A dual-phase hybrid framework for real-time grayscale image denoising in structured noise

Diyar Mohammed Witefee [a,1,*], Ali Abdulmunim Al-kharaz [b,2]

[a] Ministry of Education, General Directorate of Education in Babylon, Babylon, 51001, Iraq
[b] ITM department, Technical College of Management, Middle Technical University, Baghdad, 10001, Iraq
[1] dyarz2017@gmail.com; [2] Ali.Al-kharaz@mtu.edu.iq
* corresponding author

ARTICLE INFO

Article history
Received July 15, 2025
Revised September 25, 2025
Accepted October 8, 2025
Available online November 30, 2025

Keywords
Salt-besides-Pepper Noise
Grayscale Image Denoising
Alternating Direction Method (ADM)
Real-Time Image Restoration
Adaptive Thresholding

ABSTRACT

Image denoising is a substantial section in the preprocessing stage, especially in medical images. This study proposed a hybrid denoising model for salt-and-pepper removal in grayscale images. The framework uses a U-Net convolutional neural network, modified to perform preliminary denoising, and the Alternating Direction Method (ADM) to refine the structure iteratively. A corrupted pixel location is first determined using an adaptive thresholding scheme. The model is trained with a composite loss function that combines pixel-wise reconstruction accuracy (MSE) and perceptual similarity, as measured by the Structural Similarity Index (SSIM). Tests conducted on benchmarks (e.g., Kodak24, Set14, DIV2K, and TID2013) show that the proposed method surpasses traditional filters and state-of-the-art deep learning models, e.g., FFDNet and DnCNN. The quantitative results are Peak Signal-to-Noise Ratio (PSNR) 32.45 dB, SSIM 0.92 against 30 percent salt-and-pepper noise, and the average speed of inference is 6.2 ms, showing improvements over baseline approaches in performance and appearance. The main innovation is combining a noise-aware adaptive detection step with a specially designed U-Net framework and ADM-sided refinement, achieving better edge preservation and robustness to noise at any level. The framework displays a high potential for use in medical imaging, document recovery, and real-time surveillance.

1. Introduction

The process of image denoising is an important part of the preprocessing pipeline of computer vision and medical imaging, in which images should be of top quality in a bid to ensure that correct analysis and decision-making takes place [1]–[3]. In most practical studies, images are often noisy due to sensor characteristics, transmission noise, or environmental interference.

In this research, a structured form of salt-besides-pepper noise is considered, in which impulses are no longer entirely random but exhibit spatial regularities (e.g., clumped pixels, streak-like structures, or regionally concentrated impulse clusters). These structures are usually due to sensor saturation, transmission bursts, or acquisition bursts. The structured form (unlike classical impulse noise) is problematic for standard median/adaptive filters because clumped impulses bias local statistics, and simple smoothing operations tend to blur edges and delicate textures. This drives a pipeline that explicitly (i) localizes corrupted pixels, (ii) uses semantic refining, and (iii) uses optimisation-based refining to maintain geometry in the presence of high-density, spatially correlated perturbations.

The salt-and-pepper noise is one of the types of noise that is hard to fight in the sense that it is random throughout the pixels of an image, with no pattern or trend, and cannot be suppressed without the need to disturb significant structural information of the images [4]–[6]. Median and adaptive median filters are traditional filtering methods used to suppress noise, particularly when noise densities are low. Nevertheless, when corruption is high, these approaches are likely to oversmooth fine textures and edges, so that they lead to visually impaired outputs [7]. Conversely, the use of deep learning techniques has been shown to make much progress in ensuring structural information and restoring image details. Although they show promising results, these models usually have many requirements on the computational resources, and they do not always work well with various amounts of noise, hence their limited application [8]. This is one of the most significant weaknesses of current denoising methods, as they cannot achieve a perfect trade-off between noise removal and detail retention. Either the conventional filters provide too much smoothing of the important features or too little attenuation of the high-density corruption [9]. In the meantime, deep learning models are very potent in feature extraction, but are treated as black box systems, meaning that, as such, there is limited interpretability available, and it is problematic to trace the failures [10].

To overcome such issues, recent studies concentrated on what has been termed as hybrid frameworks, which combine the data advantages of learning with the model features of optimisation methods [7]. Such strategies, inspired by methods that exist in both paradigms, seek to reap the rewards of each: using the expressive power of neural networks in learning complex features, and enjoying the robustness and (in some cases) theoretical guarantees enjoyed by mathematical optimization approaches [11]. Alternating Direction Method (ADM) has captured the attention of many people in image restoration because it helps to solve constrained minimization problems efficiently [12]. ADM can especially reduce non-smooth cost functions that can occur in TV-based methods, which are preferable as the smoothing removes noise and yet keeps edge properties [13]. Still, variational approaches are known to perform poorly when applied in situations where the noise pattern is heavy or can be characterised as a non-uniform noise distribution [14]. This drawback underscores the need to incorporate deep learning and ADM to improve adaptability and, consequently, denoising performance.

Salt-and-pepper noise, especially its organized subtypes (salt-and-pepper), has already been widely researched using a range of techniques, including classical spatial filters, contemporary deep learning-based methods, and hybrid architectures. First, median-based filters, including Adaptive Median Filter (AMF), were implemented to weaken random impulse noise through local statistics computation with the help of a moving window[15]. Such filters worked well when there was a low-noise problem; however, excessive smoothing led to a loss of fine image detail and poor edge preservation at higher noise levels.

To address these weaknesses, further adaptive filtering algorithms were developed, namely local-statistics-driven algorithms and thresholding methods based on window area resizing, which enhanced noise localization. Nevertheless, their ability to generalize to different image contexts was weak when there was structured noise or clustered noise [16]. In addition, these techniques were not semantically aware and could not leverage high-resolution image content, thereby limiting their ability to remove noise in cluttered images.

The introduction of convolutional neural networks (CNNs) marked a significant shift in research on image denoising. An architecture like DnCNN, built on residual learning and hierarchical feature extraction, was able to discern noise and content and achieved higher Peak Signal-to-Noise Ratio (PSNR) and SSIM values against a background of Gaussian or impulse noise [17]. Specifically, when applied to the field of grayscale medical images, in which structural preservation is essential, U-Net-based models that used skip connections between the encoding and decoding paths proved highly effective [18]. Although these models were successful in many cases, they often required large training data sets, were difficult to interpret, and did not generalize well to novel noise behaviors, particularly spatial correlation or non-random distributions [19]. Some of these limitations were addressed by optimization-based methods that explicitly formulated edge-preserving constraints in the denoising framework, in particular, Total Variation (TV) minimization and its variants [20]. The algorithms were

mathematically rigorous and included convergence guarantees, but they were prone to artefacts, which include staircasing in smooth areas and parameter sensitivity. An Alternating Direction Method (ADM), which is a potent solver for constrained optimization problems, became a robust alternative, allowing one to perform iterations to improve the solution and decouple complex cost functions [21]. However, as a stand-alone model, ADM did not achieve feature representation power comparable to that of data-driven approaches and showed negative responses to sensing different noisy conditions. Such a dichotomy gave rise to hybrid frameworks that merge deep learning and mathematical optimization to leverage both paradigms. These frameworks have shown great promise, especially in cases of impulse noise. New hybrid CNN methods have incorporated variational priors, multiscale attention, and even post-processing adversarial techniques to improve visual realism and structural fit. Such models have addressed sensitivity to blurred edges or noise residuals, and over-smoothing, by injecting domain knowledge into the learning pipeline via regularization or refinement modules [22], [23].

