symmetry

MDPI

Article

Adaptive Optics Retinal Image Restoration Using Total Variation with Overlapping Group Sparsity

Xiaotong Chen, Yurong Shi and Hongsun Fu *

School of Science, Dalian Maritime University, Dalian 116026, China; chenxiaotong@dlmu.edu.cn (X.C.); shiyurong@dlmu.edu.cn (Y.S.)
* Correspondence: fuhongsun@dlmu.edu.cn

Abstract: Adaptive optics (AO)-corrected retina flood illumination imaging technology is widely used for investigating both structural and functional aspects of the retina. Given the inherent low-contrast nature of original retinal images, it is necessary to perform image restoration. Total variation (TV) regularization is an efficient regularization technique for AO retinal image restoration. However, a main shortcoming of TV regularization is its potential to experience the staircase effects, particularly in smooth regions of the image. To overcome the drawback, a new image restoration model is proposed for AO retinal images. This model utilizes the overlapping group sparse total variation (OGSTV) as a regularization term. Due to the structural characteristics of AO retinal images, only partial information regarding the PSF is known. Consequently, we have to solve a more complicated myopic deconvolution problem. To address this computational challenge, we propose an ADMM-MM-LAP method to solve the proposed model. First, we apply the alternating direction method of multiplier (ADMM) as the outer-layer optimization method. Then, appropriate algorithms are employed to solve the ADMM subproblems based on their inherent structures. Specifically, the majorization–minimization (MM) method is applied to handle the asymmetry OGSTV regularization component, while a modified version of the linearize and project (LAP) method is adopted to address the tightly coupled subproblem. Theoretically, we establish the complexity analysis of the proposed method. Numerical results demonstrate that the proposed model outperforms the existing state-of-the-art TV model across several metrics.

Keywords: image restoration; retinal imaging; total variation; overlapping group sparsity; ADMM

check for updates

Academic Editors: Theodore E. Simos, Shou-Fu Tian and Shufei Wu

Received: 20 March 2025
Revised: 15 April 2025
Accepted: 24 April 2025
Published: 26 April 2025

Citation: Chen, X.; Shi, Y.; Fu, H. Adaptive Optics Retinal Image Restoration Using Total Variation with Overlapping Group Sparsity. *Symmetry* **2025**, *17*, 660. https://doi.org/10.3390/sym17050660

Copyright: © 2025 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

Adaptive optics (AO) is an optoelectronic technique that compensates for wavefront distortion in optical systems. This technique was initially developed in the field of astronomical observation [1,2] and later applied to retinal imaging. AO retinal imaging is a widely used technique to explore the structure and function of the living retina [3]. By compensating for wavefront distortions during real-time imaging, AO enables us to examine the architecture of the retina at a cellular scale. However, AO retinal images inherently contain information from both in-focus and out-of-focus planes, leading to resolution degradation in AO flood images. Therefore, accurate interpretation of these images requires advanced post-processing techniques. From a mathematical perspective, retinal imaging can be modeled as a three-dimensional (3D) convolution. When the real object demonstrates translational invariance along the optical axis, the original 3D model can be simplified to a two-dimensional (2D) representation.

In conventional 2D image restoration, the point spread function (PSF) is generally fully known or treated as completely unknown [4]. However, AO retinal image restoration presents a distinctive scenario where the true PSF (and the blurring matrix A) is only partially known. This intermediate regime between fully known and completely unknown PSFs is formally classified as a mildly blind or myopic deconvolution problem. Specifically, the global PSF of AO retinal images can be represented through a model that combines unknown parameters and a few PSFs. This combination tends to create coupled terms, making the problem more complex. Therefore, this poses a challenge to the existing methods for AO retinal image restoration problems [5]. The central difficulty lies in the inherent composite nature of the global PSF, which requires simultaneous parameter estimation during image restoration.

The mathematical formulation of myopic deconvolution for AO retinal imaging can be expressed as

$$\boldsymbol{d} = A(\boldsymbol{w})\boldsymbol{x} + \boldsymbol{\eta},$$

where $\boldsymbol{d} \in \mathbb{R}^{n^2}$ denotes an observed image with noise, $\boldsymbol{\eta} \in \mathbb{R}^{n^2}$ represents additive noise, and $\boldsymbol{x} \in \mathbb{R}^{n^2}$ represents unknown real images. $A(\boldsymbol{w}) \in \mathbb{R}^{n^2 \times n^2}$ is the blurring matrix defined by PSF, and $\boldsymbol{w} \in \mathbb{R}^p$ is an unknown parameter in the blurring matrix $A(\boldsymbol{w}) \in \mathbb{R}^{n^2 \times n^2}$. $A(\boldsymbol{w})$ is constructed as a linear combination of p known blurring matrices A_j, formulated as

$$A(\boldsymbol{w}) = \sum_{j=1}^{p} w_j A_j = w_1 A_1 + w_2 A_2 + \ldots + w_p A_p.$$

Regularization methods play a central role in solving such deconvolution problems. As a classical technique in image denoising, total variation (TV) regularization [6] effectively balances noise suppression and detail preservation through its edge-preserving diffusion mechanism. TV regularization has demonstrated robust performance across various noise environments, including Gaussian noise, Poisson noise, and other complex noise scenarios [7–10]. To address the AO retinal image denoising challenge, Chen et al. developed a novel model incorporating isotropic TV regularization [5]

$$\begin{cases} \min\limits_{\boldsymbol{x}\in\mathcal{C}_{\boldsymbol{x}},\boldsymbol{w}\in\mathcal{C}_{\boldsymbol{w}}} \Phi(\boldsymbol{x},\boldsymbol{w}) = \frac{\mu}{2}\|A(\boldsymbol{w})\boldsymbol{x} - \boldsymbol{d}\|_2^2 + \sum\limits_{i=1}^{n^2} \|D_i\boldsymbol{x}\|_2, \\ \text{s.t.} \quad \sum\limits_{j=1}^{p} w_j = 1, \end{cases}$$

where $\mathcal{C}_{\boldsymbol{x}} = \{\boldsymbol{x} \mid x_i \geq 0 \text{ for } i = 1, \ldots, n^2\}$, $\mathcal{C}_{\boldsymbol{w}} = \{\boldsymbol{w} \mid w_j \geq 0 \text{ for } j = 1, \ldots, p\}$. $\frac{\mu}{2}\|A(\boldsymbol{w})\boldsymbol{x} - \boldsymbol{d}\|_2^2$ denotes the data fidelity term. $\sum\limits_{i=1}^{n^2} \|D_i\boldsymbol{x}\|_2$ denotes the isotropic TV regularizer, which has symmetry properties or rotational invariance. $\mu > 0$ denotes the weighting parameter to balance the two terms in the objective function, w_j satisfies certain constraints. The TV model achieves simultaneous optimization of image restoration and unknown parameter estimation.

However, the primary limitation of TV regularization lies in its tendency to produce step artifacts. To address this issue, numerous enhanced regularization methods have been developed, including fractional-order TV (FOTV) [11,12], higher-order TV (HOTV) [13–15], non-local TV (NLTV) [16,17], total generalized variation (TGV) [18,19], total variation with overlapping group sparse (OGSTV) [20–22], and so on. Among them, Selesnick et al. proposed OGSTV [22] as an extension of TV, which has been successfully applied to image denoising in recent years by considering the group sparsity feature of the signal derivatives and utilizing the correlation of neighboring gradient values to retain the image detail information, which effectively reduces the step artifacts.

Specifically, the OGSTV regularizer divides the image gradient field into overlapping $K \times K$ pixel blocks (K is a fixed constant) and extends them in all directions centered on the pixel blocks. This ensures that every gradient is jointly constrained by multiple overlapping groups. Mathematically, for the two-dimensional gradient operators, $D^{+}_{(1)}\boldsymbol{u}$ and $D^{+}_{(2)}\boldsymbol{u}$, OGSTV constructs corresponding overlapping group matrices $\boldsymbol{u}_{i,j,K}$ and computes the sum of L_2 norm of gradients within each group, enforcing the global consistency of local smoothness. The OGSTV regularizer has been widely applied to different noise removal scenarios. Liu et al. effectively eliminated salt-and-pepper noise using OGSTV [23]. Yin et al. enhanced impulse noise suppression by integrating OGSTV with an L_0 norm data fidelity term [24]. Li et al. achieved high-quality restoration of Gaussian noise-degraded images through a hybrid framework combining generalized nonconvex nonsmooth fourth-order total variation with OGS regularization [25].

Inspired by the AO retinal image model proposed by Chen et al. in [5], and the better restoration effect of OGSTV in image restoration, we propose the image myopic deconvolution model with OGSTV regularization term

$$\begin{cases} \min\limits_{\boldsymbol{x}\in\mathcal{C}_{\boldsymbol{x}},\boldsymbol{w}\in\mathcal{C}_{\boldsymbol{w}}} & \Phi(\boldsymbol{x},\boldsymbol{w}) = \frac{\mu}{2}\|A(\boldsymbol{w})\boldsymbol{x}-\boldsymbol{d}\|_2^2 + \varphi(\bigtriangledown\boldsymbol{x}) \\ \text{s.t.} & \sum\limits_{j=1}^{p} w_j = 1, \end{cases} \tag{1}$$

where $\varphi(\bigtriangledown\boldsymbol{x})$ denotes the OGSTV regularization function. This is a non-convex model. To the best of our knowledge, there have been no attempts to apply OGSTV to AO retinal imaging to simultaneously perform image restoration and parameter estimation in the coupled problem.

Since the proposed model is non-convex, we adopt the alternating direction method of multipliers (ADMM) as the outer-layer optimization strategy. While ADMM was originally designed for 2-block convex optimization problems, recent theoretical advances have demonstrated its empirical effectiveness in handling non-convex objective functions or non-convex sets [26–31]. This motivates our application of ADMM to the AO retinal image restoration task. In the concrete steps, the computational architecture employs the ADMM as the outer-layer optimization framework. According to the special structure in our proposed AO retinal image restoration model, we also use the majorization–minimization (MM) method [24] and the linearize and project (LAP) method [32]. Among them, the MM method deals with the OGSTV regularizer that appears during each ADMM iteration, while the LAP method solves the tightly coupled subproblem.

The main contributions of this paper are outlined below:

(i) We propose a new myopic deconvolution model with an OGSTV regularization term arising from AO retinal image restoration. The proposed model achieves superior image restoration and enhanced parameter estimation accuracy.
(ii) To address the nonconvex nature of the model and the tight coupling between variables $\boldsymbol{x}$ and $\boldsymbol{w}$, we propose an ADMM-MM-LAP method.
(iii) We conduct comprehensive numerical experiments to evaluate the performance of our proposed OGSTV model.

The following outlines the organization of this paper. First, we propose an AO retinal imaging model with the OGSTV regularization term in Section 2. In Section 3, we present the iterative framework of ADMM and provide a concise overview of the MM method and the LAP method. Then, we propose the ADMM-MM-LAP method. Theoretically, we also analyze the computational complexity. Section 4 provides the numerical experiments, and Section 5 offers the concluding remarks.

2. AO Retinal Image Restoration Model with OGSTV Regularization

In this section, we construct a myopic deconvolution model with overlapping group sparse total variation (OGSTV) regularization. First, we introduce the discrete gradient operator $\bigtriangledown \boldsymbol{x}$ and the overlapping group sparsity (OGS) function and provide the definition of the OGSTV regularization term. Subsequently, we propose a new myopic deconvolution model with an OGSTV regularization term for AO retinal image restoration.

