EURASIAN JOURNAL OF MATHEMATICAL AND COMPUTER APPLICATIONS
ISSN 2306–6172
Volume 12, Issue 3 (2024) 117 – 134

CAUCHY NOISE REMOVAL USING HIGH-ORDER TOTAL VARIATION AND OVERLAPPING GROUP SPARSITY

Pham C. T., Tran T. T. T, Truong T. C., Nguyen N. A., Pham T. H.

Abstract Image denoising presents a significant challenge in computer vision, aiming to eliminate unwanted noise from images and restore their original quality. Cauchy noise, characterized by its heavy-tailed distribution, poses a unique hurdle among various types of noise. While several denoising models have been proposed, Total Variation (TV)-based methods, such as the Remove Cauchy Noise model, are widely used for their effectiveness. However, these approaches often suffer from issues such as oversmoothing, staircase artifacts, and the introduction of false details. To address these limitations, recent research has explored the combination of high-order TV with Overlapping Group Sparsity (OGS), showing promising results in noise removal. Inspired by this, our article introduces a novel Cauchy denoising model. Our approach leverages OGS and directional higher-order TV to effectively remove Cauchy noise while preserving image details and minimizing aliasing and smoothing artifacts. The Chambolle-Pock algorithm efficiently solves the underlying optimization problem. Through qualitative and quantitative evaluations, including visualization and parameter measurements, we demonstrate the competitiveness of our model compared to existing methods.

Key words: Image denoising, Cauchy noise, Overlapping group sparsity, Directional total variation, Chambolle-Pock algorithm.

AMS Mathematics Subject Classification: 35A15, 94A08.

DOI: 10.32523/2306-6172-2024-12-3-117-134

1 Introduction

Image noise or degradation, a ubiquitous phenomenon, results from various unwanted factors affecting signal acquisition and processing [1]. Because noise contamination is inherently irreversible, image denoising and deblurring remain fundamental challenges in computer vision. The impact of image noise spans diverse disciplines such as digital photography, medical imaging, remote sensing, and wireless communication. Its manifestations are equally diverse, encompassing Gaussian, Poisson, salt-and-pepper, and multiplicative noise, among others [2, 3]. This work specifically addresses Cauchy noise, which is a non-Gaussian additive variant characterized by its heavy-tailed distribution. Such noise is encountered in real-world applications such as low-frequency atmospheric signals [4, 5], underwater acoustic engineering [6], sonar and radar applications [7, 8], wireless communication systems [9, 10]. For instance, in multi-access networks, interference tends to be non-Gaussian and heavy-tailed [9], and sonar signal processing in shallow water often grapples with highly impulsive ambient noise [7]. These examples underscore the close correspondence between these real-world scenarios and the

characteristics of the Cauchy distribution. Now, let's consider an image corrupted by additive Cauchy noise, which can be modeled as follows:

$$\mathbf{f} = \mathbf{H}\mathbf{u} + \mathbf{v},$$

where $\mathbf{f} \in \mathbb{R}^{m\times n}$ represents the observed noisy image degraded from the original true image $\mathbf{u} \in \mathbb{R}^{m\times n}$, $\mathbf{H}$ denotes the blur matrix, and $\mathbf{v}$ is the additive noise captured by the Cauchy distribution [11, 12]:

$$g(\mathbf{v}) = \frac{1}{\pi}\frac{\gamma}{\gamma^2 + (\mathbf{v}-\sigma)^2},$$

where γ is the scale parameter, and σ is the localization parameter, controlling the spread and median of the distribution, respectively. Without loss of generality, in the following discussions, we assume $\sigma = 0$, indicating independent and identically distributed noise. Cauchy noise exhibits several notable characteristics. It possesses heavier tails compared to Gaussian and Laplacian distributions, implying a higher probability of rare events. This feature results in its impulsive behavior, akin to genuine impulse noise. However, a crucial distinction lies in the impact: Cauchy noise affects all pixels in an image, while some pixels in impulse-corrupted images remain noise-free. The data fidelity term for the Cauchy noise removal model has the following form [13]: $\int_\Omega \log(\gamma^2 + (\mathbf{H}\mathbf{u} - \mathbf{f})^2)dx$. The data fidelity term is non-convex, posing challenges in finding the optimum solution. In recent years, researchers have proposed various solutions for the Cauchy noise removal problem. Authors in [14] introduced a TV-based variational method with convexity guaranteed by adding a quadratic penalty term:

$$\tilde{\mathbf{u}} = \underset{\mathbf{u}}{\operatorname{argmin}} \frac{\lambda}{2}\left(\int_\Omega \log(\gamma^2 + (\mathbf{u}-\mathbf{f})^2)dx + \nu||\mathbf{u}-\mathbf{u}_0||_2^2\right) + TV(\mathbf{u}), \tag{1}$$

where $\mathbf{u}_0$ is the image obtained by applying the median filter to the noisy image $\mathbf{f}$. The authors employed a median filter to partially remove Cauchy noise, preserving image details and maintaining convexity. The above model (1) can be readily generalized for restoring a blurred and corrupted image by incorporating the linear blur matrix $\mathbf{H}$:

$$\tilde{\mathbf{u}} = \underset{u}{\operatorname{argmin}} \frac{\lambda}{2}\left(\int_\Omega \log(\gamma^2 + (\mathbf{H}\mathbf{u}-\mathbf{f})^2)dx + \nu||\mathbf{H}\mathbf{u}-\mathbf{u}_0||_2^2\right) + TV(\mathbf{u}). \tag{2}$$

While the TV-based method has demonstrated significant effectiveness in removing Cauchy noise and preserving edges, it occasionally produces staircase artifacts or oversmoothing. The model (2) incorporates a regularization parameter λ to address this issue, controlling the trade-off between TV regularization and fitting to $\mathbf{f}$ and $\mathbf{u}_0$, with ν serving as a positive penalty parameter. It is worth noting that if $8\gamma^2\nu \geq 1$, the objective function becomes strictly convex [14, 15]. To overcome the TV-based method's limitations, some proposed solutions include replacing the original TV regularization with total generalized variation (TGV) regularization [16], combining TV and high-order TV [17, 18], using shearlet along with TGV regularization [19], and applying overlapping group sparsity with total variation (OGS-TV) [20, 21]. OGS-based methods have proven remarkably effective in various models, particularly in alleviating

undesirable artifacts and accurately reconstructing details [21]. In [22], the authors replaced the TV regularization in (2) with OGSTV, proposing a convex model based on total variation with OGS for recovering blurred images corrupted by Cauchy noise. This approach significantly enhances the Total Variation (TV) characteristics while mitigating the staircase effect, making it widely applicable for solving various types of image degradation. However, one challenge is that over-preserving details may diminish the effectiveness of noise reduction, potentially leading to the misidentification of thick noise ripples as false details. In [23], the authors introduced a variational model to remove Cauchy noise using the weighted nuclear norm (WNNM). Additionally, in [24], a Cauchy noise removal model based on the weighted hyper-Laplacian prior with OGS was proposed. Recently, a tensor recovery model with composite regularization was presented for Cauchy noise removal [25]. Moreover, in the pursuit of enhancing the restoration quality of directional images, researchers have explored Directional TV (DTV) regularization. DTV, in this context, offers increased and adjustable sensitivity to variations in a specified direction [26]. Various strategies have been employed for image noise removal using DTV [27, 28]. However, these directional regularization techniques have been primarily introduced for discretized problems, leaving the corresponding continuous problems unexplored. Additionally, it remains unclear in these previous works whether directional information can be incorporated into higher-order derivatives, such as through the use of a TGV regularizer. To address these gaps, authors in [29, 30] proposed a model based on directional total generalized variation (DTGV) functional to incorporate directional information. The DTGV-based models effectively remove noise and preserve edges based on the image's main direction, though the drawback of TV-based smoothing somewhat affects noise reduction. Recent studies have shown the promise of OGS-TV and DTGV methods, yet each has its limitations.

