IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 6 January 2024, accepted 18 January 2024, date of publication 24 January 2024, date of current version 8 February 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3357835

RESEARCH ARTICLE

Denoising on Textured Image Using Total Generalized Variation With Overlapping Group Sparsity Based on Fast Split Bregman Method

CHENG ZHANG AND KIN SAM YEN
School of Mechanical Engineering, Universiti Sains Malaysia, Engineering Campus, Nibong Tebal, Penang 14300, Malaysia
Corresponding author: Kin Sam Yen (meyks@usm.my)

This work was supported by the Ministry of Higher Education Malaysia for Fundamental Research Grant Scheme under Project FRGS/1/2021/ICT02/USM/02/2.

ABSTRACT The Total Generalized Variation (TGV) model proves effective in removing texture and background patterns in natural images while suppressing the staircase effect introduced by traditional Total Variation (TV) regularization. Nevertheless, TGV falls short in preserving structural features due to its lack of consideration for such structural features. This paper introduces Overlapping Group Sparsity (OGS) regularization into the TGV model with the specific aim of enhancing denoising, particularly in textured images. By leveraging prior knowledge of sparse structures discernible from first-order and second-order gradients, this model surpasses the conventional TGV models in achieving superior denoising and staircase effect elimination. The model proposed employs a fast split Bregman iteration method to address the L1 regularization problem within the complex TGV model combined with OGS. The experimental results comparing various state-of-the-art denoising TV- and TGV-based models highlight a significant improvement in the denoising performance of the proposed model. Specifically, in comparison to the average values of peak signal-to-noise ratio (PSNR) and structural similarity index measure (SSIM) obtained from other state-of-the-art models, the proposed model demonstrated improvements of 3.5% in PSNR and 3.4% in SSIM.

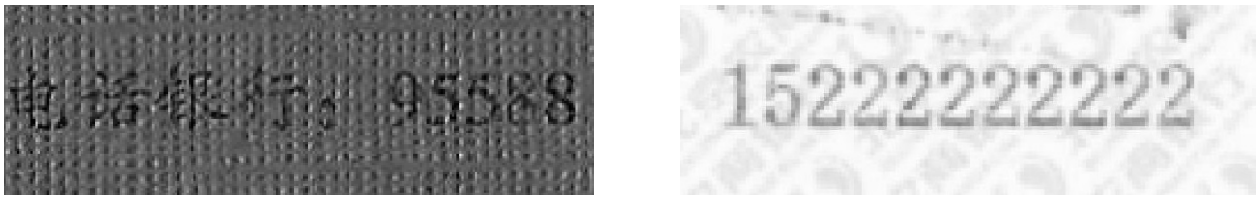
INDEX TERMS Total generalized variation, overlapping group sparsity, texture denoising, split Bregman.

I. INTRODUCTION

Texture typically denotes surface patterns that share similarities in appearance and local statistics [1]. These patterns can generally be categorized as regular, near-regular, or irregular. Textured materials such as paper or fabric find application in the printing industry for security and anti-counterfeit purposes. The smoothening of texture ensures that any modifications to prints, inked characters, or writing become easily detectable, thereby guaranteeing the authenticity and originality of the content. Security printing on textured materials found its application in passbooks, including their covers, internal pages, and check papers. Figure 1 illustrates instances of printed characters on a passbook cover and an internal page. Automated recognition of these printed characters or writing enhances the level of automation in inspection processes and improves authentication accuracy. However, owing to the surface complexity of textured materials, printing conditions, the similarity of adjacent regions or patterns (e.g., guilloche patterns or watermarks), and variations in illumination, the segmentation of printed characters or writing remains a persistent challenge, potentially compromising the accuracy of character recognition [2], [3], [4]. Consequently, effective denoising becomes paramount before undertaking character segmentation and recognition.

Efficient character segmentation methods for textured images can be generally categorized into two classes: deep learning-based methods [5] and variational-based

The associate editor coordinating the review of this manuscript and approving it for publication was Gustavo Callico.



(a) (b)

FIGURE 1. Characters printed on: (a) a passbook cover, (b) an internal page of a passbook.

methods [6]. Deep learning methods are usually complex and require substantial datasets and high computational capabilities for training and fine-tuning. In contrast, variational methods are generally more straightforward. The effectiveness of variational methods relies heavily on the precise representation of prior knowledge regarding the texture model and the assumptions of symmetry and regularity within that model. Therefore, these methods often yield unsatisfactory results when applied to general and randomly textured images.

The widespread application of variational methods in image processing can be traced back to Rudin's research in 1992 [7]. Subsequent algorithms have enhanced Rudin's total variation (TV) algorithm by incorporating various regularization terms tailored to diverse application scenarios. Nevertheless, any modifications or improvements must address the convex optimization problem, a computational challenge known for its sluggishness. Traditional TV regularization assumes images to be piecewise constant functions, which often leads to the staircase effects in solutions [8]. To overcome this limitation, representative enhancements to TV model have surfaced, including non-local total variation [9] and total generalized variation (TGV) [10]. These models have consistently demonstrated commendable outcomes. The TGV model, introduced by Bredies et al. [10], harnesses the benefits of both first-order and multi-order regularization. effectively maintaining smooth and gradient regions in the image and notably mitigating the staircase effect. However, it is essential to acknowledge that the TGV model overlooks the structural similarity within the image [11]. Earlier studies have effectively removed textures from useful information using TGV-based models, affirming the viability of TGV in textured image processing [12], [13], [14].

In recent years, overlapping group sparsity (OGS) regularization term has been proposed by Liu et al. [8] and Chen et al. [15], [16]. Subsequently, it is combined with the TV regularization term to constitute the TV-OGS regularization term [17]. This term adeptly exploits the structured sparseness in the image gradient. However, the TV model lacks the constraint of higher-order difference, and the incorporation of sparse gradients in TV has limited efficacy in suppressing the staircase effect.

This paper introduces an enhanced solution for TGV, which leverages the OGS convergence operator (TGV-OGS), which is employed for the removal of texture and background patterns. The challenge in denoising textured images lies in the conflict between preserving edges and removing textures. In image processing, high-order gradient constraints can control the continuity of low-order gradients [18], which is crucial for image reconstruction. The proposed model attempts to achieve effectiveness in simultaneously eliminating the influence of noise and texture while maintaining image edges. Additionally, the proposed model allows for exploring image pixel-level neighborhood information. Considering the rapid convergence properties associated with the split Bregman iteration technique [19], [20] for partial differential equations involving the L1 norm [19], this study employs the split Bregman technique to minimize the proposed energy equation. To further enhance the efficiency of the split Bregman iteration, we incorporate a fast Fourier transform (FFT) [21] to solve the OGS convergence method within the proposed model.

The subsequent sections of the paper are structured as follows. In the upcoming section, we provide comprehensive definitions and reviews of TV, TGV, OGS, split Bregman, and alternating direction method of multipliers (ADMM). Section III offers an in-depth explanation of the proposed algorithm utilizing the fast split Bregman method to address the proposed model. In Section IV, we present experimental results that illustrate the applicability and effectiveness of our proposed approach. Finally, Section V serves as the conclusion summarizing our work and providing final remarks.

II. RELATED WORK

A. TOTAL VARIATION

Rudin et al. [7] conceptualize an image's noise or texture structure as a model characterized by a small gradient amplitude at each pixel. It is noted that the total variation of a noise-polluted image surpasses that of a noiseless one. Consequently, constraining total variation effectively mitigates noise. Rudin solved the Euler-Lagrange equation for a TV model using the gradient descent method. The traditional TV denoising model introduced by Rudin et al. [7] is shown as follows:

$$\begin{aligned} \boldsymbol{u} &= \arg\min_{\boldsymbol{u}} \frac{\lambda}{2}\|\boldsymbol{u}-\boldsymbol{f}\|_2^2 + R_{TV}(\boldsymbol{u}) \\ &= \arg\min_{\boldsymbol{u}} \frac{\lambda}{2}\|\boldsymbol{u}-\boldsymbol{f}\|_2^2 + \|\nabla_h \boldsymbol{u}\|_1 + \|\nabla_v \boldsymbol{u}\|_1 , \end{aligned} \tag{1}$$

where $\boldsymbol{f}$ denotes the observed image containing noise, while $\boldsymbol{u}$ represents the target image. $\nabla_h \boldsymbol{u}$ and $\nabla_v \boldsymbol{u}$ denote horizontal and vertical differential matrix. (1) comprises two primary components. The term $\|\boldsymbol{u}-\boldsymbol{f}\|_2^2$ is the data fidelity component, ensuring that the resulting image closely approximates the original image. The term $R_{TV}(\boldsymbol{u})$ is the TV regularization component, primarily for image smoothing. λ is a parameter balancing the data fidelity term and the regularization term. $\|\bullet\|_2$ represents the Euclidean norm (short for L2 norm). $\|\bullet\|_1$ represents the Manhattan norm (short for L1 norm). The TV model achieves image denoising by minimizing the energy function (1), ultimately achieving the desired denoising results $\boldsymbol{u}$.

B. TOTAL GENERALIZED VARIATION

TGV model leverages the benefits of both first-order and multi-order regularization. It exhibits adaptive adjustment of the gradient order based on the detailed information and local edge features of different regions within an image, thereby proving effective in simultaneously eliminating the influence of noise and texture while suppressing the staircase effect. It is noteworthy that the TGV model overlooks the structural similarity of the image [11]. Consequently, various enhancements to the TGV model have been consistently proposed.

Ren and Qiu [22] substituted the Lp norm for the L1 norm in the TGV regularization term. In comparison to the L1 operator, the Lp contraction operator introduces an additional degree of freedom, allowing for a more accurate representation of sparse gradient information in the image. However, this enhancement comes at the cost of increased computational complexity [23], [24], [25]. Ban et al. [26] introduced a non-local self-similarity in the transform domain as a prior TGV information, incorporating a multi-directional TGV regularization constraint calculated within the eight-neighborhood space to safeguard the structural characteristics of the image. Fractional order differentiation, an extension of integer order differentiation, was effectively utilized by Xie et al. [27] who combined the fractional gradient operator with the TGV regularization term, thereby enhancing the details of the target image.

