big data and cognitive computing

MDPI

Article

Reinforced Residual Encoder–Decoder Network for Image Denoising via Deeper Encoding and Balanced Skip Connections

Ismail Boucherit and Hamza Kheddar *

Laboratory of Advanced Electronic Systems, Department of Electrical Engineering, Faculty of Technology, University of Medea, Medea 26000, Algeria; boucherit.ismail@univ-medea.dz
* Correspondence: kheddar.hamza@univ-medea.dz

Abstract: Traditional image denoising algorithms often struggle with real-world complexities such as spatially correlated noise, varying illumination conditions, sensor-specific noise patterns, motion blur, and structural distortions. This paper presents an enhanced residual denoising network, R-REDNet, which stands for Reinforced Residual Encoder–Decoder Network. The proposed architecture incorporates deeper convolutional layers in the encoder and replaces additive skip connections with averaging operations to improve feature extraction and noise suppression. Additionally, the method leverages an iterative refinement approach, further enhancing its denoising performance. Experiments conducted on two real-world noisy image datasets demonstrate that R-REDNet outperforms current state-of-the-art approaches. Specifically, it attained a peak signal-to-noise ratio of 44.01 dB and a structural similarity index of 0.9931 on Dataset 1, and it obtained a peak signal-to-noise ratio of 46.15 dB with a structural similarity index of 0.9955 on Dataset 2. These findings confirm the efficiency of our method in delivering high-quality image restoration while preserving fine details.

Keywords: image denoising; REDNet; skip connections; residual networks; deep learning; autoencoder

check for updates

Academic Editor: Carson K. Leung

Received: 9 February 2025
Revised: 25 March 2025
Accepted: 28 March 2025
Published: 31 March 2025

Citation: Boucherit, I.; Kheddar, H. Reinforced Residual Encoder–Decoder Network for Image Denoising via Deeper Encoding and Balanced Skip Connections. *Big Data Cogn. Comput.* **2025**, *9*, 82. https://doi.org/10.3390/bdcc9040082

Copyright: © 2025 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

In the rapidly evolving landscape of technology, AI-driven autonomous systems like autonomous vehicles [1,2], underwater computer vision applications [3], video surveillance [4], person re-identification [5], biometric access control [6,7], and medical diagnostics are becoming indispensable. The effectiveness of these systems hinges significantly on the quality of the imagery they process. Clear images are crucial for optimal performance, yet real-world conditions often compromise image quality due to various factors, such as adverse weather, underwater environments, variable lighting, and camera motion. For instance, surveillance systems and medical imaging devices frequently capture images at lower resolutions; moving cameras introduce motion blur; underwater photography is plagued by color distortions and noise; and images captured in misty or rainy conditions are marred by blurs and additional noise. These degradations can drastically hinder the ability of visual systems to perform tasks like detection, segmentation, and target tracking effectively. To counter these challenges, developing robust IR algorithms is essential. These algorithms enhance the adaptability of visual systems across diverse environments. IR, a critical yet complex inverse problem, involves recovering a high-quality image from one that has been degraded. This process is typically categorized into six principal types based on the nature of the degradation: super-resolution, noise reduction, blur removal, rain removal, haze removal, and color adjustment.

Image denoising is a critical and dynamic area of research within image processing that is integral to numerous applications across various fields such as satellite imaging, defense reconnaissance, biometric image improvement [8], industrial automation, and forensics, among others. In medical and biomedical imaging such as COVID-19 [9], denoising algorithms serve as essential pre-processing steps to eliminate noise types like speckle, Rician, and Quantum [10]. These methods also reduce speckle in synthetic aperture radar images [11] and remove salt and pepper as well as additive white Gaussian noise in remote sensing [12]. In forensics, images can be corrupted by various kinds of noise, impacting the quality of evidence. Steganography techniques [13] can be used to hide data within images, further complicating the noise environment and the process of IR. Denoising techniques help suppress this noise [14]. Furthermore, denoising has applications in agriculture, such as filtering paddy leaves to detect rice plant diseases. Digital images often suffer distortions due to defective imaging equipment, problematic transmission channels, adverse atmospheric conditions, or environmental factors. These images can also be degraded during processes like encryption, compression, and format conversion [15]. Thus, the role of denoising extends to improving the integrity and usability of images in diverse and challenging conditions. Super-resolution techniques seek to generate a high-resolution image using one or multiple low-resolution images; deraining (along with dehazing and deblurring) focuses on removing impediments like raindrops, haze, or blur; and color correction is crucial for addressing distortions, particularly in underwater imaging.

Early image processing techniques relied on linear, non-linear, and non-adaptive filters as primary tools for noise reduction. Traditional denoising methods often utilized fundamental image filtering techniques such as Gaussian smoothing and diffusion-based approaches [16]. These methods generally function as low-pass filters, attenuating sudden intensity variations by replacing pixel values with an average or a similar statistical measure derived from neighboring pixels [17]. While these techniques effectively suppress high-intensity noise, they also tend to eliminate important details, leading to potential degradation in image quality [18]. To address this issue, adaptive filters, which assign weighted coefficients to the pixels within a specified window based on the fundamental principle of adaptive filtering [19], have been developed, including common types such as bilateral [20] and median filters [21]. Although these methods are computationally less demanding and yield reasonably good results, they also have significant drawbacks, including poor optimization during the test phase, the need for manual parameter settings, and reliance on specific denoising models.

Recent advances in DL have substantially improved the capabilities of IR technologies. These versatile models are designed to tackle multiple IR tasks simultaneously, benefiting from the substantial learning priors provided by DNNs. Enhanced by training on diverse and comprehensive publicly available datasets, this multi-task capability not only streamlines the restoration process but also boosts the efficiency and effectiveness of autonomous systems across various applications. The flexibility of DL-based methods has shown potential in overcoming these limitations. Particularly, the ongoing challenges of digital image deterioration have consistently created a strong demand for restoration through advanced computing technologies. The advent of machine learning and DL has been crucial, as these disciplines involve a learner that acquires knowledge from data and subsequently applies this learned insight to new, unseen data, proving instrumental in enhancing image denoising techniques.

This study introduces a reinforced residual denoising network, R-REDNet, which is designed specifically to address the inherent complexity and variability of real-world image noise. Traditional approaches and existing DL models often suffer from limitations such as the inadequate preservation of fine image details, poor handling of spatially correlated

noise, and residual noise amplification due to additive skip connections. To overcome these challenges, our proposed architecture integrates deeper convolutional layers to capture richer hierarchical features and introduces averaging-based skip connections instead of conventional additive methods. These enhancements not only improve feature extraction and reduce residual noise amplification but also result in a significant improvement in both quantitative metrics and visual quality. Thus, the proposed R-REDNet fills a critical gap by delivering robust and efficient denoising performance, which is essential for applications that rely heavily on image clarity, such as surveillance, medical imaging, and low-light photography. Our contribution is summarized as follows:

- Develop a reinforced REDNet architecture termed R-REDNet that enhances feature representation by increasing the depth of the encoder and refining the decoder structure.
- Introduce averaging-based skip connections to achieve a more balanced fusion of encoder and decoder features, reducing the amplification of residual noise.
- Extensive experiments are conducted on two real-world noisy image datasets to assess the generalizability of the proposed approach, demonstrating that our approach significantly outperforms SOTA denoising methods in terms of PSNR and SSIM.
- The proposed model exhibits enhanced generalization across diverse noise patterns, making it a robust solution for real-world applications such as surveillance, medical imaging, and photography under challenging conditions.

The paper is organized as follows: Section 2 provides background on residual neural networks (ResNet) and discusses the challenges in image denoising. Section 3 reviews related work, emphasizing the limitations of existing denoising techniques and the motivation for the proposed approach. Section 4 introduces the R-REDNet architecture, describing key modifications such as deeper convolutional layers and averaging-based skip connections. Section 5 outlines the experimental setup, detailing the datasets, pre-processing steps, and evaluation metrics. Section 6 provides a comparative analysis against existing denoising techniques. Finally, Section 7 summarizes the paper's key findings and outlines potential directions for future research.

2. Background

ResNets were developed to tackle the vanishing gradient problem, which is a frequent challenge in training extremely DNNs. This problem occurs because as gradients are backpropagated through layers, they tend to diminish, leading to very small updates to the weights and, ultimately, stagnating the training process.

