

Article

# On the Synergy between Nonconvex Extensions of the Tensor Nuclear Norm for Tensor Recovery

Kaito Hosono <sup>1,\*</sup> , Shunsuke Ono <sup>2</sup>  and Takamichi Miyata <sup>2</sup> <sup>1</sup> Chiba Institute of Technology, Narashino-shi 275-0016, Japan<sup>2</sup> Department of Computer Science, School of Computing, Tokyo Institute of Technology, Meguro-ku 152-8552, Japan; ono@c.titech.ac.jp (S.O.); takamichi.miyata@it-chiba.ac.jp (T.M.)

\* Correspondence: s1122276HU@s.chibakoudai.jp; Tel.: +81-47-478-0136

**Abstract:** Low-rank tensor recovery has attracted much attention among various tensor recovery approaches. A tensor rank has several definitions, unlike the matrix rank—e.g., the CP rank and the Tucker rank. Many low-rank tensor recovery methods are focused on the Tucker rank. Since the Tucker rank is nonconvex and discontinuous, many relaxations of the Tucker rank have been proposed, e.g., the sum of nuclear norm, weighted tensor nuclear norm, and weighted tensor Schatten- $p$  norm. In particular, the weighted tensor Schatten- $p$  norm has two parameters, the weight and  $p$ , and the sum of nuclear norm and weighted tensor nuclear norm are special cases of these parameters. However, there has been no detailed discussion of whether the effects of the weighting and  $p$  are synergistic. In this paper, we propose a novel low-rank tensor completion model using the weighted tensor Schatten- $p$  norm to reveal the relationships between the weight and  $p$ . To clarify whether complex methods such as the weighted tensor Schatten- $p$  norm are necessary, we compare them with a simple method using rank-constrained minimization. It was found that the simple methods did not outperform the complex methods unless the rank of the original tensor could be accurately known. If we can obtain the ideal weight,  $p = 1$  is sufficient, although it is necessary to set  $p < 1$  when using the weights obtained from observations. These results are consistent with existing reports.

**Keywords:** nuclear norm; optimization; Schatten- $p$  norm; tensor recovery; Tucker decomposition



**Citation:** Hosono, K.; Ono, S.; Miyata, T. On the Synergy between Nonconvex Extensions of the Tensor Nuclear Norm for Tensor Recovery. *Signals* **2021**, *2*, 108–121. <https://doi.org/10.3390/signals2010010>

Academic Editor: Jonghwa Kim  
Received: 30 September 2020  
Accepted: 1 February 2021  
Published: 18 February 2021

**Publisher's Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**Copyright:** © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

A tensor is a powerful tool that can describe multidimensional information and the complex relationships among elements, and it is widely used in the field of signal and image processing [1–12]. Usually, such information cannot be fully obtained through observation, and we need to complete or recover a full tensor from incomplete or degraded measurements, which are corrupted by noise, missing entries, and/or outliers. Among various tensor completion/recovery approaches, low-rank-based methods have attracted much attention because they exploit the essential structure of tensors and achieve accurate estimation.

Unlike the matrix rank, there are several different definitions of the tensor rank; well-known examples include the CANDECOMP/PARAFAC (CP) rank [13] and the Tucker rank [14]. Since determining the CP rank is NP-hard [15], many existing low-rank tensor recovery methods are focused on the Tucker rank. The Tucker rank is defined as follows: (1) an input tensor is converted to the matrices (by the unfolding operation); (2) the average rank of these matrices is calculated. The Tucker rank is very difficult to handle because of its nonconvexity and discontinuity.

To address this problem, the sum of nuclear norm, which is a convex surrogate of the Tucker rank, is proposed [1,3]. Methods based on the Tucker rank replace the rank of unfolding matrices with their nuclear norms, where the nuclear norm is known as a continuous tightest convex surrogate of the matrix rank [16].

On the other hand, the weighted nuclear norm and the Schatten- $p$  norm have been proposed as different surrogates of the matrix rank [17–20]. Both are a generalization of the nuclear norm and usually perform better than the nuclear norm for low-rank matrix recovery. Following this trend, a weighted tensor nuclear norm and a tensor Schatten- $p$  norm have also been proposed [11,12]. They are extensions of the weighted nuclear norm and the Schatten- $p$  norm for tensors, respectively, and they generally perform better for low-rank tensor recovery as well. However, for effective use, we need to select appropriate weights and parameters  $p$ .

The ideal (oracle) weights for the weighted nuclear norm are the inverses of the singular values of the original matrix. This is because the weighted nuclear norm with the oracle weights of the original matrix is identical to the rank. Generally, obtaining the singular values of the original matrix is difficult. Therefore, for practical usage, we need some methods to estimate the singular values of the original matrix to determine the weights [19]. On the other hand, the parameter  $p$  for the Schatten- $p$  norm is generally determined in a heuristic manner and, in most cases,  $p < 1$  is employed [8,12,18]. We should note that both the weighted tensor nuclear norm and the tensor Schatten- $p$  norm (with  $p < 1$ ) are in general nonconvex, as is the case with the matrix counterparts.

Now, some natural questions arise: Are the effects of the weightings for singular values and the Schatten- $p$  extension synergistic, or does one of them encompass? Is there any chance that a simple rank-constrained minimization, which is also a nonconvex optimization, can compete with these advanced and complicated methods?

In this paper, to answer the questions above, we propose a novel general constrained optimization problem combining the weighting and the Schatten- $p$  extension for tensors, and we develop an efficient algorithm to solve it. We performed exhaustive experiments, and the results showed that, if we can use the oracle weights, the combination of  $p = 1$  and the weighting is the most effective choice for all cases. We also found that the combination of  $p = 1/2$  and the weighting is effective when using the weights estimated from degraded measurements. The rank-constrained minimization problem performs well as long as we know the rank of the original tensor. If we are agnostic toward the correct rank, the performance drops sharply.

The main contributions of this paper are summarized as follows:

- We propose a general constrained optimization problem and an efficient solver for analyzing the relationship between the weightings of singular values and the Schatten- $p$  extension for tensors.
- We show that the weighting and the Schatten- $p$  extension are synergetic and that the effective value of  $p$  is dependent on how the weights are determined.
- We show that the rank constrained minimization problem is not able to outperform the advanced methods unless the true rank of the original tensor is known. The performance is sensitive to the rank values used as the constraints.

## 2. Low-Rank Tensor Completion

In what follows,  $\mathbb{N}$ ,  $\mathbb{R}$ , and  $\mathbb{R}_+$  denote the set of all nonnegative integers, all real numbers, and all nonnegative real numbers. We use capital calligraphic letters for tensors, capital bold letters for matrices, and lowercase bold letters for column vectors.

In this paper, we assume that an observation model of tensor recovery can be described as

$$\mathcal{Y} = A_{\Omega}(\mathcal{X}_{\text{org}} + \mathcal{V}), \quad (1)$$

where  $\mathcal{Y} \in \mathbb{R}^{n_1 \times \dots \times n_N}$ ,  $\mathcal{X}_{\text{org}} \in \mathbb{R}^{n_1 \times \dots \times n_N}$ , and  $\mathcal{V} \in \mathbb{R}^{n_1 \times \dots \times n_N}$  are an  $N$ th order observation tensor, an  $N$ th order low-rank original tensor, and an  $N$ th order random tensor, respectively, whose entries are independent and identically distributed Gaussian variables with zero mean and known variance  $\sigma_n^2$ .

The degradation operator is defined as

$$A_{\Omega}(\mathcal{X})_{i_1, \dots, i_N} = \begin{cases} \mathcal{X}_{i_1, \dots, i_N} & (i_1, \dots, i_N \in \Omega) \\ 0 & (\text{otherwise}) \end{cases}, \quad (2)$$

where  $\Omega$  is a set of indicators of observable entries.

