A Mathematical Algorithm for Improving the Medical Image

ANANTACHAI PADCHAROEN, DUANGKAMON KITKUAN*
Department of Mathematics, Faculty of Science and Technology,
Rambhai Barni Rajabhat University,
Chanthaburi 22000,
THAILAND
*Corresponding author

Abstract: -In this paper, we present a hybrid model based on total generalized variation (TGV) and shearlet with non-quadratic fidelity data terms for blurred images corrupted by impulsive and Poisson noises. Numerical experiments demonstrate that the proposed can reduce the staircase effect while preserving edges and outperform classical TV-based models in the peak signal-to-noise ratio (PSNR).

Key-Words: Total generalized variation (TGV), image restoration, Mathematical algorithm, Medical Image, non-quadratic fidelity

Received: August 19, 2023. Revised: February 6, 2024. Accepted: March 3, 2024. Published: April 25, 2024.

1 Introduction

Image restoration is an inverse problem where the objective is to recover a sharp image from a blurry and noisy observation. Mathematically, a linear shift-invariant imaging of the system is modeled as

$$b = Au + c, \tag{1}$$

where b is the observed image, u is the unknown image, matrix A is a linear transformation representing convolution operation and c is the noise. The goal of image restoration is to u recover from b.

The structured matrix A has many singular values of different orders of magnitude close to the origin. In particular, A is severely ill-conditioned and may be singular. This makes the solution of (1) be very sensitive to the noise c in the right-hand side b. In general, a regularization method can be employed to compute the approximate solutions that are less sensitive to noise than the naive solution. Probably one of the most popular regularization methods is Tikhonov regularization, [1], which replaces (1) by the minimization problem

$$\min_u \{\|Au - b\|_2^2 + \alpha\|u\|_2^2\}, \tag{2}$$

where $\|\cdot\|_2$ denotes the Euclidean norm and α is the regularization parameter that controls the balance between the two terms for minimization.

In fact, Tikhonov regularization estimate is similar to low pass filtering, therefore, it produces a smoothing effect on the restored image, i.e., it penalizes edges, which is not a good approximation of the original image if it contains edges. To overcome this shortcoming, [2], proposed a total variation (TV) based regularization technique, which preserved the edge information in the restored image. In the case of TV regularization, the estimated solution is obtained by minimizing the objective function (the ROF model)

$$\min_u \{\|Au - b\|_2^2 + \alpha\|u\|_{TV}\}, \tag{3}$$

and, [3], studied l_1-TV denoising model:

$$\min_u \{\alpha\|Au - b\|_1 + \|u\|_{TV}\}, \tag{4}$$

where $\|\cdot\|_1 = \sum_i |x_i|$ and $\|u\|_{TV}$ is the discrete TV regularization term. Several efforts have been made to improve the TV output, [4], [5], [6], [7], [8], [9].

2 Second order TGV (TGV^2) and algorithm

Since optimization solution with total variational filter is very effective for preserving sharp edges, corners and other fine details of an image. However, it also has some disadvantages, most notably the so-called staircase effect, which is the unwanted occurrence of edges.

Recently, [10], developed the total generalized variation (TGV) regularizer, which is assumed to be the generalization of the total variational filter. Total generalized variation consists and balances higher-order derivatives of u. We are introducing some properties of the second order TGV which is given by

$$\mathrm{TGV}_\alpha^2(u) = \sup\Big\{ \int_\Omega u\,\mathrm{div}^2 v\,dx \,|\, v \in \mathcal{C}_c^2(\Omega, \mathcal{S}^{2\times 2}), \\ \|v\|_\infty \le \alpha_0, \|\mathrm{div} v\|_\infty \le \alpha_1 \Big\}, \tag{5}$$

where $S^{2\times 2}$ is the set of all symmetric matrices and $\mathcal{C}_c^2(\Omega, S^{2\times 2})$ denotes the space of compactly supported. The divergences $\mathrm{div} v \in \mathcal{C}_c^2(\Omega, \mathbb{R}^2)$ and

$\mathrm{div}^2 v \in \mathcal{C}_c^2(\Omega, \mathbb{R})$ are defined by

$$(\mathrm{div} v)_i = \sum_{j=1}^{2} \frac{\partial v_{ij}}{\partial x_j} \quad \text{and} \quad \mathrm{div}^2 v = \sum_{i,j=1}^{2} \frac{\partial^2 v_{ij}}{\partial x_i \partial x_j}.$$

