AIMS Mathematics, 10(12): 28207–28220.
DOI: 10.3934/math.20251240
Received: 28 July 2025
Revised: 02 November 2025
Accepted: 20 November 2025
Published: 01 December 2025

https://www.aimspress.com/journal/Math

Research article

An adaptive anisotropic diffusion model integrating total variation and non-local means for image denoising

Xiaojuan Zhang*

School of Mathematics and Statistics, North China University of Water Resources and Electric Power, Zhengzhou,450046, China

* **Correspondence:** Email: zhangxiaojuan@ncwu.edu.cn; Tel:+86 13674926126.

Abstract: This study proposes an adaptive diffusion equation for noise removal that combines total variation (TV) and non-local means (NL-means). By incorporating a weighted non-local data fidelity term, the model adaptively switches between TV and NL-means based on image features. A key advantage of this approach is its ability to correct over-smoothed low-contrast areas, minimize residual noise near edges, and reduce staircasing artifacts during denoising. Furthermore, the existence of a weak solution for the proposed model is rigorously established.

Keywords: total variation; non-local means; image denoising; weak solution
Mathematics Subject Classification: 35K25, 35K61, 68U10

1. Introduction

Image denoising is a fundamental challenge in image processing. This problem can be formally stated as the recovery of a clean image u from its noisy observation as follows:

$$f = u + n$$

where f denotes the observed noisy image, and n denotes the additive white Gaussian noise with a zero mean. Numerous effective approaches have been developed to address this inverse problem and estimate solutions, such as, anisotropic diffusion PDE models [1–3]variational methods [4–6] and nonlocal methods [7–10].

The total variation (TV) minimization was introduced by Rudin, Osher, and Fatemi [4]. This method is particularly effective for denoising while preserving edges; however, it presents three major drawbacks: textures tend to be overly smoothed, flat areas are approximated by a piecewise constant surface, resulting in a staircasing effect, and the image suffers from contrast loss. Subsequent research has developed various improvements to address the limitations of TV-based methods [7, 8, 10].

The non-local means (NL-means) algorithm proposed by Buades et al. [7] is a highly regarded contemporary denoising technique. Despite its strong performance, it tends to oversmooth details [10]. In response, Louchet and Moisan [10] introduced the TV-means algorithm, which combines NL-means with TV regularization in an iterative process to reduce staircasing and rare-patch effects. Building on this, Sutour et al. [8] devised an adaptive framework that uses a weighted non-local data fidelity term to automatically tune the TV regularization based on NL-means confidence. Shi [11] proposed a coupled local-nonlocal diffusion equation to mitigate artifacts inherent in both approaches.

This article presents a hybrid TV and NL-means model that compensates for their individual shortcomings. Our framework adaptively selects between the two methods: NL-means for preserving textured regions with repetitive patterns, and TV for restoring edges and homogeneous areas.

The remainder of this paper is organized as follows. Section 2 reviews the relevant literature. Section 3 introduces the proposed model and establishes the existence of a weak solution. Section 4 details the numerical scheme and presents a comparative analysis of denoising results. Finally, Section 5 concludes the paper.

2. Related methods

2.1. The TV model

The total variation (TV) model [4] uses total variation regularization to enforce image smoothness while preserving edge information. The denoised image u^{TV} is obtained by solving the following energy-minimization problem:

$$u^{TV} = \underset{u\in\mathbb{R}^n}{\operatorname{argmin}} \left(\frac{\lambda}{2} \int_\Omega (u-f)^2 dx + \int_\Omega |\nabla u| dx \right). \tag{2.1}$$

The first term is the data fidelity, while the second is the TV regularization. The parameter $\lambda > 0$ balances the weight between these two terms. This minimization problem can be solved by evolving the associated partial differential equation:

$$\begin{cases} u_t = \operatorname{div}\left(\frac{\nabla u}{|\nabla u|}\right) + \lambda(f-u), & \text{in } \Omega_T, \\ \frac{\partial u}{\partial n} = 0, & \text{on } \partial\Omega \times (0,T), \end{cases} \tag{2.2}$$

where $\Omega_T = \Omega \times (0,T]$, with $\Omega \subset \mathbb{R}^2$ being a bounded smooth domain with Lipschitz continuous boundary $\partial\Omega$. Several efficient numerical methods have been developed to solve this minimization problem [12, 13].

2.2. The NL-means algorithm

Non-local methods in image processing extend the classical Yaroslavsky filter [14] and patch-based approaches. The fundamental principle involves estimating the value of each pixel through the weighted averaging of similar pixels across the image. This approach is particularly effective because pixels with analogous local structures (patches) typically provide better estimates than the immediate spatial neighbors.