If we can assume that the original tensor is low rank, it can be estimated by finding a tensor that is close to the observation tensor and also low rank. In particular, assuming that the rank of the original tensor is known and that the variance of the noise is 0, estimating the original tensor is the problem of finding a tensor whose rank is identical to the rank of the original tensor and whose known elements match the observation tensor, i.e., finding a tensor within the following set:

$$\text{Find } \mathcal{X} \in \mathbb{R}^{n_1 \times \dots \times n_N} \quad \text{s.t. } A_{\Omega}(\mathcal{X}) = A_{\Omega}(\mathcal{Y}), \text{rank}_m(\mathcal{X}) = \hat{r}_m \quad (3)$$

where  $\text{rank}_m(\mathcal{X}) = \text{rank}(\text{unfold}_m(\mathcal{X}))$  and  $m = 1, \dots, N$ . We denote rank as the matrix rank and  $\hat{r}_m$  as the matrix rank of an  $m$ th mode unfolded original tensor.

However, in general, the set containing the equations for the ranks, as in Equation (3), is hard to determine. Thus, we employ an alternative set with an inequality constraint instead of Equation (3):

$$\text{Find } \mathcal{X} \in \mathbb{R}^{n_1 \times \dots \times n_N} \quad \text{s.t. } A_{\Omega}(\mathcal{X}) = A_{\Omega}(\mathcal{Y}), \text{rank}_m(\mathcal{X}) \leq \hat{r}_m, \quad (4)$$

The sets shown in Equations (3) and (4) are used directly for the estimation of the original tensor because they generally contain multiple matrices. Additionally, if the observation process includes noise ( $\sigma_N \neq 0$ ), it may yield the empty set. Thus, we employ the L2 norm between the observation tensor, which corresponds to the negative log-likelihood of the Gaussian distribution, and the solution of the following minimization problem, including this norm as the estimated tensor:

$$\min_{\mathcal{X} \in \mathbb{R}^{n_1 \times \dots \times n_N}} \|A_{\Omega}(\mathcal{X}) - \mathcal{Y}\|_2^2 \quad \text{s.t. } \text{rank}_m(\mathcal{X}) \leq \hat{r}_m (m = 1, \dots, N), \quad (5)$$

where  $\|\cdot\|_2$  is an  $\ell_2$  norm of the tensor that is defined as the square root of the sum of the squares of each element of the tensor. Although this problem is one of nonconvex optimization, we can efficiently solve it by using the alternating direction method of multipliers (ADMM) [21], which is known as an algorithm for solving convex optimization problems and is effective in practice for solving nonconvex optimization problems [22–24]. A detail of an algorithm to solve Equation (5) explains in Appendix A.

Equations (3) and (5) include the matrix rank of each unfolded tensor, which is very difficult to handle since it is not only nonconvex but also discontinuous. Moreover, the situation in which we know the rank of each unfolded matrix of the original tensor  $\hat{r}_m$  is unrealistic.

The weighted tensor Schatten- $p$  norm (WTSPN) is proposed as a representation of a nonconvex but continuous tensor rank,

$$\|\cdot\|_{\mathbf{w}, \gamma, p}: \mathbb{R}^{n_1 \times \dots \times n_N} \rightarrow \mathbb{R}_+: \mathcal{X} \mapsto \sum_{m=1}^N \gamma_m \|\text{unfold}_m(\mathcal{X})\|_{\mathbf{w}_m, p}^p, \quad (6)$$

where  $\|\cdot\|_{\mathbf{w}, p}^p$  is a weighted Schatten- $p$  norm raised to the power  $p$  (WSPN) [20];  $\mathbf{w} = [\mathbf{w}_1^{\top}, \dots, \mathbf{w}_N^{\top}]^{\top}$  is the weight vector of the WSPNs;  $\gamma_m$  is a positive constant satisfying  $\sum_{m=1}^N \gamma_m = 1$ ;  $\gamma = [\gamma_1, \dots, \gamma_N]^{\top}$ , and  $\text{unfold}(\cdot)$  is an unfolding operator.

The WTSPN is generally a nonconvex function that is consistent with the weighted tensor nuclear norm [11] when  $p = 1$ , the tensor Schatten- $p$  norm [12] when  $\mathbf{w}$  is uniform (all elements of  $\mathbf{w}$  are the same value), and the sum of nuclear norm [1,3] when  $p = 1$  and  $\mathbf{w}$  is uniform.

The  $m$ th mode tensor unfolding operator of the  $n$ th order tensor  $\text{unfold}_m : \mathbb{R}^{n_1 \times \dots \times n_N} \rightarrow \mathbb{R}^{n_m \times I_m}$  is defined as a map from the tensor elements  $(i_1, \dots, i_N)$  to the corresponding matrix elements  $(i_m, j_m)$ , where  $I_m = \prod_{\substack{k=1 \\ k \neq m}}^N i_k$ ,

$$j_m = 1 + \sum_{\substack{k=1 \\ k \neq m}}^N (i_k - 1) \prod_{\substack{l=1 \\ l \neq m}}^{k-1} i_l. \tag{7}$$

The WSPN is described as

$$\|\cdot\|_{\mathbf{w},p}^p : \mathbb{R}^{n_v \times n_h} \rightarrow \mathbb{R}_+ : \mathbf{X} \mapsto \sum_{k=1}^{n_m} w_k \sigma_k(\mathbf{X})^p, \tag{8}$$

where  $0 < p$ ,  $n_m = \min(n_v, n_h)$ ,  $\sigma_k(\mathbf{X}) \in \mathbb{R}_+$  ( $k = 1, \dots, n_m$ ) is the  $k$ th largest singular value of  $\mathbf{X}$ , and  $\mathbf{w} = [w_1, \dots, w_{n_m}]^\top \in \mathbb{R}_+^{n_m}$  is a weight vector that satisfies  $0 \leq w_1 \leq w_2 \leq \dots \leq w_{n_m}$ . The WSPN is a generalization of the nuclear norm and the weighted nuclear norm [17,19], which are often used in low-rank matrix recovery.

As mentioned in Section 1, the proper weights of the WTSPN and the proper value of  $p$  have not been investigated in detail. Revealing them is one of the objectives of this paper.

On the other hand, when recovering the observation model of Equation (1) with the WTSPN as the regularization term and the  $\ell_2$  norm as the fidelity term, even if the noise variance does not change, the optimal hyperparameter (balancing the regularization term and the fidelity term) varies according to the parameters of the regularization term,  $\mathbf{w}$ , and  $p$ . This makes a fair comparison difficult. In addition, a parameter that is so difficult to tune is not desirable for practical use. Therefore, in the next section, we propose a method to solve this problem.

### 3. Proposed Method

To solve the above problem, we propose a method using  $\ell_2$  ball constraints and WTSPN minimization. Specifically, we formulate the following minimization problem:

$$\min_{\mathcal{X}} \|\mathcal{X}\|_{\mathbf{w},\gamma,p} \quad \text{s.t. } A_\Omega(\mathcal{X}) \in B(\mathcal{Y}, \sigma_n \sqrt{|\Omega|}), \tag{9}$$

where  $B(\mathcal{Y}, r)$  is an  $\ell_2$  ball, and the  $\ell_2$  ball with center  $\mathcal{Y} \in \mathbb{R}^{n_1 \times \dots \times n_N}$  and radius  $r \in \mathbb{R}$  is defined as

$$B(\mathcal{Y}, r) := \{\mathcal{X} \in \mathbb{R}^{n_1 \times \dots \times n_N} \mid \|\mathcal{X} - \mathcal{Y}\|_2 \leq r\}. \tag{10}$$

By using the ball constraint, it is possible to determine the appropriate parameters based only on the variance of the noise [25–27], which is convenient when comparing various regularization parameters, as in this paper. Additionally,  $|\Omega|$  is the number of elements of the set  $\Omega$ .

