applied sciences

MDPI

Article

Image Dehazing Algorithm Based on Deep Learning Coupled Local and Global Features

Shuping Li [1,*], Qianhao Yuan [1,*], Yeming Zhang [1,2,3], Baozhan Lv [1] and Feng Wei [1]

[1] School of Mechanical and Power Engineering, Henan Polytechnic University, Jiaozuo 454000, China
[2] State Key Laboratory of Fluid Power and Mechatronic Systems, Zhejiang University, Hangzhou 310027, China
[3] Beijing Key Laboratory of Pneumatic Thermal Energy Storage and Energy Supply Technology, Beijing 100191, China
* Correspondence: lishuping@hpu.edu.cn (S.L.); yuanqianhaotech@163.com (Q.Y.)

Abstract: To address the problems that most convolutional neural network-based image defogging algorithm models capture incomplete global feature information and incomplete defogging, this paper proposes an end-to-end convolutional neural network and vision transformer hybrid image defogging algorithm. First, the shallow features of the haze image were extracted by a preprocessing module. Then, a symmetric network structure including a convolutional neural network (CNN) branch and a vision transformer branch was used to capture the local features and global features of the haze image, respectively. The mixed features were fused using convolutional layers to cover the global representation while retaining the local features. Finally, the features obtained by the encoder and decoder were fused to obtain richer feature information. The experimental results show that the proposed defogging algorithm achieved better defogging results in both the uniform and non-uniform haze datasets, solves the problems of dark and distorted colors after image defogging, and the recovered images are more natural for detail processing.

Keywords: image dehazing; convolutional neural network; vision transformer; hybrid feature fusion

Citation: Li, S.; Yuan, Q.; Zhang, Y.; Lv, B.; Wei, F. Image Dehazing Algorithm Based on Deep Learning Coupled Local and Global Features. *Appl. Sci.* **2022**, *12*, 8552. https://doi.org/10.3390/app12178552

Academic Editors: Xinwei Yao, Yougang Sun and Xiaogang Jin

Received: 24 July 2022
Accepted: 24 August 2022
Published: 26 August 2022

1. Introduction

Fog is a relatively common atmospheric phenomenon, where the light in the process of transmission, vulnerable to large amounts of water vapor condensation in the air of smoke dust aerosols such as the absorption and dispersion, leads to the observed object reflected light attenuation in reaching the imaging device, and the image contrast shows low resolution, fuzzy white, color saturation, and decline [1]. At the same time, the observed targets tend to lose a lot of important details in foggy scenes, which seriously affect the effect of subsequent visual tasks such as target resolution and capturing details. Therefore, it is particularly important to study effective fogging methods and enhance and restore details of atmospheric degradation in images.

1.1. Traditional fog Removal Methods

At present, traditional defogging methods are mainly divided into the following two categories. The first category is the image restoration method, based on the non-physical model. This kind of method mainly achieves the purpose of defogging by enhancing the contrast and saturation of the image, but it cannot solve the problem of image defogging well because it does not analyze the essential cause of the image degradation in a foggy environment from the root. The main algorithms based on this method include histogram equalization [2], Retinex theory [3], the wavelet decomposition transformation method [4], etc. The second type is the image restoration method based on the physical model. This type of method deeply investigates the objective cause of image degradation, establishes the mathematical formula according to the atmospheric scattering model, and uses it as the theoretical basis to recover the fog-free image from the haze map. For example, He et al. [5]

proposed the dark channel prior defogging algorithm (DCP), which solved the atmospheric scattering model based on the prior knowledge that foggy images have a channel pixel value with a low value. This method is simple and effective, but for images containing large bright white areas such as snow, sky, and white buildings, the effect of defogging is not ideal, and the recovered image will produce serious color distortion. Subsequently, KeYan Wang et al. [6] used a sky segmentation algorithm to de-fog the sky region to a certain extent to avoid the distortion of the sky region. Meng et al. [7] proposed a defogging algorithm based on boundary constraint and regularization of the transmittance map, which solved the problem of low brightness of the defogging image, but color distortion still occurred in some areas of the defogging image. Although the above traditional image defogging methods have made great progress and shown good results, most of them rely on various prior information. Since these a priori and assumptions are introduced for specific scenes, the quality of the fog-free images obtained when the a priori does not hold is poor, and the features need to be extracted manually, which still has great limitations.

1.2. Defogging Method Based on Convolutional Neural Network

In recent years, with the rapid development of machine learning, particularly deep learning, and its superior performance in the field of image processing, more and more scholars are using convolutional neural networks to deal with image defogging problems. The earliest deep learning fog removal network is the DehazeNet network proposed by Cai et al. [8], which chunked the complete image as the input to the network, used multi-scale convolutional neural networks and MaxPooling to learn the haze features and estimate the transmittance map of the fogged image, and then inverse performed the fog-free image. The method achieved a good defogging effect, but because the dataset was a local image after cut, the features learned by the network lacked global representation, and the method did not fuse deep and shallow information, the defogged image showed incomplete defogging, inaccurate color reproduction, etc. Ren et al. [9] proposed the MSCNN network, which takes the whole image as the input and first estimates the scene transmittance map using a coarse-scale network, and then refines it using a fine-scale network to obtain the fine transmittance map, which improved the accuracy of transmittance, but the method used the pooling layer to lose the detailed information, and only estimated the transmittance map, which is not accurate for atmospheric light estimation and affects the defogging effect. The above method only uses neural networks to learn the haze features to estimate the transmittance, but for the estimation of the atmospheric light value, another important parameter in image defogging, the a priori-based method, is still used, resulting in image defects such as color distortion in the recovered image due to the estimation error of the atmospheric light value.