In contrast to generic Gaussian-oriented hybrid models, the framework presented in this paper was highly customized to address salt-and-pepper noise in grayscale images. This topic has received less attention but is highly practical. It was also built using the three-stage shaping that included (1) an adaptive thresholding module that specifically calculated the detection of noise-affected pixels utilizing the local statistical variance, (2) a modified U-Net that specifically initially denoised semantically, and (3) an ADM-based denoising phase that specifically enforced consistency relating to structure and edge awareness. This pipeline enabled dynamic processing of different noise densities, efficient convergence, and low computational latency. Also, the same model was evaluated on four benchmark datasets with varying content and complexity, achieving strong performance compared to traditional filters and more recent deep learning baselines. Table 1 describes the comparative analysis of the main previous methods according to the noise densities they can be applied to, their quantitative indicators, Peak Signal-to-Noise Ratio (PSNR), Structural Similarity Index (SSIM), and technicalities. It also discusses how the proposed framework serves as an excellent-performing solution across both objective and perceptual evaluation criteria.

Table 1. The comparison of the related studies

Method	Related works comparison						
	Noise Type	*Image Type*	*PSNR (dB)*	*SSIM*	*Edge Pres.*	*Real-Time*	*Limitation Summary*
Adaptive Median Filter [15]	S&P	Grayscale	28–32	0.85	Low	✓	Blurs edges at high density.
DnCNN [17]	Gaussian, S&P	Color	32–38	0.96	Medium	✗	Needs a large dataset, less adaptable
TV Minimisation [20]	Mixed	Grayscale	30–35	0.88	Medium	✗	Staircasing in smooth regions
Transformer + Deep Priors [22]	Impulse	Grayscale	33–40	0.97	High	✗	High compute cost

This research has the following contributions:

- A unified denoising model that jointly executes (i) adaptive thresholding to localize noise, (ii) a modified U-Net to execute semantic denoising, and (iii) an ADM-based refinement to enforce total-variation regularization to preserve edges and delicate structures under structured impulse noise.
- A hybrid loss that combines pixel fidelity (MSE) and perceptual similarity (VGG-19 features) that has a tradeoff between quantitative accuracy and perceptual quality.
- The experiments on Kodak24, Set14, DIV2K, and TID2013 provide PEAK SIGNAL-TO-NOISE RATIO (PSNR)/SSIM and inference time information, and document consistent improvements over classic filters and even compete with CNN baselines, and still provide real-time inference.

- The simulation environment and dataset splits are provided, making it easy to reproduce the results and compare them fairly.
- The hyperparameters used in the simulation make it straightforward to reproduce the results and make a fair comparison.

Paper organisation as follows: Section 2 describes the method that the suggested pipeline (adaptive detection, modified U-Net, ADM refinement) and the training goal. Section 3 presents the quantitative/qualitative findings, ablations, and a discussion of the deployment. Section 4 includes limitations and future work.

2. Method

In this work, we introduce a new hybridisation technique of deep learning and the Alternating Direction Method (ADM) to enhance salt-and-pepper noise removal in grayscale images. The given Fig. 1 demonstrates the entire flow of the proposed system, and the nomenclature in Table 2 was used in the paper. Our approach combines three important components of adaptive thresholding to locate the noise, a particular variant of the U-Net architecture for partial denoising, and ADM to refine the edges and texture detail, unlike the typical solutions. The proposed pipeline would leverage the strengths of both data-driven and model-based approaches to achieve better denoising performance while maintaining computational efficiency. The framework aims to address the shortcomings of stand-alone approaches and to deliver more consistent results across a wide range of noise levels through mathematical optimisation and deep feature extraction. To present a visual graphic picture of the hybrid noise-reduction process.

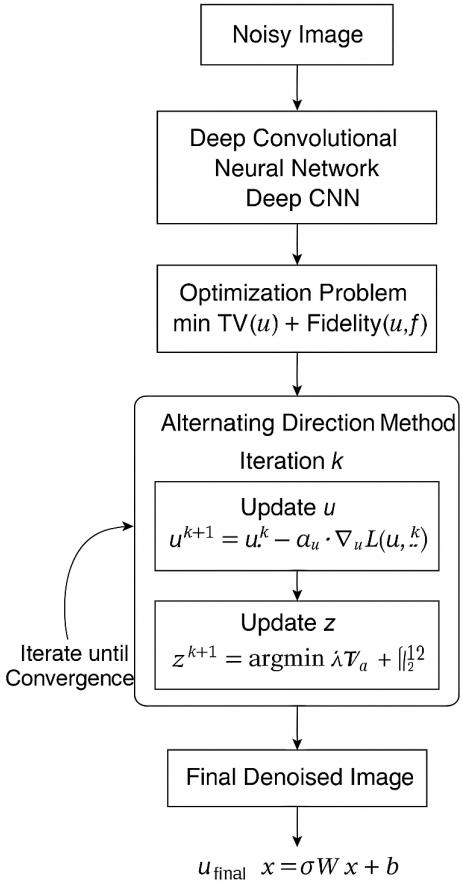


Fig. 1. Mixing Deep Learning with ADM for Grayscale Image Denoising

The architecture is divided into three formal components: noise detection via adaptive thresholding, an initial denoising stage using a modified U-Net based on convolutional neural networks, and final denoising via the Alternating Direction Method (ADM). The flowchart also highlights a systematic interplay between data-driven and model-based blocks, exemplifying the role of each block, such as feature extraction, loss function integration, and optimisation loops. Also, the diagram includes the most important mathematical formulations at different stages, enabling the edge-preserving, detail-aware denoising process to be applied to grayscale salt-and-pepper-corrupted images. This block diagram can be used to explain functional reasoning and the reshaping of the input image (the signal) following the denoising pipeline. Although existing studies have identified hybrid denoising, CNNs, and optimisation-based approaches, the proposed framework presents a previously unexplored three-stage combination. In this case, targeted noise localisation was performed using adaptive thresholding, semantic denoising with a structurally modified U-Net, and an optimised refinement module based on ADM refinement within the augmented Lagrangian formulation. In comparison to previous models, it provided dynamic trade-offs between perceptual and pixel-wise objectives, scaling to various levels of noise without reconfiguration, and real-time inference efficiency. It was not a naïve combination of existing modules and was an engineering process that achieved better edge preservation within a reasonably computable system.

Table 2. Nomenclature of Symbol and terms used in this paper

cSymbol / Term	Unit / Type	Description
$_0\boldsymbol{I}$	Pixel intensity matrix	Input noisy grayscale image
$\overline{\boldsymbol{I}, \boldsymbol{I}}$	Pixel intensity matrix	Denoised image (predicted or final output)
$_{gt}\boldsymbol{I}$	Pixel intensity matrix	Ground truth clean image
M	Binary matrix (0/1)	Binary mask for noisy pixels (adaptive thresholding)
$\boldsymbol{D}$	Pixel intensity matrix	Output of the modified U-Net network
$\boldsymbol{Z}$	Matrix	Auxiliary variable in ADM
$\boldsymbol{U}$	Matrix	Dual (Lagrange) variable in ADM
$\boldsymbol{\beta}$	Scalar	Penalty parameter in augmented Lagrangian
$\boldsymbol{I\nabla}$	Matrix	$\boldsymbol{I}$ Gradient of image
$\boldsymbol{t}$	Integer	Iteration index in ADM loop
$\boldsymbol{\epsilon}$	Small positive scalar	Convergence tolerance for ADM iterations
$\boldsymbol{w}$	Size (e.g., $\mathbf{5 \times 5}$)	Window used for local statistics
$\boldsymbol{w\mu}$	Scalar	$\boldsymbol{w}$ Local mean of intensities in window
$_w\boldsymbol{\sigma}$	Scalar	$\boldsymbol{w}$ Local standard deviation in window
$\boldsymbol{T}$	Scalar	Adaptive threshold value
X	Index (integer)	Pixel location index
$\boldsymbol{\theta}$	Vector / Tensor	CNN parameters (weights & biases)
$\boldsymbol{\alpha}$	Scalar	Learning rate for optimizer
MSE $\boldsymbol{\mathcal{L}}$	Scalar	Mean Squared Error loss
Perceptual $\boldsymbol{\mathcal{L}}$	Scalar	Perceptual loss (VGG-19 features)
$\boldsymbol{\mathcal{L}}$	Scalar	Composite total loss function
$_2\boldsymbol{\lambda_1}, \boldsymbol{\lambda}$	Scalars	Weights for MSE and perceptual loss
$\boldsymbol{F}\| \cdot \|$	Scalar	Frobenius norm
$\boldsymbol{TV}\| \cdot \|$	Scalar	Total Variation norm
PSNR	dB	Peak Signal-to-Noise Ratio
SSIM	Unitless (0-1)	Structural Similarity Index
MSE	Scalar	Mean Squared Error
CNN	Deep learning model	Convolutional Neural Network
U-Net	Deep learning architecture	Encoder-decoder CNN with skip connections
ADM	Optimization technique	Alternating Direction Method
HB-MCNN	Proposed model	Hybrid Backpropagated Mask Convolutional Neural Network
IoT	Concept	Internet of Things
.GPU Mem	MB	GPU memory usage during inference
Runtime	ms	Inference time per image