2.1. The First-Order Difference Operator $\bigtriangledown \boldsymbol{x}$

First, we provide the definition of the first-order finite difference. For a gray image $\boldsymbol{x} \in \mathbb{R}^{n^2}$ under periodic boundary conditions, the first-order difference operator $\nabla : \mathbb{R}^{n^2} \to \mathbb{R}^{n^2 \times 2n^2}$ is defined as

$$(\bigtriangledown \boldsymbol{x})_i = \left((D^+_{(1)}\boldsymbol{x})_i, (D^+_{(2)}\boldsymbol{x})_i \right), \tag{2}$$

where $D^+_{(1)}\boldsymbol{x}$ and $D^+_{(2)}\boldsymbol{x}$ are the first-order forward differences of $\boldsymbol{x}$ in the horizontal and vertical directions, respectively. Specifically, we have

$$\left(D^+_{(1)}\boldsymbol{x}\right)_i := \begin{cases} x_{i+n} - x_i, & \text{if } 1 \le i \le n(n-1), \\ x_{\mathrm{mod}(i,n)} - x_i, & \text{else,} \end{cases}$$

and

$$\left(D^+_{(2)}\boldsymbol{x}\right)_i := \begin{cases} x_{i+1} - x_i, & \text{if } \mathrm{mod}(i,n) \neq 0, \\ x_{i-n+1} - x_i, & \text{else,} \end{cases}$$

where $\mathrm{mod}(\cdot,\cdot)$ denotes the residual function, $\boldsymbol{x}_i$ stands for pixel of i in the image, $i = 1,\ldots, n^2$.

2.2. OGSTV and Proposed Model

For a given vector $\boldsymbol{y} \in \mathbb{R}^n$, the group K-point is defined as

$$\boldsymbol{y}_{i,K} = [\boldsymbol{y}(i), \ldots, \boldsymbol{y}(i+K-1)] \in \mathbb{R}^K,$$

where $\boldsymbol{y}_{i,K}$ represents a block of size K starting at index i. A widely used group sparse regularization [22] is formulated as

$$\phi(\boldsymbol{y}) = \sum_i \left[\sum_{k=0}^{K-1} |\boldsymbol{y}(i+k)|^2 \right]^{1/2}.$$

In the 2D scenario, the $K \times K$-point group of an image $\boldsymbol{g} \in \mathbb{R}^{n \times n}$ is given by

$$\begin{aligned} &\tilde{\boldsymbol{g}}(i,j)_K \\ &= \begin{bmatrix} \boldsymbol{g}(i-m_1, j-m_1) & \boldsymbol{g}(i-m_1, j-m_1+1) & \ldots & \boldsymbol{g}(i-m_1, j+m_2) \\ \boldsymbol{g}(i-m_1+1, j-m_1) & \boldsymbol{g}(i-m_1+1, j-m_1+1) & \ldots & \boldsymbol{g}(i-m_1+1, j+m_2) \\ \vdots & \vdots & \ddots & \vdots \\ \boldsymbol{g}(i+m_2, j-m_1) & \boldsymbol{g}(i+m_2, j-m_1+1) & \ldots & \boldsymbol{g}(i+m_2, j+m_2) \end{bmatrix} \\ &\in \mathbb{R}^{K \times K}, \end{aligned}$$

with $m_1 = \lfloor \frac{K-1}{2} \rfloor, m_2 = \lfloor \frac{K}{2} \rfloor$, where $\lfloor x \rfloor$ represents the maximum integer that is less than or equal to x. The center of $\tilde{\boldsymbol{g}}(i,j)_K$ is (i,j). Let $\boldsymbol{g}(i,j)_K$ denote a vector formed by stacking the K columns of matrix $\tilde{\boldsymbol{g}}(i,j)_K$, i.e., $\boldsymbol{g}(i,j)_K = \tilde{\boldsymbol{g}}(i,j)_K(:)$. Subsequently, the OGS function for the grayscale image $\boldsymbol{g}$ is defined as

$$\phi(\boldsymbol{g}) = \sum_{i,j=1} \|\boldsymbol{g}(i,j)_K\|_2. \tag{3}$$

Thus, combining (2) and (3), the OGSTV regularization term can be represented as

$$\varphi(\bigtriangledown \boldsymbol{x}) = \phi\left(D^+_{(1)}\boldsymbol{x}\right) + \phi\left(D^+_{(2)}\boldsymbol{x}\right). \tag{4}$$

We would like to point out that when $K = 1$, the OGSTV regularization term is a traditional anisotropic TV functional with asymmetry.

Subsequently, the proposed myopic deconvolution model incorporating the OGSTV regularization term can be expressed as

$$\begin{cases} \min\limits_{\boldsymbol{x}\in\mathcal{C}_x, \boldsymbol{w}\in\mathcal{C}_w} & \Phi(\boldsymbol{x},\boldsymbol{w}) = \frac{\mu}{2}\|A(\boldsymbol{w})\boldsymbol{x} - \boldsymbol{d}\|_2^2 + \varphi(\bigtriangledown \boldsymbol{x}) \\ \text{s.t.} & \sum\limits_{j=1}^{p} w_j = 1, \end{cases} \tag{5}$$

where $\mathcal{C}_x = \{\boldsymbol{x} \mid x_i \geq 0 \text{ for } i = 1,\ldots,n^2\}$, $\mathcal{C}_w = \{\boldsymbol{w} \mid w_j \geq 0 \text{ for } j = 1,\ldots,p\}$, $A(\boldsymbol{w}) = \sum_{j=1}^{p} w_j A_j$ and $\varphi(\bigtriangledown \boldsymbol{x})$ denotes the OGSTV regularization term.

Moreover, to limit the 'sum of weights' to be close to 1, a suitable regularization term is introduced on w. Specifically, a quadratic penalty function $S(\boldsymbol{w})$ is introduced, which for sufficiently large ξ effectively guarantees that the weights $w_j, j = 1,\ldots,p$ are summed equal to 1. Consequently, we propose the following model:

$$\min_{\boldsymbol{x}\in\mathcal{C}_x, \boldsymbol{w}\in\mathcal{C}_w} \frac{\mu}{2}\|A(\boldsymbol{w})\boldsymbol{x} - \boldsymbol{d}\|_2^2 + \varphi(\bigtriangledown \boldsymbol{x}) + S(\boldsymbol{w}), \tag{6}$$

where the regularization $S(\boldsymbol{w})$ is defined as

$$S(\boldsymbol{w}) = \frac{\xi}{2}(\boldsymbol{e}^\top \boldsymbol{w} - 1)^2,$$

where $\xi > 0$ denotes a weighting parameter, and $\boldsymbol{e} \in \mathbb{R}^p$ denotes the all-ones vector.

3. Optimization Schemes

In this section, we employ the alternating direction method of multipliers (ADMM) as an outer-layer optimization framework to solve the proposed model (6). The ADMM subproblems are efficiently solved by two strategies: the majorization–minimization (MM) method is applied to address the OGSTV component, while a modified version of the linearize and project (LAP) method is implemented to address the tightly coupled subproblem. Based on the strategies above, we propose an ADMM-MM-LAP method.

3.1. ADMM Scheme

Considering the structure of the model (6), we apply ADMM as an outer-layer algorithm to solve it. By introducing the artificial vector $\boldsymbol{v}$, we rewrite (6) in the following equivalent optimization problem

$$\begin{cases} \min\limits_{\boldsymbol{x},\boldsymbol{w},\boldsymbol{v}} \frac{\mu}{2}||A(\boldsymbol{w})\boldsymbol{x} - \boldsymbol{d}||_2^2 + \varphi(\boldsymbol{v}) + S(\boldsymbol{w}) + \delta_{\mathcal{C}_x}(\boldsymbol{x}) + \delta_{\mathcal{C}_w}(\boldsymbol{w}) \\ \text{s.t.} \quad \boldsymbol{v} = \bigtriangledown \boldsymbol{x}, \end{cases} \tag{7}$$

where $\delta_{\mathcal{C}}(\cdot)$ represents the indicator function

$$\delta_{\mathcal{C}}(s) = \begin{cases} 0, & s \in \mathcal{C}, \\ \infty, & s \notin \mathcal{C}. \end{cases}$$

The augmented Lagrangian function for (7) is given by

$$\begin{aligned} &L(\boldsymbol{x}, \boldsymbol{w}, \boldsymbol{v}, \boldsymbol{\alpha}) \\ =&\frac{\mu}{2}\|A(\boldsymbol{w})\boldsymbol{x} - \boldsymbol{d}\|_2^2 + S(\boldsymbol{w}) + \delta_{\mathcal{C}_x}(\boldsymbol{x}) + \delta_{\mathcal{C}_w}(\boldsymbol{w}) + \varphi(\boldsymbol{v}) - \boldsymbol{\alpha}^T(\boldsymbol{v} - \bigtriangledown \boldsymbol{x}) + \frac{\rho}{2}\|\boldsymbol{v} - \bigtriangledown \boldsymbol{x}\|_2^2, \end{aligned} \tag{8}$$

where $\boldsymbol{\alpha}$ represents the Lagrangian multiplier and $\rho > 0$ represents the penalty parameter.

The main idea of ADMM for solving (7) is to alternately update variables by minimizing the augmented Lagrangian function (8). Then, we treat $\boldsymbol{v}$ as a block of variables and $(\boldsymbol{x}, \boldsymbol{w})$ as another block of variables. Given an initial point $(\boldsymbol{v^0}, \boldsymbol{x^0}, \boldsymbol{w^0}, \boldsymbol{\alpha^0})$, ADMM iteratively solves the following $\boldsymbol{v}$-subproblem and $(\boldsymbol{x}, \boldsymbol{w})$-subproblem

$$\begin{cases} \boldsymbol{v}^{k+1} = \arg\min\limits_{\boldsymbol{v}} L(\boldsymbol{x}^k, \boldsymbol{w}^k, \boldsymbol{v}, \boldsymbol{\alpha}^k), \\ (\boldsymbol{x}^{k+1}, \boldsymbol{w}^{k+1}) = \arg\min\limits_{\boldsymbol{x},\boldsymbol{w}} L(\boldsymbol{x}, \boldsymbol{w}, \boldsymbol{v}^{k+1}, \boldsymbol{\alpha}^k), \\ \boldsymbol{\alpha}^{k+1} = \boldsymbol{\alpha}^k + \rho\Big(\bigtriangledown \boldsymbol{x}^{k+1} - \boldsymbol{v}^{k+1}\Big). \end{cases}$$

First, the $\boldsymbol{v}$-subproblem can be formulated as

$$\begin{aligned} \boldsymbol{v}^{k+1} &= \arg\min_{\boldsymbol{v}} \frac{\rho}{2}\Big\|\boldsymbol{v} - \bigtriangledown \boldsymbol{x}^k\Big\|_2^2 - \boldsymbol{\alpha}^T(\boldsymbol{v} - \bigtriangledown \boldsymbol{x}^k) + \varphi(\boldsymbol{v}) \\ &= \arg\min_{\boldsymbol{v}} \frac{\rho}{2}\|\boldsymbol{v} - (\bigtriangledown \boldsymbol{x}^k + \frac{\boldsymbol{\alpha}^k}{\rho})\|_2^2 + \varphi(\boldsymbol{v}). \end{aligned} \tag{9}$$