ResNet tackle this issue by incorporating skip connections, also known as *identity shortcuts*, between layers. Skip connections are designed to connect the input of a layer directly to the output of a subsequent layer, bypassing one or more intermediate layers. This innovation significantly improves the training of DNNs by allowing the gradients to flow more easily through the network. The key idea behind ResNet is that instead of learning an unreferenced function $F(x)$, the network learns the residual function: $F(x) = H(x) - x$, where $H(x)$ is the true underlying mapping that the network seeks to learn. The residual function can then be rewritten as

$$H(x) = F(x) + x \tag{1}$$

Here, x represents the input to the layers, and $F(x)$ is the output of the stacked layers (i.e., the transformation the network learns). The identity mapping x is added to the output of $F(x)$, which simplifies the learning process, making it easier for the network to optimize and train, especially as the network depth increases. Skip connections play a multi-faceted role in the success of ResNet architectures:

- Mitigating the vanishing gradient problem: Skip connections help prevent the gradient from diminishing too much as it backpropagates through the network. This ensures that the weights of earlier layers are updated more effectively, enabling the training of very deep networks.
- Enabling the training of deeper networks: Skip connections allow the network to learn identity mappings more easily. In scenarios where additional layers do not improve the performance, the network can effectively "skip" those layers by learning to output the identity function $F(x) = 0$, thereby passing the input directly to the output. This adaptability enables the training of networks with hundreds or even thousands of layers without encountering the degradation issue, where deeper networks underperform compared to shallower ones.
- Facilitating residual learning: The core idea behind residual learning is that it is often easier to learn the residual than the full mapping. In image denoising, for example, the network can focus on learning the noise pattern in the image, which is then subtracted from the noisy input to produce a clean image. The skip connection ensures that the original input is always available at later stages, making it easier for the network to learn the residual noise rather than reconstructing the entire image from scratch.
- Improving gradient flow and network convergence: By creating multiple pathways for gradient propagation, skip connections improve the overall flow of gradients throughout the network. This leads to faster convergence during training and reduces the likelihood of getting stuck in poor local minima.

In the context of image denoising, ResNet-based models have been widely used because they can effectively capture both low-level details and high-level abstract features. Skip connections are particularly beneficial in denoising tasks as they allow the network to preserve important image details across layers. Since the noise is typically a small perturbation added to the original clean image, it is beneficial for the network to have access to the original image data at multiple stages of the network. Skip connections ensure that these original details are preserved and combined with the learned noise patterns at each residual block. Mathematically, if we denote the noisy image as y, the clean image as x, and the noise as n, we aim to estimate the noise n such that $y = x + n$. The ResNet model learns a mapping $F(y)$ that approximates the noise n, and the clean image can be recovered by $\hat{x} = y - F(y)$. Figure 1 illustrates a typical residual block, where the shortcut connection is depicted by the arch encircling the layers, representing the addition of the input directly to the output of the convolutional layers.

This architecture has proven highly effective in various image denoising tasks, as it allows the network to concentrate on modeling the noise pattern, leading to improved denoising performance and generalization across different types of noise and image datasets.

The challenge of removing noise while preserving important image details has spurred considerable research, leading to the development of various techniques ranging from traditional filters to advanced DL models. Historically, image denoising techniques have included approaches such as Gaussian filtering, median filtering, and wavelet-based methods [22]. These methods, while effective in some scenarios, often struggle with maintaining image details and textures, especially in the presence of high levels of noise. Recent advancements in DL have revolutionized the field of image denoising. One such DL technique is the REDNet, which has demonstrated its effectiveness in denoising real-world images [23]. The REDNet architecture is a DNN that has the strength of residual learning to achieve SOTA performance in image denoising [24].

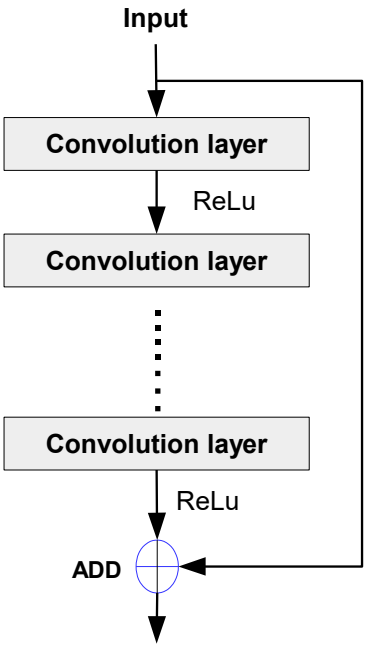


Figure 1. Diagram of a residual block. The shortcut connection is depicted by the arch encircling the layers, which directly adds the input to the output of the convolutional layers.

While REDNet has shown promising results, the performance of image denoising models can be further enhanced by incorporating additional refinement mechanisms. The primary limitation of REDNet, as observed in practice, is that the output quality can still be improved particularly in terms of fine detail preservation and noise suppression in highly textured regions. To address this limitation, researchers have proposed integrating refinement networks with existing denoising models. These refinement networks act as post-processing units that fine-tune the outputs of primary denoising models. For instance, methods such as the residual dense network for image restoration [25] have demonstrated that adding a refinement step can significantly enhance the visual quality and reduce artifacts in the denoised images.

In this study, the authors propose an enhanced approach to image denoising by modifying the existing REDNet architecture to create a R-REDNet. Our approach improves feature extraction and noise suppression by introducing deeper convolutional layers, replacing additive skip connections with averaging operations. These enhancements enable the model to achieve superior denoising performance.

3. Related Work

This section provides an analytical overview of recent CNN-based image denoising approaches. We categorize the existing literature into supervised and unsupervised methods, highlighting their contributions, limitations, and relevance to our proposed method.

3.1. Supervised-Based Image Denoising

Supervised learning approaches have dominated the image denoising landscape due to their strong performance on synthetic datasets and controlled environments. These models typically rely on paired noisy–clean data and are trained to minimize pixel-wise differences. A foundational example is DN-ResNet by Ren et al. [26], which uses residual blocks and edge-aware loss functions to boost perceptual quality. Their follow-up, DS-DN-ResNet, reduces computational cost using depthwise separable convolutions while maintaining performance across various noise types. Building on these designs, SRR-Net [27] integrates multi-scale dual attention with a high-resolution reconstruction module,

enhancing feature recovery in noisy images. Attention mechanisms have proven particularly effective in focusing the model on informative regions and have thus become common in recent architectures.

In addition to structural innovations, supervised methods have begun incorporating iterative refinement strategies. For instance, Pan et al. [28] introduce a correction mechanism for noise-level maps, leveraging guided feature domain residual networks to reduce artifacts. Meanwhile, TSIDNet [29] improves generalization through a dual-subnet structure, combining transfer learning with cross-fusion modules to handle diverse noise patterns. In the realm of low-light conditions, LIVENet [30] proposes a two-stage pipeline that replaces noisy luminance channels using a latent subspace and spatial feature transforms, thereby optimizing both enhancement and denoising simultaneously. While these supervised models show impressive results, they often struggle to generalize to real-world noisy data due to their dependence on clean–noisy pairs and assumptions about noise distribution.

3.2. Unsupervised-Based Image Denoising

Unsupervised denoising methods address the lack of clean target data, which is a significant constraint in real-world scenarios. These approaches generally adopt noise modeling, self-supervision, or zero-shot learning to train models without paired data. FADNet [31] uses a dynamic filtering mechanism with attention modules and Fourier transforms to adapt to different noise types. This adaptive strategy provides robustness to noise variations, which many fixed-architecture supervised methods lack. Complementing this, the clean-to-noisy framework [32] simulates realistic noise from clean data, making traditional CNNs more applicable to real-world settings. Generative methods have also evolved with self-collaboration strategies, as in Lin et al. [33], which iteratively improve synthetic noisy–clean pairs by refining weaker denoisers. This incremental learning approach addresses convergence issues commonly seen in GAN-based methods. Blind-spot network architectures have become prominent for self-supervised denoising. The Conditional Blind-Spot Network by Jang et al. [34] introduces spatial decorrelation while including central pixel information, which was previously excluded. Similarly, Zhang et al. [35] incorporate Transformer blocks for global–local feature fusion, enhancing noise discrimination capabilities. These approaches address a key issue in semi-supervised methods: balancing context exploitation with noise suppression.

Noise estimation has also been tackled using non-deep learning techniques. NLH [36] proposes a pixel-level non-local prior using the Haar transform and Wiener filtering. Although less flexible than deep models, such approaches provide strong baselines and insight into interpretable noise reduction. AP-BSN [37] represents an evolution in blind-spot design, using asymmetric pixel-shuffle downsampling and random-replace refinement to counteract pixel-wise independence assumptions, which are often unrealistic in natural images. Furthermore, dataset-free and lightweight approaches have gained attention. Zero-shot Noise2Noise [38] uses a simple two-layer architecture for real-time denoising without any prior training or noise statistics. This minimalistic yet effective model is ideal for constrained environments. MBPDR3 [39] and MASH [40] extend blind-spot learning with multi-branch architectures and refinement via masking and shuffling techniques. These methods address spatially correlated noise—often overlooked in pixel-independent models—by incorporating contextual blending. Lastly, curvelet-based methods like CTuNLM [41] offer an alternative to CNNs by focusing on multi-scale frequency decomposition, outperforming several deep models in RGB denoising tasks. The introduction of Deep Curvelet-Net extends this approach with neural layers focused on the finest curvelet coefficients, bridging the gap between traditional and deep learning methods.

Despite their effectiveness, these existing approaches still encounter limitations related to the insufficient preservation of fine-grained image details, amplification of residual noise due to additive skip connections, and inadequate feature extraction depth. Therefore, our research specifically addresses these gaps by proposing R-REDNet, which integrates deeper convolutional layers and averaging-based skip connections. By doing so, we aim to provide substantial improvements in denoising performance, especially in complex real-world scenarios.

4. Proposed Architecture

The original REDNet model, which employs a combination of convolutional and transposed convolutional layers with additive skip connections, was designed to efficiently reconstruct images by learning residual representations. However, to improve its feature extraction capabilities and smooth the reconstructed output, we propose a modified version, termed R-REDNet, which introduces deeper convolutional layers in the encoder and replaces additive skip connections with averaging operations.

