applied sciences

MDPI

Article

Attention-Gate-Based Model with Inception-like Block for Single-Image Dehazing

Cheng-Ying Tsai and Chieh-Li Chen *

Department of Aeronautics and Astronautics, National Cheng Kung University, Tainan 70101, Taiwan; p46091204@gs.ncku.edu.tw
* Correspondence: chiehli@mail.ncku.edu.tw

Abstract: In recent decades, haze has become an environmental issue due to its effects on human health. It also reduces visibility and degrades the performance of computer vision algorithms in autonomous driving applications, which may jeopardize car driving safety. Therefore, it is extremely important to instantly remove the haze effect on an image. The purpose of this study is to leverage useful modules to achieve a lightweight and real-time image-dehazing model. Based on the U-Net architecture, this study integrates four modules, including an image pre-processing block, inception-like blocks, spatial pyramid pooling blocks, and attention gates. The original attention gate was revised to fit the field of image dehazing and consider different color spaces to retain the advantages of each color space. Furthermore, using an ablation study and a quantitative evaluation, the advantages of using these modules were illustrated. Through existing indoor and outdoor test datasets, the proposed method shows outstanding dehazing quality and an efficient execution time compared to other state-of-the-art methods. This study demonstrates that the proposed model can improve dehazing quality, keep the model lightweight, and obtain pleasing dehazing results. A comparison to existing methods using the RESIDE SOTS dataset revealed that the proposed model improves the SSIM and PSNR metrics by at least 5–10%.

Keywords: single-image dehazing; deep learning; attention gate; lightweight; real-time

Citation: Tsai, C.-Y.; Chen, C.-L. Attention-Gate-Based Model with Inception-like Block for Single-Image Dehazing. *Appl. Sci.* **2022**, *12*, 6725. https://doi.org/10.3390/app12136725

Academic Editor: Samuel Morillas

Received: 24 May 2022
Accepted: 29 June 2022
Published: 2 July 2022

1. Introduction

Low-visibility environments have a significant influence on vision-based autonomous driving systems. For example, haze can cause serious damage to image quality, including color and brightness deviations that lead to numerous visual information losses. Consequently, the performance of computer vision algorithms, such as object detection, semantic segmentation, and visual simultaneous localization and mapping (visual SLAM), are degraded, and the safety of autonomous car driving that relies heavily on the above-mentioned algorithms is jeopardized accordingly. To preserve the performance quality of these vision-based algorithms, the images obtained by cameras require preprocessing before feeding to these functional processes.

The main cause of haze is the scattering of atmospheric particles. Gui et al. [1] divided image-dehazing methods into two categories depending on whether an atmospheric scattering model (ASM) [2] is used. Tan [3] mentioned that single-image dehazing using an ASM may face an ill-posed problem due to the availability of the transmission map and atmospheric light. Therefore, statistical experience is required to obtain a reasonable assumption called the prior, such that the ASM-based approach can be applied to a single-image dehazing problem. He et al. [4] obtained the dark channel prior (DCP) from large numbers of outdoor images by statistical methods. This is because the dark channel value of the haze-free outdoor image captured during daytime is close to zero, except for the sky and white areas. Other priors were also proposed in the literature. Zhu et al. [5] found

that the difference between brightness and saturation is proportional to the depth, which is the so-called color attenuation prior (CAP). The haze-line prior [6] is based on the observation that the pixel values of a hazy image can be modeled as lines in the RGB space with atmospheric light as the origin. The distance from the origin of the same haze line is affected by the transmission map. Ju et al. [7] added an absorption function to the ASM and solved the image dehazing problem with the gray-world assumption. Based on DCP, Yang et al. [8] improved the estimation method of atmospheric light to avoid mistaking the sky region as containing haze and smoothed the transmission map with two morphological operators, namely, dilation and erosion.