In this paper, we present an effective method for restoring images corrupted by Cauchy noise. The proposed model, named D-OGSTV, combines DTGV and OGSTV to leverage their strengths and address their drawbacks in the context of Cauchy noise removal. Successfully applied to images affected by Cauchy noise, our proposed model is mathematically formulated as D-OGSTV with a guaranteed convex objective function, ensuring the existence of an optimal solution. We employ a primal-dual algorithm to solve the minimization problem in our restoration model. Numerical results demonstrate that the proposed algorithm outperforms the compared methods for Cauchy noise removal. The rest of the paper is organized as follows: Section (2) lays out relevant preliminaries. Section (3) presents the proposed method. In Section (4), we show some numerical results of our proposed method and compare them with the results obtained using other existing methods. Finally, some conclusions are drawn in Section (5).

2 Preliminaries

This section provides an overview of the notations frequently used within the following discussion. Firstly, when applying total variation and higher-order total variation methods, the use of first and second-order finite differential matrices is essential. Let image be denoted as $\mathbf{u} \in \mathbb{R}^{m\times n}$ with periodic boundary conditions, i.e., $\mathbf{u}$ is periodically

extended [1]. The first-order difference operator is then defined as: $\nabla \mathbf{u} = \left(\nabla_x^+ \mathbf{u}, \nabla_y^+ \mathbf{u}\right)$, where $\nabla_x^+ \mathbf{u}$ and $\nabla_y^+ \mathbf{u}$ represent the first-order forward differences of $\mathbf{u}$ along the x-axis and y-axis, respectively, defined as follows:

$$\nabla_x^+ \mathbf{u}_{i,j} = \begin{cases} \mathbf{u}_{i,j+1} - \mathbf{u}_{i,j} & \text{if } 1 \le i \le m, 1 \le j < n, \\ \mathbf{u}_{i,1} - \mathbf{u}_{i,n} & \text{if } 1 \le i \le m, j = n, \end{cases} \quad \text{and}$$

$$\nabla_y^+ \mathbf{u}_{i,j} = \begin{cases} \mathbf{u}_{i+1,j} - \mathbf{u}_{i,j} & \text{if } 1 \le i < m, 1 \le j \le n, \\ \mathbf{u}_{1,j} - \mathbf{u}_{m,j} & \text{if } i = m, 1 \le j \le n. \end{cases}$$

Respectively, the first-order backward differences $\nabla_x^- \mathbf{u}$ and $\nabla_y^- \mathbf{u}$ are given by:

$$\nabla_x^- \mathbf{u}_{i,j} = \begin{cases} \mathbf{u}_{i,j} - \mathbf{u}_{i,j+1} & \text{if } 1 \le i \le m, 1 \le j < n, \\ \mathbf{u}_{i,1} - \mathbf{u}_{i,n} & \text{if } 1 \le i \le m, j = n, \end{cases} \quad \text{and}$$

$$\nabla_y^- \mathbf{u}_{i,j} = \begin{cases} \mathbf{u}_{i,j} - \mathbf{u}_{i+1,j} & \text{if } 1 \le i < m, 1 \le j \le n, \\ \mathbf{u}_{1,j} - \mathbf{u}_{m,j} & \text{if } i = m, 1 \le j \le n. \end{cases}$$

Building upon the previously defined first-order forward and backward differences, we construct the following second-order difference operator:

$$\nabla^2 \mathbf{u} = \begin{pmatrix} \nabla_x^-(\nabla_x^+ \mathbf{u}_{i,j}), & \nabla_y^+(\nabla_x^+ \mathbf{u}_{i,j}) \\ \nabla_x^+(\nabla_y^+ \mathbf{u}_{i,j}), & \nabla_y^-(\nabla_y^+ \mathbf{u}_{i,j}) \end{pmatrix}.$$

Overlapping group sparsity. For a one-dimensional signal, such as an audio signal $\mathbf{u} \in \mathbb{R}^n$, a K-point group of the signal is defined as follows [31]: $\mathbf{u}_{i,K} = [u(i), ..., u(i+K-1)] \in \mathbb{R}^K$, where $u_{i,K}$ can be viewed as a window of size K, where any element with an index higher than n is set to 0 [20]. In [21], the authors explored the following group sparse regularization function:

$$\phi(\nabla \mathbf{u}) = \sum_{i=0}^{n-1} \left[\sum_{k=0}^{K} |\nabla \mathbf{u}_{i+k}|^2 \right] = \sum_{i=0}^{n-1} ||\nabla \mathbf{u}||_2, \tag{3}$$

where $\nabla \in \mathbb{R}^{(n-1)\times n}$ represents the first-order differential matrix or gradient of the column signal $\mathbf{u} \in \mathbb{R}^n$. Note that the above function applies the overlapping sparsity method in conjunction with total variation for one-dimensional signal denoising, with the $\phi(\cdot)$ regularization taking $\nabla \mathbf{u}$ as the input parameter. Additionally, if the group size K is chosen to be $K = 1$, the regularization function above becomes the well-known l_1-norm regularizer, i.e., $\phi(\nabla \mathbf{u}) = ||\nabla \mathbf{u}||_1$. For a two-dimensional signal, such as an image $\mathbf{u} \in \mathbb{R}^{m\times n}$ with entries $u(i,j)$, the $K \times K$ sized group is defined as follows [32]:

$$\tilde{\mathbf{u}}_{i,j,K} = \begin{bmatrix} u_{i-m_1,j-m_1} & u_{i-m_1,j-m_1+1} & \dots & u_{i-m_1,j+m_2} \\ u_{i-m_1+1,j-m_1} & u_{i-m_1+1,j-m_1+1} & \dots & u_{i-m_1+1,j+m_2} \\ \dots & \dots & \dots & \dots \\ u_{i+m_2,j-m_1} & u_{i+m_2,j-m_1+1} & \dots & u_{i+m_2,j+m_2} \end{bmatrix},$$

with $m_1 = \lfloor \frac{K-1}{2} \rfloor$ and $m_2 = \lfloor \frac{K}{2} \rfloor$, where $\lfloor u \rfloor$ denotes the largest integer smaller than u. To apply the above group sparse regularization function $\phi(\cdot)$, we flatten the $K \times K$ sized group into a vector of size K^2, i.e., $\tilde{\mathbf{u}}_{i,j,K}(:) = \mathbf{u}_{i,j,K}$, and we can express the group sparse regularizer as follows [20]: $\phi(\nabla \mathbf{u}) = \sum_{i=0}^{n-1} ||\nabla \mathbf{u}||_2$. Note that, considering the

input signal is two-dimensional, the gradient ∇ must consist of ∇_x and ∇_y, representing both x-axis and y-axis differentiation. As denoted in 2.1, the first-order differential matrix of $\mathbf{u}$ is $\nabla\mathbf{u} = (\nabla_x\mathbf{u}, \nabla_y\mathbf{u})$.

Directional total generalized variation method. Given an image $\mathbf{u} \in \mathbb{R}^{N\times N}$, the second-order DTGV for $\mathbf{u}$ is defined as follows [29]:

$$DTGV^2(\mathbf{u}) = \min_{\mathbf{w}} \alpha_0 \|\tilde{\nabla}\mathbf{u} - \mathbf{w}\|_{2,1|\mathbb{R}^{2n}} + \alpha_1 \|\tilde{\mathcal{E}}\mathbf{w}\|_{2,1|\mathbb{R}^{4n}}, \tag{4}$$

where $\mathbf{w} \in \mathbb{R}^{2N\times N}$, $\tilde{\nabla} \in \mathbb{R}^{2N\times N}$ is the discrete gradient operator, and $\tilde{\varepsilon}$ is the directional symmetrized derivative of the tensor $\mathbf{w}$, with $\alpha_0, \alpha_1 \in (0, +\infty)$. For any vector $\mathbf{v} \in \mathbb{R}^{2N}$, we define: $\|\mathbf{v}\|_{2,1|\mathbb{R}^{2n}} = \sum_{j=1}^{n} \sqrt{v_j^2 + v_{n+j}^2}$, for any vector $\mathbf{y} \in \mathbb{R}^{4N}$, we define: $\|\mathbf{y}\|_{2,1|\mathbb{R}^{4n}} = \sum_{j=1}^{n} \sqrt{y_j^2 + y_{n+j}^2 + y_{2n+j}^2 + y_{3n+j}^2}$. Given $\theta \in [-\pi, \pi]$ and a scaling parameter $a > 0$, the discrete gradient operator is defined as: $\nabla\mathbf{u} = \begin{bmatrix}\nabla_x^+\mathbf{u} \\ \nabla_y^+\mathbf{u}\end{bmatrix}$, where θ and $\theta^\perp = \theta + \frac{\pi}{2}$. The operators $\nabla_x^+ u$ and $\nabla_y^+ u \in \mathbb{R}^{N\times N}$ can be obtained by applying a forward finite difference scheme along each axis of the image.

