

Efficient Differentiable Approximation of Generalized Low-rank Regularization

Naiqi Li¹, Yuqiu Xie¹, Peiyuan Liu¹, Tao Dai^{2,*}, Yong Jiang^{1,*} and Shu-Tao Xia¹

¹Tsinghua Shenzhen International Graduate School

²Shenzhen University

{linaiqi, jiangy, xiaist}@sz.tsinghua.edu.cn, {jieyq22, lpy23}@mails.tsinghua.edu.cn,
daitao.edu@gmail.com

Abstract

Low-rank regularization (LRR) has been widely applied in various machine learning tasks, but the associated optimization is challenging. Directly optimizing the rank function under constraints is NP-hard in general. To overcome this difficulty, various relaxations of the rank function were studied. However, optimization of these relaxed LRRs typically depends on singular value decomposition, which is a time-consuming and nondifferentiable operator that cannot be optimized with gradient-based techniques. To address these challenges, in this paper we propose an efficient differentiable approximation of the generalized LRR. The considered LRR form subsumes many popular choices like the nuclear norm, the Schatten- p norm, and various nonconvex relaxations. Our method enables LRR terms to be appended to loss functions in a plug-and-play fashion, and the GPU-friendly operations enable efficient and convenient implementation. Furthermore, convergence analysis is presented, which rigorously shows that both the bias and the variance of our rank estimator rapidly reduce with increased sample size and iteration steps. In the experimental study, the proposed method is applied to various tasks, which demonstrates its versatility and efficiency. Code is available at <https://github.com/naiqili/EDLRR>.

1 Introduction

Low-rank structures have proven to be effective in a wide range of machine learning tasks, encompassing computer vision [Ren *et al.*, 2022], model compression [Idelbayev and Carreira-Perpinán, 2020], representation learning [Fu *et al.*, 2021; Fu *et al.*, 2022], and large language model adaptation [Hu *et al.*, 2022a]. To discover and utilize the low-rank structures, a typical paradigm is to introduce low-rank regularization (LRR) terms to the models, which can conveniently express various low-rank priors and assumptions.

However, with the presence of LRR terms, the optimization problem can be extremely difficult. It is well-known that

even under linear constraints, optimizing the rank function is NP-hard [Wright and Ma, 2022]. To alleviate this computational intractability, many relaxations of the rank function have been proposed. For example, the nuclear norm is known to be the tightest convex approximation of the rank function, and thus has been extensively investigated and applied [Yang *et al.*, 2016]. The Schatten- p norm and its variants are considered as the generalization of the nuclear norm, and also found successful applications [Xu *et al.*, 2017; Chen *et al.*, 2021]. These relaxation techniques penalize all singular values of the target matrix simultaneously, while in practical applications it is desired that the large singular values are less impacted, so the key information of the target matrix is preserved. With this motivation, recent studies proposed novel nonconvex relaxations [Kang *et al.*, 2015; Peng *et al.*, 2015; Friedman, 2012; Gao *et al.*, 2011].

However, it is still challenging to efficiently optimize these models with relaxed low-rank regularization. Current optimization methods can be categorized as the matrix factorization approach and the rank function optimization approach, but both of them suffer severe shortcomings in practice.

The basic idea of matrix factorization is to decompose the target matrix into the product of multiple low-rank submatrices. This formulation allows gradient to be propagated, and so the optimization is straightforward. A key problem of this approach is that it demands strong prior knowledge about the matrix’s true rank or its upper bound. In almost all practical situations such knowledge is unavailable, and so the good performance frequently depends on laborious parameter selection. Arguably, the goal of rank regularization is to discover the true rank of a matrix, but this approach leaves this burden to the users.

In contrast, the rank function optimization approach incorporates a regularization term into the loss function, allowing the appropriate matrix rank to be automatically determined. However, solving the associated optimization problem is non-trivial. Previous works attempted to adapt techniques from the convex optimization literature, such as the proximal gradient and alternating direction method of multipliers (ADMM). Unfortunately, many of these methods require the loss function to be convex, which severely limits their applicability. Additionally, these approaches are inconvenient to implement, and each work needs to laboriously derive the specific optimization rules. Critically, all these methods rely

*Corresponding authors: Tao Dai and Yong Jiang.

on singular value decomposition (SVD). Besides being time-consuming, SVD is generally a nondifferentiable operation that hinders gradient propagation (although there exists sophisticated method for computing SVD that allows gradient propagation, it incurs $O(D^4)$ computational complexity which is prohibitive for practical applications [Papadopoulos and Lourakis, 2000]). Consequently, applying gradient-based optimization techniques or utilizing popular deep learning libraries (e.g., PyTorch and TensorFlow) becomes infeasible, and high-performing GPUs cannot be fully utilized.

In this paper, we propose a novel differentiable approximation of a general form of LRR, which covers a broad range of relaxations of the rank function. The main idea of our work is to introduce an equivalent stochastic definition of the rank function, as well as its relaxed variants. This significant discovery enables us to approximate the LRR term with finite samples and a partial sum of the series expansion. Our implementation is publicly available in the supplementary material. The main contributions of this paper are summarized as follows:

1. We propose an efficient differentiable approximation of the generalized LRR, which covers the nuclear norm, the Schatten- p norm, and many recently proposed nonconvex relaxations of the rank function.
2. The proposed LRR approximation is convenient and efficient. It can be directly appended to loss functions and optimized by existing deep learning libraries. In most cases, a few lines of code are sufficient to adapt to new problems. The operations are GPU-friendly, which enable highly parallel and efficient computation.
3. Theoretical convergence analysis is presented, which rigorously demonstrates that both the bias and the variance of the proposed rank estimator rapidly reduce with increased sample size and iteration steps.
4. We performed extensive experiments over various tasks, including matrix completion, video fore-background separation, and image denoising, which demonstrate the versatility and efficiency of our proposed method.

2 Related Work

2.1 Relaxations of the Rank Function

Minimizing the rank function is challenging, and even finding the optimal solution under the linear constraint is NP-hard [Wright and Ma, 2022]. To address this challenge, various relaxations of the rank function have been proposed. The nuclear norm $\|\mathbf{S}\|_* = \sum_{i=1}^r \sigma_i(\mathbf{S})$ is the tightest convex relaxation of the rank function, and is one of the most popular substitution [Yang *et al.*, 2016]. However, the nuclear norm penalizes all singular values simultaneously. Since for many matrices in practical applications, the major information is captured by a few singular values, so it is desired that they are less impacted when reducing the rank. Motivated by this, advanced relaxations of the general form $\mathcal{R}(\mathbf{S}) = \sum_{i=1}^r h(\sigma_i(\mathbf{S}))$ are considered, where h is a function that increases penalties on small singular values. Typical examples of the relaxed rank function include the γ -nuclear norm [Kang *et al.*, 2015], Laplace [Trzasko and Manduca, 2008],

LNN [Peng *et al.*, 2015], Logarithm [Friedman, 2012], ETP [Gao *et al.*, 2011], and Geman [Geman and Yang, 1995]. A summary of these relaxations and the corresponding penalty function h can be found in [Hu *et al.*, 2021].

