Niğde Ömer Halisdemir Üniversitesi Mühendislik Bilimleri Dergisi
Niğde Ömer Halisdemir University Journal of Engineering Sciences

Araştırma makalesi / Research article

www.dergipark.org.tr/tr/pub/ngumuh / www.dergipark.org.tr/en/pub/ngumuh

An image enhancement approach based on total variation denoising and high-pass filtering with low-runtime

Toplam varyasyon tabanlı gürültü giderme ve yüksek geçiren filtrelemeye dayalı düşük işlem süresine sahip bir görüntü iyileştirme yaklaşımı

Kubilay Küçük[1,], Derya Yılmaz[2,*]

[1,2] Gazi University, Faculty of Engineering, Department of Electrical and Electronics Engineering, 06570, Ankara, Türkiye
[1] Roketsan Inc., Department of Flight Test Engineering and Technology, 06780, Ankara, Türkiye

Abstract

The increasing demand for high-quality visual content necessitates the development of image enhancement techniques which can effectively mitigate noise while preserving essential details. Traditional methods often struggle to balance noise elimination and detail protection. This challenge is further compounded when computational resources are limited, making it imperative to devise solutions that are both efficient and effective. In response to these challenges, an approach to image enhancement that integrates total variation denoising (TVD) with a high-pass filter (HPF) is presented. The proposed method leverages the edge-preserving capabilities of TVD to reduce noise while maintaining structural integrity, followed by a HPF to enhance fine details and sharpen the image. To emulate realistic camera-induced degradations, a Gaussian blur noise model, which simulates loss of detail caused by flicker, zoom fluctuations, and minor handshake is employed. Experimental results demonstrate that the proposed approach improves visual quality and structural similarity index (SSIM) and additionally achieves substantial runtime advantage. The proposed framework reaches a remarkably good real-time performance of 60 frames per second (FPS) on standard CPU hardware, thereby demonstrating its suitability for time-critical and resource-constrained applications.

Keywords: Image sharpening, Total variation denoising, Edge preservation, Noise reduction, Low run-time

Öz

Yüksek kaliteli görsel içeriğe olan artan talep, temel ayrıntıları korurken gürültüyü etkili bir şekilde azaltabilen görüntü iyileştirme tekniklerinin geliştirilmesini zorunlu kılmaktadır. Geleneksel yöntemler genellikle gürültü giderme ve ayrıntı koruma arasında denge kurmakta zorlanır. Bu zorluk, hesaplama kaynakları sınırlı olduğunda daha da artarak verimli ve etkili çözümler geliştirmeyi zorunlu hale getirir. Bu zorluklara yanıt olarak, toplam değişim gürültü giderme (TVD) ve yüksek geçiren filtreyi (HPF) birleştiren bir görüntü iyileştirme yaklaşımı sunulmaktadır. Önerilen yöntem, yapısal bütünlüğü korurken gürültüyü azaltmak için TVD'nin kenar koruma yeteneklerinden yararlanır ve ardından ince ayrıntıları iyileştirmek ve görüntüyü keskinleştirmek için bir HPF kullanır. Gerçekçi kamera kaynaklı bozulmaları taklit etmek için, titreme, yakınlaştırma dalgalanmaları ve küçük el titremelerinden kaynaklanan ayrıntı kaybını simüle eden bir Gauss bulanıklığı gürültü modeli kullanılmıştır. Deneysel sonuçlar, önerilen yaklaşımın görsel kaliteyi ve yapısal benzerlik indeksini (SSIM) iyileştirdiğini ve ayrıca önemli çalışma süresi avantajı sağladığını göstermektedir. Önerilen çerçeve, standart CPU donanımında saniyede 60 kare (FPS) gibi oldukça iyi bir gerçek zamanlı performans elde ederek, zaman açısından kritik ve kaynak kısıtlı uygulamalar için uygunluğunu göstermektedir.

Anahtar kelimeler: Görüntü keskinleştirme, Toplam değişim gürültü gidermėsi, Kenar koruma, Gürültü azaltma, Düşük işlem süresi

1 Introduction

Image denoising and sharpening are critical tasks in image processing, particularly for enhancing the quality of visual information in applications ranging from medical imaging to satellite image analysis. Denoising aims to remove unwanted noise that arises during acquisition, while sharpening focuses on accentuating edges and details. However, achieving a good balance between noise elimination and detail protection remains challenging: excessive smoothing can lead to the loss of vital structural information, whereas insufficient smoothing may fail to suppress noise adequately.

There are numerous efficient algorithms proposed for improving performance and reducing artefacts such as the staircase effect [1-6]. Among existing denoising techniques, total variation denoising (TVD) has become a cornerstone due to its ability to reduce noise while preserving edges [1]. On the other hand, image sharpening methods typically rely on emphasizing high-frequency components. High-pass filter (HPF) is widely used to enhance edges and textures and has been implemented in both spatial and frequency domains using convolution kernels, transform-based representations, and adaptive schemes [7-11].

* Corresponding author, e-mail: deryayilmaz@gazi.edu.tr (D. Yılmaz)
Geliş / Received: 05.12.2025 Kabul / Accepted: 25.05.2026 Yayımlanma / Published: 14.07.2026
doi: 10.28948/ngumuh.1836401

Beyond TVD, conventional filtering-based denoising techniques such as non-local means and bilateral filtering have been extensively studied [12, 13], together with diffusion- and spatial-domain filters designed for robust noise reduction [14, 15]. Following these classical approaches, a broad line of research has explored advanced sharpening and enhancement strategies, including adaptive guided filtering, diffusion-based sharpening and framelet-based operators, often combining denoising and sharpening in a unified framework [16-21]. Pham and Jeon [16] proposed an adaptive guided filtering-based framework that iteratively refines local structures through edge-aware smoothing and sharpening operations. Although effective, the method involves multiple local window computations and adaptive weight estimations, which increase computational overhead. Horiuchi et al. [20] introduced an adaptive color image sharpening technique based on spatial-domain filtering strategies, requiring pixel-wise adaptive parameter updates. Similarly, Toh and Mat Isa [21] proposed a locally adaptive bilateral clustering method that integrates edge-preserving filtering with clustering mechanisms. While these approaches achieve satisfactory visual performance, their adaptive and clustering-based formulations introduce additional computational complexity.