Buades *et al.* [7] formalized this concept into the well-known non-local means (NL-means) filter:

$$u_i^{NL} = \sum_{j\in\Omega} w_{i,j}^{NL} f_j. \tag{2.3}$$

The weighting coefficients are determined as follows:

$$w_{i,j}^{NL} = \frac{1}{Z_i} \exp\left(-\frac{|V(N_i) - V(N_j)|_{2,\alpha}^2}{h^2}\right) \tag{2.4}$$

where Z_i is the normalization factor that ensures that the weights sum to 1, N_k represents a fixed-size square neighborhood centered at pixel k, the similarity metric $|V(N_i) - V(N_j)|_{2,\alpha}$ computes the weighted Euclidean distance between patches, $\alpha > 0$ denotes the standard deviation of the Gaussian kernel used for distance weighting, and h controls the decay of the exponential function.

2.3. Regularized NL-Means(R-NL)

Sutour *et al.* [8] introduced a variational approach that corrects the over-smoothing and reduces the residual noise of the NL-means by adaptively regularizing non-local methods with the total variation. They assume that in the non-local neighborhood of pixel i, the observations f_i are all realizations of the random variable $f_i = u_i + n_i$. u_i represents the true signal with a standard deviation σ_i^{signal}. n_i denotes the noise with a zero mean and known standard deviation σ_i^{noise}. Both random variables are assumed to be independent.

The regularized estimate combines non-local means denoising with the original data through a convex combination as follows:

$$u_i^{NLDJ} = (1 - \alpha_i)u_i^{NL} + \alpha_i f_i. \tag{2.5}$$

The confidence index α_i is defined as

$$\alpha_i = \frac{(\sigma_i^{signal})^2}{(\sigma_i^{signal})^2 + (\sigma_i^{noise})^2} \approx \frac{|(\hat{\sigma}_i^{NL})^2 - (\sigma_i^{noise})^2|}{|(\hat{\sigma}_i^{NL})^2 - (\sigma_i^{noise})^2| + (\sigma_i^{noise})^2} \tag{2.6}$$

with the empirical variance estimate

$$(\hat{\sigma}_i^{NL})^2 = \sum_{j\in\Omega} w_{i,j}^{NL} f_j^2 - \left(\sum_{j\in\Omega} w_{i,j}^{NL} f_j\right)^2. \tag{2.7}$$

Note that the solution u_i^{NLDJ} can be rewritten as the weighted sum

$$u_i^{NLDJ} = \sum_{j\in\Omega} w_{i,j}^{NLDJ} f_j \tag{2.8}$$

where $w_{i,j}^{NLDJ} = (1 - \alpha_i)w_{i,j}^{NL} + \alpha_i\delta_{i,j}$ and $\delta_{i,j}$ is the Kronecker delta function.

The complete regularization framework performs TV minimization with a non-local fidelity term as follows:

$$u^{R-NL} = \underset{u\in\mathbb{R}^n}{\operatorname{argmin}}\left(\sum_{i\in\Omega} \lambda_i \sum_{j\in\Omega} w_{i,j}^{NLDJ}(f_j - u_j)^2 + \sum_{i\in\Omega} |\nabla u_i|\right). \tag{2.9}$$

This optimization problem can be formulated as

$$u^{R-NL} = \underset{u\in\mathbb{R}^n}{\operatorname{argmin}} \left(\sum_{i\in\Omega} \lambda_i (u_i - u_i^{NLDJ})^2 + \sum_{i\in\Omega} |\nabla u_i| \right) \tag{2.10}$$

with adaptive regularization parameters

$$\lambda_i = \gamma \left[\sum_{j\in\Omega} (w_{i,j}^{NLDJ})^2 \right]^{-1/2} \tag{2.11}$$

where $\gamma > 0$ is a fixed parameter that sets the strength of the adaptive regularization.

2.4. Adaptive forward-backward diffusion equation (AFBD)

Building upon the theoretical connection between the variational formulation (2.1) and its corresponding PDE (2.2), Prasath *et al.* [15] developed an adaptive forward–backward diffusion (AFBD) model:

$$\frac{\partial u}{\partial t} = \operatorname{div}\left(h(\nabla G_\sigma * u) \frac{\nabla u}{|\nabla u|} \right) \tag{2.12}$$

where $h(\nabla G_\sigma * u)$ serves as an edge-sensitive diffusion coefficient.

$$h(\nabla G_\sigma * u) = \frac{1}{1 + K|\nabla G_\sigma * u|^2}, \quad K > 0, \tag{2.13}$$

where G_σ represents a two-dimensional Gaussian kernel.