Since we assume that the standard deviation of noise  $\sigma_n$  is known, we can expect the realization of the noise  $\mathcal{Y}$  added to the original tensor to exist inside the hypersphere determined by the standard deviation. The constraints of Equation (9) are in accordance with this fact. This method allows us to “fairly compare the performance of different regularization terms (if the variance of the noise is known)”.

In general, Equation (9) is a nonconvex optimization problem, which makes it difficult to find a globally optimal solution. As mentioned in Section 2, ADMM exhibits empirical performance on nonconvex optimization problems. Therefore, we propose solving Equation (9) using ADMM. The proposed algorithm is shown in Algorithm 1.

The objective function in line 5 of Algorithm 1 is

$$\underset{\mathbf{X}}{\text{argmin}} \|\mathbf{X}\|_{\lambda\gamma_m \mathbf{w}_m, p}^p + \frac{1}{2} \|\text{unfold}_m(\mathcal{X}^{(k+1)}) + \mathbf{Z}_{1,m}^{(k)} - \mathbf{X}\|_F^2 \tag{11}$$

which is nonconvex, although one of the solutions can be written as [8,18,20]:

$$\mathbf{U}\mathbf{S}_{\lambda\gamma_m\mathbf{w}_m,p}(\Sigma)\mathbf{V}^\top, \tag{12}$$

where  $\mathbf{U}\Sigma\mathbf{V}^\top = \text{unfold}_m(\mathcal{X}^{(k+1)}) + \mathbf{Z}_{1,m}^{(k)}$  is a singular value decomposition (SVD) and  $\mathbf{S}_{\mathbf{w},p}(\cdot)$  is a weighted thresholding operator.

---

**Algorithm 1** Proposed algorithm

---

**Input:**  $\mathcal{Y}, \sigma_n, \gamma = [\gamma_1, \dots, \gamma_N], \mathbf{w} = [\mathbf{w}_1^\top, \dots, \mathbf{w}_N^\top]^\top, p, \lambda$

1: Initialize  $\mathbf{Y}_{1,m}^{(0)} = \text{unfold}_m(\mathcal{Y}), \mathcal{Y}_2^{(0)} = \mathcal{Y}, \mathbf{Z}_{1,m}^{(0)} = 0, \mathbf{Y}_{1,m}^{(0)}, \mathcal{Z}_2^{(0)} = 0$

2: **while** A stopping criterion is not satisfied **do**

3:  $\mathcal{X}^{(k+1)} = \underset{\mathcal{X}}{\text{argmin}} \frac{1}{2} \sum_{m=1}^N \|\mathbf{Y}_{1,m}^{(k)} - \text{unfold}_m(\mathcal{X}) - \mathbf{Z}_{1,m}^{(k)}\|_2^2 + \lambda \|\mathcal{Y}_2^{(k)} - \mathbf{A}_\Omega(\mathcal{X}) - \mathcal{Z}_2^{(k)}\|_2^2$

4: **for**  $m = 1$  to  $N$  **do**

5:  $\mathbf{Y}_{1,m}^{(k+1)} = \underset{\mathbf{X}}{\text{argmin}} \|\mathbf{X}\|_{\lambda\gamma_m\mathbf{w}_m,p}^p + \frac{1}{2} \|\text{unfold}_m(\mathcal{X}^{(k+1)}) + \mathbf{Z}_{1,m}^{(k)} - \mathbf{X}\|_F^2$

6:  $\mathbf{Z}_{1,m}^{(k+1)} = \mathbf{Z}_{1,m}^{(k)} + \text{unfold}_m(\mathcal{X}^{(k+1)}) - \mathbf{Y}_{1,m}^{(k+1)}$

7: **end for**

8:  $\mathcal{Y}_2^{(k+1)} = \text{proj}_{\mathbf{B}(\mathcal{Y}, \sigma_n \sqrt{|\omega|})}(\mathbf{A}_\Omega(\mathcal{X}^{(k+1)} + \mathcal{Z}_2^{(k)})) + \mathbf{A}_\Omega(\mathcal{X}^{(k+1)} + \mathcal{Z}_2^{(k)})$

9:  $\mathcal{Z}_2^{(k+1)} = \mathcal{Z}_2^{(k)} + \mathbf{A}_\Omega(\mathcal{X}^{(k+1)}) - \mathcal{Y}_2^{(k+1)}$

10:  $\lambda = 0.99\lambda$

11:  $k = k + 1$

12: **end while**

**Output:**  $\mathcal{X}^{(k)}$

---

Each element of the weighted thresholding operator for a rectangular diagonal matrix  $\mathbf{Y}$  ( $\mathbf{S}_{\mathbf{w},p}(\mathbf{Y})_{i,j}$ ) is defined as a solution to the following minimization problem:

$$\underset{\mathbf{x}}{\text{argmin}} \frac{1}{2} (\mathbf{Y}_{i,j} - \mathbf{x})^2 + \mathbf{w}_i |\mathbf{x}|^p. \tag{13}$$

The solution of Equation (13) is a soft thresholding  $\max(\mathbf{Y}_{i,j} - \mathbf{w}_i, 0)$  when  $p = 1$  and the closed-form thresholding proposed in [28] when  $p = \{1/2, 2/3\}$ .

The first term in line 8 of Algorithm 1 is

$$\text{proj}_{\mathbf{B}(\mathcal{Y}, \sigma_n \sqrt{|\Omega|})}(\mathbf{A}_\Omega(\mathcal{X}^{(k+1)} + \mathcal{Z}_2^{(k)})) \tag{14}$$

which is a metric projection of the set  $\mathbf{B}(\mathcal{Y}, \sigma_n \sqrt{|\Omega|})$ . The metric projection is defined as

$$\text{proj}_S: \mathbb{R}^N \rightarrow \mathbb{R}^N: \mathbf{x} \mapsto \underset{\mathbf{y} \in S}{\text{argmin}} \frac{1}{2} \|\mathbf{x} - \mathbf{y}\|_2^2. \tag{15}$$

Equation (14) has a closed-form solution,

$$\mathcal{Y} - \min \left( \frac{\sigma_n \sqrt{|\Omega|}}{\|\mathcal{Y} - A_\Omega(\mathcal{X}^{(k+1)} + \mathcal{Z}_2^{(k)})\|_2}, 1 \right) (\mathcal{Y} - A_\Omega(\mathcal{X}^{(k+1)} + \mathcal{Z}_2^{(k)})). \quad (16)$$

The set of  $\bar{\Omega}$  in the second term of line 8,

$$A_{\bar{\Omega}}(\mathcal{X}^{(k+1)} + \mathcal{Z}_2^{(k)}) \quad (17)$$

is the complement of the set  $\Omega$ , which is a set of indicators of missing entries.  $A_{\bar{\Omega}}$  is defined as

$$A_{\bar{\Omega}}(\mathcal{X})_{i_1, \dots, i_N} = \begin{cases} \mathcal{X}_{i_1, \dots, i_N} & (i_1, \dots, i_N \in \bar{\Omega}) \\ 0 & (\text{otherwise}) \end{cases}. \quad (18)$$

#### 4. Experimental Comparison

##### 4.1. Setting

In Section 1, we posed two questions:

- Are the effects of the weighting and  $p$ -squared on singular values synergistic? Or does one encompass the other?
- Is simple rank-constrained minimization insufficient?