For simplicity, we assume the image is a square matrix. Second-order TGV is the most commonly used form. Higher-order TGV model has been demonstrated to cause excessive blurring of image edges [28]. The second-order TGV denoising model can be defined as (2) [29]:

$$\begin{aligned}\boldsymbol{u} &= \arg\min_{u}\left\{\frac{1}{2}\|\boldsymbol{u}-\boldsymbol{f}\|_2^2+\tau \mathrm{TGV}_{\alpha}^2(\boldsymbol{u})\right\}\\ &=\arg\min_{u,v}\left\{\frac{1}{2}\|\boldsymbol{u}-\boldsymbol{f}\|_2^2+\tau\alpha_0\|\nabla\boldsymbol{u}-\boldsymbol{v}\|_1+\tau\alpha_1\|\varepsilon(v)\|_1\right\},\end{aligned} \tag{2}$$

where $\boldsymbol{u}\in\mathbf{R}^N$ is the discrete vectorization of the denoising image u. The term $\|\boldsymbol{u}-\boldsymbol{f}\|_2^2$ representing the data fidelity term, and $\mathrm{TGV}_{\alpha}^2(\boldsymbol{u})$ denotes second-order TGV regularization. The parameter τ balances the data fidelity and TGV regularization. The non-negative gradient weights α_0 and α_1 are employed to manage the first-order and second-order gradients. $\|\nabla\boldsymbol{u}-\boldsymbol{v}\|_1$ represents the total magnitude of the first-order gradient, while $\|\varepsilon(\boldsymbol{v})\|_1$ signifies the total magnitude of the second-order gradient. The gradient operator $\nabla\boldsymbol{u}=\left[\nabla_x\boldsymbol{u};\nabla_y\boldsymbol{u}\right]\in\mathbf{R}^{N,2}$ includes the horizontal differential operator $\nabla_x\boldsymbol{u}$ and vertical differential operator $\nabla_y\boldsymbol{u}$. The vector $\boldsymbol{v}=\left[\boldsymbol{v}_x;\boldsymbol{v}_y\right]\in\mathbf{R}^{N,2}$ constrains the sparsity of the first-order gradient, where $\boldsymbol{v}_x$ and $\boldsymbol{v}_y$ are the horizontal and vertical differentials of $\boldsymbol{v}$. The symmetric gradient derivative $\varepsilon(\boldsymbol{v})$ is as follows:

$$\varepsilon(\boldsymbol{v})=\frac{\nabla\boldsymbol{v}+\nabla\boldsymbol{v}^{\mathbf{T}}}{2}=\begin{bmatrix}\nabla_x\boldsymbol{v}_x & \frac{1}{2}\left(\nabla_x\boldsymbol{v}_y+\nabla_y\boldsymbol{v}_x\right)\\ \frac{1}{2}\left(\nabla_x\boldsymbol{v}_y+\nabla_y\boldsymbol{v}_x\right) & \nabla_y\boldsymbol{v}_y\end{bmatrix} \tag{3}$$

Thus, the following equation can be derived:

$$\begin{aligned}(\boldsymbol{u},\boldsymbol{v})=\underset{u,v}{argmin}\,&\frac{1}{2}\|\boldsymbol{u}-\boldsymbol{f}\|_2^2+\tau\alpha_0(\|\nabla_x\boldsymbol{u}-\boldsymbol{v}_x\|_1\\ &+\|\nabla_y\boldsymbol{u}-\boldsymbol{v}_y\|_1)+\tau\alpha_1\\ &\left(\|\nabla_x\boldsymbol{v}_x\|_1+\|\nabla_x\boldsymbol{v}_y+\nabla_x\boldsymbol{v}_y\|_1+\|\nabla_y\boldsymbol{v}_y\|_1\right).\end{aligned} \tag{4}$$

C. OVERLAPPING GROUP SARSITY

Traditional TV processes the gradient of each pixel in isolation, neglecting its neighborhood information, which results in the ineffective separation of pixels with high noise pollution from the edge of the image. Liu et al. [8] introduced a TV model based on the OGS regularization term. This regularization term not only accounts for the sparsity of the image difference domain but also exploits neighborhood difference information for each pixel, thereby enhancing the distinction between smooth and boundary regions and significantly reducing the occurrence of staircase artifacts. Later studies [11], [30] have integrated OGS with high-order TV to further diminish the staircase effect. Regrettably, this model has led to increased blurring of edges.

The OGS shrinkage as formulated by Liu et al. [8] is expressed as follows:

$$\begin{aligned}\boldsymbol{u}&=\arg\min_{\boldsymbol{u}}\frac{\lambda}{2}\|\boldsymbol{u}-\boldsymbol{f}\|_2^2+R_{TVOGS}(\boldsymbol{u})\\ &=\arg\min_{\boldsymbol{u}}\frac{\lambda}{2}\|\boldsymbol{u}-\boldsymbol{f}\|_2^2+\varphi(\nabla_h\boldsymbol{u})+\varphi(\nabla_v\boldsymbol{u})\end{aligned} \tag{5}$$

where $R_{\mathrm{TVOGS}}(\boldsymbol{u})$ represents the TV-OGS regularization term, two-dimensional image $\boldsymbol{u}\in\mathbf{R}^{n\times n}$, $\tilde{\boldsymbol{u}}_{i,j,k,k}$ is the overlapping group matrix, which is defined as follows:

$$\tilde{\boldsymbol{u}}_{i,j,k,k}=\begin{pmatrix}\boldsymbol{u}_{i-K_l,j-K_l} & \boldsymbol{u}_{i-K_l,j-K_l+1} & \cdots & \boldsymbol{u}_{i-K_l,j+K_r}\\ \boldsymbol{u}_{i-K_l+1,j-K_l} & \boldsymbol{u}_{i-K_l+1,j-K_l+1} & \cdots & \boldsymbol{u}_{i-K_l+1,j+K_r}\\ \vdots & \vdots & \ddots & \vdots\\ \boldsymbol{u}_{i+K_r,j-K_l} & \boldsymbol{u}_{i+K_r,j-K_l+1} & \cdots & \boldsymbol{u}_{i+K_r,j+K_r}\end{pmatrix}\in\mathbf{R}^{n\times n}, \tag{6}$$

where $K_l=\lfloor(K-1)/2\rfloor, K_r=\lfloor K/2\rfloor$, $\lfloor x\rfloor$ represents the rounding operator. From (6), it can be seen that the OGS

regularization term considers the neighborhood gradient information of $K \times K$ pixels within an image. In this context, K is also referred to as the group size of the neighborhood pixels. The overlapping group gradient of the processed pixel is as follows:

$$\varphi(\boldsymbol{u}) = \sum_{i=1}^{K}\sum_{j=1}^{K} \|\tilde{\boldsymbol{u}}_{i,j,k,k}\|_2. \tag{7}$$

Using the majorization-minimization (MM) method [31], (5) can be easily solved. The iterated solution is as follows:

$$\boldsymbol{u}^{(k+1)} = \left[\boldsymbol{I} + \gamma \boldsymbol{\Lambda}^2\left(\boldsymbol{u}^{(k)}\right)\right]^{-1} \boldsymbol{f} \tag{8}$$

$\boldsymbol{I} \in \boldsymbol{R}^{N^2 \times \acute{A}N^2}$ represents the unit matrix. $\boldsymbol{\Lambda} \in \boldsymbol{R}^{N^2 \times \acute{A}N^2}$ is a diagonal matrix.

D. SPLIT BREGMAN AND ADMM

In addition to investigating regularization terms for effective denoising, addressing computational complexity in variational models stands out as another major focus. Numerous efficient and effective algorithms have emerged since Rudin's initial suggestion to employ the gradient descent algorithm. Notably, the split Bregman [19], [20] and ADMM algorithms [32] have gained popularity and have been introduced to address the intricate computational challenges associated with TV-based variational models.

Ren et al. [33] introduced a second-order TGV model for image deblurring. To address the associated computational challenges, the split Bregman iteration was devised to solve the Euler-Lagrange equation of the TGV model. Simultaneously, Liu et al. [8] employed the split Bregman method to seek a solution for the TV-OGS in the proposed image deblurring model. In a related vein, Chen et al. [34] incorporated the split Bregman iteration framework to address the partial differential equation of the TV-OGS denoising model. These models showcase enhanced efficiency through the application of the split Bregman iteration method for solving TV-related models. Later in 2019, Chen et al. [35] further extended this approach by integrating the OGS concept into the TGV regularization term for seismic signal analysis. The work introduced the ADMM to solve the proposed model specifically for Gaussian noise denoising.

Both ADMM and Split Bregman are iterative methods designed for convex optimization. The ADMM method is a more general-purpose method applicable to a wider range of optimization problems [32]. In general, split Bregman is a specialized form of ADMM tailored for problems involving L1 norm regularization, offering efficiency advantages in scenarios where sparsity is a key feature [19], [20]. Typically, regularization terms like the L1 norm in ADMM are managed through auxiliary variables and Lagrangian multipliers, rather than direct implementation of L1 regularization. However, the Split Bregman algorithm directly incorporates the L1 regularization term into its variable update step, enhancing efficiency, especially for problems where sparsity is a crucial characteristic. As a result, it finds predominant use in TV-based models and other applications where sparsity is a desired property. The Split Bregman algorithm exhibits rapid convergence, particularly in specific problem instances [36], while the convergence of ADMM can be sluggish in certain cases [37]. Nevertheless, the broad applicability of ADMM contributes to commendable performance across diverse problem domains. For a better understanding, the procedures of both ADMM and Split Bregman in solving the TV model as in (1) are demonstrated in Appendix A.

III. THE PROPOSED MODEL

A. TGV-OGS (FAST SPLIT BREGMAN)

The TGV model generally excels in alleviating the staircase effect, yet its efficacy in preserving weak edges and details may be inadequate [38]. Both TV and TGV, along with their enhanced variations [27], serve as global regularity priors, focusing on overall image regularity without accounting for diverse local structures within the image. In contrast, the OGS regularization term enhances distinctiveness between smooth gradient and edge gradient regions. Noting that the TV-OGS model only considers first-order combined gradients and neglects higher-order gradients, we propose an extension from TGV to the TGV-OGS model. This new model incorporates the similarity of first-order and second-order gradients, ensuring a more comprehensive regularization method. The aforementioned studies suggest that employing TGV and OGS for image denoising in textured images holds promise and is feasible. Nevertheless, the inclusion of OGS introduces shortcomings, notably the high computational complexity associated with L1 regularization and a decrease in algorithm efficiency concerning speed.

By applying OGS to the TGV regularization term, the proposed denoising model is as follows.

$$\begin{aligned}\boldsymbol{u} &= \underset{\boldsymbol{u}}{arg\min}\ \frac{1}{2}\|\boldsymbol{u}-\boldsymbol{f}\|_2^2 + \tau TGV^2_{\alpha-OGS}(\boldsymbol{u}) \\ &= \underset{\boldsymbol{u},\boldsymbol{v}}{arg\min}\ \frac{1}{2}\|\boldsymbol{u}-\boldsymbol{f}\|_2^2 + \tau\alpha_0[\varphi(\nabla_x\boldsymbol{u}-\boldsymbol{v}_x)+\varphi(\nabla_y\boldsymbol{u}-\boldsymbol{v}_y)] \\ &\quad + \tau\alpha_1[\varphi(\nabla_x\boldsymbol{v}_x)+\varphi(\nabla_y\boldsymbol{v}_y)+\varphi(\nabla_x\boldsymbol{v}_x+\nabla_y\boldsymbol{v}_y)]\end{aligned} \tag{9}$$

where $TGV^2_{\alpha-OGS}(\boldsymbol{u})$ denotes the second-order TGV regularization term after adding OGS regularization.

Compared to the typical TGV model shown in (4), the proposed model is more suitable to depict the structural similarity prior and the high-order gradient sparsity prior.