Second, for the $(\boldsymbol{x}, \boldsymbol{w})$-subproblem,

$$(\boldsymbol{x}^{k+1}, \boldsymbol{w}^{k+1}) = \arg\min_{\boldsymbol{x},\boldsymbol{w}} \frac{\mu}{2}\|A(\boldsymbol{w})\boldsymbol{x} - \boldsymbol{d}\|_2^2 + S(\boldsymbol{w}) - \boldsymbol{\alpha}^T(\boldsymbol{v}^{k+1} - \bigtriangledown \boldsymbol{x}) + \frac{\rho}{2}\Big\|\boldsymbol{v}^{k+1} - \bigtriangledown \boldsymbol{x}\Big\|_2^2,$$

let

$$C(\boldsymbol{x}, \boldsymbol{v}, \boldsymbol{\alpha}) = \varphi(\boldsymbol{v}) - \alpha^T(\boldsymbol{v} - \bigtriangledown \boldsymbol{x}) + \frac{\rho}{2}\|\boldsymbol{v} - \bigtriangledown \boldsymbol{x}\|_2^2,$$

and define

$$\hat{\Psi}(\boldsymbol{x}, \boldsymbol{w}, \boldsymbol{v}, \boldsymbol{\alpha}) := \frac{\mu}{2}\|A(\boldsymbol{w})\boldsymbol{x} - \boldsymbol{d}\|_2^2 + S(\boldsymbol{w}) + C(\boldsymbol{x}, \boldsymbol{v}, \boldsymbol{\alpha}).$$

Then, the $(\boldsymbol{x}, \boldsymbol{w})$-subproblem can be formulated as

$$(\boldsymbol{x}^{k+1}, \boldsymbol{w}^{k+1}) = \min_{\boldsymbol{x}\in\mathcal{C}_x, \boldsymbol{w}\in\mathcal{C}_w} \hat{\Psi}(\boldsymbol{x}, \boldsymbol{w}, \boldsymbol{v}^{k+1}, \boldsymbol{\alpha}^k). \tag{10}$$

Finally, we update the Lagrangian multiplier

$$\boldsymbol{\alpha}^{k+1} = \boldsymbol{\alpha}^k + \rho\Big(\bigtriangledown \boldsymbol{x}^{k+1} - \boldsymbol{v}^{k+1}\Big). \tag{11}$$

Next, we will solve these two subproblems according to their respective structural characteristics.

3.2. The Majorization–Minimization Algorithm (MM)

To tackle the $\boldsymbol{v}$-subproblem (9), we adopt the MM method in [24]. Before providing the solution to (9), we specifically explore a more generalized problem formulation, i.e.,

$$\min_{\boldsymbol{v}}\{F(\boldsymbol{v}) = \frac{1}{2}\|\boldsymbol{v} - \boldsymbol{v}_0\|_2^2 + \frac{1}{\rho}\varphi(\boldsymbol{v})\}, \boldsymbol{v} \in \mathbb{R}^{n^2}, \tag{12}$$

where $\boldsymbol{v}_0 = \bigtriangledown \boldsymbol{x}^k + \frac{\boldsymbol{\alpha}^k}{\rho}$ and the function $\varphi(\cdot)$ is given by (4). Rather than directly optimizing the complicated problem (9), we employ the MM method to iteratively solve it. Specifically, we simply look for a sequence of convex functions: $G\left(\boldsymbol{v}, \boldsymbol{v}^k\right), k = 0, 1, 2, \ldots$. The principle underlying the MM method is that each $G(\boldsymbol{v}, \boldsymbol{v}^k)$ serves as an upper bound on $F(\boldsymbol{v})$ for any $\boldsymbol{v}$ and $\boldsymbol{v}^k$ and attains $F(\boldsymbol{v})$ at $G(\boldsymbol{v}, \boldsymbol{v}) = F(\boldsymbol{v})$. The updated iteration yields a reduced cost. However, the complexity within the MM framework lies in constructing an appropriate function $G(\boldsymbol{v}, \boldsymbol{v}^k)$.

Notice that the inequality

$$\frac{1}{2}\left(\frac{1}{\|\boldsymbol{u}\|_2}\|\boldsymbol{v}\|_2^2 + \|\boldsymbol{u}\|_2\right) \geq \|\boldsymbol{v}\|_2, \tag{13}$$

for all $\boldsymbol{v}$ and $\boldsymbol{u} \neq 0$ with equality when $\boldsymbol{u} = \boldsymbol{v}$. By plugging each group of $\varphi(\boldsymbol{v})$ into (13) and aggregating the results, we obtain a majorizor of $\varphi(\boldsymbol{v})$. Let

$$T(\boldsymbol{v}, \boldsymbol{u}) = \frac{1}{2}\sum_{i,j=1}^{n}\left[\frac{1}{\|\boldsymbol{u}_{i,j,K}\|_2}\|\boldsymbol{v}_{i,j,K}\|_2^2 + \|\boldsymbol{u}_{i,j,K}\|_2\right]$$

with $T(\boldsymbol{v}, \boldsymbol{u}) \geq \varphi(\boldsymbol{v})$, $T(\boldsymbol{u}, \boldsymbol{u}) = \varphi(\boldsymbol{u})$, provided that $\|\boldsymbol{u}_{i,j,K}\|_2 \neq 0$. By collating, $T(\boldsymbol{v}, \boldsymbol{u})$ can be reformulated as

$$T(\boldsymbol{v}, \boldsymbol{u}) = \frac{1}{2}\|\Omega(\boldsymbol{u})\boldsymbol{v}\|_2^2 + C(\boldsymbol{u}), \tag{14}$$

where $C(\boldsymbol{u})$ represents a constant, which is independent on $\boldsymbol{v}$, and $\Omega(\boldsymbol{u})$ denotes a diagonal matrix whose diagonal elements are defined as

$$[\Omega(\boldsymbol{u})]_{l,l} = \sqrt{\sum_{i,j=-m_1}^{m_2}\left[\sum_{k_1,k_2=-m_1}^{m_2}|\boldsymbol{u}_{r-i+k_1,t-j+k_2}|^2\right]^{-\frac{1}{2}}}, \tag{15}$$

with $l = (r-1)n + t$, $\forall r, t = 1, 2, \ldots, n$, the entries of $\Omega(\boldsymbol{u})$ can be efficiently computed via convolution operation.

Then, the majorizer of $F(\boldsymbol{v})$ can be expressed as

$$G(\boldsymbol{v}, \boldsymbol{u}) = \frac{1}{2}\|\boldsymbol{v} - \boldsymbol{v}_0\|_2^2 + \frac{1}{\rho}T(\boldsymbol{v}, \boldsymbol{u}) = \frac{1}{2}\|\boldsymbol{v} - \boldsymbol{v}_0\|_2^2 + \frac{1}{2\rho}\|\Omega(\boldsymbol{u})\boldsymbol{v}\|_2^2 + \mu C(\boldsymbol{u}).$$

We have $G(\boldsymbol{v}, \boldsymbol{u}) \geq F(\boldsymbol{v})$ for any $\boldsymbol{u}$, $\boldsymbol{v}$, and $G(\boldsymbol{u}, \boldsymbol{u}) = F(\boldsymbol{u})$. In order to obtain the optimal $\boldsymbol{v}$ from $F(\boldsymbol{v})$, the MM iterations can be performed as

$$\boldsymbol{v}^{k+1} = \left(I + \mu\Omega\left(\boldsymbol{v}^k\right)^\top\Omega\left(\boldsymbol{v}^k\right)\right)^{-1}\boldsymbol{v}_0,\ k = 0, 1, \ldots, \tag{16}$$

where I represents the identity matrix of the same dimension as $\Omega\left(\boldsymbol{v}^k\right)$. We would like to point out that the inversion of $I + \mu\Omega\left(\boldsymbol{v}^k\right)^\top\Omega\left(\boldsymbol{v}^k\right)$ can be computed efficiently, as it requires only straightforward computations of its components. To sum up, Algorithm 1 shows the iterative format of the MM method for (12).

Algorithm 1 The MM method for (12)

1: **Input**: $v_0 \in \mathbb{R}^{n^2}$.
2: **Output**: Set $v = v^{k+1}$.
3: **Initialize**: $v^0 = v_0$, $k = 0$, ρ, group size K, maximum iterations N.
4: **Iterates**
5: Calculate $\Omega(v^k)$ according to (15).
6: Calculate v^{k+1} according to (16).
7: $k = k + 1$.
8: until iteration N is satisfied.

3.3. The Linearize and Project (LAP)

To solve the (x, w)-subproblem (10), we employ a modified version of the linearize and project (LAP) method. The original LAP framework is by Herring et al. [32] for solving coupled optimization problems, incorporating fundamental principles derived from the Gauss–Newton method. By using linearization and projection steps to deal with coupled variables, this method can effectively deal with tightly coupled optimization problems and complex problems with pointwise boundary constraints.

The main step of the LAP method is to linearize the residuals first, and then eliminate a set of variables by projection operation to obtain the approximation problem. The variables can be classified into two distinct groups: active set variables and inactive set variables. Subsequently, the update steps δx and δw can be calculated within both sets. First, we formulate the set of feasible indicators as

$$\mathcal{N} := \left\{ q \in \mathbb{N} \middle| \begin{bmatrix} x_q \\ w_q \end{bmatrix} \geq 0 \right\}.$$

The active set and inactive set are defined as

$$\mathcal{A} := \left\{ q \in \mathbb{N} \middle| \begin{bmatrix} x_q \\ w_q \end{bmatrix} = \mathbf{0} \right\},$$
$$\mathcal{I} := \mathcal{N} \backslash \mathcal{A},$$

respectively.

Then, the variables can be divided into active set variables $\begin{bmatrix} x_{\mathcal{A}} \\ w_{\mathcal{A}} \end{bmatrix}$, and inactive set variables $\begin{bmatrix} x_{\mathcal{I}} \\ w_{\mathcal{I}} \end{bmatrix}$. The update step can be represented as $\begin{bmatrix} \delta x_{\mathcal{A}} \\ \delta w_{\mathcal{A}} \end{bmatrix}$ and $\begin{bmatrix} \delta x_{\mathcal{I}} \\ \delta w_{\mathcal{I}} \end{bmatrix}$, respectively.

The computation method of $\delta x_{\mathcal{I}}$ and $\delta w_{\mathcal{I}}$ is similar to the unconstrained situation. It follows the idea of Newton's method, the update steps $\begin{bmatrix} \delta x \\ \delta w \end{bmatrix}$ are given by

$$\begin{bmatrix} \delta x \\ \delta w \end{bmatrix} = - \bigtriangledown^2_{x,w} \hat{\Psi}\left(x, w, v^{k+1}, \alpha^k\right)^{-1} \bigtriangledown_{x,w} \hat{\Psi}\left(x, w, v^{k+1}, \alpha^k\right).$$