Although the authors established the existence of weak solutions for the AFBD model in (2.12), and experimental validation showed that it successfully avoids the edge-smearing of conventional diffusion while maintaining its benefits, the method still exhibits staircasing artifacts in noisy regions.

3. The proposed model

Motivated by the framework of [15] and the regularization functional (2.10), we consider the following homogeneous Dirichlet problem:

$$\begin{cases} u_t = \operatorname{div}\left(h(\nabla G_\sigma * u) \frac{\nabla u}{|\nabla u|} \right) + \lambda(u^{NLDJ} - u), & \text{in } \Omega_T, \\ \frac{\partial u}{\partial n} = 0, & \text{on } \partial\Omega \times (0,T), \\ u(x,0) = f(x), & \text{in } \Omega, \end{cases} \tag{3.1}$$

where $\lambda(x)$ is the continuous formulation of (2.11), defined as

$$\lambda(x) = \gamma \left(\int_\Omega w^2(x,y)dy \right)^{-1/2}. \tag{3.2}$$

The discrete counterpart of $w(x,y)$ is defined as $w_{i,j}^{NLDJ}$.

The inverse mollification function $h(\nabla G_\sigma * u)$ self-adjusts based on an estimate of the edge information from the smoothed gradient of the image. By reducing the TV flow in homogeneous regions, the staircasing problem is alleviated.

This hybrid model adaptively blends TV's edge preservation and NL-means' texture denoising via a residual-variance-guided parameter $\lambda(x)$. In homogeneous regions, high patch redundancy yields low variance and a large $\lambda(x)$, favoring NL-means for smooth results. Near edges, low redundancy increases variance and shrinks $\lambda(x)$, engaging TV to maintain edge sharpness. This adaptive mechanism strikes an excellent balance between effective noise reduction, sharp edge preservation, and the maintenance of fine texture details.

We now establish the existence of a weak solution to the proposed problem (3.1).

Definition 1. (Test Function Space [3]) Let $C^k(\Omega\times(0,T);\mathbb{R}^2)$ denote the space of k-order continuously differentiable functions. For any $v=(v_1,v_2)\in C^k(\Omega\times(0,T);\mathbb{R}^2)$ satisfying the boundary conditions:

$$\frac{\partial^i v_1}{\partial x^i}n_1\Big|_{\partial\Omega\times(0,T)}=0 \quad \text{and} \quad \frac{\partial^i v_2}{\partial y^i}n_2\Big|_{\partial\Omega\times(0,T)}=0$$

for all $i=0,1,...,k-1$, then v belongs to a space which is denoted by $\mathcal{V}_0^k(\Omega\times(0,T);\mathbb{R}^2)$.

Definition 2. (Weighted Partial Variation [3]) Let $\mathcal{M}$ be the Banach space of finite signed Radon measures on $\Omega\times(0,T)$, dual to $C_0(\Omega\times(0,T))$. For $u\in\mathcal{M}$ and $\Phi\in C(\overline{\Omega}\times[0,T])$ with $\Phi\geq 0$, the weighted partial k-order variation is:

$$\mathrm{PV}_\Phi^k(u)=\sup_{\psi\in\mathcal{V}_0^k(\Omega\times(0,T);\mathbb{R}^2)\ |\psi|\leq\Phi}\langle u,\mathrm{div}^k\psi\rangle\mathcal{M}\times C_0$$

where $|\psi|=\sqrt{\psi_1^2+\psi_2^2}$ and $\mathrm{div}^k\psi=\frac{\partial^k\psi_1}{\partial x^k}+\frac{\partial^k\psi_2}{\partial y^k}$.

Proposition 1. [3] Assume that $u\in L^1(0,T;W_1^k(\Omega))$. Then, $\mathrm{PV}_\Phi^k(u)=\int_0^T\int_\Omega\Phi(x,t)|D^k u(x,t)|dxdt$.

The space functions of k-bounded partial variation on $\Omega\times(0,T)$ can be defined by

$$\mathrm{BPV}^k=\left\{u\in\mathcal{M}\,\middle|\,\|u\|_{\mathrm{BPV}^k}:=\|u\|_{\mathcal{M}}+\mathrm{PV}_1^k(u)<+\infty\right\}.$$

Definition 3. A function u in the class

$$u\in C_w(0,T;L^2(\Omega))\cap\mathrm{BPV}\cap W_\infty^1(0,T;[L^2\cap W^{1,1}(\Omega)]^*) \tag{3.3}$$

is called a weak solution to problem (3.1) with $f\in L^2(\Omega)$ if

(i) there exists $z\in L^\infty(\Omega\times(0,T);\mathbb{R}^2)$, $\|z\|_{L^\infty}\leq 1$, so that for all $v\in L^2\cap W^{1,1}(\Omega)$,

$$\left\langle\frac{du}{dt},v\right\rangle+(h(\nabla G_\sigma*u)z,\nabla v)+(\lambda(u^{NLDJ}-u),v)=0 \tag{3.4}$$

a.e. on $(0,T)$;

