electronics

MDPI

Article

A Dynamic Network with Transformer for Image Denoising

Mingjian Song [1,2,3], Wenbo Wang [4] and Yue Zhao [5,*]

[1] School of Software, Northwestern Polytechnical University, Xi'an 710129, China; songmingjian@mail.nwpu.edu.cn
[2] Guangdong Key Laboratory of Intelligent Information Processing & Shenzhen Key Laboratory of Media Security, Shenzhen University, Shenzhen 518060, China
[3] Yangtze River Delta Research Institute, Northwestern Polytechnical University, Taicang 215400, China
[4] School of Computer and Information, Hefei University of Technology, Hefei 230009, China; 2019214607@mail.hfut.edu.cn
[5] School of Computer Science, College of Science, Mathematics and Technology, Wenzhou-Kean University, Wenzhou 325060, China
* Correspondence: yuezhao@kean.edu

Abstract: Deep convolutional neural networks (CNNs) can achieve good performance in image denoising due to their superiority in the extraction of structural information. However, they may ignore the relationships between pixels to limit effects for image denoising. Transformer, focusing on pixel to pixel relationships can effectively solve this problem. This article aims to make a CNN and Transformer complement each other in image denoising. In this study, we propose a dynamic network with Transformer for image denoising (DTNet), with a residual block (RB), a multi-head self-attention block (MSAB), and a multidimensional dynamic enhancement block (MDEB). Firstly, the RB not only utilizes a CNN but also lays the foundation for the combination with Transformer. Then, the MSAB adds positional encoding and applies multi-head self-attention, which enables the preservation of sequential positional information while employing the Transformer to obtain global information. Finally, the MDEB uses dimension enhancement and dynamic convolution to improve the adaptive ability. The experiments show that our DTNet is superior to some existing methods for image denoising.

Keywords: image denoising; transformer; CNN; dynamic convolution

Citation: Song, M.; Wang, W.; Zhao, Y. A Dynamic Network with Transformer for Image Denoising. *Electronics* **2024**, *13*, 1676. https://doi.org/10.3390/electronics13091676

Academic Editor: Manohar Das

Received: 22 March 2024
Revised: 17 April 2024
Accepted: 23 April 2024
Published: 26 April 2024

1. Introduction

Image denoising, as a vital research topic in computer vision, has established an important foundation for other high-level computer vision technologies [1]. The mathematical model of image degradation can be represented by the formula $y = H(x) + n$ [2], where y represents a clean image, H is the degenerate function, and n is additive noise, including Gaussian noise and salt and pepper Noise. According to this model, various models have been exploited for modeling image priors, as follows. Wang et al. proposed a progressive switching median (PSM) filter to remove salt–pepper impulse noise [3]. According to the zero-mean Laplacian random variables with high local correlation, Rebbani et al. proposed a new spatially adaptive denoising method [4]. Dabov et al. enhanced sparse representation by the 3D transformation of 2D image fragments [5]. Image guidance was introduced into image denoising by learning sparse representations of images [6]. However, the quality of the guidance image is an important factor to address the blurring of denoised edges [7]. To solve the blurring problem, the joint image denoising algorithm measures the absolute cosine value of the angle between the gradient vector of the guidance image and the gradient vector of the image to restore them in parallel [7]. Dong et al. presented a variational framework unifying the local image module based on a dictionary of basic functions and a nonlocal image module based on clustering. Although most of these prior methods demonstrate good performance in image denoising, they have the following two disadvantages [8]:

(1) these methods require complex optimization algorithms, causing low denoising efficiency; and (2) they have the problem of requiring parameters to be manually selected.

To overcome the limitations of these prior methods, a large number of end-to-end deep learning methods have been proposed [9], including generative adversarial networks (GANs) [10], encoder–decoder architectures [11], CNN-based methods [6,8,12,13], and Transformer-based methods [14–17]. Through end-to-end deep learning methods, the model can automatically learn feature representations from the data and make final predictions for the task [9]. There is usually a significant difference between the saliency of 3D views and 2D views [10]. To lower this difference, a multi-input multi-output generative adversarial network (MIMO-GAN) is used in computer vision, which is also an effective architecture for image perception [10]. By expanding the encoder–decoder architecture, a reconstructor was introduced to a reconstruction network (RecNet) to reproduce video features after the encoder–decoder structure [11]. This introduces a new method for computer vision architecture. In this study, we will mainly discuss the research on CNNs and Transformer. Zhang et al. used a residual learning strategy to remove latent clean images in the hidden layers [8]. CNNs may encounter situations where shallow information is difficult to transmit to deep layers [6]. Therefore, Tian et al. used a long path to fuse information from both shallow and deep layers [6]. In addition, a memory block includes multi-level representations of the current state and reservations of previous states [18]. The use of an attention mechanism to improve the granularity of extracting information in complex environments is possible [6]. Self-attention is the core idea in Transformer, which can capture global features [19]. Dosovitsky et al. [20] proposed the Vision Transformer model ViT and adopted the Transformer structure for image classification for the first time. DeiT [21] proposed several strategies to train ViT on smaller datasets, achieving better results. The application of Transformer to promote the modeling of image generation sequences has achieved significant results in image generation tasks. Afterwards, more Transformer applications were developed in other image processing fields. Moreover, the use of Transformer to promote the modeling of image generation sequences has achieved significant results in image generation tasks [22]. A texture converter for joint feature learning across reference images and low-resolution images is used for image super-resolution [23]. There are also numerous studies in the field of image denoising. Liu et al. proposed Swin Transformer to limit self-attention to non-overlapping local windows for cross-window connection [15]. Reformer applies self-attention across feature dimensions instead of spatial dimensions, replacing the original ordinary multi-head attention [24]. Wang et al. enabled the network layer to adaptively adjust and employed specific attention mechanisms to serve its U-shaped structure [25]. But both CNN-based methods and Transformer-based methods need large data and huge amounts of computing resources. In addition, they may ignore the relationships between pixels to limit effects for image denoising. Considering the advantages of CNNs in structural feature extraction and Transformer's perception of pixel relationships, we aim to combine these two advantages to improve image denoising performance while overcoming their computational complexity. Therefore, in-depth research and contributions are conducted.