1.3. End-to-End Deep Learning Defogging Approach

In order to solve the problems of the inaccurate estimation of atmospheric light values, we conducted the separate estimation of the haze image transmittance maps and the cumulative effect of errors brought by atmospheric light. Recently, many scholars have used convolutional neural networks to simultaneously estimate the atmospheric light and transmittance of the haze images and directly output the defogged images, thus realizing an "end-to-end" deep learning defogging method that obtains defogging maps from the fog maps in one step [10]. Li et al. [11] proposed the AOD-Net network, which assumes the atmospheric light value and transmittance as a parameter K, and then uses a large number of cascaded convolutional layers and a neural network constructed across the hierarchy to estimate the K value, thus completing end-to-end image defogging. The method improved the quality of the defogged images while avoiding the accumulated errors in the recovered images, but the network model was limited to a shallow structure and failed to learn the features of the fogged images well, which affected the quality of the recovered images. CHEN et al. [12] proposed an end-to-end gated contextual aggregation network to generate clear and fog-free images directly. The network utilized a smooth

expansion technique that eliminates grid artifacts and fuses different levels of features through gated subnetworks. The input fogged image in this network was encoded into a feature map by an encoder, and the image was enhanced by aggregating information from nearby regions and fusing different levels of features. However, the current end-to-end defogging method based on convolutional neural networks still has some limitations, although it is free from the constraints of atmospheric scattering models. After the action of shallower convolutional layers, each pixel point of the feature image is only a feature extraction of the local information of the original image, the size of the convolutional kernel determines the scope of the local weighting of the image, and the image is rich in detailed information, but the image has little contextual information. Convolutional operations are good at extracting local features, but it is difficult to capture the global representation, and it is crucial to obtain a global perception when performing image recovery. To alleviate these limitations, it is necessary to obtain a larger field of perception, and the classical way to increase the field of perception is to downsample the image or feature map to obtain multi-scale information. However, downsampling may lose some useful detailed information that cannot be recovered by upsampling. The use of larger convolutional kernels or more convolutional layers can also increase the perceptual field, but at the same time, the computational effort increases significantly. Dilated convolution can expand the perceptual field without increasing the computational cost, but as the dilation rate increases, the information of neighboring elements of the convolution kernel varies widely, leading to grid artifacts in the defogged results. Attentional mechanisms can quickly capture long-range dependencies by calculating the relationship between two locations in the channel and space to obtain a larger receptive field. Momenta Hu Jie's team proposed the SENet attention module, which uses global average pooling to aggregate the global contexts to make them globally attentional information, and then reweighting the feature channels to amplify the weights of important feature maps [13]. However, SENet only focuses on the synthesis of information within the channel and does not take into account the importance of adjacent channel information. "Attention is all you need" published by Vaswani et al., in 2017 introduces the transformer model with self-attention as the basic unit that makes the attention mechanism really successful [14]. Transformer extracts global view features by its core operation of self-attention, and the self-attention mechanism has a clear advantage in capturing long-range dependencies in natural language processing. In contrast to CNN, each hidden unit in each feature learning layer of the transformer involves the global contextual information of the input. Therefore, the transformer architecture has been widely introduced to vision tasks in recent years and has received more and more research and attention. However, the vision transformer ignores local detailed features, and the heavy computational load of the self-attentive mechanism limits the depth of application of vision transformer in image defogging coding and decoding frameworks. Narasimhan S G et al., published "Vision Transformers for Single Image Dehazing" in 2022 using the network structure of vision transformer for haze removal [15]. The network borrowed the network structure of the Swin transformer and U-Net and made some modifications based on them, and then conducted haze removal experiments on the synthetic haze dataset and remote sensing haze dataset, and achieved good results. However, its proposed method of shifted window partitioning with reflection padding consumes a lot of cost and reduces its operational efficiency. Second, the network is less effective in removing non-uniform haze, and its ability to remove real haze is not outstanding.

In order to solve the problems that the convolutional neural network-based image defogging algorithm model captures incomplete global feature information and incomplete defogging, a parallel depth coding and decoding structure of a defogging network model is proposed in this paper. The encoder and decoder consist of a CNN branch and a vision transformer branch to capture the local and global features of the haze image, respectively, and a convolutional layer is used to fuse the hybrid features, which covers the global representation while retaining the local features. Finally, the features obtained by the encoder and decoder are fused to obtain richer feature information.

2. Atmospheric Scattering Model

In computer vision, the atmospheric scattering physical model is usually used to simulate the degradation process of foggy images. This model was proposed by Narasimhan and Nayar [16,17], and it is believed that the degradation of the observed imaging image is caused by two parts: one is the background light formed by the scattering of ambient light such as sunlight by the scattering medium in the atmosphere, and the other is the absorption of the target object by the suspended particles in the atmosphere and the scattering of the object itself. As a result, the brightness of the imaging system is reduced and the contrast is reduced to form a fog map. The spatial representation of the physical model of atmospheric scattering is shown in Figure 1, which can be mathematically represented as:

$$I(x) = J(x)\,t(x) + A[1 - t(x)] \tag{1}$$

where $I(x)$ is the foggy image acquired by an imaging device; $J(x)$ is a fog-free image to be restored; A is the global atmospheric light value; $t(x)$ is the image transmittance; x is the pixel coordinates of the image. In addition, $t(x)$ can also be expressed as:

$$t(x) = \exp[-\beta d(x)] \tag{2}$$

where β is the scattering coefficient of the atmosphere; $d(x)$ is the distance from the object to the imaging system.

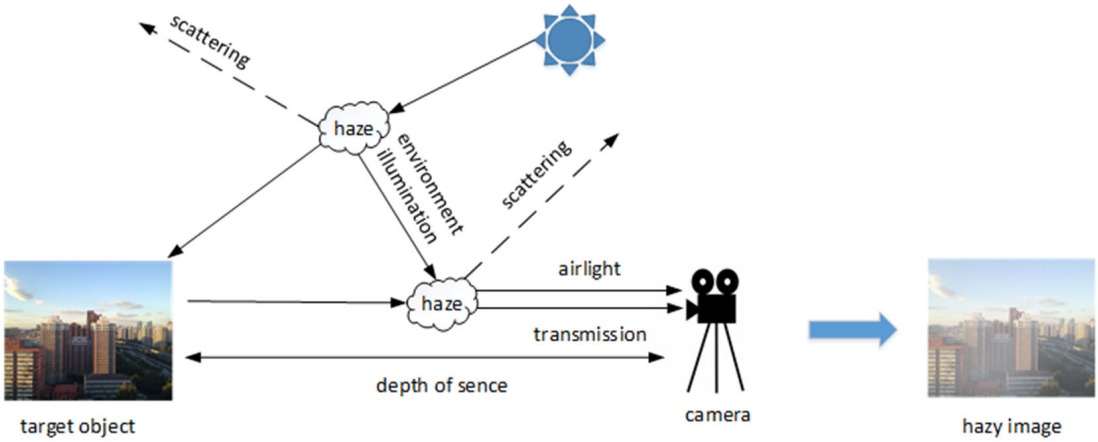

Figure 1. The physical model of atmospheric scattering.