B. NUMERICAL ALGORITHM

The solutions for image denoising in (9) involve optimization problems with multiple L1 norms. The non-smooth nature of the L1 norm introduces challenges when implementing the proposed model in conjunction with the FFT [21]. To address this, we employ the fast split Bregman iteration framework to solve (9). Drawing inspiration from [8], [33], and [34], this paper employs the following steps in the equation-solving process:

First, introducing auxiliary variables $\boldsymbol{S}_x, \boldsymbol{S}_y, \boldsymbol{L}_{xx}, \boldsymbol{L}_{xy}, \boldsymbol{L}_{yy}$. (9) can be written as follows.

$$\begin{aligned}(\boldsymbol{u}, \boldsymbol{v}) = \underset{\boldsymbol{u},\boldsymbol{v}}{argmin}\ & \frac{1}{2} \parallel \boldsymbol{u} - \boldsymbol{f} \parallel_2^2 + \tau\alpha_0[\varphi\left(\nabla_x \boldsymbol{u} - \boldsymbol{v}_x\right) \\ & + \varphi\left(\nabla_y \boldsymbol{u} - \boldsymbol{v}_y\right)] \\ & + \tau\alpha_1[\varphi\left(\nabla_x \boldsymbol{v}_x\right) + \varphi\left(\nabla_x \boldsymbol{v}_y + \nabla_x \boldsymbol{v}_y\right) + \varphi\left(\nabla_y \boldsymbol{v}_y\right)] \\ s.t.\ & \boldsymbol{S}_x = \nabla_x \boldsymbol{u} - \boldsymbol{v}_x, \boldsymbol{S}_y = \nabla_y \boldsymbol{u} - \boldsymbol{v}_y, \boldsymbol{L}_{xx} = \nabla_x \boldsymbol{v}_x, \\ & \boldsymbol{L}_{xy} = \nabla_x \boldsymbol{v}_x + \nabla_y \boldsymbol{v}_y, \boldsymbol{L}_{yy} = \nabla_y \boldsymbol{v}_y. \end{aligned} \tag{10}$$

According to the principle of split Bregman [19], the Lagrangian multipliers and quadratic penalty terms are needed to establish the augmented Lagrangian function. Therefore, we have the following:

$$\begin{aligned}&\left(\boldsymbol{u}^{k+1}, \boldsymbol{v}^{k+1}, \boldsymbol{S}^{k+1}, \boldsymbol{L}^{k+1}\right) \\ &= \arg \underset{\boldsymbol{u},\boldsymbol{v},\boldsymbol{S},\boldsymbol{L}}{min} \left\{ \frac{1}{2}\|\boldsymbol{u} - \boldsymbol{f}\|_2^2 + \tau a_0 \left[\varphi\left(\boldsymbol{S}_x\right) + \varphi\left(\boldsymbol{S}_y\right)\right] \right. \\ &\quad + \tau a_1 \left[\varphi\left(\boldsymbol{L}_{xx}\right) + \varphi\left(\boldsymbol{L}_{yy}\right) + \varphi\left(\boldsymbol{L}_{xy}\right)\right] \\ &\quad + \frac{\lambda_1}{2} \left\| \boldsymbol{S}_x - \left(\nabla_x \boldsymbol{u} - \boldsymbol{v}_x\right) - \boldsymbol{d}_x^k \right\|_2^2 \\ &\quad + \frac{\lambda_1}{2} \left\| \boldsymbol{S}_y - \left(\nabla_y \boldsymbol{u} - \boldsymbol{v}_y\right) - \boldsymbol{d}_y^k \right\|_2^2 \\ &\quad + \frac{\lambda_2}{2} \left\| \boldsymbol{L}_{xx} - \nabla_x \boldsymbol{v}_x - \boldsymbol{d}_{xx}^k \right\|_2^2 \\ &\quad + \frac{\lambda_2}{2} \left\| \boldsymbol{L}_{xy} - \left(\nabla_y \boldsymbol{v}_x + \nabla_x \boldsymbol{v}_y\right) \right. \\ &\quad \left. \left. - \boldsymbol{d}_{xy}^k \right\|_2^2 + \frac{\lambda_2}{2} \left\| \boldsymbol{L}_{yy} - \nabla_y \boldsymbol{v}_y - \boldsymbol{d}_{yy}^k \right\|_2^2 \right\} \end{aligned} \tag{11}$$

where λ_1 and λ_2 are non-negative penalty parameters. $\boldsymbol{d}_x, \boldsymbol{d}_y, \boldsymbol{d}_{xx}, \boldsymbol{d}_{xy}$, and $\boldsymbol{d}_{yy}$ are the Lagrange multipliers, these auxiliary variables can be updated as follows:

$$\begin{aligned} \boldsymbol{d}_x^{k+1} &= \boldsymbol{d}_x^k + \left(\nabla_x \boldsymbol{u}^{k+1} - \boldsymbol{v}_x^{k+1} - \boldsymbol{S}_x^{k+1}\right) \\ \boldsymbol{d}_y^{k+1} &= \boldsymbol{d}_y^k + \left(\nabla_y \boldsymbol{u}^{k+1} - \boldsymbol{v}_y^{k+1} - \boldsymbol{S}_y^{k+1}\right) \\ \boldsymbol{d}_{xx}^{k+1} &= \boldsymbol{d}_{xx}^k + \nabla_x \boldsymbol{v}_x^{k+1} - \boldsymbol{L}_{xx}^{k+1} \\ \boldsymbol{d}_{xy}^{k+1} &= \boldsymbol{d}_{xy}^k + \left(\nabla_x \boldsymbol{v}_y^{k+1} + \nabla_y \boldsymbol{v}_x^{k+1}\right) - \boldsymbol{L}_{xy}^{k+1} \\ \boldsymbol{d}_{yy}^{k+1} &= \boldsymbol{d}_{yy}^k + \nabla_y \boldsymbol{v}_y^{k+1} - \boldsymbol{L}_{yy}^{k+1} \end{aligned} \tag{12}$$

According to reference [19], it can be inferred that as the Lagrange multipliers approach zero, the optimization problem presented in (11) converges to the solution of the original problem outlined in (9).

To efficiently address (12), the Split Bregman algorithm commences its iteration to update each variable, namely, u, v, $\boldsymbol{S}_x, \boldsymbol{S}_y, \boldsymbol{L}_{xx}, \boldsymbol{L}_{xy}$, and $\boldsymbol{L}_{yy}$ in (11) by iteratively minimizing the equation. This iterative process can be decomposed into several sub-problems for optimization.

The solution for the sub-problem u can be mathematically expressed as follows in (13).

$$\begin{aligned} \boldsymbol{u}^{k+1} = \arg \underset{\boldsymbol{u}}{min} \left\{ \frac{1}{2}\|\boldsymbol{u} - \boldsymbol{f}\|_2^2 \right. \\ + \frac{\lambda_1}{2} \left\| \boldsymbol{S}_x - \left(\nabla_x \boldsymbol{u} - \boldsymbol{v}_x\right) - \boldsymbol{d}_x^k \right\|_2^2 \\ \left. + \frac{\lambda_1}{2} \left\| \boldsymbol{S}_y - \left(\nabla_y \boldsymbol{u} - \boldsymbol{v}_y\right) - \boldsymbol{d}_y^k \right\|_2^2 \right\}. \end{aligned} \tag{13}$$

By sorting out the objective function of the above equation and finding the gradient of $\boldsymbol{u}$, we obtain:

$$\begin{aligned} &\left(1 + \lambda_1 \nabla_x^{\mathrm{T}} \nabla_x + \lambda_1 \nabla_y^{\mathrm{T}} \nabla_y\right) \boldsymbol{u} \\ &= \boldsymbol{f} + \lambda_1 \nabla_x^{\mathrm{T}} \left(\boldsymbol{v}_x^k + \boldsymbol{S}_x^k - \boldsymbol{d}_x^k\right) + \lambda_1 \nabla_y^{\mathrm{T}} \left(\boldsymbol{v}_y^k + \boldsymbol{S}_y^k - \boldsymbol{d}_y^k\right) \end{aligned} \tag{14}$$

Under the assumption of periodic boundary conditions, the above equation can be solved using the FFT as follows.

$$\boldsymbol{u}^{k+1} = \mathcal{F}^{-1} \frac{\mathcal{F}\left[\boldsymbol{f} + \lambda_1 \nabla_x^{\mathrm{T}} \left(\boldsymbol{v}_x^k + \boldsymbol{S}_x^k - \boldsymbol{d}_x^k\right) + \lambda_1 \nabla_y^{\mathrm{T}} \left(\boldsymbol{v}_y^k + \boldsymbol{S}_y^k - \boldsymbol{d}_y^k\right)\right]}{\mathcal{F}\left(\boldsymbol{1} + \lambda_1 \nabla_x^{\mathrm{T}} \nabla_x + \lambda_1 \nabla_y^{\mathrm{T}} \nabla_y\right)} \tag{15}$$

Among them, $\mathcal{F}$ is the two-dimensional Fourier transform operator, and $\mathcal{F}^{-1}$ is the inverse two-dimensional Fourier transform operator. Similarly, the solution for the sub-problems $\boldsymbol{v}_x$ and $\boldsymbol{v}_y$ can be mathematically expressed in (16).

$$\begin{aligned} (\boldsymbol{v}_x^{k+1}, \boldsymbol{v}_y^{k+1}) = \arg \underset{v_x, v_y}{max} \{ & \frac{\lambda_1}{2} \|\boldsymbol{S}_x^k - (\nabla_x \boldsymbol{u}^{k+1} - \boldsymbol{v}_x) - \boldsymbol{d}_x^k\|_2^2 \\ & + \frac{\lambda_1}{2} \|\boldsymbol{S}_y^k - \left(\nabla_y \boldsymbol{u}^{k+1} - \boldsymbol{v}_y\right) - \boldsymbol{d}_y^k\|_2^2 \\ & + \frac{\lambda_2}{2} \|\boldsymbol{L}_{xx}^k - \nabla_x \boldsymbol{v}_x - \boldsymbol{d}_{xx}^k\|_2^2 \\ & + \frac{\lambda_2}{2} \|\boldsymbol{L}_{xy}^k - \left(\nabla_y \boldsymbol{v}_x + \nabla_x \boldsymbol{v}_y\right) - \boldsymbol{d}_{xy}^k\|_2^2 \\ & + \frac{\lambda_2}{2} \|\boldsymbol{L}_{yy}^k - \nabla_y \boldsymbol{v}_y - \boldsymbol{d}_{yy}^k\|_2^2 \} \end{aligned} \tag{16}$$

The sorting process is the same as sub-problem $\boldsymbol{u}$. We can obtain a system of linear equations about $\boldsymbol{v}_x$ and $\boldsymbol{v}_y$:

$$\begin{aligned} &\left[\boldsymbol{I} + \tau_1 \nabla_x^{\mathrm{T}} \nabla_x + \tau_1 \nabla_y^{\mathrm{T}} \nabla_y\right] \boldsymbol{v}_x + \tau_1 \nabla_y^{\mathrm{T}} \nabla_x \boldsymbol{v}_y \\ &= \tau_1 \nabla_x^{\mathrm{T}} \left(\boldsymbol{L}_{xx}^k - \boldsymbol{b}_{xx}^k\right) + \tau_1 \nabla_y^{\mathrm{T}} \left(\boldsymbol{L}_{xy}^k - \boldsymbol{b}_{xy}^k\right) \\ &\quad + \nabla_x \boldsymbol{u}^{k+1} - \boldsymbol{S}_x^k + \boldsymbol{d}_x^k \\ &\left[\boldsymbol{I} + \tau_1 \nabla_x^{\mathrm{T}} \nabla_x + \tau_1 \nabla_y^{\mathrm{T}} \nabla_y\right] \boldsymbol{v}_y + \tau_1 \nabla_x^{\mathrm{T}} \nabla_y \boldsymbol{v}_x \\ &= \tau_1 \nabla_y^{\mathrm{T}} \left(\boldsymbol{L}_{yy}^k - \boldsymbol{d}_{yy}^k\right) + \tau_1 \nabla_x^{\mathrm{T}} \left(\boldsymbol{L}_{xy}^k - \boldsymbol{d}_{xy}^k\right) \\ &\quad + \nabla_y \boldsymbol{u}^{k+1} - \boldsymbol{S}_y^k + \boldsymbol{d}_y^k \end{aligned} \tag{17}$$