In this study, we present a dynamic network with Transformer for image denoising (DTNet). DTNet utilizes three modules, including a residual block (RB), a multi-head self-attention block (MSAB), and a multidimensional dynamic enhancement block (MDEB). In the RB, we use cutting operations and residual learning to segment images into feature sequences that conform to the Transformer. This not only utilizes a CNN for local feature extraction, but also facilitates the subsequent input of vectors into the Transformer. The MSAB adds positional encoding and applied multi-head self-attention, which preserves sequence order relationships and global features. The MDEB performs dimension enhancement and dynamic convolution to improve adaptability and computational efficiency. DTNet uses pixel relationships to extract salient information. In addition, multidimensional operation fusion using multi-level information can improve the generalization ability of the obtained denoiser. Experiments demonstrate that the proposed DTNet has good performance in image denoising.

The contributions of this paper are as follows.

1. A dynamic convolution is used in a CNN to adaptively learn parameters to improve the robustness of an obtained denoiser, according to the given noisy images.
2. The combination of a CNN and Transformer can extract more structural information and salient information in terms of network architecture and relationships between image pixels.
3. The fusion of multi-level information designed into a CNN can improve the performance of image denoising.

2. Related Work

2.1. Dynamic Convolutions for Image Applications

Common CNNs usually use certain parameters in the test phase to achieve good performance of image applications [26]. However, they cannot adjust parameters to limit the robustness based on different scenes. To address this issue, dynamic convolution [27] was proposed. According to different inputs, it can leverage multiple parallel convolutions in attention mechanisms to adjust weights, aiming to improve the adaptive abilities [27]. To avoid the need to increase the model capacity when improving performance, conditionally parameterized convolutions (CondConv) can obtain different convolution parameters when the input is different. In addition, it parameterizes the convolutional kernel in CondConv into a linear combination of multiple expert knowledge [28]. The final parameters used for the convolution kernel may vary depending on the input [28]. To overcome the previous model's excessive reliance on static conditions, dynamic convolution aggregates multiple parallel convolution kernels based on attention dynamics. Attention dynamically adjusts the weight of each convolution kernel based on the input, thus generating self-adaption dynamic convolutions, including different weight convolution kernels [27]. In line with the input, the final parameters used for the convolution kernel vary [27]. However, these studies only focus on the information of a single one-dimensional convolution kernel, ignoring information including convolution size, and number of input and output channels [29]. To adopt all of the dimensional information, omni-dimensional dynamic convolution (ODConv) utilizes a multidimensional attention mechanism to learn convolutional kernels from four dimensions and applies these attention weights to the corresponding convolutional kernels [30]. To deal with the issue of redundant information between convolutional kernels, dynamic convolution is used in multiple CNN networks to maintain the performance while lowering the costs [30]. Dynamic convolution is designed with coefficient prediction on the basis of image contents and convolution kernels generation [30]. The main concern is the application of dynamic convolution, which involves applying dynamic attention over channel groups after the projection into a higher-dimensional latent space [31]. To tackle this concern, the approach of dynamic channel fusion is used as a replacement for dynamic attention over channel groups [31]. Dynamic channel fusion operates by condensing the input channels into lower dimensions, followed by channel fusion through a matrix to restore them to higher dimensions, which can effectively reduce the dimensionality of the prospective space [31]. The fusion of a CNN and dynamic convolution does not increase the computational complexity, but can effectively improve the image denoising performance [32]. Capitalizing on its outstanding performance, this study integrates dynamic convolution into a CNN for image denoising.

2.2. Transformer for Image Denoising

With the significant achievements of Transformer [19,26] in natural language processing tasks, numerous researchers have started to investigate its application in the field of computer vision. DeiT [21] proposed several strategies to train ViT on smaller datasets and achieve better results. Owing to the presence of the self-attention mechanism, Transformer can model the relationships between pixels at different positions in the image without being constrained by the local receptive fields of traditional convolution operations. In image denoising, this modeling of nonlocal correlations enables a more effective captur-

ing of the global distribution of noise and the interplay between different regions in the image, thereby enhancing the denoising accuracy. As a result, Transformer is gradually gaining attention in research within the field of image denoising. Tian et al. employed a deep layer as the current state to provide guidance for the preceding layer, serving as the previous state, with the purpose of distinguishing between the foreground and background to effectively suppress noise [6]. Nikzad et al. proposed a pyramid dilated lattice to better utilize residuals and dense clustering in feature extraction, and train the model using a new strategy about pyramid dilated convolution [33]. Following the multi-view concept, image denoising incorporates multi-head attention, involving both a single network [25] and multiple networks [34]. Wang et al. utilized multi-head attention within a single network by leveraging diverse inputs to modulate specific layers within a CNN. Moreover, this adaptation adjusts the influence of critical information, which can, thus, enhance the efficiency of image denoising [25]. In addition, they constructed an attention mechanism for U-Transformers to enable learning of biases between decoder layers and the extraction of local contextual features, thereby improving the performance of image prediction [25]. To obtain more comprehensive structural information, the image is rotated in different directions and multiple rotated images are input into a multi-head CNN using a multi-path attention mechanism. This facilitates the integration of complementary salient information within the context of image denoising [34]. Furthermore, Shi et al. utilized the attention mechanisms of two parallel branches in space and spectrum, respectively, and extract features from the fused spatial and spectral information at multiple scales [35]. Based on the aforementioned discussion, it is evident that the adoption of Transformer proves to be a favorable option for the image denoising task.

