



OPEN

# Edge-guided second-order total generalized variation for Gaussian noise removal from depth map

Shuaihao Li<sup>1,2</sup>✉, Bin Zhang<sup>4</sup>, Xinfeng Yang<sup>3</sup> & Weiping Zhu<sup>3</sup>

Total generalized variation models have recently demonstrated high-quality denoising capacity for single image. In this paper, we present an accurate denoising method for depth map. Our method uses a weighted second-order total generalized variational model for Gaussian noise removal. By fusing an edge indicator function into the regularization term of the second-order total generalized variational model to guide the diffusion of gradients, our method aims to use the first or second derivative to enhance the intensity of the diffusion tensor. We use the first-order primal–dual algorithm to minimize the proposed energy function and achieve high-quality denoising and edge preserving result for depth maps with high -intensity noise. Extensive quantitative and qualitative evaluations in comparison to bench-mark datasets show that the proposed method provides significant higher accuracy and visual improvements than many state-of-the-art denoising algorithms.

Since the advent of ToF sensor, the denoising research on the depth map acquired by it has never stopped. As a major noise type in depth map, Gaussian noise has been widely concerned by researchers. Consequently effective denoising methods have appeared, such as, filtering-based methods<sup>1–3</sup>, partial differential equation (PDE)-based methods<sup>4–6</sup>, sparse representation-based dictionary learning methods<sup>7–11</sup>, deep learning-based methods<sup>12–20</sup>, and recent variation minimization-based methods<sup>21–24</sup>.

The filtering methods mainly include spatial domain filtering and frequency domain filtering. No matter which filter, there are some common phenomena such as incomplete denoising, blur edge, and inconsistency between the smooth area of image denoised and the constant area of ground truth.

The PDE-based methods can diffuse the smooth area of image quickly and the edge area slowly by setting diffusion factors, and finally achieve the effect of denoising and edge preservation. However, for images polluted by high-density noise, the denoising effect of the PDE-based methods is poorly achieved as the termination of the iterative process is hard to control and the convergence speed is slow causing the poor real-time performance.

For sparse representation-based methods, their performance mainly depends on image sparse domain. But, in practice, it is usually not easy to choose the sparse domain due to the variety of images.

The deep learning-based methods have recently demonstrated high-quality denoising for various images. However, such methods also have some drawbacks, such as, the optimization of network structure requires a lot of training, and the datasets required for training are often very large. In addition, specific models need to be trained for specific noise intensity, so their application universality is still poor.

The variation minimization-based methods denoise by iterating the variational model composed of regularization term and data item. The most representative one of this type of method is the total generalized variation (TGV) model proposed by Bredies et al.<sup>24</sup>. The TGV model can not only remove noise effectively, but also preserve the detailed feature of image edge well. However, when the image edge is polluted by noise heavily, the TGV model is prone to misjudge the noises as the edge and therefore unable to extract the noises from the edge.

The goal of our work is to denoise depth maps polluted by high-intensity Gaussian noise and effectively remove the noises to preserve the edges. In this work, we have presented an edge-guided second-order total generalized variation for Gaussian noise removal from depth map (ESTGV). In the ESTGV model, the edge indicating function Canny algorithm is added into the regularization term of the second-order generalized total variational, which can guide the diffusion of the gradients, and then adaptively select the first or second

<sup>1</sup>Research Center for International Business and Economy, Sichuan International Studies University, Chongqing 400031, China. <sup>2</sup>International Business School, Sichuan International Studies University, Chongqing 400031, China. <sup>3</sup>School of Computer Science, Wuhan University, Wuhan 430072, China. <sup>4</sup>Department of Computer Science, City University of Hong Kong, Hong Kong 999077, China. ✉email: lishuaihao@whu.edu.cn

derivative to update the diffusion tensor strength according to the edge indicator function value in the edge area and the smooth area. Then the minimum approximation of the energy function is solved by the first-order primal–dual algorithm (FOPD)<sup>25</sup>, which can effectively preserve the edge by denoising the depth map with high-density Gaussian noise. The experimental results showed that the ESTGV outperforms state-of-the-art denoising methods not only in PSNR index, but also in visual comparison.

**Related work**

**Total variation (TV) for image denoising.** The total variation model (TV) is a classical image denoising algorithm proposed by Rudin et al.<sup>22</sup>, also known as the ROF model. The minimum energy functional of the TV model can be expressed as:

$$\min_{\Omega} a \int_{\Omega} |\nabla u| dx dy + \frac{\lambda}{2} \int_{\Omega} (u - f)^2 dx dy, \tag{1}$$

where  $u$  denotes the predicted image,  $u \in \Omega \in R$ ,  $f$  denotes the noisy image, and the regularization term  $\int_{\Omega} |\nabla u| dx dy$  denotes the prior constraint of image  $u$ , which is used for image denoising.  $|\nabla u|$  denotes the model of the gradient of  $u$ .  $\frac{\lambda}{2} \int_{\Omega} (u - f)^2 dx dy$  is the data fidelity term, also known as the approximation term, which measures the approximation degree of  $u$  and  $f$  by distance and constrains the solution of total variation minimizing to limit the deviation of solution in a small range, so as to achieve the purpose of protecting the structural information of image and reducing the distortion of image.  $\alpha$  and  $\lambda$  are Lagrange scale constants for adjusting regularization term and data fidelity term. They need a proper balancing between the edge preservation and the noise removal. The larger  $\lambda$  is, the closer  $f$  approaches  $u$ , but if the value of  $\lambda$  is too large, the smoothing intensity of local features will be weakened, and the denoising effect will be decreased; the smaller  $\lambda$  is, the greater the denoising degree will be, but if the value of  $\lambda$  is too small, the image will become blurred.

The Euler–Lagrange equation of energy functional (1) can be expressed as:

$$-\nabla \cdot \left( \frac{\nabla u}{|\nabla u|} \right) + \lambda(u - f) = 0, \tag{2}$$

where the first term  $-\nabla \cdot \left( \frac{\nabla u}{|\nabla u|} \right)$  is the diffusion tensor and  $\frac{1}{|\nabla u|}$  is the diffusion coefficient. In the edge area with large gradient,  $|\nabla u|$  is larger, the diffusion coefficient is small, and the diffusion intensity along the edge direction is smaller, so the edge detail features can be well preserved. In the smooth area with small gradient,  $|\nabla u|$  is smaller, the diffusion coefficient is higher, the denoising intensity is higher.

The TV model determines the edge and the noise by the gradient value, and the diffusion is performed only in the gradient orthogonal direction of the image, so that the edge structure can be effectively preserved. But in the smooth area of the image, the TV model can only approximate the piecewise constant function effectively, so it is easy to misjudge the noise as edge to be preserved, and eventually produce the “staircasing artifacts” of the piecewise area.

**Second-order total generalized variation.** In order to solve the problem of “staircasing artifacts” in denoising of TV model, Bredies et al.<sup>24</sup> proposed a generalized mathematical model of TV algorithm, that is, total generalized variation (TGV). Different from the fact that the TV model can only approximate the piecewise constant function, the convex optimization model generated by the TGV model can approximate the piecewise multinomial function of any order, thus effectively avoiding the “staircasing artifacts” in the process of denoising.

