR, Singapore

⁴School of Information Science and Engineering, Wuhan University of Science and Technology, China
{fate311, chaoma, yanyichao, zelin.peng, xkyang}@sjtu.edu.cn, yulun100@gmail.com,
slxie@i2r.a-star.edu.sg, shiqian.wu@wust.edu.cn

Abstract

A fundamental task in the realms of computer vision, Low-Rank Matrix Recovery (LRMR) focuses on the inherent low-rank structure precise recovery from incomplete data and/or corrupted measurements given that the rank is a known prior or accurately estimated. However, it remains challenging for existing rank estimation methods to accurately estimate the rank of an ill-conditioned matrix. Also, existing LRMR optimization methods are heavily dependent on the chosen parameters, and are therefore difficult to adapt to different situations. Addressing these issues, A novel **LE**arning-based low-rank matrix recovery with **R**ank **E**stimation (**LERE**) is proposed. More specifically, considering the characteristics of the Gerschgorin disk's center and radius, a new heuristic decision rule in the Gerschgorin Disk Theorem is significantly enhanced and the low-rank boundary can be exactly located, which leads to a marked improvement in the accuracy of rank estimation. According to the estimated rank, we select row and column sub-matrices from the observation matrix by uniformly random sampling. A 17-iteration feedforward-recurrent-mixed neural network is then adapted to learn the parameters in the sub-matrix recovery processing. Finally, by the correlation of the row sub-matrix and column sub-matrix, LERE successfully recovers the underlying low-rank matrix. Overall, LERE is more efficient and robust than existing LRMR methods. Experimental results demonstrate that LERE surpasses state-of-the-art (SOTA) methods. The code for this work is accessible at <https://github.com/zhengqinxu/LERE>.

Introduction

Low-rank matrix recovery (LRMR) is extensively employed across several applications, notably in collaborative filtering for recommendation system (Zadeh Kashani and Hamidzadeh 2020), background subtraction in video processing (Markowitz et al. 2022), robust principal component analysis (RPCA) for feature extraction (Wang et al. 2022), matrix sensing (Ma, Li, and Chi 2021) and matrix completion (Tong, Ma, and Chi 2021). Mathematically, for a large-scale observation matrix $\mathbf{Y} = f(\mathbf{L}_*)$, where operator $f(\cdot)$ denotes the sensing process, LRMR seeks to recover its underlying rank- r_* low-rank matrix $\mathbf{L}_* \in \mathbb{R}^{n_1 \times n_2}$. Commonly, LRMR

is formulated as:

$$\min_{\mathbf{L} \in \mathbb{R}^{n_1 \times n_2}} \frac{1}{2} \|\mathbf{Y} - f(\mathbf{L})\|_F^2 \quad \text{s.t.} \quad \text{rank}(\mathbf{L}) \leq r_*, \quad (1)$$

which is a constrained optimization problem. Apparently, the rank function constraint is NP-hard and computationally intractable. Several seminal works (Candès et al. 2011; Recht 2011) relaxed problem (1) as:

$$\min_{\mathbf{L} \in \mathbb{R}^{n_1 \times n_2}} \frac{1}{2} \|\mathbf{Y} - f(\mathbf{L})\|_F^2 + \lambda \|\mathbf{L}\|_*, \quad (2)$$

where $\|\mathbf{L}\|_*$ denotes the nuclear norm of \mathbf{L} . The nuclear norm is recognized to offer the most stringent convex envelope of the rank function, which can yield precise recovery under matrix-restricted isometry property (RIP) settings (Recht, Fazel, and Parrilo 2010). To address the challenge of the high computational complexity inherent in existing convex solutions to the problem (2), in spire of the ideas of Singular Value Decomposition (SVD) and Factorization, a range of efficient LRMR techniques have been developed.

For SVD-based methods, Phan et al. (Phan and Nguyen 2021) proposed a faster algorithm involving iterative reweighting of nuclear norm, which ensures that each limit point becomes a critical point to facilitate rapid outcomes. Gregor et al. (Gregor and LeCun 2010) achieved the most accurate possible approximation of the sparse code by training a non-linear feed-forward predictor with a specific architecture and a fixed depth. Their approach successfully implemented deep unfolding for sparse coding and achieved significant acceleration. Later works (Monga, Li, and Eldar 2021; Shen et al. 2021) successfully expanded their method to various tasks and network architectures. Inspired by these efforts, recent works (Cohen et al. 2019; Solomon et al. 2019) employed deep unfolding techniques to accelerate computation. Nevertheless, existing learning-based approaches for the nuclear-norm optimizations entail several singular value thresholding (SVT) (Van Luong et al. 2021), which necessitate a significant amount of computations for SVD (Hu et al. 2012). Consequently, these methods cannot be scaled to high-dimensional problems. When the prior rank is known beforehand, Cai et al. (Cai et al. 2021a) recently substituted SVD with CUR decomposition to streamline the computational process in LRMR.

*Corresponding Author: Xiaokang Yang, xkyang@sjtu.edu.cn
Copyright © 2024, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

Alternatively, in line with matrix factorization theory, when the prior rank r_p is known, \mathbf{L}_* is factorized as $\mathbf{L}_* = \mathbf{U}_* \mathbf{V}_*^T$, where $\mathbf{U}_* \in \mathbb{R}^{n_1 \times r_p}$ and $\mathbf{V}_* \in \mathbb{R}^{n_2 \times r_p}$, the optimization problem (1) is reformulated as:

$$\min_{\mathbf{U}=\mathbf{UV}^T} \frac{1}{2} \|\mathbf{Y} - f(\mathbf{UV}^T)\|_F^2. \quad (3)$$

As such, this optimization can be solved by performing gradient descent (GD) on \mathbf{U} and \mathbf{V} alternatively, thereby circumventing the expensive computation of SVD. Despite problem (3) being a non-convex optimization, recent advancements have shown that GD (Yi et al. 2016) and alternating minimization (Mishra, Apuroop, and Sepulchre 2012) algorithms can converge to the accurate low-rank factors, provided they are based on mild statistical assumptions and are appropriately initialized (Tong, Ma, and Chi 2021). To balance the two factors \mathbf{U} and \mathbf{V} , these matrix factorization approaches typically incorporate additional regularization terms. These terms take forms such as $\frac{1}{2} \|\mathbf{U}^T \mathbf{U} - \mathbf{V}^T \mathbf{V}\|_F^2$ (Park et al. 2017) or $\frac{1}{2} \|\mathbf{U}\|_F^2 + \frac{1}{2} \|\mathbf{V}\|_F^2$ (Chen et al. 2021, 2020).