$$(\nabla_x^+\mathbf{u})_{i,j} = \begin{cases} u_{i+1,j} - u_{i,j} \;\; if \;\; i < N, \\ 0 \;\; if \;\; i = N, \end{cases} \quad \text{and} \quad (\nabla_y^+\mathbf{u})_{i,j} = \begin{cases} u_{i,j+1} - u_{i,j} \;\; if \;\; i < N, \\ 0 \;\; if \;\; i = N. \end{cases}$$

The divergence operator is defined as $div = -\nabla = (\nabla_x^-, \nabla_y^-)$ where ∇_x^- and ∇_y^- represent the backward finite difference schemes. Moreover, according to [30], the directional divergence for tensor $\mathbf{v}$ can be obtained by:$(\tilde{div}\mathbf{v}) = div\tilde{\mathbf{v}}$, where $\tilde{\mathbf{v}} = \Lambda_a R_\theta \mathbf{v}$. The associated directional gradient operator is $\tilde{\nabla}\mathbf{u} = \Lambda_a R_{-\theta}(\nabla\mathbf{u})$. Here, R_θ represents the rotation matrix $R_\theta = \begin{bmatrix}\cos\theta & -\sin\theta \\ \sin\theta & \cos\theta\end{bmatrix}$, and Λ_a is the translation matrix: $\Lambda_a = \begin{bmatrix}1 & 0 \\ 0 & a\end{bmatrix}$. The directional symmetrized derivative of the tensor $\mathbf{w}$ can be defined as:

$$(\tilde{\mathcal{E}}\mathbf{w})_{i,j} = \frac{1}{2}\left[\Lambda_a R_\theta \begin{bmatrix}(\nabla_x^- w^1)_{i,j} & (\nabla_x^- w^2)_{i,j} \\ (\nabla_y^- w^1)_{i,j} & (\nabla_y^- w^2)_{i,j}\end{bmatrix} + \begin{bmatrix}(\nabla_x^- w^1)_{i,j} & (\nabla_y^- w^1)_{i,j} \\ (\nabla_x^- w^2)_{i,j} & (\nabla_y^- w^2)_{i,j}\end{bmatrix} R_\theta^T \Lambda_a\right].$$

with $\tilde{div} = -\tilde{\mathcal{E}}$.

3 Proposed method

Based on (3) and (4), we propose the regularizer as follows:

$$\text{D-OGSTV}(\mathbf{u}) = \min_{\mathbf{w}} \alpha_0 \|\Lambda_a R_{-\theta}\phi(\nabla\mathbf{u}) - \mathbf{w}\|_{2,1|\mathbb{R}^{2n}} + \alpha_1 \|\tilde{\mathcal{E}}\mathbf{w}\|_{2,1|\mathbb{R}^{4n}}. \tag{5}$$

In this paper, we combine our regularization model D-OGSTV($\mathbf{u}$) (5) with the data fidelity term for Cauchy noise in (2). The combined model is given as follows:

$$\tilde{\mathbf{u}} = \underset{\mathbf{u}}{\operatorname{argmin}} \frac{\lambda}{2}\left(\int_\Omega \log(\gamma^2 + (\mathbf{Hu} - \mathbf{f})^2)dx + \nu||\mathbf{Hu} - \mathbf{u}_0||_2^2\right) + \text{D-OGSTV}(\mathbf{u}). \tag{6}$$

To solve the optimization problem for the proposed model (6), we can use many different algorithms, typically the Nesterov method [33], ADMM [34], FISTA algorithm [35], primal-dual (TV-L1) [36]. In this article, we decide to employ the Chambolle-Pock dual algorithm to optimize the model. Based on [36, 37], our proposed regularizer (5) can be transformed to the dual form as follows:

$$\min_{u\in\mathbb{R}^n, w\in\mathbb{R}^{2n}} \max_{p\in\mathcal{P}, v\in\mathcal{W}} \langle \Lambda_a R_\theta \phi(\nabla \mathbf{u}) - \mathbf{w}, \mathbf{p}\rangle + \langle \tilde{\mathcal{E}}\mathbf{w}, \mathbf{v}\rangle,$$

where $\mathcal{P} = \{\mathbf{p} : \Omega \rightarrow \mathbb{R}^{\mathbf{2}} \mid ||\mathbf{p_{i,j}}||_{\mathbf{2}} \leq \lambda_1 \ \forall(\mathbf{i}, \mathbf{j}) \in \Omega\}$, and $\mathcal{W} = \{\mathbf{v} : \Omega \rightarrow \mathbf{Sym^2}(\mathbb{R}^{\mathbf{2}}) | \ ||\mathbf{v}_{i,j}||_{\mathbf{F}} \leq \lambda_0 \ \forall(\mathbf{i}, \mathbf{j}) \in \Omega\}$. This is a saddle point problem, and we can use the Chambolle-Pock algorithm to solve it. In this algorithm, we break down the entire problem into subproblems for resolution. We define the set-projection operator as follows [36, 37]: $[S_\lambda(\mathbf{x})]_{i,j} = \frac{\mathbf{x}_{i,j}}{\max\left(1, \frac{\mathbf{x}_{i,j}}{\lambda}\right)}$. Here, if $x \in \mathcal{P}$, then $|x_{i,j}|$ is the l_2 norm. If $x \in \mathcal{W}$, then $|x_{i,j}|$ is the Frobenius norm. In each subproblem, we use set-projection to solve them, which aligns with the essence of the Chambolle-Pock algorithm. First, we construct the duality problem:

$$\min_{\mathbf{u},\mathbf{w}} \max_{\mathbf{p},\mathbf{q},\mathbf{v}} \ \langle \log(\gamma^2 + (\mathbf{Hu} - \mathbf{f})^2) + \nu \, ||\mathbf{Hu} - \mathbf{u}_0||_2^2, \mathbf{q}\rangle + \lambda_1 \langle (\Lambda_a R_\theta \phi(\nabla \mathbf{u}) - \mathbf{w}), \mathbf{p}\rangle + \lambda_0 \langle \widetilde{\mathcal{E}}\mathbf{w}, \mathbf{v}\rangle. \tag{7}$$

The Chambolle-Pock algorithm is employed alongside the set-projection operator to address the regularization problem (7) by dividing it into smaller subproblems:

$$\begin{cases} \mathbf{p}^{k+1} = \underset{\mathbf{p}}{\text{argmax}} \ \lambda_1 \langle \Lambda_a R_{-\theta} \phi(\nabla \mathbf{u}^k) - \bar{\mathbf{w}}^{\mathbf{k}}, \mathbf{p}\rangle - \frac{1}{2\eta}\|\mathbf{p} - \mathbf{p}^k\|_2^2, \\ \mathbf{v}^{k+1} = \underset{\mathbf{v}}{\text{argmax}} \ \lambda_0 \langle \tilde{\mathcal{E}}\bar{\mathbf{w}}^k, \mathbf{v}\rangle - \frac{1}{2\eta}\|\mathbf{v} - \mathbf{v}^k\|_2^2, \\ \mathbf{q}^{k+1} = \underset{\mathbf{q}}{\text{argmax}} \ \langle \log(\gamma^2 + (\mathbf{H}\bar{\mathbf{u}}^k - \mathbf{f})^2) + \nu||\mathbf{H}\bar{\mathbf{u}}^k \\ \qquad -u_0||_F^2, \mathbf{q}\rangle - \frac{1}{2\eta}\|\mathbf{q} - \mathbf{q}^k\|_2^2, \\ \mathbf{u}^{k+1} = \underset{\mathbf{u}}{\text{argmin}} \ \langle \log(\gamma^2 + (\mathbf{Hu} - \mathbf{f})^2) + \nu||\mathbf{Hu} - \mathbf{u}_0||_2^2, \mathbf{q}^{k+1}\rangle \\ \qquad +\langle \Lambda_a R_{-\theta}\phi(\nabla \mathbf{u}), \mathbf{p}^{k+1}\rangle + \frac{1}{2\tau}\|\mathbf{u} - \mathbf{u}^k\|_2^2, \\ \mathbf{w}^{k+1} = \underset{\mathbf{w}}{\text{argmin}} \ -\langle \mathbf{w}, \mathbf{p}^{k+1}\rangle + \langle \tilde{\mathcal{E}}\mathbf{w}^k, \mathbf{v}^{k+1}\rangle + \frac{1}{2\tau}\|\mathbf{w} - \mathbf{w}^k\|_2^2. \end{cases} \tag{8}$$