The  $k$ -order TGV model can be defined as:

$$TGV_{\alpha}^k(u) = \sup \left\{ \int_{\Omega} u \operatorname{div}^k v \, dx \mid v \in C_c^k(\Omega, \operatorname{Sym}^k(\mathbb{R}^d)), \|\operatorname{div}^l v\|_{\infty} \leq \alpha_l, l = 0, \dots, k-1 \right\} \tag{3}$$

where  $\Omega$  is an open interval,  $\Omega \in \mathbb{R}^d$ , and  $u$  denotes the original image,  $u \in L_{loc}^1(\Omega)$ ,  $TGV_{\alpha}^k$  denotes the  $k$ -order total generalized bounded variations with the weight coefficient  $\alpha(k \geq 1 \alpha = (\alpha_0, \dots, \alpha_{k-1})0)$ , and  $TGV_{\alpha}^k$  provides a way of automatically balancing between the first derivative and the  $k$ -order derivative.  $\operatorname{div}^k v$  denotes the  $k$ -order symmetric divergence operator.  $v \in C_c^k(\Omega, \operatorname{Sym}^k(\mathbb{R}^d))$  denotes the functions of  $k$ -order symmetric tensor in the  $\Omega$ , where  $C_c^k$  is  $k$ -order continuous function space,  $\operatorname{Sym}^k(\mathbb{R}^d)$  is  $k$ -order symmetric tensor space based on  $\mathbb{R}^d$ , and  $\alpha_l$  is a fixed positive parameter.

When  $k = 1$  and  $\alpha_1 = 1$ , Eq. (3) is the dual form of TV model:

$$TGV_1^1(u) = \sup \left\{ \int_{\Omega} u \operatorname{div} v \, dx \mid \|v\|_{\infty} \leq 1 \right\} = TV(u) \tag{4}$$

According to Legendre–Fenchel convex conjugate transformation<sup>26</sup>, the original form of formula (3) can be obtained:

$$TGV_{\alpha}^k(u) = \inf_{\substack{u_l \in C_c^{k-l}(\Omega, \operatorname{Sym}^l(\mathbb{R}^d)) \\ l=1, \dots, k-1, u_0=u, u_k=0}} \sum_{l=1}^k \alpha_{k-l} \|\varepsilon(u_l - 1) - u_l\|_1, \tag{5}$$

where the gradient operator  $\varepsilon(u_l - 1)$  denotes symmetric partial derivative, which can be expressed as:

$$\varepsilon(u_l - 1) = \frac{1}{2} \left( \nabla u_{l-1} + (\nabla u_{l-1})^T \right). \tag{6}$$

In particular, for  $k = 2$ , Eq. (3) is the second-order TGV:

$$TGV_\alpha^2(u) = \min_{\omega \in BD(\Omega)} \alpha_1 \int_\Omega |\nabla u - \omega| dx + \alpha_0 \int_\Omega |\varepsilon(\omega)| dx \tag{7}$$

where  $BD(\Omega)$  is a bounded and distorted vector field<sup>27</sup>, the gradient operator  $\varepsilon(\omega) = \frac{1}{2} (\nabla u_\omega + \nabla u^T)$ .

The corresponding second-order denoising model is expressed as:

$$\min_u TGV_\alpha^2(u) + \frac{\|u - f\|_2^2}{2} \tag{8}$$

The TGV model can effectively avoid the “staircasing artifacts” of piecewise constant areas and is superior to TV model in denoising and edge preservation.

**The first-order primal–dual algorithm (FOPD).** The FOPD algorithm is a particularly effective iterative algorithm proposed by Chambolle et al.<sup>25</sup> in 2011, which is used to solve the convex optimization problems with a special proximal regularization parameter. The FOPD algorithm converts the original minimizing problem into the dual maximization problem by dual variables, and then represents the original problem and the dual problem as their corresponding saddle-point optimization problem, and then iterates the original variables and dual variables alternately until the optimal solution of the original problem is finally approximated. The FOPD algorithm is easy to implement and can effectively iterate alternately, so it can accelerate convergence.

The general convex optimization problem is defined as:

$$\min_{x \in X} F(Ix) + G(x), \tag{9}$$

the corresponding dual problem can be expressed as:

$$\max_{y \in Y} -\{F^*(y) + G^*(-I^*y)\}, \tag{10}$$

where  $x$  is the primal variable,  $y$  is the dual variable.  $X$  and  $Y$  denote the finite-dimensional real vector space of the image with an inner product  $\langle \cdot, \cdot \rangle$  and norm  $\|\cdot\| = \sqrt{\langle \cdot, \cdot \rangle}$ ,  $I : X \rightarrow Y$  is a continuous linear operator,  $F(Ix)$  and  $G(x)$  are semi-continuous convex functions,  $F^*$  and  $G^*$  are conjugate convex functions that are topologically dual to functions  $F$  and  $G$ , respectively.  $I^*$  is the adjoint operator of  $I$ .

The saddle-point formulation of (9) and (10) is given by

$$\min_{x \in X} \max_{y \in Y} [Ix, y] + G(x) - F^*(y). \tag{11}$$

For  $F(Ix) = \|\nabla x\|_1$  and  $G(x) = \frac{\lambda}{2} \|x - g\|_2^2$ , (11) becomes

$$\min_{f \in X} \max_{y \in Y} \left\{ \langle \nabla f, y \rangle_Y + \frac{\lambda}{2} \|f - g\|_2^2 - \delta_P(y) \right\}, \tag{12}$$

where, the function  $\delta_P(y)$  is the dual function of  $F(g) = \|g\|_1$ :

$$F^*(y) = \delta_P(y) = \begin{cases} 0 & \|y\|_\infty \leq 1 \\ \infty & \text{otherwise} \end{cases} \tag{13}$$

The convex set  $P$  is expressed as:

$$P = \{p \in Y : \|p\|_\infty \leq 1\} \tag{14}$$

$\|p\|_\infty$  denotes the discrete maximum norm defined as:

$$\|p\|_\infty = \max |p_{i,j}|, |p_{i,j}| = \sqrt{(p_{i,j}^1)^2 + (p_{i,j}^2)^2} \tag{15}$$

According to the projection approximation algorithm, the iteration formulas of the original variable  $x$  and the dual variable  $y$  can be expressed as:

$$\begin{aligned} y^{n+1} &= (I + \sigma \partial F)^{-1} (y^n + \nabla f^n) \\ \Leftrightarrow y^{n+1} &= \arg \min_{y \in Y} \left\{ \frac{1}{2\sigma} \cdot \|y - (y^n + \sigma \nabla f^n)\|_2^2 + \delta(y) \right\} \\ \Leftrightarrow y_{i,j}^{n+1} &= \left( y_{i,j}^n + \sigma \nabla f_{i,j}^n \right) \cdot \max \left( 1, \left| y_{i,j}^n + \sigma \nabla f_{i,j}^n \right| \right)^{-1}, \end{aligned} \tag{16}$$

and

$$\begin{aligned}
 f^{n+1} &= (I + \tau \partial G)^{-1}(f^n + \tau \operatorname{div} y^{n+1}) \\
 \Leftrightarrow f^{n+1} &= \arg \min_f \left\{ \frac{1}{2\tau} \cdot \|f - (f^n + \tau \operatorname{div} y^{n+1})\|_2^2 + \frac{\lambda}{2} \|f - g\|_2^2 \right\} \\
 \Leftrightarrow f_{i,j}^{n+1} &= \left( (f_{i,j}^n + \tau \operatorname{div} y_{i,j}^{n+1}) + \tau \lambda g_{i,j} \right) \cdot (1 + \tau \lambda)^{-1},
 \end{aligned}
 \tag{17}$$

and

$$\bar{f}^{n+1} = f^{n+1} + \theta (f^{n+1} - f^n) = 2f^{n+1} - f^n.
 \tag{18}$$

### The proposed method

**Edge-guided second-order total generalized variation for Gaussian noise removal.** In order to solve the edge blurring issue during denoising depth maps with high-intensity Gaussian noise, we propose an edge-guided second-order total generalized variation model (ESTGV) for depth map Gaussian noise removal. The ESTGV model aims to preserve the depth edge and detail features profoundly when denoising Gaussian noise heavily polluted depth maps by adding an edge indication function into the second-order total generalized variation model.