Among them, $\tau_1 = \frac{\lambda_2}{\lambda_1}$. Let:

$$\begin{aligned} \boldsymbol{A}_1 &= \boldsymbol{I} + \tau_1 \nabla_x^{\mathrm{T}} \nabla_x + \tau_1 \nabla_y^{\mathrm{T}} \nabla_y \\ \boldsymbol{A}_2 &= \tau_1 \nabla_y^{\mathrm{T}} \nabla_x \\ \boldsymbol{B}_1^k &= \tau_1 \nabla_x^{\mathrm{T}} \left(\boldsymbol{L}_{xx}^k - \boldsymbol{b}_{xx}^k\right) + \tau_1 \nabla_y^{\mathrm{T}} \left(\boldsymbol{L}_{xy}^k - \boldsymbol{b}_{xy}^k\right) \\ &\quad + \nabla_x \boldsymbol{u}^{k+1} - \boldsymbol{S}_x^k + \boldsymbol{d}_x^k, \\ \boldsymbol{B}_2^k &= \tau_1 \nabla_y^{\mathrm{T}} \left(\boldsymbol{L}_{yy}^k - \boldsymbol{b}_{yy}^k\right) + \tau_1 \nabla_x^{\mathrm{T}} \left(\boldsymbol{L}_{xy}^k - \boldsymbol{b}_{xy}^k\right) \\ &\quad + \nabla_y \boldsymbol{u}^{k+1} - \boldsymbol{S}_y^k + \boldsymbol{d}_y^k \end{aligned} \tag{18}$$

using the two-dimensional FFT operator $\mathcal{F}$, (18) can be written as:

$$\begin{cases} \mathcal{F}\boldsymbol{A}_1 .* \mathcal{F}\boldsymbol{v}_x + \mathcal{F}\boldsymbol{A}_2 .* \mathcal{F}\boldsymbol{v}_y = \mathcal{F}\boldsymbol{B}_1^k \\ \mathcal{F}\boldsymbol{A}_2^{\mathrm{T}} .* \mathcal{F}\boldsymbol{v}_x + \mathcal{F}\boldsymbol{A}_1 .* \mathcal{F}\boldsymbol{v}_y = \mathcal{F}\boldsymbol{B}_2^k \end{cases} \tag{19}$$

The symbol .∗ represents the corresponding dot product between vector elements, which the following equations can quickly solve:

$$\begin{cases} \boldsymbol{v}_x^{k+1} = \mathcal{F}^{-1}\left[\dfrac{\mathcal{F}\boldsymbol{A}_1 * \mathcal{F}\boldsymbol{B}_1^k - \mathcal{F}\boldsymbol{A}_2^{\mathrm{T}} * \mathcal{F}\boldsymbol{B}_2^k}{\mathcal{F}\boldsymbol{A}_1 * \mathcal{F}\boldsymbol{A}_1 - \mathcal{F}\boldsymbol{A}_3^{\mathrm{T}} * \mathcal{F}\boldsymbol{A}_3}\right] \\ \boldsymbol{v}_y^{k+1} = \mathcal{F}^{-1}\left[\dfrac{\mathcal{F}\boldsymbol{A}_1 * \mathcal{F}\boldsymbol{B}_2^k - \mathcal{F}\boldsymbol{A}_2^{\mathrm{T}} * \mathcal{F}\boldsymbol{B}_1^k}{\mathcal{F}\boldsymbol{A}_1 * \mathcal{F}\boldsymbol{A}_1 - \mathcal{F}\boldsymbol{A}_2^{\mathrm{T}} * \mathcal{F}\boldsymbol{A}_2}\right] \end{cases} \tag{20}$$

Lastly, the solution for the sub-problems $\boldsymbol{S}_x, \boldsymbol{S}_y$ can be mathematically expressed in (21) as the following.

$$\begin{cases} \boldsymbol{S}_x^{k+1} = \arg(min)_{\boldsymbol{S}_x} \left\{ \tau a_0 \varphi\left(\boldsymbol{S}_x\right) + \dfrac{\lambda_1}{2} \|\boldsymbol{S}_x - \left(\nabla_x \boldsymbol{u}^{k+1} - \boldsymbol{v}_x^{k+1}\right) - \boldsymbol{d}_x^k\|_2^2 \right\} \\ \mathbf{S}_y^{k+1} = \arg(min)_{\boldsymbol{S}_y} \left\{ \tau a_0 \varphi\left(\boldsymbol{S}_y\right) + \dfrac{\lambda_1}{2} \|\boldsymbol{S}_y - \left(\nabla_y \boldsymbol{u}^{k+1} - \boldsymbol{v}_y^{k+1}\right) - \boldsymbol{d}_y^k\|_2^2 \right\} \end{cases} \tag{21}$$

According to Eqs. (5) and (8), the updated formula for $\boldsymbol{S_x}$ and $\boldsymbol{S_x}$ can be obtained as:

$$\begin{cases} \boldsymbol{S}_{x(n+1)}^{(k+1)} = \left[\boldsymbol{I} + \dfrac{\tau\alpha_0}{\lambda_1} \boldsymbol{\Lambda}^2\left(\boldsymbol{S}_{x(n)}^{(k+1)}\right)\right]^{-1} \boldsymbol{S}_{x(0)}^{(k+1)} \\ \boldsymbol{S}_{y(n+1)}^{(k+1)} = \left[\boldsymbol{I} + \dfrac{\tau\alpha_0}{\lambda_1} \boldsymbol{\Lambda}^2\left(\boldsymbol{S}_{y(n)}^{(k+1)}\right)\right]^{-1} \boldsymbol{S}_{y(0)}^{(k+1)} \end{cases} \tag{22}$$

where $\boldsymbol{S}_{x(n)}^{(k)}$ and $\boldsymbol{S}_{y(n)}^{(k)}$ represent the k updated by the $(k)\,th$ outer loop and the $(n)\,th$ inner loop iteration (group sparse convergence iteration cycle), and $\boldsymbol{S}_{x(0)}^{k+1} = \nabla_x \boldsymbol{u}^{k+1} - \boldsymbol{v}_y^{k+1} + \boldsymbol{d}_x^k$, $\boldsymbol{S}_{y(0)}^{k+1} = \nabla_y \boldsymbol{u}^{k+1} - \boldsymbol{v}_k^{k+1} + \boldsymbol{d}_y^k$. Similarly, the updated formula for $\boldsymbol{L}_{xx}, \boldsymbol{L}_{xy}, \boldsymbol{L}_{yy}$.

$$\begin{cases} \mathbf{L}_{xx(n+1)}^{(k+1)} = \left[\boldsymbol{I} + \dfrac{\tau\alpha_1}{\lambda_2} \boldsymbol{\Lambda}^2\left(\boldsymbol{L}_{xx(n)}^{(k+1)}\right)\right]^{-1} \boldsymbol{L}_{xx(0)}^{(k+1)} \\ \mathbf{L}_{yy(n+1)}^{(k+1)} = \left[\boldsymbol{I} + \dfrac{\tau\alpha_1}{\lambda_2} \boldsymbol{\Lambda}^2\left(\boldsymbol{L}_{yy(n)}^{(k+1)}\right)\right]^{-1} \boldsymbol{L}_{yy(0)}^{(k+1)} \\ \mathbf{L}_{xy(n+1)}^{(k+1)} = \left[\boldsymbol{I} + \dfrac{\tau\alpha_1}{\lambda_0} \boldsymbol{\Lambda}^2\left(\boldsymbol{L}_{xy(n)}^{(k+1)}\right)\right]^{-1} \boldsymbol{L}_{xy(0)}^{(k+1)} \end{cases} \tag{23}$$

where, $\boldsymbol{L}_{xx(0)}^{k+1} = \nabla_x \boldsymbol{v}_x^{k+1} + \boldsymbol{d}_{xx}^k$, $\boldsymbol{L}_{yy(0)}^{k+1} = \nabla_y \boldsymbol{v}_y^{k+1} + \boldsymbol{d}_{yy}^k$, $\boldsymbol{L}_{xy(0)}^{k+1} = \nabla_x \boldsymbol{v}_x^{k+1} + \nabla_y \boldsymbol{v}_y^{k+1} + \boldsymbol{d}_{xy}^k$.

The pseudocode for the algorithm described in this work, based on fast split Bregman iteration for solving (9) is as follows:

Algorithm 1 Pseudo-Code of TGV-OGS

Input: Observed image f, model parameters $\tau, \alpha_1, \alpha_2, \lambda_1, \lambda_2$, convergence parameters ϵ, OGS group size K.

Initialize: $\boldsymbol{u}^0 = \boldsymbol{f}, \boldsymbol{L}^0 = \left[\boldsymbol{L}_{xx}^0, \boldsymbol{L}_{xy}^0, \boldsymbol{L}_{yy}^0\right] = 0, \boldsymbol{S}^0 = \left[\boldsymbol{S}_x^0, \boldsymbol{S}_y^0\right] = 0, \left(\boldsymbol{d}_x^0, \boldsymbol{d}_y^0\right) = 0, \left(\boldsymbol{d}_{xx}^0, \boldsymbol{d}_{xy}^0, \boldsymbol{d}_{yy}^0\right) = 0, K = 3.$

k=1

while $\dfrac{\|\boldsymbol{u}^k - \boldsymbol{u}^{k-1}\|_2^2}{\|\boldsymbol{u}^k\|_2^2} \le \epsilon$ do

1. Execute (15) to obtain the solution of $\boldsymbol{u}$ sub-problem.
2. Execute (20) to obtain the solution of $(\boldsymbol{v}_x, \boldsymbol{v}_y)$ sub-problem.
3. Execute (22) and its changing forms to obtain the solution of $(\boldsymbol{S}_x, \boldsymbol{S}_y)$ sub-problem.
4. Execute (23) and its changing forms to obtain the solution of $(\boldsymbol{L}_{xx}, \boldsymbol{L}_{xy}, \boldsymbol{L}_{yy})$ sub-problem.
5. Execute (12), Update auxiliary variables $\boldsymbol{d}_x^0, \boldsymbol{d}_y^0, \boldsymbol{d}_{xx}^0, \boldsymbol{d}_{xy}^0, \boldsymbol{d}_{yy}^0$.

k = k + 1.

end while.

Output: Denoising image u.

IV. EXPERIMENT

A. EXPERIMENTAL SETUP

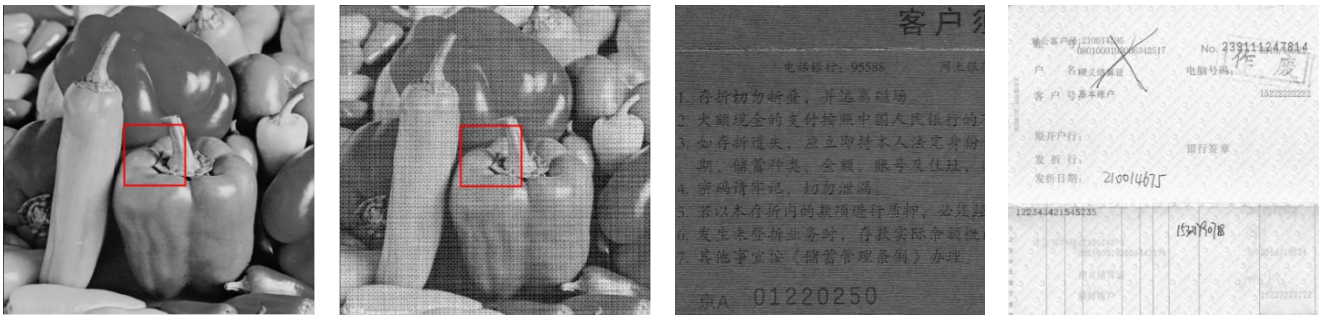
This study conducted two experiments to assess the performance of the proposed model under varying noise conditions. The first experiment utilized a grayscale ''peppers'' image (Figure 2(a)), and its contaminated version with synthetic texture (Figure 2(b)). The regions enclosed by red rectangles in both images were designated as regions of interest for observation post-denoising. These images were employed to evaluate the proposed model's efficacy in removing strong synthetic textures.