2.2 Optimization of the Low-rank Regularization

Matrix factorization and rank function optimization are two prominent approaches for optimizing low-rank models.

The concept of matrix factorization involves decomposing the target matrix into the product of multiple low-rank components. Under this formulation, the optimization is straightforward. Notable examples include the decomposition of weight matrices into low-rank factors. Additionally, Geng *et al.* demonstrated that this strategy enables neural networks to learn global information and can even replace the attention mechanism [Geng *et al.*, 2021]. Ornhag *et al.* (2020) introduced a differentiable bilinear parameterization of the nuclear norm, relying on matrix decomposition. The authors further proposed VarPro, which utilizes second-order optimization that enjoys faster convergence [Ornhag *et al.*, 2021]. This idea was further extended in [Xu *et al.*, 2017; Chen *et al.*, 2021], where the multi-Schatten- p norm was considered as a generalization of the bilinear parameterization. However, a significant challenge of this approach lies in the requirement of strong prior knowledge about the true rank of the matrix.

On the other hand, the rank function optimization approach introduces a regularization term to the loss, allowing the appropriate matrix rank to be automatically determined during training. Initial research attempts applied existing convex optimization techniques to the problem, such as the proximal gradient algorithm [Yao *et al.*, 2018] and the iteratively re-weighted algorithm [Mohan and Fazel, 2012]. However, these methods require the loss function to be convex. Alternating direction method of multipliers (ADMM) is a popular method for optimizing the regularized loss. Shang *et al.* (2017) proposed the double nuclear penalty, which covers the Schatten- p norm with $p = 1/2$ and $2/3$. Recently, it is discovered that one of the optimization sub-procedures within ADMM can be seen as performing image denoising, enabling the integration of existing denoising neural networks into the ADMM framework [Hu *et al.*, 2022b; Liu *et al.*, 2023]. However, all these ADMM based methods do not support gradient propagation, so deep learning libraries and high-performing GPUs cannot be fully utilized.

There are a few works that consider SVD-free optimization of LRRs, which draw inspirations from the variational characterization of the nuclear norm, i.e., $\|\mathbf{X}\|_* = \min_{\mathbf{A}\mathbf{B}=\mathbf{X}} \frac{1}{2}(\|\mathbf{A}\|_F^2 + \|\mathbf{B}\|_F^2)$, where $\mathbf{X} \in \mathbb{R}^{m \times n}$, $\mathbf{A} \in \mathbb{R}^{m \times d}$, $\mathbf{B} \in \mathbb{R}^{d \times n}$ and $d \geq \text{rank}(\mathbf{X})$ [Srebro *et al.*, 2004; Rennie and Srebro, 2005]. Recent research further extends these methods to handle the Schatten- p quasi-norm, i.e., $\|\mathbf{x}\|_p = (\sum_i |x_i|^p)^{\frac{1}{p}}$ with $p \in (0, 1)$ [Shang *et al.*, 2016; Fan *et al.*, 2019; Giampouras *et al.*, 2020]. Jia *et al.* (2020) proposed GUIG for low-rank matrix recovery, and the associated bilinear variational problem can be solved without computing SVD.

To summarize, all these methods suffer at least one of

the following drawbacks: 1) Their optimization is based on ADMM or its variant, which hinder gradient propagation. 2) They require the upper bound of the true rank d as input. When d is too large, the slow convergence brings huge computational burden, while a too small d deteriorates the performance. 3) They can only optimize a restricted class of LRRs, particularly the nuclear norm and the Schatten- p quasi-norm with $p \in (0, 1)$. They cannot be applied to various recently proposed nonconvex LRRs like the γ -Nuclear norm [Kang *et al.*, 2015] and Laplace [Trzasko and Manduca, 2008].

3 Methodology

3.1 Notations and Problem Statement

Notations In this paper, uppercase bold letters (*e.g.*, \mathbf{X}) denote matrices, and lowercase bold letters (*e.g.*, \mathbf{x}) denote vectors. For a vector \mathbf{x} , $\|\mathbf{x}\|_p = (\sum_i |x_i|^p)^{\frac{1}{p}}$ represents its l_p -norm. For a matrix \mathbf{X} , $\sigma_i(\mathbf{X})$ is its i th largest singular value, or abbreviated as σ_i . We use $\mathbf{X} \succeq 0$ to indicate matrix \mathbf{X} is positive semi-definite. $\|\mathbf{X}\|_p = (\sum_{i=1}^r \sigma_i(\mathbf{X})^p)^{\frac{1}{p}}$ is the Schatten- p norm, where r is the rank of \mathbf{X} . Particularly, the nuclear norm is $\|\mathbf{X}\|_* = \|\mathbf{X}\|_1$, the spectral norm is $\|\mathbf{X}\| = \sigma_1(\mathbf{X})$, and the rank is equivalently represented as $\|\mathbf{X}\|_0$. \mathbf{X}^\top and \mathbf{X}^\dagger denote the transpose and the pseudo-inverse of the matrix respectively. $\text{span}(\mathbf{X})$ is the linear space spanned by the columns of \mathbf{X} , and $P_{\mathbf{X}}[\mathbf{u}] = \mathbf{X}\mathbf{X}^\dagger\mathbf{u}$ denotes the projection of the vector \mathbf{u} onto the column space $\text{span}(\mathbf{X})$.

Problem statement For a given input-output pair (\mathbf{X}, \mathbf{Y}) where $\mathbf{X} \in \mathbb{R}^{a \times b}$ and $\mathbf{Y} \in \mathbb{R}^{c \times d}$, consider the empirical loss w.r.t. a learning function f parameterized by θ , denoted as:

$$\mathcal{L}(\mathbf{X}, \mathbf{Y}, \theta) = l(f(\mathbf{X}; \theta), \mathbf{Y}) + \mathcal{R}(\mathbf{S}), \quad (1)$$

where $\mathbf{S} = g(\mathbf{X}, \mathbf{Y}, \theta) \in \mathbb{R}^{m \times n}$. Here the matrix \mathbf{S} can depend on both the data and function parameters.