Our algorithm needs to go through many loops for the result to converge. We define $\bar{\mathbf{u}}$ and $\bar{\mathbf{w}}$ as the previous loop's $\mathbf{u}$ and $\mathbf{w}$, respectively. In the first loop, we initialize $\bar{\mathbf{w}} = \mathbf{w}$ and $\bar{\mathbf{u}} = \mathbf{u}$. Next, we are going to solve the subproblems in (8).

For the subproblems involving $\mathbf{p}$, the set-projection operator is employed to find the root of these problems:

$$\mathbf{p}^{k+1} = \mathcal{S}_{\lambda_1}\left(\mathbf{p}^k + \eta\left(\Lambda_a R_{-\theta}\phi(\nabla \bar{\mathbf{u}}^k) - \bar{\mathbf{w}}^k\right)\right) = \frac{\|\mathbf{p}^k + \eta\left(\Lambda_a R_{-\theta}\phi(\nabla \bar{\mathbf{u}}^k) - \bar{\mathbf{w}}^k\right)\|}{\max\left(1, \frac{\|\mathbf{p}^k + \eta\left(\Lambda_a R_{-\theta}\phi(\nabla \bar{\mathbf{u}}^k) - \bar{\mathbf{w}}^k\right)\|}{\lambda_1}\right)}. \tag{9}$$

Similar to $\mathbf{v}$, $\mathbf{q}$, we have:

$$\mathbf{v}^{k+1} = \mathcal{S}_{\lambda_0}\left(\mathbf{v}^k + \eta\tilde{\mathcal{E}}\bar{\mathbf{w}}^k\right) = \frac{\|\mathbf{v}^k + \eta\tilde{\mathcal{E}}\bar{\mathbf{w}}^k\|}{\max\left((1, \frac{\|\mathbf{v}^k+\eta\tilde{\mathcal{E}}\bar{\mathbf{w}}^k\|}{\lambda_0}\right)}, \tag{10}$$

$$\mathbf{q}^{k+1} = \frac{\|\mathbf{q}^k + \eta\left(\log(\gamma^2 + (\mathbf{H}\bar{\mathbf{u}}^k - \mathbf{f})^2) + \nu||\mathbf{H}\bar{\mathbf{u}}^k - u_0||_F^2\right)}{\max\left(1, \|\mathbf{q}^k + \eta\left(\log(\gamma^2 + (\mathbf{H}\bar{\mathbf{u}}^k - \mathbf{f})^2) + \nu||\mathbf{H}\bar{\mathbf{u}}^k - \mathbf{u}_0||_2^2\right)\right)}. \tag{11}$$

The minimization can be solved by applying the Gradient Descent method to update the variables $\mathbf{u}$ iteratively:

$$\mathbf{u}^{k+1} = \mathbf{u}^k + \tau\left(\tilde{\text{div}}\mathbf{p}^{k+1} - \left(\frac{2\mathbf{H}^\top(\mathbf{H}\mathbf{u}^k - \mathbf{f})}{\gamma^2 + (\mathbf{H}\mathbf{u}^k - \mathbf{f})^2} - 2\nu\mathbf{H}^\top\|\mathbf{H}\mathbf{u}^k - \mathbf{u}_0\|\right)\mathbf{q}^{k+1}\right), \tag{12}$$

where $\tilde{div}$ is the directional divergence operator, calculated as $\Lambda_a R_{-\theta}\text{div}$.

The same procedure applies to $\mathbf{w}$:

$$\mathbf{w}^{k+1} = \mathbf{w}^k + \tau\left(\tilde{\text{div}}\mathbf{v}^{k+1} + \mathbf{p}^{k+1}\right). \tag{13}$$

In summary, the algorithm for our model D-OGSTV for cauchy noise is given in **Algorithm 1**.

Algorithm 1: Algorithm for solving the model D-OGSTV (6)

1. **Initialize**: $\mathbf{p} = 0, \mathbf{q} = 0, \mathbf{v} = \mathbf{0}, \mathbf{u} = \bar{\mathbf{u}} = 0, \mathbf{w} = \bar{\mathbf{w}} = 0$, τ, η, $\mathbf{H}$;
2. **While** Stopping condition is not satisfied **do**
3. -Compute $\mathbf{v}^{k+1}$ using (10).
4. -Compute $\mathbf{q}^{k+1}$ using (11).
5. -Compute $\mathbf{u}^{k+1}$ using (12).
6. -Compute $\mathbf{w}^{k+1}$ using (13).
7. -Update $\bar{\mathbf{u}}^{k+1} = 2\mathbf{u}^{k+1} - \mathbf{u}^k$.
8. -Update $\bar{\mathbf{w}}^{k+1} = 2\mathbf{w}^{k+1} - \mathbf{w}^k$.
9. -$k = k + 1$.
10. **End while**
11. **Return u**.

The stopping condition of the algorithm can be determined based on the maximum number of allowed outer iterations N has been carried out (to guarantee an upper bound on running time) or the following condition is satisfied for some prescribed tolerance ς:

$$\frac{\|\mathbf{u}^k - \mathbf{u}^{k-1}\|_2}{\|\mathbf{u}^k\|_2} < \varsigma,$$

where ς is a small positive parameter. For our experiments, we set tolerance $\varsigma = 0.0001$ and $N = 300$.

4 Experimental results

In this section, we will experimentally verify the effectiveness of the proposed model on 10 sample images, as shown in Figure (1). The sample images used are in grayscale with a size of 256×256. To test the denoising procedure, we simulate Cauchy noise with noise levels $\gamma = 0.02$ and $\gamma = 0.05$. A larger γ value indicates denser noise. Specifically, we simulate Cauchy noise of level γ using the following formula:

$$\mathbf{f} = \mathbf{u} + \gamma \frac{\zeta_1}{\zeta_2}, \tag{14}$$

where $\mathbf{u}$ is our free-noise sample image, ζ_1, ζ_2 are random matrices following normal distribution $\mathcal{N}(0, 1)$. For the denoising problem, blur operator $\mathbf{H}$ is identity matrix $\mathbf{I}$. Experimental results are evaluated using visual methods along with standard PSNR and SSIM scales [38]. The PSNR value determines the quality of the restored image compared to the original image, higher PSNR value means the result image is more closely resemble the original image. The mathematical representation of the PSNR is as below:

$$\text{PSNR} = 20 \log_{10} \left(\frac{\max_{\text{px}}}{\sqrt{\text{MSE}}} \right),$$

where $\max_{\text{px}}$ is the maximum pixel value; the MSE (Mean Squared Error) is calculated as:

$$\text{MSE} = \frac{1}{mn} \sum_{0}^{m-1} \sum_{0}^{n-1} ||u_{i,j} - f_{i,j}||^2.$$