4.1. REDNet Architecture

The proposed model is based on a fully convolutional architecture, incorporating deconvolution layers to restore the image to its original resolution. Rectification layers are applied after each convolution and deconvolution step to enhance information propagation. The convolutional layers function as feature extractors, capturing the essential structural components of the image while suppressing unwanted noise. Once the noisy input image passes through these layers, a refined version is generated. However, some fine details may be lost in the process. To address this, deconvolution layers are employed to recover lost details and improve reconstruction accuracy. Additionally, skip connections are introduced between corresponding convolutional and deconvolutional layers. These connections facilitate the direct transfer of feature maps from the encoding path to the decoding path, where they are merged element-wise before undergoing rectification, ensuring better preservation of spatial details.

4.1.1. Convolution–Deconvolution Decoder

Recently, architectures that integrate convolutional and deconvolutional layers have been suggested for semantic segmentation tasks [23]. The convolution operation entails calculating the integral of the product between two functions: the input (x) and the kernel (w). The outcome, commonly known as the feature map or activation map (s), is calculated as follows:

$$s(t) = (w * x)(t) = \sum_{a} [x(t + a)w(a)] \quad (2)$$

Unlike convolutional layers, which aggregate multiple input activations within a filter window to generate a single output activation, deconvolutional layers distribute a single input activation across multiple outputs. Typically, deconvolution serves as a learnable upsampling mechanism. In [23], convolutional layers incrementally down-sample the input image, condensing it into a compact representation, while deconvolutional layers up-sample this representation to reconstruct the original resolution. Unlike high-level tasks such as segmentation or recognition, where pooling can discard fine image details and degrade restoration performance, the REDNet mitigates this issue. Alternatively, substituting deconvolution with convolution results in an architecture resembling recently introduced deep fully convolutional neural networks.

4.1.2. Skip Connections

Skip connections are introduced between corresponding convolutional and deconvolutional layers, as illustrated in Figure 1. These connections serve two main purposes. First, as the network deepens, image details may be lost, reducing the effectiveness of deconvolution in restoring them. Skip connections, however, transfer feature maps rich in image details, enhancing deconvolution's ability to produce a cleaner version of the image. Second, skip connections also facilitate better gradient backpropagation to the lower layers, making it easier to train deeper networks. It is important to note that our skip connections differ from other approaches, where the focus was solely on optimization [23].

4.2. Proposed R-REDNet

In this work, we propose a modified version of the REDNet architecture, referred to as reinforced REDNet, which introduces a series of enhancements aimed at improving its capacity for complex feature extraction and more refined image restoration. The base REDNet architecture employs a series of convolutional layers for feature encoding, followed by transposed convolutional layers for decoding, utilizing additive skip connections to recover fine spatial details. The architecture of the proposed network is shown in Figure 2. The modifications made to the original REDNet were the result of extensive experimentation and testing. The input image size was 256 × 256 pixels.

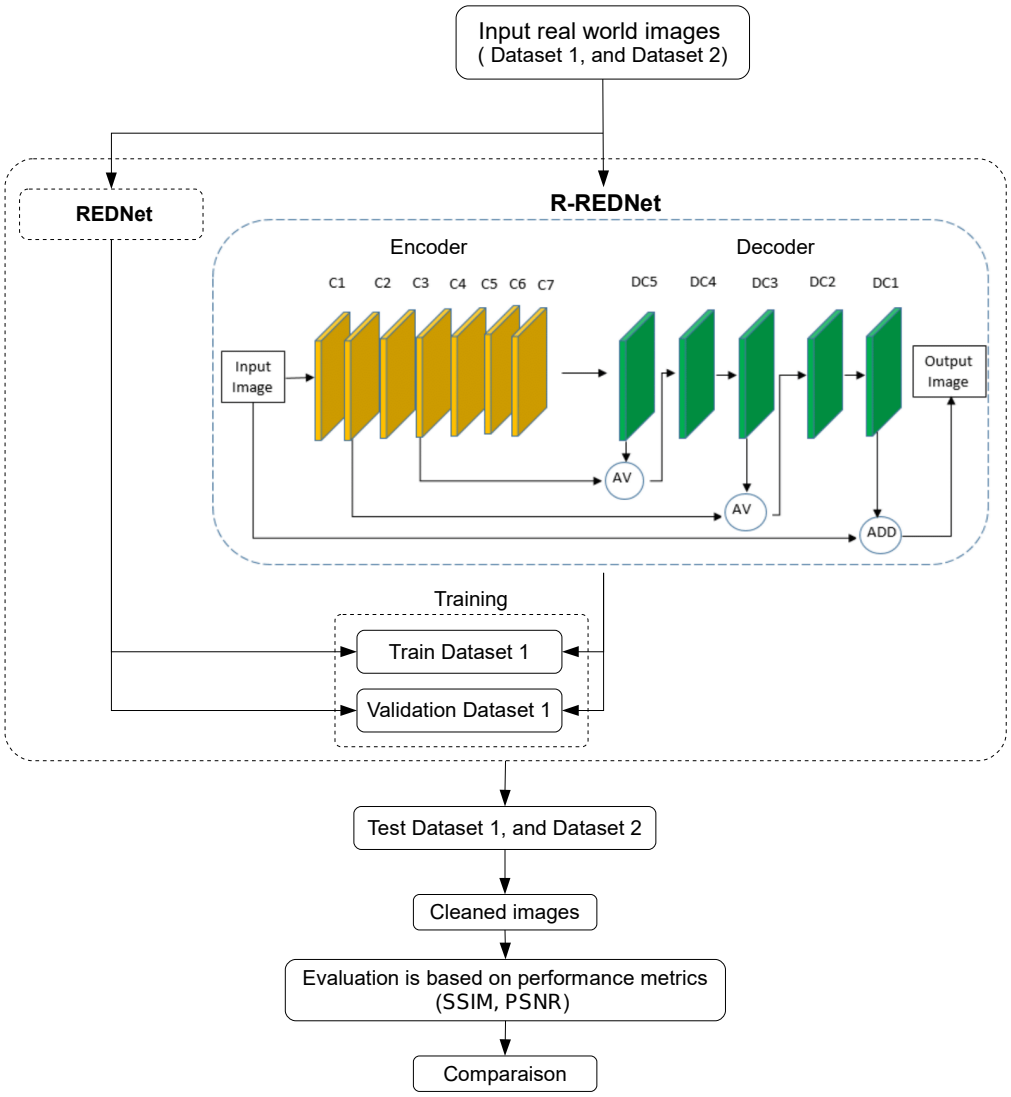


Figure 2. The design of the proposed architecture R-REDNet.

4.2.1. Deeper Encoder

In the original REDNet, the encoder consisted of five convolutional layers with progressively reduced filter counts, starting from 256 and ending with 128 filters. To reinforce the model's capacity for feature extraction, two additional convolutional layers were added, yielding a total of seven layers in the encoder. Specifically, two new layers (conv layer 6 and conv layer 7) were introduced after the fifth convolutional layer, with 128 and 64 filters, respectively. Increasing the encoder depth allows the network to progressively refine feature representations, capturing both low-level textures and high-level abstract patterns. By stacking additional layers, the model gains the ability to better separate noise from meaningful image features, making denoising more effective. The expanded depth also increases the receptive field, enabling neurons to analyze broader spatial dependencies and structural patterns within the image. This hierarchical learning process ensures that the encoder extracts more robust and noise-resilient feature maps, ultimately leading to improved reconstruction accuracy and enhanced visual clarity in denoised images. The deeper encoder also facilitates the extraction of higher-order spatial features, thereby improving the model's performance in tasks such as image denoising or restoration.

4.2.2. Employing Modified Skip Connections

The REDNet architecture employs additive skip connections (via Add layers) between corresponding encoder and decoder layers to preserve spatial information during the upsampling process. In the reinforced REDNet, we propose an alternative to this approach by using averaging (Average layers) for the skip connections. The rationale behind this modification is to allow a more balanced blending of features between the encoder and decoder, potentially leading to smoother reconstructions.

Specifically, averaging is applied at two key points in the network:

- Between the output of the second convolutional layer (encoder) and the corresponding decoder layer (transposed convolution).
- To preserve the residual learning framework, the final skip connection, which adds the input image to the output of the decoder, remains intact. This residual connection helps the model predict image residuals and subsequently reconstruct the denoised or restored version by adding these residuals back to the input.

The average layer serves as a merging operation that combines multiple noisy feature maps by performing an element-wise average. This operation is especially valuable when integrating feature maps from different stages of a denoising network, ensuring that the spatial dimensions of the image are preserved while simultaneously smoothing out noise.