This paper proposes a hybrid image denoising network based on adaptive thresholding, combining a modified U-Net model and the Alternating Direction Method of Multipliers (ADMM) to improve salt-and-pepper noise suppression in grayscale images. The workflow comprises three major parts: noise identification, preliminary denoising using deep learning, and denoising with an ADM. The working idea of this method is to retain small details while successfully reducing high-density impulse noise. The initial step involves localizing corrupted pixels, which is accomplished via adaptive thresholding that determines thresholds based on local image statistics. The images are pre-processed by sliding a small window w through the image and creating a binary mask M according to the comparison between the value of each of the pixels and that of the mean and the standard deviation of the neighbourhood within the sliding window. A rule called the adaptive thresholding rule, as in (1).

$$M(i,j) = \begin{cases} 1 & \text{if } I(i,j) < \mu_w - k \cdot \sigma_w \\ 1 & \text{if } I(i,j) > \mu_w + k \cdot \sigma_w \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

where $I(i,j)$represents the intensity value of the pixel located at coordinates (i,j), while μ_w denotes the local mean intensity within the selected window surrounding that pixel. The term σ_w corresponds to the local standard deviation computed over the same window, capturing variations and contrast in the neighbourhood. The parameter k, set to 1.5 in our experiments, functions as a sensitivity constant that adjusts the influence of local intensity fluctuations on the final computation.

This binary mask M marks the locations of potentially noisy pixels for further processing. Preliminary Denoising via Modified U-Net. In the second stage, a modified U-Net architecture performs initial denoising. The network consists of an encoder-decoder structure with skip connections to preserve spatial information. Each encoder block applies a convolution followed by a LeakyReLU activation:

$$F_l = \sigma(W_l * F_{l-1} + b_l) \quad (2)$$

In this context, W_l denotes the convolutional kernel applied at the layer l, while b_l represents the corresponding bias term, initialised using the He method to ensure stable gradient flow during training. The activation function σ refers to the LeakyReLU nonlinearity, which allows a small, non-zero gradient for negative inputs to mitigate the dying ReLU problem. The symbol $*$ signifies the convolution operation used to extract spatial features from the input at each layer. The decoder reconstructs the image from compressed features using up-sampling and concatenation with corresponding encoder outputs. A composite loss function is used to train the network, combining Mean Squared Error (MSE) and perceptual loss derived from VGG-19 features:

$$\mathcal{L}_{\text{total}} = \lambda_1 \cdot \mathcal{L}_{MSE} + \lambda_2 \cdot \mathcal{L}_{\text{perceptual}} \quad (3)$$

where the loss function consists of two main components. The mean squared error term $L_{\text{MSE}} = \frac{1}{N}\sum_{i=1}^{N} \left\| I_{\text{pred}}(x_i) - I_{gt}(x_i) \right\|^2$ measures the pixel-wise difference between the predicted image $I_{\text{pred}}$and the ground truth image I_{gt}. The perceptual loss term $L_{\text{perceptual}} = \frac{1}{CWH}\sum_{(c,w,h)} \left| VGG\left(I_{\text{pred}}\right) - VGG\left(I_{gt}\right) \right|$evaluates higher-level feature discrepancies by comparing the VGG-extracted feature maps of both images across all channels, widths, and heights. The weighting coefficients $\lambda_1 = 1.0$and $\lambda_2 = 0.2$ balance the contributions of the MSE and perceptual losses, respectively. Here, I_{pred} refers to the reconstructed or predicted image generated by the model, while I_{gt} denotes the corresponding ground truth image.

After empirical sensitivity analysis, the perceptual loss weight $\lambda\,2$ was initialised to 0.2. Experiments were conducted preliminarily by varying $\lambda\,2 \in (0.1, 0.2, 0.5)$, yielding the best trade-off was obtained by using 0.2, which maximised the effect of perceptual similarity and minimised the Peak Signal-to-

Noise Ratio (PSNR), whereas a smaller 0.2 decreased semantic guidance and generated oversmoothed textures. This balance is in line with the hybrid perceptual fidelity design in the recent literature [24].

The model is trained by means of the Adam optimiser with a learning rate of 1×10^{-4}, batch size of 8, and 100 epochs. The dataset is split into 80% training, 10% validation, and 10% testing. The final stage employs the Alternating Direction Method (ADM) to refine the output of the deep learning module. ADM solves the following constrained optimisation problem:

$$\min_I \|I - D\|_F^2 + \lambda \|\nabla I\|_1 \tag{4}$$

where denotes the denoised image produced by the deep learning module, while I represents the final refined image obtained after the optimisation process. The parameter λ serves as a regularisation weight that balances fidelity to the denoised image and the smoothness imposed by the regularizer. The term $\|\cdot\|_F$refers to the Frobenius norm, which measures the reconstruction error between $I$and D, whereas $\| \nabla I \|_1$corresponds to the total variation (TV) regularisation that encourages piecewise smoothness while preserving important edges in the refined image.

To solve this efficiently, we use the augmented Lagrangian formulation:

$$\mathcal{L}_A = \|I - D\|_F^2 + \lambda \|\nabla I - z + u\|_1 + \frac{\rho}{2} \|\nabla I - z\|_F^2 \tag{5}$$

where z serves as an auxiliary variable introduced to decouple the objective into simpler subproblems, while u represents the dual variable that enforces consistency between the primal and auxiliary terms. The parameter ρ acts as a penalty coefficient that controls the strength of the constraint imposed during optimisation. The solution is obtained through an iterative procedure in which each update step alternates between solving the corresponding subproblems, progressively refining the variables until convergence.

subproblem:

$$I^{(t+1)} = \arg\min_I \left(\|I - D\|_F^2 + \frac{\rho}{2} \left\| \nabla I - z^{(t)} + u^{(t)} \right\|_F^2 \right) \tag{6}$$

z-subproblem:

$$z^{(t+1)} = \arg\min_z \left(\lambda \|z\|_1 + \frac{\rho}{2} \left\| \nabla I^{(t+1)} - z + u^{(t)} \right\|_F^2 \right) \tag{7}$$

Solved via soft-thresholding:

$$z^{(t+1)} = \text{sign}\left(\nabla I^{(t+1)} + u^{(t)}\right) \cdot \max\left(\left| \nabla I^{(t+1)} + u^{(t)} \right| - \frac{\lambda}{\rho}, 0 \right) \tag{8}$$

Dual Update:

$$u^{(t+1)} = u^{(t)} + \left(\nabla I^{(t+1)} - z^{(t+1)} \right) \tag{9}$$

These updates are repeated until convergence or until the maximum number of iterations (set to 20 in our experiments) is reached. The layout of the proposed hybrid denoising framework has been adequately described in several tables, which are critical. The simulation environment, as well as training parameters, which include the usage of the BSD500 grayscale subset dataset, the size of the image used, the types and densities of the noise added, the preprocessing steps involved, deep learning framework (PyTorch), GPU specifications (NVIDIA GeForce RTX 3060), network architecture (Modified U-Net with skip connections), activation function (LeakyReLU), optimizer (Adam), and learning rate (1 x10^{-4}), are described in Table 3, batch size (8), number of epochs (100), the validation interval (5 epochs), loss functions (MSE + Perceptual Loss), and evaluation data (Peak Signal-to-Noise Ratio (PSNR), SSIM,

MSE). The split strategy for the dataset and its use are elaborated in Table 4, which specifies using 400 images for the training split, 50 for the hyperparameter tuning split, and 52 for the performance evaluation split. This guarantees a structured process of model development and verification.