The recent development of graphics processing units (GPUs), which deliver extraordinary acceleration in workloads involving artificial intelligence and machine learning, leads to an alternative dehazing approach based on these related technologies and becomes a research hotspot without using the ASM. Deep learning methods can be divided into supervised and unsupervised learning methods. The method based on supervised learning requires both hazy and ground truth images as inputs to a specially designed convolutional neural network (CNN) and trains a dehazing network using an appropriate loss function and backpropagation. Since the transmission map and atmospheric light are required when using the ASM approach, Cai et al. [9] set the atmospheric light to 1 and constructed a network structure called DehazeNet to estimate the transmission map. However, because the atmospheric light is defined in advance, the result of image dehazing is poor. Li et al. [10] integrated the transmission map and atmospheric light into a function as the learning target of the proposed AOD-Net. Furthermore, Yang et al. [8] redesigned the network architecture of AOD-Net [10], inheriting the advantages of few trainable parameters and real-time capabilities. Based on the idea of integrating the transmission map and atmospheric light as the learning target, Zhang et al. [11] designed a multi-scale network architecture using 1×1 convolution and pooling. Qin et al. [12] concatenated two attention mechanisms to FFA-Net: channel attention and pixel attention. These mechanisms result in a greater weight in the blurred region, but the number of trainable parameters is too large. Liu et al. [13] proposed a generic model-agnostic CNN that is composed of an encoder and decoder associated with residual blocks, and the whole network can be trained end-to-end which means that no physical knowledge should be obtained in advance. It is worth mentioning that this network architecture can be applied to other tasks in addition to image dehazing.

The classic method based on unsupervised learning for the image dehazing problem involves the use of generative adversarial nets (GAN). The discriminator is used to determine authenticity, and the generator and the discriminator are trained separately through backpropagation. Engin et al. [14] imported cyclic perceptual consistency loss into CycleGAN [15] to improve the quality of image dehazing, but it cannot recover color deviation and object edge well. Qu et al. [16] proposed a two-scale generator and discriminator, and an enhancer with a multi-scale average pooling architecture has been constructed to provide more different receptive fields. In heavily hazy scenes, there is still room for improvement; therefore, it is necessary to apply more enhancing blocks to enhanced Pix2pix [16]. Mehta et al. [17] modified CycleGAN [15] and conditional GAN [18] to formulate their image-dehazing model. The proposed model of [17] outperforms the aforementioned GANs, but it may generate unnatural colors for some scenes. However, GANs are difficult to train efficiently because the generator and the discriminator are hard to converge at the same time. In general, the model sizes of image-dehazing GANs are comparatively large, which has a significant impact on execution time.

In practical applications, the image-dehazing algorithm belongs to the pre-processing part for autonomous driving systems; thus, while improving the quality of image dehazing, it is also necessary to consider the execution time and size of the trainable parameters. This paper proposes a lightweight CNN containing an attention gate, an inception-like block, and a spatial pyramid pooling (SPP) block with an appropriate loss func-

tion to achieve better image-dehazing quality. Finally, using an existing dataset, comparisons between other state-of-the-art and proposed image-dehazing algorithms based on image-dehazing quality, trainable parameters, and the average execution time per image are presented. In conclusion, the main contributions of this study can be summarized as follows:

- It provides a real-time and lightweight end-to-end image-dehazing CNN model that restores the blur caused by the haze environment without intermediate component computation for ASM and knowledge of the physical model.
- Before feeding hazy images to the CNN, an image pre-processing block was used to map normalized RGB images to different color spaces. The image pre-processing block increases the number of layers of feature maps such that the image-dehazing CNN can extract features effectively.
- The inception-like block uses two 3×3 convolutions to replace one 5×5 convolution, which effectively increases the receptive field. The SPP block utilizes different pooling kernel sizes to extract feature values and to generate more feature maps. The proposed attention gate can smartly pay more attention to unclear structures, such as buildings and pedestrians.
- Through an ablation study and a quantitative evaluation, this study demonstrates the advantages of using the image pre-processing block, the inception-like block, the spatial pooling blocks, and the proposed attention gate.
- The number of trainable parameters of the image-dehazing CNN is 207.3 K, the model size is 0.86 MB, and the average execution time per frame under GPU acceleration is 0.018 s, which is equivalent to 55.56 fps.
- RESIDE-SOTS and RESIDE-HSTS were used as the testing image-dehazing-quality datasets, which improved the peak signal-to-noise ratio (PSNR) and structural similarity (SSIM) compared to other state-of-the-art image-dehazing methods.

2. Preliminaries

2.1. Atmospheric Scattering Model

In the field of computer vision, atmospheric scattering models are often used to describe the formation formula for hazy images, as shown in Equation (1).

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

where x is the pixel position, $I(x)$ is the hazy image, $J(x)$ is the clear image, A is the global atmospheric light, and $t(x)$ is the transmission map described by Equation (2). $J(x)t(x)$ is called direct attenuation, and $A(1 - t(x))$ is called airlight.

$$t = e^{-\beta d(x)}, \tag{2}$$

where β is the scattering coefficient of the atmosphere and $d(x)$ is the distance between the camera and the observed objects. According to Equation (2), the larger the distance, $d(x)$, the lower the transmittance. From the direct attenuation term, $J(x)t(x)$, a clear image becomes blurred and unclear for a larger $d(x)$ due to the atmospheric light, which makes the image brighter and whiter.