For the choice of edge indication function, we have compared some other edge detection algorithms, and found that Roberts detector usually misses some edges, Sobel detector and Prewitt detector often generate false edges, Laplacian detector is over-sensitive to noise and not suitable for noisy images, and LOG (Laplacian of Gaussian) detector usually loses sharp edges, while Canny detector adopts the principle of optimal edge detection, which has strong anti-noise ability, high integrity and continuity of edge detected, and adaptive threshold generation, and its edge preserving performance is obviously better than other detectors.

By adding the Canny edge detection algorithm<sup>28</sup> into the regularization term of the second-order TGV, we propose a weighted second-order TGV minimization denoising model:

$$\min_{\Omega} (u) + \frac{\lambda}{2} \int_{\Omega} (u - f)^2 dx
 \tag{19}$$

where  $\min_{\Omega}(u)$  is the regularization term,  $\frac{\lambda}{2} \int_{\Omega} (u - f)^2 dx$  is the data fidelity term,  $u$  denotes the predicted depth map denoised,  $f$  denotes the noisy depth map,  $\lambda$  is the Lagrange constant factor for balancing the regularization term and the data fidelity term,  $\Omega$  is the definition domain of pixels on the depth map,  $(x, y) \in \Omega$ .

The regularization term  $\min_{\Omega}(u)$  can be expressed as  $\Phi(u)$ , that is, the second order total generalized variation  $TGV_a^2(u)$ :

$$\Phi(u) = TGV_a^2(u) = \min_p a_1 \| \varepsilon(p) \|_1 + a_2 \| T(\nabla u - p) \|_1
 \tag{20}$$

where  $p \in R^{mn} \times R^{mn}$  is a bounded two-dimensional first-order vector field,  $p_{(i,j)} = [p_{i,j,1}, p_{i,j,2}]$ .  $a_1$  and  $a_2$  are two constants used to coordinate the first derivative and second derivative;  $\nabla u$  is the gradient map of  $u$ , which is used to judge the scale of noise and edge. The gradient formulation of pixel  $u(x, y)$  on depth map  $u$  is given by:

$$|\nabla u(x, y)|_n = \sqrt{\left(\frac{\partial u}{\partial x}\right)_n^2 + \left(\frac{\partial u}{\partial y}\right)_n^2}
 \tag{21}$$

where  $\left(\frac{\partial u}{\partial x}\right)_n$  and  $\left(\frac{\partial u}{\partial y}\right)_n$  denote the horizontal and vertical gradients in the  $n$ th iteration process, respectively, which can be calculated by the median difference:

$$\begin{cases} \left(\frac{\partial u}{\partial x}\right)_n = \frac{1}{2}(u(x+1, y)_n - u(x-1, y)_n) \\ \left(\frac{\partial u}{\partial y}\right)_n = \frac{1}{2}(u(x, y+1)_n - u(x, y-1)_n) \end{cases}
 \tag{22}$$

$T$  of (20) denotes the edge indicator function:

$$T = \frac{1}{1 + M|\nabla G_{\sigma} * f|^2}
 \tag{23}$$

where  $M \geq 0$  is the contrast factor,  $G_{\sigma}$  is the Gaussian filtering kernel with a mean value of 0 and a standard deviation of  $\sigma$ .  $*$ denotes convolution operation, and  $G_{\sigma} * f$  denotes the depth map preprocessed by Gaussian filtering kernel  $G_{\sigma} * f$ .

$\varepsilon(p)$  in (20) denotes the second-order symmetric gradient operator:

$$\varepsilon(p) = \frac{\nabla p + (\nabla p)^T}{2} = \begin{bmatrix} \varepsilon(p)_{i,j,1} & \varepsilon(p)_{i,j,3} \\ \varepsilon(p)_{i,j,3} & \varepsilon(p)_{i,j,2} \end{bmatrix}
 \tag{24}$$

The regularization item  $G_\sigma(x)$  is the Canny edge detection algorithm:

$$G_\sigma(x) = \frac{1}{2\pi\sigma^2} \exp\left[-\frac{x^2 + y^2}{2\sigma^2}\right] \tag{25}$$

where  $\sigma$  is the weight of the Gaussian filter to adjust the smoothing level. The smaller the  $\sigma$  value, the higher the positioning accuracy of the filter, but the lower the signal-to-noise ratio (SNR). The larger the  $\sigma$  value, the higher the SNR, but the positioning accuracy of the filter is also reduced.

According to the denoising model (19), the edge detection of the depth map is performed by  $|\nabla G_\sigma * f|^2$ , if the value of  $|\nabla G_\sigma * f|^2$  is large, the position of the observed pixel is the depth edge, and the value of  $T$  decreases because of the larger denominator, so the first derivative diffusion becomes relatively weak, so that the edge structure can be preserved well. If the value of  $|\nabla G_\sigma * f|^2$  is small, the observed pixel is located in the smooth area of the depth map, the value of  $T$  is close to 1 and the diffusion degree of the first derivative increases, so that the noise can be filtered effectively.

In conclusion, the edge indicator function can adaptively adjust the diffusion intensity of different frequency domains in depth images. For the edge areas, the first derivative is used to preserve their detailed features. For the smooth areas, the higher order derivative is used to filter out the noise, so as to effectively preserve the edge while denoising the depth map with high-density gaussian noise.

**Numerical algorithms.** In our denoising model, the regularization term is proper, convex and lower semi-continuous, and the data fidelity term is strictly convex and completely non-smooth. According to the convex analysis theory<sup>29</sup>, (19) is an unconstrained optimization problem, and there is a unique minimum solution, that is, the global optimal solution<sup>30</sup>.

We employ the FOPD method<sup>25</sup> as the numerical algorithm in order to solve the ESTGV model (19). The FOPD algorithm can solve optimization problems with efficient iterations.

According to the Legendre-Fenchel transformation<sup>24</sup>, the dual form of (19) can be obtained from:

$$\min_{u,w} \max_{m \in M, n \in N} \langle \nabla u - p, m \rangle + \langle \varepsilon(p), m \rangle + \frac{\lambda}{2} \| (u - f)^2 \|_2^2 \tag{26}$$

where

$$\begin{aligned} M &= \{m = (m_1, m_2)^T \mid |m(x)| \leq Ta_2\}, \\ N &= \left\{ n = \begin{pmatrix} n_{11}, n_{12} \\ n_{21}, n_{22} \end{pmatrix} \mid \|n\|_\infty \leq a_1 \right\} \end{aligned} \tag{27}$$