3. The Proposed Method

3.1. Network Architecture

The designed DTNet includes three parts: head, body, and tail. The head is a residual block (RB); the main body includes a multi-head self-attention block (MSAB) and a multidimensional dynamic enhancement block (MDEB); the tail is composed of a convolutional layer, as shown in Figure 1. The RB performs cutting operations on the original image to extract low-level features, and refines feature extraction based on residual learning operations. The MSAB extracts global structural features by combining the multi-head self-attention mechanism with location information. The MDEB expands its dimensions and adjusts the parameters of multiple-dimensional convolutions via dynamic convolution to enhance the association with neighboring spatial dimensions. The MSAB and MDEB constitute the main body of the encoder. The following equation is used for describing the network process:

$$\begin{aligned} I_C &= f_{Conv}(f_{MDEB}(f_{MSAB}(f_{RB}(I_N)))) \\ &= f_{DTNet}(I_N), \end{aligned} \tag{1}$$

where I_N and I_C denote a noisy image and a clean image, respectively. f_{Conv}, f_{RB}, f_{MSAB}, and f_{MDEB} are functions of Conv, RB, MSAB, and MDEB, respectively. f_{DTNet} indicates a function of DTNet.

In addition, the multi-head self-attentions (MSAs) in the MSAB and MDEB are combined into a basic encoder module, which can be adjusted in quantity to meet the accuracy requirements of different hardware device limitations and tasks.

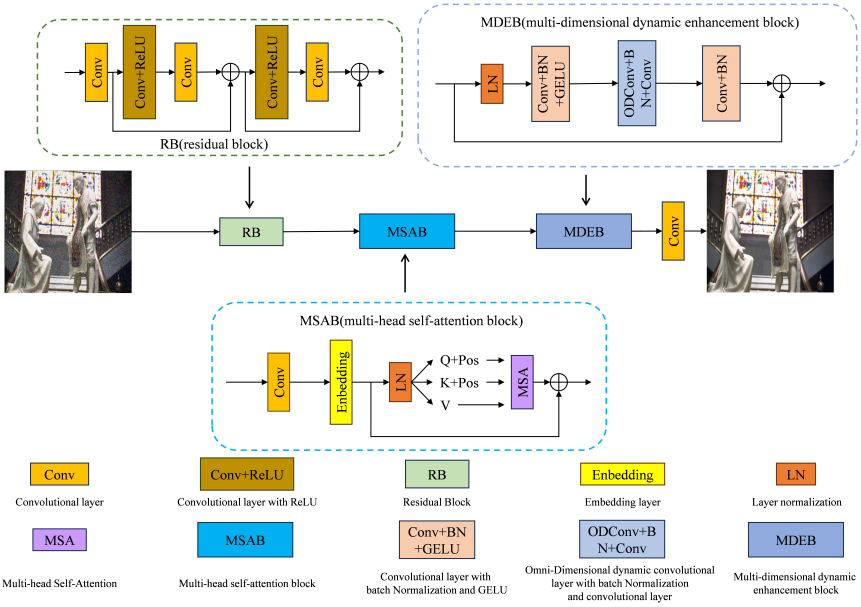

Figure 1. Network architecture of DTNet.

3.2. Loss Function

Mean square error (MSE) was used in numerous popular methods, including a feedforward denoising convolutional neural network (DnCNN) [8] and a fast and flexible denoising convolutional neural network (FFDNet) [13]. However, MSE still has certain limitations, such as a significant difference in image quality between MSE and human perception [36]. This is because, different from the human visual system (HVS), which perceives images in terms of overall structure, MSE is more sensitive to errors at the local pixel level [36]. Nevertheless, due to its ease of computation and differentiability, MSE has advantages over other loss functions in image denoising [37]. To train our DTNet denoiser, MSE is used as our loss function to determine the difference between the denoised images and real images [38]. Specifically, MSE calculates the square difference between the denoised image $DTNet(I_N^i)$ and the real image I_R^i. Then, the square differences of all the images are averaged to obtain the result of the loss function. Based on the value of the loss function, the MSE is minimized to optimize the model. The optimization of DTNet involves utilizing the Adam optimizer to obtain suitable parameter values [39]. The above-mentioned process can be expressed using a formula, as written in Equation (2) [37].

$$MSE = \frac{1}{n}\sum_{i=1}^{n}\left(DTNet\left(I_N^i\right) - I_R^i\right)^2, \tag{2}$$

where MSE stands for the result of the MSE loss function, and I_N^i and I_R^i stand for the i-th noisy image and the i-th real image, respectively. $DTNet(I_N^i)$ represents the clear image after our DTNet denoising.

3.3. Residual Block

The RB includes image segmentation operations and residual learning. To adapt to the specific isometric vector format of Transformer, it is necessary to segment the image and cut the original image into equal-sized image blocks. By assuming an image size of $x \in \mathbb{R}^{3\times H\times W}$, segmentation of each image block size is $P \times P$, so the original image will be cut into HW/P^2 $P \times P$-sized tiles. Since the above direct image segmentation does not make full use of lower-level features in the image, this study employs a convolution operation to further segment the image and gives full play to the advantages of the CNN in extracting lower-level features. Then, residual learning blocks are used to extract image features from the original image and obtain the feature map $x \in \mathbb{R}^{C\times H\times W/P^2\times P\times P}$

(C represents the number of channels after channel expansion). Specifically, the RB consists of a convolutional layer and two residual learning blocks. The residual learning block includes the first convolutional layer, ReLU, and the second convolutional layer. The above process can be represented by the following equations:

$$RL(y) = C(ReLU(C(y))) + y, \tag{3}$$

$$\begin{aligned} O_{RB} &= f_{RB}(I_N) \\ &= RL(RL(C(I_N))), \end{aligned} \tag{4}$$

where RL denotes a function of residual learning, and O_{RB} refers to the output of the RB. C and $ReLU$ stand for a convolutional layer and the ReLU activation function, respectively.

3.4. Multi-Head Self-Attention Block

The MSAB includes the operation of adding positional information to feature maps and MSA. The feature map O_{RB} is transformed into a vector sequence similar to a word sequence, aiming to adapt to the special format of the Transformer. Meanwhile, global modeling in Transformer will lead to loss of image correlation, requiring a learnable position code pos to be added for each vector in the sequence. The resulting encoded vector sequence is input into the subsequent MSA. MSA obtains the encoding representation of the output of this block. q is used for calculating the similarity with the given key k, thereby generating the corresponding attention weights. q and k contain information for comparison, while v contains the actual values corresponding to each query. These vectors are involved in our model to calculate self-attention weights and generate the final output [19]. The MSAB includes a convolutional layer, an embedding, a layer normalization, an operation to add positional information to the q, k, and v vectors, and an MSA. In addition, the input before position coding y performs cross-layer addition using dropout operations.