In order to simplify the computation, we give the discretization of TGV_α^2. Firstly, we denote $\mathcal{U} = \mathcal{C}_c^2(\Omega, \mathbb{R})$, $\mathcal{V} = \mathcal{C}_c^2(\Omega, \mathbb{R}^2)$ and $\mathcal{W} = \mathcal{C}_c^2(\Omega, S^{2\times 2})$. According to [27], [29], the discretized TGV_α^2 is approximatively rewritten as the following minimization:

$$TGV_\alpha^2(u) = \min_{w\in\mathcal{V}} \alpha_1 \|Du - w\|_1 + \alpha_0 \|\xi(w)\|_1, \quad (6)$$

where $\xi(w) = \frac{1}{2}(Dw + (Dw)^T)$, $D = (D_1, D_2)$, D_1 and D_2 represent two first-order forward finite difference operators along the horizontal and vertical directions, respectively. Here, the operations $D : \mathcal{U} \to \mathcal{V}$ and $\xi : \mathcal{V} \to \mathcal{W}$ are written as

$$Du = \begin{pmatrix} D_1 \\ D_2 \end{pmatrix} \quad \text{and}$$

$$\xi(w) = \begin{pmatrix} D_1 w_1 & \frac{1}{2}(D_2 w_1 + D_1 w_2) \\ \frac{1}{2}(D_2 w_1 + D_1 w_2) & D_2 w_2 \end{pmatrix}.$$

3 Shearlet Transform

The shearlet transformis a very effective tool for tackling the piecewise smooth images containing corners, edges and spikes etc. The shearlets transform can completely approximate the piecewise smooth images' singular structures. Such property of shearlets is suitable particularly in image processing task since irregular structures and singularities carry important details in an observed image. We can propose our high order deblurring model for impulsive noise which is denoted by

$$\min_u \mu \|Au - b\|_1 + \lambda \sum_{j=1}^{N} \|\mathcal{SH}_j(u)\|_1 + TGV_\alpha^2(u), \quad (7)$$

where $\mathcal{SH}_j(u)$ is the j th subband of the shearlet transform of u. For numerical computation, we adopt the fast finite shearlet transform (FFST), [11], in which the construction is based on the Meyer scaling and wavelet functions. Moreover, all the band wise discrete shearlet transforms can be computed fastly using the fast Fourier transform(FFT) and the discrete inverse Fourier transform(IFT). For notational simplicity, we use $\mathcal{SH}_j(MAT(u))$ to interchangeably represent the continuous and the discrete shearlet transform of continuous and discrete u, respectively. Let H_1 be the FFT of the discrete 2D scaling function, and $\mathcal{H}_j (j \geq 2)$ be those of the discrete shearlets. Let $VEC : \mathcal{C}^{k\times k} \to \mathcal{C}^{k^2}$ and $MAT : \mathcal{C}^{k\times k} \to \mathcal{C}^{k^2}$ be the vectorizing and the matricizing operators. Then we have

$$\begin{aligned} \mathcal{SH}_j(MAT(u)) &= F^{-1}(\mathcal{H}_j . * F(MAT(u))) \\ &= F^{-1}(\mathcal{H}_j) * MAT(u), \end{aligned}$$

where $*$ and $.*$ denote convolution and componentwise multiplication. The above equation in vector form is given by

$$\begin{aligned} \mathcal{SH}_j(u) &= VEC(\mathcal{SH}_j(u)) \\ &= F^* \mathrm{diag}(VEC(\mathcal{H}_j)) Fu = M_{\mathcal{H}_j} u, \end{aligned}$$

where $M_{\mathcal{H}_j} = F^* \mathrm{diag}(VEC(\mathcal{H}_j))F$, and diag is defined as

$$\mathrm{diag} : \mathcal{C}^N \to \mathcal{C}^{N\times N} \quad \text{and} \quad \mathrm{diag}(u)_{ij} = u_i \delta_{ij},$$

where $\delta_{ij} = 0$ if $i \neq j$ and $\delta_{ii} = 1$.

We begin with a short review of ADMM, which solves the model in the form of

$$\min_{s,t} \{f(s) + g(t)\} \quad \text{subject to} \quad As + Bt = d. \quad (8)$$

The Lagrangian is $\mathcal{L}(s, t, r) = f(s) + g(t) + \frac{\beta}{2}\|As + Bt - d - r\|_2^2$, where r is the scaled Lagrange multiplier and β is a positive parameter. The ADMM algorithm, [12], [13], starts from $t^0 = 0$ and $r^0 = 0$ and iterates

$$\begin{cases} s^{k+1} = \arg\min_s \mathcal{L}(s, t^k, r^k) \\ t^{k+1} = \arg\min_t \mathcal{L}(s^{k+1}, t, r^k) \\ r^{k+1} = r^k + \beta(d - (As^{k+1} + Bt^{k+1})). \end{cases} \quad (9)$$