(ii) for all $w\in L^1(0,T;W^{1,1}(\Omega))\cap W^{1,1}(0,T;L^2(\Omega))$, one has

$$\begin{aligned}&\|u(T)-w(T)\|^2+2\int_0^T\Big(\frac{dw}{ds}(s),u(s)\Big)ds+2PV^1_{h(\nabla G_\sigma*u)}(u)+2\int_0^T\Big(\lambda(u^{NLDJ}-u)(s),u(s)-w(s)\Big)ds\\ \leq&\|f-w(0)\|^2+\|w(T)\|^2-\|w(0)\|^2+2\int_0^T\Big(h(\nabla G_\sigma*u)(s)z(s),\nabla w(s)\Big)ds;\end{aligned} \tag{3.5}$$

(iii) the initial condition f holds in the space $L^2(\Omega)$.

We first need to study the following auxiliary problem:

$$\begin{cases} \frac{\partial u}{\partial t} + \epsilon A(u) = \operatorname{div}(h(\nabla G_\sigma * u)\frac{\nabla u}{\epsilon+|\nabla u|}) + \lambda(u^{NLDJ} - u), & \text{in } \Omega_T, \\ \frac{\partial u}{\partial \boldsymbol{n}} = 0, & \text{on } \partial\Omega \times (0,T), \\ u(x,0) = f(x), & \text{in } \Omega, \end{cases} \tag{3.6}$$

where $\epsilon > 0$ is parameter.

Lemma 1. Let $f \in L^2(\Omega)$. Problem (3.6) admits a solution u in the class

$$L^2(0,T;H^m(\Omega)) \cap H^1(0,T;(H^m(\Omega))^*) \cap C([0,T];L^2(\Omega)), \tag{3.7}$$

where $m > 4$.

Lemma 2. Let u be a solution to problem (3.6). Then, u satisfies the a priori bound

$$\|u\|_{L^\infty(0,T;L^2(\Omega))} + \|u\|_{L^1(0,T;W^{1,1}(\Omega))} + \|\frac{du}{dt}\|_{L^2(0,T;(H^m(\Omega))^*)} \le C. \tag{3.8}$$

The constant C is independent of ϵ.

Theorem 1. Letting $f \in L^2(\Omega)$, there exists a weak solution to (3.1) in the class (3.3).

Owing to the similarity of the proofs to the analogous arguments in [3, 15], the proofs of Lemmas 1 and 2 and Theorem 1 are omitted.

4. Numerical implementation of the proposed model

4.1. Numerical approximation

The proposed model (3.1) is discretized using standard finite difference schemes and solved numerically via the additive operator splitting (AOS) method [16]. In our numerical implementation, we represent an m-dimensional discrete image as a vector $f \in R^N$, where each component $f_i (i = 1, \cdots, N)$ corresponds to the grayscale intensity at pixel location x_i. The spatial discretization uses grid spacing h_l in each dimension l, while temporal discretization employs time steps $t_k := k\tau$ for $k \in N$ with step size τ. The numerical approximations $u_i^k \approx u(x_i, t_k)$ and $C_i^k \approx h(\nabla G_\sigma * u(x_i, t_k))\frac{1}{|\nabla u(x_i,t_k)|}$ are computed using central difference approximations for the gradient terms, ensuring second-order spatial accuracy in our implementation.

The simplest discretization of (3.1) with reflecting boundary conditions is

$$\frac{u_i^{k+1} - u_i^k}{\tau} = \sum_{l=1}^{m} \sum_{j \in N_l(i)} \frac{C_j^k + C_i^k}{2h_l^2}(u_j^{k+1} - u_i^{k+1}) + \lambda_i(u_i^{NLDJ} - u_i^k)$$

where $N_l(i)$ consists of the two neighbors of pixel i along the l direction (boundary pixels may have only one neighbor). In vector-matrix notation, this becomes

$$\frac{u^{k+1} - u^k}{\tau} = \sum_{l=1}^{m} A_l(u^k)u^{k+1} + \lambda(u^{NLDJ} - u^k)$$

where $A_l(u^k) = [a^l_{ij}(u^k)]$ with

$$a^l_{ij}(u^k) := \begin{cases} \dfrac{C^k_i + C^k_j}{2h^2_l}, & j \in N_l(i), \\ -\displaystyle\sum_{n \in N_l(i)} \dfrac{C^k_i + C^k_n}{2h^2_l}, & j = i, \\ 0, & \text{otherwise.} \end{cases}$$