Replacing additive skip connections with averaging operations in R-REDNet enhances denoising performance by reducing noise amplification and improving feature fusion. Additive skip connections directly sum feature maps, which can inadvertently amplify residual noise, especially in high-SNR scenarios. In contrast, averaging normalizes the combined feature maps, mitigating noise propagation and ensuring a more balanced feature representation. This approach also stabilizes gradient flow, preventing issues such as vanishing or exploding gradients in deep networks. Additionally, averaging promotes more effective feature fusion by reducing the dominance of noisy components, leading to a refined reconstruction process. Since noise is often stochastic and uncorrelated across layers, averaging operations suppress high-frequency noise artifacts while retaining crucial structural details in images. As a result, this modification enhances the network's ability to recover clean images while ensuring more stable training. By incorporating averaging instead of direct addition, R-REDNet achieves improved denoising performance and more effective noise suppression. In contrast to element-wise addition, which can amplify certain

pixel values and potentially overemphasize noise, the average layer ensures a balanced contribution from all inputs, allowing for effective denoising without amplifying artifacts. This makes the average layer particularly useful in multi-path denoising architectures, as it facilitates the fusion of multiple noisy representations in a way that emphasizes the clean underlying signal. By providing a robust, noise-reducing combination of features, the average layer plays a key role in improving the overall quality and accuracy of the denoised image.

4.2.3. Mathematical Description of R-REDNet

The architecture of the modified REDNet, referred to as "Reinforced R-REDNet", includes encoder and decoder layers with skip connections utilizing averaging. Below is the detailed mathematical description:

- Encoder layers: The encoder consists of seven convolutional layers $C_1, C_2, \ldots, C_7$. Let the input image be $\mathbf{I}$. Each encoder layer applies a convolution operation followed by an activation function:

$$\mathbf{F}_i = f(\mathbf{W}_i * \mathbf{F}_{i-1} + \mathbf{b}_i), \quad i \in \{1, 2, \ldots, 7\},$$

  where the input image is represented as $\mathbf{F}_0 = \mathbf{I}$. The weights and biases of the i-th layer are denoted by $\mathbf{W}_i$ and $\mathbf{b}_i$, respectively. The convolution operation is represented by $*$, while f denotes the activation function, such as ReLU.
- Decoder layers: The decoder consists of five transposed convolutional layers DC_1, DC_2, …, DC_5. These layers perform upsampling and reconstruction. For the j-th decoder layer, the operation is

$$\mathbf{G}_j = g(\mathbf{V}_j *^T \mathbf{G}_{j-1} + \mathbf{c}_j), \quad j \in \{1, 2, \ldots, 5\},$$

  where the weights and biases of the j-th decoder layer are represented by $\mathbf{V}_j$ and $\mathbf{c}_j$, respectively. The transposed convolution operation is denoted by $*^T$, while g represents the activation function. The output image is given by $\mathbf{G}_5$.
- Skip connections: At certain layers, the skip connections from the encoder outputs are concatenated with the decoder outputs using averaging. Let $\mathbf{F}_k$ be the output of encoder layer k and $\mathbf{G}_j$ be the output of decoder layer j. The averaging operation is defined as

$$\mathbf{AV}(\mathbf{F}_k, \mathbf{G}_j) = \frac{\mathbf{F}_k + \mathbf{G}_j}{2}.$$

  The decoder layer input is updated with this concatenated information:

$$\mathbf{G}'_j = \mathbf{AV}(\mathbf{F}_{7-j}, \mathbf{G}_j).$$

- Final output: The output image $\mathbf{O}$ is produced after processing through the final decoder layer DC_1, incorporating all relevant skip connections and averaging operations:

$$\mathbf{O} = \mathbf{G}_1.$$

- Complexity: While R-REDNet is computationally more expensive than standard shallow CNN-based denoising models due to its increased depth and hierarchical feature extraction, it offers better denoising performance with less noise amplification. The additional cost is justified by the model's ability to preserve fine image details while reducing artifacts. However, for real-time applications, optimized implementations (e.g., quantization, pruning) may be required to balance computational efficiency and performance. The computational complexity of a single convolutional layer is given by

$$\mathcal{C} = O\Big(L \cdot K^2 \cdot C_{in} \cdot C_{out} \cdot H \cdot W\Big) \tag{3}$$

where L is the number of layers, K is the kernel size, C_{in} and C_{out} represent input and output channels, and H, W denote the spatial dimensions of the feature map. The original REDNet consists of five convolutional layers in its encoder and decoder, resulting in a total of $L_{\text{REDNet}} = 10$ layers. In contrast, R-REDNet introduces two additional convolutional layers in the encoder, making its total depth $L_{\text{R-REDNet}} = 12$.
Thus, the complexity ratio between R-REDNet and REDNet can be expressed as

$$\frac{\mathcal{C}_{\text{R-REDNet}}}{\mathcal{C}_{\text{REDNet}}} = \frac{L_{\text{R-REDNet}}}{L_{\text{REDNet}}} = \frac{12}{10} = 1.2 \tag{4}$$

This indicates that R-REDNet has approximately 20% higher computational complexity than REDNet due to the increased depth. However, the introduction of averaging operations instead of additive skip connections has negligible computational overhead, since element-wise averaging involves only division operations. Despite the increased complexity, R-REDNet improves feature extraction, reduces noise amplification, and achieves better denoising performance compared to REDNet.

Table 1 provides a detailed summary of the hyperparameters for each layer in both the encoder and decoder. While the REDNet base and R-REDNet models share a similar architectural foundation, they differ in training time, parameter count, and overall model size, as summarized in Table 1. R-REDNet introduces a slight increase in complexity with a marginally larger parameter count and extended training time. However, this modest increase results in enhanced denoising efficiency, making R-REDNet a more effective choice for noise reduction. The model achieves a well-balanced trade-off between size and performance, offering improved denoising capabilities without a substantial rise in computational cost.

Table 1. Layers of the suggested model.

Index	Layer Type	Number of Filters	Output Shape
#1	Input	–	(256, 256, 3)
#2	Convolution	256	(256, 256, 256)
#3	Convolution	256	(256, 256, 256)
#4	Convolution	256	(256, 256, 256)
#5	Convolution	256	(256, 256, 256)
#6	Convolution	128	(256, 256, 128)
#7	Convolution	128	(256, 256, 128)
#8	Convolution	64	(256, 256, 64)
#9	Deconvolution	256	(256, 256, 256)
#10	Average		(256, 256, 256)
#11	Deconvolution	256	(256, 256, 256)
#12	Average		(128, 256, 64)
#13	Deconvolution	256	(256, 256, 256)
#14	Deconvolution	256	(256, 256, 3)
#15	ADD		(256, 256, 3)
#16	Output		(256, 256, 3)
Model summary	REDNet	TTT TP (Size)	27.55 min 3,023,235 (11.53 MB)
	R-REDNet	**TTP TP (Size)**	**28.31 min 3,179,075 (12.13 MB)**

Abbreviations: total training time (TTT); total parameters (TPs).

5. Experimental Results

This section presents the evaluation framework used to assess the proposed method. It includes a detailed description of the metrics utilized to measure performance, providing insight into the effectiveness and reliability of the approach. Additionally, Section 5.2 outlines the characteristics of the data employed in the experiments, highlighting their relevance to the study. Finally, Section 5.4 analyzes the findings, comparing them with baseline methods and discussing their implications in the context of the research objectives.

5.1. Metrics

As evaluation metrics, PSNR [42] and SSIM [43] are employed to assess the effectiveness of the proposed system, as defined in Equations (5) and (6). PSNR quantifies the peak signal-to-noise ratio between the original and reconstructed images with higher values indicating better reconstruction quality. In contrast, SSIM evaluates the perceptual similarity between the original and reconstructed (or denoised) image. An SSIM value of '1' indicates that the original and reconstructed images are identical, while values less than '1' suggest increasing dissimilarity between the two images.

$$PSNR = 10 \cdot \log_{10}\left(\frac{255^2}{RMSE}\right) \tag{5}$$

SSIM evaluates the similarity between two images, which is crucial for image denoising as it helps assess how effectively the denoising method has reconstructed the clean image from the noisy input. Unlike individual pixel-based metrics, such as MSE, SSIM focuses on windows or groups of pixels that carry essential information about the object, making it more meaningful in image denoising tasks. Since SSIM takes a window-based approach rather than evaluating individual pixels like MSE, it provides a more comprehensive comparison. Given two windows, X and Y, each of size M $\times$ M, SSIM is computed as follows:

$$SSIM(X,Y) = \frac{(2\mu_X\mu_Y + c_1)(2\sigma_{XY} + c_2)}{(\mu_X^2 + \mu_Y^2 + c_1)(\sigma_X^2 + \sigma_Y^2 + c_2)} \tag{6}$$

where μ_X, μ_Y, σ_X^2, σ_Y^2, and σ_{XY} denote the mean of X, the mean of Y, the variance of X, the variance of Y, and the covariance between X and Y, respectively. The constants $c_1 = (k_1 L)^2$ and $c_2 = (k_2 L)^2$ are introduced to maintain numerical stability when the denominator is too small or approaches zero with default values $k_1 = 0.01$ and $k_2 = 0.03$. Here, L represents the dynamic range of pixel values. The SSIM score ranges from -1 to 1, where 1 indicates perfect similarity and 0 signifies no similarity. Therefore, a higher SSIM value, approaching 1, reflects the better performance of the denoising technique.