Recently, Jia et al. (Jia et al. 2020) proposed the Generalized Unitarily Invariant Gauge (GUIG) regularization which achieves equilibrium between \mathbf{U} and \mathbf{V} , and showed that the nuclear norm (Cabral et al. 2013), Schatten- p quasi norm (Xu, Lin, and Zha 2017) and log sum of singular values are special cases of the GUIG regularization. Although the regularization terms simplify the theoretical analysis, (Tong, Ma, and Chi 2021) pointed out that they can be unnecessary as long as the initialization is selected properly. Meanwhile, the works in (Chen et al. 2020) and (Mishra, Apuroop, and Sepulchre 2012) argued that GD and its variations cost $O(\kappa \log(1/\epsilon))$ flops per iteration, making the convergence rate significantly reliant on the condition number κ and reach ϵ -accuracy, where κ of matrix \mathbf{L} of rank r is defined as $\kappa = \sigma_1(\mathbf{L})/\sigma_r(\mathbf{L})$, and $\sigma_i(\mathbf{L})$ is the i -th singular value. However, a diminishing convergence rate was observed when the underlying matrix becomes ill-conditioned. To address this issue, ScaledGD (Tong, Ma, and Chi 2021) incorporated a scaling factor into the gradient descent steps, effectively eliminating the impact of condition number κ on the GD convergence rate. With the aim of finding the optimal iteration step size of ScaledGD, Cai et al. (Cai, Liu, and Yin 2021) proposed LRPCA, a scalable and learned hybrid feedforward-recursive neural network model designed for potentially infinite iterative deployments of RPCA. LRPCA employs a straightforward formula and differentiable operators to enable breakthroughs in tackling higher-dimensional problems. However, there remain two major limitations of LRPCA: (1) A large number of iterations for LRPCA will make the optimization of network parameters excessively costly, and the convergence process that only relies on gradient descent is easily trapped in a local optimum; (2) LRPCA necessitates a known rank that can be estimated or obtained through prior knowledge. Similar to LRPCA, the factorization-based LRMR relies heavily on the precision of the estimated rank.

It should be noted that the rank is often unknown in practical applications. To address this issue, LMaFit (Wen, Yin, and Zhang 2012) employed two heuristic strategies to estimate the

rank. OptSpace (Keshavan, Montanari, and Oh 2010) determines the rank through the computation of SVD of trimmed observations. Inspired by the idea of fully Bayesian treatment, BCPF (Zhao, Zhang, and Cichocki 2015) designed a hierarchical probabilistic model and automatically determined the rank. Both Shi et al. (Shi, Lu, and Cheung 2017) and Li et al. (Li et al. 2023) implemented a rank-one matrix decomposition model for recovering the target low-rank matrix. They automatically estimated the rank by applying a l_1 norm regularization to the weight coefficients of the rank-one matrix during iteration. Yet, their rank estimation method based on iterative approximation incurs a high computational cost. Recently, Xu et al. (Xu et al. 2021a,b) took into consideration the distribution of the singular values, and the target rank was estimated by adapting the Gerschgorin disk method. Compared with aforementioned rank estimation methods based on iterative approximation, the approach grounded in the Gerschgorin Disk Theorem offers a more effective estimation of the target rank. These methods capitalize on the inherent low-rank attributes within the singular value space of the observation \mathbf{Y} , providing a more precise iteration direction and reducing computational expenses. While the using of the singular values' distribution is effective in estimating the rank, it is still limited by the adjustment factor. More importantly, when the matrix is an ill-conditional matrix, it is difficult to accurately divide the low-rank boundary by the existing adjustment factor method. Meanwhile, unlike matrix eigenvalues matrix eigenvalues which are calculated in descending order, the radii of Gerschgorin disks are not necessarily in descending order which further reduces the accuracy.

Contributions. A novel method, named learning-based Low-Rank Matrix Recovery via Rank Estimation (LERE), is proposed to accurately solve LRMR problems of the ill-conditional matrix in this work. The primary contributions of this work are outlined as:

- **LERE is a LRMR method that does not require rank priors.** By estimating the rank of the observed data's inherent low-rank structure, it provides the rank parameters for the subsequent non-convex optimization model based on low-rank matrix factorization, effectively eliminating the necessity of a known rank prior. In particular, **the rank estimation stage of LERE is more tolerant to the disturbance and the ill-conditional matrix.**
- **LERE is an accurate LRMR method.** In accordance with the estimated rank, it recovers the low-rank sub-matrix by randomly selecting column and row sub-matrices of the observation matrix \mathbf{Y} . Then, via the relationship of the column and row low-rank sub-matrices, it obtains the underlying low-rank matrix \mathbf{L}_* . The column/row random sampling can greatly reduce the size of the matrix to be processed in each iteration and effectively prevent the model from getting trapped in a local optimum throughout the entire iteration process.
- LERE only employs a 17-iteration feedforward-recurrent-hybrid neural network within the proposed optimization framework, which breaks the limitations imposed by specific train datasets and more iterations of traditional feedforward neural networks. LERE guarantees an enhanced

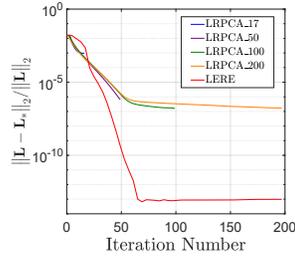
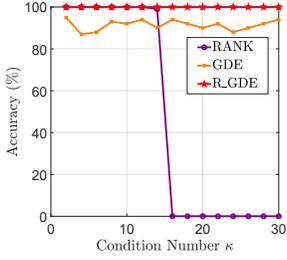


Figure 1: Rank estimation accuracy for GDE, RANK and R_GDE with varying κ .

Figure 2: Convergence comparison for LERE and varying LRPCA.

performance for low-rank matrix recovery through finite random sampling based on estimated rank.

- Across both synthetic and real-world datasets, comprehensive experimental findings demonstrate that the proposed LERE surpasses SOTA methods concerning performance.

Limitation and Analysis

We reveal and analyze the limitations of the ill-conditioned matrix rank estimation task and the current learning-based LRMR, respectively, in this section.

Rank Estimation The target rank is a critical parameter during LRMR, it is, however, unknown in most practical applications. While two different rank estimation methods were proposed in GDE (Xu et al. 2021a) and RANK (Xu et al. 2021b), they are not suitable for an ill-conditional matrix characterized by a large condition number κ , as illustrated in Fig. 1. Compared to RANK, GDE employed different degrees of contraction for Gershgorin disk radii based on the value of the center of each Gershgorin disk. When the condition number κ is small, this significantly increases the boundary distance between the sparse subspace and the low-rank subspace effectively improving the accuracy of the estimated rank. However, in the case of a larger condition number κ , the disparity in the center values among distinct Gershgorin disks within the low-rank subspace will notably increase, which causes the contraction scheme of GDE to push the subspace corresponding to the small center value Gershgorin disk away from the low-rank subspace. It’s worth emphasizing that the radii do not necessarily follow a descending order similar to that of the center values, which results in the failure of GDE when confronted with a large condition number κ , rank $r = 5$, and $\alpha = 0.4$, as shown in Fig. 1.