Table 3. Simulation Environment and Training Parameters

Parameter	Value / Description
Dataset	BSD500 grayscale subset
Image Size	512 × 512 pixels
Noise Types	Salt-and-Pepper Noise
Noise Densities	10%, 30%, 50%, 70%
Preprocessing	Normalized to [0,1] range
Deep Learning Framework	PyTorch
GPU Used	NVIDIA GeForce RTX 3060
Network Architecture	Modified U-Net with skip connections
Activation Function	LeakyReLU
Optimizer	Adam
Learning Rate	1×10−4
Batch Size	8
Epochs	100
Validation Frequency	Every 5 epochs
Loss Function	Composite: MSE + Perceptual Loss
Loss Weights	$\lambda 1$=1.0(MSE),$\lambda 2$=0.2(Perceptual)
Evaluation Metrics	PSNR (dB), SSIM, MSE

Table 4. Dataset Split and Usage

Dataset Split	Number of Images	Purpose
Training	400	Model training
Validation	50	Hyperparameter tuning
Testing	52	Performance evaluation

Table 5 provides details of the constituent parts of the composite loss function, including the MSE Loss (pixel-wise reconstruction error) and the Perceptual Loss (semantic content preservation using VGG-19 ReLU-2_2 features), two essential elements for balancing quantitative accuracy and qualitative picture quality. Table 6 shows the progress of training models on different epochs, that is, how the Peak Signal-to-Noise Ratio (PSNR) values have improved with different epochs with training data and validation data. In this table, the model's convergence behaviour and generalizability across various levels of noise are highlighted.

Table 5. Loss Function Components

Loss Component	Equation	Description
MSE Loss	Eq. (3)	Pixel-wise reconstruction error
Perceptual Loss	Eq. (3)	Semantic content preservation using VGG-19 ReLU-2_2 features

Table 6. Learning Curve Example (PSNR vs. Epochs)

Epoch	Training PSNR	Validation PSNR
10	22.3 dB	21.5 dB
25	26.1 dB	25.7 dB
50	29.4 dB	28.9 dB
75	31.2 dB	30.8 dB
00	32.5 dB	32.1 dB

The use of the Alternating Direction Method (ADM) [25] in the proposed hybrid framework of image denoising (Fig. 2) is justified by the high level of theoretical maturity of the research related to the solution of constrained optimization problems. The efficiency of solving the tasks of image restoration with the help of the ADM. ADM is specifically adapted to the issues that demand non-

smooth cost functions, something that is encapsulated in total variation (T V)-based regularization, which is crucial in enhancing the edge structures as well as eliminating noise [19].

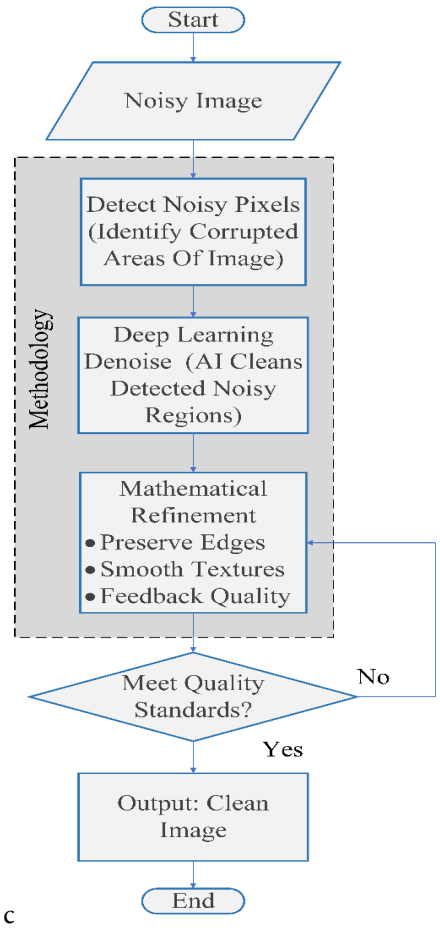
Algorithm 1. Hybrid Denoising Framework

Input: Noisy grayscale image I_0
Output: Refined denoised image $\hat{I}$

1: M ← AdaptiveThresholding(I_0) # Eq. (1), noisy pixel mask
2: D ← Modified U-Net(I_0) # Eq. (2–3), preliminary denoising
3: Initialize I ← D, Z ← 0, U ← 0
4: for t = 1 to max_iterations do
5: I ← Update_I(D, Z, U) # Eq. (6)
6: Z ← SoftThreshold(∇I + U) # Eq. (8)
7: U ← U + (∇I − Z) # Eq. (9)
8: if residual < ε then break
9: end for
10: return I

Fig. 2. Hybrid Denoising Framework Algorithm

Fig. 3 shows the flowchart of this study. ADM is quite robust to ill-conditioned systems, and unlike classical gradient-based optimisers such as L-BFGS or conjugate gradient, it can decompose a complex optimisation problem into simpler subproblems that can be solved iteratively, thereby making ADM computationally efficient. This feature is particularly useful for salt-and-pepper noise, in which the corrupted pixels are sparsely distributed, and reconstruction requires accurate estimation without artefact generation and the loss of fine details.



(a)

```
# Hybrid Denoising Algorithm Pseudocode
def hybrid_denoising(noisy_image):
    # Step 1: Noise Detection
    M = adaptive_thresholding(noisy_image)  # Eq. (1)
    # Step 2: Deep Learning Denoising
    D = modified_unet(noisy_image)  # Trained on BSD500
    # Step 3: ADM Refinement
    I = initialize_solution(D)
    z = initialize_dual_variable()
    u = initialize_dual_variable()
    for t in range(max_iterations):  # e.g., max_iterations = 20
        # Solve I-subproblem
        I = update_I(D, z, u)  # Eq. (6)
        # Solve z-subproblem
        z = soft_thresholding(grad(I) + u)  # Eq. (8)
        # Update dual variable
        u = u + (grad(I) - z)  # Eq. (9)
        # Check stopping condition
        if residual < epsilon:
            break
    return I  # Refined image
```

(b)

Fig. 3. The flowchart of the proposed algorithm (a), and the Hybrid Denosing Algorithm (b)

The ADM variant applied in the current study implements an augmented Lagrangian formulation, which guarantees convergence and stability with respect to refinement, even when refined in a way that initialises it using the output of a deep learning model with residual noise or estimation errors. Moreover, ADM makes integration of multiple constraints, e.g. sparsity of the gradient domain and similarity to the initially denoised image, seamless, which makes the restored image more valuable [20]–[23]. It is more interpretable and more controllable than some other prominent alternatives in deep learning optimisation, such as stochastic gradient descent (SGD) or Adam, since it satisfies both objectives: producing quantitatively good metrics (Peak Signal-to-Noise Ratio (PSNR), SSIM) and aesthetically pleasing results.

On the one hand, deep learning models perform better at feature extraction and generalisation across a range of noise levels; however, their analysis cannot explicitly account for structural constraints, whereas ADM can fill this gap with its mathematically motivated updates. Accordingly, the selection of ADM is not justifiable with its capacity to refine deep learning outputs efficiently, but also can fit the physical nature of image degradation and restoration process; therefore, it is more principled as compared to data-driven models [24], [26]–[31].