To answer these questions, we performed some experiments using an artificial tensor. Each element of an  $N$ th order artificial tensor  $\mathcal{X} \in \mathbb{R}^{n_1 \times \dots \times n_N}$  is generated by using the Tucker model:

$$\mathcal{X}(i_1, \dots, i_N) = \sum_{1 \leq j_1 \leq r_1, \dots, 1 \leq j_N \leq r_N} \mathcal{S}(j_1, \dots, j_N) \prod_k \mathbf{U}_k(i_k, j_k), \quad (19)$$

where  $[r_1, \dots, r_N]$ ,  $\mathcal{S} \in \mathbb{R}^{r_1 \times \dots \times r_N}$  and  $\mathbf{U}_k \in \mathbb{R}^{n_k \times r_k}$  are the matrix ranks of  $\text{unfold}_m(\mathcal{X})$  ( $m = 1, \dots, N$ ), the core tensor, and the factor tensor. Each element of  $\mathcal{S}$  and  $\mathbf{U}_k$  is generated uniformly over the intervals  $[0, 1]$  and  $[-0.5, 0.5]$ , respectively. Finally, we normalized the difference between the maximum and minimum elements of  $\mathcal{X}$  to 1.

As mentioned in Section 1, the widely used weights used as the ideal weights for the singular values are the inverses of the singular values of the unfolded original tensor  $\mathcal{X}_{\text{org}}$ . Throughout this paper, we define our *ideal weights* as

$$(\mathbf{w}_{\text{Id}}(\alpha))_{i,j} = R \frac{\sigma_i(\text{unfold}_j(\mathcal{X}_{\text{org}}))^{-\alpha}}{\sum_k \sigma_k(\text{unfold}_j(\mathcal{X}_{\text{org}}))^{-\alpha}}, \quad (20)$$

where  $R$  is the smaller of the row and column dimensions of  $\text{unfold}_j(\mathcal{X}_{\text{org}})$ . Since the ideal weights are not always optimal in terms of the recovery performance, we introduce a parameter  $\alpha$  to bring additional flexibility to the setting of the weights. However, the true singular values are not available in practical applications.

A method that does not require the true singular values is to use the singular values obtained from observations for estimating the ideal weights. One of these methods is as follows:

$$(\mathbf{w}_{\text{Obs}}(\alpha))_{i,j} = R \frac{\sigma_i(\text{unfold}_j(\tilde{\mathcal{Y}}))^{-\alpha}}{\sum_k \sigma_k(\text{unfold}_j(\tilde{\mathcal{Y}}))^{-\alpha}} \quad (21)$$

where  $\tilde{\mathcal{Y}}$  is a tensor the missing elements of the observation tensor  $\mathcal{Y}$  filled in using the average of the observed entries of the observation tensor  $\mathcal{Y}$ . We refer to these weights as *observation weights*.

The tensor Schatten- $p$  norm with no weights is a special case of the WTSPN. Therefore, we can use the WTSPN with the following special weights as the tensor Schatten- $p$  norm:

$$(\mathbf{w}_{\text{Uni}})_{i,j} = 1. \quad (22)$$

In the following experiments, we use three types of weights (the ideal weights  $\mathbf{w}_{\text{Id}}$ , observation weights  $\mathbf{w}_{\text{Obs}}$  and uniform weights  $\mathbf{w}_{\text{Uni}}$ ) to reveal the relationship between the weighting and Schatten- $p$  extension on the performance of WTSPN. The weight determination parameter  $\alpha$  varies in increments of 0.25 in the range  $[1, 4]$ . The parameters of Schatten- $p$  are chosen from  $p = \{1/2, 2/3, 1\}$ , and each element of  $\gamma$  is set to  $1/N$ , where  $N$  is the order of the target tensor. In all cases, the parameter of ADMM is set to  $\lambda = 100$ .

We compare the performance of Algorithm 1 using  $\mathbf{w}_{\text{Id}}$ ,  $\mathbf{w}_{\text{Obs}}$ , and  $\mathbf{w}_{\text{Uni}}$  as well as rank-constrained minimization. The following error is used to evaluate the performance of each method:

$$\text{error}(\tilde{\mathcal{X}}, \mathcal{Y}) = \frac{1}{\prod_{m=1}^N I_m} \|\tilde{\mathcal{X}} - \mathcal{Y}\|_2, \quad (23)$$

where  $\tilde{\mathcal{X}}$  is the estimated tensor obtained by each method.