LRMR To address the optimization problem (3), ScaledGD introduces a scaling gradient descent method with a carefully selected step size. Subsequently, when the sparsity is known, LRPCA learns the step sizes corresponding to ScaledGD by employing the proposed Feedforward-Recurrent-Mixed Neural Network (FRMNN). Nevertheless, the above-mentioned methods still face challenges as they can only adapt to specific scenarios and are susceptible to getting trapped in local optima. While increasing the number of iterations of the FRMNN in LRPCA can lead to a more accurately recovered low-rank matrix L_* , it is important to note that once LRPCA’s convergence starts to occur gradually as the number

of iterations surpasses 50, as observed in Fig. 2, where rank $r = 5$, $\kappa = 10$, $\alpha = 0.4$, and LRPCA_100 means that LRPCA with 100 iterations. Furthermore, LRPCA with varying iteration numbers yields distinct convergence and differing points of extremity. Consequently, in real-world applications, it might be essential to learn multiple instances of LRPCA employing different iterations.

Proposed Method

Robust Rank Estimation

In this section, a robust rank estimation method named R_GDE is proposed to perform rank estimation of an ill-conditional low-rank matrix with a large condition number κ . For the purpose of estimating the rank of the underlying low-rank structure L implicit in an observation matrix Y , we first define the covariance matrix R_Y of the matrix $Y \in \mathbb{R}^{n_1 \times n_2}$ is $R_Y = YY^T$ (in this work, we assume that $n_1 < n_2$), where the target rank of L is r and $R_Y \in \mathbb{R}^{n_1 \times n_1}$. Subsequently, through unitary transformation matrix U_t and shrinkage matrix Σ_Y from the work of (Xu et al. 2021a), the resultant transformed matrix R_{Σ_Y} can be acquired as below:

$$R_{\Sigma_Y} = \Sigma_Y U_t^H R_Y U_t \Sigma_Y^{-1} = \begin{bmatrix} \sigma'_1 & \cdots & 0 & \frac{\sigma'_1}{\sigma'_{n_1}} \rho_1 \\ \vdots & \ddots & \vdots & \vdots \\ 0 & \cdots & \sigma'_{n_1-1} & \frac{\sigma'_{n_1-1}}{\sigma'_{n_1}} \rho_{n_1-1} \\ \frac{\sigma'_{n_1}}{\sigma'_1} \rho'_1 & \cdots & \frac{\sigma'_{n_1}}{\sigma'_{n_1-1}} \rho'_{n_1-1} & R_{n_1, n_1} \end{bmatrix}. \quad (4)$$

The diagonal elements of the transformed matrix R_{Σ_Y} satisfy the interlacing property (Wu, Yang, and Chen 1995): $\sigma'_1 \geq \sigma'_2 \geq \cdots \geq \sigma'_r \geq \cdots \geq \sigma'_{n_1-2} \geq \sigma'_{n_1-1}$. According to the Gerschgorin disk theorem (Wu, Yang, and Chen 1995), when the i -th Gerschgorin disk is in the non-low-rank space, its corresponding radius, denoted as $rad_i = |\frac{\sigma'_i}{\sigma'_{n_1}} \rho_i|$, becomes an extremely small, near-zero value. Conversely, if the disk is in the low-rank space, its radius will be far from zero. Therefore, in the works of (Xu et al. 2021a,b), the rank is identified through a heuristic decision rule given by:

$$RE(k) = rad_k - \frac{D_Y}{n_1 - 1} \sum_{i=1}^{n_1-1} rad_i, \quad (5)$$

where $k = 1, 2, 3, \dots, n_1 - 2$, and D_Y is a adjustment factor. When $RE(k)$ first becomes negative, $r = k - 1$ is the rank of the underlying low-rank structure. It is important to highlight that both the GDE, which involves manually setting the adjustment factor D_Y based on the size of matrix Y , and the RANK, which automatically sets the adjustment factor D_Y using the diagonal elements of matrix R_{Σ_Y} , face challenges in accurately solving the problem of rank estimation for a matrix with a large condition number and significant sparsity. This difficulty, illustrated in Fig 1, arises because the radii of the Gershgorin disk rad_i do not follow a descending order, unlike the center values σ'_i . For a significant Gershgorin disk center value, its corresponding radius might be smaller than the radii corresponding to other significant Gershgorin disk

center values. Consequently, even though the heuristic decision rule (5) halts at the first negative value, there could still be other positive values remaining.

In light of this, leveraging the characteristic of the descending order in Gerschgorin disk center values, we propose a new heuristic decision rule, named **R_GDE**, formulated as:

$$\mathbf{R_GDE}(k) = \sigma'_k - \frac{D_{kY}}{n_1 - 1} \sum_{i=1}^{n_1-1} \sigma'_i. \quad (6)$$

Taking into account the impact of both the large condition number and the significant sparsity on the Gerschgorin disk, we combine the radii and the center values of Gerschgorin disks into a novel unified adjustment factor:

$$D_{kY_c} = 1 - \frac{\sigma'_k}{\sqrt{\sum_{i=1}^{n_1-1} \sigma_i^2}} \quad (7)$$

$$D_{kY_r} = 1 - \frac{rad_k}{\sqrt{\sum_{i=1}^{n_1-1} rad_i^2}} \quad (8)$$

$$D_{kY} = (D_{kY_c} + D_{kY_r})/2, \quad (9)$$

where $k = 1, 2, \dots, n_1 - 1$. The descending property of the Gerschgorin disk center values ensures that the heuristic decision rule **R_GDE** does not miss any positive value. When **R_GDE**(k) first becomes negative value, rank $r = k - 1$. Due to different sensitivities of D_{kY_c} and D_{kY_r} to the presence of a large condition number and significant sparsity, with D_{kY_r} being less influenced by significant sparsity and D_{kY_c} remaining relatively unaffected by a large condition number, the unified adjustment factor D_{kY} exhibits enhanced robustness to the challenges posed by a large condition number and significant sparsity.

Consequently, the heuristic decision rule **R_GDE**, which introduces the unified adjustment factor D_{kY} , has a stronger ability to capture the boundary of the low-rank subspace, and can automatically obtain an accurate rank estimation. Compared to GDE and RANK, **R_GDE** achieves accurate rank estimation with varying sparsity α and varying condition numbers κ , as shown in Fig 1.

Learning-based Low-Rank Matrix Recovery with Rank Estimate (LERE)

This section presents a novel learning-based low-rank matrix recovery method via the above-described rank estimation and uniformly random sampling. It aims to robustly solve the optimization problem (3) while effectively eliminating the need for a rank prior.