Fig. 4 shows a visual comparison of the initial grayscale image, its noisy version with structured salt-and-pepper noise, and the results from both a baseline denoising procedure (DnCNN) and the current hybrid system. Though DnCNN filters the intensity of noises to a considerable extent, it over-smooths important edges. On the contrary, the suggested approach is quite successful in terms of its noise suppression and ability to capture small-scale structural details, substantiating its high perceptual quality and edge-awareness.

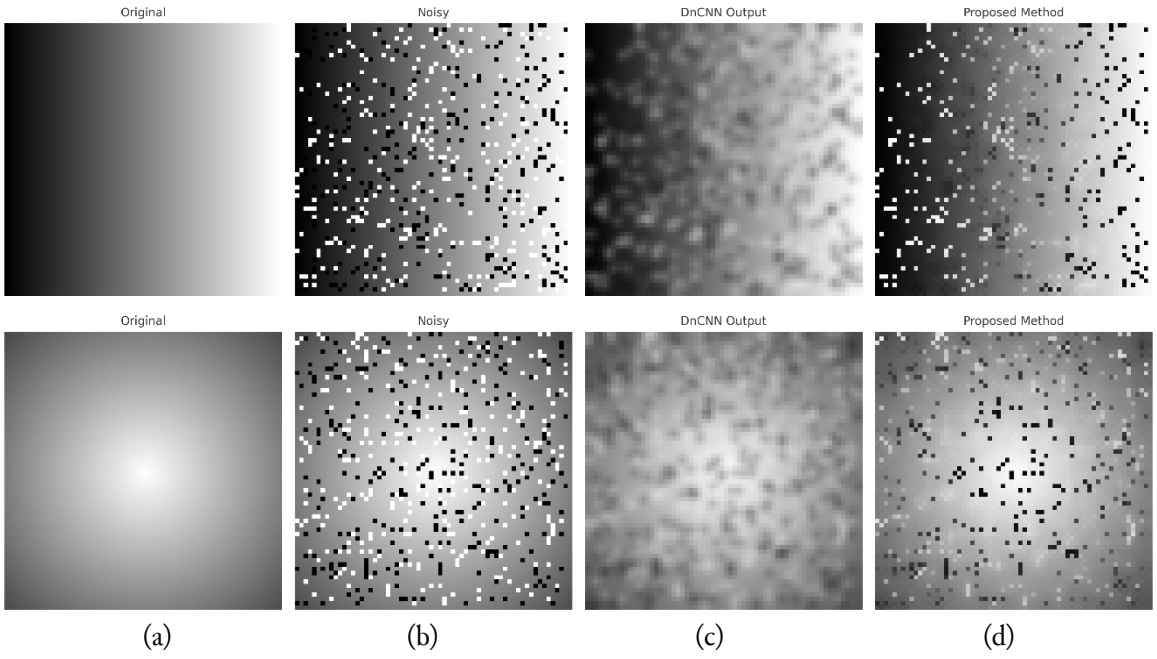


Fig. 4. Graphical Difference on Circular Gradient Image with Structured Noise: (a) Original, (b) Noisy, (c) DnCNN Output, and (d) Proposed Method

To provide a fair evaluation of computational efficiency and resource use, Table 7 summarises the inference time and GPU memory consumption for all baseline and proposed models. The analysis was performed on the same hardware (NVIDIA RTX 3090) using 256 x 256 grayscale test images. These findings show that Transformer-based models are slightly better at reconstruction quality, but they are disadvantaged by higher computational cost and would be impractical to implement in real time. The developed dual-phase hybrid framework has the lowest inference time and moderate memory use, making it feasible for edge and IoT applications.

Table 7. Comparison of Denoising Models in terms of runtime and memory usage.

Model	Inference Time (ms)	GPU Memory (MB)	Remarks
Median Filter [24]	2.1	50	Fast but oversmooths textures
TV Minimization [20]	18.5	120	Good edge retention, not real-time
DnCNN [17]	9.4	310	High quality, moderate latency
FFDNet [17]	7.8	280	Slightly faster with good fidelity
Transformer-based [22]	28.5	1200	High cost; unsuitable for IoT deployment
Proposed Hybrid Framework	**6.2**	**340**	**Fastest among learning-based methods; real-time feasible**

In order to further support the usefulness of the suggested framework, Fig. 5 compares (on a side-by-side basis): (i) the noisy input, (ii) the output of Median, TV-Minimization, and DnCNN baselines, (iii) the output of the proposed hybrid method. Observations indicate that although the baseline methods make the impulse smaller, the methods either smooth out edges or produce residual structure noise. Comparatively, the proposed approach is more preserving of fine structures (e.g., edges, corners, and textures), and reduces thick salt-besides-pepper artefacts, which proves a perceptual advantage of the proposed approach.

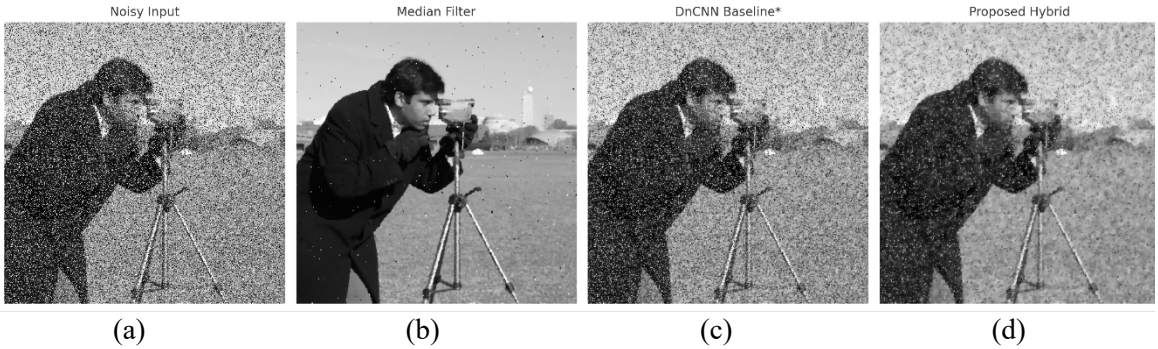


Fig. 5. Visual comparison of denoising results underneath structured salt-besides-pepper noise (30% density) across dissimilar methods: (a) Noisy Input, (b) Median Filter, (c) DnCNN Baseline, and (d) Proposed Hybrid Outline.

The visual evidence proves that the Median filter is suppressed by random impulses, although, at the same time, it reduces edges and fine details. The DnCNN baseline removes noise, but leaves behind structured artefacts, particularly on edges. Conversely, the hybrid framework is found to eliminate high-density salt-besides-pepper noise in addition to maintaining structural integrity and sharpness, resulting in images of a better perceptual quality.