5.2. Datasets

To improve the generalizability of the proposed paper, tests were conducted on two datasets instead of one. These datasets are widely used for denoising real images, addressing challenges that extend beyond additive white Gaussian noise to tackle the more complex task of real-world noisy image denoising. The datasets utilized are outlined below:

- Dataset 1: The dataset introduced by Xu et al. [44] is considered as benchmark dataset specifically designed to capture the complexity of real-world noise across diverse natural scenes. This dataset consists of noisy images captured using five different camera models from three leading brands: Canon EOS (models 5D Mark II, 80D, and 600D), Nikon D800, and Sony A7 II, ensuring a broad representation of sensor characteristics and noise patterns. The dataset includes 40 unique scenes, covering a variety of indoor environments and objects. From these scenes, 100 image patches of size 512 $\times$ 512 pix-

els were cropped, providing a standardized set for evaluation. The dataset offers a diverse collection of real-world noise types, making it an essential resource for developing and benchmarking denoising algorithms under practical conditions. Figure 3 showcases sample images from this dataset, demonstrating the presence of complex noise patterns across different lighting conditions and scene compositions.

- Dataset 2: The dataset introduced by Nam et al. [45] provides a comprehensive collection of noisy images captured from 11 static scenes using three different camera models: the Canon 5D Mark 3, Nikon D600, and Nikon D800. For each scene, 500 JPEG images were taken under consistent settings, and an averaged image was generated to serve as the noise-free ground truth reference. This averaging technique effectively minimizes noise, providing a reliable benchmark for evaluating denoising performance. To facilitate comparative analysis, 15 regions of size 512 $\times$ 512 pixels were extracted from the captured scenes, offering a standardized evaluation set for testing denoising algorithms. Figure 4 illustrates sample images from this dataset, showcasing the diversity of scene content and noise characteristics.



Figure 3. Selected cropped sections of the "clean" images (**left**) and their respective noisy versions "real" (**right**) in Dataset 1.



Figure 4. Selected cropped sections of the "clean" images (**left**) and their respective noisy versions "real" (**right**) in Dataset 2.

5.3. Implementation Details

- Hardware configuration: The experiments were conducted on the Kaggle cloud-based computing platform using a dual NVIDIA Tesla P100/T4 GPU setup. The system configuration includes four vCPUs and 16 GB of RAM, ensuring efficient model training and evaluation. The DL model was implemented using TensorFlow and the Keras API, leveraging GPU acceleration for faster processing. Model training and inference were conducted using the CUDA and cuDNN-optimized TensorFlow environment, significantly reducing computation time.
- Experimental summary: This study employs a deep convolutional autoencoder with skip connections for real-world image denoising. Dataset 1, used for training, comprises 100 images, which are divided into 60 for training, 30 for validation, and 10 for testing. The training images were resized to 256 × 256 pixels and normalized within the [0, 1] range. Table 2 summarizes the selected hyperparameters.

Table 2. Hyperparameter configuration for model training.

Hyperparameter	Value
Learning rate	0.0001
Batch size	4 (Optimized for GPU)
Number of epochs	100
Optimizer	Adam
Loss function	MSE
Image size	256 × 256 × 3

5.4. Result Discussion

To better understand the impact of the individual components within the proposed R-REDNet architecture, an ablation study was conducted and presented in Table 3. The goal of this study was to evaluate the contribution of key modifications, such as the introduction of deeper convolutional layers in the encoder and the replacement of additive skip connections with averaging operations, to the overall denoising performance. The baseline REDNet model achieves a PSNR of 38.93 dB and an SSIM of 0.9846.

Table 3. Effect of an enhanced encoder depth, averaging-based skip connections, and iterative refinement on PSNR and SSIM.

Model Variant	Encoder Depth	Skip Connection	Iterative Refinement	PSNR (dB)	SSIM
Baseline REDNet	5 layers	Addition	No	38.93	0.9846
Deeper encoder	7 layers	Addition	No	41.25 (+2.32)	0.9881
Averaging skip connections	7 layers	Averaging	No	42.78 (+1.53)	0.9903
Final R-REDNet	7 layers	Averaging	Yes	**44.01** (+1.23)	**0.9931**

5.4.1. Effect of Deeper Encoder

The encoder depth was increased by adding two additional convolutional layers to the original REDNet architecture. This modification aims to enhance feature extraction by capturing finer details and deeper spatial information. To evaluate the impact of this change, we conducted experiments comparing the original REDNet and the proposed R-REDNet, excluding the additional convolutional layers. The results, summarized in Table 4, show a significant improvement in both PSNR and SSIM, highlighting the crucial role of the deeper encoder in noise reduction and preserving fine image details. Increasing the encoder depth from five to seven convolutional layers results in a PSNR improvement of +2.32 dB and an SSIM increase of 0.0035, confirming that a deeper encoder improves the model's ability to capture complex noise structures while maintaining fine details.

Table 4. Comparison of the Mean PSNR (dB) and SSIM scores for various denoising algorithms, including R-REDNet, evaluated on 100 cropped images from Dataset 1.

Metrics	Classical Methods				CNN Based Methods		
	TNRD [46]	NC [47]	MCWNNM [48]	NI [49]	DnCNN [50]	REDNet [23]	R-REDNet
SSIM	0.9640	0.9449	0.9671	0.9570	0.9161	0.9846	**0.9931**
PSNR	38.17	36.92	38.51	37.77	36.08	38.93	**44.01**

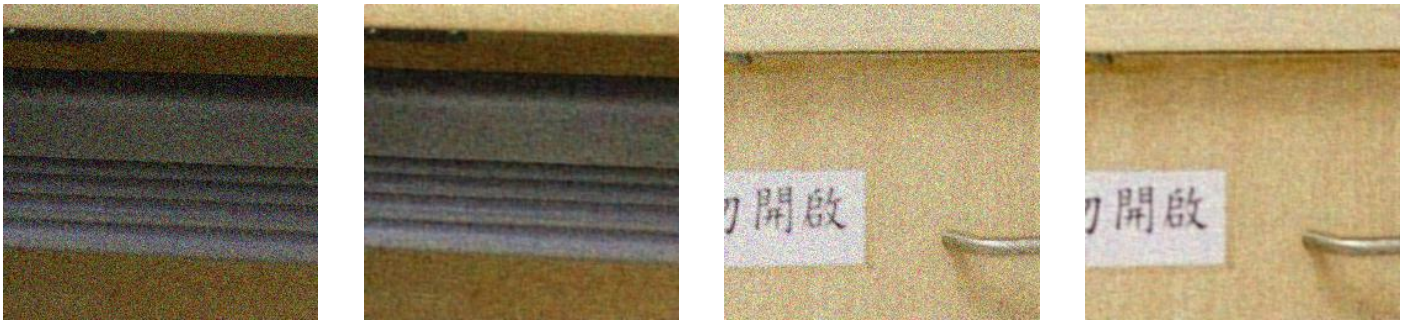
5.4.2. Effect of Averaging-Based Skip Connections

Introducing iterative refinement improves PSNR by an additional +1.23 dB, bringing the final R-REDNet performance to 44.01 dB. This confirms that iterative refinement progressively suppresses residual noise, improving overall image quality without excessive smoothing. The results validate that each proposed modification significantly enhances denoising performance with the final R-REDNet outperforming the baseline REDNet by 5.08 dB in PSNR and 0.0085 in SSIM.

To highlight the effectiveness of the proposed R-REDNet model, Figures 5 and 6 present a qualitative comparison between input images degraded by Gaussian noise (mean = 0, standard deviation = 0.1) and their corresponding denoised outputs for Dataset 1 and Dataset 2, respectively. The visual results demonstrate that R-REDNet effectively removes noise while preserving fine image details and textures. Compared to the original noisy images, the denoised outputs exhibit significant improvements in clarity with reduced noise artifacts and enhanced structural consistency. These observations align with the quantitative performance gains in PSNR and SSIM, confirming that R-REDNet not only achieves superior numerical results but also produces visually appealing restorations. The ability to generalize across different noise levels and image textures makes R-REDNet a robust solution for real-world denoising applications.



(**a**) Noised image 1 (**b**) Denoised image 1 (**c**) Noised image 2 (**d**) Denoised image 2

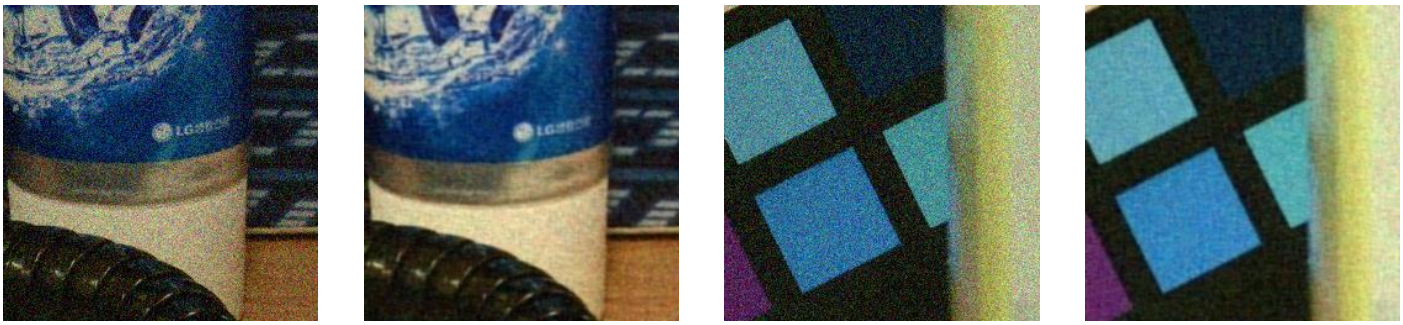


(**e**) Noised image 3 (**f**) Denoised image 3 (**g**) Noised image 4 (**h**) Denoised image 4

Figure 5. Comparison of noised and denoised images across four different images of Dataset 1.