Positional encoding is obtained by adding an embedding layer. At first, the input sequence is numbered from 0 to the maximum based on its length. The numbered sequence is expanded into one row according to the original input dimension, corresponding to the tensor of the column. This tensor is input into the embedding layer, with the output being positional encoding. During the process of applying positional encoding, the positional encoding is added to the input vector, where the input vector is treated as q, k, and the original input vector is treated as v. Then, the module calculates attention using q, k, and v.

This process can be expressed as Equation (5).

$$\begin{aligned} y = v &= LN(Embedding(Conv(O_{RB}))) \\ q = k &= y + pos \\ O_{MSAB} &= MSA(q, k, v) + y, \end{aligned} \tag{5}$$

where y denotes the output vector sequence of LN, and pos is the position code. q, k, and v represent the queries, keys, and values in the attention mechanism, respectively. MSA and LN stand for multi-head self-attention and layer normalization, respectively. O_{MSAB} is the output of the MSAB.

3.5. Multidimensional Dynamic Enhancement Block

The multi-head attention module learns the global feature representation through equal-dimensional transformations. To enhance the representation, the feedforward neural network module in a pre-trained Image Processing Transformer (IPT) [40] extends the dimensionality by linear transformations and introduces nonlinear transformations based on activation functions. However, the feedforward network module does not focus on the spatial relationships of the images, which is why it requires very large datasets for training. However, spatial relationships between representations that are vital in vision are not taken into consideration. This resulted in the original ViT requiring a large amount of training data to learn the existing inductive biases.

To solve the above problems and combine the advantages of a CNN and Transformer, this section proposes a multidimensional dynamic enhancement module based on dynamic convolution, aiming to replace the original feedforward network part in our baseline IPT. In the encoder, the MSA module retains the capability of modeling global similarity, and the feedforward network is replaced with the MDEB. Compared to the original feedforward network, thanks to dimension expansion, our MDEB better captures spatial relationships in images. Dynamic convolution enables our MDEB to have higher efficiency, also reducing model complexity.

The MDEB first receives the $O_{\mathrm{MSAB}} \in \mathbb{R}^{N\times\ C}$ output from the MSA module, then expands the vector to a higher dimension $x_{up} \in \mathbb{R}^{N\times R\times\lambda}$ by an ascending module, where λ is a scale factor, introducing more powerful expression ability for the subsequent convolutional operations. On the basis of location coding, the vector in the spatial dimension is restored to $x_p^s \in \mathbb{R}^{\sqrt{N}\times\sqrt{N}\times(\rceil\times C)}$ in the image block in order to adapt to proceeding to the two-dimensional convolution kernels. To enhance its correlation with the adjacent space range, the subsequent dynamic convolution layer fits the corresponding dynamic convolution kernel in four dimensions step by step according to the input image block. Finally, the image block flattens the recovery for the $x_{up} \in \mathbb{R}^{N\times R\times\lambda}$ vector sequence, and performs dimension reduction back to the dimension of the input before the MDEB $x \in \mathbb{R}^{N\times\ C}$. The MDEB includes a dimension expansion operation, a dynamic convolution operation and a dimension reduction operation. The dimension expansion operation consists of a convolutional layer, a batch normalization, and a GELU activation function. The dynamic convolution operation includes a dynamic convolution, a batch normalization, and a GELU activation function. The dimension reduction operation consists of a convolutional layer and a batch normalization. The input O_{MSAB} performs cross-layer addition using dropout operations. The increase and decrease in dimensions can be achieved by changing the number of dimensions through the corresponding convolutional layers of the module. The implementation of ODConv depends on computing four types of attention: spatial position attention, input channel attention, output channel attention, and entire convolutional kernel attention. First, the input is compressed into a feature vector through a global average pooling operation along the channel dimension. Then, through a fully connected layer and four head branches, each head branch includes a fully connected layer and a softmax or sigmoid function to generate normalized attention, which is gradually multiplied by the convolution kernel to make the convolution operation different in all the dimensions of spatial position, input channel, output channel, and the convolution kernel [30]. Finally, a convolutional layer is used to obtain output in the tail.

The above process can be represented using the following formula:

$$\begin{aligned} I_C &= Conv(O_{MDEB}) \\ &= Conv(x_{down} + O_{MSAB}) \\ &= Conv(BN(Conv_{down}(x_{OD}) + O_{MSAB}) \\ &= Conv\big(BN\big(Conv_{down}\big(GELU\big(BN\big(ODConv\big(x_{up}\big)\big)\big)\big) + O_{MSAB}\big) \\ &= Conv\big(BN\big(Conv_{down}\big(GELU\big(BN\big(ODConv\big(GELU\big(BN\big(Conv_{up}(O_{MSAB})\big)\big)\big)\big)\big)\big) + O_{MSAB}\big), \end{aligned} \tag{6}$$

where O_{MSAB} is the output of the MSAB and O_{MDEB} is the output of the MDEB. $x_{up} \in \mathbb{R}^{N\times R\times\lambda}$ represents the vector after the dimension expansion operation, while $x_{down} \in \mathbb{R}^{N\times\ C}$ stands for the vector after the dimension reduction operation; x_{OD} refers to the vector after the dynamic convolution operation. $Conv_{up}$ and $Conv_{down}$ indicate the convolutional layer of the dimension expansion operation and convolutional layer of the dimension reduction operation, respectively. BN, $GELU$, and $ODConv$ are batch normalization, the GELU activation function, and dynamic convolution, respectively.

The DTNet proposed in this section retains the attention mechanism, global feature fusion, and generalization ability of Transformers, while utilizing the local receptive field and subspace sampling advantages of a CNN. The experiment in the next section demonstrates that the structure exerts a good effect.