That is,

$$\begin{bmatrix} \delta x \\ \delta w \end{bmatrix} = - \begin{bmatrix} \mathbf{J}_x^\top \mathbf{J}_x + \frac{1}{\mu} \bigtriangledown^2_x C\left(x, v^{k+1}, \alpha^k\right) & \mathbf{J}_x^\top \mathbf{J}_w \\ \mathbf{J}_w^\top \mathbf{J}_x & \mathbf{J}_w^\top \mathbf{J}_w + \bigtriangledown^2_w S(w) \end{bmatrix}^{-1} \begin{bmatrix} \mathbf{J}_x^\top r + \frac{1}{\mu} \bigtriangledown_x C\left(x, v^{k+1}, \alpha^k\right) \\ \mathbf{J}_w^\top r + \bigtriangledown_w S(w) \end{bmatrix},$$

where $\bigtriangledown_x \boldsymbol{C}(\boldsymbol{x}, \boldsymbol{v^{k+1}}, \boldsymbol{\alpha^k}) = \bigtriangledown^T \boldsymbol{\alpha}^k - \rho \bigtriangledown^T (\boldsymbol{v}^{k+1} - \bigtriangledown \boldsymbol{x})$ and $\bigtriangledown_x^2 \boldsymbol{C}(\boldsymbol{x}, \boldsymbol{v^{k+1}}, \boldsymbol{\alpha^k}) = \rho \bigtriangledown^T \bigtriangledown$. $\mathbf{J}_{\boldsymbol{x}}$ and $\mathbf{J}_{\boldsymbol{w}}$ denote the Jacobian matrices with respect to the $\boldsymbol{x}$ block and the $\boldsymbol{w}$ block, respectively. However, it should be noted that the variables must be projected onto the inactive set. To be specific, the value of $\delta \boldsymbol{x}_{\mathcal{I}}$ at the present iteration point $(\boldsymbol{x}, \boldsymbol{w})$ can be computed by

$$\hat{\boldsymbol{M}} \delta \boldsymbol{x}_{\mathcal{I}} = \hat{\boldsymbol{b}}, \tag{17}$$

$$\begin{aligned}
\hat{\boldsymbol{M}} :=& \hat{\mathbf{J}}_{\boldsymbol{x}}^{\top} \left(\boldsymbol{I} - \hat{\mathbf{J}}_{\boldsymbol{w}} \left(\hat{\mathbf{J}}_{\boldsymbol{w}}^{\top} \hat{\mathbf{J}}_{\boldsymbol{w}} + \bigtriangledown_{\boldsymbol{w}}^2 \hat{S}(\boldsymbol{w}) \right)^{-1} \hat{\mathbf{J}}_{\boldsymbol{w}}^{\top} \right) \hat{\mathbf{J}}_{\boldsymbol{x}} + \frac{1}{\mu} \bigtriangledown_{\boldsymbol{x}}^2 \hat{C} \left(\boldsymbol{x}, \boldsymbol{v}^{k+1}, \boldsymbol{\alpha}^k \right), \\
\hat{\boldsymbol{b}} :=& - \hat{\mathbf{J}}_{\boldsymbol{x}}^{\top} \left(\boldsymbol{I} - \hat{\mathbf{J}}_{\boldsymbol{w}} \left(\hat{\mathbf{J}}_{\boldsymbol{w}}^{\top} \hat{\mathbf{J}}_{\boldsymbol{w}} + \bigtriangledown_{\boldsymbol{w}}^2 \hat{S}(\boldsymbol{w}) \right)^{-1} \hat{\mathbf{J}}_{\boldsymbol{w}}^{\top} \right) \boldsymbol{r} - \frac{1}{\mu} \bigtriangledown_{\boldsymbol{x}} \hat{C} \left(\boldsymbol{x}, \boldsymbol{v}^{k+1}, \boldsymbol{\alpha}^k \right) \\
& + \hat{\mathbf{J}}_{\boldsymbol{x}}^{\top} \hat{\mathbf{J}}_{\boldsymbol{w}} \left(\hat{\mathbf{J}}_{\boldsymbol{w}}^{\top} \hat{\mathbf{J}}_{\boldsymbol{w}} + \bigtriangledown_{\boldsymbol{w}}^2 \hat{S}(\boldsymbol{w}) \right)^{-1} \bigtriangledown_{\boldsymbol{w}} \hat{S}(\boldsymbol{w}),
\end{aligned}$$

where $\hat{\mathbf{J}}_{\boldsymbol{x}}, \hat{\mathbf{J}}_{\boldsymbol{w}}, \bigtriangledown_x \hat{C}, \bigtriangledown_x^2 \hat{C}, \bigtriangledown_w \hat{S}, \bigtriangledown_w^2 \hat{S}$ denote $\mathbf{J}_{\boldsymbol{x}}, \mathbf{J}_{\boldsymbol{w}}, \bigtriangledown_x C, \bigtriangledown_x^2 C, \bigtriangledown_w S, \bigtriangledown_w^2 S$ restricted to the inactive set , respectively. The reduced problem (17) does not require a high-precision solution. For instance, in [32], a stopping tolerance of 10^{-1} is employed for solving the reduced problem iteratively. Once the value of $\delta \boldsymbol{x}_{\mathcal{I}}$ has been obtained through the solving process, the value of $\delta \boldsymbol{w}_{\mathcal{I}}$ can then be calculated by

$$\delta \boldsymbol{w}_{\mathcal{I}} = - \left(\hat{\mathbf{J}}_{\boldsymbol{w}}^{\top} \hat{\mathbf{J}}_{\boldsymbol{w}} + \bigtriangledown_{\boldsymbol{w}}^2 \hat{S}(\boldsymbol{w}) \right)^{-1} \left(\hat{\mathbf{J}}_{\boldsymbol{w}}^{\top} \hat{\mathbf{J}}_{\boldsymbol{x}} \delta \boldsymbol{x}_{\mathcal{I}} + \hat{\mathbf{J}}_{\boldsymbol{w}}^{\top} \boldsymbol{r} + \bigtriangledown_w \hat{S}(\boldsymbol{w}) \right). \tag{18}$$

For the active set, $\delta \boldsymbol{x}_{\mathcal{A}}$ and $\delta \boldsymbol{w}_{\mathcal{A}}$ can be obtained through

$$\begin{bmatrix} \delta \boldsymbol{x}_{\mathcal{A}} \\ \delta \boldsymbol{w}_{\mathcal{A}} \end{bmatrix} = -\mu \begin{bmatrix} \tilde{\mathbf{J}}_{\boldsymbol{x}}^{\top} \boldsymbol{r} \\ \tilde{\mathbf{J}}_{\boldsymbol{w}}^{\top} \boldsymbol{r} \end{bmatrix} - \begin{bmatrix} \bigtriangledown_x \tilde{C} \left(\boldsymbol{x} + \delta \boldsymbol{x}_{\mathcal{A}}, \boldsymbol{v}^{k+1}, \boldsymbol{\alpha}^k \right) \\ \bigtriangledown_w \tilde{S}(\boldsymbol{w}) \end{bmatrix}, \tag{19}$$

where $\tilde{\mathbf{J}}_{\boldsymbol{x}}, \tilde{\mathbf{J}}_{\boldsymbol{w}}, \bigtriangledown_x \tilde{C}$, $\bigtriangledown_w \tilde{S}$ denote the projection of $\mathbf{J}_{\boldsymbol{x}}, \mathbf{J}_{\boldsymbol{w}}, \bigtriangledown_x C$, $\bigtriangledown_w S$ onto the active set, respectively.

Subsequently, $\delta \boldsymbol{x}$ and $\delta \boldsymbol{w}$ can be computed as a combination of $\delta \boldsymbol{x}_{\mathcal{A}}$, $\delta \boldsymbol{w}_{\mathcal{A}}$, $\delta \boldsymbol{x}_{\mathcal{I}}$, and $\delta \boldsymbol{w}_{\mathcal{I}}$ in the following equation

$$\begin{bmatrix} \delta \boldsymbol{x} \\ \delta \boldsymbol{w} \end{bmatrix} = \begin{bmatrix} \delta \boldsymbol{x}_{\mathcal{I}} \\ \delta \boldsymbol{w}_{\mathcal{I}} \end{bmatrix} + \theta \begin{bmatrix} \delta \boldsymbol{x}_{\mathcal{A}} \\ \delta \boldsymbol{w}_{\mathcal{A}} \end{bmatrix}, \tag{20}$$

the appropriate parameter θ can be selected based on [33].

After that, the update steps are integrated through the projection gradient descent method, with the active set and inactive set being iteratively updated to guarantee the point-wise boundary constraints. This comprehensive approach enables the effective handling of large-scale optimization problems with tightly coupled variables and point-wise boundary constraints. The LAP method is used to solve the $(\boldsymbol{x}, \boldsymbol{w})$-subproblem inexactly. The residual is

$$\begin{bmatrix} \eta_{\boldsymbol{x}}^{k+1} \\ \eta_{\boldsymbol{w}}^{k+1} \end{bmatrix} := \bigtriangledown_{(\boldsymbol{x}, \boldsymbol{w})} \hat{\Psi}(\boldsymbol{x}, \boldsymbol{w}, \boldsymbol{v}, \boldsymbol{\alpha}).$$

For the stopping criterion, let

$$\begin{bmatrix} \eta_{\boldsymbol{x}}^{k+1} \\ \eta_{\boldsymbol{w}}^{k+1} \end{bmatrix} \leq \varepsilon_{k+1},$$

where $\{\varepsilon_{k+1}\}_{k=0}^{\infty}$ is a non-negative sequence satisfying $\sum_{k=0}^{\infty} \varepsilon_{k+1} < \infty$.

In summary, from the above analysis, we provide the iteration scheme for solving (12) in Algorithm 2.

Algorithm 2 The LAP method for (12)

1: **Input**: $(\boldsymbol{x}^0, \boldsymbol{w}^0) \in \mathbb{R}^{n^2} \times \mathbb{R}^p$. Let $k = 0$.
2: **Output**: $(\boldsymbol{x}^k, \boldsymbol{w}^k)$.
3: **Iterates**
4: Apply the LAP method to compute $\delta \boldsymbol{x}_{\mathcal{I}}$ and $\delta \boldsymbol{w}_{\mathcal{I}}$ according to (17) and (18), respectively.
5: Apply the projection gradient descent method to compute $\delta \boldsymbol{x}_{\mathcal{A}}$ and $\delta \boldsymbol{w}_{\mathcal{A}}$ according to (19).
6: Combine $\delta \boldsymbol{x}_{\mathcal{A}}, \delta \boldsymbol{w}_{\mathcal{A}}, \delta \boldsymbol{x}_I, \delta \boldsymbol{w}_I$ according to (20).
7: Compute $\boldsymbol{x}^{k+1}$ and $\boldsymbol{w}^{k+1}$ with a projected Armijo line search.
8: Update the active set $\mathcal{A}$ and the inactive set $\mathcal{I}$.
9: $k := k + 1$.
10: until the termination condition is satisfied.

3.4. ADMM-MM-LAP Method

Based on the representations and analysis above, we present the iterative scheme of the ADMM-MM-LAP method for solving the AO retinal image model (7) with the OGSTV regularization term in Algorithm 3.

Algorithm 3 ADMM-MM-LAP method for (7)

1: **Input**: v_0, x_0, w_0 and parameters $\mu > 0, \rho > 0$, group size K.
2: **Output**: $v^k, x^k, w^k, \boldsymbol{\alpha}^k$.
3: **Initialize**: $x_0 = f, k = 0, \boldsymbol{\alpha}^0 = \mathbf{0}$.
4: Apply Algorithm 1 to compute $\boldsymbol{v}^{(k+1)}$.
5: Apply Algorithm 2 to compute $(\boldsymbol{x}^{(k+1)}, \boldsymbol{w}^{(k+1)})$.
6: Update the multipliers using (11).
7: $k := k + 1$.
8: until the termination condition is satisfied.

3.5. Complexity Analysis

In this subsection, the computational cost generated at each step of Algorithm 3 is considered, and the computational complexity analysis is given.

First, the $\boldsymbol{v}$-subproblem can be solved by Algorithm 1. Each iteration needs to compute the diagonal matrix $\Omega\left(\boldsymbol{v}^k\right)$, and the computational complexity of each iteration is $O(n)$ by calculating diagonal elements through convolution operation. Then, the matrix inverse calculation can be computed efficiently by simple element-by-element calculation, and the overall computational complexity remains $O(n)$. Moreover, the computational cost is also related to the group size K, and K is a constant. Therefore, based on the above analysis, the complexity of $\boldsymbol{v}$-subproblem is $O(n)$.