Using the original-dual algorithm<sup>25</sup> to solve (26), The iterative process is obtained:

$$\begin{cases} m^{k+1} = \text{proj}_M(m^k + a(\nabla \bar{u}^k - \bar{p}^k)) \\ n^{k+1} = \text{proj}_N(n^k + a(\varepsilon(\bar{p}^k))) \\ u^{k+1} = (u^k + \tau(\text{div } m^{k+1} + \lambda f)) \cdot (1 + \tau \lambda) \\ p^{k+1} = p^k + \tau(\text{div}^h n^{k+1} + m^k) \\ \bar{u}^{k+1} = u^k + \eta(u^{k+1} - \bar{u}^k) \\ \bar{p}^{k+1} = p^{k+1} + \eta(p^{k+1} - \bar{p}^k) \end{cases} \tag{28}$$

where  $k$  denotes the number of iterations, the divergence operator  $\text{div}$  is the dual operator of the gradient operator  $\nabla$ , which denotes the negative conjugate of the gradient operator  $\nabla$ , i.e.  $\text{div} = -\nabla^*$ , the discrete version of  $\text{div}$  is  $\text{div}(m_1, m_2) = \partial_x^- m_1 + \partial_y^- m_2$ ,  $\text{div}^h$  denotes the negative conjugate of the second-order symmetric gradient operator  $\varepsilon$ ,  $\text{div}^h = -\varepsilon^*$ . When

$$m = \begin{pmatrix} m_{11}, m_{12} \\ m_{21}, m_{22} \end{pmatrix}, \text{div}^h(n) = \begin{pmatrix} \partial_x^- m_{11} + \partial_y^- m_{12} \\ \partial_x^- m_{21} + \partial_y^- m_{22} \end{pmatrix}.$$

$\partial_x^-, \partial_y^-$  is the first-order backward differences operator.

The projections of (24) can be easily obtained by applying pointwise operations below:

$$\begin{cases} \text{proj}_M(m) = m \cdot \left(\max\left(1, \frac{|m|}{Ta_2}\right)\right)^{-1} \\ \text{proj}_N(n) = n \cdot \left(\max\left(1, \frac{|n|}{Ta_1}\right)\right)^{-1} \end{cases} \tag{29}$$

In conclusion, the solution of denoising model (19) is an iterative process for solving original variables  $u, p$  and dual variables  $m, n$ . The specific solving process of the primal-dual algorithm is as follows:

Step 1: Initialization:

$$\begin{aligned} u^0 = \bar{u}^0 = f, p^0 = \bar{p}^0 = 0, m^0 = n^0 = 0, \\ \eta_m > 0, \eta_n > 0, \tau_u > 0, \tau_p > 0, \lambda > 0, k \geq 0. \end{aligned}$$

*Step 2:* Using the Iteration formulation (22) to solve  $m^{i,k+1}, n^{j,k+1}, u^{i,k+1}, p^{j,k+1}, \bar{u}^{j,k+1}, \bar{p}^{j,k+1}$ .

*Step 3:* When  $u^{k+1}, p^{k+1}$  appear to converge, the algorithm is terminated; otherwise, set  $k = k + 1$ , and continue to iterate until the stopping criterion is satisfied.

The numerical operation of the ESTGV model (19) needs to be implemented in a discrete version of the gradient operator and the divergence operator.

Assuming that the resolution of depth map is  $s \times s$ , then the discrete versions of the first-order forward and backward difference operators are given by

$$\begin{aligned} (\partial_x^+ u)_{i,j} &= \begin{cases} u_{i+1,j} - u_{i,j}, & 1 \leq i < n \\ 0, & i = n \end{cases} \\ (\partial_y^+ u)_{i,j} &= \begin{cases} u_{i,j+1} - u_{i,j}, & 1 \leq j < n \\ 0, & j = n \end{cases} \\ (\partial_x^- u)_{i,j} &= \begin{cases} u_{i,j} - u_{i-1,j}, & 1 < i < n \\ u_{1,j}, & i = 1 \\ -u_{n-1,j}, & i = n \end{cases} \\ (\partial_y^- u)_{i,j} &= \begin{cases} u_{i,j} - u_{i,j-1}, & 1 < j < n \\ u_{i,1}, & j = 1 \\ -u_{i,n-1}, & j = n \end{cases} \end{aligned} \quad (30)$$

The discrete version of the gradient operator is expressed as  $\nabla = (\partial_x^+, \partial_y^+)^T$ , and the discrete second-order symmetric gradient operator  $\varepsilon$  is expressed as:

$$\varepsilon(p) = \frac{1}{2}(\nabla p + \nabla p^T) = \begin{bmatrix} \partial_x^+ p_1, \frac{1}{2}(\partial_y^+ p_1 + \partial_x^+ p_2) \\ \frac{1}{2}(\partial_x^+ p_2 + \partial_y^+ p_1), \partial_y^+ p_2 \end{bmatrix} \quad (31)$$

## Experimental results

In this section, we evaluate our proposed method both quantitatively and qualitatively with respect to other benchmark and state-of-the-art image denoising methods.

**Experimental platform:** The simulations are carried out on MATLAB R2018a using a laptop with a quad-core 2.2 GHz Intel(R) i7-4770HQ CPU, 16 GB RAM, Intel (R) Iris (TM) Pro Graphics 5200.

**Experimental Dataset:** We experimented with our denoising method on the public dataset, namely the Middlebury stereo dataset<sup>31,32</sup>. The resolution of each depth map is  $256 \times 256$ .

**Baseline Methods:** We compare our results with the following five categories of the methods. (1) *Filtering-based methods:* Bilateral Filter (BF)<sup>4</sup> and Block-Matching and 3D Filtering (BM3D)<sup>2</sup>. (2) *PDE-based methods:* Weighted Nuclear Norm Minimization (WNNM)<sup>6</sup>. (3) *Sparse representation algorithm:* K-Singular Value Decomposition (K-SVD)<sup>7</sup>. (4) *Deep learning-based methods* including Multi-Layer Perceptron (MLP)<sup>12</sup>, Trainable Non-linear Reaction Diffusion (TNRD)<sup>13</sup> and A Cascade of Shrinkage Fields (CSF)<sup>14</sup>. (5) *Variational-based methods:* Total Variation (TV)<sup>22</sup> and Total Generalized Variation (TGV)<sup>24</sup>. The results of the comparison algorithms are either obtained from the original papers or from the source code provided by the authors.

**Parameters Setting:** The parameters of our ESTGV algorithm are set as follows: Lagrange multiplier  $\lambda = 10$ , Gaussian noise of standard deviation  $\sigma = 0.2$ , contrast factor  $M = 5$ , filter kernel size  $7 \times 7$ ,  $a_2 = 2$ ,  $a_1 = 4$ ,  $\tau = 0.04$ . We also set parameters for each comparison method. Note that at least one of these parameters is from the default parameters provided by the authors, while the others are selected by ourselves.

**Quantitative results.** We first evaluate our results on six benchmark maps (Art, Books, Dolls, Laundry, Moebius and Reindeer) from Middlebury Stereo dataset<sup>32</sup>. Note that we added Zero-Mean-Gaussian noise with standard deviation 15, 20, 25 and 50 to all depth maps. We use peak signal-to-noise ratio (PSNR) as the evaluation metric. Tables 1 and 2 show the comparison of our proposed method with respect to various methods.