4. Experiments

4.1. Datasets

The datasets used for training consist of synthetic-noise image datasets and real-noise image datasets. The Berkeley segmentation dataset (BSD432) dataset is chosen as the synthetic-noise image training dataset, with 432 color images of different sizes [41]. We choose the fuzzy image database named PolyU, provided by The Hong Kong Polytechnic University, as the real-noise image training dataset, which includes 100 images captured by five different cameras, each with a size of 512 × 512 [42]. All the above-mentioned images are randomly selected for data augmentation using one of the eight methods during training including original image; images rotated at 90, 180, and 270 degrees; vertically flipped images; rotated 90 degrees after vertical flipping; rotated 180 degrees after vertical flipping; and rotated 270 degrees after vertical flipping.

The datasets used for testing consist of synthetic-noise image datasets. Some public datasets are chosen as the synthetic-noise image datasets, including the Berkeley Segmentation Dataset (BSD68) [43], Set12 [43], and Color Berkeley Segmentation Dataset (CBSD68) [43], to test our model in color- and gray-image denoising [17]. BSD68 is a part of the image segmentation database provided by the University of California, Berkeley, which includes a total of 68 images. Set12 is a part of the Super Resolution Benchmark Dataset, which includes 12 images.

4.2. Experimental Settings

To train our model in this study, we adopt an initial learning rate of 1×10^{-4} and employ a learning rate decay strategy to optimize network training in subsequent stages, with 126 training epochs. The experiments are carried out with the development environment of Ubuntu18.04 as the operating system, Python 3.9 as the development language version, and Pytorch1.11 as the deep learning framework. In terms of hardware configuration, we use the 24 GB memory GPU of model NVIDIA GeForce RTX3090, the 28-core CPU of model Intel Xeon Gold 6330 CPU @ 2.00 GHz, and 93 GB RAM. Nvidia driver version 470.141.03 and CUDA 11.4 are used for the training acceleration.

4.3. Experimental Results and Analysis

This section evaluates the noise reduction performance of DTNet with both qualitative and quantitative analyses. A qualitative analysis is performed by visual comparison of different methods of noise reduction effects. The quantitative analysis employs evaluation indicators such as PSNR [25], running time, and complexity to be compared with algorithms with good results, such as block-matching and 3D filtering (BM3D) [5], DnCNN [8], and attention-guided denoising convolutional neural network (ADNet) [44].

In terms of the qualitative analysis, this study uses some pictures from CBSD68 and Set12 to visually show the comparison of the noise reduction effects of different methods. Through the enlarged observation of the image after denoising, the excellent denoising effect should have a high coincidence degree with the original image area, and be visually clearer and more in line with human cognitive feelings.

The denoising results of the noisy gray image are shown in Figure 2. In Figure 3, we can observe the denoising results of the noisy color image. The proposed DTNet model yields a clearer image compared to the current best denoising methods (such as DnCNN [8], ADNet [44], IPT [40]).

In terms of quantitative analysis, the PSNR index is used to evaluate the denoising quality of the proposed DTNet model, and compared with current classic algorithms and the best-performing algorithms, including BM3D [5], weighted nuclear norm minimization (WNNM) [45], trainable nonlinear reaction diffusion (TNRD) [46], DnCNN [8], image restoration CNN (IRCNN) [47], single-stage blind real-image denoising network (RIDNet) [12], FFDNet [13], ADNet [44], graph convolution image denoising network (GCDN) [48], basis learning network (NBNet) [49], enhanced convolutional neural denoising network (ECNDNet) [50], and IPT [40]. In general, artificial-noise images, including gray and color noisy images, were tested with additive Gaussian white noise levels of 15,

25, and 50. For IPT, to ensure fairness, metrics were measured under the same hardware configuration and environment. For BM3D, WNNM, TNRD, DnCNN, RIDNet, FFDNet, ADNet, GCDN, NBNet, and ECNDNet, we used the data from the relevant papers.

Figure 2. Results of different denoising methods on one image from Set12 when $\sigma = 25$.

Figure 3. Results of different denoising methods on one image from CBSD68 when $\sigma = 25$.

For gray noisy images, experiments were conducted on BSD68 and Set12. The results are shown in Tables 1 and 2. Based on the results of BSD68 shown in Table 1, DTNet performed the best at noise levels of 15, 25, and 50. When the noise level is 15, our DTNet is 0.05 dB higher than the baseline IPT's PSNR. However, when the noise level is 25, our DTNet is 0.03 dB higher than IPT. As the noise level is increased to 50, our DTNet still exhibits a 0.02 dB advantage. For image denoising methods, an improvement in PSNR is extremely rare [1]. Therefore, our DTNet's ability to improve at all three noise levels exhibits the effectiveness of our method. Based on experiments on Set12, presented in Table 2, with a noise level of 15 our DTNet performs better than previous methods in denoising most images. For example, in the Monarch image, our PSNR has an increase of 0.07 when compared with the IPT. Nevertheless, on a few images, such as peppers, the performance is poor. This can be attributed to the unique characteristics and structure of peppers. However, it needs to be emphasized that our model has made progress as a whole. This also shows that the DTNet model presented in this section remains stable regarding characteristic noise level and blind denoising.

Table 1. Average PSNR (dB) of different methods on BSD68. Red represents the best effect, blue represents the second best effect.

Method	σ = 15	σ = 25	σ = 50
BM3D	31.07	28.57	25.62
WNNM	31.37	28.63	25.87
TNRD	31.42	28.92	25.97
DnCNN	31.72	29.23	26.23
IRCNN	31.63	29.19	26.29
FFDNet	31.63	29.29	26.25
ADNet	31.74	29.25	26.29
RIDNet	31.81	29.34	26.40
GCDN	31.83	29.35	26.38
IPT	31.90	29.43	26.47
DTNet (ours)	31.95	29.46	26.49

Table 2. Average PSNR (dB) of different methods on Set12 with noise level of 15. Red represents the best effect, blue represents the second best effect.