The additive operator splitting (AOS) [16] scheme is given by

$$u^{k+1} = \frac{1}{m} \sum_{l=1}^{m} \left(I - m\tau A_l(u^k)\right)^{-1} [u^k + \tau\lambda(u^{NLDJ} - u^k)].$$

The operators $B_l(u^k) = I - m\tau A_l(u^k)$ govern one-dimensional diffusion processes along each spatial dimension x_l. When employing consecutive pixel numbering along the l-th coordinate direction, these operators reduce to strictly diagonally dominant tridiagonal matrices. We refer readers to [16] for complete implementation details.

4.2. Parameters setting

The quantitative evaluation uses two widely adopted image quality metrics: peak signal-to-noise ratio (PSNR) and mean structural similarity index (MSSIM), where higher values indicate better denoising performances. All experiments were conducted using MATLAB R2020a on an 11th Gen Intel® Core™ i7-1165G7 processor (2.80 GHz) and 8 GB RAM. To validate the effectiveness of our proposed method (3.1), we performed comprehensive comparative evaluations against four established denoising approaches: TV [4], NL-means [7], R-NL [8], and BM3D [17].

The experimental configuration uses consistent parameters across all evaluations: A fixed time step $\tau = 0.2$, $K = 10^{-4}$, and spatial filtering with a 5×5 rotationally symmetric Gaussian lowpass filter ($\sigma = 0.8$). The algorithm utilizes standard 5×5 pixel patches and 21×21 search windows for non-local processing. The TV [4] regularization parameter λ was set to 0. The parameter γ in our model (3.1) was adjusted adaptively: A value of 10 was used for low-texture images with low noise, while it was set to 100 for texture-rich images with medium or high noise levels. The stopping time was chosen so that the best PSNR is obtained.

4.3. Test cases and discussion

Figure 1 shows the ten test images used for the experimental validation. Table 1 presents quantitative comparisons between the proposed method and state-of-the-art approaches, evaluating both PSNR and MSSIM metrics across all test images corrupted by additive white Gaussian noise (AWGN) with standard deviations of 20, 30, and 40. As can be noted, the proposed scheme performs well for a variety of images. Compared with TV [4], NL-means [7], and R-NL [8], the proposed method achieves the best performance in terms of both PSNR and MSSIM. It also surpasses BM3D in terms of MSSIM for the Mosaic. The proposed method outperformed TV [4], NL-means [7], and R-NL [8]. The proposed method required more time than the other four methods. For a 512×512 image corrupted by AWGN

with a standard deviation of 40, the proposed algorithm requires approximately 50 s to obtain the best PSNR.

Table 1. Qualitative comparison of denoising algorithms.