As shown in Tables 1 and 2, the PSNR results of each denoising algorithm on different depth maps are different due to the internal structure variance of the six datasets, and the proposed ESTGV achieves the highest PSNR values on different depth maps in the vast majority of the cases, which shows significant improvement in comparison to other denoising methods. The second-best denoising algorithm i.e. MLP<sup>12</sup>, despite outperformed the proposed ESTGV slightly on Dolls dataset at  $\sigma = 25$  and Moebius dataset at  $\sigma = 50$ , with only 0.01 dB and 0.02 dB, respectively, it underperformed the proposed ESTGV model on all datasets with two noise levels of  $\sigma = 15$  and  $\sigma = 20$ . Our results verify that the proposed Edge-guided Second-order Total Generalized Variation (ESTGV) can provide more superior performance for depth map denoising. Our results also indicate that MLP<sup>12</sup> is suitable for training a separate denoising model for each individual noise level. However, if the noise level of the test dataset deviates from the trained dataset, its performance would be compromised and it is impractical to train a separate MLP<sup>12</sup> denoising model for each noise level. On average, the proposed ESTGV achieves 0.11 dB, 0.22 dB, 0.29 dB and 0.22 dB performance improvements over MLP<sup>12</sup> at four individual noise levels ( $\sigma = 20$ ,  $\sigma = 25$  and  $\sigma = 50$ ), respectively. Compared with the benchmark BF method<sup>4</sup>, the proposed ESTGV has improved 3.29 dB, 3.17 dB, 3.11 dB and 4.22 dB, respectively. In summary, our results show that the proposed ESTGV can provide robust performance at different noise levels in comparison to its counterparties.

Datasets	Art		Reindeer		Dolls		Book		Laundry		Moebius		Average	
	$\sigma = 15$	$\sigma = 20$	$\sigma = 15$	$\sigma = 20$	$\sigma = 15$	$\sigma = 20$	$\sigma = 15$	$\sigma = 20$	$\sigma = 15$	$\sigma = 20$	$\sigma = 15$	$\sigma = 20$	$\sigma = 15$	$\sigma = 20$
BF <sup>1</sup>	36.97	35.67	35.69	34.39	35.62	34.21	35.85	34.53	36.26	34.97	35.99	34.66	36.06	34.74
TV <sup>22</sup>	37.71	36.31	37.22	35.77	36.99	35.46	37.05	35.52	37.56	36.09	37.12	35.61	37.28	35.80
K-SVD <sup>7</sup>	38.13	36.77	37.66	36.29	37.37	35.97	37.55	36.05	37.97	36.57	37.33	35.93	37.67	36.26
BM3D <sup>2</sup>	38.96	37.35	38.25	36.73	38.09	36.56	38.12	36.57	38.61	36.99	38.04	36.43	38.35	36.77
CSF <sup>14</sup>	39.07	37.66	38.63	37.21	38.31	36.90	38.27	36.83	38.52	37.12	37.77	36.47	38.43	37.03
TGV <sup>24</sup>	39.23	37.73	38.67	37.16	38.61	37.12	38.56	37.05	38.92	37.40	38.33	36.82	38.72	37.21
WNNM <sup>6</sup>	39.35	37.85	<b>39.36</b>	<u>37.81</u>	<u>38.91</u>	37.31	38.82	<u>37.33</u>	39.01	37.50	38.44	36.87	38.98	37.45
TNRD <sup>13</sup>	<b>39.52</b>	<u>37.91</u>	39.26	37.74	38.89	<u>37.44</u>	<u>38.87</u>	37.27	<u>39.09</u>	<u>37.55</u>	<b>38.62</b>	<u>37.05</u>	<u>39.04</u>	<u>37.50</u>
MLP <sup>12</sup>	<u>39.36</u>	<b>37.95</b>	<u>39.27</u>	<b>37.93</b>	<b>38.97</b>	<b>37.51</b>	<b>39.09</b>	<b>37.79</b>	<b>39.11</b>	<b>37.71</b>	<u>38.56</u>	<b>37.26</b>	<b>39.06</b>	<b>37.69</b>
ESTGV	39.57	38.06	39.44	38.02	39.29	37.97	39.33	38.01	39.39	37.81	39.07	37.59	39.35	37.91

**Table 1.** PSNR (dB) evaluation of different algorithms under two noise levels ( $\sigma = 15, \sigma = 20$ ) on the Middlebury dataset. Italics indicates the best, bold indicates the second best, underscore indicates the third best.

Datasets	Art		Reindeer		Dolls		Book		Laundry		Moebius		Average	
	$\sigma = 25$	$\sigma = 50$	$\sigma = 25$	$\sigma = 50$	$\sigma = 25$	$\sigma = 50$	$\sigma = 25$	$\sigma = 50$	$\sigma = 25$	$\sigma = 50$	$\sigma = 25$	$\sigma = 50$	$\sigma = 25$	$\sigma = 50$
BF <sup>1</sup>	34.52	31.52	33.25	30.26	33.07	30.07	33.39	30.39	33.82	30.85	33.52	30.53	33.60	30.60
TV <sup>22</sup>	35.15	33.35	34.61	32.81	34.31	32.50	34.37	32.56	34.95	33.13	34.47	32.65	34.64	32.83
K-SVD <sup>7</sup>	35.61	33.80	35.13	33.32	34.81	33.01	34.89	33.08	35.41	33.62	34.77	32.96	35.10	33.30
BM3D <sup>2</sup>	36.18	34.39	35.56	33.77	35.39	33.62	35.4	33.63	35.82	34.03	35.26	33.44	35.60	33.81
CSF <sup>14</sup>	36.50	34.69	36.05	34.22	35.74	33.95	35.67	33.86	35.96	34.16	35.31	33.52	35.87	34.07
TGV <sup>24</sup>	36.57	34.75	35.97	34.18	35.96	34.16	35.89	34.07	36.26	34.44	35.67	33.85	36.05	34.24
WNNM <sup>6</sup>	36.68	34.87	<b>36.65</b>	<u>34.85</u>	36.13	34.33	<u>36.16</u>	<u>34.35</u>	36.32	34.52	35.69	33.87	36.27	34.47
TNRD <sup>13</sup>	<u>36.71</u>	<u>34.92</u>	36.52	34.75	<u>36.26</u>	<u>34.44</u>	36.05	34.27	<b>36.51</b>	<b>34.71</b>	<u>35.87</u>	<u>34.16</u>	<u>36.30</u>	<u>34.54</u>
MLP <sup>12</sup>	<b>36.75</b>	<b>34.95</b>	<u>36.73</u>	<b>34.92</b>	36.31	<b>34.52</b>	<b>36.59</b>	<b>34.82</b>	<u>36.49</u>	<u>34.71</u>	<b>36.06</b>	34.33	<b>36.49</b>	<b>34.71</b>
ESTGV	36.86	35.07	36.82	35.02	<b>36.30</b>	34.67	36.81	35.03	36.61	34.82	36.39	<b>34.31</b>	36.71	34.82

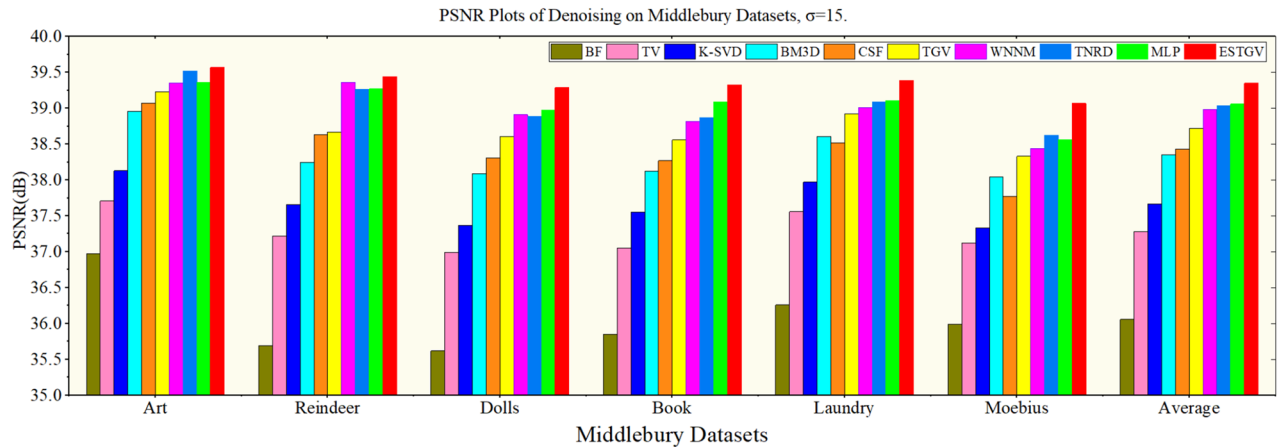
**Table 2.** PSNR (dB) evaluation of different algorithms under two noise levels ( $\sigma = 25, \sigma = 50$ ) on the Middlebury dataset. Italics indicates the best, bold indicates the second best, underscore indicates the third best.