Second, the $(\boldsymbol{x}, \boldsymbol{w})$-subproblem can be efficiently solved using Algorithm 2, whose computational cost comprises several main subcomponents, such as $\delta \boldsymbol{x}_{\mathcal{I}}, \delta \boldsymbol{w}_{\mathcal{I}}, \delta \boldsymbol{x}_{\mathcal{A}}, \delta \boldsymbol{w}_{\mathcal{A}}$. The computation primarily involves three core components: matrix–vector product, vector norm computation, and conjugate gradient method. Then, we proceed with a detailed step-by-step complexity analysis. First, for computing the update step $\delta \boldsymbol{x}_{\mathcal{I}}$, it is necessary to calculate the product of matrix $\frac{1}{\mu} \bigtriangledown_{\boldsymbol{x}}^2 \hat{C}\left(\boldsymbol{x}, \boldsymbol{v}^{k+1}, \boldsymbol{\alpha}^k\right) + \hat{\mathbf{J}}_{\boldsymbol{x}}^\top \left(\boldsymbol{I} - \hat{\mathbf{J}}_{\boldsymbol{w}} \left(\hat{\mathbf{J}}_{\boldsymbol{w}}^\top \hat{\mathbf{J}}_{\boldsymbol{w}} + \bigtriangledown_{\boldsymbol{w}}^2 \hat{S}(\boldsymbol{w})\right)^{-1} \hat{\mathbf{J}}_{\boldsymbol{w}}^\top\right) \hat{\mathbf{J}}_{\boldsymbol{x}}$, vectors $-\hat{\mathbf{J}}_{\boldsymbol{x}}^\top \left(\boldsymbol{I} - \hat{\mathbf{J}}_{\boldsymbol{w}} \left(\hat{\mathbf{J}}_{\boldsymbol{w}}^\top \hat{\mathbf{J}}_{\boldsymbol{w}} + \bigtriangledown_{\boldsymbol{w}}^2 \hat{S}(\boldsymbol{w})\right)^{-1} \hat{\mathbf{J}}_{\boldsymbol{w}}^\top\right) \boldsymbol{r}$ and $\hat{\mathbf{J}}_{\boldsymbol{x}}^\top \hat{\mathbf{J}}_{\boldsymbol{w}} \left(\hat{\mathbf{J}}_{\boldsymbol{w}}^\top \hat{\mathbf{J}}_{\boldsymbol{w}} + \bigtriangledown_{\boldsymbol{w}}^2 \hat{S}(\boldsymbol{w})\right)^{-1} \bigtriangledown_{\boldsymbol{w}} \hat{S}(\boldsymbol{w})$, respectively. Due to the Jacobi matrix $\mathbf{J}_{\boldsymbol{x}} = A(\boldsymbol{w})^\top$ and each PSF matrix, A_i supports vector multiplication via the Fast Fourier Transform (FFT), the computational complexity of multiplying $\mathbf{J}_{\boldsymbol{x}}$ with a vector is $O(n^2 log n)$. Similarly, multiplying $\mathbf{J}_{\boldsymbol{w}}$ with a vector also achieves $O(n^2 log n)$ complexity by FFTs. So the matrix–vector product involving $\hat{\mathbf{J}}_{\boldsymbol{x}}^\top \left(\boldsymbol{I} - \hat{\mathbf{J}}_{\boldsymbol{w}} \left(\hat{\mathbf{J}}_{\boldsymbol{w}}^\top \hat{\mathbf{J}}_{\boldsymbol{w}} + \bigtriangledown_{\boldsymbol{w}}^2 \hat{S}(\boldsymbol{w})\right)^{-1} \hat{\mathbf{J}}_{\boldsymbol{w}}^\top\right) \hat{\mathbf{J}}_{\boldsymbol{x}}$ has the same $O(n^2 log n)$ complexity. Moreover, the computational cost for the matrix–vector product of $\frac{1}{\mu} \bigtriangledown_{\boldsymbol{x}}^2 \hat{C}\left(\boldsymbol{x}, \boldsymbol{v}^{k+1}, \boldsymbol{\alpha}^k\right)$ is $O(n^2)$. Fur-

thermore, the operations involving the expressions $-\hat{\mathbf{J}}_{\boldsymbol{x}}^{\top}\left(\boldsymbol{I}-\hat{\mathbf{J}}_{\boldsymbol{w}}\left(\hat{\mathbf{J}}_{\boldsymbol{w}}^{\top}\hat{\mathbf{J}}_{\boldsymbol{w}}+\bigtriangledown_{\boldsymbol{w}}^{2}\hat{S}(\boldsymbol{w})\right)^{-1}\hat{\mathbf{J}}_{\boldsymbol{w}}^{\top}\right)\boldsymbol{r}$ and $\hat{\mathbf{J}}_{\boldsymbol{x}}^{\top}\hat{\mathbf{J}}_{\boldsymbol{w}}\left(\hat{\mathbf{J}}_{\boldsymbol{w}}^{\top}\hat{\mathbf{J}}_{\boldsymbol{w}}+\bigtriangledown_{\boldsymbol{w}}^{2}\hat{S}(\boldsymbol{w})\right)^{-1}\bigtriangledown_{\boldsymbol{w}}\hat{S}(\boldsymbol{w})$ can be computed via FFTs with a complexity of $O(n^2logn)$. Thus, the overall computational cost of $\delta\boldsymbol{x}_{\mathcal{I}}$ is $O(n^2logn)$. Similarly, for $\delta\boldsymbol{w}_{\mathcal{I}}, \delta\boldsymbol{x}_{\mathcal{A}}, \delta\boldsymbol{w}_{\mathcal{A}}$, the computations of their matrix–vector products also demand acceleration by FFTs, such as $\hat{\mathbf{J}}_{\boldsymbol{w}}^{\top}\hat{\mathbf{J}}_{\boldsymbol{x}}\delta\boldsymbol{x}_{\mathcal{I}}$, $\hat{\mathbf{J}}_{\boldsymbol{w}}\boldsymbol{r}$,$\tilde{\mathbf{J}}_{\boldsymbol{x}}^{\top}\boldsymbol{r}$, $\tilde{\mathbf{J}}_{\boldsymbol{w}}^{\top}\boldsymbol{r}$, etc. Therefore, the computational cost of calculating $\delta\boldsymbol{w}_{\mathcal{I}}, \delta\boldsymbol{x}_{\mathcal{A}}, \delta\boldsymbol{w}_{\mathcal{A}}$ is $O(n^2logn)$.

In addition, update steps $\delta\boldsymbol{x}$ and $\delta\boldsymbol{w}$ mainly involve scalar–vector multiplication and vector summation calculation, while the projection Armijo line search involves the calculations of $\bigtriangledown_{x,w}\hat{\Psi}\left(\boldsymbol{x},\boldsymbol{w},\boldsymbol{v}^{k+1},\boldsymbol{\alpha}^{k}\right)$ and the inner product. The computational cost is $O(n^2)$. In summary, the $(\boldsymbol{x},\boldsymbol{w})$-subproblem complexity is $O(n^2logn)$.

Finally, in the Lagrangian multiplier update step, the computational cost is dominated by the calculations of $\rho\boldsymbol{v}^{k+1}$ and $\rho\bigtriangledown\boldsymbol{x}^{k+1}$, each requiring $O(n^2)$ operations. Therefore, we can conclude the following theorem that the computational complexity per iteration of Algorithm 3 is $O(n^2\log n)$.

Theorem 1. *The computational complexity of Algorithm 3 is* $O(n^2logn)$.

4. Numerical Experiments

In this section, we present the numerical experimental results of the proposed model, which incorporates the OGSTV regularization term for myopic deconvolution of AO retinal images. All experimental results were implemented using MATLAB 2020a on a PC with 16.0 GB RAM and AMD Ryzen 5 4600U with Radeon Graphics at 2.10 GHz.

The restoration performance is evaluated by five principal metrics, including the relative errors of $\boldsymbol{x}$ and $\boldsymbol{w}$, signal-to-noise ratio (SNR), peak signal-to-noise ratio (PSNR), and structural similarity index (SSIM). Specifically, the metrics are defined as follows:

- $\mathrm{RE_x} = \frac{\|\boldsymbol{x}-\boldsymbol{x}^*\|_2}{\|\boldsymbol{x}^*\|_2}$, where $\boldsymbol{x}^*$ denotes the true image and $\boldsymbol{x}$ denotes the restored image.
- $\mathrm{RE}_w = \frac{\|\boldsymbol{w}-\boldsymbol{w}^*\|_2}{\|\boldsymbol{w}^*\|_2}$, where $\boldsymbol{w}^*$ is the real parameter, and $\boldsymbol{w}$ is the calculated parameter.
- $\mathrm{SNR} = 10\cdot\log_{10}\frac{\|\boldsymbol{x}^*-\tilde{\boldsymbol{x}}\|_2^2}{\|\boldsymbol{x}^*-\boldsymbol{x}\|_2^2}$, where $\boldsymbol{x}^*$ denotes the true image, $\tilde{\boldsymbol{x}}$ denotes the mean intensity value of $\boldsymbol{x}^*$, and $\boldsymbol{x}$ denotes the restored image.
- $\mathrm{PSNR} = 10\cdot\log_{10}\frac{N_p}{\|\boldsymbol{x}^*-\boldsymbol{x}\|_2}$, where N_p refers to the size of the image, $\boldsymbol{x}^*$ and $\boldsymbol{x}$ denote the true image and the restored image, respectively.
- $\mathrm{SSIM} = \frac{(2\mu_x\mu_{x^*}+\mathrm{C}_1)(2\sigma+\mathrm{C}_2)}{(\mu_x^2+\mu_{x^*}^2+\mathrm{C}_1)(\sigma_x^2+\sigma_{x^*}^2+\mathrm{C}_2)}$, where C_1 and C_2 are constants. σ denotes the covariance between the images $\boldsymbol{x}$ and $\boldsymbol{x}^*$. μ_{x^*}, σ_{x^*}, and μ_x, σ_x denote the average and standard variance of the AO retinal image $\boldsymbol{x}^*$ and $\boldsymbol{x}$, respectively.

The quantitative evaluation of retinal image restoration requires multiple metrics. Although PSNR provides computationally efficient distortion measurements via mean square error (MSE), its correlation with clinical diagnostic relevance remains limited due to the non-linear nature of human visual perception. In contrast, SSIM demonstrates superior alignment with subjective evaluations by modeling luminance, contrast, and structural correlations. For the comprehensive analysis, we employ both metrics alongside relative errors to ensure mathematical rigor and parameter estimation.

The parameters are carefully configured through experimental verification. In the OGSTV model, the group size K is set to 3 for ensuring restoration performance [12]. We set the parameter $\xi = 100$ so that the obtained parameter meets $\sum_{j=1}^{p}\omega_j = 1$. The parameter ρ is chosen as 9. In order to guarantee the non-negative sequence $\{\varepsilon_{k+1}\}_{k=0}^{\infty}$ satisfying

$\sum_{k=0}^{\infty} \varepsilon_{k+1} < \infty$, we choose $\varepsilon_{k+1} = \frac{1}{b(k+1)^2}$, with $b > 0$. We implement a unified stopping criterion

$$\frac{|\hat{\Psi}(\boldsymbol{x}^{k+1}, \boldsymbol{\omega}^{k+1}, \boldsymbol{v}^{k+1}, \boldsymbol{\alpha}^{k+1}) - \hat{\Psi}(\boldsymbol{x}^{k}, \boldsymbol{w}^{k}, \boldsymbol{v}^{k}, \boldsymbol{\alpha}^{k})|}{|\hat{\Psi}(\boldsymbol{x}^{k}, \boldsymbol{w}^{k}, \boldsymbol{v}^{k}, \boldsymbol{\alpha}^{k})|} < \varepsilon,$$

where $\hat{\Psi}(\boldsymbol{x}^{k+1}, \boldsymbol{\omega}^{k+1}, \boldsymbol{v}^{k+1}, \boldsymbol{\alpha}^{k+1})$ represents the function value at the $(k+1)$-th iteration. The maximum number of iterations for the ADMM is set to 50. For the MM method, the inner iteration number N is set to 10.