Images	σ	TV [4] PSNR/MSSIM	NL-means [7] PSNR/MSSIM	R-NL [8] PSNR/MSSIM	BM3D [17] PSNR/MSSIM	Proposed PSNR/MSSIM
Lena	20	31.48/0.9157	31.11/0.9067	32.00/0.9256	**33.08/0.9414**	32.07/0.9258
	30	29.82/0.8811	29.20/0.8682	30.15/0.8929	**31.30/0.9140**	30.25/0.8942
	40	28.70/0.8534	27.74/0.8308	28.73/0.8614	**30.00/0.8875**	28.85/0.8639
Cameraman	20	32.26/0.9149	31.92/0.9049	32.74/0.9269	**34.21/0.9452**	32.83/0.9269
	30	30.38/0.8779	29.97/0.8691	30.67/0.8907	**32.19/0.9186**	30.76/0.8915
	40	29.02/0.8464	28.49/0.8382	29.33/0.8650	**30.82/0.8940**	29.46 /0.8667
Airplane	20	31.59/0.9229	30.92/0.9150	31.83/0.9314	**32.79/0.9435**	32.04/0.9362
	30	29.67/0.8871	28.91/0.8827	29.77/0.9019	**30.82/0.9197**	30.09/0.9118
	40	28.29/0.8541	27.31/0.8453	28.35/0.8765	**29.42/0.8973**	28.62/0.8881
Parrot	20	28.61/0.8290	28.33/0.8236	29.55/0.8588	**29.89/0.8705**	29.57/0.8598
	30	26.76/0.7826	26.95/0.7770	27.61/0.8159	**28.06/0.8330**	27.72/0.8190
	40	25.51/0.7544	25.67/0.7296	26.07/0.7817	**26.83/0.8229**	26.35/0.7880
Hats	20	31.93/0.9066	32.10/0.9155	32.91/0.9313	**33.71/0.9430**	32.95/0.9321
	30	30.38/0.8807	30.23/0.8793	30.99/0.8988	**31.90/0.9162**	31.06/0.9016
	40	29.46/0.8601	28.83/0.8439	29.80/0.8752	**30.68/0.8916**	29.92/0.8778
Boat	20	29.29/0.8937	28.68/0.8559	29.93/0.9008	**30.92/ 0.9269**	29.95/0.9025
	30	27.61/0.8451	26.79/0.8011	27.94/0.8433	**29.11/0.8866**	28.03/0.8483
	40	26.47/0.7987	25.49/0.7519	26.52/0.7903	**27.80/0.8478**	26.68/0.7994
Barbara	20	26.59/0.8415	29.53/0.9089	30.18/0.9294	**32.08/0.9508**	30.35/0.9316
	30	24.76/0.7699	27.07/0.8556	27.62/0.8762	**29.99/0.9223**	28.01/0.8845
	40	23.86/0.7481	25.44/0.8043	25.82 /0.8258	**28.42/0.8899**	26.26/0.8437
Monarch	20	31.44/0.9430	31.12/0.9546	32.23/0.9658	**32.59/0.9665**	32.30/0.9678
	30	29.36/0.9133	28.99/0.9269	29.93/0.9467	**30.59/0.9483**	30.09/0.9504
	40	27.85/0.8875	27.32/0.8967	28.22/0.9243	**29.16/0.9301**	28.59/0.9304
Man	20	29.69/0.8977	28.75/0.8543	29.97/0.8926	**30.58/0.9166**	30.12/0.9052
	30	28.09/0.8484	27.06/0.7972	28.05/0.8332	**28.81/0.8720**	28.35/0.8556
	40	26.98/0.8074	25.93/0.7508	26.76/0.7827	**27.64/0.8332**	27.19/0.8131
Mosaic	20	36.75/0.9829	37.63/0.9807	**40.49**/0.9887	39.94/0.9916	40.40/**0.9928**
	30	34.64/0.9758	34.26/0.9544	**36.90**/0.9767	36.88/0.9805	36.85/**0.9831**
	40	33.05/0.9695	32.14/0.9260	34.32/0.9611	**34.80**/0.9649	34.41/**0.9704**

(a) Lena (b) Cameraman (c) Airplane (d) Parrot (e) Hats

(f) Boat (g) Barbara (h) Monarch (i) Man (j) Mosaic

Figure 1. Test images used in Table 1.

Figures 2–4 show a visual comparison of natural images corrupted by AWGN with standard deviations of 30 and 40. Figures 2 and 3 present the restored images of the Barbara image, which shows that the TV not only introduces a staircasing effect on Barbara's face and the background, but also fails to preserve the texture. Although the NL-means [7] method preserves the texture, its results suffer from oversmoothing. The texture recovery performance of BM3D [17] is inferior to those of R-NL [8] and our method. BM3D [17] introduces artifacts, particularly in the facial region. Our proposed method and R-NL [8] effectively preserve texture while achieving natural facial restoration. Figure 4 presents the restoration results of the five methods for the test image Airplane with $\sigma_n = 40$. The regions in the red squares are displayed in the lower-right corner for a better visual comparison. The TV [4] produces noticeable staircasing artifacts, whereas the NL-means [7] leads to over-smoothed number regions. Our method outperforms R-NL [8] in digital restoration, but its results are still inferior to those of BM3D [17].

(a) Barbara

(b) Noise

(c) TV [4]

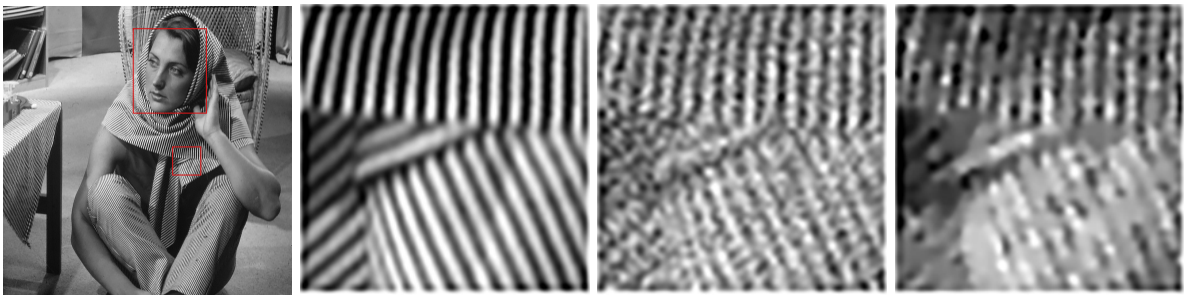
(d) NL-means [7]