Method	BM3D	WNNM	TNRD	DnCNN	FFDNet	ECNDNet	ADNet	IPT	DTNet (Ours)
C.man	31.91	32.17	32.19	32.61	32.43	32.56	32.81	32.64	32.73
House	34.93	35.13	34.53	34.97	35.07	34.97	35.22	35.29	35.31
Peppers	32.69	32.99	33.04	33.30	33.25	33.25	33.49	33.23	33.22
Starfish	31.14	31.82	31.75	32.20	31.99	32.17	32.17	32.54	32.52
Monarch	31.85	32.71	32.56	33.09	32.66	33.11	33.17	33.40	33.47
Airplane	31.07	31.39	31.46	31.70	31.57	31.70	31.86	31.85	31.93
Parrot	31.37	31.62	31.63	31.83	31.81	31.82	31.96	32.02	32.03
Lena	34.26	34.27	34.24	34.62	34.62	34.52	34.71	34.75	34.76
Barbara	33.10	33.60	32.13	32.64	32.54	32.41	32.80	33.22	33.27
Boat	32.13	32.37	32.14	32.42	32.38	32.37	32.57	32.59	32.64
Man	31.90	32.11	32.23	32.46	32.41	32.39	32.47	32.60	32.59
Couple	32.10	32.17	32.11	32.47	32.46	32.39	32.58	32.60	32.63
Average	32.37	32.70	32.50	32.86	32.77	32.81	32.98	33.07	33.10

For noisy color images, the currently popular denoising method of CBSD68 is chosen. Based on Table 3, it can be seen that DTNet achieves good results on artificial noise color images. At noise levels of 15, 25, and 50, our DTNet improves the PSNR by 0.06, 0.03, and 0.01 relative to IPT, respectively. Clearly, our method also performs well for color images. With the increasing noise level, the degree of improvement in the PSNR index gradually decreases, suggesting that the difficulty of denoising is gradually increasing. Perhaps DTNet has certain limitations in high-level noise. Our model's denoising ability at high noise levels may still require further research.

Table 3. Average PSNR (dB) of different methods on CBSD68. Red represents the best effect, blue represents the second best effect.

Method	σ = 15	σ = 25	σ = 50
BM3D	33.52	30.71	28.89
DnCNN	33.98	31.31	29.65
IRCNN	33.86	31.16	27.86
FFDNet	33.80	31.18	29.57
ADNet	33.99	31.31	29.66
NBNet	34.15	31.54	28.31
IPT	34.23	31.57	29.93
DTNet (ours)	34.29	31.60	29.94

In terms of running time, four denoising methods were selected to be compared with our DTNet for experiments on noisy images of different sizes: 256 × 256, 512 × 512, and 1024 × 1024. The results are shown in Table 4.

Table 4. Runtime(s) of different methods on images of different sizes.

Method	Equipment	256 × 256	512 ×512	1024 × 1024
CBM3D	GPU	0.59	2.52	10.77
DnCNN	GPU	0.0344	0.0681	0.1556
ADNet	GPU	0.0467	0.0798	0.2077
IPT	GPU	1.2323	1.8047	3.0117
DTNet (ours)	GPU	0.55933	0.7862	1.3295

In terms of computational costs, runtime, parameters, and FLOPS are used as metrics to reveal the effectiveness of our model modules. In Table 4, we select different methods for experiments to compare the running time of denoising noisy images of different sizes with our DTNet, including 256 × 256, 512 × 512, and 1024 × 1024. It can be observed that our method significantly reduces runtime compared to IPT, proving the effect of ODConv applied in our MDEB module. Relative to IPT, our model has made significant improvements in both parameters and FLOPS, as seen in Table 5. As presented in Tables 4 and 5, our method has an improved parameter count compared to DnCNN and ADNet, while it also shows improvements in FLOPS. Compared to defects with larger parameter quantities, ODCOnv performs more calculations to better capture features, exhibiting greater flexibility. These advantages are greater.

Table 5. Complexity of different methods.

Method	Parameters	FLOPS
DnCNN	0.55 M	1.39 G
ADNet	0.52 M	1.29 G
IPT	35 M	0.95 G
DTNet (ours)	12 M	4.61 G

For an ablation experiment, CBSD68 is used as the baseline dataset with a Gaussian noise level of 25. Considering the impact of the main structure of the network on the overall performance, the multidimensional enhancement module and dynamic convolutional module are gradually subtracted. The experimental results are shown in Table 6. A single dynamic convolution has a significant improvement, and the applied multidimensional enhancement module further enhances the performance, which is caused by its advantage in multidimensional feature extraction. Moreover, the ablation experiment further demonstrates the feasibility of the dynamic convolution and multidimensional enhancement modules.

Table 6. PSNR (dB) results of some networks on CBSD68 with σ = 25.

Method	PSNR (dB)
DTNet without dynamic convolutional module	31.11
DTNet without multidimensional enhancement module	31.57
DTNet (ours)	31.60

5. Conclusions

To conclude, in this study, a dynamic network with Transformer is proposed for image denoising (DTNet). Our DTNet utilizes three modules, including a residual block (RB), a multi-head self-attention block (MSAB), and a multidimensional dynamic enhancement block (MDEB). In the first block (RB) of DTNet, we continuously employ cutting operations

and residual learning to segment the image into a sequence conforming to the Transformer and extract features using the CNN. This not only utilizes the CNN but also lays the foundation for the combination with Transformer. The second block (MSAB) adds positional encoding and applies multi-head self-attention, which can enable the preservation of sequential positional information while utilizing the transformer to obtain global information. In the third block (MDEB), images undergo dimension enhancement and dynamic convolution to better represent image relationships. To improve the adaptive ability, a dynamic convolution is used to learn adaptive parameters to enhance the robustness of a denoiser. Dynamic convolution can effectively reduce the complexity and facilitate computation. DTNet uses relations of pixels to extract salient information. In addition, using multi-dimensional operation fusion of multi-level information can improve the generalization ability of an obtained denoiser. The experimental work demonstrates that our DTNet is effective for image denoising. In future studies, we will further explore other ways to improve the performance of DTNet. It may be essential to continue to explore the effects of combining and modifying existing blocks with other architectures. Then, we will continue to expand the denoising range, generalization ability for different datasets, and real-world applications. Finally, this study hopes to extend DTNet to other related image processing tasks, such as image enhancement, and image super-resolution. Moreover, this may have the potential to advance the technological development in the field of image denoising.