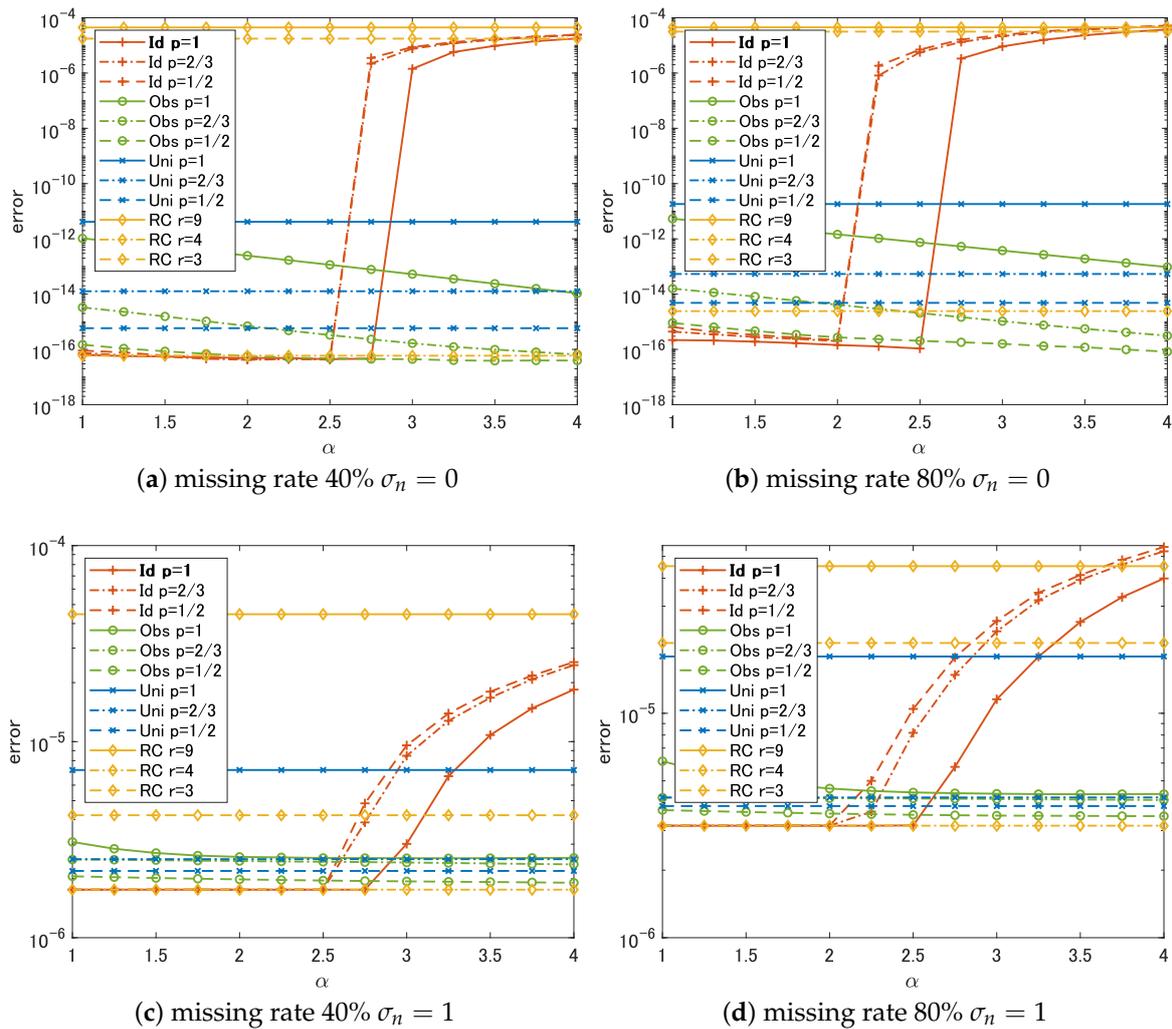
#### 4.2. Results and Discussion

We performed recovery from observed tensors with missing rate of 0.4 and 0.8 and standard deviations of noise  $\sigma_n$  of 0 and 1, and the results of the combinations of these parameters are shown in (a) to (d) in Figures 1–4. The horizontal axis of each graph is the parameter  $\alpha$  used in Equations (20) and (21) for determining the weights, and the vertical axis is the performance of each method defined by Equation (23). The red, green, blue, and yellow lines show the results of Algorithm 1 with  $\mathbf{w}_{\text{Id}}$  (Id in the legends of the graphs), with  $\mathbf{w}_{\text{Obs}}$  (Obs), with  $\mathbf{w}_{\text{Uni}}$  (Uni), and with the rank-constrained minimization shown in Equation (5) (RC). In the case of Id, Obs, and Uni, the results corresponding to the different values of  $p$  are shown with different line types. Similarly, in the case of RC, the results corresponding to the different target ranks  $r$  are shown with different line types.

Figure 1 is the result when the size of the original tensor is  $40 \times 40 \times 40$  and the rank is  $[4, 4, 4]$ . From the graphs (a)–(d) in Figure 1, one can see that

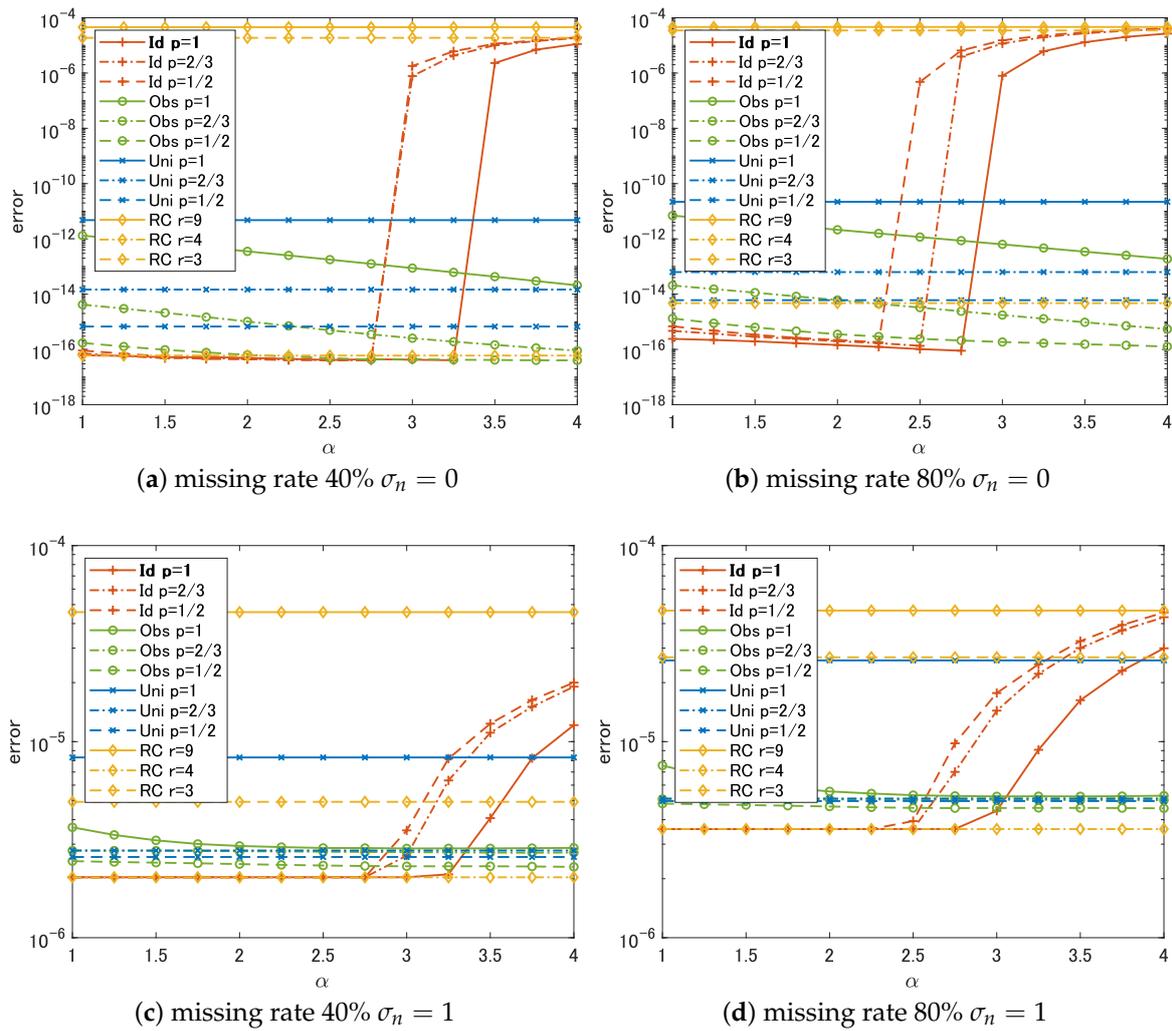
- In the case of Id, if we choose  $\alpha < 2$ , the choice of  $p$  does not have much effect on the performance. The slowest degradation of performance due to the change in  $\alpha$  is obtained at  $p = 1$ .
- In the case of Obs,  $p = 1/2$  shows the best result across (a)–(d). These results are consistent with the results in previous studies [8,12,18,20].
- Regardless of  $p$ , the performance of Id and Obs is the same or better than that of Uni in all cases.
- In all cases, RC shows the worst performance unless we can choose the correct rank  $r$ .

Figure 2 shows the results when we only change the tensor rank to  $[5, 5, 5]$ . Note that the size of the tensor is still  $40 \times 40 \times 40$ . The results show a similar trend as in the case of Figure 1. From the results in Figures 1 and 2, for the third-order tensor, we can conclude that the effect of the choice of the weights,  $p$ , and the algorithms on performance is rank-independent.



**Figure 1.** Experimental results for the original tensor with the order 3 and the rank 4. (a–d) are the results of varying the missing rate and the standard deviation of the noise  $\sigma_n$  during the observation process. The horizontal axis of the graph is the parameter  $\alpha$  used for determining the weights and the vertical axis is the error between the estimated tensor and the original tensor calculated by each method. Red, green, and blue indicate the results for the proposed method (Algorithm 1) with different types of weight vectors— $\mathbf{w}_{Id}$ ,  $\mathbf{w}_{Obs}$ , and  $\mathbf{w}_{Uni}$ , respectively—and yellow is the result of the rank-constrained minimization. The line type corresponds to the value of the parameter  $p$  or  $r$ .