The regularization term $\mathcal{R}(\mathbf{S})$ enforces \mathbf{S} to be low-rank. Ideally, the definition could be directly applied, *i.e.*, $\mathcal{R}(\mathbf{S}) = \|\mathbf{S}\|_0$. However, this regularization term is nondifferentiable, and the associated optimization problem is NP-hard in general. To address this problem, in this work $\mathcal{R}(\mathbf{S})$ will represent some form of approximation or relaxation of the rank. For example, it is well-known that the nuclear norm (*i.e.*, $\|\mathbf{S}\|_* = \sum_{i=1}^r \sigma_i(\mathbf{S})$) is the convex envelope of the rank function [Wright and Ma, 2022], and is used as the surrogate function in many works. However, the nuclear norm penalizes all singular values simultaneously, while in practical problems it is desired that large singular values are less impacted, so the important information of the matrix is preserved. Motivated by this observation, many nonconvex relaxations of the rank function have been introduced [Kang *et al.*, 2015; Peng *et al.*, 2015; Friedman, 2012; Gao *et al.*, 2011]. To encompass all these formulations, in this work the regularization term is considered to be of the form:

$$\mathcal{R}(\mathbf{S}) = \sum_{i=1}^r h(\sigma_i(\mathbf{S})). \quad (2)$$

Here function h generally increases the penalty of small singular values. In this paper, we refer to this form of regularization as the **generalized low-rank regularization (LRR)**.

The goal of this paper is to develop a method to approximately compute the generalized LRR $\mathcal{R}(\mathbf{S})$ in Eq. (2), which is a differentiable operation that allows the gradients to be computed and propagated. Thus $\mathcal{R}(\mathbf{S})$ can be conveniently used in a plug-and-play fashion, and the optimization problem in Eq. (1) can be conveniently solved by off-the-shelf optimization frameworks, and utilize high-performing GPUs for efficient parallel computation.

Several concrete applications are presented below:

Matrix completion Suppose $\mathbf{S} \in \mathbb{R}^{m \times n}$ is a low-rank matrix, $\Omega \subset \{1, \dots, m\} \times \{1, \dots, n\}$ is an index set that denotes the observable entries, and $P_\Omega[\mathbf{S}]_{ij} = \mathbf{1}_{\{(i,j) \in \Omega\}} \cdot \mathbf{S}_{ij}$ represents the observation projection. The problem of matrix completion aims to recover \mathbf{S} based on partial observations, *i.e.*, $\min_{\mathbf{X}} \|P_\Omega[\mathbf{X}] - P_\Omega[\mathbf{S}]\|_F^2 + \lambda \mathcal{R}(\mathbf{X})$.

Video fore-background separation A video sequence is represented as a 3D tensor $\mathbf{V} \in \mathbb{R}^{a \times b \times t}$, where a, b denote the width and height of each frame, and t indexes time. Let $\mathbf{V}' \in \mathbb{R}^{ab \times t}$ be a reshaped 2D matrix. In video fore-background separation it is assumed that the reshaped matrix can be decomposed as $\mathbf{V}' = \mathbf{S} + \mathbf{O}$, where \mathbf{S} is a low-rank matrix that represents the background, and \mathbf{O} is a sparse matrix that represents the foreground object. So the problem can be solved by optimizing $\min_{\mathbf{X}} \|\mathbf{V}' - \mathbf{X}\|_1 + \lambda \mathcal{R}(\mathbf{X})$.

DNN-based image denoising In image denoising an observed image \mathbf{X} is considered as a clean image \mathbf{S} corrupted by noise \mathbf{N} , *i.e.*, $\mathbf{X} = \mathbf{S} + \mathbf{N}$. A DNN model learns a function f to predict the noise by optimizing $\min_{\theta} E[\|f(\mathbf{X}; \theta) - \mathbf{N}\|_F^2]$. Since the clean images are approximately low-rank, it regularizes the loss function as $\min_{\theta} E[\|f(\mathbf{X}; \theta) - \mathbf{N}\|_F^2 + \lambda \mathcal{R}(f(\mathbf{X}; \theta) - \mathbf{N})]$. Note that most existing work has difficulty in optimizing such general formulations.

3.2 Efficient Differentiable Low-rank Regularization

Iterative matrix pseudo-inverse and square root

We first show that two fundamental operations, *i.e.*, matrix pseudo-inverse and matrix square root, have differentiable approximations. These results serve as the cornerstones of the proposed method, as we later demonstrate that the general LRR can be constructed with these operations.

Proposition 1 (Iterative matrix pseudo-inverse [Ben-Israel and Cohen, 1966]). *For a given matrix $\mathbf{S} \in \mathbb{R}^{m \times n}$, define the recursive sequence $\mathbf{S}_{i+1} = 2\mathbf{S}_i - \mathbf{S}_i\mathbf{S}\mathbf{S}_i$, with $\mathbf{S}_0 = \alpha\mathbf{S}^\top$. Then $\lim_{i \rightarrow \infty} \mathbf{S}_i = \mathbf{S}^\dagger$, provided $0 < \alpha < 2/\sigma_1^2(\mathbf{S})$.*

Intuitively, when the iteration converges we have $\mathbf{S}_i = \mathbf{S}_{i+1}$. So $\mathbf{S}_i = \mathbf{S}_i\mathbf{S}\mathbf{S}_i$, which is exactly the definition of pseudo-inverse. In practice, taking a large enough iteration step N , results in a satisfactory approximation of the matrix's pseudo inverse, *i.e.*, $\mathbf{S}_N \approx \mathbf{S}^\dagger$. Furthermore, since the iterative computation only involves matrix multiplication and subtraction, it is obviously a differentiable operation.

As a consequence, the projection operator $P_{\mathbf{S}}[\mathbf{u}]$ also has a differentiable approximation. Recall that $P_{\mathbf{S}}[\mathbf{u}] = \mathbf{S}\mathbf{S}^\dagger\mathbf{u}$ denotes the projection of the vector \mathbf{u} onto the column space of matrix \mathbf{S} . So $P_{\mathbf{S}}[\mathbf{u}] \approx \mathbf{S}\mathbf{S}_N\mathbf{u}$, where \mathbf{S}_N is the approxima-

tion of the pseudo-inverse computed by the iterative method. Again the r.h.s. is obviously differentiable.

The matrix square root can also be computed by an iterative method, which is called the Newton-Schulz iteration:

Proposition 2 (NS iteration for matrix square root). *For $\mathbf{A} \in \mathbb{R}^{m \times m}$, initialize $\mathbf{Y}_0 = \frac{1}{\|\mathbf{A}\|_F} \mathbf{A}$, $\mathbf{Z}_0 = \mathbf{I}$. The Newton-Schulz method defines the following iteration:*

$$\mathbf{Y}_{k+1} = \frac{1}{2} \mathbf{Y}_k (3\mathbf{I} - \mathbf{Z}_k \mathbf{Y}_k), \mathbf{Z}_{k+1} = \frac{1}{2} (3\mathbf{I} - \mathbf{Z}_k \mathbf{Y}_k) \mathbf{Z}_k.$$

Then $\sqrt{\|\mathbf{A}\|_F} \mathbf{Y}_k$ quadratically converges to $\mathbf{A}^{\frac{1}{2}}$, i.e., $\sqrt{\|\mathbf{A}\|_F} \mathbf{Y}_k \rightarrow \mathbf{A}^{\frac{1}{2}}$.