The deformation of Equation (1) can be obtained as

$$J(x) = \frac{I(x) - A[1 - t(x)]}{t(x)} \tag{3}$$

The above equation shows that to solve the fog-free image with a known fog image, we need to estimate the transmittance map and the atmospheric light value from the fog image first, and then solve it based on the atmospheric scattering model. Therefore, an accurate estimation of the transmittance map and atmospheric light values is crucial to recover fog-free images [18].

3. Method of This Paper

In response to the traditional image defogging algorithm model based on a convolutional neural network with poor ability to capture global features and incomplete defogging, this paper proposes a parallel end-to-end defogging network model with a deep coding and decoding structure. The encoder of this model consists of two branches: a CNN and a vision transformer, in parallel, which continuously downsamples the features of the image to extract local and global features of haze images captured at different scales, respectively.

The decoder part of the model consists of the vision transformer, which restores the image to its original size by upsampling. To better fuse local and global features obtained through the encoder and decoder, enhanced feature extraction was finally performed using feature pyramid networks (FPN).

3.1. Network Structure

The CNN branch uses a feature pyramid structure in which the resolution of the feature map shrinks with the increasing network depth and its number of channels expands with the network depth. In this paper, the branch was divided into two parts: the downsampling layer and the feature extraction layer. The downsampling layer uses a convolution operation with a convolution kernel size of 2 and a step size of 2 to downsample the input haze image for the first time, so the original shape of the haze image is reduced by half in width and height and the number of channels becomes 96. A layer norm (layer normalization) layer was also added for normalization to ensure the stability of the data feature distribution while reducing overfitting. The feature extraction layer adopted a network structure with two thin ends and a thick middle, starting with a grouped convolution with a convolution kernel size of 7, followed by the addition of a layer norm layer. Then, the number of channels was expanded by four times using a convolution operation with a convolution kernel size of 1. The results were fed into the GELU activation function used to add nonlinear factors as a way to improve the neural network's ability to express the model, and finally the number of channels was reduced back to its original size by a convolution operation with a convolution kernel size of 1, and the inputs and outputs of the structure were connected with residuals to achieve feature extraction in the convolutional neural network. The overall network structure is shown in Figure 2. The downsampling and feature extraction layers were alternated, with a total of three downsamplings, and each downsampling feature map was reduced by half in width and height, and the number of output channels was 96, 192, and 384, respectively. The number of cycles of the three feature extraction layers was (1,1,3), and the local features of the learned haze map was saved for fusion with the features obtained from the vision transformer structure.

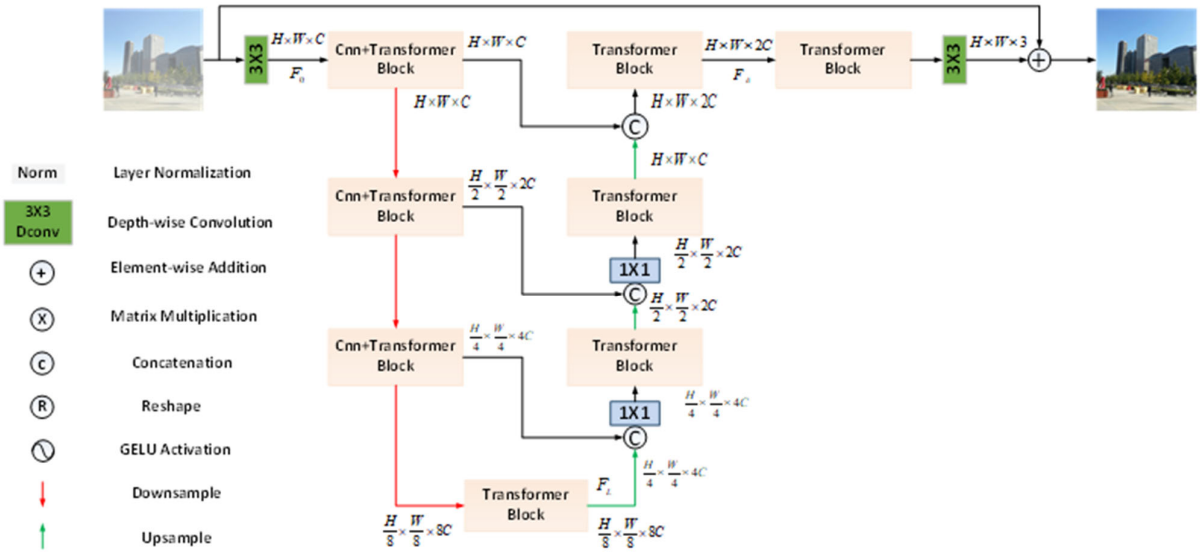

Figure 2. The overall network structure diagram.

The vision transformer branch is first given a haze image $I \in H \times W \times 3$, which is convolved to obtain low-level features $F_0 \in H \times W \times C$, where H and W are the spatial dimensions and C is the number of channels. These shallow features are then converted to deep features $F_d \in H \times W \times 2C$ by a 4-stage symmetric encoder–decoder. Each layer of the encoder contains multiple transformer blocks, where the number of blocks gradually increases from top to bottom. Starting from the high-resolution input, the encoder reduces

the space size of the feature map step-by-step while expanding the channel capacity. The decoder takes the low-resolution potential features $F_L \in \frac{H}{8} \times \frac{W}{8} \times 8C$ as input and gradually recovers the high-resolution representation. For feature downsampling and upsampling, Pixel-unshuffle and Pixel-shuffle operations are used, respectively. In order to fuse the semantic information at different levels, the encoder features are stitched with the decoder features along the channel dimension by hopping connections to make the number of channels twice the original size. Subsequently, a 1×1 convolution operation is used for all feature layers, except for the top one, to change the number of stitched and expanded channels to their original size. For the top feature layer, the low-level image features of the encoder were aggregated with the high-level features of the decoder, which facilitates the preservation of the fine structure and texture details of the recovered image. Then, the feature map was cycled at high spatial resolution using four transformer blocks to obtain high spatial refinement of the features, and this stage further enriched the depth features. Finally, the haze-free image $R \in H \times W \times 3$ was obtained by stacking the residuals of the feature map obtained from the refinement layer with the haze map using the convolution layer.