To quantitatively assess the quality of denoising results across different models, we utilized three key metrics: Peak signal to noise ratio (PSNR), Structural similarity index (SSIM), and processing time [39] The PSNR quantifies the ratio of peak signal energy to the average noise energy. A higher PSNR indicates a lower level of noise or distortion in the denoised image. The expression of PSNR is expressed in (26).

$$\mathrm{PSNR}\left(\boldsymbol{X}, \boldsymbol{Y}\right) = 10 \lg \frac{255^2}{\frac{1}{N^2}\sum_{i=1}^{N}\sum_{j=1}^{N}\left(\boldsymbol{X}_{ij} - \boldsymbol{Y}_{ij}\right)^2} \tag{24}$$

The SSIM assesses the similarity between two images, considering factors such as brightness, contrast, and structural details. It serves as an objective standard for gauging image distortion and noise suppression. A higher SSIM value signifies a more remarkable similarity between the denoised and original images with SSIM values bounded between zero and one. The expression of SSIM is expressed in (27).

$$\mathrm{SSIM}\left(\boldsymbol{X}, \boldsymbol{Y}\right) = \frac{\left(2\boldsymbol{u}_X\boldsymbol{u}_Y + (255k_1)^2\right)\left(2\sigma_{XY} + (255k_2)^2\right)}{\left(\boldsymbol{u}_X^2 + \boldsymbol{u}_Y^2 + (255k_1)^2\right)\left(\sigma_X^2 + \sigma_Y^2 + (255k_2)^2\right)} \tag{25}$$



(a) (b) (c) (d)

FIGURE 2. (a) the "peppers" image, (b) the "peppers" image imposed with synthetic textured noise, (c) the cover page of a passbook, (d) the passbook page.

In this context, X represents the original image. Y represents the restored image; $\boldsymbol{u}_X$ and $\boldsymbol{u}_y$ represents the mean of images X and Y, respectively. $\boldsymbol{u}_X^2$ and $\boldsymbol{u}_Y^2$ represents the variance of images X and Y, respectively. σ_{XY} signifies the covariance between X and Y.

The stopping criteria for all models were standardized. It required that the relative difference between successive iterations of the denoised image satisfy the following inequality:

$$\frac{\|\boldsymbol{F}^{k+1} - \boldsymbol{F}^{k}\|_2}{\|\boldsymbol{F}^{k+1}\|_2} \leq 1 \times 10^{-4} \tag{26}$$

where $\boldsymbol{F}^k$ and $\boldsymbol{F}^{k+1}$ respectively represent the current iteration and the next iteration of an image.

The second experiment employed two realistic images (Figure 2(c) and 2(d)) to assess the denoising model's performance concerning the preservation of weak edges and fine details against various noise types, such as those arising from natural fabric texture and watermark patterns. Figure 2(c) displays a passbook cover featuring noise induced by the natural fabric texture and printed characters of varying sizes. Figure 2(d) exhibits an internal passbook page with a background adorned with watermark patterns, incorporating both printed and handwritten characters of different sizes. The contrast between some characters or foreground elements and the background texture in both Figure 2(c) and 2(d) is relatively low. To enhance the visualization of differences between the original and denoised images, an adaptive thresholding method [40] was applied for image binarization. This method aimed to standardize binarization across denoised images, utilizing a fixed kernel size and threshold value computed based on the local neighborhood. This approach ensured a more accurate reflection of the denoising effect on local regions within the image.

As there is no standard or noise-free reference image for Figure 2(c) and 2(d), conventional performance metrics like PSNR and SSIM, employed in the initial experiment, are not applicable. Consequently, line intensity profiles were extracted for in-depth analysis. By comparing these profiles at specific locations within the images before and after the denoising operation, it becomes possible to assess the effectiveness of preserving edges and fine details. Typically, small and repetitive fluctuation patterns observed in the horizontal direction within a line intensity profile represent the textured regions in an image. The abrupt changes in the line intensity profile signify the presence of edge features. Therefore, a smaller fluctuation in the line intensity profile within the textured region after denoising indicates improved texture elimination. Additionally, a larger value of the peak-to-valley height retained after denoising suggests better edge protection.

Several state-of-the-art denoising models, namely TV [7], TGV [10], TV-OGS [8], and TGV-OGS (ADMM) [35], were employed for comparative analysis of their denoising effects and ability to eliminate the staircase effect. These models and the proposed one were developed using MATLAB 2022b and implemented on a laptop computer with a processing CPU of 4.6 GHz and RAM of 16 GB.

B. COMPARISON OF ALGORITHM PERFORMANCE

Figure 3 displays enlarged image patches before and after denoising operations. Specifically, Figure 3(a) and 3(b) correspond to the original image patch in Figure 2(a) and the image patch with imposed synthetic texture in Figure 2(b). In this context, the group size K was standardized to three for all OGS-based models, as suggested in [8] for optimality. Figure 3(c) to 3(g) depict the texture removal effects of TV, TGV, TV-OGS, TGV-OGS (ADMM), and the proposed model, respectively, on the same image patches shown in Figure 2(b). While all TV- and TGV-based models can generally remove synthetic textures, the observed staircase effects vary across models. Notably, in the regions enclosed by a green rectangle, Figure 3(c) demonstrates a significant staircase effect when using the TV model for denoising. Figure 3(d) and 3(e), processed with TGV and TV-OGS models, respectively, still exhibit minor residual staircase effects in the green rectangular box area.

Simultaneously, slight remnants of texture noise persist, particularly within the region enclosed by the green rectangle. This observation indicates that TGV and TV-OGS exhibit a certain degree of efficacy in mitigating the staircase effect. However, these models fall short of complete removal of robust textures, leaving behind residual artifacts. In Figure 3(f) and 3(g), both images were denoised using

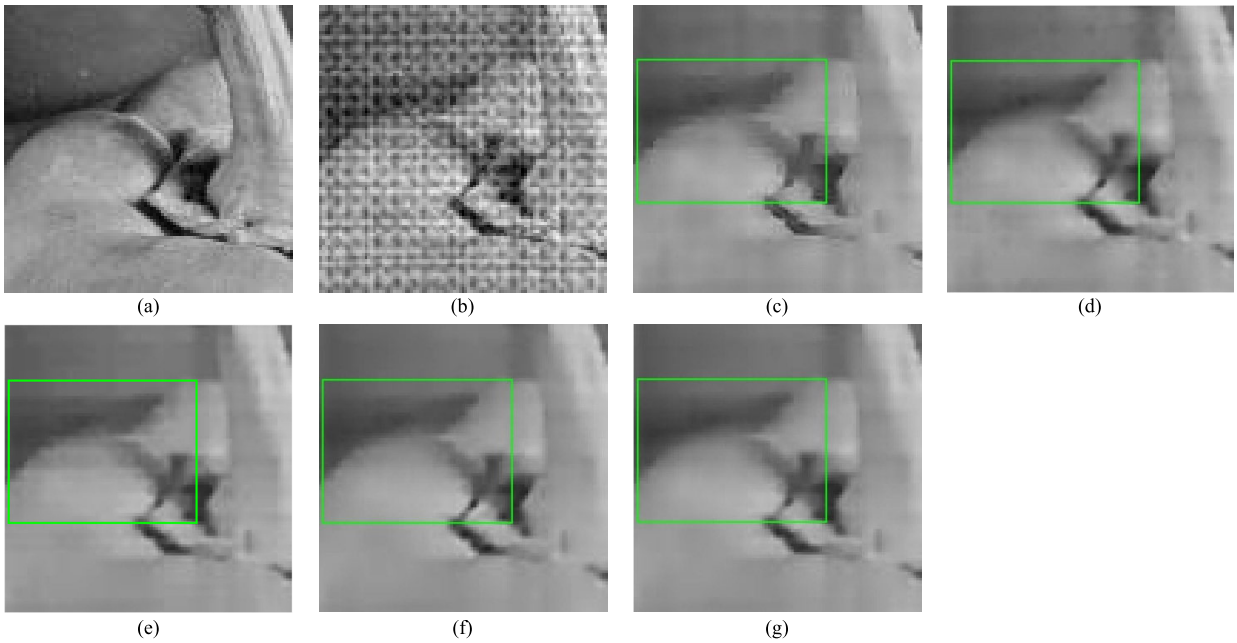


FIGURE 3. a) image of "peppers" patch enlarged without texture, (b) image of "peppers" patch enlarged with synthetic texture noise, (c) denoised image patch enlarged using TV model, (d) denoised image patch enlarged using TGV model, (e) denoised image patch enlarged using TV-OGS model, (f) denoised image patch enlarged using TGV-OGS(ADMM) model, (g) denoised image patch enlarged using TGV-OGS (split Bregman) model.

TABLE 1. Performance metrics of the different denoising models.

	split Bregman				ADMM			
Model	PSNR/dB	SSIM	Time/s	Iteration	PSNR/dB	SSIM	Time/s	Iteration
TV	22.21	0.7239	**0.449**	**24**	22.18	0.7231	0.480	**24**
TGV	22.96	0.7384	1.826	34	22.94	0.7385	2.789	38
TV-OGS	23.03	0.7375	1.382	31	23.02	0.7373	2.901	43
TGV-OGS	**23.53**	**0.7585**	6.045	49	23.51	0.7583	9.090	68

TGV-OGS models but with distinct solvers. Notably, the residual noise in the green rectangular area was effectively eliminated, while the presence or elimination of staircase artifacts was not apparent.

Table 1 presents a comprehensive overview of these models' performances in the conducted experiment, evaluating PSNR and SSIM concerning denoised images in comparison to the original image depicted in Figure 3(a). Additionally, the table includes the time and iteration number required for the denoising operations. The results indicate that TGV incorporating the OGS regularization term consistently achieved higher PSNR and SSIM values than their counterparts. This suggests that, despite the varied solvers employed by TGV-OGS models, they demonstrated more effective suppression of textural noises and background patterns, minimizing or eliminating staircase effects. However, due to the complexity of these models, their computational costs are relatively higher compared to TV, TGV, and TV-OGS models. Nevertheless, the proposed TGV-OGS model in this study surpasses the TGV-OGS with ADMM solvers in terms of processing time. The proposed model demonstrated a two-thirds reduction in processing time compared to the TGV-OGS with ADMM solver while maintaining similar denoising effects.

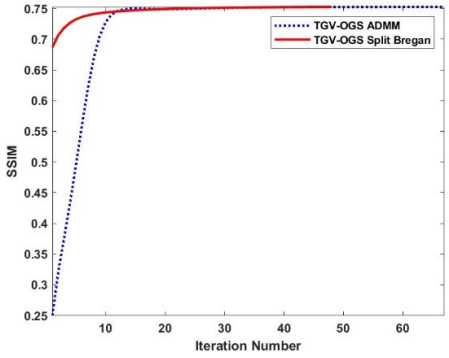


FIGURE 4. The convergence curves of TGV-OGS (ADMM) and TGV-OGS (split Bregman).

Moreover, the proposed model required fewer iterations than the TGV-OGS with the ADMM solver. According to Table 1, the average PSNR and SSIM values for the other three comparison models are 22.73dB and 0.7333, respectively. The PSNR and SSIM values for the proposed model surpass the average values by 0.8dB (3.5%) and 0.0252 (3.4%), respectively.