We highlight the following points of Proposition 1 and 2: 1) Both are **differentiable** operations, which allow gradient-based methods for the associated optimization; 2) Both are **parallelizable** operations, which are GPU-friendly and thus efficient for large-scale datasets. Our proposed method inherits these advantages, which results in differentiable and parallelizable approximation of the generalized LRR.

Differentiable rank approximation

The rank of a matrix is defined as the dimension of its column space (or equivalently the row space). The most well-known method for computing the rank is to first apply SVD, and then count the number of nonzero singular values. However, the SVD step is nondifferentiable. Interestingly, the following proposition presents an alternative approach for computing ranks without using SVD. Due to space limitation, all the proofs are deferred to the Appendix.

Proposition 3 (Equivalent definition of matrix rank [Wright and Ma, 2022]). *The rank of a matrix \mathbf{S} can be equivalently computed as the average squared length of a random Gaussian vector $\mathbf{g} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ projected onto its column space:*

$$\|\mathbf{S}\|_0 = \text{rank}(\mathbf{S}) = E [\|P_{\mathbf{S}}[\mathbf{g}]\|_2^2]. \quad (3)$$

To apply the result for practical rank computation, first sample N independent random Gaussian vectors $\mathbf{g}_1, \dots, \mathbf{g}_N \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$. Then the sample average is used to approximate the rank, i.e., $\|\mathbf{S}\|_0 = E [\|P_{\mathbf{S}}[\mathbf{g}]\|_2^2] \approx \frac{1}{N} \sum_{i=1}^N \|P_{\mathbf{S}}[\mathbf{g}_i]\|_2^2$. Since it has been shown that the projection operator $P_{\mathbf{S}}[\mathbf{g}]$ has a differentiable approximation, by substituting the routine we obtain a differentiable method for calculating the matrix rank.

However, rank is a piecewise constant function. Though it is now differentiable, it cannot provide useful gradient information for the overall optimization problem. To address this challenge, in the following we consider several relaxations of matrix rank, the approximations of which are not only differentiable, but also provide informative gradients.

Differentiable nuclear norm approximation

The nuclear norm is the convex envelope of rank, and is the most popular relaxation. The common approach for computing the nuclear norm is to first apply SVD, and then sum up the singular values. The following proposition presents an alternative method, similar to the case of rank calculation.

Proposition 4. *The nuclear norm of a matrix \mathbf{S} can be equivalently computed as:*

$$\|\mathbf{S}\|_* = E [\langle P_{\mathbf{S}}[\mathbf{g}], (\mathbf{S}\mathbf{S}^\top)^{\frac{1}{2}} \mathbf{g} \rangle], \quad (4)$$

where $\mathbf{g} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ is a random Gaussian vector.

Note that besides the projection operator $P_{\mathbf{S}}[\mathbf{g}]$, this proposition further requires calculating the matrix square root $(\mathbf{S}\mathbf{S}^\top)^{\frac{1}{2}}$, which can be obtained with Proposition 2. Finally, by replacing the expectation with the sample mean, the nuclear norm can be approximately computed using differentiable operators. Furthermore, the approximated nuclear norm can provide useful gradient information for optimization.

Differentiable approximation of the generalized low-rank regularization

In this subsection, we consider general LRR of the form $\mathcal{R}(\mathbf{S}) = \sum_{i=1}^r h(\sigma_i(\mathbf{S}))$, where r denotes the rank of \mathbf{S} . The function h increases the penalty of the small singular values, so that the principle information of the matrix is preserved while reducing the rank.

First, we introduce a lemma that allows any Schatten- p norm to be stochastically computed, which can be viewed as an extension of Proposition 3 and 4.

Theorem 1. *For a matrix \mathbf{S} , its Schatten- p norm, defined as $\|\mathbf{S}\|_p = (\sum_{i=1}^r \sigma_i(\mathbf{S})^p)^{\frac{1}{p}}$ where r is the rank of \mathbf{S} , can be alternatively computed as*

$$\|\mathbf{S}\|_p^p = \sum_{i=1}^r \sigma_i^p = E [\langle P_{\mathbf{S}}[\mathbf{g}], (\mathbf{S}\mathbf{S}^\top)^{\frac{p}{2}} \mathbf{g} \rangle], \quad (5)$$

where $p \in \mathbb{N}^+$ and $\mathbf{g} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$.

Next we show that a broad range of relaxations of the rank function can be stochastically computed. Particularly, we introduce two methods to utilize the above lemma, which are based on the Taylor expansion and the Laguerre expansion.

Taylor Expansion-based Generalized LRR

Theorem 2. *Let \mathbf{S} be a matrix of rank r , and $h : \mathbb{R} \rightarrow \mathbb{R}$ be a sufficiently smooth function and $\mathbf{g} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$. Then the generalized LRR defined in Eq. (2) can be computed as*

$$\sum_{i=1}^r h(\sigma_i(\mathbf{S})) = \sum_{p=0}^{\infty} \frac{h^{(p)}(0)}{p!} E [\langle P_{\mathbf{S}}[\mathbf{g}], (\mathbf{S}\mathbf{S}^\top)^{\frac{p}{2}} \mathbf{g} \rangle]. \quad (6)$$

However, Taylor expansion approximates the target function based on a fixed initial point, and the truncated error grows when the evaluation location moves away from the initial point. This motivates the application of advanced approximation techniques.

Laguerre Expansion-based Generalized LRR

Orthogonal polynomial approximation is an important family of function approximation techniques, including the well-known Legendre expansion and Laguerre expansion. In our problem the Laguerre expansion is particularly suitable, since it approximates functions with range $(0, +\infty)$.