We introduce one auxiliary variable and one quadratic penalty term for each l_1 term. More specifically, we introduce auxiliary variables $x_j (j = 1, \ldots, N)$,

$$y = \begin{bmatrix} y_1 \\ y_2 \end{bmatrix} \in \mathcal{V} \quad \text{and} \quad z = \begin{bmatrix} z_1 & z_3 \\ z_3 & z_2 \end{bmatrix} \in \mathcal{W},$$

such that (7) is equivalent to

$$\begin{aligned} \min_{u,w,x_i,y,z} \mu\|p\|_1 + \lambda \sum_{j=1}^{N} \|x_i\|_1 + \alpha_1\|y\|_1 + \alpha_0\|z\|_1 \\ \text{subject to} \quad p = Au - b,\ x_j = \mathcal{SH}_j(u), \\ y = Du - w,\ z = \xi(w). \end{aligned} \quad (10)$$

After applying the ADMM, we arrive at the following algorithm:

$$\begin{cases} p^{k+1} = \arg\min_{p} \|p\|_1 + \frac{\beta_0}{2}\|p - (Au^k - b) - \tilde{p}^k\|_2^2, \\ x_j^{k+1} = \arg\min_{x_j} \|x_j\|_1 + \frac{\beta_1}{2}\|x_j - \mathcal{SH}_j(u^k) - \tilde{x}_j^k\|_2^2, \\ y^{k+1} = \arg\min_{y} \|y\|_1 + \frac{\beta_2}{2}\|y - (Du^k - w^k) - \tilde{y}^k\|_2^2, \\ z^{k+1} = \arg\min_{z} \|z\|_1 + \frac{\beta_3}{2}\|z - \xi(w^k) - \tilde{z}^k\|_2^2, \\ \quad (u^{k+1}, w^{k+1}) \\ \quad = \arg\min_{u} \frac{\mu\beta_0}{2}\|p^{k+1} - (Au - b) - \tilde{p}^k\|_2^2 \\ \qquad + \frac{\lambda\beta_1}{2}\|x_j^{k+1} - \mathcal{SH}_j(u) - \tilde{x}_j^k\|_2^2 \\ \qquad + \frac{\alpha_1\beta_2}{2}\|y^{k+1} - (Du - w) - \tilde{y}^k\|_2^2 \\ \qquad + \frac{\alpha_0\beta_3}{2}\|z^{k+1} - \xi(w) - \tilde{z}^k\|_2^2, \\ \tilde{p}^{k+1} = \tilde{p}^k + \beta((Au^k - b) - p^{k+1}), \\ \tilde{x}_j^{k+1} = \tilde{x}_j^k + \beta(\mathcal{SH}_j(u^k) - x_j^{k+1}), \\ \tilde{y}^{k+1} = \tilde{y}^k + \beta(Du^k - w^k - y^{k+1}), \\ \tilde{z}^{k+1} = \tilde{z}^k + \beta(\xi(w^k) - z^{k+1}). \end{cases} \tag{11}$$

The first four sub-problems are solved by shrinkage explicitly. The p-subproblem and x-subproblem can be solved by

$$\begin{aligned} p^{k+1} &= \text{shrink}_1(Au^k - b + \tilde{p}^k, \frac{1}{\beta_0}), \\ x_j^{k+1} &= \text{shrink}_1(\mathcal{SH}_j(u^k) + \tilde{x}^k, \frac{1}{\beta_1}), \quad j = 1, \dots, N, \end{aligned} \tag{12}$$

where $\text{shrink}_1(v, \lambda) = \text{sgn}(v).*\max(|v| - \lambda, 0)$.

Since the y-subproblem is componentwise separable, the solution to the y-subproblem reads as

$$y^{k+1}(l) = \text{shrink}_2(Du^k(l) - w^k(l) + \tilde{y}^k, \frac{1}{\beta_2}), \quad l \in \Omega, \tag{13}$$

where the component of $y^{(k+1)}(l)$ located at $l \in \Omega$ is denoted by $y^{(k+1)}(l) \in \mathbb{R}^2$, and the shrinkage operator shrink_2 can be formulated as follows

$$\text{shrink}_2(a, \rho) = \begin{cases} 0, & \text{if} \quad a = \mathbf{0}, \\ (\|a\|_2 - \rho)\frac{a}{\|a\|_2}, & \text{if} \quad a \neq \mathbf{0}. \end{cases} \tag{14}$$