In many real imaging scenarios, degradations are not limited to stochastic noise but also include blur introduced by camera flicker, zoom fluctuations or minor hand-held shake. Such distortions are effectively modeled with a camera point-spread function (PSF) [22]. For mild degradations, this PSF is commonly approximated by an isotropic Gaussian kernel, providing a simple yet realistic model of high-frequency attenuation [23].

Performing denoising and sharpening jointly offers the potential to simultaneously suppress noise and enhance salient image features. Prior work has demonstrated that appropriate combinations of these processes can significantly improve perceptual quality and quantitative metrics in challenging settings [12]. Nevertheless, many existing methods either incur high computational cost or struggle to maintain a good consistency between noise denoising and edge conservation, particularly in resource-constrained environments.

In this study, a method that integrates TVD with a HPF to effectively denoise and sharpen images is presented. The TVD component is used to reduce noise while maintaining the important structure, and subsequent HPF step enhances fine details and edges. The overall procedure is designed using fundamental mathematical operations to be suitable for implementations with limited computational resources.

The proposed framework:

i. implements a shift-based primal-dual TVD scheme using explicit mathematical operators,

ii. combines it with a separable high-pass enhancement stage in a unified and deterministic structure, and

iii. achieves real-time performance exceeding 60 FPS under CPU-only constraints without relying on GPU acceleration or external optimization toolboxes.

This procedure distinguishes the proposed approach from prior denoising-sharpening combinations by emphasizing structural simplicity, deterministic execution, and deployability in resource-constrained environments.

Through experiments realized on both some standard images and a video; the performance metrics of structural similarity index (SSIM), peak signal-to-noise ratio (PSNR) and runtime/FPS were calculated. The results show that the proposed approach achieves competitive SSIM values compared to representative methods, while also delivering notable gains in computational efficiency.

2 Material and method

2.1 *Total variation denoising (TVD)*

TVD is a variational method reducing noise while preserving edges by penalizing the image total variation. The denoising problem is formulated as the following optimization problem given in Equation (1) that balances quality to the observed data and the smoothness of the solution:

$$u = \arg\min_{x} \left\{ \frac{1}{2} |y - x|_2^2 + \lambda |Dx|_1 \right\} \tag{1}$$

where $y \in \mathbb{R}H \times W \times C$ is the noisy input, x is the denoised image, $\lambda > 0$ is the regularization parameter, and D denotes the discrete gradient operator. A typical one-dimensional form of D can be written as in Equation (2).

$$D = \begin{bmatrix} -1 & 1 & 0 & \cdots & 0 \\ 0 & -1 & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & \cdots & -1 & 1 \end{bmatrix} \in \mathbb{R}^{(N-1)N} \tag{2}$$

In Equation (2), N represents the total number of pixels in the (vectorized) image. In two dimensions, the gradient operator is typically applied in both directions (horizontal and vertical), following the classical TVD denoising formulations and their extensions [1-6].

To solve the TVD optimization problem efficiently, a primal-dual iterative scheme can be employed, as in widely used TVD minimization algorithms [2, 4, 5]. In this work, the iterative procedure of the proposed TVD scheme is summarized below as Algorithm 1.

2.2 *High-pass filter (HPF)*

The HPF is used to emphasize the high-frequency components of an image while attenuating low-frequency components, as in standard digital image processing practices [7, 8, 11]. It is particularly effective for enhancing edges and fine details that may be blurred or suppressed by smoothing operations. In this study, the HPF is constructed by subtracting a Gaussian low-pass filtered image from the original image.

First, a Gaussian kernel is defined by Equation (3) for low-pass filtering [7, 8]:

$$G(x, y) = \frac{1}{2\pi\sigma^2} exp\left(-\frac{x^2 + y^2}{2\sigma^2}\right) \tag{3}$$

Algorithm 1: TVD Scheme

Input: Noisy image $\mathbf{y}$, parameters λ, number of iterations N, step size
Output: Denoised image $\mathbf{u}$
Initialize primal variable $\mathbf{u} \leftarrow \mathbf{y}$
Initialize dual variables $\mathbf{p}_x \leftarrow 0$, $\mathbf{p}_y \leftarrow 0$
for $i = 1$ **to** N **do**
  Compute gradients
  $\mathbf{u}_x \leftarrow \text{shiftRight}(\mathbf{u}) - \mathbf{u}$
  $\mathbf{u}_y \leftarrow \text{shiftDown}(\mathbf{u}) - \mathbf{u}$
  Update dual variables
  $\mathbf{p}_x \leftarrow \mathbf{p}_x + \frac{\tau}{\lambda}\mathbf{u}_x$
  $\mathbf{p}_y \leftarrow \mathbf{p}_y + \frac{\tau}{\lambda}\mathbf{u}_y$
  Normalize dual variables
  $\text{norm} \leftarrow \sqrt{\mathbf{p}_x^2 + \mathbf{p}_y^2}$
  $\text{norm} \leftarrow \max(1, \text{norm})$
  $\mathbf{p}_x \leftarrow \frac{\mathbf{p}_x}{\text{norm}}$
  $\mathbf{p}_y \leftarrow \frac{\mathbf{p}_y}{\text{norm}}$
  Compute divergence
  $\mathbf{d} \leftarrow (\mathbf{p}_x - \text{shiftLeft}(\mathbf{p}_x)) + (\mathbf{p}_y - \text{shiftUp}(\mathbf{p}_y))$
  Update primal variable
  $\mathbf{u} \leftarrow \mathbf{y} - \lambda\mathbf{d}$
end for
Return u

In Equation (3), σ is the standard deviation of the Gaussian distribution. The kernel is normalized that the sum of all its elements equals one, ensuring that the overall brightness of the image is preserved after filtering.