The Laguerre expansion is based on the Laguerre polynomials $\mathcal{L} = \{L_1(x), L_2(x), \dots\}$, which is a countable infinite set of mutually orthogonal polynomials, each denoted

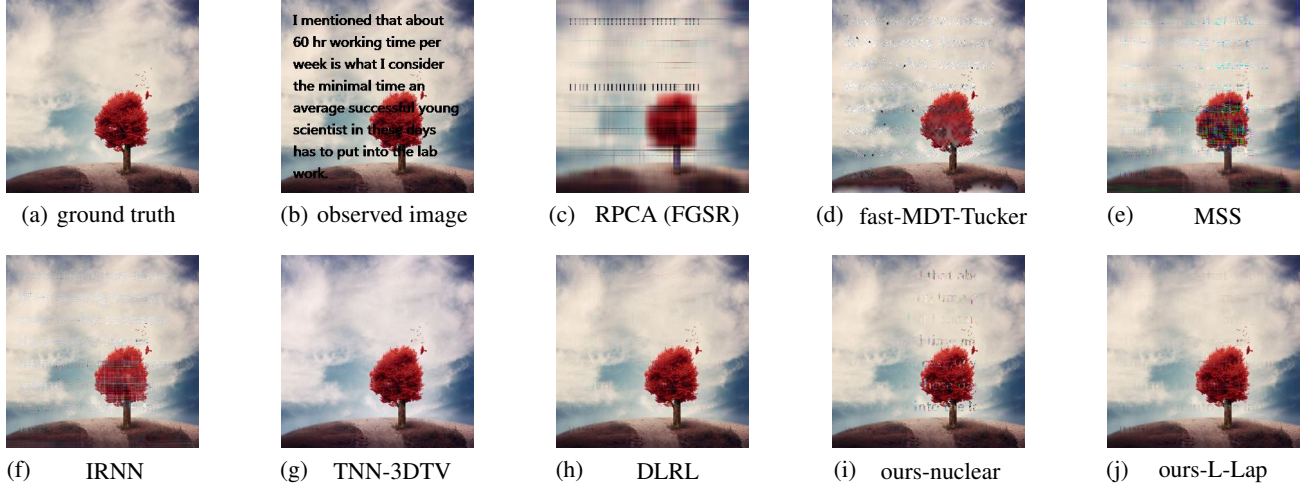


Figure 1: Comparison of matrix completion for text removal. (a) ground truth; (b) image with text; (c)-(j) recovered images.

as $L_k(x) = \sum_p a_{k,p} x^p$. For a target function $f(x)$ with range $(0, +\infty)$, the method decomposes it into $f(x) = \sum_{k \geq 0} c_k L_k(x)$, where c_k are the coefficients computed as $c_k = \int_0^\infty L_k(x) e^{-x} f(x) dx$.

Theorem 3. *Keeping the same notations, we have*

$$\sum_{i=1}^r h(\sigma_i(\mathbf{S})) = \sum_{k \geq 0} \sum_p c_k a_{k,p} E \left[\langle P_{\mathbf{S}}[\mathbf{g}], (\mathbf{S}\mathbf{S}^\top)^{\frac{p}{2}} \mathbf{g} \rangle \right].$$

Here $h(x) = \sum_{k \geq 0} c_k L_k(x)$ utilizes the Laguerre expansion, $a_{k,p}$ are the polynomial coefficients.

In both Theorem 2 and 3, by using a finite sum to approximate the infinite series, the general regularization form becomes differentiable, and can provide useful gradients for the optimization. Furthermore, the computation solely depends on matrix multiplication, which is a GPU-friendly operation that allows highly efficient parallel implementation.

3.3 Algorithm and convergence analysis

In what follows, we present the algorithm for computing the differentiable approximation of the Schatten- p norm (Theorem 1) and the main convergence result. Due to space limit, detailed proofs, descriptions of applying truncated series expansion (Theorem 2 and 3), and further corresponding convergence analysis are left in the Appendix.

Theorem 4. *We use underline notations (i.e., $\underline{P_{\mathbf{S}}[\mathbf{g}_i]}$ and $\underline{(\mathbf{S}\mathbf{S}^\top)^{\frac{p}{2}}}$) to denote the approximation results obtained by the iterative methods, with iteration steps k_1 for matrix pseudo inverse and k_2 for matrix root. Let $X = \frac{1}{N} \sum_{i=1}^N \langle \underline{P_{\mathbf{S}}[\mathbf{g}_i]}, \underline{(\mathbf{S}\mathbf{S}^\top)^{\frac{p}{2}}} \mathbf{g}_i \rangle$. Then for any $\epsilon > 0$,*

$$\Pr \left(\frac{|X - \sum_i \sigma_i^p|}{\sum_i \sigma_i^p} \leq \epsilon \right) \geq 1 - \frac{2C(k_1, k_2)^2 \|\mathbf{S}\|_{2p}^{2p}}{N (\sum_i \sigma_i^p (\epsilon + E(k_1, k_2)))^2},$$

where $C(k_1, k_2) \rightarrow 1$, $E(k_1, k_2) \rightarrow 0$ exponentially as k_1 and k_2 increase.

Algorithm 1 Differentiable approximation of $\|\mathbf{S}\|_p^p$

Require: $\mathbf{S} \in \mathbb{R}^{m \times m}$, p , sample sizes (N), iteration steps for projection and matrix pseudo inverse (k_1 and k_2)

Ensure: Approximation of $\|\mathbf{S}\|_p^p$

```

1:  $res \leftarrow 0$ 
2: for  $i$  in  $1, 2, \dots, N$  do
3:   Sample  $\mathbf{g}_i \sim \mathcal{N}(0, \mathbf{I})$ 
4:    $\mathbf{v} \leftarrow \text{ApproxProject}(\mathbf{S}, \mathbf{g}_i, k_1)$  {Approx.  $P_{\mathbf{S}}[\mathbf{g}_i]$ }
5:    $\mathbf{M} \leftarrow \text{ApproxRoot}(\mathbf{S}\mathbf{S}^\top, p, k_2)$  {Approx.  $(\mathbf{S}\mathbf{S}^\top)^{\frac{p}{2}}$ }
6:    $res \leftarrow res + \mathbf{v}^\top \mathbf{M} \mathbf{g}_i$ 
7: end for
8: return  $res/N$ 
```

Remarks: 1) The theorem fully characterize the convergence behavior w.r.t. three determining factors, i.e., sample size (N), and the iteration steps (k_1 and k_2); 2) X is an unbiased estimator of the Schatten- p norm; 3) $C(k_1, k_2) \rightarrow 1$ and $E(k_1, k_2) \rightarrow 0$ converges exponentially fast as k_1 and k_2 increase; 3) Larger sample size N can effectively reduce the estimator's variance.

4 Experimental Results

In this section, we perform various experiments to demonstrate the versatility, convenience, as well as efficiency of our method. We first examine two classic LRR tasks, i.e., matrix completion and video fore-background separation. One advantage of our proposed method is to conveniently introduce LRR terms into any loss function, particularly deep neural networks. So we further exploit this property in DNN-based image denoising. Experiments about convergence and parameter sensitivity is deferred to Appendix due to space constraints. All experiments were conducted on a machine equipped with 3080Ti GPU.