Similarly, solution for the z-subproblem is formulated as

$$z^{k+1}(l) = \text{shrink}_F(\xi(w^k)(l) + \tilde{z}^k, \frac{1}{\beta_3}), \quad l \in \Omega, \tag{15}$$

where $z^{k+1}(l) \in \mathcal{S}^{2\times 2}$ is the component of z^{k+1} corresponding to the pixel $l \in \Omega$ and

$$\text{shrink}_F(f, \rho) = \begin{cases} 0, & \text{if} \quad f = \mathbf{0}, \\ (\|f\|_F - \rho)\frac{a}{\|f\|_F}, & \text{if} \quad f \neq \mathbf{0}. \end{cases} \tag{16}$$

where $\mathbf{0}$ is a square null matrix and the Frobenius norm of a matrix is denoted by $\|\cdot\|_F$.

To solve the (u, w)-subproblem, we obtain the first-order necessary conditions for optimality as follows:

$$\begin{cases} \mu\beta_0 A^*(Au - b - p^{k+1} + \tilde{p}^k) \\ \quad + \lambda\beta_1 \sum_{j=1}^{N} M_{\mathcal{H}_j}^*(M_{\mathcal{H}_j}u - x_j^{k+1} + \tilde{x}_j^k) \\ \quad + \alpha_1\beta_2 \sum_{j=1}^{2} D_j^T(D_j u - w_j - y_j^{k+1} + \tilde{y}_j^k) = 0, \\ \alpha_1\beta_2(w_1 - D_1 u + y_1^{k+1} - \tilde{y}_1^k) \\ \quad + \alpha_0\beta_3\Big(D_1^T(D_1 w_1 - z_1^{k+1} + \tilde{z}_1^k) \\ \quad + \frac{1}{2}D_2^T(D_2 w_2 + D_1 w_2 - 2z_3^{k+1} + 2\tilde{z}_3^k\Big) = 0, \\ \alpha_1\beta_2(w_2 - D_2 u + y_2^{k+1} - \tilde{y}_2^k) \\ \quad + \alpha_0\beta_3\Big(D_2^T(D_2 w_2 - z_2^{k+1} + \tilde{z}_2^k) \\ \quad + \frac{1}{2}D_1^T(D_1 w_2 + D_2 w_1 - 2z_3^{k+1} + 2\tilde{z}_3^k\Big) = 0. \end{cases} \tag{17}$$

After grouping the like terms in (17), we obtain the following linear system

$$\begin{bmatrix} b_1 & b_4^T & b_5^T \\ b_4 & b_2 & b_6^T \\ b_5 & b_6 & b_3 \end{bmatrix} \begin{bmatrix} u \\ w_1 \\ w_2 \end{bmatrix} = \begin{bmatrix} K_1 \\ K_2 \\ K_3 \end{bmatrix},$$

where the block matrices are defined as

$$\begin{cases} b_1 = \mu\beta_0 A^*A + \mu\beta_1 \sum_{j=1}^{N} M^*_{\mathcal{H}_j} M_{\mathcal{H}_j} \\ \qquad + \alpha_1\beta_2 \sum_{j=1}^{2} D_j^T D_j, \\ b_2 = \alpha_1\beta_2 I + \alpha_0\beta_3 D_1^T D_1 + \frac{1}{2} D_2^T D_2, \\ b_3 = \alpha_1\beta_2 I + \alpha_0\beta_3 D_2^T D_2 + \frac{1}{2} D_1^T D_1, \\ b_4 = -\alpha_1\beta_2 D_1, \\ b_5 = -\alpha_1\beta_2 D_2, \\ b_6 = \frac{1}{2} D_1^T D_2, \end{cases} \tag{18}$$

and

$$\begin{cases} K_1 = \mu\beta_0 A^*(b + p^{k+1} - \tilde{p}^k) \\ \qquad + \mu\beta_1 \sum_{j=1}^{N} M^*_{\mathcal{H}_j}(x_j^{k+1} - \tilde{x}_j^k) \\ \qquad + \alpha_1\beta_2 \sum_{j=1}^{2} D_j^T(y_j^{k+1} - \tilde{y}_j^k), \\ K_2 = \alpha_1\beta_2(\tilde{y}_1^k - y_1^{k+1}) + \alpha_0\beta_3 D_1^T(z_1^{k+1} - \tilde{z}_1^k) \\ \qquad + \frac{1}{2}\alpha_0\beta_3 D_2^T(2z_3^{k+1} - 2\tilde{z}_3^k), \\ K_3 = \alpha_1\beta_2(\tilde{y}_2^k - y_2^{k+1}) + \alpha_0\beta_3 D_2^T(z_2^{k+1} - \tilde{z}_2^k) \\ \qquad + \frac{1}{2}\alpha_0\beta_3 D_1^T(2z_3^{k+1} - 2\tilde{z}_3^k). \end{cases} \tag{19}$$