The convergence curves depicted in Figure 4 were utilized to assess the efficiency of the TGV-OGS model employing

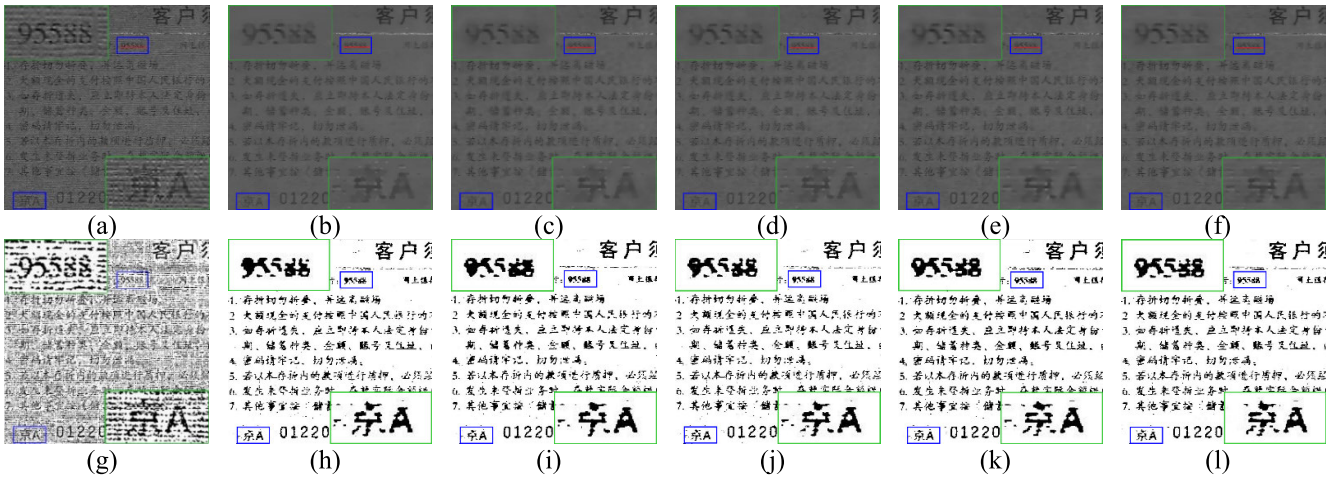


FIGURE 5. (a) Original image of a passbook cover and (b) denoised image using TV model, (c) denoised image using TGV model, (d) denoised image using TV-OGS model, (e) denoised image using TGV-OGS (ADMM) model, (f) denoised image using TGV-OGS (split Bregman) model, (g)- (l) are binarized version of image 6(a)-(f).

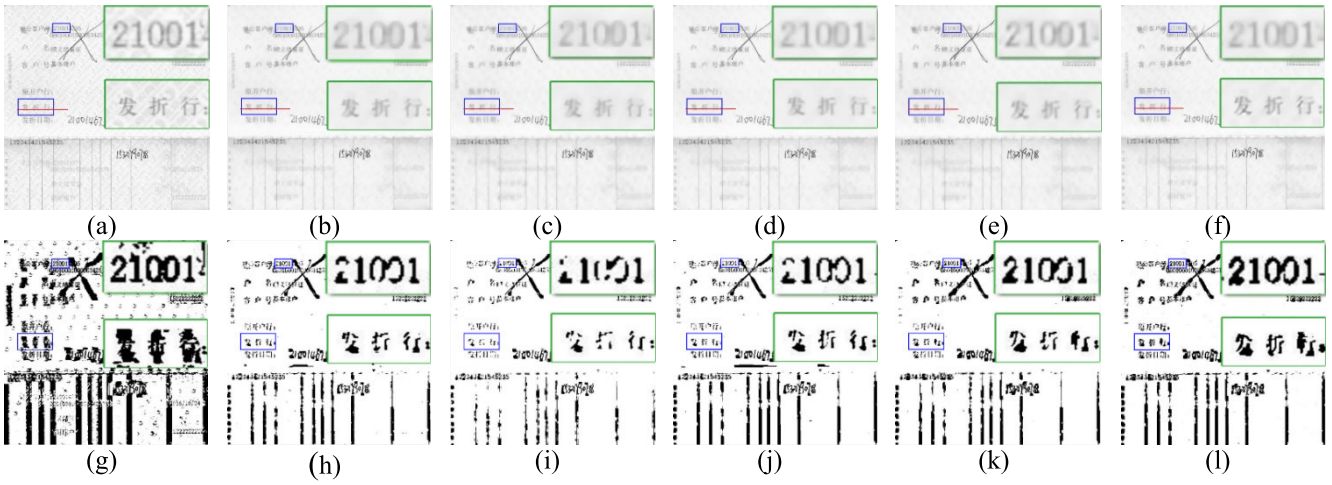


FIGURE 6. (a) Original image of a passbook page and (b) denoised image using TV model, (c) denoised image using TGV model, (d) denoised image using TV-OGS model, (e) denoised image using TGV-OGS (ADMM) model, (f) denoised image using TGV-OGS (split Bregman) model, (g)- (l) are binarized version of image (a)-(f).

the ADMM solver in comparison to the proposed model utilizing a fast split Bregman solver. Notably, it is observed that the TGV-OGS model, when integrated with the fast split Bregman solver, exhibits a superior convergence speed compared to its ADMM counterpart. Specifically, the TGV-OGS with a fast split Bregman solver demonstrates a more advantageous starting point for convergence, registering 0.7 in contrast to the ADMM solver's starting point of 0.25. Consequently, the TGV-OGS model with a fast split Bregman solver requires less time and fewer iterations to achieve convergence.

Figure 5 and 6 depict the denoising effects applied to the images in Figure 2(c) and 2(d) using various models, including TV, TGV, TV-OGS, TGV-OGS (ADMM), and the proposed model. Specifically, Figure 5(a) presents the original cover page image of a passbook, while Figure 5(b) to 5(f) exhibit the denoising outcomes achieved by various models, respectively. Additionally, Figure 6(a) displays the original image of an internal passbook page with background patterns, and Figure 6(b) to 6(f) showcase the denoising results using various models, respectively. Visual differentiation of denoising effects among the models is challenging; hence, binarized versions were generated using the thresholding method [40]. Figure 5(g) to 5(l) represent the binarized versions corresponding to Figure 5(a) to 5(f), while Figure 6(g) to 6(l) depict the binarized versions for Figure 6(a) to 6(f), respectively.

Figure 5(g) reveals the binary representation of the original passbook cover, exhibiting a substantial presence of textural noise necessitating removal. Despite the denoising operations carried out by various models, which effectively mitigated textural noises, complete elimination was not achieved.

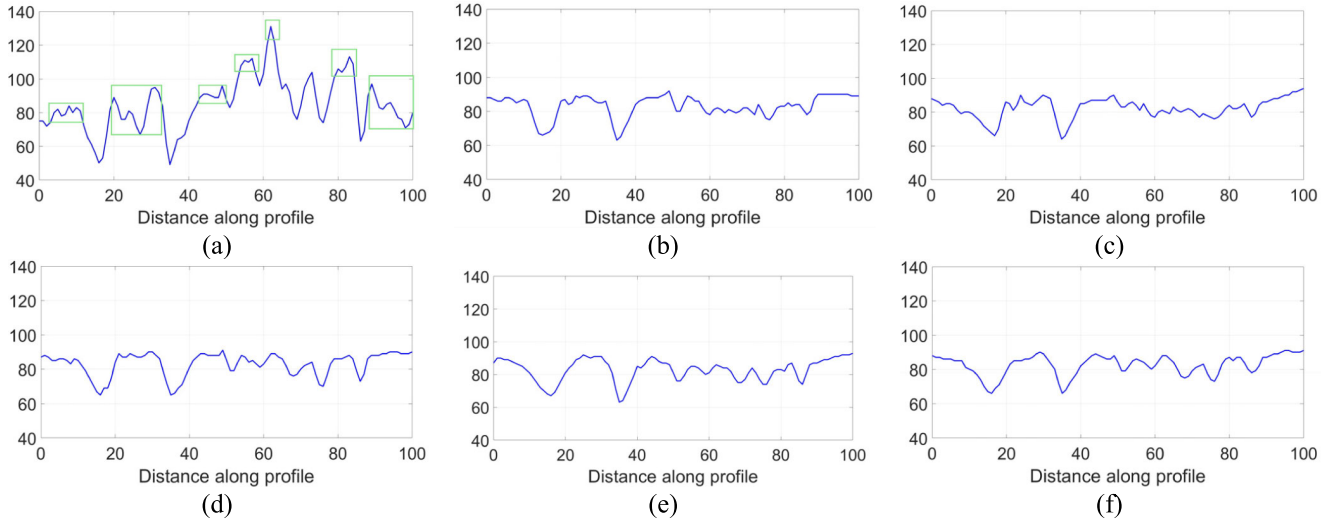


FIGURE 7. Line profile of a passbook cover and (b) line profile of the denoised image using TV model, (c) line profile of the denoised image using TGV model, (d) line profile of the denoised image using TV-OGS model, (e) line profile of the denoised image using TGV-OGS (ADMM) model, (f) line profile of the denoised image using TGV-OGS (split Bregman) mode.

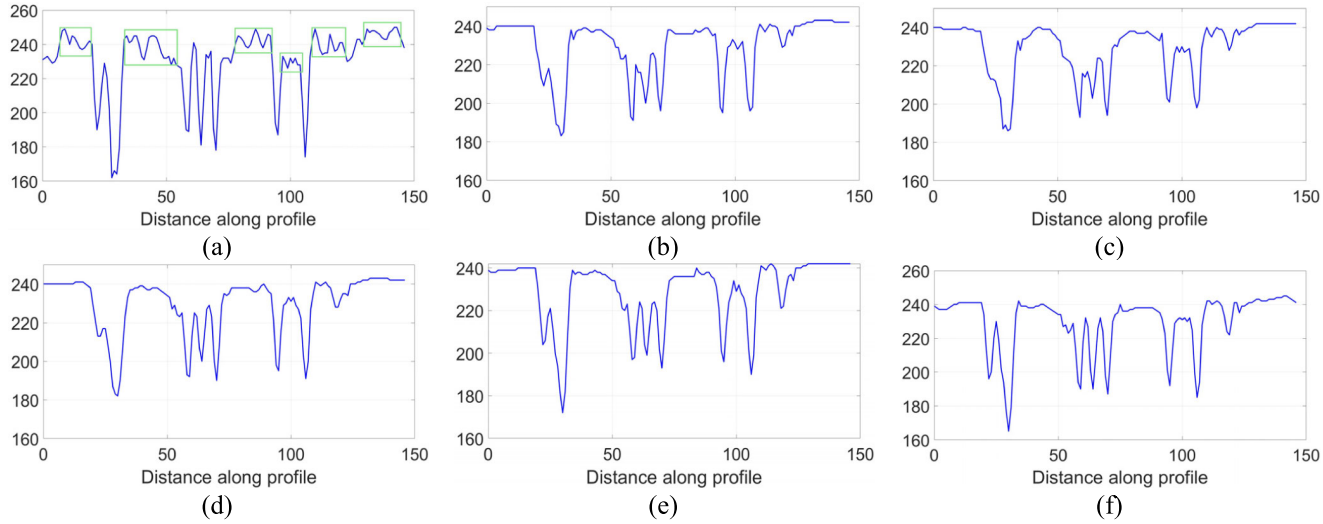


FIGURE 8. (a) Line profile of a passbook page and (b) line profile of the denoised image using TV model, (c) line profile of the denoised image using TGV model, (d) line profile of the denoised image using TV-OGS model, (e) line profile of the denoised image using TGV-OGS (ADMM) model, (f) line profile of the denoised image using TGV-OGS (split Bregman) mode.

Figure 5(h) to 5(l) emphasize the presence of minimal residual noise. Notably, the consistent level of residual noise across all images indicates the uniform impact of denoising models in addressing textural noise. This observation underscores the overall effectiveness of the denoising models in managing textural noise

Examining the characters "京**A**" and "**95588**" enclosed in the blue rectangle in denoised images, along with their enlarged versions, allows for the observation and comparison of the edge protection capabilities of each denoising model. Generally, TGV-OGS models exhibit superior edge protection capabilities compared to other denoising models, as illustrated in Figure 5(k) and 5(l) in contrast to Figure 5(h) – 5(j). Notably, the TGV model displays the least edge protection capability, as depicted in Figure 5(i), where the images reveal the most disruptions in the characters.

Similar observations can be made regarding the characters "**21001**" and "发折行" enclosed in the blue rectangle in Figure 6(h) – 6(i), and their enlarged versions in binarized images after denoising. The TGV-OGS models demonstrate superiority over other denoising models in edge protection and textured noise removal.