The main computational overhead in transformer comes from the self-attention mechanism. In traditional self-attention, the time and storage complexity of the key-query dot product interaction grows quadratically with the spatial resolution of the input, which is less suitable for certain high-resolution image restoration tasks that apply its larger computational load. To alleviate the above problem, a deep separable convolution method was used to implement the self-attentive mechanism, and the specific structure is shown in Figure 3. The key factor is the application of the self-attention mechanism across channels, rather than the spatial dimension (i.e., the calculation of cross-covariance across channels to generate an attention map that implicitly encodes the global context). A deep convolution was introduced to emphasize the local context before the feature covariance was computed to generate a global attention map. We normalized the tensor $Y \in R^{\hat{H} \times \hat{W} \times \hat{C}}$ by applying a 1×1 convolution to aggregate the cross-pixel level channel context and expand the number of channels to three times to make the tensor shape $H \times W \times 3C$. Then, the channel-level spatial context was encoded using 3×3 deep convolution, and the resulting tensor was divided equally into three parts along the channel dimension to obtain $W_d^Q W_p^Q$,$W_d^K W_p^K$ and $W_d^V W_p^V$. $W_p^{(\bullet)}$ is the 1×1 point convolution, $W_d^{(\bullet)}$ is the 3×3 depth convolution. The values obtained from the above operations were used to calculate the query (Q), key (K), and value (V) projections, respectively, by using the formulas $Q = W_d^Q W_p^Q Y$,$K = W_d^K W_p^K Y$ and $V = W_d^V W_p^V Y$. Then, the mapping of queries and keys is reshaped so that their dot products interact to produce a transposed attention map of size $R^{\hat{C} \times \hat{C}}$. In general, the calculation process can be expressed as the following equation:

$$\hat{X} = W_P Attention\left(\hat{Q}, \hat{K}, \hat{V}\right) + X \tag{4}$$

$$Attention\left(\hat{Q}, \hat{K}, \hat{V}\right) = \hat{V} \cdot Softmax\left(\hat{K} \cdot \hat{Q} / \alpha\right) \tag{5}$$

where X,$\hat{X}$ are the input and output feature maps, respectively. $\hat{Q} \in R^{\hat{H}\hat{W} \times \hat{C}}$, $\hat{K} \in R^{\hat{C} \times \hat{H}\hat{W}}$ and $\hat{V} \in R^{\hat{H}\hat{W} \times \hat{C}}$ are obtained by reconstructing the original size of $R^{\hat{H} \times \hat{W} \times \hat{C}}$. In the case of the Softmax function, α is a learnable scale parameter used to control the size of the dot product of $\hat{K}$ and $\hat{Q}$ before applying the Softmax function. Similar to the traditional multi-headed self-attention, the number of channels is divided into multiple heads that learn separate attention maps in parallel. In addition to the attention layer, the vision transformer contains a fully connected feedforward network that performs the same operation on each pixel location separately. It uses two 1×1 convolutions: one for expanding the feature

channels (usually by a factor $\gamma = 4$) and the other for reducing the channels to the original input dimension. A ReLU activation function between two convolution operations was also applied in the hidden layer to incorporate nonlinear factors.

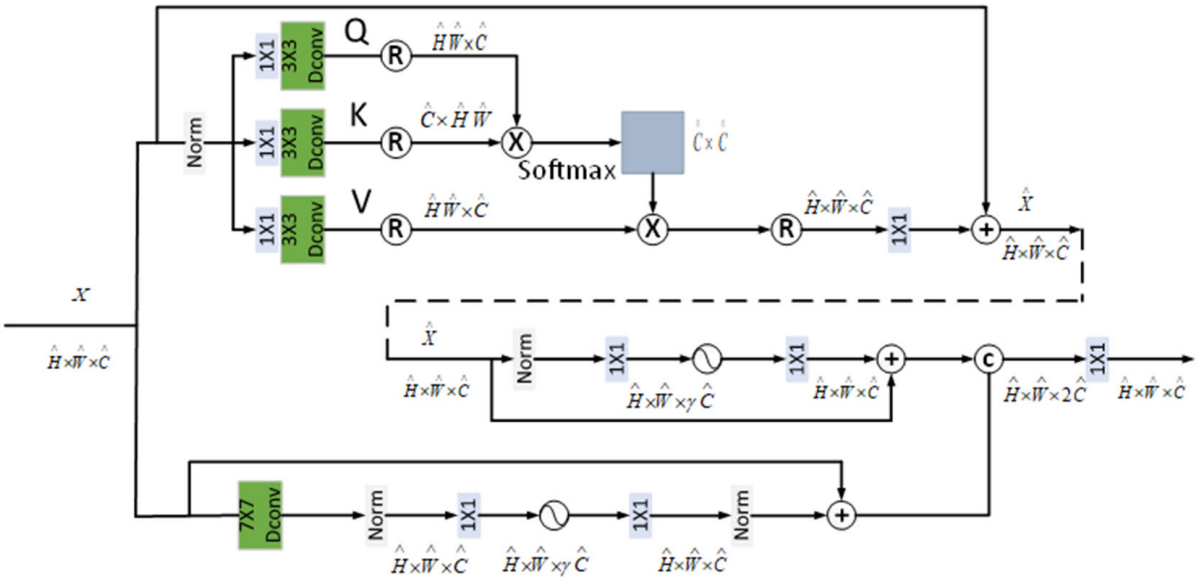

Figure 3. The CNN and transformer structure diagram.

3.2. Loss Function

The network model proposed in this paper uses two loss functions, denoted as Loss_mse and Loss_ssim, where Loss_mse is the minimum mean square error of the predicted outcome and the corresponding Groundtruth, also known as the MSE loss function, can be expressed by the formula:

$$L(y_i, \mathbf{f}(x_i)) = \frac{1}{N}\sum_{i=1}^{i=N}(y_i - f(x_i))^2 \tag{6}$$

where x_i denotes the i-th group of fogged images;y_i denotes the i-th group of non-fogged images; $f(x_i)$ denotes the result of the network model after defogging the fogged images; and $i = 1,2, \ldots, N$ denotes the number of samples for training. Loss_ssim is the SSIM loss, and SSIM is also called the structural similarity, which is a measure of the similarity of two images, comparing the defogged image with the real standard fog-free image, the larger the value of SSIM, the smaller the distortion of the defogged image. Moreover, SSIM takes values in the range of 0 to 1. Therefore, 1-SSIM was taken as the SSIM loss. The total loss function is the sum of Loss_mse and Loss_ssim, which can be expressed by the formula:

$$\text{Loss\_sum} = \text{Loss\_mse} + \text{Loss\_ssim} = \text{Loss\_mse} + (1 - \text{ssim}) \tag{7}$$

The whole training process was optimized by the ADAM solver, the initial learning rate was set to 0.0001, where $\beta_1 = 0.9$, $\beta_2 = 0.999$, and the learning rate was adjusted using cosine annealing to set eight iterations to complete one cosine cycle, the number of iterations was set to 10, and the number of batch images was set to 4.