Next we multiply a preconditioner matrix from the left to the linear system such that the coefficient matrix is blockwise diagonal

$$\begin{bmatrix} F & 0 & 0 \\ 0 & F & 0 \\ 0 & 0 & F \end{bmatrix} \begin{bmatrix} b_1 & b_4^T & b_5^T \\ b_4 & b_2 & b_6^T \\ b_5 & b_6 & b_3 \end{bmatrix} \begin{bmatrix} F & 0 & 0 \\ 0 & F & 0 \\ 0 & 0 & F \end{bmatrix}^* \begin{bmatrix} Fu \\ Fw_1 \\ Fw_2 \end{bmatrix} = \begin{bmatrix} F & 0 & 0 \\ 0 & F & 0 \\ 0 & 0 & F \end{bmatrix} \begin{bmatrix} K_1 \\ K_2 \\ K_3 \end{bmatrix}.$$

This operation can also be equivalently performed by multiplying each equation in (18) from the left with F. By denoting) $\widetilde{b}_j = \text{diag}(Fb_jF^*)$ and $\widetilde{b_j^T} = \text{diag}(Fb_j^TF^*) = \text{conj}(\text{diag}(Fb_jF^*))$, we have

$$\begin{cases} \widetilde{b_1}.*(Fu) + \widetilde{b_4^T}.*(Fw_1) + \widetilde{b_5^T}.*(Fw_2) = FK_1, \\ \widetilde{b_4}.*(Fu) + \widetilde{b_2^T}.*(Fw_1) + \widetilde{b_6^T}.*(Fw_2) = FK_2, \\ \widetilde{b_5}.*(Fu) + \widetilde{b_6^T}.*(Fw_1) + \widetilde{b_3^T}.*(Fw_2) = FK_3. \end{cases} \tag{20}$$

Similarly to the scalar case, Fu, Fw_1, and Fw_2 can be obtained by applying Cramer's rule. Hence u, w_1, and w_2 have the following closed forms

$$\begin{cases} u = F^* \left(\begin{vmatrix} FK_1 & \widetilde{b_4^T} & \widetilde{b_4^T} \\ FK_2 & \widetilde{b_2^T} & \widetilde{b_6^T} \\ FK_3 & \widetilde{b_6^T} & \widetilde{b_3^T} \end{vmatrix}_* \cdot \Big/ \begin{vmatrix} \widetilde{b_1} & \widetilde{b_4^T} & \widetilde{b_5^T} \\ \widetilde{b_4} & \widetilde{b_2} & \widetilde{b_6^T} \\ \widetilde{b_5} & \widetilde{b_6} & \widetilde{b_3} \end{vmatrix}_* \right), \\ w_1 = F^* \left(\begin{vmatrix} \widetilde{b_1} & FK_1 & \widetilde{b_4^T} \\ \widetilde{b_4} & FK_2 & \widetilde{b_6^T} \\ \widetilde{b_5} & FK_3 & \widetilde{b_3^T} \end{vmatrix}_* \cdot \Big/ \begin{vmatrix} \widetilde{b_1} & \widetilde{b_4^T} & \widetilde{b_5^T} \\ \widetilde{b_4} & \widetilde{b_2} & \widetilde{b_6^T} \\ \widetilde{b_5} & \widetilde{b_6} & \widetilde{b_3} \end{vmatrix}_* \right), \\ w_2 = F^* \left(\begin{vmatrix} \widetilde{b_1} & \widetilde{b_4^T} & FK_1 \\ \widetilde{b_4} & \widetilde{b_2} & FK_2 \\ \widetilde{b_5} & \widetilde{b_6} & FK_3 \end{vmatrix}_* \cdot \Big/ \begin{vmatrix} \widetilde{b_1} & \widetilde{b_4^T} & \widetilde{b_5^T} \\ \widetilde{b_4} & \widetilde{b_2} & \widetilde{b_6^T} \\ \widetilde{b_5} & \widetilde{b_6} & \widetilde{b_3} \end{vmatrix}_* \right), \end{cases} \tag{21}$$

where the division is componentwise. Here $|\cdot|_*$ is defined to be

$$\begin{vmatrix} b_{11} & b_{12} & b_{13} \\ b_{21} & b_{22} & b_{23} \\ b_{31} & b_{32} & b_{33} \end{vmatrix}_* = b_{11}.*b_{22}.*b_{33} + b_{12}.*b_{23}.*b_{31} + b_{13}.*b_{21}.*b_{32} - b_{13}.*b_{22}.*b_{31} - b_{12}.*b_{21}.*b_{33} - b_{11}.*b_{32}.*b_{23},$$

where $.*$ is componentwise multiplication and $b_{ij} \in \mathbb{R}^n$.