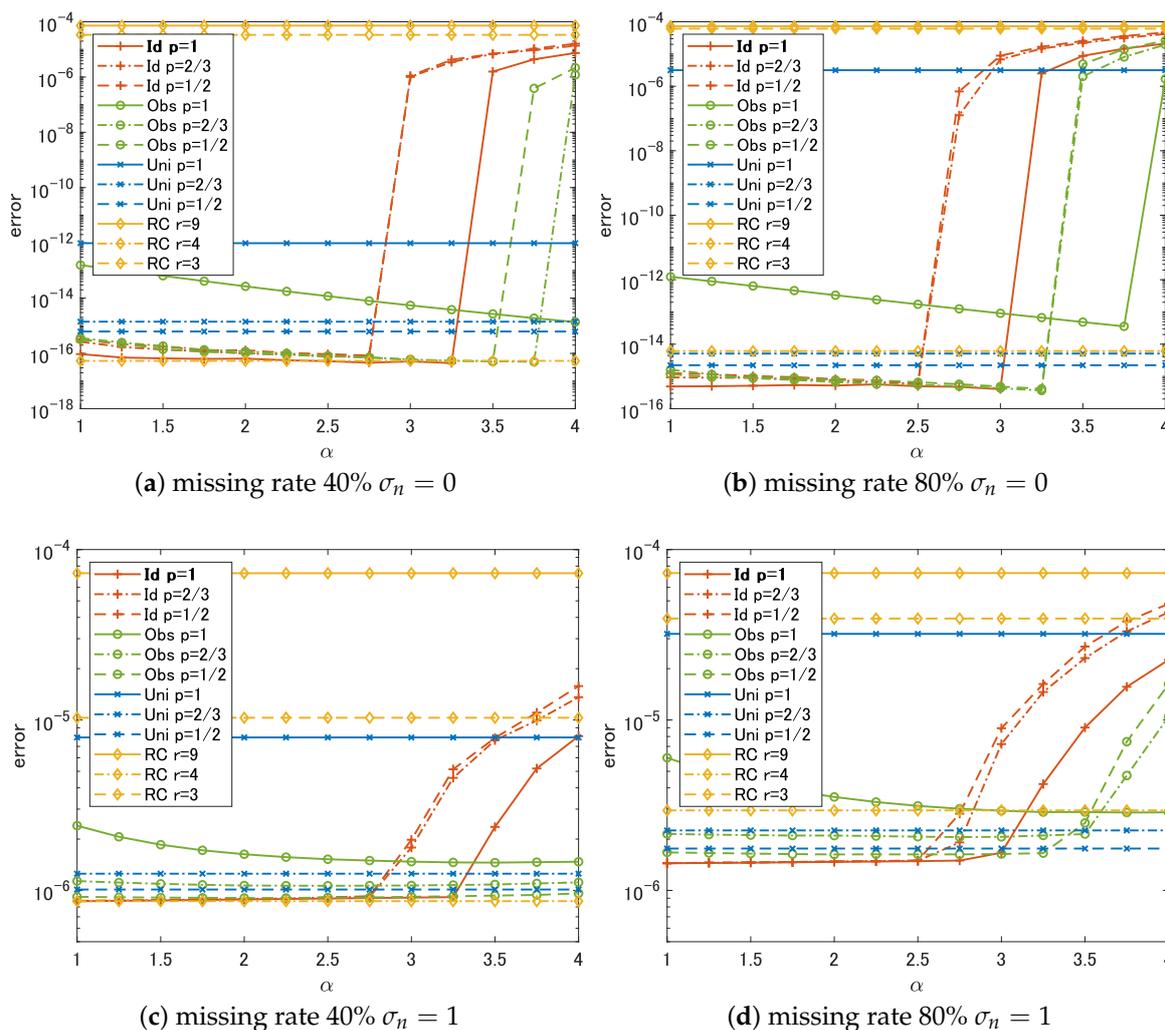
To reveal the impact of the changes in the order of the tensor on the common trend in Figures 1 and 2, we performed experiments on the 4th-order tensor. The results are shown in Figures 3 and 4. In Figures 3 and 4, the sizes of the original tensors are both  $16 \times 16 \times 16 \times 16$ , and the ranks are  $[2, 2, 2, 2]$  and  $[3, 3, 3, 3]$ , respectively. In the case of the 4th-order tensor, there was no change in the common trend of each graph when the rank was varied. The same trend is observed in comparison with Figures 1 and 2. From these observations, we can say that the relationship between the weighting and Schatten- $p$  extension and the performance gap between the proposed algorithm and the rank-constrained minimization that we revealed is a law and is independent of the tensor rank and order.



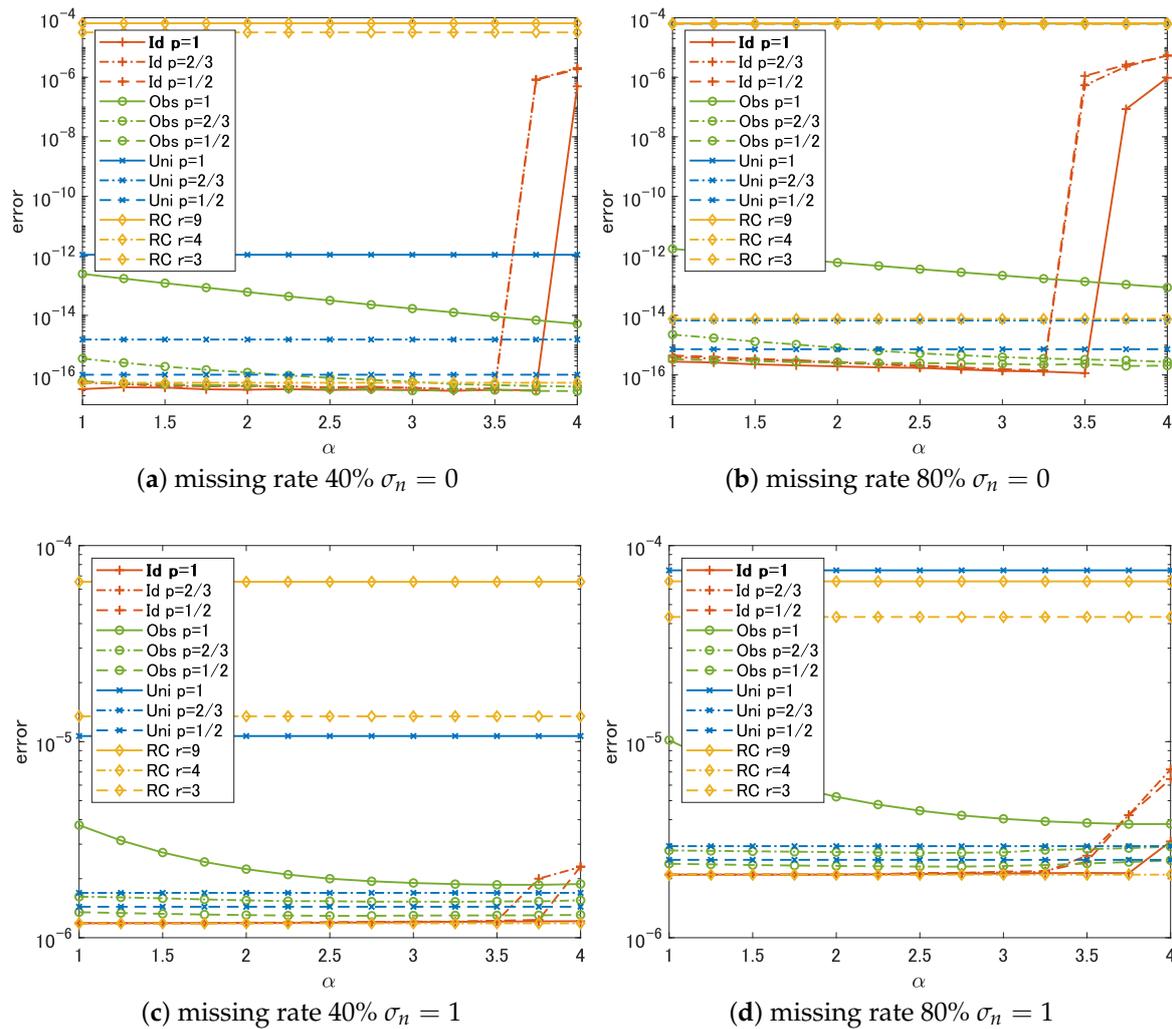
**Figure 2.** Experimental results for the original tensor with the order 3 and the rank 5. The results of different missing rates and the standard deviations are shown in (a–d) and the meaning of the axes and of the colors and types of lines are the same as in Figure 1. Although the rank of the original tensor is different, the relative performance of all methods shows a similar trend in Figure 1. This supports the fact that our conclusions in Section 4.2 are independent of the rank of the original tensor.