The low-pass filtered image is obtained by convolving the input image I with the Gaussian kernel, using standard 2D convolution as in Equation (4) [7, 8, 11]:

$$\text{LowPass}(x,y) = \sum_{i=-\infty}^{\infty}\sum_{j=-\infty}^{\infty} I(x+i, y+j)\, G(i,j) \tag{4}$$

This operation smooths high-frequency details and retains mainly the low-frequency content. The high-pass component is then computed as given Equation (5):

$$\text{HighPass}(x,y) = I(x,y) - \text{LowPass}(x,y) \tag{5}$$

This subtraction highlights edges and fine textures that correspond to high-frequency information. The resulting high-pass image is subsequently normalized to the range [0, 255] for visualization and further processing. In the proposed framework, this high-pass component is later combined with the TVD-based denoised image. The corresponding HPF procedure is given in Algorithm 2 as pseudo code.

Algorithm 2: High-Pass Filtering Procedure

Input: Image **src**, kernel size k
Output: High-pass filtered image **highPass**
Convert image to floating-point format: **srcFloat** $\leftarrow$ toFloat(**src**)
Create Gaussian kernel **G** of size $k \times k$
Compute low-pass image:
lowPass $\leftarrow$ **srcFloat** $*$ **G**
Compute high-pass image:
highPass $\leftarrow$ **srcFloat** $-$ **lowPass**
Normalize **highPass** to $[0, 255]$
Return highPass

2.3 Proposed TVD

In the proposed implementation of TVD, we adopt a parallelizable primal-dual scheme to improve computational efficiency. The method iteratively updates the primal and dual variables using simple shift-based operations, which are suitable for vectorized and parallel hardware. The main stages of the algorithm are summarized below, following the spirit of existing primal-dual and diffusion-based TVD schemes [2-6, 14, 15, 17].

1. Initialization: The primal variable u is initialized as a copy of the input image y. This variable represents the denoised image and is updated at each iteration. The dual variables p_x and p_y, which capture the horizontal and vertical gradients, are initialized to zero.

2. Gradient computation: The gradients of u in horizontal and vertical directions are computed by shift operations with border replication as given in Equation (6) and (7).

$$u_x = u_{shift\,right} - u \tag{6}$$
$$u_y = u_{shift\,down} - u \tag{7}$$

3. Dual variable update: The dual variables are updated using the computed gradients as in Equation (8) and (9):

$$p_x = p_x + \frac{\tau}{\lambda}u_x \tag{8}$$
$$p_y = p_y + \frac{\tau}{\lambda}u_y \tag{9}$$

In here, τ is the step size and λ is the regularization parameter. These updates refine the representation of edge information in the dual domain.

4. Projection: To keep the dual variables within the unit ball, they are normalized as in Equation (10), (11), (12) and (13).

$$\text{norm} = \sqrt{p_x^2 + p_y^2} \tag{10}$$
$$\text{norm} = \max(1, \text{norm}) \tag{11}$$
$$p_x = p_x / \text{norm} \tag{12}$$
$$p_y = p_y / \text{norm} \tag{13}$$

This projection step prevents the dual variables from growing unbounded and helps preserve edges without excessive smoothing.

5. Divergence computation: The divergence of the dual variables is computed as in Equation (14), where $p_{x,shift\,left}$ and $p_{y,shift\,up}$ denote the dual variables shifted left and upward, respectively. The divergence term enforces spatial consistency in the denoised image.

$$d = (p_x - p_{x,shift\,left}) + (p_y - p_{y,shift\,up}) \tag{14}$$

6. Primal variable update: Finally, the primal variable is updated using the divergence $u = y - \lambda d$. This update balances the fidelity to the noisy observation y and the regularization imposed by the total variation term. By expressing all operations in terms of simple shifts, additions, and pointwise functions, the proposed TVD implementation can be efficiently parallelized, leading to significant reductions in computation time while maintaining strong edge-preserving denoising performance.

2.4 Proposed HPF and overall scheme

In the proposed framework, the high-pass filtering stage is designed to be both computationally efficient and easily combined with the TVD-based denoising described in the previous subsections, similar in spirit to existing sharpening and enhancement schemes that combine denoising with high-frequency boosting [16-21]. The implementation consists of the following steps.