In conclusion, summing up the statements above, this yields the resulting alternating minimization method generalized in Algorithm 1.

Algorithm 1 TGV^2-ADMM.

Input: A - linear transformation representing convolution operation.
b - observed image.
Choose: α_0, α_1, λ, μ, $\beta_j\,(j = 0, 1, 2, 3)$.
Initialize: u^0, p^0, $\tilde{p}^0$, w_j^0, y_j^0, $\tilde{y}_j^0\,(j = 1, 2)$, $z_j\,(j = 1, 2, 3)$,
x_j^0, $\tilde{x}_j^0\,(j = 1, \ldots, N)$.
For $k = 0, 1, 2, \ldots$, do the following computations
p^{k+1} is determined by (12)
x^{k+1} is determined by (12)
y^{k+1} is determined by (13)
y^{k+1} is determined by (15)
u^{k+1}, w_1^{k+1}, w_2^{k+1} are determined by (21)
$\tilde{p}_j^{k+1} = \tilde{p}_j^k + \beta((Au^k - b) - p_j^{k+1})$,
$\tilde{x}_j^{k+1} = \tilde{x}_j^k + \beta(\mathcal{SH}_j(u^k) - x_j^{k+1})$,
$\tilde{y}_j^{k+1} = \tilde{y}_j^k + \beta(D_j u^k - w_j^k - y_j^{k+1})$, $j = 1, 2$.
$\tilde{z}_j^{k+1} = \tilde{z}_j^k + \beta(\xi(w^k)_j - z_j^{k+1})$, $j = 1, 2, 3$.
If the halting criteria are satisfied, it returns u^{k+1} and stops.

4 Convergence analysis

The convergence follows directly from that of the classic ADMM because the problem is convex and the variables p, x, y, z, u, w can be grouped into two blocks $\{p, x, y, z\}$ and $\{u, w\}$. For fixed values of $\{u, w\}$, the updates of p, x, y, z are independent of one another. Because of this, the above iteration is a direct application of ADMM.

Theorem 4.1. *For* $\beta_0, \beta_1, \beta_2, \beta_3 > 0$ *and* $\beta \in (0, \frac{1+\sqrt{5}}{2})$, *then ADMM* (11) *converges.*

Proof. By letting $s = (p, x, y, z)$, $t = (u, w), r = (\sqrt{\frac{\mu\beta_0}{\beta}}\tilde{p}, \sqrt{\frac{\lambda\beta_1}{\beta}}\tilde{x}_j, \sqrt{\frac{\alpha_1\beta_2}{\beta}}\tilde{y}, \sqrt{\frac{\alpha_0\beta_3}{\beta}}\tilde{z})$ in (8), and the Lagrangian function be of the form

$$\begin{aligned}
&\mathcal{L}(p, x, y, z, u, w, \tilde{p}, \tilde{x}, \tilde{y}, \tilde{z}) \\
&= \mu\|p\|_1 + \lambda\sum_{j=1}^{N}\|x_j\|_1 + \alpha_1\|y\|_1 + \alpha_0\|z\|_1 \\
&\quad + \frac{\beta}{2}\left[\left\|\sqrt{\frac{\lambda\beta_0}{\beta}}(p - (Au - b)) - \sqrt{\frac{\lambda\beta_0}{\beta}}\tilde{p}\right\|_2^2\right. \\
&\quad + \left\|\sqrt{\frac{\lambda\beta_1}{\beta}}(x_j - \mathcal{SH}_j(u)) - \sqrt{\frac{\lambda\beta_1}{\beta}}\tilde{x}_j\right\|_2^2 \\
&\quad + \left\|\sqrt{\frac{\alpha_1\beta_2}{\beta}}(y - (Du - w)) - \sqrt{\frac{\alpha_1\beta_2}{\beta}}\tilde{y}\right\|_2^2 \\
&\quad \left. + \left\|\sqrt{\frac{\alpha_0\beta_3}{\beta}}(z - \xi(w)) - \sqrt{\frac{\alpha_0\beta_3}{\beta}}\tilde{z}\right\|_2^2\right]
\end{aligned}$$

the convergence analysis of the classical results from current ADMM, [10], [12], [13], [14], yields the following result. □

5 Numerical experiments

In this section, we show some numerical examples using the Algorithm 1 in image restoration compared with TV-ADMM and ADMM.