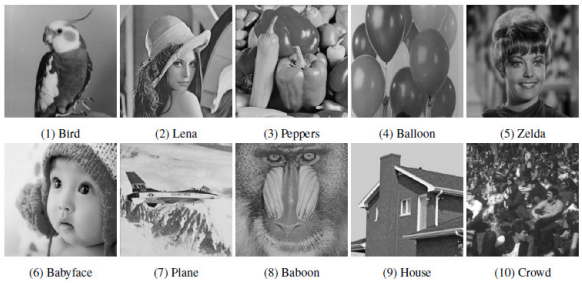


Figure 1: Test images

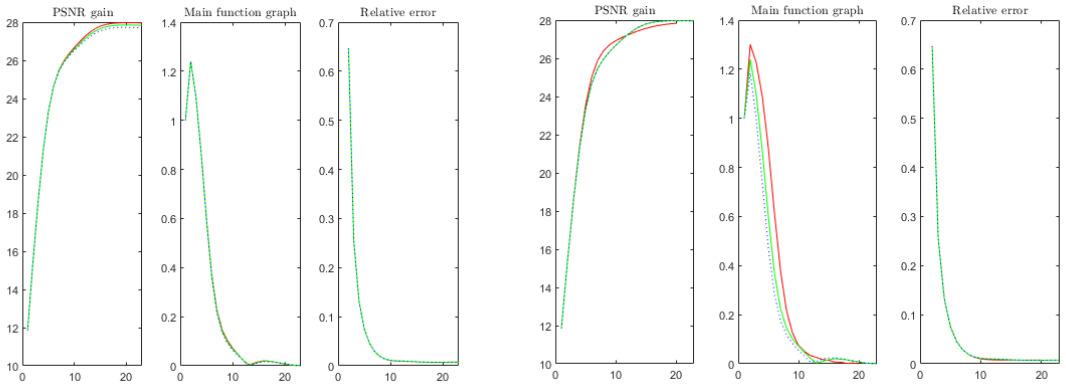
Besides, the SSIM scale determines the structural similarity between the restored image and the original image, which has its value in the range [0,1]. The closer the SSIM value of two images is to 1, means the more similar the two images are. The SSIM index is calculated on various windows of the image pair. The measurement between two images $\mathcal{A}$ and $\mathcal{B}$ is:

$$\text{SSIM}(\mathcal{A}, \mathcal{B}) = \frac{(2\mu_{\mathcal{A}}\mu_{\mathcal{B}} + c_1)(2\sigma_{\mathcal{AB}} + c_2)}{(\mu_{\mathcal{A}}^2 + \mu_{\mathcal{B}}^2 + c_1)(\sigma_{\mathcal{A}}^2 + \sigma_{\mathcal{B}}^2 + c_2)}, \tag{15}$$

where $\mu_{\mathcal{A}}, \mu_{\mathcal{B}}, \sigma_{\mathcal{A}}, \sigma_{\mathcal{B}}$ are the means and standard deviations of $\mathcal{A}$ and $\mathcal{A}$ respectively, $\sigma_{\mathcal{AB}}$ is the cross-covariance for $\mathcal{A}$ and $\mathcal{B}$, and c_1, c_2 are parameters that depends on the pixels value range of the image.

Parameter analysis

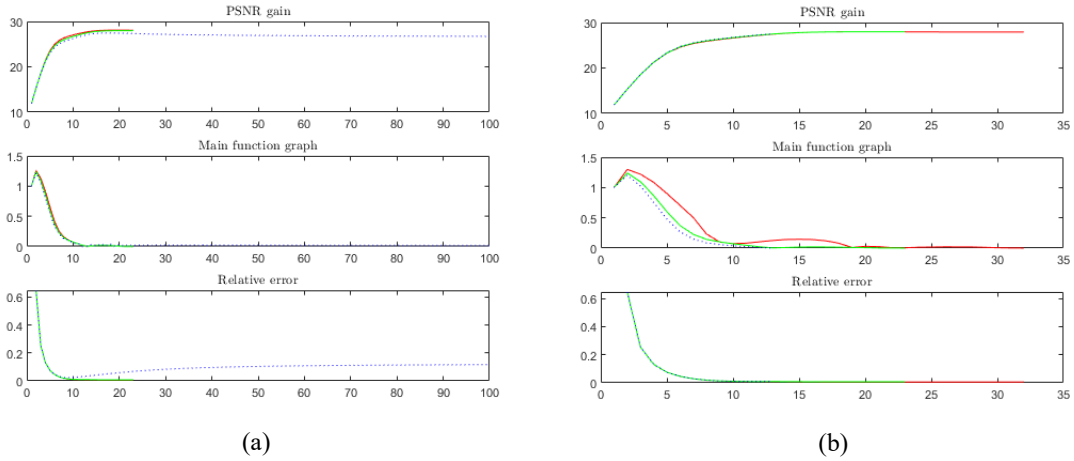
Through experimental methods, we investigate the influence of the parameters included in the aforementioned proposed algorithm on denoising results, considering resulting image quality, execution time (number of iterations), and convergence speed. We conducted tests on the same noisy image (Figure (1)-(1) Bird with $\gamma = 0.02$) using different parameter sets, and present the following graphs: PSNR value gain graph, objective function graph, and relative error graph after each iteration step. Consequently, the best set of parameters can be chosen for conducting further experiments.



(a) (b)

Figure 2: Influence of the parameters: (a) $\lambda_0 = 0.6$ (red), $\lambda_0 = 0.8$ (green), $\lambda_0 = 1.0$ (blue); (b) $\lambda_1 = 5$ (red), $\lambda_1 = 15$ (green), $\lambda_1 = 25$ (dotted blue)

Figure (2)(a) illustrates that variations in λ_0 have a limited impact on the number of iterations and convergence speed of the objective function and solution. Nevertheless, it does affect the PSNR (Peak Signal-to-Noise Ratio) of the output, with the highest PSNR result (red line) observed at $\lambda_0 = 0.6$. Figure (2)(b) reveals that, with a λ_1 value of 5, the objective function converges more slowly, but the PSNR growth is slightly faster, reaching a good solution earlier. Similarly, the green line with $\lambda_1 = 15$ demonstrates faster convergence of the objective function and a high PSNR, achieving early results. However, the blue line representing $\lambda_1 = 25$, despite the fastest initial convergence, experiences slower convergence in later stages, leading to the highest number of iterations. Figure (3)(a) illustrates the impact of the parameter a on the model. Comparing the red and green lines, which correspond to $a = 0.7$ and $a = 1$ respectively, the red line shows a higher PSNR value result, while maintaining the same number of iterations. Regarding the blue line, where $a = 1.3$, the model becomes non-convex, and no optimum solution can be found. Through further experimentation, we determined that the safe threshold for the parameter a is around 1.2. When $a > 1.2$, the non-convex problem emerges. Next, we consider the parameter K as the window size of the OGS method. From Figure (3)(b), it is evident that with a window size of $K = 1$ or in other words, without applying OGS, the number of iterations increases, and the convergence of the objective function appears unstable. Thus, utilizing the OGS method helps accelerate mathematical convergence. For $K = 3$ and $K = 5$, the convergence speed improves significantly. However, at $K = 5$, we observe a decrease



(a) (b)

Figure 3: Influence of the parameters: (a) $a = 0.7$ (red), $a = 1$ (green), $a = 1.3$ (dotted blue); (b) $K = 1$ (red), $K = 3$ (green), $K = 5$ (dotted blue)

in PSNR even though the number of iterations is quite low. Therefore, to ensure good results in a short computational time, we consider choosing $K = 3$. In the following tests, we set the group size K to 3, the regularization parameters to $\lambda_0 = 0.6$ and $\lambda_1 = 15$, and the translation parameter $a = 0.6$. As mentioned in the proposed algorithm, we can use either the differentiation of the solution after each iteration or the objective function itself as the stopping criteria. The latter option is applied in the test.

To further evaluate the proposed model's efficacy, we compare it with two established methods: the TV-L1 model and the WNNM model for Cauchy noise removal. We implemented a demo program using MATLAB on a Windows 11 system equipped with an Intel Core i5 CPU and 16GB RAM. We compare our approach with the aforementioned models based on PSNR, SSIM, and computational time, providing valuable insights into its performance and advantages.

Image denoising

To visually assess the performance of the three methods, we present original, noisy, and denoised images from each in Figures (4), (5), (6), (7). While TV-L1 effectively eliminates Cauchy noise, it often results in over-smoothed images with compromised details, uniform color blocks, and pronounced staircase effects.