Figure 5 and 6 illustrate that the smaller the character size, the more challenging the edge protection, and the poorer the performance of each model in protection. Line intensity profiles were generated from each image to delve further into the denoising effect of each model and validate the observations. The red lines in each denoised image in Figure 5

and 6 represent the line intensity profiles extracted for texture removal observation (Figure 7) and background pattern removal (Figure 8)

In Figure 7(a) and 8(a), regions containing textures and background patterns in the image patches are delineated with green rectangular boxes. These patterns significantly compromise the clarity of characters within the designated area. A comparative analysis of the plots in Figure 7(e) and 7(f), derived from images denoised by TGV-OGS models, with those in Figure 7(b) - 7(d), obtained from images processed by alternative models, reveals a reduction in residual texture noises, as indicated by the diminished high-frequency fluctuations within the regions. Moreover, Figure 7(e), 7(f), 8(e), and 8(f) demonstrate that character edges, identifiable as abrupt changes in intensity profiles, are well preserved, exhibiting higher peak-to-valley height values. These values tend to be lower in plots extracted from images denoised by other models, given their reliance on individual pixels as processing units and iterative isolation. Notably, the proposed model addresses and rectifies this weakness. The character edges exhibit a reduced peak-to-valley height value in the plots of Figure 7(c) and 8(c) due to the TGV regularization term, which imparts more pronounced smoothing effects compared to the TV regularization term. Consequently, weak and fine edges are more susceptible to blurring in this context.

The foregoing results and observations lead to the conclusion that both TGV-OGS denoising models adeptly eliminate textural and background patterns deemed as noise. Overall, they exhibit superior performance in terms of residual noise reduction and protection capability compared to other TV and TGV models. Moreover, as depicted in Table 1 and Figure 4, the proposed TGV-OGS model with the fast split Bregman solver demonstrates significant enhancements in speed and efficiency in contrast to its counterpart utilizing the ADMM solver, all while maintaining the anticipated performance levels in denoising and edge protection.

V. CONCLUSION

This work proposes and demonstrates a combined second-order TGV-OGS model for image denoising. Leveraging the L1 norm property in the energy equation of the proposed model, we employ a fast split Bregman iterative algorithm to tackle L1 norm regularization. This transformation facilitates the resolution of the original problem through a series of sub-problems, each with easily solvable and closed solutions. The efficiency and denoising performance of the proposed model surpasses those of existing models, particularly in the removal of textural and background patterns. The experimental findings yield the following conclusions: (1) The proposed model enhances the time efficiency of image denoising and computational efficiency compared to TGV-OGS with the ADMM solver. (2) The denoising efficacy of the proposed model outperforms state-of-the-art variational denoising models, especially for images marked by substantial noise and textured regions. One drawback of the proposed model is the manual determination required for the regularization parameter. Further research could focus on the study of mechanisms or algorithms to adaptively adjusting these parameters based on the analysis of diverse texture characteristics in images.

APPENDIX A

We separately employ the ADMM and split Bregman solvers to address (1), aiming to illustrate the distinctions between them.

Firstly, the steps for solving with ADMM are as follows. Introducing auxiliary variables $\boldsymbol{d}_x, \boldsymbol{d}_y, \boldsymbol{v}$. (1) can be written as follows.

$$\begin{aligned} &\underset{\boldsymbol{d}_x,\boldsymbol{d}_y,\boldsymbol{u},\boldsymbol{v}}{\operatorname{argmin}} \lambda \left(\|\boldsymbol{d}_x\|_1 + \left\|\boldsymbol{d}_y\right\|_1\right) + \frac{1}{2}\|\boldsymbol{v}-\boldsymbol{f}\|^2 \\ &\text{s.t.} \quad \boldsymbol{d_x} = \nabla_h \boldsymbol{u}, \boldsymbol{d_y} = \nabla_v \boldsymbol{u}, \boldsymbol{v} = \boldsymbol{u}. \end{aligned} \tag{27}$$

The Lagrangian multipliers and quadratic penalty terms are needed to establish the augmented Lagrangian function. Therefore, we have the following:

$$\begin{aligned} &\mathcal{L}\left(\boldsymbol{d}_x, \boldsymbol{d}_y, \boldsymbol{u}, \boldsymbol{v}, \mu\right) \\ &\equiv \frac{1}{2}\|\boldsymbol{v}-\boldsymbol{f}\|^2 + \lambda\left(\|\boldsymbol{d}_x\|_1 + \|\boldsymbol{d}_y\|_1\right) \\ &\quad + \frac{\mu}{2}\|\boldsymbol{u}-\boldsymbol{v}-\boldsymbol{b}\|^2 + \frac{\mu}{2} \\ &\quad \left(\|\boldsymbol{d}_x - \nabla_h \boldsymbol{u} - \boldsymbol{b}_x\|^2 + \|\boldsymbol{d}_y - \nabla_v \boldsymbol{u} - \boldsymbol{b}_y\|^2\right) \end{aligned} \tag{28}$$

where μ is a non-negative penalty parameter. $\boldsymbol{b}, \boldsymbol{b}_x$, and $\boldsymbol{b}_y$ are the Lagrange multipliers. The solution for the sub-problem u can be expressed as follows.

$$\begin{aligned} &\boldsymbol{u}^{(k+1)} \\ &= \underset{\boldsymbol{u}}{argmin} \frac{1}{2}\|\boldsymbol{v}-\boldsymbol{f}\|_2^2 \\ &\quad + \frac{\mu}{2}\left(\left\|\boldsymbol{d}_x{}^{\mathrm{k}} - \nabla_h \boldsymbol{u} - \boldsymbol{b}_x{}^{\mathrm{k}}\right\|^2 + \left\|\boldsymbol{d}_y{}^{\mathrm{k}} - \nabla_v \boldsymbol{u} - \boldsymbol{b}_y{}^{\mathrm{k}}\right\|^2\right) \end{aligned} \tag{29}$$

The solution for the sub-problem $\boldsymbol{v}$ can be expressed as follows.

$$\boldsymbol{v}^{(\mathrm{k}+1)} = \left\{\frac{1}{2}\|\boldsymbol{v}-\boldsymbol{f}\|_2^2 + \frac{\mu}{2}\|\boldsymbol{u}^{\mathrm{k}} - \boldsymbol{v} + \boldsymbol{b}^{\mathrm{k}}\|^2\right\} \tag{30}$$

The solution for the sub-problem $\boldsymbol{d}_x$ and $\boldsymbol{d}_y$ can be expressed as follows.

$$\begin{cases} \boldsymbol{d}_x{}^{(\mathrm{k}+1)} = \underset{\boldsymbol{d}_x}{\operatorname{argmin}} \lambda \|\boldsymbol{d}_x\|_1 + \frac{\mu}{2}\left\|\boldsymbol{d}_x - \nabla_h \boldsymbol{u}^{\mathrm{k}} - \boldsymbol{b}_x{}^{\mathrm{k}}\right\|^2 \\ \boldsymbol{d}_y{}^{(\mathrm{k}+1)} = \underset{\boldsymbol{d}_y}{\operatorname{argmin}} \lambda \left\|\boldsymbol{d}_y\right\|_1 + \frac{\mu}{2}\left\|\boldsymbol{d}_y - \nabla_v \boldsymbol{u}^{\mathrm{k}} - \boldsymbol{b}_y{}^{\mathrm{k}}\right\|^2 \end{cases} \tag{31}$$

The solution for the sub-problem $\boldsymbol{b}, \boldsymbol{b}_x$, and $\boldsymbol{b}_y$ can be mathematically expressed as follows.

$$\begin{cases} \boldsymbol{b}^{(\mathrm{k}+1)} = \boldsymbol{b}^{(\mathrm{k})} + \boldsymbol{v}^{(\mathrm{k}+1)} - \boldsymbol{u}^{(\mathrm{k}+1)} \\ \boldsymbol{b}_x{}^{(\mathrm{k}+1)} = \boldsymbol{b}_x{}^{(\mathrm{k})} + \nabla_h \boldsymbol{u}^{(\mathrm{k}+1)} - \boldsymbol{d}_x{}^{(\mathrm{k}+1)} \\ \boldsymbol{b}_y{}^{(\mathrm{k}+1)} = \boldsymbol{b}_y{}^{(\mathrm{k})} + \nabla_v \boldsymbol{u}^{(\mathrm{k}+1)} - \boldsymbol{d}_y{}^{(\mathrm{k}+1)} \end{cases} \tag{32}$$

The following section describes the steps for solving (1) using the split Bregman method. Firstly, introducing auxiliary variables $\boldsymbol{d}_x, \boldsymbol{d}_y$. (1) can be written as follows.

$$\begin{aligned} &\underset{\boldsymbol{d}_x,\boldsymbol{d}_y,\boldsymbol{u},\boldsymbol{v}}{\operatorname{argmin}} \lambda \left(\|\boldsymbol{d}_x\|_1 + \left\|\boldsymbol{d}_y\right\|_1\right) + \frac{1}{2}\|\boldsymbol{u}-\boldsymbol{f}\|^2 \\ &\text{s.t.} \quad \boldsymbol{d}_x = \nabla_h \boldsymbol{u}, \boldsymbol{d}_y = \nabla_v \boldsymbol{u} \end{aligned} \tag{33}$$

The Lagrangian multipliers and quadratic penalty terms are needed to establish the augmented Lagrangian function. Therefore, we have the following:

$$\begin{aligned} &\mathcal{L}\left(\boldsymbol{d}_x, \boldsymbol{d}_y, \boldsymbol{u}, \mu\right) \\ &\equiv \frac{1}{2}\|\boldsymbol{u}-\boldsymbol{f}\|^2 + \lambda\left(\|\boldsymbol{d}_x\|_1 + \|\boldsymbol{d}_y\|_1\right) \\ &\quad + \frac{\mu}{2}\left(\|\boldsymbol{d}_x - \nabla_h \boldsymbol{u} - \boldsymbol{b}_x\|^2 + \|\boldsymbol{d}_y - \nabla_v \boldsymbol{u} - \boldsymbol{b}_y\|^2\right) \end{aligned} \tag{34}$$

where μ is a non-negative penalty parameter. $\boldsymbol{b}_x$, and $\boldsymbol{b}_y$ are the Lagrange multipliers. The solution for the sub-problem u can be expressed as follows.

$$\begin{aligned} \boldsymbol{u}^{(k+1)} = &\underset{\boldsymbol{u}}{argmin}\ \frac{1}{2}\|\boldsymbol{u}-\boldsymbol{f}\|_2^2 \\ &+ \frac{\mu}{2}\left(\|\boldsymbol{d}_x^{\mathrm{k}} - \nabla_h \boldsymbol{u} - \boldsymbol{b}_x^{\mathrm{k}}\|^2 + \|\boldsymbol{d}_y^{\mathrm{k}} - \nabla_v \boldsymbol{u} - \boldsymbol{b}_y^{\mathrm{k}}\|^2\right) \end{aligned} \tag{35}$$

The solution for the sub-problem $\boldsymbol{d}_x$ and $\boldsymbol{d}_y$ can be expressed as follows.

$$\begin{cases} \boldsymbol{d_x}^{(\mathrm{k}+1)} = \underset{\boldsymbol{d_x}}{\operatorname{argmin}} \lambda \|\boldsymbol{d}_x\|_1 + \dfrac{\mu}{2}\left\|\boldsymbol{d_x} - \nabla_h \boldsymbol{u}^{\mathrm{k}} - \boldsymbol{b_x}^{\mathrm{k}}\right\|^2 \\ \boldsymbol{d}_y{}^{(\mathrm{k}+1)} = \underset{\boldsymbol{d}_y}{\operatorname{argmin}} \lambda \left\|\boldsymbol{d}_y\right\|_1 + \dfrac{\mu}{2}\left\|\boldsymbol{d}_y - \nabla_v \boldsymbol{u}^{\mathrm{k}} - \boldsymbol{b}_y{}^{\mathrm{k}}\right\|^2 \end{cases} \tag{36}$$

The solution for the sub-problem $\boldsymbol{b}, \boldsymbol{b}_x$, and $\boldsymbol{b}_y$ can be mathematically expressed as follows.

$$\begin{cases} \boldsymbol{b}_x{}^{(\mathrm{k}+1)} = \boldsymbol{b}_x{}^{(\mathrm{k})} + \nabla_h \boldsymbol{u}^{(\mathrm{k}+1)} - \boldsymbol{d}_x{}^{(\mathrm{k}+1)} \\ \boldsymbol{b}_y{}^{(\mathrm{k}+1)} = \boldsymbol{b}_y{}^{(\mathrm{k})} + \nabla_v \boldsymbol{u}^{(\mathrm{k}+1)} - \boldsymbol{d}_y{}^{(\mathrm{k}+1)}. \end{cases} \tag{37}$$

By comparing the two solvers employed for solving the TV model, it becomes evident that ADMM integrates the data fidelity and regularization terms during iteration, whereas split Bregman separates the L1 regularization and L2 data fidelity terms in the TV model. This separation contributes to a simpler and more efficient solution.