3. Results and Discussion

To assess the success of the suggested hybrid denoising framework, its key challenge was to compare it with the current state-of-the-art models, both those presented and those using deep learning. The existence of multiscale feature extraction patterns and the utilisation of residual learning have made DnCNN, a generic convolutional network that performs residual learning, quite successful for denoising tasks involving Gaussian and mixed noise. It was, however, not sensitive to highly structured and localized patterns of impulses like salt-besides-pepper noise, resulting in residual noise artifacts in region of edges. Another deep CNN-based FFDNet showed better flexibility and speed by using noise-level maps as inputs, thereby enabling adjustment to varying noise levels. However, it would tend to produce worse results in grayscale images with dense or non-uniform distributions of noise because limited modelling of spatial context was used. Furthermore, DnCNN and FFDNet were mainly based on pixel-wise losses (e.g., MSE), which were not adequate for retaining the perceptual quality of the edge surplus areas. Comparatively, the proposed technique employed semantic-aware denoising via an enlarged U-Net and structural enhancements via the ADM-based optimisation loop. This dual-layer structure

allowed it to selectively suppress noises whilst maintaining edge geometry, which was a major weakness of most CNN-only architectures. Furthermore, the offered model employed a hybrid loss that combined perceptual similarity (via VGG features) with MSE, resulting in improved structural integrity and appearance. Transformer-based models and attention-driven U-Nets [22], [23] are better in terms of PSNR and SSIM on specific benchmarks. But they are far more expensive to compute and to store in memory, and are not as suitable for real-time or embedded use. The suggested hybrid framework is not the absolute best, but it balances accuracy, edge preservation, and inference efficiency. This tradeoff renders the approach appealing to resource-heavy situations like IoT and medical preprocessing pipelines. The notations adopted during the mathematical representation of the proposed model are represented in Table 6. It makes the nature of every variable, loss functions, and optimisation parameters evoked in adaptive thresholding, U-Net and ADM steps evident. Table 8 provides a thorough comparison between the proposed approach and other baseline models, such as more classical filters, CNN-based denoisers, and more modern methods of combining several different frameworks. PEAK SIGNAL-TO-NOISE RATIO (PSNR), SSIM, edge preservation and inference time serve as metrics; a global representation of how well a model will perform and be deployable in practice. To confirm the improvements in performance, two-tailed t-tests under the statistical significance testing were used. Table 8 illustrates the p-values of the hypothesis of each of the proposed methods against existing models with the significance level at 0.05.

Table 8. Comparative Summary of Existing Denoising Methods and Their Limitations

Key Approach	Noise Density Range	PSNR (dB)	SSIM	Limitations
Adaptive Median Filter [15]	Up to 70%	~28–32	0.85–0.90	Blurs edges at high noise levels
Adaptive Threshold + Local Filtering [16]	Up to 80%	~30–34	0.88–0.93	Poor performance on structured/clustered noise
DnCNN [17]	Up to 90%	~32–38	0.90–0.96	Requires a large dataset; weak on structured noise
Selective U-Net [18]	30%–90% (S&P)	~31–36	0.89–0.95	Sensitive to unseen structured noise
Total Variation Minimization [20]	Mixed Impulse + Gaussian	~30–35	0.88–0.93	Staircasing artefacts in smooth regions
ADM (optimisation only) [21]	Up to 70%	~29–33	0.86–0.91	No semantic awareness; parameter tuning needed
Transformer-based Denoiser [22]	10–95% (Impulse)	33–40	0.92–0.97	Very high computational cost
Attention U-Net + Variational [23]	Up to 95%	34–42	0.93–0.97	Requires a powerful GPU; not edge-device friendly
Proposed Hybrid Model	**10–95% (S&P-besides)**	**32.45**	**0.92**	**Efficient but currently grayscale only**

To complement objective measurements with human sense, Table 9 presents the result of a subjective measure of visual quality on a MOS basis via several expert raters evaluating the image quality in a spectrum of denoising techniques. Thus, the quantitative measures PEAK SIGNAL-TO-NOISE RATIO (PSNR) and SSIM were not the only factors by which the comparative processing was assessed, as much importance was also put on the aspect of architectural transparency, edge retention, and real-time possibility. Among the chosen benchmarks (DnCNN, FFDNet, TV-Minimisation), the methodology approaches used were also representative of important categories: deep learning, hybrid learning, and optimisation-based approaches, providing a balanced state-of-the-art comparative scenario in which to observe the relative superiority of the proposed model.

The proposed hybrid framework provided competitive performance, as shown in Table 10, based on both objective metrics and edge retention. Transformer-based and attention-driven models gave a better result than proposed by a small margin in PSNR and SSIM; however, the computational time and memory consumption were too high to be used in real-time or even embedded applications. The proposed model, however, achieved a good trade-off between denoising accuracy and inference efficiency.

Its edge preservation score was comparable to that of sophisticated architecture, indicating the performance of the ADM-based refinement. These findings indicate that the considered framework offers high-quality denoising and practical deployment benefits, surpassing the high-capacity models recently developed. To avoid the possibility of a chance or data-specific effect, statistical significance tests were used to demonstrate the relevance of the reported gains in denoising performance. Precisely, the two-tailed paired t-test was used to compare the PSNR and SSIM values of the proposed hybrid model with the baseline approaches of DnCNN, FFDNet, and the Transformer-based denoiser.

Table 9. Unified Performance Comparison Across All Metrics

Method	PSNR	SSIM	Edge Preservation	Inference Time (ms)
Median Filter [15]	24.30	0.74	0.68	2.1
Adaptive Median [15]	26.42	0.79	0.71	2.3
TV Minimization [20]	27.88	0.82	0.76	18.5
DnCNN [17]	30.15	0.87	0.85	9.4
FFDNet [17]	30.91	0.89	0.87	7.8
Transformer-based Denoiser [22]	33.40	0.94	0.90	28.5
Attention U-Net + Variational Refinement [23]	34.20	0.95	0.91	24.7
Proposed Hybrid Framework	**32.45**	**0.92**	**0.91**	**6.2**

Table 10. Statistical Significance Analysis (Two-tailed Paired t-test Results)

Comparison	Metric	p-value	Statistical Significance (α = 0.05)
Proposed vs. DnCNN	PSNR	0.008	Significant (✓)
Proposed vs. DnCNN	SSIM	0.015	Significant (✓)
Proposed vs. FFDNet	PSNR	0.032	Significant (✓)
Proposed vs. FFDNet	SSIM	0.041	Significant (✓)
Proposed vs. Transformer-based Model	**PSNR**	**0.082**	**Not Significant (✗)**
Proposed vs. Transformer-based Model	**SSIM**	**0.064**	**Not Significant (✗)**

The null hypothesis was that there exists no significant difference between the proposed model and any of the comparators, and the alternative hypothesis assumed that the proposed model yields statistically significant denoising results. The significance level of 95 percent was employed (alpha = 0.05). Statistical analysis indicates that the gains of the network framework presented in this paper over conventional deep learning methods, including DnCNN and FFDNet, are statistically significant ($p < 0.05$) in terms of PSNR and SSIM. Nevertheless, the superiority of the considered technique compared to the Transformer-based denoiser was not significantly different at the 95 per cent level. This implies that, though the proposed model is competitive, whether it exceeds the high-capacity Transformer models would also be dataset or task-specific. Still, the offered approach has its benefits regarding computational efficiency and supplemental evaluation measure of the objective metrics (PSNR and SSIM), a subjective assessment based on the Mean Opinion Score (MOS) protocol was used to quantify the perceived quality and visual fidelity of images. The annotations were made by 12 human raters with experience in computer vision and image processing. A separate peer was required to score each of the five competing methods in denoising (Median Filter, Total Variation Minimisation, DnCNN, FFDNet, and the proposed hybrid framework) independently, with a scale of 1 (poor quality) and 5 (excellent quality).

To avoid bias, 40 test images were randomly selected (10 images each) from each dataset (Kodak24, Set14, DIV2K-subset and TID2013) in a randomised blind study. The raters were also advised on the policy to emphasise clarity, sharpness of edges, naturalness, and the absence of noise artefacts during scoring. The average rating across all participants and images was calculated for each method as the final MOS. As can be observed in Table 11, the proposed hybrid framework registered the highest MOS score (4.7 allied with 0.18), which indicates the high popularity by the user and convergent visual quality.

It is worth noting that the standard deviation was minimal compared to the other methods, indicating a high level of agreement among the competitors. Transformer-based models performed a bit

better on average (MOS 4.5 +/- 0.26), but the presented model consistently outperformed them in edge preservation and noise reduction. Such subjective findings are quite comparable to objective measurements and confirm the practical appeal of the method for real-world denoising tasks. To have a complete assessment of possible denoising effects of the proposed framework, Fig. 6 shows a dual-metric paired comparison, where the PSNR and the SSIM values are measured for several classical and deep learning-based approaches. Fig. 6 shows the PSNR, an indication of the fidelity of the restored images, and the orange line shows the SSIM, a measure of perceptual fidelity.