4. Experimental Results and Analysis

In order to verify the real effect of the algorithm in this paper, the corresponding experimental verification was carried out on both the synthetic fogged images and real fogged images, and the results of this algorithm were compared with the current excellent defogging algorithms. The hardware environment was AMD EPYC 7543 32-Core Processor, NVIDIA GeForce RTX 3090, and the same hardware configuration environment was used

for the comparison experiment. The experiments were conducted under Ubuntu18.04 using the python programming language, applying the Pytorch deep learning framework to build the network and programmatically implementing the training and testing of the dataset and image defogging.

4.1. Training Data

It is normally difficult to acquire a large dataset of paired fog-free images and synthetic datasets are usually used (i.e., images with fog are synthesized based on atmospheric scattering models on the real dataset). RESIDE [19] is a large-scale synthetic dataset consisting of five subsets: the indoor training set (ITS), outdoor training set (OTS), comprehensive objective test set (SOTS), real test set (RTTS), and hybrid subjective test set (HSTS). Among them, ITS and OTS are used for training on indoor and outdoor environments, respectively, SOTS is used for testing on both indoor and outdoor scenes, and RTTS is used for testing on real fogged images. The fog removal models in this paper were trained on outdoor datasets including the outdoor training set (OTS) of RESIDE and the NTIRE2020-Dhaze dataset for uniform haze and non-uniform haze removal, respectively. For the outdoor training set (OTS), a sub-dataset of RESIDE, 41,240 samples were randomly selected from this paper for training, and 500 outdoor synthetic haze images were randomly selected from the synthetic target test dataset (SOTS) for testing. For the NTIRE2020-Dhaze dataset, 4800 samples were synthesized from the first 50 original high-resolution samples of the dataset by random cropping, flipping and rotating for training, and the last five samples of the dataset were used for testing.

4.2. Experimental Results of Synthesizing Uniform Haze Images

In the experiment of synthesizing uniform haze images, 500 randomly selected outdoor synthetic haze images from the target test dataset (SOTS) were used for testing, and some of the experimental results are shown in Figure 4. The figure shows that the method in [5] effectively removed the fog, but because the dark channel a priori theory is only suitable for the non-sky area, resulting in overexposure of the sky area and low overall brightness of the non-sky area after the fog removal (such as the first, second and fifth pictures in Figure 4b), the visual effect is poor. The defogged images of the method in [6] solve the problem of dark channel theory overexposure in the sky region, but the color deviation of the sky region after defogging is larger than that of the sky region of the standard fog-free images (e.g., the first, second, and fifth images in Figure 4c), while the overall brightness of the non-sky region is still lower compared to the standard fog-free images. After the method in [11] for the test image defogging, individual images still had the problem of more residual haze with incomplete defogging (e.g., the first and second images in Figure 4d), and the recovered images did not show color distortion. The method of fog removal in [12] had a good effect without obvious fog, but the fogged images showed local color deviations such as the ships and city buildings in the first panel of Figure 4e and the yellowish color of the ground in the fourth panel and the road in the fifth panel, which differed greatly from the standard fog-free images. Comparing the above-mentioned various defogging algorithms with the algorithm in this paper, it is obvious that the algorithm in this paper effectively removed the fog from the image without color distortion, and the image was recovered more naturally, which was closer to the real fog-free image compared with the other methods. Comparing the resulting images in [15] and the algorithm of this paper together, we found that there was no fog residue in the visual effect, and both could accomplish the task of haze removal well on the same dataset. However, the haze situation in this dataset was lighter, and when faced with complex haze situations, the algorithm in this paper was better able to highlight the advantages.

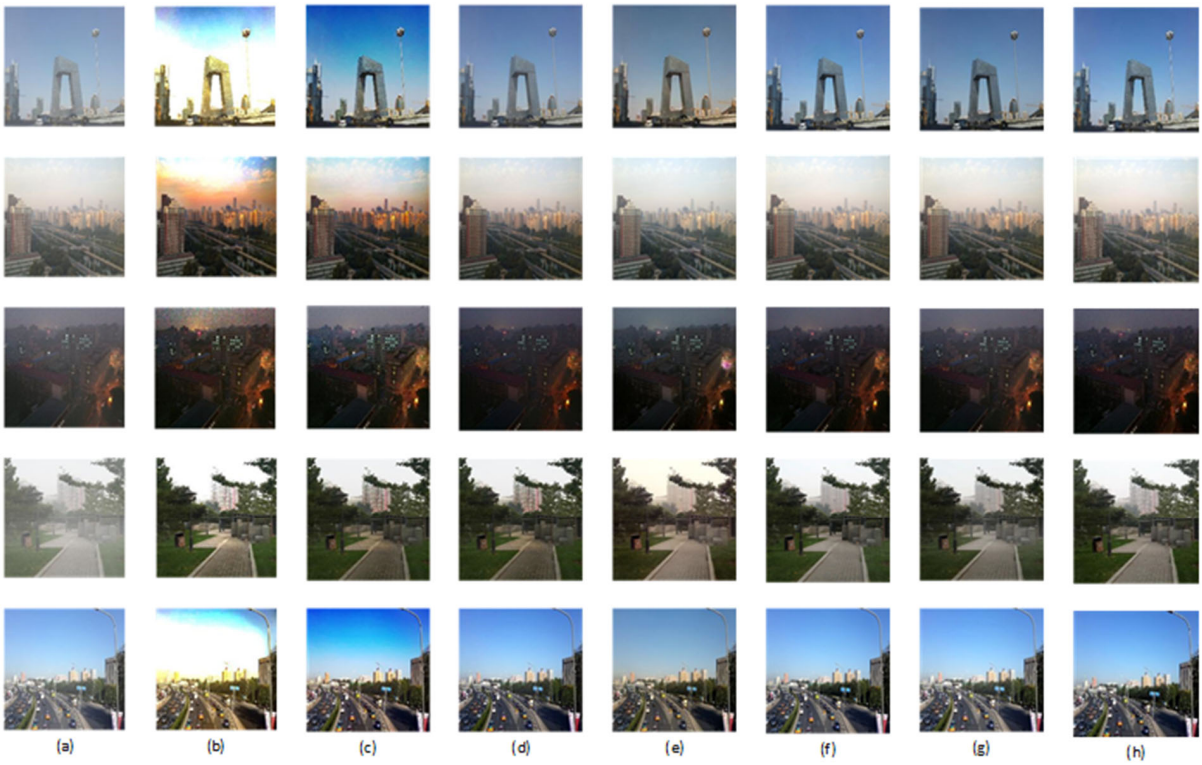

Figure 4. The experimental results of synthesizing homogeneous hazy images. (**a**) Hazy image; (**b**) method in [5]; (**c**) method in [6]; (**d**) method in [11]; (**e**) method in [12]; (**f**) method in [15]; (**g**) proposed method; (**h**) standard haze-free image.