The proposed method comprises three distinct phases: uniform random sampling, low-rank matrix recovery, and iterative update. The first phase involves uniformly and randomly sampling a column sub-matrix $\mathcal{C} = \mathbf{Y}(:, \mathcal{J})$ and a row sub-matrix $\mathcal{R} = \mathbf{Y}(\mathcal{I}, :)$ of the observation matrix \mathbf{Y} . In the second phase, by approximating $\mathbf{L}(:, \mathcal{J})$ and $\mathbf{L}(\mathcal{I}, :)$, $\hat{\mathcal{C}}$ and $\hat{\mathcal{R}}$ are obtained, respectively. Afterward, the underlying low-rank matrix \mathbf{L} is derived via the correlation matrix $\hat{\mathcal{U}}$ between $\hat{\mathcal{C}}$ and $\hat{\mathcal{R}}$. In the third phase, we repeatedly update the aforementioned procedure until the loss function $F(\mathbf{L}) := \mathbf{Y} - f(\mathbf{L})$ convergence.

Mathematically, the ℓ -th iteration of the above process is defined as:

$$\mathcal{C}_\ell = \min_{\mathbf{L}(:, \mathcal{J}) = \mathbf{U}_C \mathbf{V}_C^T} \frac{1}{2} \|\mathbf{Y}(:, \mathcal{J}) - f(\mathbf{U}_C \mathbf{V}_C^T)\|_F^2 \quad (10)$$

$$\mathcal{R}_\ell = \min_{\mathbf{L}(\mathcal{I}, :) = \mathbf{U}_R \mathbf{V}_R^T} \frac{1}{2} \|\mathbf{Y}(\mathcal{I}, :) - f(\mathbf{U}_R \mathbf{V}_R^T)\|_F^2 \quad (11)$$

$$\mathcal{U}_\ell = \mathcal{T}(\mathcal{C}_\ell(\mathcal{I}, :)) \quad \text{or} \quad \mathcal{T}(\mathcal{R}_\ell(:, \mathcal{J})) \quad (12)$$

where $\mathcal{C}_\ell \in \mathbb{R}^{n_1 \times \mathcal{J}}$, $\mathcal{R}_\ell \in \mathbb{R}^{\mathcal{I} \times n_2}$, and $\mathcal{U}_\ell \in \mathbb{R}^{\mathcal{J} \times \mathcal{I}}$. The low-rank matrix $\mathbf{L}_\ell = \mathcal{C}_\ell \mathcal{U}_\ell \mathcal{R}_\ell$. $\mathcal{T}(\cdot)$ denotes the truncated SVD based on the above estimated rank.

In this work, we take RPCA which is a canonical low-rank recovery task for illustration. The observation data matrix $\mathbf{Y} = \mathbf{L} + \mathbf{S}$, where \mathbf{S} is a sparse matrix that satisfies a certain sparsity \mathcal{S}_α defined in (Tong, Ma, and Chi 2021). We apply alternating project to solve the \mathbf{L} and \mathbf{S} sub-problems until convergence.

Update for \mathbf{L} The Eq. (10) and Eq. (11) can be rewritten:

$$\mathcal{C}_\ell = \min_{\mathbf{L}(:, \mathcal{J})} \frac{1}{2} \|\mathbf{Y}(:, \mathcal{J}) - \mathbf{U}_C \mathbf{V}_C^T - \mathbf{S}_{\ell-1}(:, \mathcal{J})\|_F^2 \quad (13)$$

$$\mathcal{R}_\ell = \min_{\mathbf{L}(\mathcal{I}, :)} \frac{1}{2} \|\mathbf{Y}(\mathcal{I}, :) - \mathbf{U}_R \mathbf{V}_R^T - \mathbf{S}_{\ell-1}(\mathcal{I}, :)\|_F^2, \quad (14)$$

where $\mathbf{L}(:, \mathcal{J}) = \mathbf{U}_C \mathbf{V}_C^T$ and $\mathbf{L}(\mathcal{I}, :) = \mathbf{U}_R \mathbf{V}_R^T$. In order to efficiently solve problems (13) and (14), a learning-based method is adapted. In this work, we employ the Feedforward-Recurrent-Mixed Neural Network (FRMNN) (Cai, Liu, and Yin 2021) to learn the step size parameters in the above problems. It should be emphasized that, unlike classical FRMNN which requires a greater number of iterations to achieve optimal performance, the proposed method only needs 17 iterations of FRMNN in each update of \mathbf{L} due to the convergence. As the update process continues, the low-rank matrix \mathbf{L} is gradually less influenced by the sparse matrix \mathbf{S} . Hence, the proposed method only requires training one set of parameters by the small sparsity dataset and obviates the need for specific training tailored to different sparsity.

Finally, we update \mathbf{L}_ℓ via $\mathbf{L}_\ell = \mathcal{C}_\ell \mathcal{U}_\ell \mathcal{R}_\ell$.

Update for \mathbf{S} The \mathbf{S} sub-problem is formulated as:

$$\mathbf{S}_\ell = \min_{\mathbf{S}} \frac{1}{2} \|\mathbf{Y} - \mathbf{L}_{\ell-1} - \mathbf{S}\|_F^2 = \mathcal{H}(\mathbf{Y} - \mathbf{L}_{\ell-1}). \quad (15)$$

The hard thresholding operator $\mathcal{H}(\cdot)$ denotes as follow:

$$[\mathcal{H}(\mathbf{S})]_{i,j} = \begin{cases} \mathbf{S}_{i,j}, & \text{if } |\mathbf{S}_{i,j}| > \zeta \\ 0 & \text{otherwise} \end{cases} \quad (16)$$

The iteration is terminated when

$$\frac{\|\mathbf{Err}(:, \mathcal{J})\|_F^2 + \|\mathbf{Err}(\mathcal{I}, :)\|_F^2}{\|\mathbf{Y}(:, \mathcal{J})\|_F^2 + \|\mathbf{Y}(\mathcal{I}, :)\|_F^2} < \delta \quad (17)$$

where $\mathbf{Err} = \mathbf{Y} - \mathbf{L}_\ell - \mathbf{S}_\ell$.

The proposed LERE for RPCA is summarized in Algorithm 1.

Convergence According to Theorem 3.10 of (Cai et al. 2021b), if low-rank matrix $\mathbf{L} \in \mathbb{R}^{n_1 \times n_2}$ and sparse matrix $\mathbf{S} \in \mathbb{R}^{n_1 \times n_2}$ satisfy the $\{\mu_1(\mathbf{L}), \mu_2(\mathbf{L})\}$ -incoherence and α -sparsity assumptions, respectively, the error of the Algorithm 1 output \mathbf{L}_* will satisfy

$$\frac{\|\mathbf{L} - \mathbf{L}_*\|_2}{\|\mathbf{L}\|_2} < \epsilon \kappa^{-1}, \quad (18)$$

Algorithm 1: LERE for RPCA

Input: $\mathbf{Y} = \mathbf{L} + \mathbf{S} \in \mathbb{R}^{n_1 \times n_2}$, rank r of \mathbf{Y} is estimated by Eq. (6), $\zeta_0 = \max(\mathbf{Y})$, \mathcal{I} , \mathcal{J} , $\delta = 1e^{-7}$, $\eta = 0.7$.