1. Gaussian kernel construction: A separable one-dimensional Gaussian kernel is first generated as in Equation (15) and (16). Here, where the standard deviation σ is chosen as $\frac{(\text{kernelsize}-1)}{2}$. The 2D kernel is obtained as the outer product of the 1D kernels along each axis.

$$\mathrm{GaussKernel}(x) = \exp\left(-\frac{x^2}{2\sigma^2}\right) \tag{15}$$

$$\mathrm{Kernel2D}(i,j)=\mathrm{GaussKernel}(i),\,\mathrm{GaussKernel}(j) \tag{16}$$

2. Low-pass filtering: The denoised image is converted to floating-point precision and convolved with the Gaussian kernel by Equation (17), where $*$ denotes 2D convolution. This step suppresses high-frequency noise and yields a smoothed version of the denoised image.

$$\mathrm{LowPass} = \mathrm{DenoisedFloat} * \mathrm{Kernel2D} \tag{17}$$

3. High-pass component: The high-pass image is obtained by subtracting the low-pass image from the denoised image by using Equation (18).

$$\mathrm{HighPass} = \mathrm{DenoisedFloat} - \mathrm{LowPass} \tag{18}$$

4. Normalization and combination: The high-pass component is normalized to the range [0, 255] and combined with the denoised image using a weighted sum given in Equation (19), which is analogous to high-boost and unsharp masking strategies used in previous works [7, 10, 19-21]. The parameters α and β control the relative contribution of the denoised image and the high-pass component, respectively. This formulation does not represent a conventional weighted average but follows a high-boost filtering strategy. In this context, β acts as a gain factor that amplifies high-frequency details, and therefore the condition $\alpha + \beta = 1$ is not required.

In our experiments, α was fixed to 1.0 to fully preserve the structural information obtained from the TVD stage. The parameter β was empirically selected as 0.6 after testing different values. A brief sensitivity analysis was conducted for β in the range [0.3, 1.0]. Values below 0.4 led to weak sharpening, whereas values above 0.8 produced noticeable over-enhancement artifacts. So, it was observed that smaller β values resulted in insufficient sharpening, while larger values introduced excessive detail amplification and visual artifacts. The selected value β=0.6 provided the most visually balanced result. The chosen value provided the best balance between detail enhancement and structural consistency. By adjusting these weights, the trade-off between noise suppression and detail enhancement can be controlled.

The TV regularization parameter λ controls the trade-off between noise suppression and edge preservation. Smaller λ values reduce smoothing strength, whereas larger values may lead to excessive attenuation of structural details. In this study, λ was selected within a moderate range to preserve dominant edges while suppressing blur-induced artifacts.

Similarly, the Gaussian kernel width σ was chosen to simulate mild camera-induced blur without removing essential structural components. Very small σ values provide negligible enhancement, whereas excessively large σ values may introduce artificial edge amplification. The selected parameters were determined through empirical evaluation over a limited range to ensure stable structural similarity and visually balanced enhancement.

$$\mathrm{Result} = \alpha\ \mathrm{DenoisedFloat} + \beta\ \mathrm{HighPassNorm} \tag{19}$$

The overall proposed method, which combines TVD-based denoising with the high-pass sharpening stage, is summarized in the following pseudo code as Algorithm 3.

Algorithm 3: Proposed TVD–HPF Enhancement Framework

```
Input: Original image src, TVD parameters (λ, τ, N),
       Gaussian kernel size k, weights α, β
Output: Sharpened image sharpened
Step 1: Total Variation Denoising
denoised ← totalVariationDenoising(src, λ, N, τ)
Step 2: High-Pass Filtering
Convert denoised to float: DenoisedFloat ← toFloat(denoised)
Construct Gaussian kernel Kernel2D of size k × k
lowPass ← conv2D(DenoisedFloat, Kernel2D)
highPass ← DenoisedFloat − lowPass
HighPassNormalized ← normalize(highPass)
Step 3: Combination
sharpened ← α · DenoisedFloat + β · HighPassNormalized
Return sharpened
```

2.5 Image degradation model

Camera-induced blur arising from flicker, zoom variations or minor handshake could be effectively modeled mathematically, by convolving the latent sharp image with the PSF [22]. For mild degradations, such PSFs are commonly approximated by isotropic Gaussian kernels, which provide a simple yet realistic representation of high-frequency attenuation in practical imaging conditions [23].

To evaluate the proposed TVD-HPF enhancement framework under controlled conditions, the observed input image is generated by applying a blur–based noise model that reflects common camera degradations. In real scenarios, image quality is frequently affected by flicker, zoom fluctuations, minor handshake and autofocus instabilities, all of which suppress high-frequency details and can be approximated by a low-pass Gaussian response.

Accordingly, the degraded observation y is modeled as Equation (20) and where x denotes the latent sharp image and $*$ indicates 2D convolution with reflective boundary conditions.

$$y = x * G_\sigma \tag{20}$$

The blur kernel $G\sigma$ is an isotropic Gaussian defined by Equation (21) with a standard deviation of $\sigma = 2$. The kernel size is selected as $k = 6\sigma + 1 = 13$, which captures more than 99% of the Gaussian energy [23].

$$G_\sigma(i,j) = \frac{1}{2\pi\sigma^2}\exp\left(-\frac{i^2+j^2}{2\sigma^2}\right) \tag{21}$$

In practice, this degradation is implemented using the OpenCV operator of GaussianBlur$(x, y, 13 \times 13, \sigma = 2)$ with given parameters, and a widely adopted method for generating Gaussian-smoothed images in applied computer vision pipelines [24].