(e) R-NL [8]

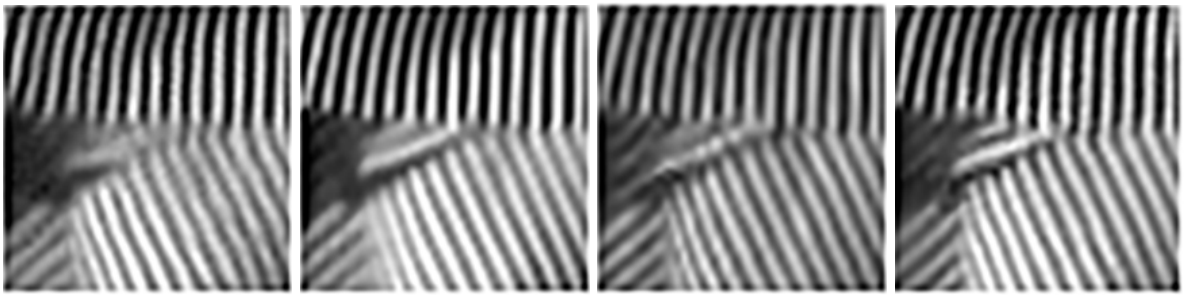
(f) BM3D [17]

(g) Proposed

Figure 2. Restoration results of five methods for test image Barbara with $\sigma_n = 30$.



(a) Original (b) Original (c) Noise (d) TV [4]

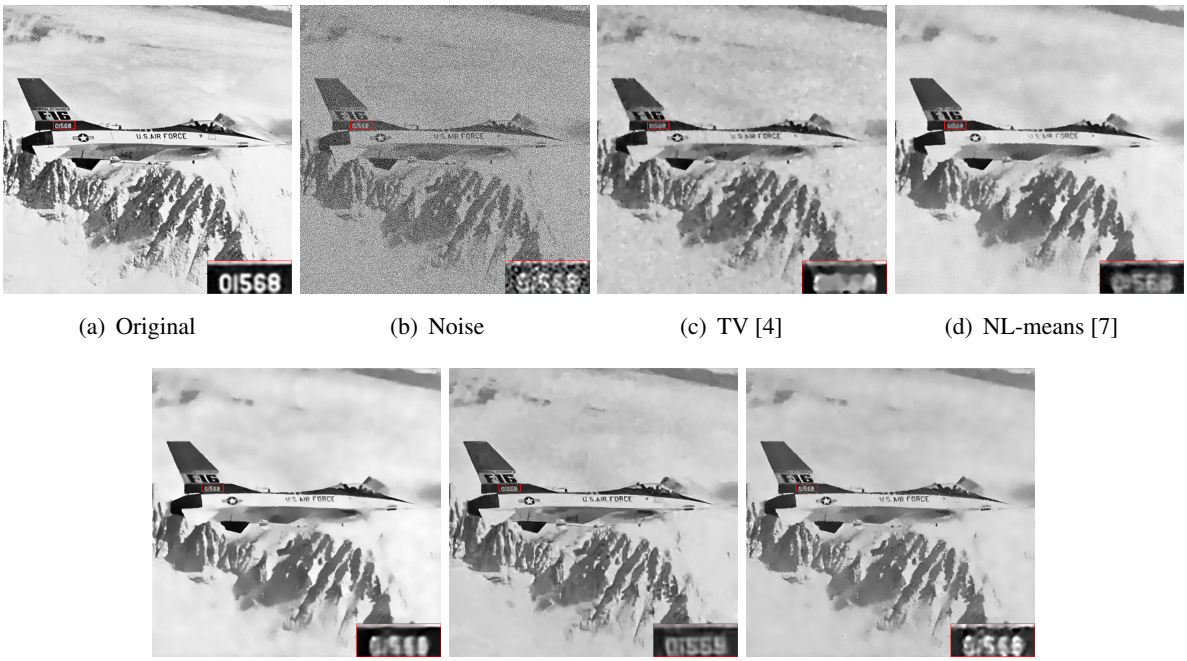


(e) NL-means [7] (f) R-NL [8] (g) BM3D [17] (h) Proposed

(i) Original (j) Noise (k) TV [4] (l) NL-means [7]

(m) R-NL [8] (n) BM3D [17] (o) Proposed

Figure 3. Zoomed parts of Barbara are shown for better visual comparison.



(a) Original (b) Noise (c) TV [4] (d) NL-means [7]

(e) R-NL [8] (f) BM3D [17] (g) Proposed

Figure 4. Restoration results of five methods for test image Airplane with $\sigma_n = 40$.