- 1: // Iterative updates
- 2: **while** Not (Stopping Condition) **do**
- 3: Update \mathbf{S}_ℓ according to Eq. (15)
- 4: Estimate the rank r of $\mathbf{Y} - \mathbf{S}_\ell$ by Eq. (6),
- 5: Resample \mathcal{I} , \mathcal{J}
- 6: Update \mathcal{C}_ℓ by Eq. (13), \mathcal{R}_ℓ by Eq.(14) and \mathcal{U}_ℓ by Eq. (12)
- 7: $\mathbf{L}_\ell = \mathcal{C}_\ell \mathcal{U}_\ell \mathcal{R}_\ell$
- 8: Update $\zeta_\ell = \eta \zeta_{\ell-1}$
- 9: **end while**

Output: \mathbf{L}_* and \mathbf{S}_*

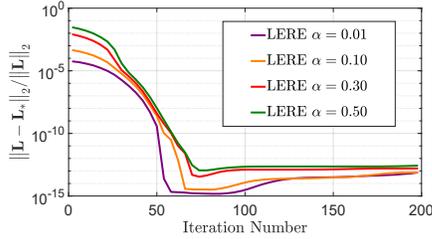


Figure 3: Convergence of LERE with varying sparsity α . Data dimension $n_1 = n_2 = 3000$, $\kappa = 10$ and rank $r = 5$.

when

$$|\mathcal{I}| \geq c_1 \max\{\mu_1(\mathbf{L})r \log(n_1), \log(n_1)/\alpha\}$$

$$|\mathcal{J}| \geq c_2 \max\{\mu_2(\mathbf{L})r \log(n_2), \log(n_2)/\alpha\}$$

are respectively uniformly selected. As shown in Fig. 3, LERE demonstrates the ability to achieve convergence across diverse sparsity scenarios. Furthermore, as illustrated in Fig. 2, the error incurred in LERE is notably smaller in comparison to that observed in LRPCA with 200 iterations.

Experimental Results

For the purpose of assessing the performance of the proposed LERE, we carry out comprehensive experiments using RPCA as an example of LRMR. This section presents the experimental outcomes of LERE, contrasting them with SOTA methods, as assessed on synthetic datasets and two real-world visual understanding scenarios. In our method, the sampling numbers are $\mathcal{I} = 4r \cdot \log(n_1)$ and $\mathcal{J} = 4r \cdot \log(n_2)$; the iteration number of FRMNN is 17. All parameters in compared methods follow their default settings. Moreover, all of our tests run on a Windows 10 laptop with Intel i7-9750H CPU, 64G RAM. The parameters learning processes run on an Ubuntu workstation with Intel i9-9900K CPU and two Nvidia RTX-2080Ti GPUs.

Comparison of Various Rank Estimation Methods

To confirm the efficiency and accuracy of R_GDE, diverse data matrices are utilized in testing to estimate the rank, and the method is compared with two SOTA rank estimate

n_2	3000						1000					
α	0.1			0.4			0.1			0.4		
κ	1	15	30	1	15	30	1	15	30	1	15	30
GDE	98	65	0	100	37	0	99	58	0	100	29	0
RANK	100	95	87	79	78	77	98	94	87	53	52	46
R_GDE	100	100	100	100	100	100	100	100	100	100	100	97

Table 1: The accuracy of estimated rank by using R_GDE (ours), GDE (Xu et al. 2021a), and RANK (Xu et al. 2021b) with varying size of matrix ($n_1 = 3000$), varying outlier sparsity α and varying condition number κ .

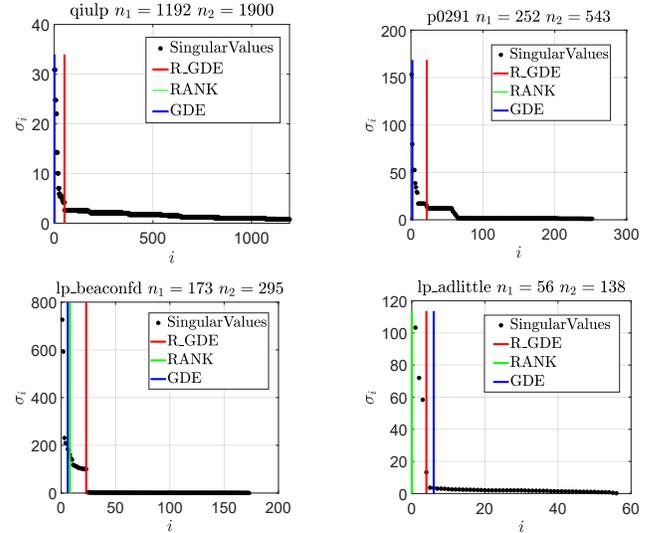


Figure 4: Rank estimation of sparse matrices.

approaches: GDE (Xu et al. 2021a) and RANK (Xu et al. 2021b). The tests of synthetic matrices with varying settings are based on 100 Monte Carlo runs, Table 1 illustrates the results of these tests.

It is evident that while GDE successfully accomplishes precise rank estimation for various sparsity α when the condition number κ is equal to 1, its performance sharply deteriorates as κ increases, ultimately leading to failed rank estimation. Even though RANK demonstrates robustness in handling varying condition number κ , it is more vulnerable to the impact of the sparsity α compared to GDE. Meanwhile, both GDE and RANK experience a decline in their performance when confronted with narrow matrices whose dimensions are $n_1 = 3000$ and $n_2 = 1000$. Compared with GDE and RANK, the proposed R_GDE method exhibits stronger adaptability to different sparsity α and different condition number κ , and achieves a more accurate rank estimation of ill-conditioned low-rank matrix. It's noteworthy to highlight that R_GDE maintains excellent performance even when applied to narrow matrices.