In the context of AO retinal imaging, we use the TV model combined with the ADMM-LAP algorithm as the benchmark method [5]. In the following numerical experiments, we present quantitative comparisons between the proposed OGSTV model and the classical TV model, including relative errors for both reconstructed images and estimated parameters, as well as SNR, PSNR, and SSIM.

4.1. Example 1

Problem Setting

For the myopic deconvolution problem of AO retinal images, $p = 2$ is established in this example. The global PSF is synthesized by combining two distinct components: a focused PSF and a defocused PSF. The test image is a 256×256 segment extracted from an AO retinal image. The test problem is constructed using the regularization toolbox IR Tools [34]. It can produce different BlurLevel indicators ('mild', 'medium', 'severe'), and simulate spatially invariant Gaussian blur and out-of-focus blur using the *PRblurgauss* function and *PRblurdefocus* function in the IR Tools, respectively. For example, the *PRblurgauss* function simulates a spatially invariant Gaussian blur to construct a PSF, while the *PRblurdefocus* function simulates a spatially invariant out-of-focus blur for another PSF. In the example, three cases are designed. One PSF is fixed as a spatially invariant Gaussian blur generated by the *PRblurgauss* function with a 'mild' blur level, whereas the other PSF is a composite of two components: a 'mild' Gaussian blur from *PRblurgauss*, and a defocus blur from *PRblurdefocus* with three distinct levels ('mild', 'medium', and 'severe'). Next, we choose the parameter $\mu = 6 \times 10^5$ and set ε in the stopping criterion to $\varepsilon = 0.05$. Gaussian noise is added via the *PRnoise* function and the noise level is 0.01. The true parameter is set to $\omega^\star$ = [0.3; 0.7]. We choose the initial value of the parameter ω^0 = [0.5; 0.5]. The initial guess x^0 is a random image with 256×256 pixels.

Experimental results

As an example, when the BlurLevel of *PRblurdefocus* is set to 'medium', comparing the image restoration effects of our proposed OGSTV model and the classical TV model, it can be seen from Figure 1c,d that both models restored the images relatively clearly. It is not easy to judge which one of the OGSTV model and the TV model restores better by the eye only. To show the differences more clearly, Figure 1e–h presents an enlarged fragment of the corresponding images. In Figure 1g, the image recovered by the OGSTV model exhibits enhanced clarity and contrast, while the background in Figure 1h demonstrates a coarser background texture and has more residual noise. This indicates that the restored image obtained by the OGSTV model is more similar to the true image, and its restoration effect is better than that of the TV model.

For a quantitative comparison of Example 1, Figure 2 and Table 1 enumerate the RE_x, RE_w, SNR, PSNR, and SSIM of the restored images. In Figure 2, we plot the curve of RE_x and RE_w changing with the number of iterations. We can see that the relative errors of the OGSTV model in the recovered image and estimating parameters are all lower than those of the TV model. It shows that our OGSTV model simultaneously enhances image restoration

accuracy and improves parameter recovery precision. This validates our OGSTV model's capability to outperform the TV model in AO retinal image processing.

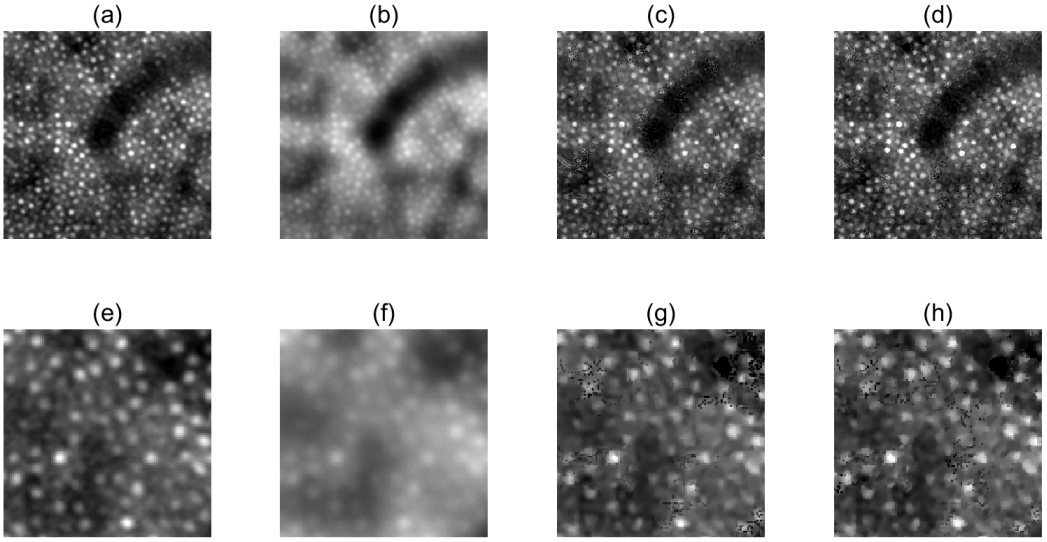


Figure 1. Results of AO retinal image restoration in Example 1. The BlurLevel of $PRblurdefocus$ is set to 'medium'. (**a**) True image; (**b**) degraded image; (**c**) restored image by the OGSTV model; (**d**) restored image by the TV model; (**e**–**h**) correspond to the enlarged segments of (**a**-**d**).

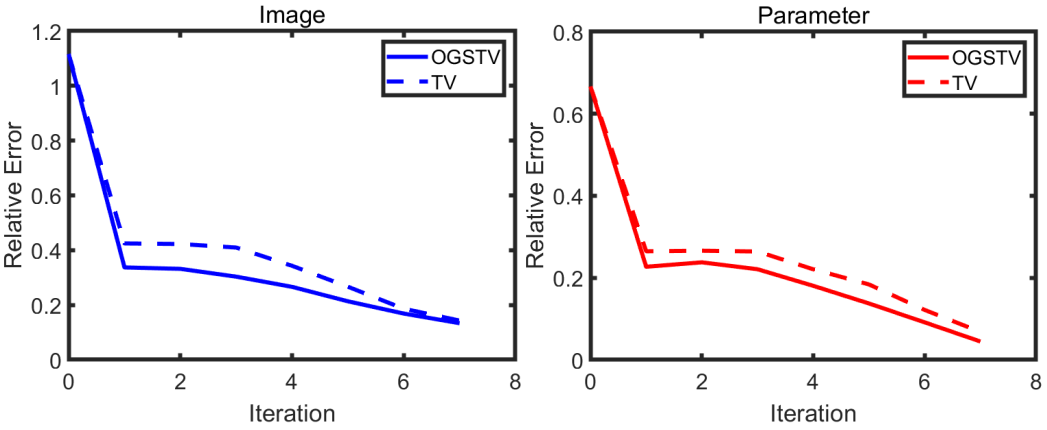


Figure 2. RE_x (**left**) and RE_w (**right**) versus iteration number in Example 1.

In Table 1, the data comparison between the two models under three Blurlevels is detailed. Our OGSTV model outperforms the TV model in the key metrics. Specifically, our proposed OGSTV achieves consistently lower relative errors RE_x and RE_w than the TV model, reflecting its enhanced precision in reconstructing AO retinal images and parameter estimation. Simultaneously, the OGSTV model's accuracy is further corroborated by its superior performance in SNR, PSNR, and SSIM levels. Although the OGSTV model exhibits slightly higher computational time compared to the TV model, the disparity remains minor. More critically, the OGSTV model recovers AO retinal images of better quality within a similar computation time. Notably, under the 'severe' $BlurLevel$, the OGSTV achieves a 35.7% lower RE_x and a 33.2% higher SNR compared to the TV model. It indicates that the OGSTV model demonstrates increasing gaps in computationally demanding cases, and the image restoration effect is getting better. Therefore, it again shows that the OGSTV model outperforms the TV model in this test problem.

Table 1. The numerical results for Example 1.

BlurLevel	Model	RE_x	RE_w	SNR	PSNR	SSIM	Time (s)
'mild'	OGSTV	$\mathbf{1.61 \times 10^{-1}}$	$\mathbf{6.01 \times 10^{-2}}$	**8.86**	**25.24**	**0.72**	33.65
	TV	2.98×10^{-1}	2.21×10^{-1}	3.50	19.88	0.52	**21.46**
'medium'	OGSTV	$\mathbf{1.33 \times 10^{-1}}$	$\mathbf{4.48 \times 10^{-2}}$	**10.47**	**26.86**	**0.81**	35.60
	TV	1.43×10^{-1}	6.81×10^{-2}	9.87	26.25	0.78	**31.29**
'severe'	OGSTV	$\mathbf{7.52 \times 10^{-2}}$	$\mathbf{4.38 \times 10^{-2}}$	**15.46**	**31.84**	**0.92**	**37.53**
	TV	1.17×10^{-1}	4.39×10^{-2}	11.60	27.99	0.85	50.11

The use of bold indicates that the numerical results are better.

4.2. Example 2

Problem Setting

The myopic deconvolution problem of AO retinal imaging is also considered, and the image size is 256 × 256. In this example, we build a combined PSF using *PRblurgauss* with a 'mild' BlurLevel and *PRblurdefocus* with three BlurLevels ('mild', 'medium', 'severe') in IR Tools. The other PSF is constructed using only *PRblurgauss* with a 'mild' BlurLevel. We choose the parameter $\mu = 5 \times 10^5$ and set the stopping criterion with $\varepsilon = 0.05$. Add Gaussian noise with a noise level of 0.01. Let the real parameter be $\omega^\star$ = [0.3; 0.7]. To initiate the process, we select an arbitrary 256 × 256 image as the initial value x^0 and select the initial value of the parameter $\omega^0 = [\omega_1^0; \omega_2^0] \in \mathbb{R}^2$, where ω_1^0, ω_2^0 are constants in [0, 1] and satisfy $\omega_1^0 + \omega_2^0 = 1$.

Experimental results

The experimental results are consistent with the results of Example 1. In this example, the BlurLevel of *PRblurdefocus* is 'mild'. We still need to observe the enlarged fragments of the two models in Figure 3. It can be observed that the background region in Figure 3g retains more complete texture details with significantly reduced residual noise levels. The images in Figure 3 further indicate that the image restored by our proposed OGSTV model resembles the true image more closely.