From these results, we can conclude that

- It is sufficient to use  $p = 1$  if weights that are close to the ideal weights can be estimated in some way.
- It is better to set a small  $p$ -value if the weights estimated from the degraded singular values are not reliable.
- Simple methods using rank constraints are very sensitive to the choice of ranks used for the constraints and cannot outperform complex methods like the proposed algorithm unless one can correctly estimate the original ranks.



**Figure 3.** Experimental results for the original tensor with the order 4 and the rank 2. The results of different missing rates and the standard deviations are shown in (a–d) and the meaning of the axes and of the colors and types of lines are the same as in Figure 1. Although the rank and the order of the original tensor are different, the relative performance of all methods shows a similar trend to Figure 1. This supports the fact that our conclusions in Section 4.2 are independent of the rank and the order of the original tensor.



**Figure 4.** Experimental results for the original tensor with the order 4 and the rank 3. The results of different missing rates and the standard deviations are shown in (a–d) and the meaning of the axes and of the colors and types of lines are the same as in Figure 1. Although the rank and the order of the original tensor are different, the relative performance of all methods shows a similar trend to Figure 1. This supports the fact that our conclusions in Section 4.2 are independent of the rank and the order of the original tensor.

### 5. Conclusions

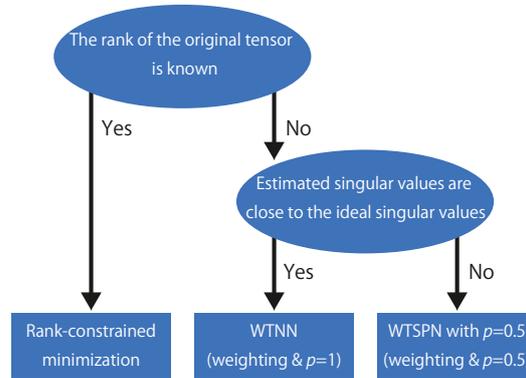
In this paper, to reveal the relationships between the weighting and Schatten- $p$  extension, we propose a general tensor recovery model that combines them and propose an algorithm to solve it. From the experiments with artificial data using the proposed algorithms, the effect of the recovery performance in the presence or absence of the weighting and the Schatten- $p$  extension for various situations is determined.

Consequently, the simple rank-constrained minimization method cannot outperform complex methods such as the proposed algorithm unless the rank  $r$  used in the constraint is chosen properly. The relationships between the weighting and Schatten- $p$  extension in WTSPNs vary with the degree to which we can estimate the ideal weights. The Schatten- $p$  extension does not affect the performance if the ideal weight is available. On the other hand, the effect of the Schatten- $p$  and the weighting on singular values is synergistic if we need to determine the weights from heavily degraded observations.

Our conclusion is summarized in the flowchart in Figure 5, where “weighting” indicates that using the weights  $\mathbf{w}$ , which are determined based on the estimates of the singular values of the unfolded original tensor. This flowchart implies that, if we can access limited

information about the rank (or the singular values) of the original tensor, we need to use complex methods to obtain good results.

The proposed tensor restoration model was based on the observation model of Equation (1). Therefore, the proposed guideline is not applicable to problems that can not be represented in the current observation model, e.g., a multiplicative noise. We will solve this problem and validate the relationships between the weighting and the Schatten- $p$  extension in various situations in our future work.



**Figure 5.** The flowchart for determining a method and parameter for the low-rank tensor recovery problem.

**Author Contributions:** Conceptualization, K.H., S.O., and T.M.; methodology, K.H., S.O., and T.M.; software, K.H.; validation, K.H., S.O., and T.M.; formal analysis, K.H. and S.O.; investigation, K.H. and S.O.; resources, T.M.; data curation, K.H.; writing—original draft preparation, K.H.; writing—review and editing, S.O. and T.M.; visualization, K.H.; supervision, T.M.; project administration, S.O. and T.M.; funding acquisition, T.M. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research was funded by the Japan Society for the Promotion of Science KAKENHI Grant No. JP19K04377.

**Conflicts of Interest:** The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript, or in the decision to publish the results.

**Appendix A. Algorithm for Rank-Constrained Minimization**

In Section 2, we mentioned that we can solve Equation (5) efficiently by using ADMM, although we did not show a specific algorithm. The algorithm for solving Equation (5) is shown in Algorithm A1.

The objective function in line 5 of Algorithm A1 is

$$\text{proj}_{\{\mathbf{X}|\text{rank}(\mathbf{X})\leq\hat{r}_m\}}(\text{unfold}_m(\mathcal{X}^{(k+1)}) + \mathbf{Z}_{1,m}^{(k)}), \tag{A1}$$

which is a nonconvex function because the set  $\{\mathbf{X}|\text{rank}(\mathbf{X}) \leq \hat{r}_m\}$  is a nonconvex set. However, one of the solutions of Equation (A1) can be obtained as

$$\mathbf{U}\mathbf{T}_{\hat{r}_m}(\Sigma)\mathbf{V}^\top, \tag{A2}$$

where  $\mathbf{U}\Sigma\mathbf{V}^\top = \text{unfold}_m(\mathcal{X}^{(k+1)}) + \mathbf{Z}_{1,m}^{(k)}$  is an SVD and  $\mathbf{T}_{\hat{r}_m}(\cdot)$  is a truncation operator. The truncation operator for a rectangular diagonal matrix  $\mathbf{Y}$   $\mathbf{T}_r(\mathbf{Y})$  is defined as

$$\mathbf{T}_r(\mathbf{Y})_{i,i} = \begin{cases} \mathbf{Y}_{i,i} & (i \geq r) \\ 0 & (\text{otherwise}) \end{cases} . \tag{A3}$$

**Algorithm A1** Algorithm for rank-constrained minimization**Input:**  $\mathcal{Y}, \lambda, r_1, \dots, r_N$ 1: Initialize  $\mathbf{Y}_{1,m}^{(0)} = \text{unfold}_m(\mathcal{Y}), \mathcal{Y}_2^{(0)} = \mathcal{Y}, \mathbf{Z}_{1,m}^{(0)} = 0, \mathbf{Y}_{1,m}^{(0)}, \mathcal{Z}_2^{(0)} = 0$ 2: **while** A stopping criterion is not satisfied **do**3:  $\mathcal{X}^{(k+1)} = \underset{\mathcal{X}}{\text{argmin}} \frac{1}{2} \sum_{m=1}^N \|\mathbf{Y}_{1,m}^{(k)} - \text{unfold}_m(\mathcal{X}) - \mathbf{Z}_{1,m}^{(k)}\|_2^2 + \lambda \|\mathcal{Y}_2^{(k)} - \mathbf{A}_\Omega(\mathcal{X}) - \mathcal{Z}_2^{(k)}\|_2^2$ 4: **for**  $m = 1$  to  $N$  **do**5:  $\mathbf{Y}_{1,m}^{(k+1)} = \text{proj}_{\{\mathcal{X} | \text{rank}(\mathcal{X}) \leq r_m\}}(\text{unfold}_m(\mathcal{X}^{(k+1)}) + \mathbf{Z}_{1,m}^{(k)})$ 6:  $\mathbf{Z}_{1,m}^{(k+1)} = \mathbf{Z}_{1,m}^{(k)} + \text{unfold}_m(\mathcal{X}^{(k+1)}) - \mathbf{Y}_{1,m}^{(k+1)}$ 7: **end for**8:  $\mathcal{Y}_2^{(k+1)} = \underset{\mathcal{X}}{\text{argmin}} \lambda \|\mathbf{A}_\Omega(\mathcal{X}) - \mathcal{Y}\|_2^2 + \frac{1}{2} \|\mathcal{X}^{(k+1)} + \mathcal{Z}_2^{(k)} - \mathcal{X}\|_F^2$ 9:  $\mathcal{Z}_2^{(k+1)} = \mathcal{Z}_2^{(k)} + \mathbf{A}_\Omega(\mathcal{X}^{(k+1)}) - \mathcal{Y}_2^{(k+1)}$ 10:  $\lambda = 0.99\lambda$ 11:  $k = k + 1$ 12: **end while****Output:**  $\mathcal{X}^{(k)}$ **References**