(**a**) Noised image 1 (**b**) Denoised image 1 (**c**) Noised image 2 (**d**) Denoised image 2



(**e**) Noised image 3 (**f**) Denoised image 3 (**g**) Noised image 4 (**h**) Denoised image 4

Figure 6. Comparison of noised and denoised images across four different images od Dataset 2.

6. Comparaison with the SOTA

To assess the performance of the proposed R-REDNet model, we carried out a comparative analysis against SOTA image denoising techniques on two real-world noisy image datasets. The study includes traditional denoising approaches such as trainable non-linear reaction diffusion (TNRD) [46], noise clinic (NC) [47], multi-channel weighted nuclear norm minimization (MCWNNM) [48], and neat image (NI) [49], along with DL-based models like

DnCNN [50] and REDNet. The effectiveness of these models was evaluated using PSNR and SSIM, which are two widely used metrics for measuring image restoration quality.

The results, as summarized in Tables 4 and 5, demonstrate that DL-based approaches significantly outperform classical denoising techniques. Among classical methods, MCWNNM achieved the best performance, with a PSNR of 38.51 dB and an SSIM of 0.9671 on Dataset 1 and a PSNR of 37.71 dB with an SSIM of 0.9542 on Dataset 2. However, these values remain considerably lower than those achieved by DL-based models, highlighting the limitations of traditional denoising approaches in handling real-world noise complexities. Notably, the DnCNN model exhibited the weakest performance among CNN-based methods, achieving the lowest PSNR and SSIM values in both datasets. This suggests that simple deep networks without specific architectural enhancements struggle to generalize to real-world noise.

Table 5. Comparison of the Mean PSNR (dB) and SSIM scores for various denoising algorithms, including R-REDNet, assessed on 10 cropped images from Dataset 2.

Metrics	Classical Methods				CNN Based Methods		
	TNRD [46]	**NC [47]**	**MCWNNM [48]**	**NI [49]**	**DnCNN [50]**	**REDNet [23]**	**R-REDNet**
SSIM	0.9463	0.9364	0.9542	0.9126	0.8635	0.9953	**0.9955**
PSNR	36.61	36.43	37.71	35.49	33.86	42.59	**46.15**

Figure 7 presents a comparative analysis of PSNR values across Dataset 1 and Dataset 2, allowing for a direct performance comparison of each denoising method. The results indicate that R-REDNet consistently outperforms all competing methods in both datasets with significantly higher PSNR values. Notably, while classical methods such as MCWNNM perform relatively well, their performance is notably lower than that of DL-based methods. Furthermore, the REDNet model shows clear improvements over traditional CNN-based models like DnCNN, but R-REDNet achieves the best results, demonstrating the effectiveness of its reinforced residual connections and iterative refinement.

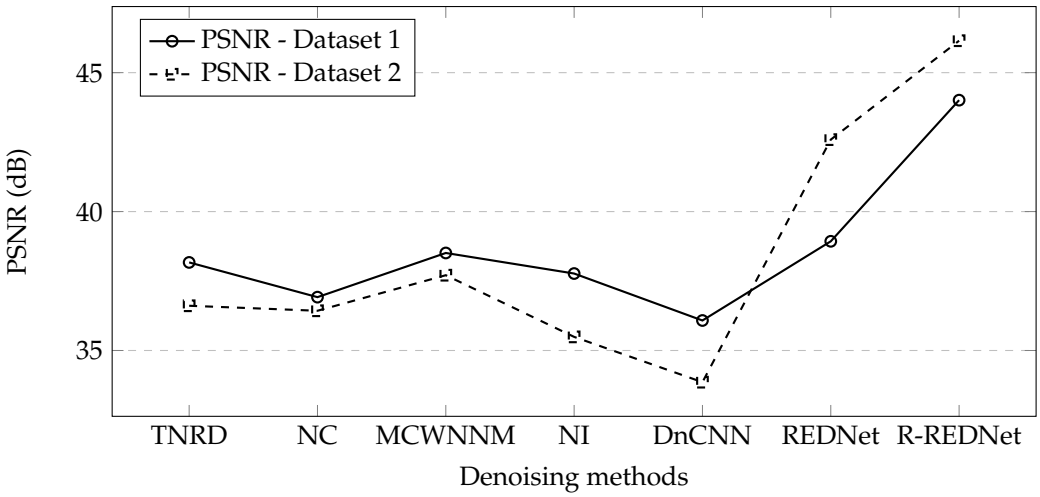


Figure 7. PSNR comparison of denoising methods (Dataset 1 and Dataset 2).

Similarly, Figure 8 compares the SSIM values for both datasets, further reinforcing the superior performance of R-REDNet. The figure highlights that DL-based methods yield significantly higher SSIM values compared to traditional approaches with R-REDNet achieving near-perfect SSIM scores on Dataset 2. Interestingly, while REDNet already shows strong structural similarity preservation, R-REDNet further improves upon it, indicating

that the proposed enhanced feature extraction and noise suppression strategies contribute to preserving finer image details.

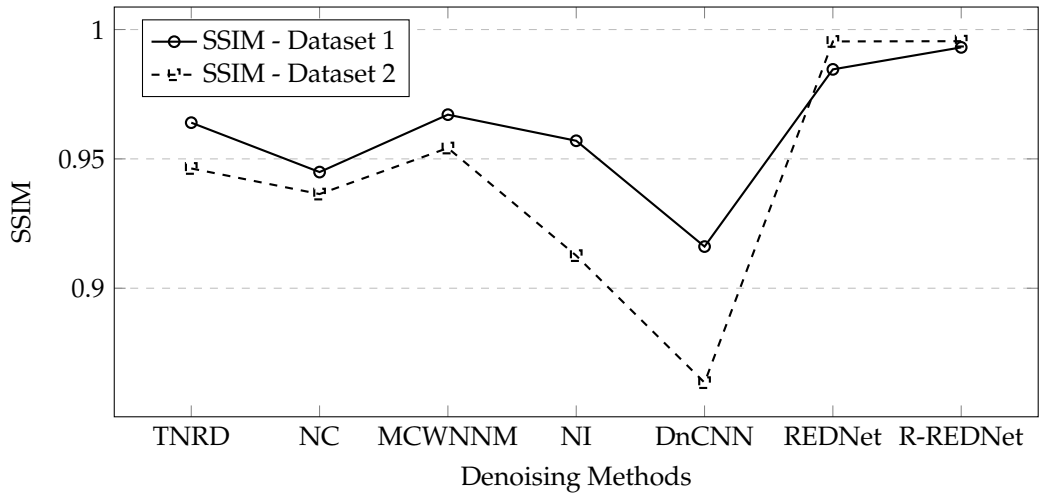


Figure 8. SSIM comparison of denoising methods (Dataset 1 and Dataset 2).

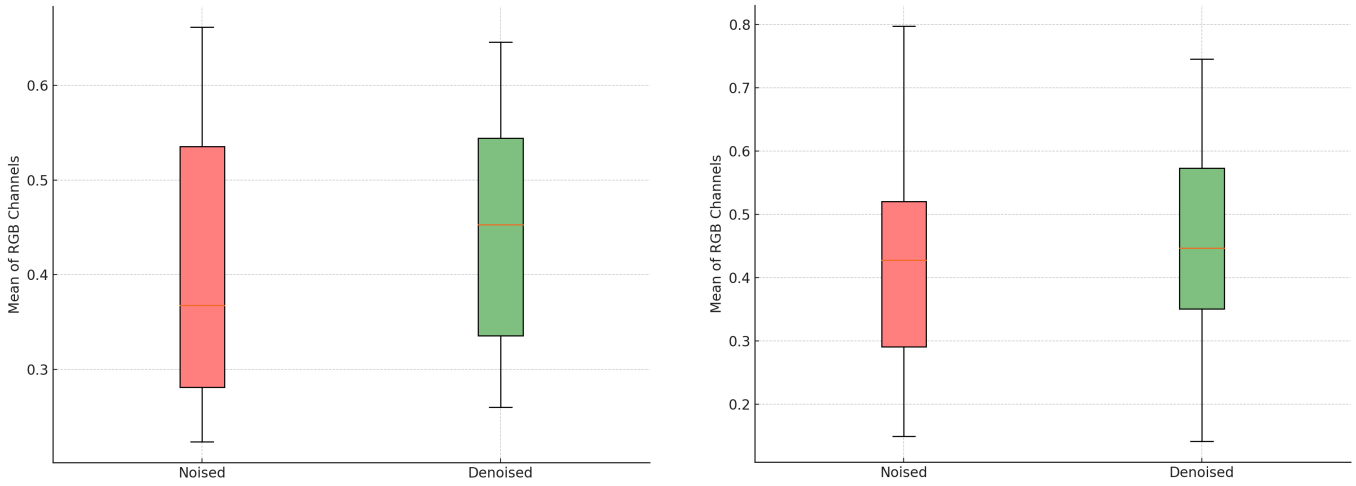
REDNet, which incorporates residual learning, demonstrated substantial improvements over both classical methods and DnCNN, achieving a PSNR of 38.93 dB and an SSIM of 0.9846 on Dataset 1 and a PSNR of 42.59 dB with an SSIM of 0.9954 on Dataset 2. This confirms the effectiveness of skip connections and deep encoder–decoder architectures in preserving image details while reducing noise. However, the proposed R-REDNet model further refines this approach, achieving the highest PSNR and an SSIM values across both datasets. Specifically, R-REDNet attained a PSNR of 44.01 dB and an SSIM of 0.9931 on Dataset 1, outperforming REDNet by +5.08 dB in PSNR and +0.0085 in SSIM. On Dataset 2, the performance gain was even more pronounced with R-REDNet achieving a PSNR of 46.15 dB and an SSIM of 0.9955, representing an improvement of +3.56 dB in PSNR compared to REDNet. These results indicate that the integration of deeper convolutional layers, averaging-based skip connections, and iterative refinement mechanisms significantly enhances the model's ability to suppress noise while preserving fine details.