Author Contributions: Conceptualization, M.S.; methodology, M.S.; software, M.S.; validation, W.W.; formal analysis, M.S.; investigation, M.S.; resources, M.S.; data curation, M.S.; writing—original draft preparation, M.S.; writing—review and editing, M.S.; visualization, M.S.; supervision, Y.Z.; project administration, M.S.; funding acquisition, Y.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported in part by the National Natural Science Foundation of China, under Grant 62072166; in part by Leading Talents in Gusu Innovation and Entrepreneurship, Grant ZXL2023170; in part by the Guangdong Key Laboratory of Intelligent Information Processing under Grant 2023B1212060076; in part by the TCL Science and Technology Innovation Fund under Grant D5140240118; in part by the 2023 College Student Innovation and Entrepreneurship Training Program Project of China, under Grant 202310699159; and in part by the Hunan Provincial Natural Science Foundation of China, under Grant 2023JJ30169.

Data Availability Statement: Data are contained within the article.

Conflicts of Interest: The authors declare no conflicts of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript; or in the decision to publish the results.

References

1. Tian, C.; Fei, L.; Zheng, W.; Xu, Y.; Zuo, W.; Lin, C.-W. Deep learning on image denoising: An overview. *Neural Netw.* **2020**, *131*, 251–275. [CrossRef] [PubMed]
2. Levin, A.; Nadler, B. Natural image denoising: Optimality and inherent bounds. In Proceedings of the CVPR 2011, Colorado Springs, CO, USA, 20–25 June 2011; pp. 2833–2840.
3. Wang, Z.; Zhang, D. Progressive switching median filter for the removal of impulse noise from highly corrupted images. *IEEE Trans. Circuits Syst. II Analog Digit. Signal Process.* **1999**, *46*, 78–80. [CrossRef]
4. Rabbani, H. Image denoising in steerable pyramid domain based on a local Laplace prior. *Pattern Recognit.* **2009**, *42*, 2181–2193. [CrossRef]
5. Dabov, K.; Foi, A.; Katkovnik, V.; Egiazarian, K. Image denoising by sparse 3-D transform-domain collaborative filtering. *IEEE Trans. Image Process.* **2007**, *16*, 2080–2095. [CrossRef] [PubMed]
6. Elad, M.; Aharon, M. Image denoising via sparse and redundant representations over learned dictionaries. *IEEE Trans. Image Process.* **2006**, *15*, 3736–3745. [CrossRef] [PubMed]
7. Li, P.; Liang, J.; Zhang, M.; Fan, W.; Yu, G. Joint image denoising with gradient direction and edge-preserving regularization. *Pattern Recognit.* **2022**, *125*, 108506. [CrossRef]
8. 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]
9. Tian, C.; Zheng, M.; Li, B.; Zhang, Y.; Zhang, S.; Zhang, D. Perceptive self-supervised learning network for noisy image watermark removal. *arXiv* **2024**, arXiv:2403.02211.

10. Song, R.; Zhang, W.; Zhao, Y.; Liu, Y.; Rosin, P.; Intelligence, M. 3D Visual saliency: An independent perceptual measure or a derivative of 2d image saliency? *IEEE Trans. Pattern Anal. Mach. Intell.* **2023**, *45*, 13083–13099. [CrossRef]
11. Zhang, W.; Wang, B.; Ma, L.; Liu, W. Reconstruct and represent video contents for captioning via reinforcement learning. *IEEE Trans. Pattern Anal. Mach. Intell.* **2019**, *42*, 3088–3101. [CrossRef]
12. Anwar, S.; Barnes, N. Real image denoising with feature attention. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Republic of Korea, 27–28 October 2019; pp. 3155–3164.
13. 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. [CrossRef] [PubMed]
14. Zamir, S.W.; Arora, A.; Khan, S.; Hayat, M.; Khan, F.S.; Yang, M.-H. Restormer: Efficient transformer for high-resolution image restoration. In Proceedings of the IEEE/CVF conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 18–24 June 2022; pp. 5728–5739.
15. Liu, Z.; Lin, Y.; Cao, Y.; Hu, H.; Wei, Y.; Zhang, Z.; Lin, S.; Guo, B. Swin transformer: Hierarchical vision transformer using shifted windows. In Proceedings of the IEEE/CVF international Conference on Computer Vision, Montreal, BC, Canada, 11–17 October 2021; pp. 10012–10022.
16. Fan, C.-M.; Liu, T.-J.; Liu, K.-H. SUNet: Swin transformer UNet for image denoising. In Proceedings of the 2022 IEEE International Symposium on Circuits and Systems (ISCAS), Austin, TX, USA, 27 May–1 June 2022; pp. 2333–2337.
17. Liang, J.; Cao, J.; Sun, G.; Zhang, K.; Van Gool, L.; Timofte, R. Swinir: Image restoration using swin transformer. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, BC, Canada, 11–17 October 2021; pp. 1833–1844.
18. Tai, Y.; Yang, J.; Liu, X.; Xu, C. Memnet: A persistent memory network for image restoration. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 4539–4547.
19. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, Ł.; Polosukhin, I. Attention is all you need. In Proceedings of the Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, Long Beach, CA, USA, 4–9 December 2017.
20. Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S.; et al. An image is worth 16 $\times$ 16 words: Transformers for image recognition at scale. *arXiv* **2020**, arXiv:2010.11929.
21. Touvron, H.; Cord, M.; Douze, M.; Massa, F.; Sablayrolles, A.; Jégou, H. Training data-efficient image transformers & distillation through attention. In Proceedings of the International Conference on Machine Learning, Virtual Event, 18–24 July 2021; pp. 10347–10357.
22. Parmar, N.; Vaswani, A.; Uszkoreit, J.; Kaiser, L.; Shazeer, N.; Ku, A.; Tran, D. Image transformer. In Proceedings of the International Conference on Machine Learning, Stockholm, Sweden, 10–15 July 2018; pp. 4055–4064.
23. Yang, F.; Yang, H.; Fu, J.; Lu, H.; Guo, B. Learning texture transformer network for image super-resolution. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 14–19 June 2020; pp. 5791–5800.
24. Chung, J.; Gulcehre, C.; Cho, K.; Bengio, Y. Empirical evaluation of gated recurrent neural networks on sequence modeling. *arXiv* **2014**, arXiv:1412.3555.
25. Wang, Z.; Cun, X.; Bao, J.; Zhou, W.; Liu, J.; Li, H. Uformer: A general u-shaped transformer for image restoration. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 18–24 June 2022; pp. 17683–17693.
26. Tian, C.; Xu, Y.; Zuo, W.; Lin, C.-W.; Zhang, D. Asymmetric CNN for image superresolution. *IEEE Trans. Syst. Man Cybern. Syst. Humans* **2021**, *52*, 3718–3730. [CrossRef]
27. Chen, Y.; Dai, X.; Liu, M.; Chen, D.; Yuan, L.; Liu, Z. Dynamic convolution: Attention over convolution kernels. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 14–19 June 2020; pp. 11030–11039.
28. Yang, B.; Bender, G.; Le, Q.V.; Ngiam, J. Condconv: Conditionally parameterized convolutions for efficient inference. In Proceedings of the Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, Vancouver, BC, Canada, 8–14 December 2019.
29. Zhong, X.; Qin, J.; Guo, M.; Zuo, W.; Lu, W. Offset-decoupled deformable convolution for efficient crowd counting. *Sci. Rep.-UK* **2022**, *12*, 12229. [CrossRef] [PubMed]
30. Li, C.; Zhou, A.; Yao, A. Omni-dimensional dynamic convolution. *arXiv* **2022**, arXiv:2209.07947.
31. Li, Y.; Chen, Y.; Dai, X.; Liu, M.; Chen, D.; Yu, Y.; Yuan, L.; Liu, Z.; Chen, M.; Vasconcelos, N. Revisiting dynamic convolution via matrix decomposition. *arXiv* **2021**, arXiv:2103.08756.
32. Tian, C.; Zheng, M.; Zuo, W.; Zhang, B.; Zhang, Y.; Zhang, D. Multi-stage image denoising with the wavelet transform. *Pattern Recognit.* **2023**, *134*, 109050. [CrossRef]
33. Nikzad, M.; Gao, Y.; Zhou, J. Attention-based Pyramid Dilated Lattice Network for Blind Image Denoising. In Proceedings of the IJCAI, Montreal, QC, Canada, 19–27 August 2021; pp. 931–937.
34. Zhang, J.; Qu, M.; Wang, Y.; Cao, L. A Multi-Head Convolutional Neural Network with Multi-Path Attention Improves Image Denoising. In Proceedings of the Pacific Rim International Conference on Artificial Intelligence, Shanghai, China, 10–13 November 2022; pp. 338–351.
35. Shi, Q.; Tang, X.; Yang, T.; Liu, R.; Zhang, L. Hyperspectral image denoising using a 3-D attention denoising network. *IEEE Trans. Geosci. Remote Sens.* **2021**, *59*, 10348–10363. [CrossRef]
36. Wang, Z.; Bovik, A.C.; Sheikh, H.R.; Simoncelli, E. Image quality assessment: From error visibility to structural similarity. *IEEE Trans. Image Process.* **2004**, *13*, 600–612. [CrossRef]