The effectiveness of the method in this paper cannot be fully explained by subjective judgment alone, so two indices, peak signal-to-noise ratio (PSNR) and structural similarity (SSIM) [20] were selected for the data analysis of the experimental results in this paper. PSNR is an important indicator of image quality, and the larger the value, the closer the defogged image is to a standard fog-free image. PSNR is calculated as follows:

$$MSE = \frac{1}{mn}\sum_{i=0}^{m-1}\sum_{j=0}^{n-1}[I(i,j) - K(i,j)]^2 \tag{8}$$

$$PSNR = 10 \cdot \log_{10}\left(MAX_I^2/MSE\right) \tag{9}$$

where I is the original fogged image; K is the defogged image; i, j are the horizontal and vertical coordinates of the pixel values, respectively; and MAX was set to 255 for an image with a bit depth of 8.

SSIM is also a fully referenced image quality evaluation metric that mainly describes image similarity and consists of three contrast modules: brightness, contrast, and structure [21]. Comparing the defogged image with the real standard fog-free image, SSIM takes the value range of [0, 1], and a larger value means less image distortion, which is calculated as follows:

$$SSIM(x,y) = \frac{(2\mu_x\mu_y + c_1)(\sigma_{xy} + c_2)}{\left(\mu_x^2 + \mu_y^2 + c_1\right)\left(\sigma_x^2 + \sigma_y^2 + c_2\right)} \tag{10}$$

where $\mu_x\mu_y$ are the means of images x and y, respectively; σ_x^2 and σ_y^2 are the standard deviations of images x and y, respectively; σ_{xy} is the x and y covariances of the images; c_1, c_2 are constants to avoid denominators of 0.

Table 1 shows the quantitative evaluation of different defogging methods on the SOTS dataset. It can be seen that the quantitative evaluation metrics of the recovered results

using [4,5] were low, while the quantitative evaluation metrics of the recovered results using these deep learning methods in [9,10] were improved to some extent. The algorithm in this paper was 6.36 and 0.08 higher than that in [9] in the PSNR and SSIM, respectively, and only below that in [15] in both the PSNR and SSIM. Therefore, from the combination of the subjective comparison of the above algorithms and objective indicators, it can be concluded that the algorithm in this paper can better remove the thin fog. Moreover, the integrity of image information is retained, which is closer to the real fog-free image.

Table 1. The quantitative evaluation of different defogging methods on the SOTS dataset.

Dataset	Evaluation Indicators	Ref. [5]	Ref. [6]	Ref. [11]	Ref. [12]	Ref. [15]	Propose Method
SOTS	PSNR	15.42	19.52	21.58	25.22	30.28	27.94
	SSIM	0.74	0.83	0.8	0.86	0.92	0.88

4.3. Experimental Results of Non-Uniform Haze Images

Since most existing deep learning-based defogging algorithms use a uniform dataset to train their own defogging models, the excessive pursuit of higher evaluation metrics on a single dataset leads to poor generalization ability of the models. Although most of the defogging algorithms can obtain good results on the SOTS dataset, they are not fully effective in recovering clear and fog-free images in the face of a complex and variable real haze environment. Therefore, in order to improve the generalization ability of this paper's defogging model and better improve the effect of recovering real haze pictures, based on the above problems, this paper proposed and produced a non-uniform haze dataset. The NTIRE2020-Dhaze dataset provided by the NTIRE2020 Image Recovery Challenge is shown in Figure 5b. This dataset is different from the synthetic dataset in that it was produced by a professional haze machine with real haze to generate real hazy images, and although the real dataset seems more attractive, it is difficult to obtain enough images. In this paper, the non-uniform haze dataset was based on the first 50 high-resolution image samples in the NTIRE2020-Dhaze dataset, and 4800 samples were synthesized from the original samples by random cropping, flipping and rotating for training, and the last five images of the dataset were used for testing.

Figure 5. The training dataset: (**a**) Outdoor dataset; (**b**) NTIRE2020-Dhaze dataset.

In the experiments of the non-uniform haze images, the experimental results of five images tested after using the NTIRE2020-Dhaze dataset are shown in Figure 6. It can be seen from Figure 6 that the method used in [5] was less effective in removing haze as it

contained a large area of white bright area after fogging, which appeared as too much exposure (such as in Figure 6b of the first, second, fifth) and in other areas after fogging, the overall color was dark. The defogged image of the method used in [6] solves the problem of overexposure after defogging the dark channel theory for bright areas containing large white areas, but the color deviation between the defogged image and the standard fog-free image was large, and the overall image was seriously dark (such as in Figure 6c). The method used in [11] for the test images was found to remove only the thin haze at the edges of the images, however, it was not effective for the areas with a higher haze concentration (such as Figure 6d). The method of fog removal in [12] was unable to remove the areas with high haze concentration, while the fogged images showed local color deviations, (such as the yellowish color of the lawn in the first, second, and third panels of Figure 6e differed significantly from the standard fog-free images). When the results of the dehazing method in [15] and the dehazing algorithm in this paper were compared with the real fog free images, it could clearly be seen that the algorithm in this paper effectively removed the mist and fog in the images. There was no color distortion, and it was closer to the real fog-free image than that of other methods.