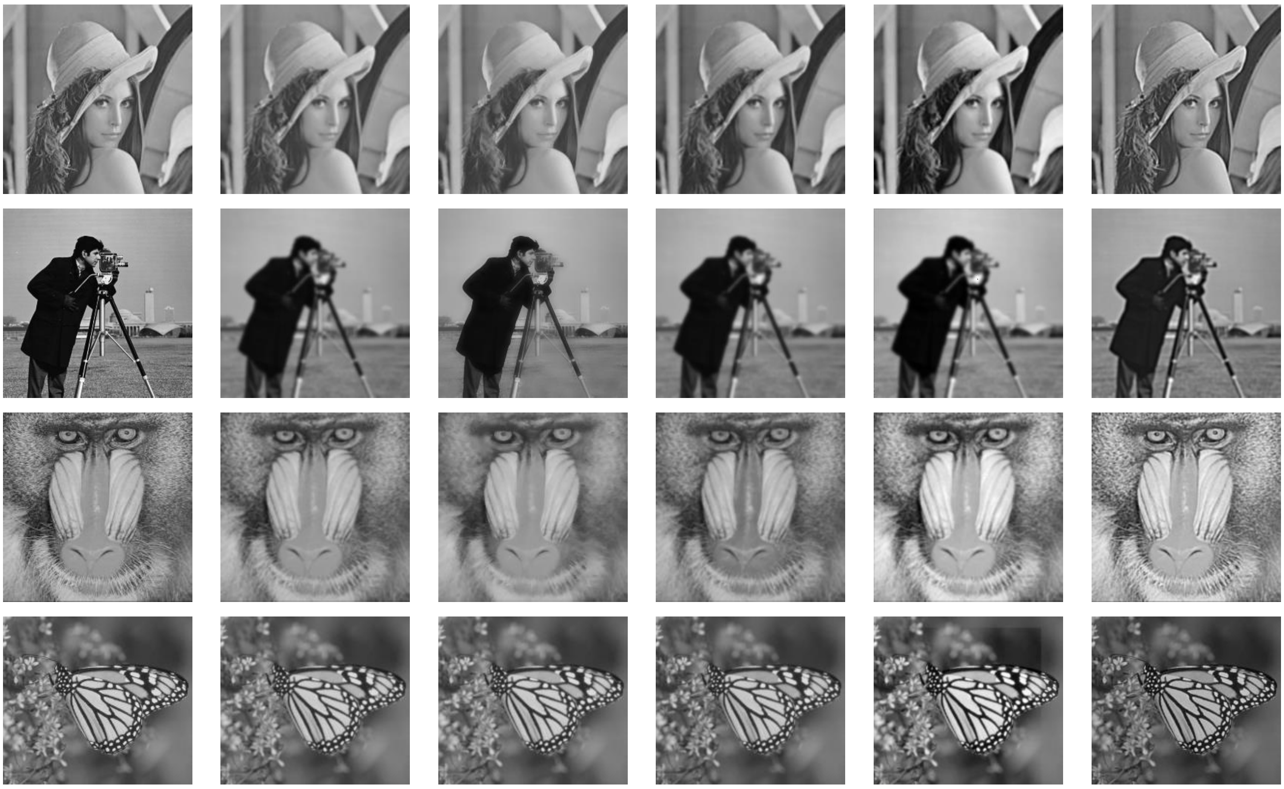
By modeling camera-induced blur as a form of noise through this Gaussian PSF approximation, the evaluation setup provides a realistic and consistent framework for assessing the proposed method's ability to restore suppressed high-frequency details.

3 Results and discussion

This section evaluates the performance of the proposed denoising-sharpening framework on four benchmark images (Lena, Cameraman, Baboon, Monarch) and a 640×512 video sequence from [25]. All test images were synthetically degraded using the Gaussian blur model described in Section 2.5, with $\sigma = 2$ and reflective boundary conditions. This degradation simulates mild camera-induced blur and high-frequency attenuation rather than additive Gaussian noise. The comparison includes three established algorithms [16, 20, 21]. The evaluation focuses on three metrics: SSIM, PSNR and computational performance measured in runtime except for the video experiment, which is presented in FPS. The experimental studies were conducted on the platform of 11th-generation Intel Core i7-11800H processor. The software was developed in C++ and compiled with the OpenCV 4.x library. The code is entirely CPU-based, all computations were performed using OpenMP parallelization.

The results of the proposed method and compared algorithms are presented in Figure 1 by using benchmark images used in literature, for visual comparison. The quantitative results with SSIM index, PSNR and runtime (in seconds) for these images are given in Table 1.

The video experiment was conducted on a publicly available 640×512 grayscale surveillance-style sequence consisting of 300 frames in [25]. The same Gaussian blur model (σ = 2) was applied frame-wise to simulate realistic camera degradation. The visual and quantitative results obtained from methods on the 150th frame segment of the video are given in Figure 2. SSIM, PSNR values and the real

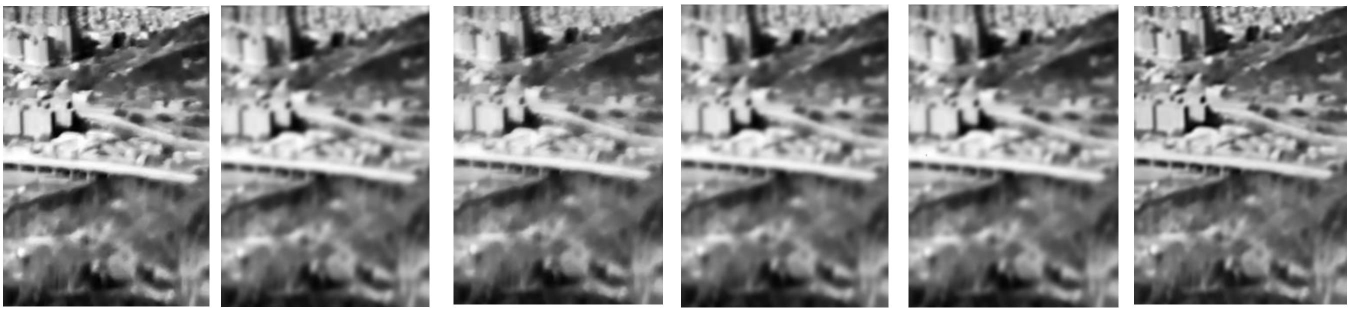


Original | Blurred | [16] | [20] | [21] | Proposed

Figure 1. Visual comparison of different methods and proposed for some images: from top to bottom Lena, Cameraman, Baboon and Monarch images