Figures 1 and 2 show intuitively the results of removing four levels of Gaussian noise from the Middlebury dataset<sup>32</sup> by 10 algorithms in the form of histogram and broken line graph, respectively.

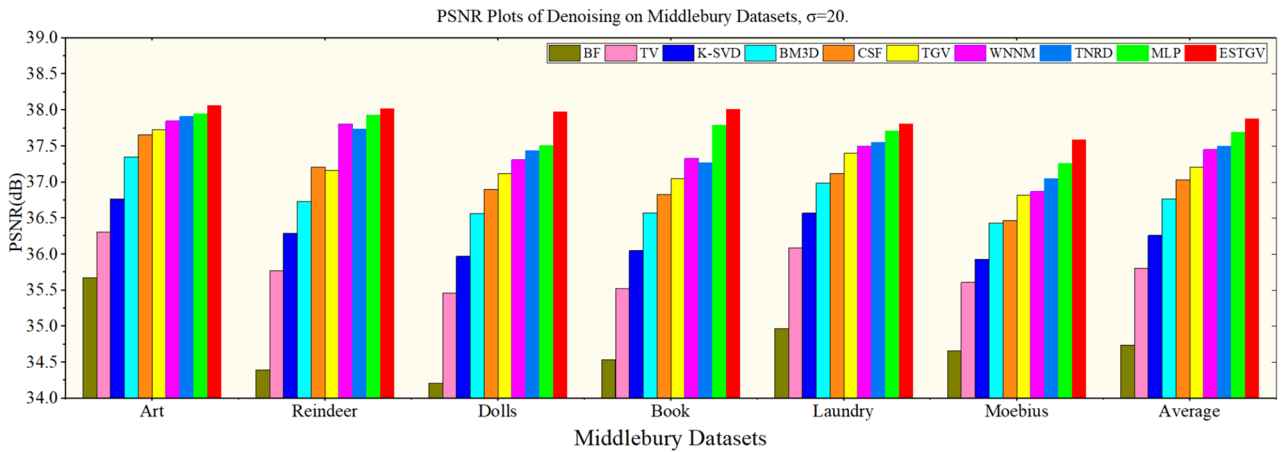
**Qualitative results.** We evaluate the proposed ESTGV method visually in Figs. 3, 4, 5, 6, which show the visual comparison of different denoising methods on *Art*, *Bowling*, *Aloe* and *Teddy* from Middlebury dataset, respectively. We have the following observations. MLP<sup>12</sup> over-smooths the textures more on the edges of depth map. TNRD<sup>13</sup> can reconstruct sharp edges and fine details, but it is easy to generate artifacts in smooth areas. Methods BM3D<sup>2</sup> and CSF also generate blurred boundaries and block artifacts. Although WNNM<sup>6</sup> has a proper balance between noise removal and edge preservation, it can still cause the depth details to be too smooth and produce ringing phenomena. K-SVD<sup>7</sup> can preserve clear edges and rich details, but it also generates structural distortion phenomena such as jags and burrs in the edge area. TV<sup>22</sup> produces obvious “staircasing artifacts”. Compared with TV<sup>22</sup>, TGV<sup>24</sup> can avoid “staircasing artifacts” and appear smoother and more natural in visual effect, but it cannot preserve the edges too well because of its over-smoothness reduces the sharpness of the depth map causing some blocky effect in the smooth area. The result of BF<sup>1</sup> is the blurriest with texture artifacts existing in various areas of the depth map. It is obvious that the proposed ESTGV generates much less artifacts and preserves the edge details much better than other denoising algorithms. In summary, the proposed ESTGV algorithm shows more robustness to noise strength, producing much more pleasant visual outputs.

## Conclusions

Aiming to extend the research in improving generalized variational models’ denoising capacity on depth maps, we have presented an edge-guided second-order total generalized variation model for Gaussian noise removal (ESTGV) in this paper. ESTGV fuses an edge indicator function into the regularization term of the second-order total generalized variational model to guide the diffusion of gradients. It can adaptively use the first or second derivative to update the intensity of the diffusion tensor, and therefore effectively denoise and preserve edges under different intensity noise levels. Our quantitative and qualitative experimental results demonstrated that the proposed ESTGV method shows more robustness on noise strength in comparison to very recent state-of-the-