Our hybrid model is implemented via the alternating minimization method with the equivalent parameters $\alpha_0 = 0.01$, $\alpha_1 = 0.07$, $\lambda = 0.07$ and $\mu = 10^3$. Moreover, the regularization coefficients are firmly chosen as $\beta_0 = 1$, $\beta_1 = 0.5$, $\beta_2 = 0.1$ and $\beta_3 = 10$. The Peak signal to noise ratio (PSNR) in decibel (dB) as follows:

$$\mathrm{PSNR} = 20\log\frac{\|u\|_2}{\|u - u^k\|_2},$$

where u is an original image and u^k is an estimated image at iteration k, respectively.

The stopping criteria of the algorithm is $\frac{\|u^{k+1} - u^k\|_2}{\|u^k\|_2} < 10^{-5}$. All codes were written in Matlab 2017b and run on Dell i-5 Core laptop. Experimental results are shown in Figure 1, Figure 2, Figure 3 and Figure 4 in the Appendix.

6 Conclusions

We propose an algorithm to perform second order total generalized variation using shearlet regularization. We compared its results with those of TV-ADMM and ADMM on four images. On the contrary, the TGV^2-ADMM had a better than two methods .

Acknowledgments

This work was supported by (i) Rambhai Barni Rajabhat University (RBRU), (ii) Thailand Science Research and Innovation (TSRI), and (iii) National Science, Research and Innovation Fund (NSRF) 4369065 (Grant number 1121/2566).

References:

[1] A. Tikhonov and V. Arsenin, *Solution of ill-Posed Problems,* Winston, Washington, DC, 1977.

[2] L. Rudin, S. Osher and E. Fatemi, Nonlinear total variation based noise removal algorithms, *Physica D: Nonlinear Phenomena,* Vol. 60, No.1-4, 1992, pp. 259-268.

[3] J. E. Aujol, G. Gilboa, T. Chan and S. Osher, Structure-texture image decomposition-modeling, algorithms, and parameter selection, *Int. J. Comput. Vision,* Vol.67, 2006, pp. 111-136.

[4] F. Knoll, K. Bredies, T. Pock and R. Stollberger , Second order total generalized variation (TGV) for MRI, *Magn. Reson. Med.,* Vol.65, No.2, 2011, pp. 480-491 .

[5] W. Guo, J. Qin and W. Yin, A new detail-preserving regularization scheme, *SIAM J. Imaging Sci.,* Vol.7, No.2, 2014, pp. 1309-1334 .

[6] T. Jia, Y. Shi, Y. Zhu and L. Wang, An image restoration model combining mixed L^1/L^2 fidelity terms, *J. Vis. Commun. Image R.,* Vol.38, 2016, pp. 461-473.

[7] X. Liu, Augmented Lagrangian method for total generalized variation based Poissonian image restoration, *Comput. Math. Appl.,* Vol.71, No.8, 2016, pp. 1694-1705.

[8] W. Wang and Q. Song, Color image restoration based on saturation-value total variation plus L1 fidelity, *Inverse Problems*, Vol.38, No.8, 2022, pp. 085009.

[9] D. V. Thong, S. Reich, P. Cholamjiak and L. D. Long, Iterative methods for solving monotone variational inclusions without prior knowledge of the Lipschitz constant of the single-valued operator, *Numerical Algorithms,* 2024.

[10] K. Bredies , K. Kunisch and T. Pock, Total generalized variation, *SIAM J. Imaging Sci.,* Vol.3, No.3, 2010, pp. 492-526.

[11] S. H*ä*user, Fast Finite Shearlet Transform, preprint, arXiv:1202.1773, 2012.

[12] D. Gabay and B. Mercier, A dual algorithm for the solution of nonlinear variational problems via finite element approximations, *Comput. Math. Appl.,* Vol.2, No.1, 1976, pp. 17-40.

[13] R. Glowinski and A. Marroco, Sur l'approximation, par éléments finis et la résolution par pénalisation-dualité d'une classe de problémes de Dirichlet non linéaires, *Revue française d'automatique, informatique, recherche opérationnelle. Analyse numérique,* Vol. 9, No. R2, 1975, pp. 41-76.

[14] C. Wu and X. C. Tai, Augmented Lagrangian method, dual methods, and split Bregman iteration for ROF, vectorial TV, and high order models, *SIAM J. Imaging Sci.,* Vol.3, No.3, 2010, pp. 300-339.

Authors' contributions

These authors contributed equally to this work. [Anantachai Padcharoen (anantachai.p@rbru.ac.th)& Duangkamon Kitkuan (duangkamon.k@rbru.ac.th)]

Conflicts of Interest

The authors have no conflicts of interest to declare that are relevant to the content of this article.