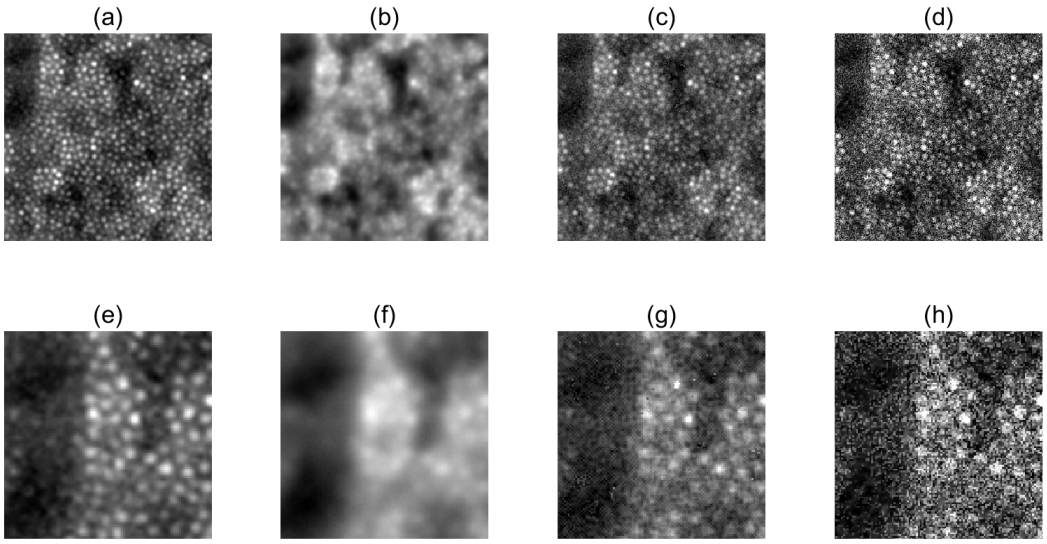


Figure 3. Results of AO retinal image restoration in Example 2. The BlurLevel of *PRblurdefocus* is set to 'mild'. (**a**) True image ; (**b**) degraded image; (**c**) restored image by the OGSTV model; (**d**) restored image by the TV model; (**e**–**h**) correspond to the enlarged segments of (**a**-**d**).

Figure 4 shows the relative errors of the reconstructed image and the estimated parameters against the number of iterations. As shown in Figure 4, although the TV model achieves a marginally lower image's relative error at the fourth iteration, the OGSTV model demonstrates superior stability. Under identical stopping criteria, the OGSTV model attains a significantly smaller final error after additional iterations. Notably, the parameter estimation error of the OGSTV model converges to a negligible magnitude.

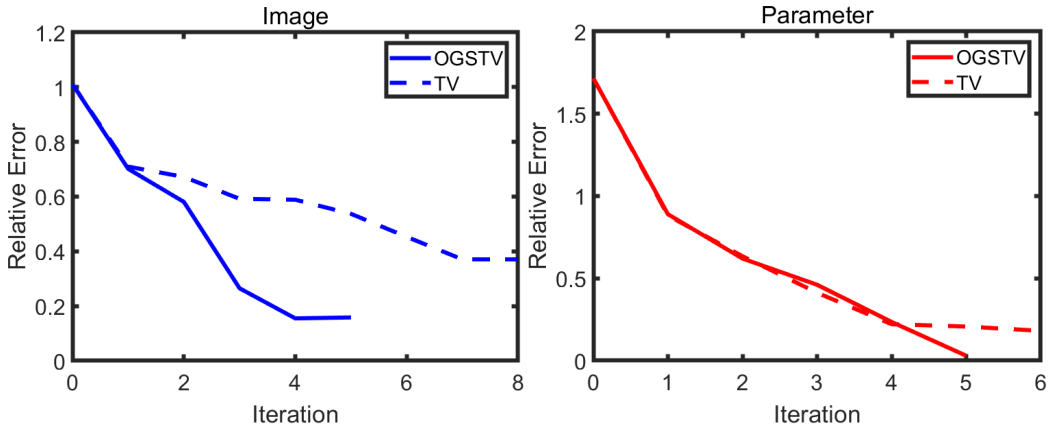


Figure 4. RE_x (**left**) and RE_w (**right**) versus iteration number in Example 2.

In Table 2, a detailed comparison of the data obtained from the two methods under three distinct Blurlevels is presented. The OGSTV model demonstrates consistent superiority over the TV model in both image reconstruction quality and parameter estimation accuracy across all tested conditions. Meanwhile, the OGSTV framework achieves significantly higher values in key quantitative metrics, including SNR, PSNR, and SSIM. Although the TV model exhibits marginally shorter computational times in some blur scenarios, the OGSTV model maintains a better balance between computational efficiency and restoration precision.

Table 2. The numerical results for Example 2.

BlurLevel	Model	RE_x	RE_w	SNR	PSNR	SSIM	Time (s)
'mild'	OGSTV	$\mathbf{1.58 \times 10^{-1}}$	$\mathbf{3.12 \times 10^{-2}}$	**8.11**	**24.86**	**0.68**	**16.12**
	TV	3.70×10^{-1}	2.01×10^{-1}	0.74	17.48	0.41	21.34
'medium'	OGSTV	$\mathbf{1.10 \times 10^{-1}}$	$\mathbf{1.58 \times 10^{-2}}$	**11.30**	**28.05**	**0.80**	**17.11**
	TV	1.63×10^{-1}	3.70×10^{-1}	7.85	24.59	0.72	49.06
'severe'	OGSTV	$\mathbf{1.13 \times 10^{-1}}$	$\mathbf{4.06 \times 10^{-1}}$	**11.05**	**27.79**	**0.89**	45.83
	TV	1.70×10^{-1}	8.23×10^{-1}	7.48	24.22	0.77	**30.19**

The use of bold indicates that the numerical results are better.

4.3. Example 3

Problem Setting

In this example, we examine the scenario where the parameter p is set to 3. We use the same AO retina image as in Example 1, with a portion of 256 × 256. In Example 3, the first PSF is constructed by *PRblurgauss* ('mild'). The second PSF is a combination of *PRblurgauss* ('mild') and *PRblurdefocus* ('medium'), and the third PSF is a combination of *PRblurgauss* ('mild') and *PRblurdefocus* ('mild', 'medium', and 'severe') combination. The parameter μ is chosen as $\mu = 6 \times 10^5$, and the parameter ε in the stopping criterion is set to $\varepsilon = 0.01$. We add Gaussian noise with a noise level of 0.01. Let the real parameter

$\omega^* = [\omega_1^*, \omega_2^*, \omega_3^*] \in \mathbb{R}^3$ be a random quantity with $\omega_i^* \geq 0$, $\omega_1^* + \omega_2^* + \omega_3^* = 1$. We select a random 256×256 image to serve as the initial value x^0. The initial value of the parameter is choosen as $\omega^0 = [\omega_1^0; \omega_2^0; \omega_3^0]$, where $\omega_1^0, \omega_2^0, \omega_3^0$ are constants in [0, 1] and satisfy $\omega_1^0 + \omega_2^0 + \omega_3^0 = 1$.

Experimental results

In Figure 5, we show the corresponding resulting image when the BlurLevel of $PRblurdefocus$ is 'severe'. A larger p means that the AO image denoising is also more complex. In Example 3, the image restoration is significantly more computationally challenging than the previous two examples. By observing the images, unlike Example 1 and 2, we do not necessitate observing enlarged images to evaluate the image restoration. Just by observing Figure 5c–d, it is obvious that the images recovered by the OGSTV model are closer to the real image than those recovered by the TV model. This demonstrates the superior denoising capability of the OGSTV model, particularly when processing images with intricate structures.

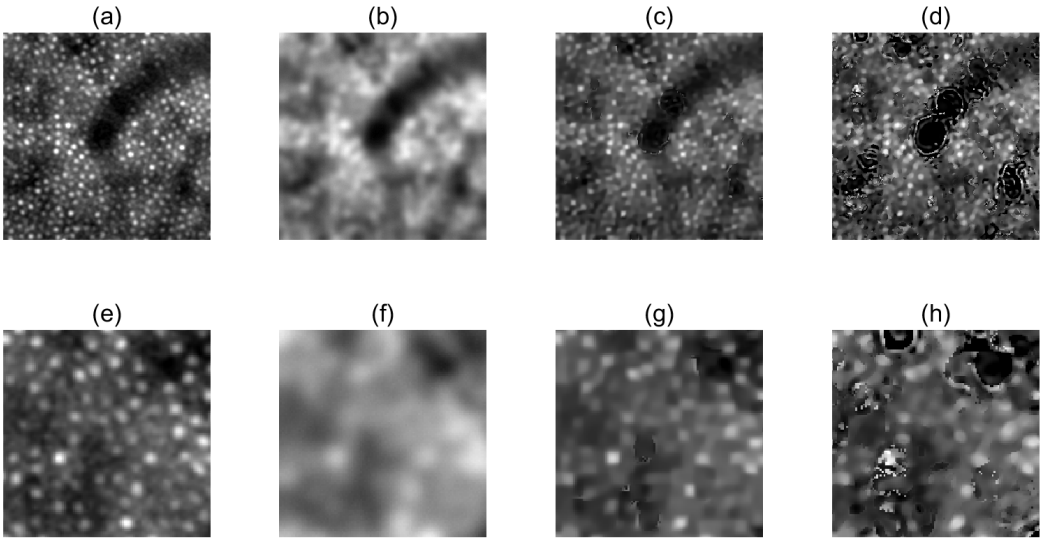


Figure 5. Results of AO retinal image restoration in Example 3. The BlurLevel of PRblurdefocus in the third PSF is set to 'severe'. (**a**) True image; (**b**) degraded image; (**c**) restored image by the OGSTV model; (**d**) restored image by the TV model; (**e**–**h**) correspond to the enlarged segments of (**a**-**d**).

In Figure 6, we plot the relative errors against the number of iterations for both models. As in Example 1 and 2, the OGSTV model outperforms the TV model in both the image restoration and parameter estimation tasks. In addition, the TV model terminates the operation at the fifth iteration, while the OGSTV model can perform more iterations and generate smaller relative error values in each iteration.

Table 3 presents the data comparison between the two models at three different Blurlevels. The results indicate a correlation between increased blur and higher computational complexity, as evidenced by the rise in processing time across both models. Although the TV model has some advantages in terms of computation time, its key indexes are significantly degraded under high blur conditions. In contrast, the OGSTV model showed significant advantages, with better performance in relative error, SNR, PSNR, and SSIM, which fully verified the validity and reliability of the proposed OGSTV model in AO retinal image restoration. In conclusion, our proposed OGSTV model outperforms the TV model in AO retinal image restoration in all test cases.

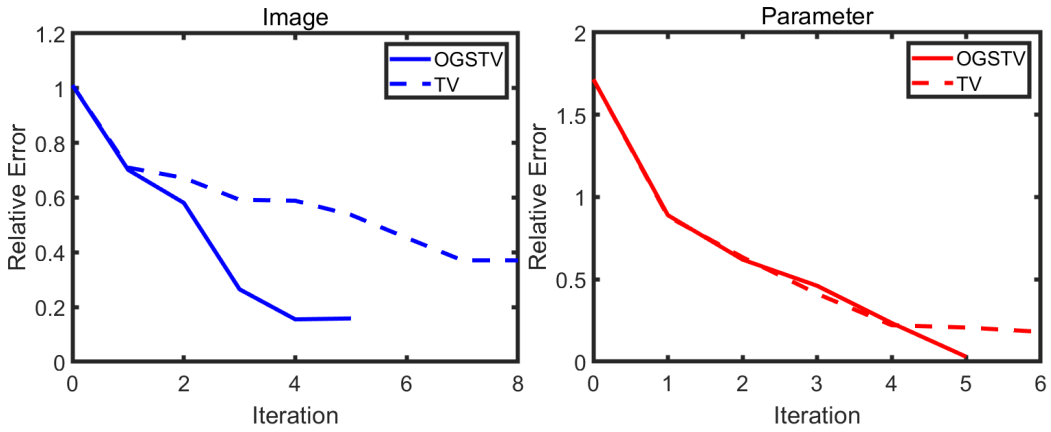


Figure 6. RE_x (**left**) and RE_w (**right**) versus iteration number in Example 3.