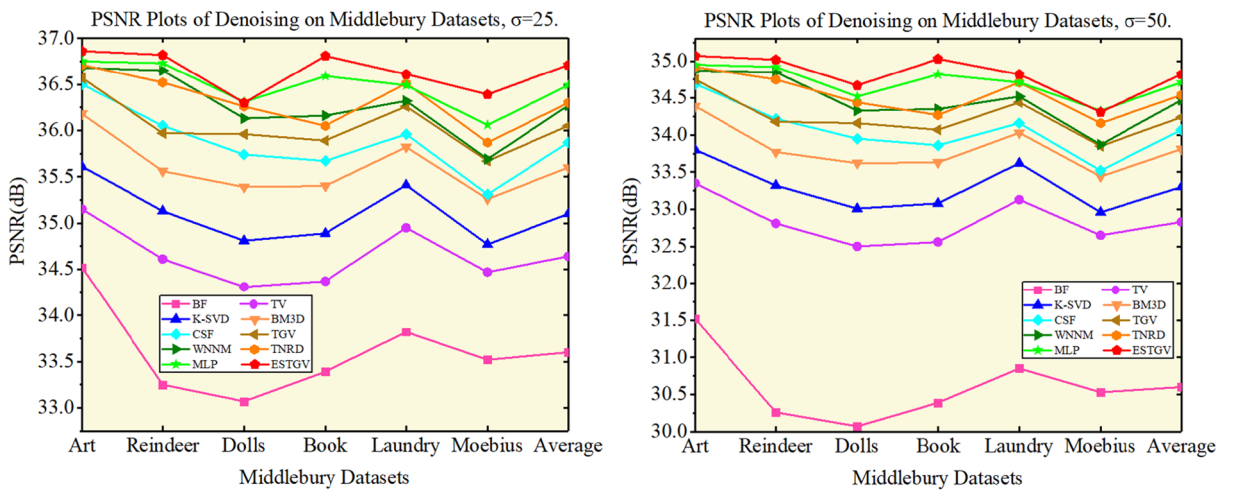


(a)  $\sigma=15$



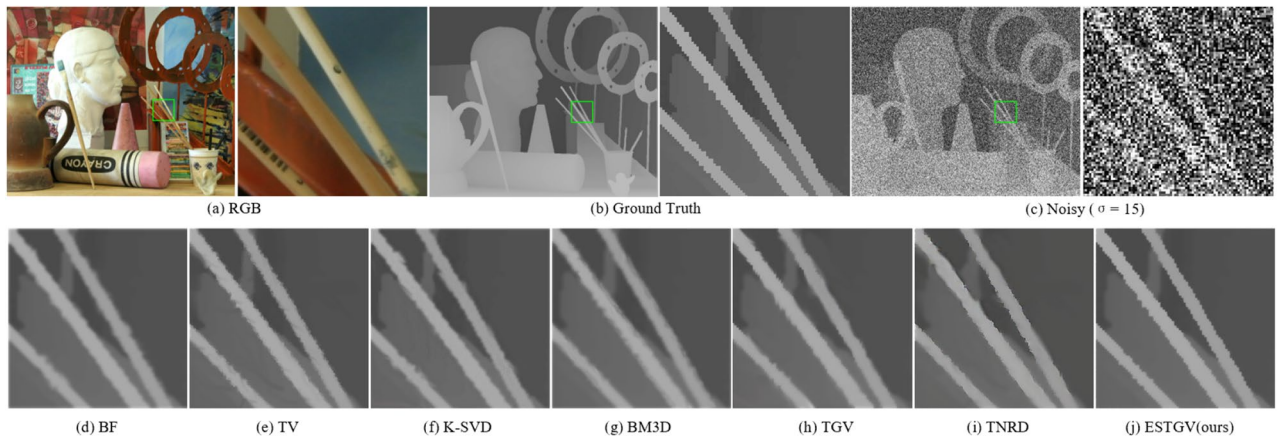
(b)  $\sigma=20$

**Figure 1.** PSNR (dB) results of different denoising methods on Middlebury datasets ( $\sigma = 15, \sigma = 20$ ).

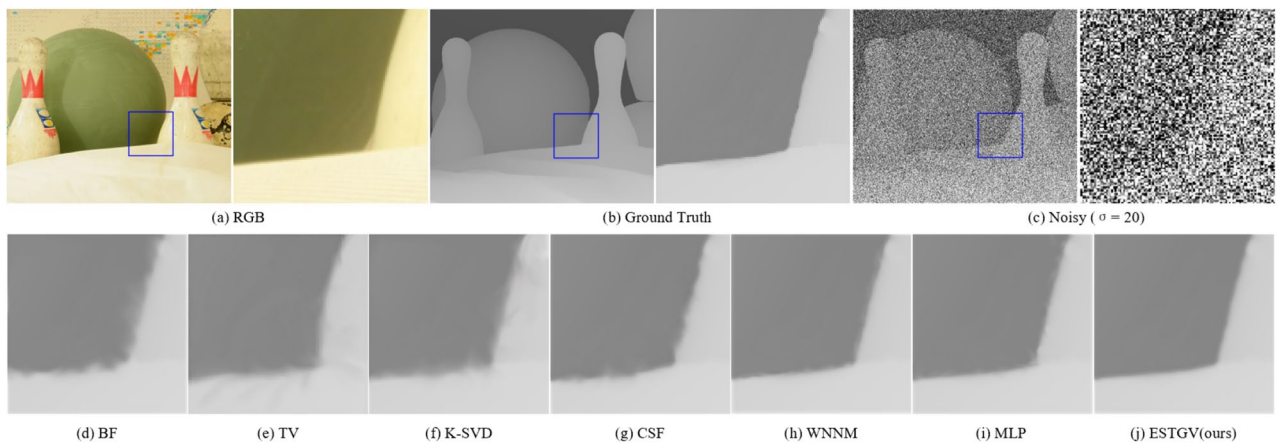


**Figure 2.** PSNR (dB) results of different denoising methods on Middlebury datasets ( $\sigma = 15, \sigma = 20$ ).

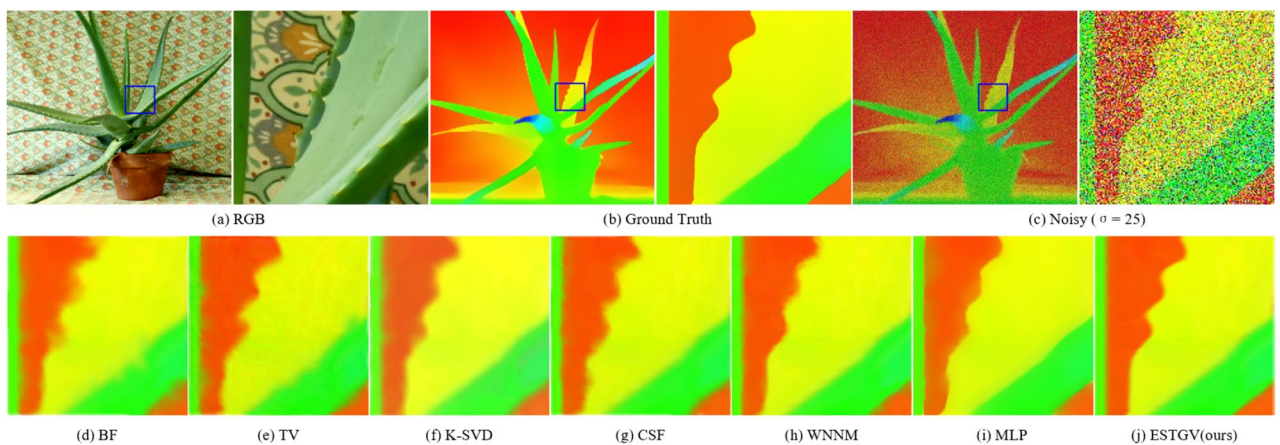




**Figure 3.** Denoising results on depth map *Art* by different methods (noise level  $\sigma = 15$ ).

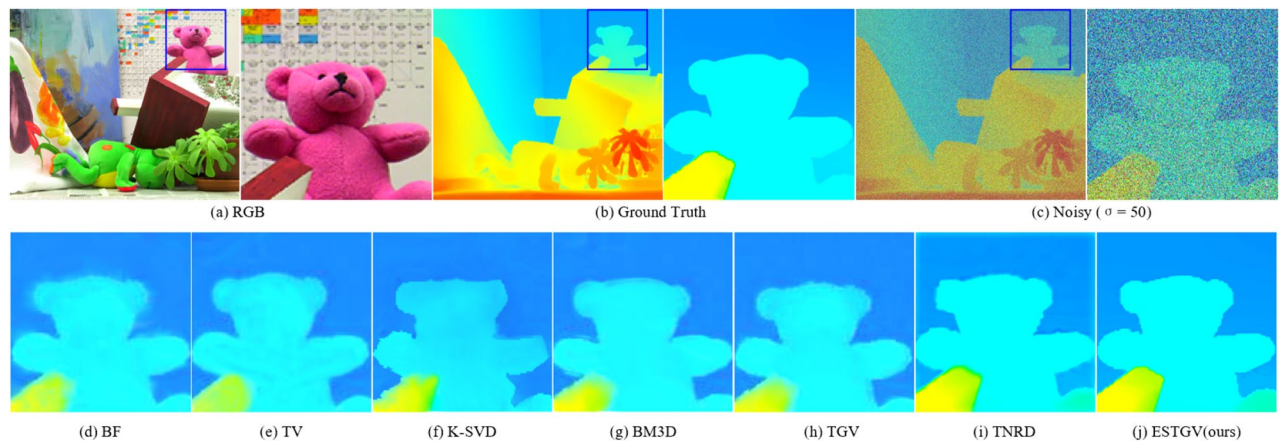


**Figure 4.** Denoising results on depth map *Bowling* by different methods (noise level  $\sigma = 20$ ).



**Figure 5.** Denoising results on depth map *Aloe* by different methods (noise level  $\sigma = 25$ ).

art denoising algorithms. It can not only lead to visible PSNR improvements over state-of-the-art methods such as MLP, WNNM and TNRD, but also preserve the depth structures much better and generate much less texture artifacts.