The superiority of R-REDNet is further highlighted by its performance in Dataset 2, where real-world noise variations pose a greater challenge. Compared to MCWNNM, the best-performing classical method, R-REDNet demonstrated a substantial improvement of +5.5 dB in PSNR on Dataset 1 and +8.44 dB on Dataset 2, along with SSIM gains of +0.026 and +0.041, respectively. These improvements validate the robustness of R-REDNet in handling complex, non-Gaussian noise patterns more effectively than traditional methods. Furthermore, the performance gap between REDNet and R-REDNet underscores the importance of reinforcement mechanisms in CNN architectures, as the proposed enhancements allow for better feature extraction, more stable gradient propagation, and improved image reconstruction.

The experimental findings indicate that R-REDNet achieves superior performance compared to existing denoising methods in both PSNR and SSIM. Its strong generalization across various noise patterns and datasets underscores its suitability for real-world applications requiring reliable image restoration. By incorporating deeper encoder layers, an averaging-based skip connection strategy, and iterative refinement, R-REDNet effectively mitigates the limitations of previous models and achieves SOTA denoising performance.

The statistical boxplots for both Dataset 1 (Figure 9a) and Dataset 2 (Figure 9b) distinctly illustrate substantial differences between noised and denoised images, strongly validating the effectiveness of the applied denoising method. In Dataset 1, the median

(Q2) values for the mean intensities across all RGB channels in denoised images were noticeably higher and more stable compared to those of noised images. This clearly indicates that denoising effectively restores original brightness levels, which were previously obscured by noise. Additionally, the interquartile range (IQR)—defined by the first quartile (Q1) and third quartile (Q3)—was considerably narrower for denoised images across all color channels, signifying a significant reduction in pixel intensity variability. The tighter grouping of minimum and maximum intensity values further emphasizes the efficacy of noise suppression. Moreover, the frequency and magnitude of outliers were substantially reduced post-denoising, highlighting the successful removal of random artifacts.



(**a**) (**b**)

Figure 9. Statistical comparison of pixel intensity distributions between noisy and denoised images. (**a**) Pixel intensity distribution of noisy images of Dataset 1; (**b**) pixel intensity distribution of denoised images of Dataset 2.

The standard deviation and variance metrics reinforce these observations. Across both datasets, median values (Q2) for standard deviation and variance were consistently lower in denoised images for all RGB channels, providing strong quantitative support for denoising efficacy. The narrowed IQR and decreased min–max ranges in denoised images suggest a significant reduction in random pixel intensity fluctuations, contributing to enhanced homogeneity. Conversely, noised images displayed greater dispersion and numerous outliers across all channels, reflecting the presence of significant noise-induced randomness.

Dataset 2 further strengthens these findings, demonstrating even more pronounced improvements. Denoised images exhibited consistently lower median (Q2) values for both standard deviation and variance along with notably narrower IQRs and reduced ranges (min–max). This enhanced consistency across statistical parameters in Dataset 2 highlights the robustness and general applicability of the denoising approach across diverse image scenarios.

7. Conclusions

In this study, we present R-REDNet, an upgraded version of the original REDNet model, tailored specifically for restoring images affected by real-world noise. The model incorporates deeper convolutional layers in the encoder and replaces additive skip connections with averaging operations, enhancing feature extraction and leading to smoother reconstructions. Our empirical evaluations demonstrate that these adjustments significantly improve the model's ability to manage the unpredictable and non-Gaussian nature of real noise commonly found in practical scenarios. REDNet provides a robust solution for image enhancement, effectively balancing noise removal with the preservation of fine details, mak-

ing it ideal for various real-world applications, including low-light photography, medical imaging, and surveillance. However, R-REDNet slightly increases computational complexity and memory requirements, creating a trade-off between resource usage and denoising performance. Although this can slightly elevate real-time processing demands, the model achieves superior image fidelity, making it suitable when denoising quality outweighs efficiency concerns. To address these, future research will focus on model optimization for real-time deployment, adaptive learning techniques, such as reinforcement learning, to enhance generalization, and expanding the dataset to improve robustness across diverse noise patterns. Furthermore, future research could explore extending R-REDNet to video denoising by incorporating temporal consistency mechanisms to effectively capture correlations between consecutive frames. This could involve integrating recurrent connections or spatiotemporal convolutional layers to enhance temporal coherence while preserving fine details. Additionally, hybrid approaches that combine R-REDNet with Transformers or self-attention mechanisms could be investigated. Such models could leverage the global contextual modeling capabilities of attention-based architectures while maintaining the efficiency of R-REDNet, potentially leading to improved performance in both image and video denoising tasks.

Author Contributions: Conceptualization, I.B. and H.K.; methodology, I.B.; software, I.B.; validation, I.B. and H.K.; formal analysis, I.B.; investigation, I.B.; resources, I.B.; data curation, I.B.; writing—original draft preparation, I.B.; writing—review and editing, I.B. and H.K.; visualization, I.B.; supervision, H.K.; project administration, H.K.; funding acquisition, H.K. Both authors contributed equally to this work. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: Data Availability Statement: No new data were created in this study. Data sharing is not applicable to this article.

Conflicts of Interest: The authors declare no conflicts of interest.

Abbreviations

CNN	convolutional neural network
DL	deep learning
DNN	deep neural network
IR	image restoration
PSNR	peak signal-to-noise ratio
REDNet	residual encoder–decoder network
ResNet	residual neural network
R-REDNet	reinforced residual denoising network
SSIM	structural similarity index
SOTA	state-of-the-art

References

1. Tahir, N.U.A.; Zhang, Z.; Asim, M.; Chen, J.; ELAffendi, M. Object detection in autonomous vehicles under adverse weather: A review of traditional and deep learning approaches. *Algorithms* **2024**, *17*, 103. [CrossRef]
2. Kebir, S.T.; Kheddar, H.; Maazouz, M.; Mekaoui, S.; Ferrah, A.; Mazari, R. Smart robot navigation using RGB-D camera. In Proceedings of the 2018 International Conference on Applied Smart Systems (ICASS), Medea, Algeria, 24–25 November 2018; pp. 1–6.
3. González-Sabbagh, S.P.; Robles-Kelly, A. A survey on underwater computer vision. *ACM Comput. Surv.* **2023**, *55*, 268.
4. Himeur, Y.; Al-Maadeed, S.; Kheddar, H.; Al-Maadeed, N.; Abualsaud, K.; Mohamed, A.; Khattab, T. Video surveillance using deep transfer learning and deep domain adaptation: Towards better generalization. *Eng. Appl. Artif. Intell.* **2023**, *119*, 105698.
5. Chouchane, A.; Bessaoudi, M.; Kheddar, H.; Ouamane, A.; Vieira, T.; Hassaballah, M. Multilinear subspace learning for Person Re-Identification based fusion of high order tensor features. *Eng. Appl. Artif. Intell.* **2024**, *128*, 107521.