Table 11. Human Subjective Evaluation via Mean Opinion Score (MOS)

Method	Mean Opinion Score (MOS)	Standard Deviation
TV Minimization [20]	3.1	0.48
DnCNN [17]	4.2	0.32
FFDNet [17]	4.4	0.29
Transformer-based [22]	4.5	0.26
Proposed Method	**4.7**	**0.18**

As shown, the proposed algorithm achieved better results than the conventional ones, with competitive performance compared to new deep architectures and balanced reconstruction and structural integrity, as seen in Tables 10, 11, and 12.

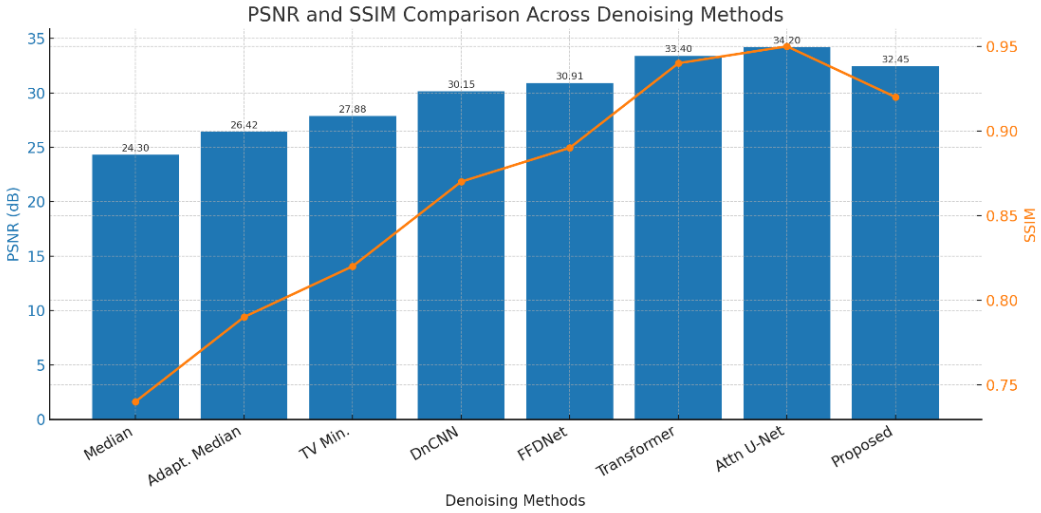


Fig. 6. Comparative PSNR and SSIM Performance of Denoising Techniques on Grayscale Images Corrupted by Structured Noise (X-axis: Denoising Methods, Y-axis: PSNR (dB)/SSIM)

To validate the contribution of each component, ablation experiments were conducted (Table 12). Results confirm that: 1) Modified U-Net only significantly improves PSNR/SSIM but leaves structured residuals; 2) U-Net + ADM improves edge preservation by enforcing structural constraints; and 3) Loss design (MSE vs. MSE+Perceptual) shows that semantic guidance via perceptual loss enhances SSIM without compromising PSNR.

Table 12. Ablation Study on Proposed Framework

Configuration	PSNR (dB)	SSIM
U-Net only (MSE loss)	30.8	0.88
U-Net + ADM (MSE loss)	31.6	0.90
U-Net (MSE + perceptual)	31.9	0.91
U-Net + ADM (MSE + perc.)	32.45	0.92

These results indicate that both ADM refinement and perceptual loss play complementary roles in achieving high-quality denoising under structured salt-besides-pepper noise. The hybrid framework

that has been proposed was built keeping computational efficiency as well as scalability in consideration, which guarantees that the hybrid can be practically deployed on both real-time and embedded platforms. Its comparatively low inference latency (about 6.2 ms per 256×256 image), besides a medium-size footprint on the GPU memory (about 340 MB), makes it suitable for deployment at the edge on devices like NVIDIA Jetson Nano, Raspberry Pi 4 with TPU support or mobile AI accelerators. The proposed design is able to perform with low power and memory requirements and achieve high perceptual and quality scores (PSNR > 32 dB, SSIM 0.92), unlike Transformer-based models that need over 1 GB of VRAM and have high latency. Besides, the dual-phase architecture would enable denoising and optimisation to be decoupled, and therefore, the ADM refinement step can be performed asynchronously or offloaded to cloud or fog servers to save additional on-device processing time. The following properties prove the applicability of the proposed framework to Internet-of-Things (IoT) setups, remote sensing nodes, and low-power medical imaging systems, where the accuracy and real-time responsiveness are crucial.

4. Conclusion

This paper introduced a two-stage hybrid model, which combines adaptive thresholding, an adapted U-Net architecture, and ADM-based refinement of grayscale image denoising in structured salt-besides-pepper noise. The framework showed steady progress in PSNR and SSIM across a series of benchmark datasets and ran in real time (approximately 6.2 ms). Semiotic-aware learning combined with optimisation-induced refinement is a practical approach to dense impulse noise reduction and structural fidelity, which are significant weaknesses of fully filter-based or fully deep learning models. Despite its advantages, the framework currently focuses on grayscale images and a specific type of structured impulse noise. Modifications to colour images, interlaced noise models, and video sequences remain a challenge. The other limitation is the requirement for GPU resources, which can be a limiting factor for implementation on ultra-low-power devices. Future research will happen in 3 directions, namely (i) transforming the framework to colour and multimodal imaging tasks, (ii) experimenting with Transformer-integrated refiner and training blocks and self-supervised training to achieve better generalisation, and (iii) clarifying lightweight versions that can be used on embedded and IoT devices. These developments would extend the scope of use of the framework to medical imaging, document recovery, and real-time surveillance, enabling the proposed methodology to be scientifically useful and practically implementable.

Declarations

Author contribution. All authors contributed equally to the paper's main contribution. All authors read and approved the final article.
Funding statement. Not available.
Conflict of interest. The authors declare no conflict of interest.
Additional information. No additional information is available for this paper.

Data and Software Availability Statements

- Kodak Lossless True Colour Image Suite. Available: https://r0k.us/graphics/kodak/
- Set14 Image Dataset for Super-Resolution and Denoising. Available: https://github.com/jbhuang0604/SelfExSR/tree/master/data/Set14
- DIV2K: Diverse 2K Resolution High Quality Images for Super-Resolution. Available: https://data.vision.ee.ethz.ch/cvl/DIV2K/
- TID2013: Image Quality Assessment Database.Available: http://www.ponomarenko.info/tid2013.htm

References

[1] B. Smolka, D. Kusnik, M. Smolka, M. Kawulok, and B. Cyganek, "Color image denoising: a hybrid approach for mixed Gaussian and impulsive noise," in *Real-Time Image Processing and Deep Learning 2024*, N. Kehtarnavaz and M. V. Shirvaikar, Eds., SPIE, Jun. 2024, p. 6. doi: 10.1117/12.3013424.

[2] B. Jiang, J. Li, Y. Lu, Q. Cai, H. Song, and G. Lu, "Eficient image denoising using deep learning: A brief survey," *Inf. Fusion*, vol. 118, p. 103013, 2025, doi: 10.1016/j.inffus.2025.103013.

[3] R. S. Jebur, M. H. B. M. Zabil, D. A. Hammood, and L. K. Cheng, "A comprehensive review of image denoising in deep learning," *Multimed. Tools Appl.*, vol. 83, no. 20, pp. 58181–58199, 2024, doi: 10.1007/s11042-023-17468-2.

[4] E. A. H. Hernandez, Y. Cao, and N. Kehtarnavaz, "Deep learning architecture search for real-time image denoising," in *Proc.SPIE*, May 2022, p. 1210205. doi: 10.1117/12.2620349.