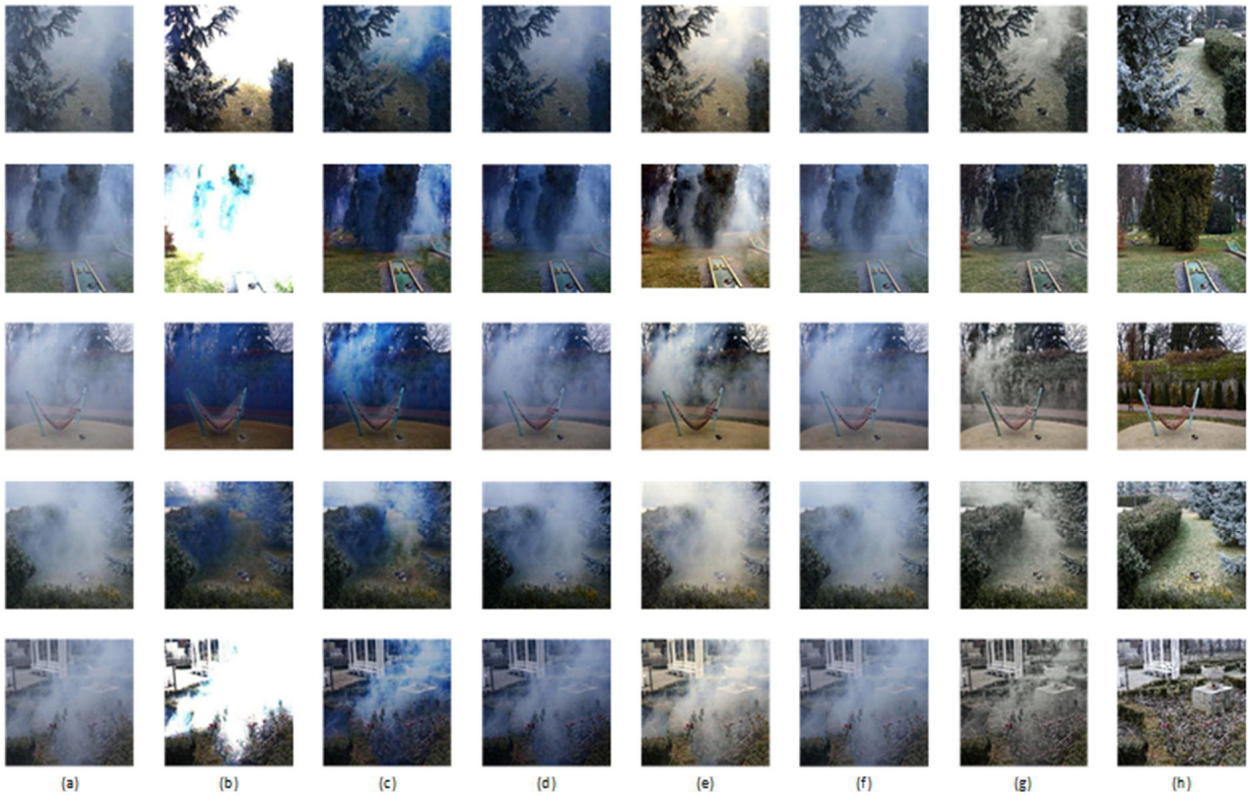

Figure 6. The experimental results of synthesizing inhomogeneous hazy images. (**a**) Hazy image; (**b**) method in [5]; (**c**) method in [6]; (**d**) method in [11]; (**e**) method in [12]; (**f**) method in [15]; (**g**) proposed method; (**h**) standard haze-free image.

The quantitative evaluation of different defogging methods on the NTIRE2020-Dhaze dataset is presented in Table 2, from which it is clear that the algorithm in this paper was higher than the comparison methods in terms of the PSNR and SSIM obtained on the non-uniform haze datasets. This shows that the model in this paper could recover fog-free images better under different haze environments, which had obvious advantages compared with other algorithms.

Table 2. The quantitative evaluation of different defogging methods on the NTIRE2020-Dhaze dataset.

Dataset	Evaluation Indicators	Ref. [5]	Ref. [6]	Ref. [11]	Ref. [12]	Ref. [15]	Propose Method
NTIRE'20	PSNR	7.80	12.13	12.30	10.86	10.28	15.74
	SSIM	0.24	0.28	0.26	0.33	0.31	0.44

4.4. Running Time

The above experimental results have verified the effectiveness of the method in this paper, and in order to further verify the performance of the method in this paper in terms of efficiency, the average running times of different algorithms were obtained by counting the above experimental times, as shown in Table 3. In the algorithm running efficiency experiments, we used the trained algorithm models on the uniform haze test set SOTS and the non-uniform haze test set NTIRE2020-Dhaze for forward inference, to input the foggy images into the model to infer the fog-free images, and calculate the average time of different algorithms to remove the haze, where the model with a similar number of algorithm parameters in [15] was chosen to be similar to this paper as dehazeformer-t for the calculation. As can be seen from the table, the running time of the method in this paper was significantly faster than that of the traditional defogging algorithms in [5,6], and the deep learning methods in [15]; compared with the deep learning defogging algorithms in [11,12], the method in this paper also had an advantage in running speed. Because the separable convolution was used to build the multi-head attention mechanism, the computation cost was greatly reduced and the running speed of the program was improved.

Table 3. The running time of the experimental algorithm.

Dataset	Ref. [5]	Ref. [6]	Ref. [11]	Ref. [12]	Ref. [15]	Propose Method
SOTS	0.91	0.98	0.81	0.83	1.16	0.74
NTIRE'20	0.93	0.95	0.79	0.85	1.36	0.72

4.5. Experimental Results of Real Haze Images

In order to verify the effect of the model in this paper on recovering real foggy images, four real outdoor foggy images were selected for the defogging experiments, and the experimental results are shown in Figure 7. There was overexposure in the sky region after the method in [4] for haze removal (e.g., panels 2 and 3 of Figure 7b) and more serious color bias in the processed images (e.g., panel 4 of Figure 7b). The defogged image of the method in [5] solved the defect of the method in [4], in which the halo appeared in the sky region, but the serious problem of color bias of the processed image remains unresolved (e.g., the fourth panel of Figure 7b). The method in [9] had more fog residue after defogging the test images. The fog removal by the method in [10] was better, with no visible fog, but the fogged images showed local color deviations (e.g., the building area in panel 1 in Figure 7b). A few images after the method of haze removal in [15] often showed noise points in the picture quality, (e.g., the building area in the second panel in Figure 7f). Moreover, the restoration effect of the fog picture was not as good as the algorithm in this paper (the middle area of the first and third pictures in Figure 7f). Through the above sufficient experiments, it is proven that the proposed method had high efficiency and could effectively remove fog without distortion in the sky area. Furthermore, the picture as a whole colorless partial phenomenon had good visual effects.

Figure 7. The experimental results of the real outdoor hazy images. (**a**) Hazy image; (**b**) method in [5]; (**c**) method in [6]; (**d**) method in [11]; (**e**) method in [12]; (**f**) method in [15]; (**g**) proposed method.