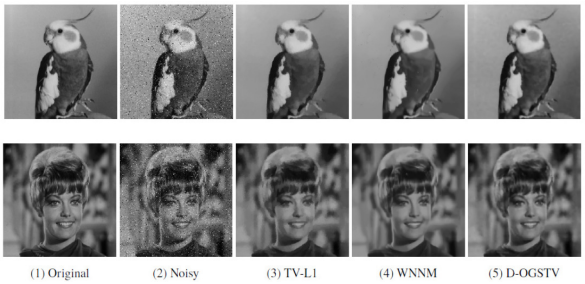


Figure 4: Denoising result of different methods on test images with noise level $\gamma = 0.02$

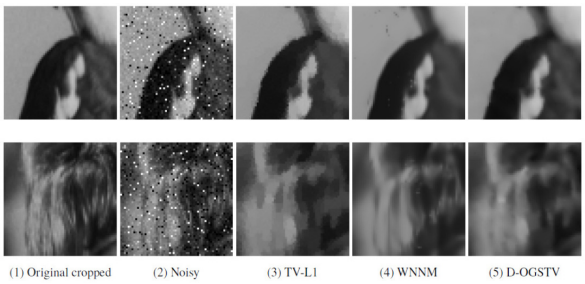


Figure 5: Zoom-in denoising results in Figure (4)

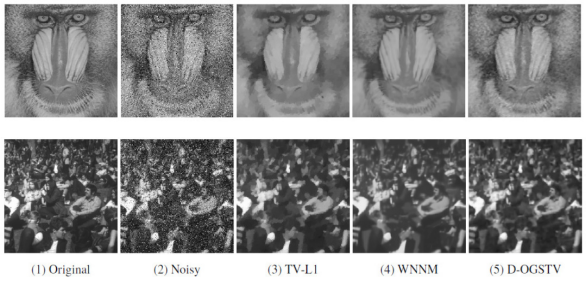


Figure 6: Denoising result of different methods on test images with noise level $\gamma = 0.05$

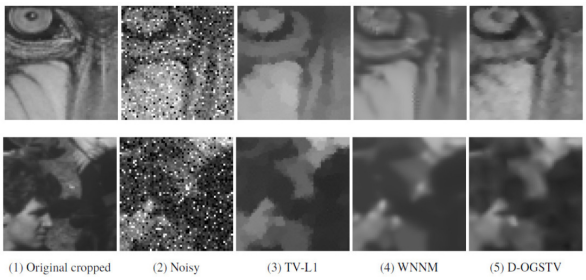


Figure 7: Zoom-in denoising results in Figure (6)

Table (1) provides a comparison of the average execution time required for each method, both for a single test image and a patch of 10 images, with a noise level of $\gamma = 0.05$. As anticipated, the TV-L1 method excels in terms of speed, requiring only 0.15 seconds per image and a mere 1.38 seconds for the entire set. In contrast, WNNM emerges as the slowest, demanding a substantial 30 seconds per image and 5.5 minutes for the full patch. This prolonged runtime is attributed to WNNM's necessity to calculate singular value decompositions (SVDs) for every patch in each iteration. Our proposed method falls between these extremes. While the use of iterative OGS calculations on small windows results in a higher execution time than TV-L1, it remains significantly lower than that of WNNM. Importantly, this relatively modest increase in complexity does not compromise the model's ability to deliver impressive denoising results.

In Tables (2) and (3), we present the PSNR and SSIM values of noisy images, as well as the result images obtained using the three methods: TV-L1, WNNM, and our proposed method, D-OGSTV. The best PSNR and SSIM values for each image

Table 1: Table of computational time comparison (s)

Time	**TV-L1**	**WNNM**	**D-OGSTV**
For single image (1)	0.1500	30.6219	5.8559
For entire set	1.3842	334.3625	62.1075

are marked in bold. Based on the measurement results, it can be seen that the noise reduction model we proposed has demonstrated competitive performance against the other two models at both noise levels. Even at a higher noise level of $\gamma = 0.05$, our method maintains its stability in removing Cauchy noise.

Table 2: PSNR and SSIM of noisy and restored images by compared methods with $\gamma = 0.02$

Image	**Noisy**		**TV_L1**		**WNNM**		**D-OGSTV**	
	PSNR	**SSIM**	**PSNR**	**SSIM**	**PSNR**	**SSIM**	**PSNR**	**SSIM**
Bird	19.105	0.2346	33.178	0.9037	34.025	0.9166	**35.130**	**0.9277**
Lena	19.028	0.3631	28.076	0.8235	28.554	0.8468	**29.486**	**0.8702**
Peppers	19.033	0.3479	29.990	0.8792	30.720	0.9087	**31.593**	**0.9197**
Balloon	19.134	0.2308	**33.135**	0.9000	32.469	0.8917	32.903	**0.9008**
Zelda	19.189	0.3199	31.220	0.8811	31.575	0.8797	**32.603**	**0.9055**
Babyface	19.251	0.3127	29.667	0.8524	30.260	0.8902	**31.616**	**0.9096**
Plane	19.054	0.2522	28.022	0.7965	27.728	0.8722	**28.758**	**0.8924**
Baboon	19.104	0.4026	26.876	0.6681	26.263	0.6249	**27.110**	**0.7026**
House	19.066	0.3328	28.933	0.7930	29.448	0.8144	**30.111**	**0.8376**
Crowd	19.144	0.4678	25.168	0.7402	26.907	0.8280	**28.256**	**0.8706**

Table 3: PSNR and SSIM of noisy and restored images by compared methods with $\gamma = 0.05$

Image	**Noisy**		**TV_L1**		**WNNM**		**D-OGSTV**	
	PSNR	**SSIM**	**PSNR**	**SSIM**	**PSNR**	**SSIM**	**PSNR**	**SSIM**
Bird	15.267	0.1159	30.195	0.8818	31.581	0.8788	**32.118**	**0.8819**
Lena	15.355	0.2129	27.098	0.8116	26.963	0.7914	**27.497**	**0.8140**
Peppers	15.314	0.2035	26.939	0.8417	28.225	0.8531	**28.910**	**0.8767**
Balloon	15.359	0.1133	30.753	**0.8723**	30.716	0.8510	**30.915**	0.8589
Zelda	15.385	0.1557	28.524	0.8268	29.273	0.8127	**30.436**	**0.8593**
Babyface	15.301	0.1622	27.570	0.8533	28.435	0.8318	**29.399**	**0.8686**
Plane	15.388	0.2201	26.546	**0.8375**	26.106	0.8210	**26.980**	0.8369
Baboon	15.328	0.2138	25.405	0.5710	25.371	0.5585	**26.000**	**0.6324**
House	15.315	0.1950	**28.286**	**0.7935**	27.856	0.7794	28.098	0.7813
Crowd	15.372	0.2843	24.583	0.7406	25.176	0.7564	**26.340**	**0.7906**

The WNNM method demonstrates a degree of inconsistency in its results. While it excels in noise removal and detail preservation in some test cases, it exhibits incomplete denoising with residual Cauchy noise artifacts in others, even under identical noise levels and parameters. This inconsistency extends to different images within a batch, with highly detailed images being more susceptible to incomplete noise reduction. Attempts to mitigate this by increasing the σ parameter often result in over-smoothing of other

images. This suggests that, while WNNM can achieve impressive results, its sensitivity to image structure and noise distribution necessitates individualized parameter tuning, hindering its practical application in batch processing scenarios. Our D-OGSTV method emerges as a compelling alternative, outperforming TV-L1 by preserving details and avoiding staircase artifacts. Additionally, it achieves PSNR and SSIM values on par with or superior to WNNM, while offering significantly faster execution and remarkable stability compared to its WNNM counterpart.

Image denoising and deblurring

In real-world image acquisition, environmental factors and signal imperfections often lead to the simultaneous presence of noise and blur. This section addresses this challenge by proposing a model that performs joint image denoising and deblurring. During image degradation simulation, we apply a Gaussian blur filter with a window size of 5x5 and a standard deviation of 1. Subsequently, Cauchy noise with a level of $\gamma = 0.02$ is added to the blurred image. For evaluating the effectiveness of our model in the combined task of denoising and deblurring, we compare the results with the three models TV-L1, WNNM, and our model D-OGSTV, similar to the above experiment. Figures (8), (9) showcase the original images, noisy-and-blurred images, and the reconstructions obtained by our method and two baseline approaches (WNNM, TV-L1, and D-OGSTV).