**Figure 6.** Denoising results on depth map *Teddy* by different methods (noise level  $\sigma = 50$ ).

Received: 25 June 2020; Accepted: 15 September 2020

Published online: 01 October 2020

## References

- Tomasi, C. & Manduchi, R. Bilateral filtering for gray and color images. In *Proceedings of IEEE Conference Computer Vision. (ICCV)*. Bombay, India, Vol. 846, 839–846 (1998).
- Dabov, K., Foi, A., Katkovnik, V. & Egiazarian, K. Image denoising by sparse 3-D transform-domain collaborative filtering. *IEEE Trans. Image Process.* **16**, 2080–2095 (2007).
- Li, S. H., Zhu, W. P., Zhang, B., Yang, X. F. & Chen, M. Depth map denoising using a bilateral filter and a progressive CNN. *J. Opt. Technol.* **87**, 361–364 (2020).
- Koenderink, J. J. The structure of images. *Biol. Cybern.* **50**, 363–370 (1984).
- Witkin, A. P. Scale-space filtering. *Read. Comput. Vis.* **42**, 329–332 (1987).
- Gu, S., Zhang, L., Zuo, W. & Feng X. Weighted nuclear norm minimization with application to image denoising. In *Proceedings of IEEE Conference Computer Vision Pattern Recognition. (CVPR)*, 2862–2869 (Columbus, Ohio, USA, 2014).
- Aharon, M., Elad, M. & Bruckstein, A. K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation. *IEEE Trans. Signal Process.* **54**, 4311–4322 (2006).
- Elad, M. & Aharon, M. Image denoising via sparse and redundant representations over learned dictionaries. *IEEE Trans. Image Process.* **15**, 3736–3745 (2006).
- Mahmoudi, M. & Sapiro, G. Sparse representations for range data restoration. *IEEE Trans. Image Process.* **21**, 2909–2915 (2012).
- Hong, Y. & Zhu, W. P. Learning visual codebooks for image classification using spectral clustering. *Soft Comput.* **22**, 1–10 (2017).
- Hong, Y. & Zhu, W. P. Spatial co-training for semi-supervised image classification. *Pattern Recognit. Lett.* **63**, 59–65 (2015).
- Burger, H. C., Schuler, C. J. & Harmeling, S. Image denoising: Can plain neural networks compete with BM3D? In *Proceedings of IEEE Conference Computer Vision Pattern Recognition (CVPR)* 2392–2399 (Providence, Rhode, USA, 2012).
- Chen, Y. & Pock, T. Trainable nonlinear reaction diffusion: A flexible framework for fast and effective image restoration. *IEEE Trans. Pattern Anal. Mach. Intell.* **39**, 1256–1272 (2017).
- Schmidt, U. & Roth, S. Shrinkage fields for effective image restoration. In *Proceedings of IEEE Conference Computer Vision Pattern Recognition (CVPR)* 2774–2781 (Columbus, Ohio, USA, 2014).
- You, C. *et al.* Structurally-sensitive multi-scale deep neural network for low-dose CT denoising. *IEEE Access.* **6**, 41839–41855 (2018).
- You, C. *et al.* CT super-resolution GAN constrained by the identical, residual, and cycle learning ensemble (GAN-CIRCLE). *IEEE Trans. Med. Imaging.* **39**, 188–203 (2019).
- Guha, I. *et al.* Deep learning-based high-resolution reconstruction of trabecular bone microstructures from low-resolution CT scans using GAN-CIRCLE. In *Proceedings of SPIE—The International Society for Optical Engineering* Vol. 11317. <https://doi.org/10.1117/12.2549318> (2020).
- You, C., Yang, L., Zhang, Y. & Wang, G. Low-dose CT via deep CNN with skip connection and network-in-network. In *Proceedings of SPIE in Developments in X-ray Tomography XII* Vol. 11113 (2019). <https://doi.org/10.1117/12.2534960>.
- Lyu, Q., You, C., Shan, H., Zhang, Y. & Wang, G. Super-resolution MRI and CT through GAN-circle. In *Proceedings of SPIE in Developments in X-Ray Tomography XII* Vol. 11113 (2019). <https://doi.org/10.1117/12.2530592>
- Li, S. H., Zhang, B., Yang, X. F., He, Y. X. & Chen, M. Depth image super resolution for 3D reconstruction of oil refinery buildings. *Chem Tech Fuels Oil+* **55**, 491–496 (2019).
- Sun, L. *et al.* A novel weighted cross total variation method for hyperspectral image mixed denoising. *IEEE Access.* **5**, 27172–27188 (2017).
- Rudin, L. I., Osher, S. & Fatemi, E. Nonlinear total variation-based noise removal algorithms. *Physica D* **60**, 259–268 (1992).
- Ranfl, R., Bredies, K. & Pock, T. Non-local total generalized variation for optical flow estimation. In *Proceedings of European Conference on Computer Vision (ECCV)* 439–454 (Springer, Cham, 2014).
- Bredies, K., Kunisch, K. & Pock, T. Total generalized variation. *SIAM J. Imaging Sci.* **3**, 492–526 (2010).
- Chambolle, A. & Pock, T. A first-order primal-dual algorithm for convex problems with applications to imaging. *J. Math. Imaging Vis.* **40**, 120–145 (2011).
- Rockafellar, R. T. & Zagrodny, D. A derivative-coderivative inclusion in second-order non-smooth analysis. *Set Valued Anal.* **5**, 89–105 (1997).
- Bredies, K., Dong, Y. & Hintermüller, M. Spatially dependent regularization parameter selection in total generalized variation models for image restoration. *Int. J. Comput. Math.* **90**, 109–123 (2013).
- Canny, J. A computational approach to edge detection. *IEEE Trans. Pattern Anal. Mach. Intell. PAMI-8*, 679–698 (1986).
- Bertsekas, D. Convex analysis and optimization. *Athena Sci.* **129**, 420–432 (2010).
- Bredies, K., Kunisch, K. & Pock, T. Total generalized variation. *SIAM J. Imaging Sci.* **3**, 492–526 (2010).

31. Scharstein, D. & Szeliski, R. High-accuracy stereo depth maps using structured light. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* 195–202 (Madison, WI, 2003).
32. Hirschmuller, H. & Scharstein, D. Evaluation of cost functions for stereo matching. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (CVPR)*1–8 (Minneapolis, MN, 2007).

### Acknowledgements

This work was supported in part by the National Key Research and Development Program of China under Grant 2018YFC1604000 and Grant 2016YFB1101702, in part by the Nature Science Foundation of China under Grant 61772573, and in part by the National Mapping and Geographic Information Bureau of China under Grant SIDT20170101.

### Author contributions

Quantitative analysis and qualitative analysis was provided by B.Z., City University of Hong Kong. Writing assistance was provided by W.Z. and X.Y., Wuhan University. The main manuscript text was written by S.L., Sichuan International Studies University.

### Competing interests

The authors declare no competing interests.

### Additional information

**Correspondence** and requests for materials should be addressed to S.L.

**Reprints and permissions information** is available at [www.nature.com/reprints](http://www.nature.com/reprints).

**Publisher's note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**Open Access** This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

© The Author(s) 2020