Creative Commons Attribution License 4.0 (Attribution 4.0 International , CC BY 4.0)

APPENDIX

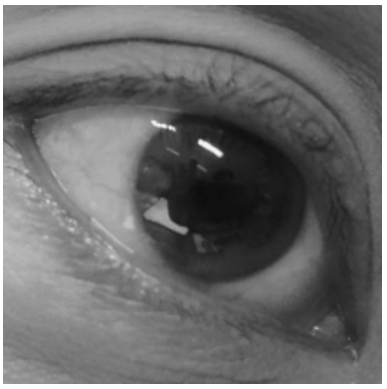
(a) Original image

(b) ADMM (PSNR=25.6174)

(c) TV-ADMM (PSNR=25.8667)

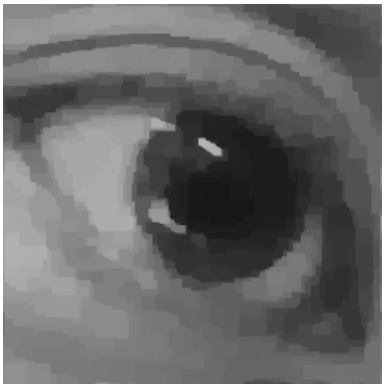
(d) Algorithm 1 (PSNR=26.3065)

Figure 1: Figure (a) shows the Original image, figure (b) shows the restored image by ADMM, figure (c) shows the restored image by TV-ADMM, figure (d) shows the restored image by Algorithm 1.



(a) Original image

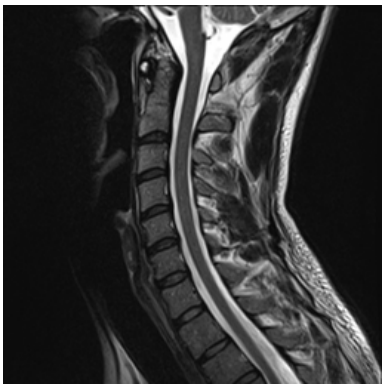
(b) ADMM (PSNR=26.4018)



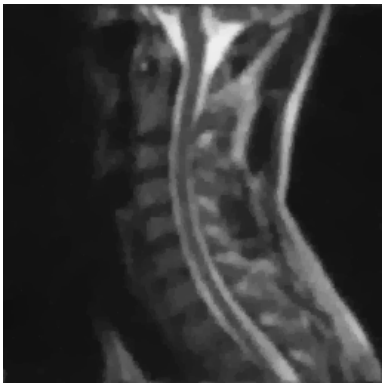
(c) TV-ADMM (PSNR=26.6161)

(d) Algorithm 1 (PSNR=29.4125)

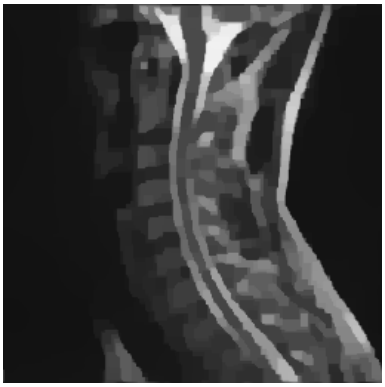
Figure 2: Figure (a) shows the Original image, figure (b) shows the restored image by ADMM, figure (c) shows the restored image by TV-ADMM, figure (d) shows the restored image by Algorithm 1.



(a) Original image



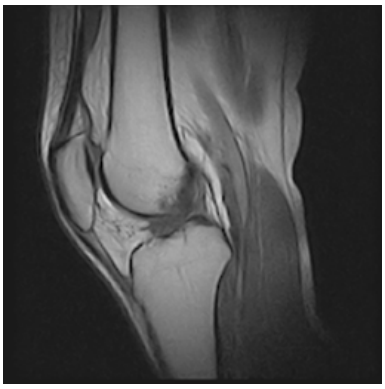
(b) ADMM (PSNR=21.2982)



(c) TV-ADMM (PSNR=22.1277)

(d) Algorithm 1 (PSNR=23.4826)

Figure 3: Figure (a) shows the original image, figure (b) shows the restored image by ADMM, figure (c) shows the restored image by TV-ADMM, figure (d) shows the restored image by Algorithm 1.



(a) Original image

(b) ADMM (PSNR=23.0395)

(c) TV-ADMM (PSNR=23.7816)

(d) Algorithm 1 (PSNR=25.6356)

Figure 4: Figure (a) shows the original image, figure (c) shows the restored image by TV-ADMM, figure (d) shows the restored image by Algorithm 1.