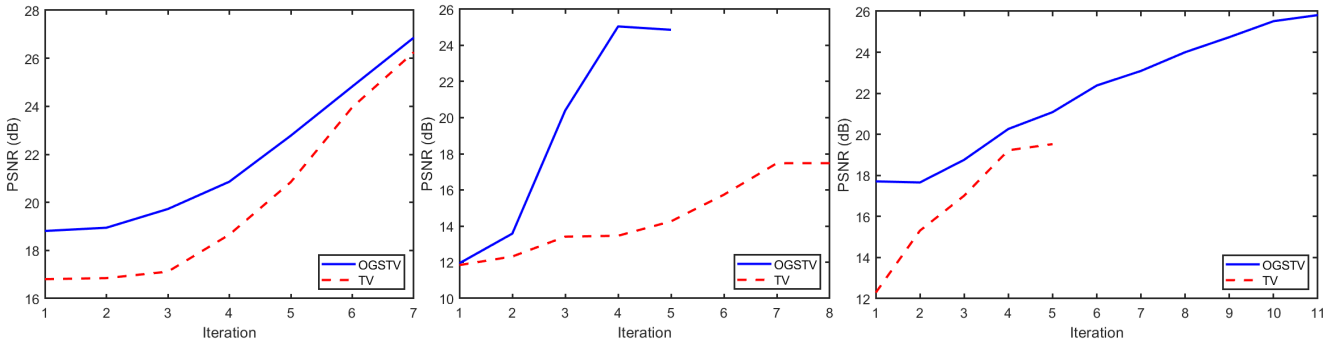
Table 3. The numerical results for Example 3.

BlurLevel	Model	RE_x	RE_w	SNR	PSNR	SSIM	Time (s)
'mild'	OGSTV	$\mathbf{1.30 \times 10^{-1}}$	$\mathbf{8.64 \times 10^{-2}}$	**10.70**	**27.09**	**0.78**	111.65
	TV	1.32×10^{-1}	1.81×10^{-1}	10.54	26.93	0.77	**93.94**
'medium'	OGSTV	$\mathbf{1.38 \times 10^{-1}}$	$\mathbf{1.97 \times 10^{-1}}$	**10.20**	**26.59**	**0.76**	548.61
	TV	3.17×10^{-1}	7.04×10^{-1}	2.97	19.35	0.52	**381.43**
'severe'	OGSTV	$\mathbf{1.51 \times 10^{-1}}$	$\mathbf{2.28 \times 10^{-1}}$	**9.43**	**25.81**	**0.73**	378.33
	TV	3.10×10^{-1}	3.28×10^{-1}	3.14	19.52	0.52	**225.78**

The use of bold indicates that the numerical results are better.

4.4. Performance Comparison

To evaluate the performance differences between the proposed OGSTV model and the TV model, we show the PSNR variations of both models during the iterative process across the above three examples. The corresponding curves of PSNR versus iteration numbers are shown in Figure 7. The results demonstrate that our proposed OGSTV model exhibits significant convergence and achieves higher PSNR levels compared to the TV model.



(**a**) (**b**) (**c**)

Figure 7. Comparison of PSNR between the proposed OGSTV model and the TV model across iterations. (**a**) Example 1, (**b**) Example 2, (**c**) Example 3.

Moreover, we would like to point out that the OGSTV model requires substantial computational resources for processing high-resolution images. Consequently, in the aforementioned simulations, OGSTV exhibited longer processing times compared to TV in some examples. Future research will prioritize the development of lightweight variants of OGSTV to reduce the computational cost while preserving the reconstruction accuracy.

5. Conclusions

In this paper, we propose a myopic deconvolution model with OGSTV regularization. To efficiently solve the proposed model, we propose an ADMM-MM-LAP method. Specifically, ADMM is applied as an outer-layer optimization method, while the MM method and the LAP method are applied as internal optimization methods, respectively. Theoretically, we give the complexity analysis of the ADMM-MM-LAP method. Numerical experiments demonstrate that the proposed OGSTV model is superior to the existing state-of-the-art TV model. In our future work, we plan to work with more real clinical AO retinal datasets and compare our algorithm with some algorithms developed for high-performance AO systems like the Large Binocular Telescope.

Author Contributions: Conceptualization, X.C. and H.F.; methodology, X.C. and H.F.; software, Y.S.; validation, X.C.; formal analysis, Y.S. and X.C.; writing—original draft preparation, X.C. and Y.S.; writing—review and editing, H.F.; visualization, Y.S.; supervision, H.F.; funding acquisition, X.C. and H.F. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the National Natural Science Foundation of China (No. 12301477, No. 42274166), the Fundamental Scientific Research Projects of Higher Education Institutions of Liaoning Provincial Department of Education (No. JYTMS20230165), and the Fundamental Research Funds for the Central Universities.

Data Availability Statement: The data presented in this study are available upon request from the corresponding author.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Beckers, J.M. Adaptive optics for astronomy: Principles, performance, and applications. *Annu. Rev. Astron. Astrophys.* **1993**, *31*, 13–62. [CrossRef]
2. Davies, R.; Kasper, M. Adaptive optics for astronomy. *Annu. Rev. Astron. Astrophys.* **2012**, *50*, 305–351. [CrossRef]
3. Blanco, L.; Mugnier, L.M. Marginal blind deconvolution of adaptive optics retinal images. *Opt. Express* **2011**, *19*, 23227–23239. [CrossRef] [PubMed]
4. Hansen, P.C.; Nagy, J.G.; O'leary, D.P. *Deblurring Images: Matrices, Spectra, and Filtering*; SIAM: Philadelphia, PA, USA, 2006. [CrossRef]
5. Chen, X.; Herring, J.L.; Nagy, J.G.; Xi, Y.; Yu, B. An ADMM-LAP method for total variation myopic deconvolution of adaptive optics retinal images. *Inverse Probl.* **2020**, *37*, 014001. [CrossRef]
6. Rudin, L.I.; Osher, S.; Fatemi, E. Nonlinear total variation based noise removal algorithms. *Phys. D Nonlinear Phenom.* **1992**, *60*, 259–268. [CrossRef]
7. Ding, M.; Huang, T.; Wang, S.; Mei, J.; Zhao, X. Total variation with overlapping group sparsity for deblurring images under Cauchy noise. *Appl. Math. Comput.* **2019**, *341*, 128–147. [CrossRef]
8. Hu, Y.; Nagy, J.G.; Zhang, J.; Andersen, M.S. Nonlinear optimization for mixed attenuation polyenergetic image reconstruction. *Inverse Probl.* **2019**, *35*, 064004. [CrossRef]
9. Huo, L.; Chen, W.; Ge, H. Image restoration based on transformed total variation and deep image prior. *Appl. Math. Model.* **2024**, *130*, 191–207. [CrossRef]
10. Zhao, R. Nanocrystalline SEM image restoration based on fractional-order TV and nuclear norm. *Electron. Res. Arch.* **2024**, *32*, 4954–4968. [CrossRef]
11. Ren, Z.; He, C.; Zhang, Q. Fractional order total variation regularization for image super-resolution. *Signal Process.* **2013**, *93*, 2408–2421. [CrossRef]
12. Zhang, X.; Cai, G.; Li, M.; Bi, S. An image-denoising framework using L_q norm-based higher order variation and fractional variation with overlapping group sparsity. *Fractal Fract.* **2023**, *7*, 573. [CrossRef]
13. Adam, T.; Paramesran, R. Image denoising using combined higher order non-convex total variation with overlapping group sparsity. *Multidimens. Syst. Signal Process.* **2019**, *30*, 503–527. [CrossRef]
14. Thanh, D.N.; Prasath, V.S.; Hieu, L.M.; Dvoenko, S. An adaptive method for image restoration based on high-order total variation and inverse gradient. *Signal Image Video Process.* **2020**, *14*, 1189–1197. [CrossRef]

15. Chan, T.; Marquina, A.; Mulet, P. High-order total variation-based image restoration. *SIAM J. Sci. Comput.* **2000**, *22*, 503–516. [CrossRef]
16. Hou, G.; Pan, Z.; Wang, G.; Yang, H.; Duan, J. An efficient nonlocal variational method with application to underwater image restoration. *Neurocomputing* **2019**, *369*, 106–121. [CrossRef]
17. Jidesh, P.; Holla, S. Non-local total variation regularization models for image restoration. *Comput. Electr. Eng.* **2018**, *67*, 114–133. [CrossRef]
18. Bredies, K.; Kunisch, K.; Pock, T. Total generalized variation. *SIAM J. Imaging Sci.* **2010**, *3*, 492–526. [CrossRef]
19. Lv, Y. Total generalized variation denoising of speckled images using a primal-dual algorithm. *J. Appl. Math. Comput.* **2020**, *62*, 489–509. [CrossRef]
20. Chen, P.; Selesnick, I.W. Group-sparse signal denoising: Non-convex regularization, convex optimization. *IEEE Trans. Signal Process.* **2014**, *62*, 3464–3478. [CrossRef]
21. Selesnick, I.W.; Farshchian, M. Sparse signal approximation via nonseparable regularization. *IEEE Trans. Signal Process.* **2017**, *65*, 2561–2575. [CrossRef]
22. Selesnick, I.W.; Chen, P. Total variation denoising with overlapping group sparsity. In Proceedings of the 2013 IEEE International Conference on Acoustics, Speech and Signal Processing, Vancouver, BC, Canada, 26–31 May 2013, pp. 5696–5700. [CrossRef]
23. Liu, J.; Huang, T.; Selesnick, I.W.; Lv, X.; Chen, P. Image restoration using total variation with overlapping group sparsity. *Inf. Sci.* **2015**, *295*, 232–246. [CrossRef]
24. Yin, M.; Adam, T.; Paramesran, R.; Hassan, M.F. An l_0-overlapping group sparse total variation for impulse noise image restoration. *Signal Process. Image Commun.* **2022**, *102*, 116620. [CrossRef]
25. Li, R.; Zheng, B. Generalized nonconvex nonsmooth four-directional total variation with overlapping group sparsity for image restoration. *J. Comput. Appl. Math.* **2024**, *451*, 116045. [CrossRef]
26. Hien, L.T.K.; Phan, D.N.; Gillis, N. Inertial alternating direction method of multipliers for non-convex non-smooth optimization. *Comput. Optim. Appl.* **2022**, *83*, 247–285. [CrossRef]
27. Wang, Y.; Yin, W.; Zeng, J. Global convergence of ADMM in nonconvex nonsmooth optimization. *J. Sci. Comput.* **2019**, *78*, 29–63. [CrossRef]
28. Xiu, X.; Liu, W.; Li, L.; Kong, L. Alternating direction method of multipliers for nonconvex fused regression problems. *Comput. Stat. Data Anal.* **2019**, *136*, 59–71. [CrossRef]
29. Zhang, B.; Zhu, G.; Zhu, Z.; Kwong, S. Alternating direction method of multipliers for nonconvex log total variation image restoration. *Appl. Math. Model.* **2023**, *114*, 338–359. [CrossRef]
30. Li, C.; Zhao, D. A non-convex fractional-order differential equation for medical image Restoration. *Symmetry* **2024**, *16*, 258. [CrossRef]
31. Xie, Z.; Liu, L.; Luo, Z.; Huang, J. Image denoising using nonlocal regularized deep image prior. *Symmetry* **2021**, *13*, 2114. [CrossRef]
32. Herring, J.L.; Nagy, J.G.; Ruthotto, L. LAP: A linearize and project method for solving inverse problems with coupled variables. *Sampl. Theory Signal Image Process.* **2018**, *17*, 127–151. [CrossRef]
33. Haber, E. *Computational Methods in Geophysical Electromagnetics*; SIAM: Philadelphia, PA, USA, 2014. [CrossRef]
34. Gazzola, S.; Hansen, P.C.; Nagy, J.G. IR Tools: A MATLAB package of iterative regularization methods and large-scale test problems. *Numer. Algorithms* **2019**, *81*, 773–811. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.