5. Conclusions

In this study, we have proposed an adaptive diffusion model that utilizes a combination of TV and NL-means terms. The proposed model incorporates a weighted non-local data fidelity term that dynamically adjusts the balance between TV and NL-means regularization based on local image features. We have investigated the existence of a weak solution for the new model. Comprehensive numerical experiments have demonstrated that the proposed method can effectively suppresses noise while preserving textural details and maintaining structural integrity across various noise levels.

Use of Generative-AI tools declaration

The authors declare they have used Deepseek in this article. The AI tools were used exclusively for the purpose of language polishing, and improving the fluency and readability of the manuscript.The AI-assisted improvements are applied throughout the entire text of the manuscript to improve its linguistic quality, and do not affect the core scientific meaning or findings.

Conflict of interest

The authors declare no conflict of interest.

References

1. P. Perona, J. Malik, Scale-space and edge detection using anisotropic diffusion, *IEEE Trans. Pattern Anal. Mach. Intell.,* **12** (1990), 629–639. https://doi.org/10.1109/34.56205
2. H. K. Rafsanjani, M. H. Sedaaghi, S. Saryazdi, Efficient diffusion coefficient for image denoising, *Comput. Math. Appl.,* **72** (2016), 893–903. https://doi.org/10.1016/j.camwa.2016.06.005
3. X. Zhang, W. Ye, An adaptive fourth-order partial differential equation for image denoising, *Comput. Math. Appl.,* **74** (2017), 2529–2545. https://doi.org/10.1016/j.camwa.2017.07.036
4. L. I. Rudin, S. Osher, E. Fatemi, Nonlinear total variation based noise removal algorithms, *Phys. D: Nonlinear Phenom.,* **60** (1992), 259–268. https://doi.org/10.1016/0167-2789(92)90242-F
5. C. Antonin, L. P. Louis, Image recovery via total variation minimization and related problems, *Numer. Math.,* **76** (1997), 167–188. https://doi.org/10.1007/s002110050258
6. T. Chan, A. Marquina, P. Mulet, High-order total variation-based image restoratio, *SIAM J. Sci. Comput.,* **22** (2000), 503–516. https://doi.org/10.1137/S1064827598344169
7. A. Buades, B. Coll, J. M. Morel, A review of image denoising algorithms, with a new one, *Multiscale Model. Simul.,* **4** (2005), 490–530. https://doi.org/10.1137/040616024
8. C. Sutour, C. A. Deledalle, J. F. Aujol, Adaptive regularization of the NL-Means: Application to image and video denoising, *IEEE Trans. Image Process.,* **23** (2014), 3506–3521. https://doi.org/10.1109/TIP.2014.2329448
9. J. Delon, A. Desolneux, C. Sutour, A. Viano, RNLp: Mixing nonlocal and TV-Lp methods to remove impulse noise from image, *J. Math. Imaging Vis.,* **61** (2019), 458–481. https://doi.org/10.1007/s10851-018-0856-3
10. C. Louchet, L. Moisan, Total variation as a local filter, *SIAM J. Imaging Sci.,* **4** (2011), 651–694. https://doi.org/10.1137/100785855
11. K. Shi, Coupling local and nonlocal diffusion equations for image denoising, *Nonlinear Anal. Real World Appl.,* **62** (2021), 103362. https://doi.org/10.1016/j.nonrwa.2021.103362
12. A. Chambolle, An algorithm for total variation minimization and applications, *J. Math. Imaging Vis.,* **20** (2004), 89–97. https://doi.org/10.1023/B:JMIV.0000011325.36760.1e
13. A. Chambolle, T. Pock, A first-order primal-dual algorithm for convex problems with applications to imagin, *J. Math. Imaging Vis.,* **40** (2011), 120–145. https://doi.org/10.1007/s10851-010-0251-1
14. L. P. Yaroslavsky, *Digital picture processing: An introduction*, Springer Science & Business Media, 2012.
15. V. B. S. Prasath, J. M. Urbano, D. Vorotnikov, Analysis of adaptive forward-backward diffusion flows with applications in image processing, *Inverse Probl.,* **31** (2015), 105008. https://dx.doi.org/10.1088/0266-5611/31/10/105008
16. J. Weickert, B. Romeny, M. Viergever, Efficient and reliable schemes for nonlinear diffusion filtering, *IEEE Trans. Image Process.,* **7** (1998), 398–410. https://doi.org/10.1109/83.661190

17. K. Dabov, A. Foi, V. Katkovnik, K. Egiazarian, Image denoising by sparse 3-D transform-domain collaborative filtering, *IEEE Trans. Image Process.,* **16** (2007), 2080–2095. https://doi.org/10.1109/TIP.2007.901238

AIMS Press