Table 1. The performance metrics for test images

Images	Lena			Cameraman			Baboon			Monarch		
Methods	**SSIM**	**PSNR**	**Runtime (s)**	**SSIM**	**PSNR**	**Runtime (s)**	**SSIM**	**PSNR**	**Runtime (s)**	**SSIM**	**PSNR**	**Runtime (s)**
[16]	0.922	**31.395**	0.168	0.807	**24.205**	0.050	0.603	20.966	0.151	**0.971**	**30.804**	0.306
[20]	0.811	28.277	0.070	0.717	22.601	0.021	0.433	**21.028**	0.070	0.909	27.727	0.118
[21]	**0.927**	22.06	0.275	0.688	21.098	0.275	0.501	18.13	0.261	0.907	25.418	0.549
Proposed	0.905	29.205	**0.015**	**0.901**	23.436	**0.008**	**0.746**	19.067	**0.018**	0.918	22.209	**0.033**



Original | Blurred | [16] SSIM:0.970 PSNR:32.01 | [20] SSIM:0.922 PSNR:29.93 | [21] SSIM:0.957 PSNR:32.68 | Proposed SSIM:0.932 PSNR:28.53

Figure 2. Visual results for the 150th frame segment of the video with quantitative metrics

time performances in FPS of these methods on video are presented in Table 2. Runtime performance was measured under CPU-only execution using OpenMP parallelization.

Table 2. Video performances on 640×512 sequence

Method	SSIM	PSNR	FPS
[16]	**0.804**	**25.11**	5.97
[20]	0.658	23.59	22.67
[21]	0.745	18.67	5.19
Proposed	0.750	23.33	**60.24**

According to the Table 1, the proposed method yields the highest SSIM values in cameraman and baboon images, and while it lags behind some other methods for Lena and Monarch, it has the potential to compete with them. Cameraman has numerous sharp edges (edge-dense), while Baboon has very dense and complex textures (furs), both exhibiting abrupt changes. Lena, on the other hand, has a smooth surface and soft transitions, while Monarch has regular patterns. Therefore, the HPF included in the proposed method showed less significant effects on Lane and Monarch. In light of these evaluations, it can be said that the proposed method is more effective in images with rapid changes and irregular structures. This indicates that the method is quite successful in edge and texture preservation.

In contrast, it is observed that the proposed method does not always yield the best results in terms of PSNR values, and some methods in the literature ([16] and [21]) achieve higher PSNR values, especially in images with lower frequency content and smoother structures such as Lena and Monarch. This suggests that the proposed method focuses on preserving perceptual quality rather than minimizing pixel-based error. This is a common occurrence in the literature, indicating that preserving high-frequency components can negatively impact pixel-based error metrics. The PSNR results support the claim that the proposed approach improves perceptual sharpness while maintaining structural consistency. Table 1 shows that the proposed method exhibits remarkable performance, particularly in terms of SSIM and runtime. The method has the lowest runtime across all images, offering a significant advantage for real-time applications.

Figure 2 shows the results obtained for a frame from video. As can be seen, methods [16] and [21] give the best performance in terms of both SSIM and PSNR. Although the proposed method gives an acceptable result in terms of SSIM, its PSNR value remains at an average level. This shows that the proposed method preserves the overall structure of the image but lags behind some other methods in terms of pixel accuracy in this example. In short, although the proposed method is advantageous in terms of speed and perceptual quality, its performance in the blur removal problem for the image in this example is average.

Table 2 compares the performance of the methods on a 640×512 resolution video. According to the results, method [16] was the most successful method in terms of quality, achieving the highest SSIM (0.804) and PSNR (25.11) values. However, the low FPS value of this method (5.97) is a limiting factor for real-time applications. Although the proposed method lags slightly behind [16] in terms of SSIM (0.750) and PSNR (23.33), it is by far the fastest method with a value of 60.24 FPS. This result shows that the proposed method is quite suitable for real-time video processing applications. In addition, the fact that the SSIM value is very close to that of method [21] and higher than that of method [20] reveals that it offers a balanced performance in terms of visual quality. Overall, the proposed method is the approach that provides the best balance between quality and speed, and stands out as a strong alternative that can be preferred, especially in real-time video applications.

Although some of compared methods produce slightly higher SSIM and PSNR values in isolated cases, their computational cost is significantly higher because of their adaptive and clustering-based formulations. The proposed method relies exclusively on shift-based gradient operations, explicit divergence computation, and separable Gaussian convolution, all implemented using parallelizable CPU routines. This structural simplicity contributes significantly to its computational efficiency. It should also be noted that certain adaptive sharpening approaches may enhance local contrast while slightly reducing global structural similarity, which explains occasional SSIM decreases. In contrast, the proposed TVD-regularized formulation preserves structural consistency before applying high-frequency enhancement.

Unlike many contemporary approaches that rely on GPU acceleration or pre-implemented library functions, the proposed framework is entirely developed from first principles using manually implemented mathematical operations and parallelized CPU routines. This design choice ensures full control over computational complexity and makes the method suitable for constrained hardware environments.

In the experimets, a simple parameter sensitivity evaluation was conducted by varying λ, σ, and β within reasonable ranges around the selected values. The results indicate that the proposed framework maintains stable SSIM performance and runtime behavior under moderate parameter variations. Significant deviations from the selected parameters primarily affect perceptual sharpness rather than structural consistency.