Next, SuiteSparse Database¹ which provides general data matrices with low numerical ranks, is tested, and these out-

¹<https://sparse.tamu.edu/>

n_2		3000						500					
κ		1			10			1			10		
α		0.1	0.3	0.5	0.1	0.3	0.5	0.1	0.3	0.5	0.1	0.3	0.5
ARE-RPCA	L_{rec}	6.5E-9	1.4E-8	1.3E-7	1.2E-8	6.1E-6	1.6E-5	4.2E-5	2.1E-1	3.1E-1	2.7E-6	2.1E-5	3.0E-1
	S_{rec}	7.0E-7	7.0E-7	9.2E-7	1.4E-6	4.7E-5	7.3E-5	1.7E-4	8.9E-1	9.9E-1	5.7E-5	8.9E-1	9.9E-1
ADW-RPCA	L_{rec}	1.3E-6	3.1E-6	5.8E-6	2.7E-6	6.2E-6	1.2E-5	9.2E-2	9.1E-1	9.1E-1	7.6E-1	7.7E-1	7.8E-1
	S_{rec}	1.4E-4	1.5E-4	1.6E-4	3.1E-4	3.1E-4	3.3E-4	6.6E+0	3.8E+0	3.9E+0	5.6E+0	3.3E+0	2.6E+0
AccAltProj	L_{rec}	7.9E-3	1.3E-2	1.8E-2	7.6E-3	1.3E-2	1.7E-2	1.5E-2	2.6E-2	3.4E-2	1.5E-02	2.5E-2	2.8E-2
	S_{rec}	5.7E-2	5.7E-2	5.8E-2	5.6E-2	5.7E-2	5.7E-2	1.1E-1	1.1E-1	1.1E-1	1.0E+0	1.1E-1	8.5E-1
RPCA_HQF	L_{rec}	8.7E-5	3.9E-5	1.2E-7	9.3E-5	3.9E-5	1.4E-7	5.4E-4	7.7E-4	7.8E-7	5.4E-4	7.3E-4	1.1E-6
	S_{rec}	6.2E-4	1.6E-4	2.8E-7	6.7E-4	1.6E-4	3.3E-7	3.9E-3	3.2E-3	1.8E-6	4.0E-3	3.1E-3	2.6E-6
LRPCA	L_{rec}	2.1E-3	1.5E-4	3.2E-4	4.5E-6	1.3E-4	4.9E-4	5.3E-2	5.2E-2	6.1E-2	1.8E-2	1.9E-2	1.9E-2
	S_{rec}	1.7E-2	6.2E-4	3.4E-3	6.8E-5	5.3E-4	5.6E-3	3.8E-1	2.2E-1	1.9E-1	1.4E-1	8.2E-2	6.5E-2
LRPCA [†]	L_{rec}	2.6E-8	5.7E-8	1.6E-7	3.7E-8	8.7E-8	2.5E-7	3.2E-2	3.3E-2	3.1E-2	1.3E-2	6.2E-3	6.1E-3
	S_{rec}	1.9E-7	2.8E-6	4.2E-6	4.3E-6	4.3E-6	6.9E-6	2.2E-1	1.4E-1	1.0E-1	1.0E-1	2.7E-2	2.0E-2
LERE	L_{rec}	7.6E-9	2.1E-8	2.9E-8	1.9E-8	1.4E-8	3.5E-8	7.8E-9	1.8E-8	5.8E-8	1.7E-8	1.7E-8	4.8E-8
	S_{rec}	5.4E-8	8.6E-8	9.7E-8	1.4E-7	6.1E-8	1.2E-7	5.6E-8	7.6E-8	1.9E-7	1.2E-7	7.4E-8	1.6E-6

Table 2: Comparison of various methods with different dimensions n_2 , sparsity α , conditional number κ , where “LRPCA”: default 17 iterations and “LRPCA[†]”: 200 iterations.

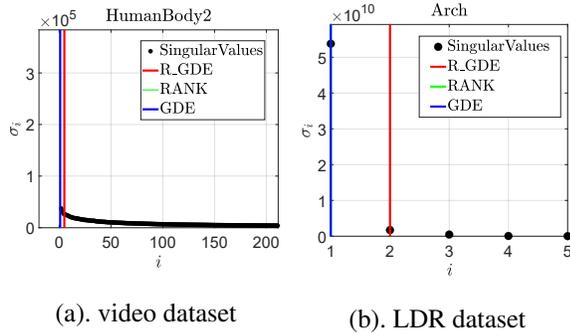


Figure 5: Rank estimation of the real world datasets.

comes are depicted in Fig. 4. Observations show that both GDE and RANK struggle to precisely capture the low-rank component when conducting rank estimation on sparse matrices within this database. In Fig. 4 (a) and (b), both GDE and RANK solely capture the low-rank component associated with the most prominent singular value. While they manage to eliminate the impact of the largest singular value in Fig. 4 (c), the low-rank component’s boundary still poses a challenge for accurate identification by both methods. In Fig. 4(d), the performance of RANK falters due to the influence caused by the non-ascending arrangement of Gerschgorin disk radii. In contrast, the proposed R_GDE method effectively and accurately estimates the rank of various sparse matrices.

Furthermore, we also conduct evaluation on the video dataset from the Scene Background Initialization (SBI) datasets² and the Low Dynamic Range (LDR) datasets³. These datasets are transformed into full matrices for evalua-

²<https://sbmi2015.na.icar.cnr.it/SBIdataset.html>

³<http://alumni.soe.ucsc.edu/orazio/deghost.html>

tion and Fig. 5 displays the corresponding results. From the plot, one can observe that whether it is the SBI datasets or the LDR datasets, the subspace corresponding to the largest singular value contains static background information from the video scene. GDE is able to accurately estimate the numerical rank corresponding to this part of the information. However, the subspaces corresponding to the secondary singular values often contain other background information. For instance, in the LDR dataset, the subspace related to the second singular value frequently encompasses the required illumination change information, which both GDE and RANK have overlooked. Compared to these two methods, R_GDE can effectively capture these secondary pieces of information, just as it can efficiently estimate the low-rank sub-space boundaries in ill-conditioned matrices.

Comparison of Various RPCA

Synthetic Datasets For synthetic datasets, we compare LERE against SOTA RPCA-based methods categorized into two groups: The first category is based on the estimated rank, including ARE-RPCA (Xu et al. 2021a) and ADW-RPCA (Xu et al. 2021b). The second group is grounded on prior rank, featuring AccAltProj (Cai, Cai, and Wei 2019), ScaledGD (Tong, Ma, and Chi 2021), RPCA_HQF (Wang et al. 2023), and LRPCA (Cai, Liu, and Yin 2021).

We corrupt the input matrix $\mathbf{Y} \in \mathbb{R}^{n_1 \times n_2}$ by sparse noise with varying corruption rates $\alpha = \{0.1, 0.3, 0.5\}$ and parameters: $n_1 = 3000$, $n_2 = \{3000, 1000, 500\}$, rank $r = 5$; the iteration stop criterion is $\|\mathbf{Y} - \mathbf{L}_* - \mathbf{S}_*\|_F / \|\mathbf{Y}\|_F < 10^{-7}$, where \mathbf{L}_* and \mathbf{S}_* are the reconstruction matrices; The reconstruction errors denote as $L_{rec} = \|\mathbf{L}_* - \mathbf{L}\|_F / \|\mathbf{L}\|_F$ and $S_{rec} = \|\mathbf{S}_* - \mathbf{S}\|_F / \|\mathbf{S}\|_F$; the maximum number of iterations is 200.