Method PSNR	nuclear	T- γ^*	T-Lap	L- γ^*	L-Lap	RPCA FGSR	f-MDT Tucker	MSS	IRNN	TNN- 3DTV	DLRL
drop 20%	37.09±0.18	38.44±0.07	38.41±0.14	38.39±0.16	38.47±0.11	26.0	24.28	26.62	28.33	30.17	35.99
drop 30%	35.03±0.10	36.24±0.13	36.01±0.19	36.19±0.13	36.20±0.13	24.89	24.44	25.58	28.51	30.03	34.58
drop 40%	33.09±0.05	34.28±0.19	34.16±0.18	34.21±0.07	<u>34.22±0.15</u>	16.41	24.16	24.83	27.52	29.82	33.58
drop 50%	31.39±0.09	32.51±0.19	32.31±0.15	32.32±0.11	32.38±0.09	6.95	23.90	24.05	26.91	29.54	<u>32.43</u>
block	17.89±0.01	31.14±0.02	29.67±0.03	24.64±0.01	25.80±0.02	13.75	23.20	26.09	25.46	28.17	<u>30.82</u>
text	26.73±0.02	<u>35.07±0.06</u>	34.99±0.03	34.97±0.05	34.98±0.01	21.49	22.68	24.56	26.20	29.99	37.19
time (s)	4.35	3.83	3.88	3.83	3.86	16.83	1.73	10.70	8.82	24.85	494.46

Table 1: Comparison of matrix completion algorithms for image inpainting. Bold **terms** and underlined terms denote the best and second best results. “time” denotes the average processing time per image. See the main text for details.

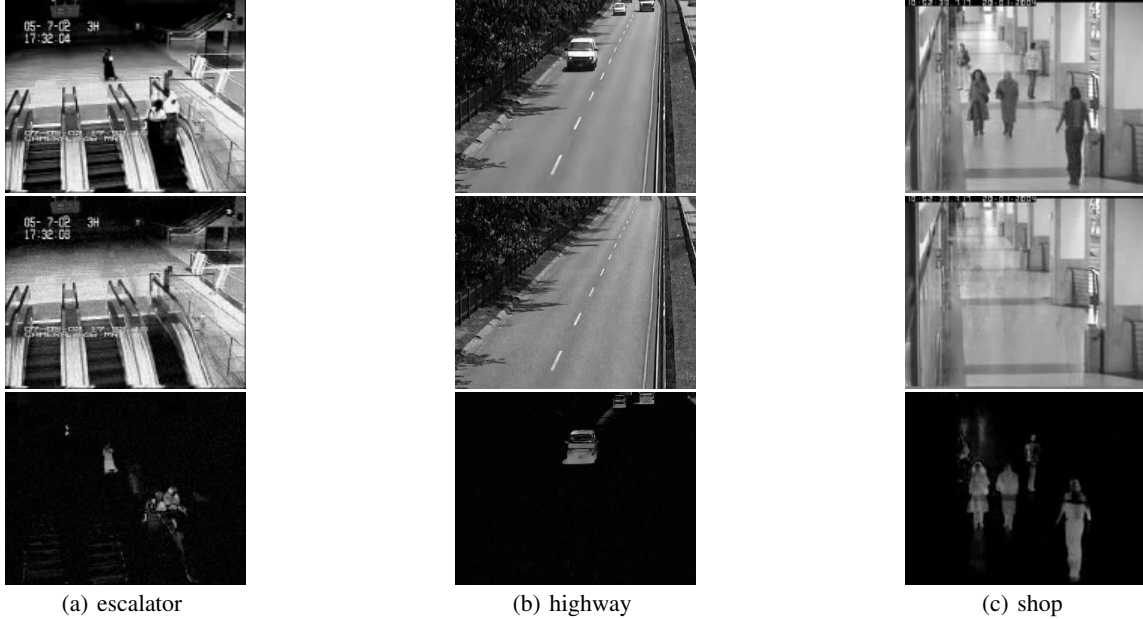


Figure 2: The results using our algorithm in fore-background separation. From top to bottom: original frames of the video, separated backgrounds, and foreground objects.

4.1 Matrix completion

Natural images can be represented as matrices that possess low-rank priors. Particularly, singular values of natural images are dominated by a few of the largest components, which allows us to model image restoration as a low-rank matrix completion problem. When dealing with colorful images, we process each color channel as an independent matrix. We evaluate the effectiveness of different methods using PSNR.

Table 1 presents the numerical result, and Figure 1 visualizes the image qualities. In Table 1, the first five columns denote our method, using different approximation and relaxation strategies. Particularly, “nuclear” denotes using the nuclear norm without other relaxation function, *i.e.*, directly apply Proposition 4. “T- γ^* ” and “T-Lap” denote using the Taylor expansion-based method for function approximation, with the γ -nuclear norm [Kang *et al.*, 2015] and Laplace [Trzasko and Manduca, 2008; Hu *et al.*, 2021] as the relaxation function (*i.e.*, h) respectively. Similarly, “L- γ^* ” and “L-Lap” denote using the Laguerre expansion-based method for func-

tion approximation. “drop XX%” means randomly removing a certain portion of pixels, while “block” and “text” use pre-defined patterns to obscure the image. The methods being compared include RPCA with FGSR [Fan *et al.*, 2019], f-MDT Tucker [Yamamoto *et al.*, 2022], MSS [Oh *et al.*, 2015], IRNN [Lu *et al.*, 2015], TNN-3DTV [Jiang *et al.*, 2018], and DLRL [Chen *et al.*, 2021].

From the results, we can see that: 1) Our method outperforms other baselines in almost all the cases; 2) By using relaxation methods, our performance can be further improved; 3) Both Taylor and Laguerre expansions are effective for function approximation, with each excelled in different scenarios; 4) By utilizing parallel computation and high-performing GPUs, our method strikes the best balance between performance and efficiency.

Further results and visualization of matrix completion can be found in the Appendix.

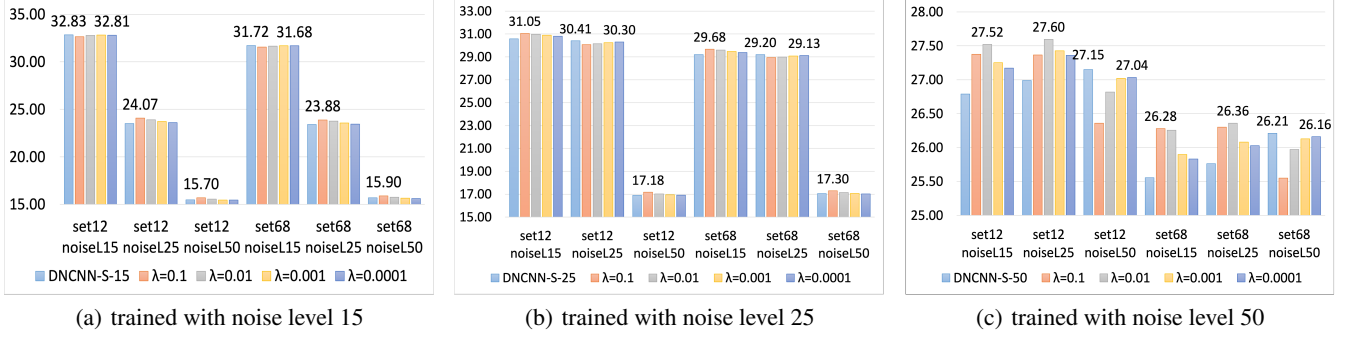


Figure 3: Results of applying low-rank regularization and the proposed differentiable approximation technique in the DnCNN denoising model, measured in PSNR.