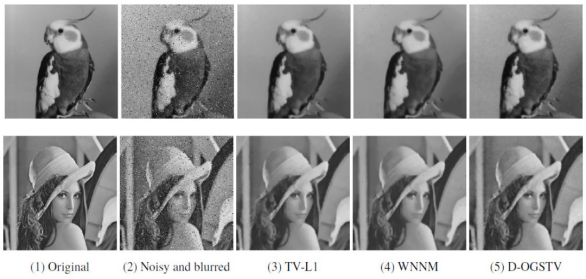


Figure 8: Denoising and deblurring result of different methods on test images with blur and noise level $\gamma = 0.02$

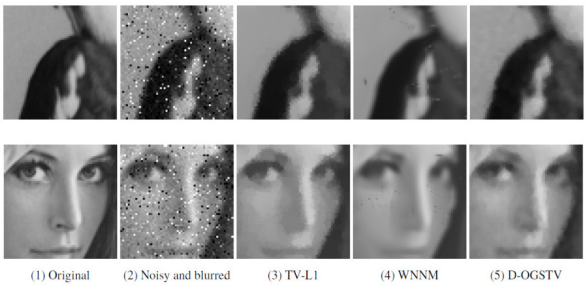


Figure 9: Zoom-in denoising and deblurring results in Figure (8)

These figures facilitate visual comparison. As observed, WNNM cannot effectively address the deblurring problem, resulting in blurry reconstructions. While both TV-L1 and D-OGSTV performed well on blurred images, our model consistently delivers superior results in most cases. In Tables (4), we present the PSNR and SSIM values of

noisy-and-blurred images, along with the recovered result images using the compared methods TV-L1, WNNM, and D-OGSTV. The best PSNR and SSIM values for each image are marked in bold. The table reveals that blurring significantly impacts the process, reducing the overall PSNR values of the result images. Despite this challenge, our proposed model achieves better results in most test images.

Table 4: PSNR and SSIM of noisy-and-blurred images and restored images by different methods, $\gamma = 0.02$

Image	**Noisy**		**TV_L1**		**WNNM**		**D-OGSTV**	
	PSNR	**SSIM**	**PSNR**	**SSIM**	**PSNR**	**SSIM**	**PSNR**	**SSIM**
Bird	19.063	0.2141	32.266	**0.8980**	30.994	0.8747	**32.728**	0.8925
Lena	18.730	0.3002	27.213	0.8093	26.612	0.7877	**27.728**	**0.8273**
Peppers	18.814	0.3173	28.407	0.8732	27.483	0.8501	**29.056**	**0.8849**
Balloon	18.996	0.1982	31.433	**0.8718**	30.864	0.8586	**31.689**	0.8688
Zelda	19.023	0.2804	30.508	0.8514	29.807	0.8477	**31.221**	**0.8740**
Babyface	18.973	0.2737	29.091	0.8622	28.076	0.8193	**29.948**	**0.8791**
Plane	18.738	0.3087	26.240	0.8332	25.159	0.7885	**26.966**	**0.8475**
Baboons	18.674	0.2670	25.842	0.5854	25.104	0.6183	**26.525**	**0.6457**
House	18.925	0.2791	27.687	0.7840	27.193	0.7581	**28.107**	**0.7984**
Crowd	18.770	0.3879	25.529	0.7757	24.142	0.7008	**26.312**	**0.8133**

5 Conclusion

In this paper, we have successfully combined Directional Total Generalized Variation (DGTV) with Overlapping Group Sparsity and Total Variation (OGSTV) to develop the D-OGSTV Cauchy denoising model. This novel model exhibits exceptional noise removal capabilities, even at relatively high noise levels, while preserving fine details and realistic edges in restored images. Experimental results confirm that D-OGSTV significantly outperforms traditional Total Variation (TV) and high-order TV methods, both visually and quantitatively. Implementing this model has the potential to greatly enhance the quality of images acquired in various information systems. Moving forward, we will continue to refine the D-OGSTV model to tackle denoising tasks with even higher noise levels. Additionally, we plan to investigate methods for predicting the noise level parameter γ directly from noisy images. This advancement would enable the training of machine learning models that can automatically adapt and remove noise across a series of images, streamlining the noise removal process.

Acknowledgement

Pham Cong Thang (corresponding author) would like to thank his colleagues at IT Faculty, DUT, Danang, Vietnam, for their helpful comments. This work was supported by The University of Danang–University of Science and Technology, code number of Project: T2023–02–05MSF.

References

[1] Aubert G., Pierre K. *Mathematical Problems in Image Processing: Partial Differential Equations and the Calculus of Variations*, Applied Mathematical Science 147. 2nd ed. New York, USA: Springer-Verlag, 2008, 379 pages. https://doi.org/10.1007/978-0-387-44588-5

[2] Pham C.T., Tran T.T.T., Gamard G. *An Efficient Total Variation Minimization Method for Image Restoration*. Informatica, 2020, vol. 31(3), pp. 539-560. https://doi.org/10.15388/20-INFOR407

[3] Thang P.C., Kopylov A.V. *Tree-serial parametric dynamic programming with flexible prior model for image denoising*. Computer Optics, 2018, vol. 42(5), pp. 838-845. https://doi.org/10.18287/2412-6179-2018-42-5-838-845

[4] Arikan O., Belge M., Cetin A.E., Erzin E. *Adaptive filtering approaches for non-Gaussian stable processes*, International Conference on Acoustics, Speech, and Signal Processing, 1995, vol. 2, pp. 1400-1403. https://doi.org/10.1109/ICASSP.1995.480503

[5] Idan M., Speyer J.L. *Cauchy estimation for linear scalar systems*. IEEE Transactions on Automatic Control, 2010, vol. 55(6), pp. 1329-1342. https://doi.org/10.1109/TAC.2010.2042009

[6] Banerjee S., Agrawal M. *Underwater acoustic communication in the presence of heavy-tailed impulsive noise with bi-parameter cauchy-Gaussian mixture model*, Ocean Electronics (SYMPOL), 2013, pp. 1-7. https://doi.org/10.1109/SYMPOL.2013.6701903

[7] Chitre M.A., Potter J.R., Ong S.H. *Optimal and near-Opptimal signal detection in snapping shrimp dominated ambient noise*, IEEE Journal of Oceanic Engineering, 2006, vol. 31(2), pp. 497-503. https://doi.org/10.1109/JOE.2006.875272

[8] Carrillo R.E., Aysal T.C., Barner K.E. *A Generalized Cauchy Distribution Framework for Problems Requiring Robust Behavior*, EURASIP Journal on Advances in Signal Processing, 2010, vol. 2010, 312989, pp. 1-19. https://doi.org/10.1155/2010/312989

[9] Ghannudi H.E., Clavier L., et al. *α-stable interference modeling and cauchy receiver for an IR-UWB Ad Hoc network*, IEEE Transactions on Communications, 2010, vol. 58 (6), pp. 1748-1757. https://doi.org/10.1109/TCOMM.2010.06.090074

[10] Gurugopinath S., Muralishankar R., Shankar H.N. *Spectrum sensing in the presence of Cauchy noise through differential entropy*, IEEE Distributed Computing, VLSI, Electrical Circuits and Robotics (DISCOVER), 2016, pp. 201-204. https://doi.org/10.1109/DISCOVER.2016.7806266

[11] Laus F., Pierre F., Steidl G. *Nonlocal Myriad Filters for Cauchy Noise Removal*, Journal of Mathematical Imaging and Vision, 2018, vol. 60, pp. 1324B–1354. https://doi.org/10.1007/s10851-018-0816-y

[12] Shi K., Dong G., Guo Z. *Cauchy noise removal by nonlinear diffusion equations*, Computers & Mathematics with Applications, 2020, vol. 80 (9), pp. 2090-2103 https://doi.org/10.1016/j.camwa.2020.08.027

[13] Lanza A., Sciacchitano F., Morigi S., Sgallari F. *A Unified Framework for the Restoration of Images Corrupted by Additive White Noise*, Scale Space and Variational Methods in Computer Vision, SSVM 2017, Lecture Notes in Computer Science, Springer, 2017, vol. 10302, pp. 498–510. https://doi.org/10.1007/978-3-319-58771-$4_4$0.