Table 2 presents the results. While the comparative methods might outperform in certain scenarios, their generaliza-

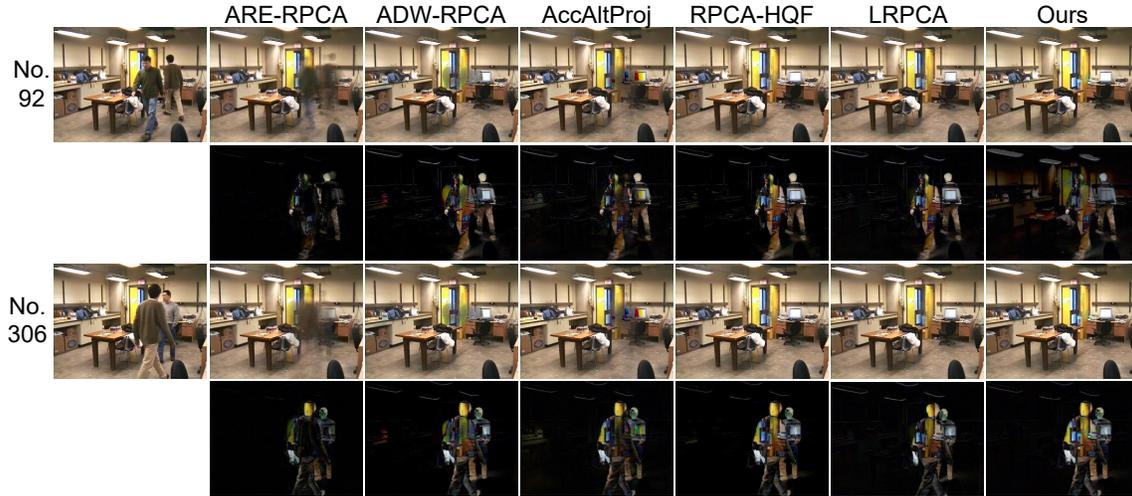


Figure 6: Visualization results of foreground-background separation task on HumanBody2 dataset. The first column contains the original 92-nd and 306-th frames, and the next seven columns are the results of other methods for comparison and ours. The results show that our LERE performs more precise separation for both of these two frames.

	AGE	pEPs%	pCEPS%	MSSSIM	PSNR	CQM
ARE_RPCA	8.024	10.894	7.483	0.934	22.97	22.58
ADW_RPCA	5.425	2.825	1.683	0.964	26.84	26.54
AccAltProj	7.860	5.773	3.272	0.927	23.07	22.58
RPCA-HQF	1.457	1.131	0.250	0.995	36.34	35.96
LRPCA	2.821	0.723	0.106	0.996	35.41	34.71
LRER	2.422	0.162	0.032	0.997	37.64	37.01

Table 3: A comparison of performance of different methods on one frame from HumanBody2 dataset.

tion performance is limited. As evident from Table 2, increasing the number of iterations for LRPCA from the default value 17 to 200, a marked improvement in its performance can be observed. Nevertheless, its performance remains constrained by matrix dimensions. In contrast, the proposed LERE consistently delivers competitive results across varying matrix dimensions, condition numbers, and sparsity. Meanwhile, LERE only needs training a 17-iteration FRMNN and obviates the need for retraining with different sparsity. Hence, it is evident from the above results that LERE possesses excellent robustness.

Foreground-Background Separation In this work, the proposed LERE is employed for the task of separating foreground from background. We choose the SBI dataset for this task. With each column corresponding to a vectorized frame, this dataset can be interpreted as a matrix. The moving foreground object represents the sparse part of the dataset, while the static background constitutes the low-rank portion. Hence we can separate the foreground and the background via the proposed LERE. The visualized results of LERE, ARE-RPCA, and ADW-RPCA on the sub-dataset HumanBody2 of the SBI dataset are shown in Fig. 6, which contain

740 frames with size 240×320 . Furthermore, we also carry out objective evaluations (i.e., AGE, pEP, pCEP, PSNR, MSSSIM, CQM) on HumanBody2. The findings, displayed in Table 3, indicate that SOTA methods are surpassed by the proposed LERE in terms of performance.

Conclusions and Future Work

A novel learning-based LRMR method, LERE, is presented in this work. LERE greatly surpasses prior arts in terms of both generalization performance and recovery accuracy. Concretely, a robust rank estimation method R_GDE is developed to estimate the rank of the ill-conditioned matrix with significant sparsity. The low-rank matrix is more optimally recovered using a novel learning-based approach built upon the estimated rank. The update process of the low-rank matrix involves uniformly random sampling. Consequently, the proposed method only requires a 17-iteration FRMNN to learn one set of parameters on the small sparsity dataset, adaptively preventing the model from falling into the local optima. Across both synthetic datasets and various partial vision tasks, extensive experimental results have consistently highlighted the superiority of our approach over SOTA methods.

In the future, our method will be extended to more vision applications. Motivated by the fact that structures of real-world data for large-scale problems tend to be more tensor-like than matrix-like, we will be dedicated to exploring the integration of tensor decomposition theories, and deep learning to address practical large-scale low-rank tensor recovery problems in our future research.

Acknowledgements

This work was supported by NSFC 62376156, 62201342, U19B2035, and Shanghai Municipal Science and Technology Major Project 2021SHZDZX0102.