For a fair comparison of the performances, it is important to evaluate methods not only in terms of quality metrics but also under similar computational conditions. Method comparisons can be done in two different ways: reporting the actual performance obtained by running all methods under similar hardware and optimization conditions, and comparing the methods in terms of quality metrics by normalizing them to the same runtime/FPS level. To perform evaluations at the same (equal) runtime/FPS, the parameters of each method can be adjusted to reach a specific target value. In this case, the SSIM and PSNR values obtained while all methods are running at the same speed are compared. However, this approach is not always applicable because some methods are inherently parameter-independent or significantly reduce quality. Therefore, the common approach in the literature is to run the methods under the same hardware and default settings and report both quality (SSIM, PSNR) and speed (runtime/FPS) metrics together. This approach is also adopted in this study, and all methods were run in the same hardware environment and with default settings. Especially in video processing applications, the FPS value is directly related to computational complexity. On the other hand, the computational approach of the methods can also be considered as one of the features that distinguish the proposed method from others, and the computational approach of the proposed method has been explained above. The results showed that the proposed method exhibits a balanced performance not only in terms of speed but also in terms of quality.

In light of these assessments, it can be said that the proposed method achieves good image enhancement performance while providing a significant computational speed advantage for real-time image and video applications.

4 Conclusion

This study presented an efficient image enhancement framework that integrates TVD with HPF to jointly address noise suppression and detail restoration. The proposed method leverages the edge-preserving characteristics of TVD while recovering high-frequency components through a lightweight high-pass sharpening stage. This combination provides a balanced solution to image enhancement: reducing noise without compromising structural fidelity. At the same time, the proposed shift-based gradient operations and separable Gaussian convolution reduce algorithmic complexity, resulting in an important runtime improvement.

The experimental evaluations conducted on multiple benchmark images and a real-time video sequence demonstrate that the proposed approach consistently yields higher structural similarity scores while maintaining sharper details, confirming the effectiveness of coupling TVD with high-pass enhancement. In addition, the computational performance of the proposed approach significantly outperforms existing techniques. The method achieves real-time capability, exceeding 60 FPS in video processing, which highlights its suitability for time-critical applications.

The results validate that the integration of variational denoising with frequency-based sharpening can deliver high-quality reconstructions with minimal computational cost. This efficiency, combined with stable performance across diverse image types, underscores the practical utility of the proposed method in domains such as medical imaging, surveillance, and embedded vision systems [26-28].

It is important to emphasize that the proposed framework was intentionally designed under CPU-only constraints, where GPU utilization and external high-level library dependencies were not permitted. All mathematical operators, including gradient computation, divergence, convolution, and projection steps, were implemented explicitly rather than relying on pre-built optimization toolboxes. This design philosophy prioritizes transparency, reproducibility, and deterministic computational behavior, making the method particularly suitable for embedded systems, real-time surveillance, and resource-constrained platforms.

In conclusion, while the proposed method demonstrates strong performance in terms of high speed (runtime/FPS) and perceptual quality, it lags behind in PSNR in some cases. Therefore, the method can be considered a suitable solution, especially for applications where visual quality is paramount and real-time processing is required.

Future research may focus on extending the framework to adaptively control the sharpening strength, exploring multi-scale formulations, or incorporating learning-based priors to further enhance robustness. Nonetheless, the presented method offers a strong and computationally efficient foundation for tasks requiring both noise reduction and detail preservation.

CRediT authorship contribution statement

Kubilay Küçük: Conceptualization, Data curation, Formal analysis, Investigation, Methodology, Resources, Software, Validation, Visualization, Writing–original draft, Writing–review and editing; **Derya Yılmaz:** Conceptualization, Formal analysis, Methodology, Supervision, Validation, Visualization, Writing–review and editing.

Acknowledgement

The authors would like to thank TÜBİTAK BİLGEM İLTAREN and Halil İbrahim Cüce for their support and guidance in using the hardware.

Conflict of interest

The authors declare that there is no conflict of interest.

Similarity (iThenticate): 8%

Declaration of generative artificial intelligence (AI) and AI-assisted technologies in the writing process

The authors did not use any artificial intelligence tools in the preparation of this work.

References

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

[2] A. Chambolle, "An algorithm for total variation minimization and applications," *Journal of Mathematical Imaging and Vision*, 20 (1), 89-97, 2004. https://doi.org/10.1023/B:JMIV.0000011325.36760.1e

[3] T. F. Chan, S. Esedoglu, and F. E. Park, "Image decomposition combining staircase reduction and texture extraction," J*ournal of Visual Communication and Image Representation*, 18 (6), 464-486, 2007. https://doi.org/10.1016/j.jvcir.2006.12.004

[4] X. Bresson and T. F. Chan, "Fast dual minimization of the vectorial total variation norm and applications to color image processing," *Inverse Problems and Imaging*, 2 (4), 455-484, 2008. https://doi.org/10.3934/ipi.2008.2.455

[5] A. Beck and M. Teboulle, "Fast gradient-based algorithms for constrained total variation image denoising and deblurring problems," *IEEE Transactions on Image Processing*, 18 (11), 2419-2434, 2009. https://doi.org/10.1109/TIP.2009.2028250

[6] I. W. Selesnick, H. L. Graber, D. S. Pfeil, and R. L. Barbour, "Simultaneous low-pass filtering and total variation denoising," *IEEE Transactions on Signal Processing*, 62 (5), 1109-1124, 2014. https://doi.org/10.1109/TSP.2014.2298836

[7] R. C. Gonzalez and R. E. Woods, *Digital Image Processing*, 2nd ed. Prentice Hall, 2002.

[8] A. K. Jain, *Fundamentals of Digital Image Processing*. Prentice Hall, 1989.