37. Wang, Z.; Bovik, A. Mean squared error: Love it or leave it? A new look at signal fidelity measures. *IEEE Signal Process. Mag.* **2009**, *26*, 98–117. [CrossRef]
38. Allen, D.M.J.T. Mean square error of prediction as a criterion for selecting variables. *Technometrics* **1971**, *13*, 469–475. [CrossRef]
39. Kingma, D.P.; Ba, J. Adam: A method for stochastic optimization. *arXiv* **2014**, arXiv:1412.6980.
40. Chen, H.; Wang, Y.; Guo, T.; Xu, C.; Deng, Y.; Liu, Z.; Ma, S.; Xu, C.; Xu, C.; Gao, W. Pre-trained image processing transformer. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Montreal, BC, Canada, 11–17 October 2021; pp. 12299–12310.
41. Martin, D.; Fowlkes, C.; Tal, D.; Malik, J. A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In Proceedings of the Eighth IEEE International Conference on Computer Vision, ICCV 2001, Vancouver, BC, Canada, 7–14 July 2001; pp. 416–423.
42. Xu, J.; Li, H.; Liang, Z.; Zhang, D.; Zhang, L. Real-world noisy image denoising: A new benchmark. *arXiv* **2018**, arXiv:1804.02603.
43. Roth, S.; Black, M.J. Fields of experts: A framework for learning image priors. In Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), San Diego, CA, USA, 20–25 June 2005; pp. 860–867.
44. Tian, C.; Xu, Y.; Li, Z.; Zuo, W.; Fei, L.; Liu, H. Attention-guided CNN for image denoising. *Neural Netw.* **2020**, *124*, 117–129. [CrossRef]
45. Hochreiter, S.; Schmidhuber, J. Long short-term memory. *Neural Comput.* **1997**, *9*, 1735–1780. [CrossRef]
46. Chen, Y.; Pock, T. Trainable nonlinear reaction diffusion: A flexible framework for fast and effective image restoration. *IEEE Trans. Pattern Anal. Mach. Intell.* **2016**, *39*, 1256–1272. [CrossRef]
47. Zhang, K.; Zuo, W.; Gu, S.; Zhang, L. Learning deep CNN denoiser prior for image restoration. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 3929–3938.
48. Valsesia, D.; Fracastoro, G.; Magli, E. Deep graph-convolutional image denoising. *IEEE Trans. Image Process.* **2020**, *29*, 8226–8237. [CrossRef]
49. Cheng, S.; Wang, Y.; Huang, H.; Liu, D.; Fan, H.; Liu, S. Nbnet: Noise basis learning for image denoising with subspace projection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Montreal, BC, Canada, 11–17 October 2021; pp. 4896–4906.
50. Tian, C.; Xu, Y.; Fei, L.; Wang, J.; Wen, J.; Luo, N. Enhanced CNN for image denoising. *Caai Trans. Intell. Technol.* **2019**, *4*, 17–23. [CrossRef]