5. Conclusions

For the traditional image defogging algorithm model based on a convolutional neural network to capture incomplete global feature information and incomplete defogging, this paper proposed a parallel end-to-end defogging network model with a deep coding and decoding structure. The encoder of the model consists of two branches: a CNN and a vision transformer in parallel, which continuously downsamples the image features and extracts local and global features of the haze images captured at different scales, respectively. The decoder part of the model consists of the vision transformer, which restores the image to its original size by upsampling. To better fuse the local and global features obtained through the encoder and decoder, the feature extraction was finally enhanced using FPN. The comparison results on the uniform, non-uniform haze datasets and real datasets showed that the algorithm in this paper had a better defogging effect and could recover fog-free images better under different haze environments. Future work will continue to optimize the algorithm structure in order to achieve better defogging results.

Author Contributions: Conceptualization, S.L. and Q.Y.; Methodology, S.L. and Q.Y.; Funding acquisition, S.L.; Validation, S.L., Q.Y. and Y.Z.; Writing—original draft preparation, Q.Y.; Writing—review and editing, S.L., Q.Y., Y.Z. and B.L.; Supervision, Y.Z., B.L. and F.W. All authors have read and agreed to the published version of the manuscript.

Funding: This work was funded by the Henan Province Science and Technology Key Project (Grant Nos. 212102210050, 202102210081); the Doctor Foundation of Henan Polytechnic University (Grant No. B2014-35, B2019-50); the Outstanding Young Scientists Program of Beijing Colleges and Universities (Grant No. BJJWZYJH01201910006021); the Open Foundation of the State Key Laboratory of Fluid Power and Mechatronic Systems (Grant No. GZKF-202016); the Sub project of strengthening key basic research projects in the basic plan of the Science and Technology Commission of the Military Commission (Grant No. 2019-JCJQ-ZD-120-13); the Postgraduate education and teaching reform project of Henan Polytechnic University (Grant No. 2021YJ04); the Fundamental Research Funds for the Universities of Henan Province (Grant No. NSFRF200403).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

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

References

1. Wang, G.J.; Jian, C.L.; Xiang, Q. Hardware reconstruction acceleration method of CNN-based single image defogging model. *J. Comput. Appl.* **2021**, *23*, 50–53. [CrossRef]
2. Wang, W.G.; Wang, B.H.; Zhang, J.J.; Li, L. Image Haze Removal Algorithm Based on Histogram Specification. *Comput. Technol. Dev.* **2014**, *24*, 241–244.
3. Land, E.H.; McCann, J.J. Lightness and retinex theory. *J. Opt. Soc. Am.* **1971**, *61*, 1–11. [CrossRef]
4. Liu, D.M.; Chang, F.L. Coarse-to-Fine Saliency Detection Based on Non-Subsampled Contourlet Transform Enhancement. *Acta Opt. Sin.* **2019**, *39*, 380–387.
5. He, K.; Jian, S. Single Image Haze Removal Using Dark Channel Prior. *IEEE Trans. Pattern Anal. Mach. Intell.* **2011**, *33*, 2341–2353. [PubMed]
6. Wang, Y.K.; Hu, Y.; Wang, H.; Li, Y.S. Image dehazing algorithm by sky segmentation and superpixel-level dark channel. *J. Jilin Univ.* **2019**, *49*, 1377–1384.
7. Meng, G.; Wang, Y.; Duan, J. Efficient Image Dehazing with Boundary Constraint and Contextual Regularization. In Proceedings of the 2013 IEEE International Conference on Computer Vision, Sydney, Australia, 1–8 December 2013.
8. Cai, B.; Xu, X.; Jia, K. An End-to-End System for Single Image Haze Removal. *IEEE Trans. Image Process.* **2016**, *25*, 5187–5198. [CrossRef] [PubMed]
9. Ren, W.; Si, L.; Hua, Z. Single Image Dehazing via Multi-scale Convolutional Neural Networks. In Proceedings of the 14th European Conference on Computer Vision, Amsterdam, The Netherlands, 11–14 October 2016.
10. Wang, Z.H.; Wang, Y.H. Image defogging algorithm based on improved convolutional neural network. In Proceedings of the 32nd Chinese Process Control Conference, Taiyuan, China, 30 July–1 August 2021.
11. Li, B.; Peng, X.; Wang, Z.; Xu, J.; Feng, D. Aod-net: All-in-one dehazing network. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 4770–4778.
12. Chen, D.; He, M.; Fan, Q. Gated Context Aggregation Network for Image Dehazing and Deraining. In Proceedings of the 2019 IEEE Winter Conference on Applications of Computer Vision (WACV), Waikoloa Village, HI, USA, 7–11 January 2019.
13. Jie, H.; Li, S.; Gang, S. Squeeze-and-Excitation Networks. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake City, UT, USA, 18–23 June 2018.
14. Vaswani, A.; Shazeer, N.; Parmar, N. Attention Is All You Need. In Proceedings of the Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems, Long Beach, CA, USA, 4–9 December 2017.
15. Narasimhan, S.G.; Nayar, S.K. Vision Transformers for Single Image Dehazing. *arXiv* **2022**, arXiv:2204.03883.
16. Narasimhan, S.G.; Nayar, S.K. Contrast restoration of weather degraded images. *IEEE Trans. Pattern Anal. Mach. Intell.* **2003**, *25*, 713–724. [CrossRef]
17. Narasimhan, S.G.; Nayar, S.K. Vision and the Atmosphere. *Int. J. Comput. Vis.* **2002**, *48*, 233–254. [CrossRef]
18. He, Y.H.; Li, Y.F.; Huang, S.K. Adaptive Image Dehazing Algorithm Based on Deep Convolutional Neural Network. *Electron. Sci. Technol.* **2020**, *33*, 70–73.
19. Li, B.; Ren, W.; Fu, D. RESIDE: A Benchmark for Single Image Dehazing. *arXiv* **2017**, arXiv:1712.04143.
20. Zhou, W.; Bovik, A.C.; Sheikh, H.R. Image quality assessment: From error visibility to structural similarity. *IEEE Trans. Image Process.* **2004**, *13*, 600–612.
21. Qi, Y.F.; LI, Z.H. Image Dehazing Method Based on Multi-scale Convolutional Neural Network and Classification Statistics. *Infrared Technol.* **2020**, *42*, 190–197.