[9] S. Mallat, *A Wavelet Tour of Signal Processing*, 2nd ed. Academic Press, 1999. https://doi.org/10.1016/B978-0-12-466606-1.X5000-4

[10] X. Chen, J. Yang, Q. Wu, J. Zhao and X. He, "Directional high-pass filter for blurry image analysis," *Signal Processing: Image Communication*, 27 (7), 760-771, 2012. https://doi.org/10.1016/j.image.2012.05.003

[11] J. R. Parker, *Algorithms for Image Processing and Computer Vision*. Wiley, 1997.

[12] A. Buades, B. Coll, and J.-M. Morel, "A non-local algorithm for image denoising," Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), pp. 60-65, 2, San Diego, CA, USA, 2005. https://doi.org/10.1109/CVPR.2005.38

[13] C. Tomasi and R. Manduchi, "Bilateral filtering for gray and color images," Proceedings of the Sixth International Conference on Computer Vision (IEEE Cat. No.98CH36271), pp. 839-846, Bombay, India, 1998. https://doi.org/10.1109/ICCV.1998.710815

[14] H. Khodabakhshi Rafsanjani, M. H. Sedaaghi and S. Saryazdi, "Efficient diffusion coefficient for image denoising," *Computers & Mathematics with Applications*, 72 (4), 893-903, 2016. https://doi.org/10.1016/j.camwa.2016.06.005

[15] A. Anchal, S. Budhiraja, B. Goyal, A. Dogra and S. Agrawal, "An efficient image denoising scheme for higher noise levels using spatial domain filters," *Biomedical and Pharmacology Journal*, 11 (2), 2018. https://dx.doi.org/10.13005/bpj/1415

[16] C. C. Pham and J. W. Jeon, "Efficient image sharpening and denoising using adaptive guided image filtering," *IET Image Processing*, 9 (1), 71-79, 2015. https://doi.org/10.1049/iet-ipr.2013.0563

[17] S. Bettahar and A. Boudghene Stambouli, "Shock filter coupled to curvature diffusion for image denoising and sharpening," *Image and Vision Computing*, 26 (11), 1481-1489, 2008. https://doi.org/10.1016/j.imavis.2008.02.010

[18] S. A. Priyanka and Y. -K. Wang, "Fully symmetric convolutional network for effective image denoising," *Applied Sciences*, 9 (4), 778, 1-19, 2019. https://doi.org/10.3390/app9040778

[19] J. Liu, Y. Lou, G. Ni and T. Zeng, "An image sharpening operator combined with framelet for image deblurring," *Inverse Problems*, 36 (4), 045015, 1-29, 2020. https://doi.org/10.1088/1361-6420/ab6df0

[20] T. Horiuchi, K. Watanabe and S. Tominaga, "Adaptive filtering for color image sharpening and denoising," Proceedings of the 14th International Conference of Image Analysis and Processing - Workshops (ICIAPW 2007), pp. 196-201, Modena, Italy, 2007. https://doi.org/10.1109/ICIAPW.2007.11

[21] K. K. Vin Toh and N. A. Mat Isa, "Locally adaptive bilateral clustering for image deblurring and sharpness enhancement," *IEEE Transactions on Consumer Electronics*, 57 (3), 1227-1235, 2011. https://doi.org/10.1109/TCE.2011.6018878

[22] R. Fergus, B. Singh, A. Hertzmann, S. T. Roweis and W. T. Freeman, "Removing camera shake from a single photograph," *ACM Transactions on Graphics*, 25 (3), 787-794, 2006. https://doi.org/10.1145/1141911.1141956

[23] T. Eboli, J. M. Morel, and G. Facciolo, "Breaking down polyblur: Fast blind correction of small anisotropic blurs," *Image Processing On Line*, 12 (14), 435-456, 2022. https://doi.org/10.5201/ipol.2022.405

[24] S. C. Kumain, M. Singh, N. Singh and K. Kumar, "An efficient Gaussian noise reduction technique for noisy images using optimized filter approach," Proceedings of the 2018 First International Conference on Secure Cyber Computing and Communication (ICSCCC), pp. 243-248, Jalandhar, India, 2018. https://doi.org/10.1109/ICSCCC.2018.8703305

[25] [Full Ver.] 640x512 15um InSb MWIR infrared Video [Online]. Available: https://www.youtube.com/watch?v=MBxXjqgdSdo, [Accessed 10 July 2025].

[26] İ. Kayadibi, G. E. Güraksın and U. Ergün, "ESA temelli göz durumu tespitinde görüntü önişlem yöntemlerinin etkisi," *Niğde Ömer Halisdemir Üniversitesi Mühendislik Bilimleri Dergisi*, 11 (3), 496-505, 2022. https://doi.org/10.28948/ngumuh.1086414

[27] Z. Balkaya, C. Özgültekin, S. Serttaş and Ç. Bakır, "Real-time traffic accident detection system on hybrid data with YOLOv9 and YOLOv11 architectures," *Niğde Ömer Halisdemir Üniversitesi Mühendislik Bilimleri Dergisi*, 14 (4), 1542-1558, 2025. https://doi.org/10.28948/ngumuh.1715199

[28] İ. H. Doğan, O. Arslan, A. B. Tat, B. Şahin, E. E. Uslu, İ. Yülüce and O. Dağdeviren, "Deep learning classification and object detection in helicopter images: Performance analysis of GoogleNet, AlexNet and YOLOv9c architectures," *Niğde Ömer Halisdemir Üniversitesi Mühendislik Bilimleri Dergisi*, 14 (1), 340-349, 2025. https://doi.org/10.28948/ngumuh. 1556995

BY NC