- Gandy, S.; Recht, B.; Yamada, I. Tensor completion and low-n-rank tensor recovery via convex optimization. *Inverse Probl.* **2011**, *27*, 025010. [[CrossRef](#)]
- Roughan, M.; Zhang, Y.; Willinger, W.; Qiu, L. Spatio-temporal compressive sensing and internet traffic matrices (extended version). *IEEE/ACM Trans. Netw.* **2012**, *20*, 662–676. [[CrossRef](#)]
- Liu, J.; Musialski, P.; Wonka, P.; Ye, J. Tensor completion for estimating missing values in visual data. *IEEE Trans. Pattern Anal. Mach. Intell.* **2013**, *35*, 208–220. [[CrossRef](#)]
- Liu, X.Y.; Aeron, S.; Aggarwal, V.; Wang, X.; Wu, M.Y. Tensor completion via adaptive sampling of tensor fibers: Application to efficient indoor RF fingerprinting. In Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Shanghai, China, 20–25 March 2016; pp. 2529–2533. [[CrossRef](#)]
- Ng, M.K.P.; Yuan, Q.; Yan, L.; Sun, J. An adaptive weighted tensor completion method for the recovery of remote sensing images with missing data. *IEEE Trans. Geosci. Remote Sens.* **2017**, *55*, 3367–3381. [[CrossRef](#)]
- Sun, W.; Chen, Y.; Thus, H.C. Tensor completion using kronecker rank-1 tensor train with application to visual data inpainting. *IEEE Access* **2018**, *6*, 47804–47814. [[CrossRef](#)]
- Xie, Q.; Zhao, Q.; Meng, D.; Xu, Z. Kronecker-basis-representation based tensor sparsity and its applications to tensor recovery. *IEEE Trans. Pattern Anal. Mach. Intell.* **2018**, *40*, 1888–1902. [[CrossRef](#)]
- Zhang, X.; Zheng, J.; Yan, Y.; Zhao, L.; Jiang, R. Joint weighted tensor Schatten  $p$ -norm and tensor  $l_p$ -norm minimization for image denoising. *IEEE Access* **2019**, *7*, 20273–20280. [[CrossRef](#)]
- Yokota, T.; Hontani, H. Simultaneous tensor completion and denoising by noise inequality constrained convex optimization. *IEEE Access* **2019**, *7*, 15669–15682. [[CrossRef](#)]
- Wang, A.; Song, X.; Wu, X.; Lai, Z.; Jin, Z. Robust low-tubal-rank tensor completion. In Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Brighton, UK, 12–17 May 2019; pp. 3432–3436. [[CrossRef](#)]
- Hosono, K.; Ono, S.; Miyata, T. Weighted tensor nuclear norm minimization for color image restoration. *IEEE Access* **2019**, *7*, 88768–88776. [[CrossRef](#)]
- Gao, S.; Fan, Q. Robust Schatten- $p$  norm based approach for tensor completion. *J. Sci. Comput.* **2020**, *82*, 11. [[CrossRef](#)]
- Harshman, R.A. Foundations of the PARAFAC procedure: Models and conditions for an “explanatory” multi-modal factor analysis. *UCLA Work. Phon.* **1970**, *16*, 1–84.
- Tucker, L.R. Some mathematical notes on three-mode factor analysis. *Psychometrika* **1966**, *31*, 279–311. [[CrossRef](#)] [[PubMed](#)]

15. Hastad, J. Tensor rank is NP-complete. *J. Algorithms* **1990**, *11*, 644–654. [[CrossRef](#)]
16. Fazel, M. Matrix Rank Minimization with Applications. Ph.D. Thesis, Stanford University, Stanford, CA, USA, 2002.
17. Chen, K.; Dong, H.; Chan, K.S. Reduced rank regression via adaptive nuclear norm penalization. *Biometrika* **2013**, *100*, 901–920. [[CrossRef](#)]
18. Xie, Y.; Gu, S.; Liu, Y.; Zuo, W.; Zhang, W.; Zhang, L. Weighted Schatten  $p$ -norm minimization for image denoising and background subtraction. *IEEE Trans. Image Process.* **2016**, *25*, 4842–4857. [[CrossRef](#)]
19. Gu, S.; Xie, Q.; Meng, D.; Zuo, W.; Feng, X.; Zhang, L. Weighted nuclear norm minimization and its applications to low level vision. *Int. J. Comput. Vis.* **2017**, *121*, 183–208. [[CrossRef](#)]
20. Zha, Z.; Zhang, X.; Wu, Y.; Wang, Q.; Liu, X.; Tang, L.; Yuan, X. Non-convex weighted  $\ell_p$  nuclear norm based ADMM framework for image restoration. *Neurocomputing* **2018**, *311*, 209–224. [[CrossRef](#)]
21. Gabay, D.; Mercier, B. A dual algorithm for the solution of nonlinear variational problems via finite element approximation. *Comput. Math. Appl.* **1976**, *2*, 17–40. [[CrossRef](#)]
22. Chartrand, R.; Wohlberg, B. A nonconvex ADMM algorithm for group sparsity with sparse groups. In Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Vancouver, BC, Canada, 26–31 May 2013; pp. 6009–6013. [[CrossRef](#)]
23. Sun, D.L.; Févotte, C. Alternating direction method of multipliers for non-negative matrix factorization with the beta-divergence. In Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Florence, Italy, 4–9 May, 2014; pp. 6201–6205. [[CrossRef](#)]
24. Ono, S. L0 gradient projection. *IEEE Trans. Image Process.* **2017**, *26*, 1554–1564. [[CrossRef](#)]
25. Afonso, M.V.; Bioucas-Dias, J.M.; Figueiredo, M.A. An augmented lagrangian approach to the constrained optimization formulation of imaging inverse problems. *IEEE Trans. Image Process.* **2011**, *20*, 681–695. [[CrossRef](#)] [[PubMed](#)]
26. Chierchia, G.; Pustelnik, N.; Pesquet, J.C.; Pesquet-Popescu, B. Epigraphical projection and proximal tools for solving constrained convex optimization problems. *Signal Image Video Process.* **2015**, *9*, 1737–1749. [[CrossRef](#)]
27. Ono, S.; Yamada, I. Signal recovery with certain involved convex data-fidelity constraints. *IEEE Trans. Signal Process.* **2015**, *63*, 6149–6163. [[CrossRef](#)]
28. Cao, W.; Sun, J.; Xu, Z. Fast image deconvolution using closed-form thresholding formulas of  $L_q$  ( $q = 1/2, 2/3$ ) regularization. *J. Vis. Commun. Image Represent.* **2013**, *24*, 31–41. [[CrossRef](#)]