[14] Sciacchitano F., Dong Y., Zeng T. *Variational Approach for Restoring Blurred Images with Cauchy Noise*, SIAM Journal on Imaging Sciences, 2015, vol. 8 (3), pp. 1894–1922. https://doi.org/10.1137/140997816

[15] Mei J.J., Dong Y., Huang T.Z., Yin W. *Cauchy Noise Removal by Nonconvex ADMM with Convergence Guarantees*, Journal of Scientific Computing, 2017, vol. 74:, pp. 743B–766. https://doi.org/10.1007/s10915-017-0460-5.

[16] Bai L. *A new approach for Cauchy noise removal*, AIMS Mathematics, 2021, vol. 6(9), pp. 10296-10312. https://doi.org/10.3934/math.2021596

[17] Wang S., Huang T.Z., Zhao X.L. et al. *Speckle noise removal in ultrasound images by first- and second-order total variation*, Numerical Algorithms, 2018, vol. 78, pp. 513B–533. https://doi.org/10.1007/s11075-017-0386-x

[18] Yang J.H., et al. *Total Variation and High-Order Total Variation Adaptive Model for Restoring Blurred Images with Cauchy Noise*, Computers & Mathematics with Applications, 2019, vol. 77 (5), pp. 1255-1272. https://doi.org/10.1016/j.camwa.2018.11.003

[19] Lv Y. *Shearlet-TGV Based Model for Restoring Noisy Images Corrupted by Cauchy Noise*, Nonlinear Differential Equations and Applications, 2020, vol. 27 (2), pp. 1-23. https://doi.org/10.1007/s00030-020-0623-1

[20] Adam T., Paramesran R., Ratnavelu K. *Combined Higher Order Non-Convex Total Variation with Overlapping Group Sparsity for Poisson Noise Removal*, Computational and Applied Mathematics, 2022, vol. 41, pp. 1-33. https://doi.org/10.1007/s40314-022-01828-z

[21] Selesnick I.W., Chen P.Y. *Total variation denoising with overlapping group sparsity*, IEEE International Conference on Acoustics, Speech and Signal Processing, 2013, pp. 5696-5700. https://doi.org/10.1109/ICASSP.2013.6638755

[22] Ding M., et al. *Total Variation with Overlapping Group Sparsity for Deblurring Images Under Cauchy Noise*, Applied Mathematics and Computation, 2019, vol. 341(15), pp. 128-147. https://doi.org/10.1016/j.amc.2018.08.014.

[23] Kim G., Cho J.,Kang M. *Cauchy Noise Removal by Weighted Nuclear Norm Minimization*, Journal of Scientific Computing, 2020, vol. 83 (15), pp. 1-21. https://doi.org/10.1007/s10915-020-01203-2.

[24] Jon K, et al. *Weighted Hyper-Laplacian Prior with Overlapping Group Sparsity for Image Restoration under Cauchy Noise*, Journal of Scientific Computing, 2021, vol. 87(64), pp. 1-32. https://doi.org/10.1007/s10915-021-01461-8.

[25] Zhao Y.T., Cao W.F., Chen Y. *Local-and-Nonlocal Spectral Prior Regularized Tensor Recovery for Cauchy Noise Removal*, Signal Processing 2023; 109176: 1-18. https://doi.org/10.1016/j.sigpro.2023.109176

[26] Bayram I., Kamasak M.E. *A directional total variation*, The 20th European Signal Processing Conference (EUSIPCO) 2012, pp. 265-269. https://doi.org/10.1109/LSP.2012.2220349

[27] Cheng Z., et al. *Four-Directional Total Variation Denoising Using Fast Fourier Transform and ADMM*, IEEE 3rd International Conference on Image, Vision and Computing (ICIVC), 2018, pp. 379-383. https://doi.org/10.1109/ICIVC.2018.8492869

[28] Zhang H., Wang Y. *Edge Adaptive Directional Total Variation*, The Journal of Engineering, 2013, vol. 2013(11), pp. 61-62. https://doi.org/10.1049/joe.2013.0116

[29] Kongskov R.D., Dong Y. *Directional Total Generalized Variation Regularization for Impulse Noise Removal*, Scale Space and Variational Methods in Computer Vision, SSVM 2017, Lecture Notes in Computer Scienc, 2017, vol. 10302, pp. 221B–231 https://doi.org/10.1007/978-3-319-58771-4$_1$8

[30] Kongskov R.D., Dong Y., Knudsen K. *Directional total generalized variation regularization*, BIT Numerical Mathematics, 2019, vol. 59 (4), pp. 903-928. https://doi.org/10.1007/s10543-019-00755-6.

[31] Chen P.Y., Selesnick I. *Translation-Invariant Shrinkage/Thresholding of Group Sparse Signals*, Signal Processing, 2014, vol. 94, pp. 476-489. https://doi.org/10.1016/j.sigpro.2013.06.011.

[32] Liu J., et al. *Image Restoration Using Total Variation with Overlapping Group Sparsity*, Information Sciences, 2015, vol. 295, pp. 232-246 https://doi.org/10.1016/j.ins.2014.10.041

[33] Nesterov Y. *Introductory Lectures on Convex Optimization: A Basic Course*, Springer New York, 2013, vol. 87, 236 pages. https://doi.org/10.1007/978-1-4419-8853-9

[34] Boyd S., et al. *Distributed Optimization and Statistical Learning via the Alternating Direction Method of Multipliers*, Foundations and Trends in Machine Learning, 2011, vol. 3 (1), pp. 1B–122. https://doi.org/10.1561/2200000016

[35] Beck A., Teboulle M. *A Fast Iterative Shrinkage-Thresholding Algorithm for Linear Inverse Problems*, SIAM Journal on Imaging Sciences, 2009, vol. 2 (1), pp. 183-202. https://doi.org/10.1137/080716542

[36] Chambolle A., Pock T. *A First-Order Primal-Dual Algorithm for Convex Problems with Applications to Imaging*, Journal of Mathematical Imaging and Vision, 2010, vol. 50, pp. 102-145. https://doi.org/10.1007/s10851-010-0251-1

[37] Bredies K. *Recovering Piecewise Smooth Multichannel Images by Minimization of Convex Functionals with Total Generalized Variation Penalty*, Efficient Algorithms for Global Optimization Methods in Computer Vision, Lecture Notes in Computer Science, 2014, vol. 8293, pp. 44B–77. https://doi.org/10.1007/978-3-642-54774-4_3

[38] Bovik A.C., Wang Z. *Modern image quality assessment, synthesis lectures on image, video, and multimedia processing*, Synthesis Lectures on Image, Video, and Multimedia Processing, 2006, 146 pages. https://doi.org/10.1007/978-3-031-02238-8.

Pham Cong Thang
The University of Danang B– University of Science and Technology
54 Nguyen Luong Bang, Danang, Vietnam
Email: pcthang@dut.udn.vn,

Tran Thi Thu Thao
The University of Danang B– University of Economics
71 Ngu Hanh Son, Danang, Vietnam
Email: thaotran@due.udn.vn

Truong Tan Cuong
The University of Danang B– University of Science and Technology
54 Nguyen Luong Bang, Danang, Vietnam
Email: 102210053@sv1.dut.udn.vn,

Nguyen Nguyen Anh
The University of Danang B– University of Science and Technology
54 Nguyen Luong Bang, Danang, Vietnam
Email: 102210147@sv1.dut.udn.vn,

Pham Trung Hieu
The University of Danang B– University of Science and Technology
54 Nguyen Luong Bang, Danang, Vietnam
Email: 102210034@sv1.dut.udn.vn,

Received 22.03.2024, Accepted 26.06.2024