6. Hemis, M.; Kheddar, H.; Bourouis, S.; Saleem, N. Deep learning techniques for hand vein biometrics: A comprehensive review. *Inf. Fusion* **2024**, *114*, 102716.
7. Boucherit, I.; Zmirli, M.O.; Hentabli, H.; Rosdi, B.A. Finger vein identification using deeply-fused Convolutional Neural Network. *J. King Saud Univ.-Comput. Inf. Sci.* **2022**, *34*, 646–656.
8. Ismail, B.; Mohamed, O.Z. Finger vein image enhancement technique based on Gabor filter and discrete cosine transform. *J. Telecommun. Electron. Comput. Eng. (JTEC)* **2019**, *11*, 43–48.
9. Manohar, B.; Das, R.; Lakshmi, M. A hybridized LSTM-ANN-RSA based deep learning models for prediction of COVID-19 cases in Eastern European countries. *Expert Syst. Appl.* **2024**, *256*, 124977.
10. Gai, S.; Zhang, B.; Yang, C.; Yu, L. Speckle noise reduction in medical ultrasound image using monogenic wavelet and Laplace mixture distribution. *Digit. Signal Process.* **2018**, *72*, 192–207.
11. Sivaranjani, R.; Roomi, S.M.M.; Senthilarasi, M. Speckle noise removal in SAR images using Multi-Objective PSO (MOPSO) algorithm. *Appl. Soft Comput.* **2019**, *76*, 671–681.
12. Vijay, M.; Devi, L.S. Speckle noise reduction in satellite images using spatially adaptive wavelet thresholding. *Int. J. Comput. Sci. Inf. Technol. (IJCSIT)* **2012**, *3*, 3432–3435.
13. Kheddar, H.; Hemis, M.; Himeur, Y.; Megías, D.; Amira, A. Deep learning for steganalysis of diverse data types: A review of methods, taxonomy, challenges and future directions. *Neurocomputing* **2024**, *581*, 127528.
14. Aljarf, A.; Amin, S. Filtering and reconstruction system for gray forensic images. *Int. J. Comput. Electr. Autom. Control Inf. Eng.* **2015**, *9*, 20–25.
15. Wali, A.; Naseer, A.; Tamoor, M.; Gilani, S. Recent progress in digital image restoration techniques: A review. *Digit. Signal Process.* **2023**, *141*, 104187.
16. Huang, T. Stability of two-dimensional recursive filters. *IEEE Trans. Audio Electroacoust.* **1972**, *20*, 158–163.
17. Rajagopal, A.; Hamilton, R.B.; Scalzo, F. Noise reduction in intracranial pressure signal using causal shape manifolds. *Biomed. Signal Process. Control* **2016**, *28*, 19–26.
18. Chiang, Y.W.; Sullivan, B. Multi-frame image restoration using a neural network. In Proceedings of the 32nd Midwest Symposium on Circuits and Systems, Champaign, IL, USA, 14–16 August 1989; pp. 744–747.
19. Ilesanmi, A.E.; Ilesanmi, T.O. Methods for image denoising using convolutional neural network: A review. *Complex Intell. Syst.* **2021**, *7*, 2179–2198.
20. Spagnolo, F.; Corsonello, P.; Frustaci, F.; Perri, S. Design of approximate bilateral filters for image denoising on FPGAs. *IEEE Access* **2023**, *11*, 1990–2000.
21. Mishiba, K. Fast guided median filter. *IEEE Trans. Image Process.* **2023**, *32*, 737–749.
22. Dabov, K.; Foi, A.; Katkovnik, V.; Egiazarian, K. Color image denoising via sparse 3D collaborative filtering with grouping constraint in luminance-chrominance space. In Proceedings of the 2007 IEEE International Conference on Image Processing, San Antonio, TX, USA, 16 September–19 October 2007; Volume 1, pp. I-313–I-316.
23. Mao, X.; Shen, C.; Yang, Y. Image restoration using convolutional auto-encoders with symmetric skip connections. *arXiv* **2016**, arXiv:1606.08921.
24. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.
25. Zhang, K.; Zuo, W.; Zhang, L. FFDNet: Toward a fast and flexible solution for CNN-based image denoising. *IEEE Trans. Image Process.* **2018**, *27*, 4608–4622.
26. Ren, H.; El-Khamy, M.; Lee, J. DN-ResNet: Efficient deep residual network for image denoising. In *Computer Vision–ACCV 2018, Proceedings of the 14th Asian Conference on Computer Vision, Perth, Australia, 2–6 December 2018*; Revised Selected Papers, Part V 14; Springer: Cham, Switzerland, 2019; pp. 215–230.
27. Xue, J.; Wu, Q. An Efficient Self-Refinement and Reconstruction Network for Image Denoising. *Int. J. Inf. Technol. Syst. Approach (IJITSA)* **2023**, *16*, 1–17.
28. Pan, Y.; Ren, C.; Wu, X.; Huang, J.; He, X. Real image denoising via guided residual estimation and noise correction. *IEEE Trans. Circuits Syst. Video Technol.* **2022**, *33*, 1994–2000.
29. Zhou, L.; Zhou, D.; Yang, H.; Yang, S. Two-subnet network for real-world image denoising. *Multimed. Tools Appl.* **2024**, *83*, 14757–14773.
30. Makwana, D.; Deshmukh, G.; Susladkar, O.; Mittal, S.; Teja R, S.C. LIVENet: A novel network for real-world low-light image denoising and enhancement. In Proceedings of the 2024 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV), Waikoloa, HI, USA, 3–8 January 2024; pp. 5856–5865.
31. Ma, R.; Li, S.; Zhang, B.; Li, Z. Generative adaptive convolutions for real-world noisy image denoising. *Proc. AAAI Conf. Artif. Intell.* **2022**, *36*, 1935–1943.
32. Jang, G.; Lee, W.; Son, S.; Lee, K.M. C2N: Practical generative noise modeling for real-world denoising. In Proceedings of the 2021 IEEE CVF International Conference on Computer Vision (ICCV), Montreal, BC, Canada, 11–17 October 2021; pp. 2350–2359.

33. Lin, X.; Ren, C.; Liu, X.; Huang, J.; Lei, Y. Unsupervised image denoising in real-world scenarios via self-collaboration parallel generative adversarial branches. In Proceedings of the 2023 IEEE/CVF International Conference on Computer Vision, Paris, France, 2–6 October 2023; pp. 12642–12652.
34. Jang, Y.I.; Lee, K.; Park, G.Y.; Kim, S.; Cho, N.I. Self-supervised image denoising with downsampled invariance loss and conditional blind-spot network. In Proceedings of the 2023 IEEE/CVF International Conference on Computer Vision, Paris, France, 2–6 October 2023; pp. 12196–12205.
35. Zhang, D.; Zhou, F. Self-supervised image denoising for real-world images with context-aware transformer. *IEEE Access* **2023**, *11*, 14340–14349.
36. Hou, Y.; Xu, J.; Liu, M.; Liu, G.; Liu, L.; Zhu, F.; Shao, L. NLH: A blind pixel-level non-local method for real-world image denoising. *IEEE Trans. Image Process.* **2020**, *29*, 5121–5135.
37. Lee, W.; Son, S.; Lee, K.M. AP-BSN: Self-supervised denoising for real-world images via asymmetric pd and blind-spot network. In Proceedings of the 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), New Orleans, LA, USA, 18–24 June 2022; pp. 17725–17734.
38. Mansour, Y.; Heckel, R. Zero-shot noise2noise: Efficient image denoising without any data. In Proceedings of the 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Vancouver, BC, Canada, 17–24 June 2023; pp. 14018–14027.
39. Ji, K.; Huang, H.; Zhu, S.; Xie, Y. A Novel Approach for Self-Supervised Denoising. In Proceedings of the 2023 International Conference on the Cognitive Computing and Complex Data (ICCD), Huaian, China, 21–22 October 2023; pp. 18–22.
40. Chihaoui, H.; Favaro, P. Masked and Shuffled Blind Spot Denoising for Real-World Images. In Proceedings of the 2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 16–22 June 2024; pp. 3025–3034.
41. Panigrahi, S.K.; Tripathy, S.K.; Bhowmick, A.; Satapathy, S.K.; Barsocchi, P.; Bhoi, A.K. Multi-scale based Approach for Denoising Real-World noisy image using Curvelet Thresholding: Scope and Beyond. *IEEE Access* **2024**, *12*, 25090–25105.
42. Yang, Q.; Yan, P.; Zhang, Y.; Yu, H.; Shi, Y.; Mou, X.; Kalra, M.K.; Zhang, Y.; Sun, L.; Wang, G. Low-dose CT image denoising using a generative adversarial network with Wasserstein distance and perceptual loss. *IEEE Trans. Med. Imaging* **2018**, *37*, 1348–1357. [CrossRef]
43. Kazeminia, S.; Baur, C.; Kuijper, A.; van Ginneken, B.; Navab, N.; Albarqouni, S.; Mukhopadhyay, A. GANs for medical image analysis. *Artif. Intell. Med.* **2020**, *109*, 101938.
44. Xu, J.; Li, H.; Liang, Z.; Zhang, D.; Zhang, L. Real-world noisy image denoising: A new benchmark. *arXiv* **2018**, arXiv:1804.02603.
45. Nam, S.; Hwang, Y.; Matsushita, Y.; Kim, S.J. A holistic approach to cross-channel image noise modeling and its application to image denoising. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 1683–1691.
46. Chen, Y.; Pock, T. Trainable Nonlinear Reaction Diffusion: A Flexible Framework for Fast and Effective Image Restoration. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *39*, 1256–1272. [CrossRef] [PubMed]
47. Lebrun, M.; Colom, M.; Morel, J.M. Multiscale image blind denoising. *IEEE Trans. Image Process.* **2015**, *24*, 3149–3161. [CrossRef] [PubMed]
48. Xu, J.; Zhang, L.; Zhang, D.; Feng, X. Multi-channel Weighted Nuclear Norm Minimization for Real Color Image Denoising. In Proceedings of the 2017 IEEE International Conference on Computer Vision (ICCV), Venice, Italy, 22–29 October 2017; pp. 1105–1113. [CrossRef]
49. ABSoft. Neat Image: Digital Image Noise Reduction Software. 2024. Available online: https://ni.neatvideo.com/home (accessed on 31 December 2024).
50. Zhang, K.; Zuo, W.; Chen, Y.; Meng, D.; Zhang, L. Beyond a gaussian denoiser: Residual learning of deep cnn for image denoising. *IEEE Trans. Image Process.* **2017**, *26*, 3142–3155. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.