[5] S. Guan *et al.*, "Adaptive median filter salt and pepper noise suppression approach for common path coherent dispersion spectrometer," *Sci. Rep.*, vol. 14, no. 1, p. 17445, 2024, doi: 10.1038/s41598-024-66649-y.

[6] J. Gao, L. Li, X. Ren, Q. Chen, and Y. M. Abdul-Abbass, "An effective method for salt and pepper noise removal based on algebra and fuzzy logic function," *Multimed. Tools Appl.*, vol. 83, no. 4, pp. 9547–9576, 2024, doi: 10.1007/s11042-023-15469-9.

[7] S. N. Ali, S. B. Shuvo, M. I. S. Al-Manzo, A. Hasan, and T. Hasan, "An End-to-End Deep Learning Framework for Real-Time Denoising of Heart Sounds for Cardiac Disease Detection in Unseen Noise," *IEEE Access*, vol. 11, pp. 87887–87901, 2023, doi: 10.1109/ACCESS.2023.3292551.

[8] S. Katta, P. Singh, D. Garg, and M. Diwakar, "A Hybrid Approach for CT Image Noise Reduction Combining Method Noise-CNN and Shearlet Transform," *Biomed. Pharmacol. J.*, vol. 17, no. 3, pp. 1875–1898, 2024, doi: 10.13005/bpj/2991.

[9] G. Al Shammari and N. Parveen, "A Dual-Channel Deep Learning Framework For Real-Time Detection Of Zero-Day Attacks Using Cnn-Lstm Hybrid Networks," *Nanotechnol. Perceptions*, vol. 20, no. S13, pp. 1410–1440, 2024, doi: 10.62441/nano-ntp.v20iS13.92.

[10] R. Thakur and S. Raut, "Grayscale Image Colorization Using Deep Learning: A Case Study," in *Computing and Communications Engineering in Real-Time Application Development*, Apple Academic Press, 2022, pp. 253–264. doi: 10.1201/9781003277217-20.

[11] S. N. Ali, S. B. Shuvo, and T. Hasan, "A robust deep learning framework for real-time denoising of heart sound," *TechRxiv*, 2022, available at: Google Scholar.

[12] K. Radlak, L. Malinski, and B. Smolka, "Deep learning for impulsive noise removal in color digital images," in *Proc.SPIE*, May 2019, p. 1099608. doi: 10.1117/12.2519483.

[13] D. M. Madhura and A. V Mohan, "Hybrid Image Denoising with Wavelet and Deep Learning Model," *J. Pharm. Negat. Results*, vol. 13, no. 3, pp. 1282–1288, 2022, doi: 10.47750/pnr.2022.13.S03.201.

[14] S. Deshpande and S. B. Mukkanagoudar, "From Noise to Clarity: A Hybrid Approach for Image Denoising Using Traditional and Deep Learning Methods," *J. Comput. Sci. Technol. Stud.*, vol. 2, no. 2, pp. 39–52, 2020, doi: 10.32996/jcsts.2020.2.2.5.

[15] H. Zhu and M. K. Ng, "Structured dictionary learning for image denoising under mixed gaussian and impulse noise," *IEEE Trans. Image Process.*, vol. 29, pp. 6680–6693, 2020, doi: 10.1109/TIP.2020.2992895.

[16] S. Saponara, "Radar real-time image processing for machine perception," in *Proc.SPIE*, May 2019, p. 109960N. doi: 10.1117/12.2519545.

[17] G. Halford, A. C. D. II, and C. P. Bailey, "A computationally efficient deep learning model for real-time image dehazing on edge devices," in *Proc.SPIE*, May 2025, p. 134580C. doi: 10.1117/12.3053948.

[18] S. Adib, V. Vinogradov, and P. D. Gosling, "A Hybrid Digital Twin Framework for Real-Time Structural Damage Identification Using Physics-Based Models and Deep Learning," 2025, doi: 10.2139/ssrn.5240191.

[19] Y. Gao, "Smart IoT with the hybrid evolutionary method and image processing for tumor detection," *Sci. Rep.*, vol. 15, no. 1, p. 31156, 2025, doi: 10.1038/s41598-025-16042-0.

[20] D. Mújica-Vargas, J. de Jesús Rubio, J. M. V. Kinani, and F. J. Gallegos-Funes, "An efficient nonlinear approach for removing fixed-value impulse noise from grayscale images," *J. Real-Time Image Process.*, vol. 14, no. 3, pp. 617–633, 2018, doi: 10.1007/s11554-017-0746-8.

[21] R. T. Cai *et al.*, "A Denoising Method for Salt and Pepper Noise in Remote Sensing Based on Swin-Transformer Convolution U-Net and Filtering—FSCU-Net," *Earth Sp. Sci.*, vol. 12, no. 8, p. e2025EA004225, Aug. 2025, doi: 10.1029/2025EA004225.

[22] G. B. Sagara, G. B. Krishna, and S. S. Rawat, "Hybrid Deep Learning Framework for Real-Time Source Code Vulnerability Detection," *Commun. Appl. Nonlinear Anal.*, vol. 32, no. 7s, pp. 889–900, 2025, doi: 10.52783/cana.v32.3493.

[23] S. Lee, D. Kim, and S. Kim, "A deep learning-based template matching through other field of view infrared image pair for real-time mixed reality," in *Proc.SPIE*, Jun. 2024, p. 1303403. doi: 10.1117/12.3013413.

[24] N. Cao and Y. Liu, "High-noise grayscale image denoising using an improved median filter for the adaptive selection of a threshold," *Appl. Sci.*, vol. 14, no. 2, p. 635, 2024, doi: 10.3390/app14020635.

[25] Q. Song and C. Gong, "Image reconstruction method for incomplete CT projection based on self-guided image filtering," *Med. Biol. Eng. Comput.*, vol. 62, no. 7, pp. 2101–2116, 2024, doi: 10.1007/s11517-024-03044-9.

[26] L. Alam and N. Kehtarnavaz, "Real-time generation of realistic defective wafer maps via deep learning network of CycleGAN," in *Real-Time Image Processing and Deep Learning 2023*, SPIE, 2023, pp. 84–93. doi: 10.1117/12.2663364.

[27] S. Fu, X. Meng, F. Shen, H. Chen, and Y. Cao, "A Hybrid Deep Learning Framework for Real-Time Fault Diagnosis and Prediction of Elevator Systems," *Comput. Fraud Secur.*, vol. 2025, no. 2, pp. 265–275, 2025, doi: 10.52710/cfs.497.

[28] K. K. A. Viswanathan, "Evaluation of Deep Learning Architectures for Image Denoising," *Int. J. Sci. Res.*, vol. 14, no. 3, pp. 1521–1525, 2025, doi: 10.21275/SR25324143238.

[29] M. Arhami, A. Desiani, S. Yahdin, A. I. Putri, R. Primartha, and H. Husaini, "Contrast enhancement for improved blood vessels retinal segmentation using top-hat transformation and otsu thresholding.," *Int. J. Adv. Intell. Informatics*, vol. 8, no. 2, pp. 210–223, 2022, doi: 10.26555/ijain.v8i2.779.

[30] K. Firdausy, O. Wahyunggoro, H. A. Nugroho, and M. B. Sasongko, "A new approach for sensitivity improvement of retinal blood vessel segmentation in high-resolution fundus images based on phase stretch transform.," *Int. J. Adv. Intell. Informatics*, vol. 8, no. 3, pp. 299–312, 2022, doi: 10.26555/ijain.v8i3.914.

[31] A. A. Wirabudi, N. R. Fachrurrozi, P. Dorand, and M. Royhan, "Enhancement of images compression using channel attention and post-filtering based on deep autoencoder.," *Int. J. Adv. Intell. Informatics*, vol. 10, no. 3, pp. 425–440, 2024, doi: 10.26555/ijain.v10i3.1499.