References

- Cabral, R.; De la Torre, F.; Costeira, J. P.; and Bernardino, A. 2013. Unifying nuclear norm and bilinear factorization approaches for low-rank matrix decomposition. In *Proceedings of the IEEE international conference on computer vision (ICCV)*, 2488–2495.
- Cai, H.; Cai, J.-F.; and Wei, K. 2019. Accelerated alternating projections for robust principal component analysis. *The Journal of Machine Learning Research*, 20(1): 685–717.
- Cai, H.; Hamm, K.; Huang, L.; Li, J.; and Wang, T. 2021a. Rapid Robust Principal Component Analysis: CUR Accelerated Inexact Low Rank Estimation. *IEEE Signal Processing Letters*, 28: 116–120.
- Cai, H.; Hamm, K.; Huang, L.; and Needell, D. 2021b. Robust CUR Decomposition: Theory and Imaging Applications. *SIAM Journal on Imaging Sciences*, 14(4): 1472–1503.
- Cai, H.; Liu, J.; and Yin, W. 2021. Learned robust PCA: A scalable deep unfolding approach for high-dimensional outlier detection. *Advances in Neural Information Processing Systems (NIPS)*, 34: 16977–16989.
- Candès, E. J.; Li, X.; Ma, Y.; and Wright, J. 2011. Robust Principal Component Analysis? *Journal of the Association for Computing Machinery*, 58(3): 1–37.
- Chen, Y.; Chi, Y.; Fan, J.; Ma, C.; and Yan, Y. 2020. Noisy matrix completion: Understanding statistical guarantees for convex relaxation via nonconvex optimization. *SIAM journal on optimization*, 30(4): 3098–3121.
- Chen, Y.; Fan, J.; Ma, C.; and Yan, Y. 2021. Bridging convex and nonconvex optimization in robust PCA: Noise, outliers and missing data. *The Annals of Statistics*, 49(5): 2948–2971.
- Cohen, R.; Zhang, Y.; Solomon, O.; Toberman, D.; Taieb, L.; van Sloun, R. J.; and Eldar, Y. C. 2019. Deep Convolutional Robust PCA with Application to Ultrasound Imaging. In *2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 3212–3216.
- Gregor, K.; and LeCun, Y. 2010. Learning Fast Approximations of Sparse Coding. In *Proceedings of the 27th International Conference on International Conference on Machine Learning (ICML)*, 399–406. ISBN 9781605589077.
- Hu, Y.; Zhang, D.; Ye, J.; Li, X.; and He, X. 2012. Fast and accurate matrix completion via truncated nuclear norm regularization. *IEEE transactions on pattern analysis and machine intelligence*, 35(9): 2117–2130.
- Jia, X.; Feng, X.; Wang, W.; and Zhang, L. 2020. Generalized unitarily invariant gauge regularization for fast low-rank matrix recovery. *IEEE Transactions on Neural Networks and Learning Systems*, 32(4): 1627–1641.
- Keshavan, R. H.; Montanari, A.; and Oh, S. 2010. Matrix completion from a few entries. *IEEE transactions on information theory*, 56(6): 2980–2998.
- Li, Z.; Nie, F.; Wang, R.; and Li, X. 2023. Robust rank-one matrix completion with rank estimation. *Pattern Recognition*, 142: 109637.
- Ma, C.; Li, Y.; and Chi, Y. 2021. Beyond Procrustes: Balancing-free gradient descent for asymmetric low-rank matrix sensing. *IEEE Transactions on Signal Processing*, 69: 867–877.
- Markowitz, S.; Snyder, C.; Eldar, Y. C.; and Do, M. N. 2022. Multimodal unrolled robust PCA for background foreground separation. *IEEE Transactions on Image Processing*, 31: 3553–3564.
- Mishra, B.; Apuroop, K. A.; and Sepulchre, R. 2012. A Riemannian geometry for low-rank matrix completion. *arXiv preprint arXiv:1211.1550*.
- Monga, V.; Li, Y.; and Eldar, Y. C. 2021. Algorithm unrolling: Interpretable, efficient deep learning for signal and image processing. *IEEE Signal Processing Magazine*, 38(2): 18–44.
- Park, D.; Kyrillidis, A.; Carmanis, C.; and Sanghavi, S. 2017. Non-square matrix sensing without spurious local minima via the Burer-Monteiro approach. In *Artificial Intelligence and Statistics (AIS)*, 65–74.
- Phan, D. N.; and Nguyen, T. N. 2021. An accelerated inner-iteratively reweighted nuclear norm algorithm for nonconvex nonsmooth low-rank minimization problems. *Journal of Computational and Applied Mathematics*, 396: 113602.
- Recht, B. 2011. A simpler approach to matrix completion. *Journal of Machine Learning Research*, 12(12): 3413–3430.
- Recht, B.; Fazel, M.; and Parrilo, P. A. 2010. Guaranteed minimum-rank solutions of linear matrix equations via nuclear norm minimization. *SIAM review*, 52(3): 471–501.
- Shen, J.; Chen, X.; Heaton, H.; Chen, T.; Liu, J.; Yin, W.; and Wang, Z. 2021. Learning a minimax optimizer: A pilot study. In *International Conference on Learning Representations (ICML)*.
- Shi, Q.; Lu, H.; and Cheung, Y.-M. 2017. Rank-one matrix completion with automatic rank estimation via L1-norm regularization. *IEEE transactions on neural networks and learning systems*, 29(10): 4744–4757.
- Solomon, O.; Cohen, R.; Zhang, Y.; Yang, Y.; He, Q.; Luo, J.; van Sloun, R. J.; and Eldar, Y. C. 2019. Deep unfolded robust PCA with application to clutter suppression in ultrasound. *IEEE transactions on medical imaging*, 39(4): 1051–1063.
- Tong, T.; Ma, C.; and Chi, Y. 2021. Accelerating ill-conditioned low-rank matrix estimation via scaled gradient descent. *The Journal of Machine Learning Research*, 22(1): 6639–6701.
- Van Luong, H.; Joukovsky, B.; Eldar, Y. C.; and Deligiannis, N. 2021. A deep-unfolded reference-based RPCA network for video foreground-background separation. In *2020 28th European Signal Processing Conference (EUSIPCO)*, 1432–1436.
- Wang, R.; Fang, H.; Zhang, Y.; Yu, L.; and Chen, J. 2022. Low-Rank Enforced Fault Feature Extraction of Rolling Bearings in a Complex Noisy Environment: A Perspective of Statistical Modeling of Noises. *IEEE Transactions on Instrumentation and Measurement*, 71: 1–14.
- Wang, Z.-Y.; Li, X. P.; So, H. C.; and Liu, Z. 2023. Robust PCA via non-convex half-quadratic regularization. *Signal Processing*, 204: 108816.

- Wen, Z.; Yin, W.; and Zhang, Y. 2012. Solving a low-rank factorization model for matrix completion by a nonlinear successive over-relaxation algorithm. *Mathematical Programming Computation*, 4(4): 333–361.
- Wu, H.-T.; Yang, J.-F.; and Chen, F.-K. 1995. Source number estimators using transformed Gerschgorin radii. *IEEE transactions on signal processing*, 43(6): 1325–1333.
- Xu, C.; Lin, Z.; and Zha, H. 2017. A unified convex surrogate for the Schatten-p norm. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, 926–932.
- Xu, Z.; He, R.; Xie, S.; and Wu, S. 2021a. Adaptive Rank Estimate in Robust Principal Component Analysis. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 6573–6582.
- Xu, Z.; Xing, H.; Fang, S.; Wu, S.; and Xie, S. 2021b. Double-Weighted Low-Rank Matrix Recovery Based on Rank Estimation. In *2021 IEEE/CVF International Conference on Computer Vision Workshops (ICCVW)*, 172–180.
- Yi, X.; Park, D.; Chen, Y.; and Caramanis, C. 2016. Fast algorithms for robust PCA via gradient descent. *Advances in neural information processing systems (NIPS)*, 29.
- Zadeh Kashani, S. M.; and Hamidzadeh, J. 2020. Improvement of non-negative matrix-factorization-based and Trust-based approach to collaborative filtering for recommender systems. In *2020 6th Iranian Conference on Signal Processing and Intelligent Systems (ICSPIS)*, 1–7.
- Zhao, Q.; Zhang, L.; and Cichocki, A. 2015. Bayesian CP factorization of incomplete tensors with automatic rank determination. *IEEE transactions on pattern analysis and machine intelligence*, 37(9): 1751–1763.