REFERENCES

[1] L. Y. Wei, S. Lefebvre, V. Kwatra, and G. Turk, "State of the art in example-based texture synthesis," in *Proc. Annu. Conf. Eur. Assoc. Comput. Graph.*, 2009, pp. 93–117.

[2] Y. Liang, K. Xu, P. Zhou, and D. Zhou, "Automatic defect detection of texture surface with an efficient texture removal network," *Adv. Eng. Inform.*, vol. 53, Aug. 2022, Art. no. 101672, doi: 10.1016/j.aei.2022.101672.

[3] R. Xu, Y. Xu, Y. Quan, and H. Ji, "Cartoon-texture image decomposition using orientation characteristics in patch recurrence," *SIAM J. Imag. Sci.*, vol. 13, no. 3, pp. 1179–1210, Jan. 2020, doi: 10.1137/19m128898x.

[4] K. Xu, H. Chen, C. Xu, Y. Jin, and C. Zhu, "Structure-texture aware network for low-light image enhancement," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 32, no. 8, pp. 4983–4996, Aug. 2022, doi: 10.1109/TCSVT.2022.3141578.

[5] Y. Han, S. Yang, and Q. Chen, "Recognition and segmentation of complex texture images based on superpixel algorithm and deep learning," *Comput. Mater. Sci.*, vol. 209, Jun. 2022, Art. no. 111398, doi: 10.1016/j.commatsci.2022.111398.

[6] T. Wu, X. Gu, Y. Wang, and T. Zeng, "Adaptive total variation based image segmentation with semi-proximal alternating minimization," *Signal Process.*, vol. 183, Jun. 2021, Art. no. 108017, doi: 10.1016/j.sigpro.2021.108017.

[7] L. I. Rudin, S. Osher, and E. Fatemi, "Nonlinear total variation based noise removal algorithms," *Phys. D, Nonlinear Phenomena*, vol. 60, nos. 1–4, pp. 259–268, Nov. 1992, doi: 10.1016/0167-2789(92)90242-F.

[8] J. Liu, T.-Z. Huang, I. W. Selesnick, X.-G. Lv, and P.-Y. Chen, "Image restoration using total variation with overlapping group sparsity," *Inf. Sci.*, vol. 295, pp. 232–246, Feb. 2015, doi: 10.1016/j.ins.2014.10.041.

[9] H. Liu, R. Xiong, S. Ma, X. Fan, and W. Gao, "Non-local extension of total variation regularization for image restoration," in *Proc. IEEE Int. Symp. Circuits Syst. (ISCAS)*, Melbourne, VIC, Australia, Jun. 2014, pp. 1102–1105, doi: 10.1109/ISCAS.2014.6865332.

[10] K. Bredies, K. Kunisch, and T. Pock, "Total generalized variation," *SIAM J. Imag. Sci.*, vol. 3, no. 3, pp. 492–526, Sep. 2010, doi: 10.1137/090769521.

[11] T. Adam and R. Paramesran, "Image denoising using combined higher order non-convex total variation with overlapping group sparsity," *Multidimensional Syst. Signal Process.*, vol. 30, no. 1, pp. 503–527, Jan. 2019, doi: 10.1007/s11045-018-0567-3.

[12] C. Lu and M. Wang, "Alternating direction method for TGV-TGV based cartoon-texture image decomposition," *IET Image Process.*, vol. 10, no. 6, pp. 495–504, Jun. 2016, doi: 10.1049/iet-ipr.2015.0573.

[13] X. Liu, "A new TGV-Gabor model for cartoon-texture image decomposition," *IEEE Signal Process. Lett.*, vol. 25, no. 8, pp. 1221–1225, Aug. 2018, doi: 10.1109/LSP.2018.2850218.

[14] Y. Gao and K. Bredies, "Infimal convolution of oscillation total generalized variation for the recovery of images with structured texture," *SIAM J. Imag. Sci.*, vol. 11, no. 3, pp. 2021–2063, Jan. 2018.

[15] P.-Y. Chen and I. W. Selesnick, "Group-sparse signal denoising: Non-convex regularization, convex optimization," *IEEE Trans. Signal Process.*, vol. 62, no. 13, pp. 3464–3478, Jul. 2014, doi: 10.1109/TSP.2014.2329274.

[16] P.-Y. Chen and I. W. Selesnick, "Translation-invariant shrinkage/thresholding of group sparse signals," *Signal Process.*, vol. 94, pp. 476–489, Jan. 2014, doi: 10.1016/j.sigpro.2013.06.011.

[17] I. Selesnick and M. Farshchian, "Sparse signal approximation via non-separable regularization," *IEEE Trans. Signal Process.*, vol. 65, no. 10, pp. 2561–2575, May 2017, doi: 10.1109/TSP.2017.2669904.

[18] X. Ningshan, W. Chen, R. Guoqiang, and H. Yongmei, "Blind image restoration method regularized by hybrid gradient sparse prior," *Opto-Electron. Eng.*, vol. 48, no. 6, 2021, Art. no. 210040.

[19] T. Goldstein and S. Osher, "The split Bregman method for L_1-regularized problems," *SIAM J. Imag. Sci.*, vol. 2, no. 2, pp. 323–343, Jan. 2009, doi: 10.1137/080725891.

[20] W. Yin, S. Osher, D. Goldfarb, and J. Darbon, "Bregman iterative algorithms for ℓ_1-minimization with applications to compressed sensing," *SIAM J. Imag. Sci.*, vol. 1, no. 1, pp. 143–168, Jan. 2008, doi: 10.1137/070703983.

[21] Y. Wang, J. Yang, W. Yin, and Y. Zhang, "A new alternating minimization algorithm for total variation image reconstruction," *SIAM J. Imag. Sci.*, vol. 1, no. 3, pp. 248–272, Jan. 2008, doi: 10.1137/080724265.

[22] X. Ren, H. Pan, Z. Jing, and L. Gao, "Multi-image restoration method combined with total generalized variation and lp-norm regularizations," *J. Shanghai Jiaotong Univ.*, vol. 24, no. 5, pp. 551–558, Oct. 2019, doi: 10.1007/s12204-019-2113-3.

[23] X. Kong, "An image foreground and background segmentation model based on improved second order total generalized variation," *Comput. Eng. Sci.*, vol. 41, no. 5, p. 865, 2019.

[24] S. Wali, H. Zhang, H. Chang, and C. Wu, "A new adaptive boosting total generalized variation (TGV) technique for image denoising and inpainting," *J. Vis. Commun. Image Represent.*, vol. 59, pp. 39–51, Feb. 2019, doi: 10.1016/j.jvcir.2018.12.047.

[25] S. Li, B. Zhang, X. Yang, and W. Zhu, "Edge-guided second-order total generalized variation for Gaussian noise removal from depth map," *Sci. Rep.*, vol. 10, no. 1, p. 16329, Oct. 2020, doi: 10.1038/s41598-020-73342-3.

[26] X. Ban, Z. Li, B. Li, and M. Xu, "Sparse image reconstruction based on improved total generalized variation," *Laser Optoelectronics Prog.*, vol. 55, no. 11, 2018, Art. no. 111103, doi: 10.3788/lop55.111103.

[27] B. Xie, H. Huang, and A. Huang, "Image deblurring based on fractional-order total variation and total generalized variation," *J. Phys., Conf.*, vol. 1345, no. 2, Nov. 2019, Art. no. 022006, doi: 10.1088/1742-6596/1345/2/022006.

[28] K. Bredies and T. Valkonen, "Inverse problems with second-order total generalized variation constraints," in *Proc. SampTA*, vol. 201, 2011, pp. 1–4.

[29] W. Guo, J. Qin, and W. Yin, "A new detail-preserving regularization scheme," *SIAM J. Imag. Sci.*, vol. 7, no. 2, pp. 1309–1334, 2014, doi: 10.1137/120904263.

[30] X. Liu, Y. Chen, Z. Peng, and J. Wu, "Infrared image super-resolution reconstruction based on quaternion and high-order overlapping group sparse total variation," *Sensors*, vol. 19, no. 23, p. 5139, Nov. 2019, doi: 10.3390/s19235139.

[31] Y. Sun, P. Babu, and D. Palomar, "Majorization-minimization algorithms in signal processing, communications, and machine learning," *IEEE Trans. Signal Process.*, vol. 65, no. 3, pp. 794–816, Feb. 2016, doi: 10.1109/TSP.2016.2601299.

[32] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Found. Trends Mach. Learn.*, vol. 3, no. 1, pp. 1–122, 2011, doi: 10.1561/9781601984616.

[33] F. Q. Ren and T. S. Qiu, "A blurred image restoration method based on second-order total generalized variation regularization," *Zidonghua Xuebao/Acta Automatica Sinica*, vol. 41, no. 6, pp. 1166–1172, 2015, doi: 10.16383/j.aas.2015.c130616.

[34] Y. Chen, L. Wu, Z. Peng, and X. Liu, "Fast overlapping group sparsity total variation image denoising based on fast Fourier transform and split Bregman iterations," in *Proc. 7th Int. Workshop Comput. Sci. Eng.*, 2017, pp. 25–27, doi: 10.18178/wcse.2017.06.048.

[35] Y. Chen, Z. Peng, M. Li, F. Yu, and F. Lin, "Seismic signal denoising using total generalized variation with overlapping group sparsity in the accelerated ADMM framework," *J. Geophys. Eng.*, vol. 16, no. 1, pp. 30–51, Feb. 2019, doi: 10.1093/jge/gxy003.

[36] H. Nien and J. A. Fessler, "A convergence proof of the split Bregman method for regularized least-squares problems," 2014, *arXiv:1402.4371*.

[37] R. Nishihara, L. Lessard, B. Recht, A. Packard, and M. I. Jordan, "A general analysis of the convergence of ADMM," 2015, *arXiv:1502.02009*.

[38] S. Ono and I. Yamada, "Second-order total generalized variation constraint," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, May 2014, pp. 4938–4942, doi: 10.1109/ICASSP.2014.6854541.

[39] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: From error visibility to structural similarity," *IEEE Trans. Image Process.*, vol. 13, no. 4, pp. 600–612, Apr. 2004, doi: 10.1109/TIP.2003.819861.

[40] D. Bradley and G. Roth, "Adaptive thresholding using the integral image," *J. Graph. Tools*, vol. 12, no. 2, pp. 13–21, 2007, doi: 10.1080/2151237X.2007.10129236.

CHENG ZHANG received the B.Eng. degree in mechanical engineering from Inner Mongolia Agricultural University, in 2009. He is currently pursuing the Ph.D. degree in the field of machine vision and artificial intelligence with Universiti Sains Malaysia.

KIN SAM YEN received the Ph.D. degree from Universiti Sains Malaysia, in 2012. He is currently a Lecturer with the School of Mechanical Engineering, Universiti Sains Malaysia. His current research interests include machine vision and application, image processing and analysis, and pattern recognition.