4.2 Video fore-background separation

We now apply our method to the task of video fore-background separation. Given a video sequence $\mathbf{V} \in \mathbb{R}^{a \times b \times t}$, where a and b represent the dimensions of each frame, and t indexes the time steps. For each frame $\mathbf{f} \in \mathbb{R}^{a \times b}$, we can reshape the matrix into a vector $\mathbf{f}' \in \mathbb{R}^{ab \times 1}$ and concatenate all frames together, resulting in the final matrix $\mathbf{V}' \in \mathbb{R}^{ab \times t}$. We assume that the reshaped matrix can be decomposed as $\mathbf{V}' = \mathbf{S} + \mathbf{O}$, where \mathbf{S} is a low-rank matrix representing the background, and \mathbf{O} is a sparse matrix representing the foreground object. Thus, the problem can be solved by optimizing $\min_{\mathbf{X}} \|\mathbf{V}' - \mathbf{X}\|_1 + \lambda \mathcal{R}(\mathbf{X})$, where $\|\cdot\|_1$ denotes the L1 norm and $\mathcal{R}(\mathbf{X})$ represents the LRR term. We use the nuclear norm as the surrogate rank function.

Figure 2 illustrates the results obtained by applying our method to fore-background separation. The algorithm effectively separates the backgrounds of individual frames by utilizing the global information of the complete video sequences. Notably, our approach produces distinct boundaries for the background, even in dynamic scenarios such as the continuously moving escalator example. Unlike conventional methods that tend to blur the details of moving objects like escalator steps, our algorithm maintains clear edges and boundaries in the obscured background region.

4.3 Regularizing DNN-based denoising models

One of the major strengths of our method is its flexibility in incorporating the LRR term into any loss function, and the optimization can be accomplished with deep learning libraries. Given the impressive performance of deep neural networks, it is highly desirable to apply our approach to leverage low-rank priors. We explore this avenue in DNN-based image denoising, particularly using the denoising convolutional neural networks (DnCNNs) [Zhang *et al.*, 2017].

Following the experimental configuration of [Zhang *et al.*, 2017], we conducted our experiments using a training set of 400 images with dimensions of 180×180 . Gaussian noise levels were set at $\sigma = 15, 25, 50$. Since the low-rank structure is only applicable to the entire image, we utilized the full image as input when calculating the regularization loss. For the reconstruction loss, the settings remained unchanged, with the images divided into patches of size 40×40 . The

datasets in our experiments were the Berkeley segmentation dataset (BSD68) and the Set12 dataset, which were consistent with previous studies.

The comparison between the original network and the network augmented with LRR is depicted in Figure 3. One of the challenges faced by these denoising networks is their reliance on prior knowledge of the noise level (σ) during training. Consequently, the trained models tend to overfit to the specific noise level provided, resulting in inferior performance when confronted with different noise levels during testing. However, as observed in the figure, this issue is significantly mitigated when the LRR and our proposed differentiable approximation are applied. Although there is a slight drop in performance when the test noise level matches the training noise level, substantial gains are observed at unseen noise levels. This is most evident in Figure 3 (c), in which our method frequently improve the performance by a large margin.

5 Conclusion

In this paper, a novel differentiable approximation of the generalized LRR was proposed. The form of the regularization is quite general, covering a broad range of both convex and nonconvex relaxations. The key advantages of the proposed method include its versatility, convenience, and efficiency. By appending the differentiable LRR to a loss function in a plug-and-play fashion, the optimization can be automatically accomplished with gradient-based machine learning libraries. The operations are GPU-friendly, which facilitates parallel and efficient computation. On the theoretical side, we rigorously prove that both the bias and the variance of the rank approximation rapidly reduce with increased sample size and iteration steps. In the experimental study, the proposed method was successfully applied to a variety of tasks.

Acknowledgements

This work is supported in part by the National Natural Science Foundation of China, under Grant (62302309, 62171248), Shenzhen Science and Technology Program (JCYJ20220818101014030, JCYJ20220818101012025), the PCNL KEY project (PCL2023AS6-1), the Shenzhen Science and Technology Innovation Committee under Grant

(KJZD20230923114059020, KJZD20240903103702004), and the Department of Science and Technology of Guangdong Province under Grant (2024A0101010001).

References

- [Ben-Israel and Cohen, 1966] Adi Ben-Israel and Dan Cohen. On iterative computation of generalized inverses and associated projections. *SIAM Journal on Numerical Analysis*, 3(3):410–419, 1966.
- [Chen *et al.*, 2021] Zhaoliang Chen, Jie Yao, Guobao Xiao, and Shiping Wang. Efficient and differentiable low-rank matrix completion with back propagation. *IEEE Transactions on Multimedia*, 2021.
- [Fan *et al.*, 2019] Jicong Fan, Lijun Ding, Yudong Chen, and Madeleine Udell. Factor group-sparse regularization for efficient low-rank matrix recovery. In *Advances in Neural Information Processing Systems*, 2019.
- [Friedman, 2012] Jerome H Friedman. Fast sparse regression and classification. *International Journal of Forecasting*, 28(3):722–738, 2012.
- [Fu *et al.*, 2021] Zhiqiang Fu, Yao Zhao, Dongxia Chang, and Yiming Wang. A hierarchical weighted low-rank representation for image clustering and classification. *Pattern Recognition*, 112:107736, 2021.
- [Fu *et al.*, 2022] Zhiqiang Fu, Yao Zhao, Dongxia Chang, Yiming Wang, Jie Wen, Xingxing Zhang, and Guodong Guo. One-step low-rank representation for clustering. In *ACM International Conference on Multimedia*, pages 2220–2228, 2022.
- [Gao *et al.*, 2011] Cuixia Gao, Naiyan Wang, Qi Yu, and Zhihua Zhang. A feasible nonconvex relaxation approach to feature selection. In *AAAI Conference on Artificial Intelligence*, volume 25, pages 356–361, 2011.
- [Geman and Yang, 1995] Donald Geman and Chengda Yang. Nonlinear image recovery with half-quadratic regularization. *IEEE transactions on Image Processing*, 4(7):932–946, 1995.
- [Geng *et al.*, 2021] Zhengyang Geng, Meng-Hao Guo, Hongxu Chen, Xia Li, Ke Wei, and Zhouchen Lin. Is attention better than matrix decomposition? In *International Conference on Learning Representations*, 2021.
- [Giampouras *et al.*, 2020] Paris Giampouras, René Vidal, Athanasios Rontogiannis, and Benjamin Haeffele. A novel variational form of the Schatten- p quasi-norm. In *Advances in Neural Information Processing Systems*, pages 21453–21463, 2020.
- [Hu *et al.*, 2021] Zhanxuan Hu, Feiping Nie, Rong Wang, and Xuelong Li. Low rank regularization: A review. *Neural Networks*, 136:218–232, 2021.
- [Hu *et al.*, 2022a] Edward J Hu, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, et al. Lora: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2022.
- [Hu *et al.*, 2022b] Yexun Hu, Tai-Xiang Jiang, and Xi-Le Zhao. Degradation accordant plug-and-play for low-rank tensor completion. In *International Joint Conference on Artificial Intelligence*, pages 1843–1849, 2022.
- [Idelbayev and Carreira-Perpinán, 2020] Yerlan Idelbayev and Miguel A Carreira-Perpinán. Low-rank compression of neural nets: Learning the rank of each layer. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8049–8059, 2020.
- [Jia *et al.*, 2020] Xixi Jia, Xiangchu Feng, Weiwei Wang, and Lei Zhang. Generalized unitarily invariant gauge regularization for fast low-rank matrix recovery. *IEEE Transactions on Neural Networks and Learning Systems*, 32(4):1627–1641, 2020.
- [Jiang *et al.*, 2018] Fei Jiang, Xiao-Yang Liu, Hongtao Lu, and Ruimin Shen. Anisotropic total variation regularized low-rank tensor completion based on tensor nuclear norm for color image inpainting. In *IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 1363–1367, 2018.
- [Kang *et al.*, 2015] Zhao Kang, Chong Peng, and Qiang Cheng. Robust pca via nonconvex rank approximation. In *IEEE International Conference on Data Mining*, pages 211–220. IEEE, 2015.
- [Liu *et al.*, 2023] Ting Liu, Qian Yin, Jungang Yang, Yingqian Wang, and Wei An. Combining deep denoiser and low-rank priors for infrared small target detection. *Pattern Recognition*, 135:109184, 2023.
- [Lu *et al.*, 2015] Canyi Lu, Jinhui Tang, Shuicheng Yan, and Zhouchen Lin. Nonconvex nonsmooth low rank minimization via iteratively reweighted nuclear norm. *IEEE Transactions on Image Processing*, 25(2):829–839, 2015.
- [Mohan and Fazel, 2012] Karthik Mohan and Maryam Fazel. Iterative reweighted algorithms for matrix rank minimization. *The Journal of Machine Learning Research*, 13(1):3441–3473, 2012.
- [Oh *et al.*, 2015] Tae-Hyun Oh, Yu-Wing Tai, Jean-Charles Bazin, Hyeonwoo Kim, and In So Kweon. Partial sum minimization of singular values in robust pca: Algorithm and applications. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 38(4):744–758, 2015.
- [Ornhag *et al.*, 2020] Marcus Valtonen Ornhag, Carl Olsson, and Anders Heyden. Bilinear parameterization for differentiable rank-regularization. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 346–347, 2020.
- [Ornhag *et al.*, 2021] Marcus Valtonen Ornhag, José Pedro Iglesias, and Carl Olsson. Bilinear parameterization for non-separable singular value penalties. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3897–3906, 2021.
- [Papadopoulos and Lourakis, 2000] Théodore Papadopoulos and Manolis IA Lourakis. Estimating the jacobian of the singular value decomposition: Theory and applications. In *European Conference on Computer Vision*, pages 554–570, 2000.

- [Peng *et al.*, 2015] Chong Peng, Zhao Kang, Huiqing Li, and Qiang Cheng. Subspace clustering using log-determinant rank approximation. In *ACM SIGKDD international conference on Knowledge Discovery and Data Mining*, pages 925–934, 2015.
- [Ren *et al.*, 2022] Jiahuan Ren, Zhao Zhang, Richang Hong, Mingliang Xu, Haijun Zhang, Mingbo Zhao, and Meng Wang. Robust low-rank convolution network for image denoising. In *ACM International Conference on Multimedia*, pages 6211–6219, 2022.
- [Rennie and Srebro, 2005] Jasson DM Rennie and Nathan Srebro. Fast maximum margin matrix factorization for collaborative prediction. In *International Conference on Machine Learning*, pages 713–719, 2005.
- [Shang *et al.*, 2016] Fanhua Shang, Yuanyuan Liu, and James Cheng. Scalable algorithms for tractable Schatten quasi-norm minimization. In *AAAI Conference on Artificial Intelligence*, volume 30, 2016.
- [Shang *et al.*, 2017] Fanhua Shang, James Cheng, Yuanyuan Liu, Zhi-Quan Luo, and Zhouchen Lin. Bilinear factor matrix norm minimization for robust PCA: Algorithms and applications. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(9):2066–2080, 2017.
- [Srebro *et al.*, 2004] Nathan Srebro, Jason Rennie, and Tommi Jaakkola. Maximum-margin matrix factorization. In *Advances in Neural Information Processing Systems*, 2004.
- [Trzasko and Manduca, 2008] Joshua Trzasko and Armando Manduca. Highly undersampled magnetic resonance image reconstruction via homotopic ℓ_0 -minimization. *IEEE Transactions on Medical Imaging*, 28(1):106–121, 2008.
- [Wright and Ma, 2022] John Wright and Yi Ma. *High-dimensional data analysis with low-dimensional models: Principles, computation, and applications*. Cambridge University Press, 2022.
- [Xu *et al.*, 2017] Chen Xu, Zhouchen Lin, and Hongbin Zha. A unified convex surrogate for the Schatten- p norm. In *AAAI Conference on Artificial Intelligence*, volume 31, 2017.
- [Yamamoto *et al.*, 2022] Ryuki Yamamoto, Hidekata Hon-tani, Akira Imakura, and Tatsuya Yokota. Fast algorithm for low-rank tensor completion in delay-embedded space. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2058–2066, 2022.
- [Yang *et al.*, 2016] Jian Yang, Lei Luo, Jianjun Qian, Ying Tai, Fanlong Zhang, and Yong Xu. Nuclear norm based matrix regression with applications to face recognition with occlusion and illumination changes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(1):156–171, 2016.
- [Yao *et al.*, 2018] Quanming Yao, James T Kwok, Taifeng Wang, and Tie-Yan Liu. Large-scale low-rank matrix learning with nonconvex regularizers. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 41(11):2628–2643, 2018.
- [Zhang *et al.*, 2017] Kai Zhang, Wangmeng Zuo, Yunjin Chen, Deyu Meng, and Lei Zhang. Beyond a Gaussian denoiser: Residual learning of deep CNN for image denoising. *IEEE Transactions on Image Processing*, 26(7):3142–3155, 2017.