The single-image dehazing problem is an ill-posed problem because it requires both global atmospheric light and transmission maps to be obtained simultaneously. Therefore, statistical methods for obtaining empirical rules are necessary for model-based approaches.

2.2. Deep-Learning-Based Method

Using U-Net [19] as the prototype of the image-dehazing network to carry out single-image dehazing is a common method. Its architecture includes an encoder and a decoder. The encoder is composed of several convolution filters and maximum pooling operations, while the decoder contains several convolution filters and transposed convolution filters

to perform upsampling. In addition, before the decoder performs convolution, the feature map of the same resolution generated by the encoder and the transposed convolution result of the previous layer are directly concatenated and used as the convolution input.

U-Net [19] is mainly used to solve the problem of image segmentation. When using this architecture to solve the problem of image dehazing, it is not possible to simply deepen the number of network layers; but a suitable neural network block should be imported. For example, TheiaNet [20] imports the bottleneck enhancer into the final output of the encoder, which extracts the feature map from coarse to fine through multi-scale pooling to obtain different feature maps and then concatenates these feature maps together. The final output of the decoder is added to the aggregation head, and the outputs of the encoder and decoder of different layers are upsampled to the same resolution and concatenated.

2.3. Downsampling and Upsampling

Downsampling and upsampling reduce and enlarge the resolution of the original image, respectively, by a specified multiple. Downsampling often uses maximum pooling, a predefined operation, to improve the receptive field without increasing the computational complexity. Upsampling often uses predefined interpolation methods, such as nearest-neighbor interpolation and bilinear interpolation.

The advantage of predefined operations is that they save memory space, but the disadvantage is that CNN cannot learn its own sampling method. Therefore, convolution and transposed convolution can be adopted to allow the CNN to learn a suitable downsampling and upsampling process during the training process. For example, to reduce the image resolution to one-half of the original image, a 2×2 convolution with a stride of two can be used instead of a 2×2 max pooling with a stride of two, without sacrificing the values of the other feature maps. Similarly, to double the image resolution of the original image, a 2×2 transposed convolution with a stride of two can be used instead of the interpolation method.

Additionally, when the kernel size cannot be divisible by stride, it causes a checkerboard effect due to the uneven overlapping. Therefore, kernel size $= 2$ and stride $= 2$ are applied to avoid generating a checkerboard-like pattern.

2.4. Multiple Input Channel

It can be observed from the ASM that the image is mainly blurred due to atmospheric light and particle scattering. Based on these phenomena, Ren et al. [21] proposed GFN, which inputs three different image processing methods, namely, white balance to deal with color deviation, contrast enhancement to increase the structure of objects, and gamma correction to solve dark regions. Then, they are concatenated to form a 9-channel input. The dehazing image is expressed by

$$J = C_{wb} \circ I_{wb} + C_{ce} \circ I_{ce} + C_{gc} \circ I_{gc}, \quad (3)$$

where C_{wb}, C_{ce}, and C_{gc}, are confidence maps for the white-balanced image, I_{wb}, the contrast enhanced image, I_{ce}, and the gamma corrected image, C_{gc}, respectively, and $\circ$ is the Hadamard product.

In addition to image pre-processing, a multichannel input can also be mapped to different color spaces to increase the number of channels in the input image. Wan et al. [22] emphasized that the halo effect can be suppressed in the HSV color space. Tufail et al. [23] experimentally proved that better contrast and brightness can be obtained in the YCbCr color space. Therefore, inspired by [22,23], Mehra et al. [20] used a multi-cue color space, including RGB, HSV, YCbCr, and Lab, to formulate a 12-channel input.

The proposed method integrates the concepts of GFN [21] and TheiaNet [20] to preprocess hazy images. In the next section, we introduce significant blocks of our image-

dehazing CNN, including the image pre-processing blocks, inception-like blocks, SPP blocks, and attention gates (AG).

3. Proposed Method

This section introduces details of the proposed image-dehazing CNN, as shown in Figure 1. First, the image preprocessing block performs image normalization and maps to different color spaces to form a 12-channel input. Subsequently, the inception-like block is used to increase the receptive field, and an SPP block is performed to increase the dimension of the feature map of high-level feature maps. The skip concatenation of U-Net [19] is replaced with an AG, which can adjust the gain of the feature map to leverage the blurry regions. Finally, an appropriate loss function for image dehazing is presented.

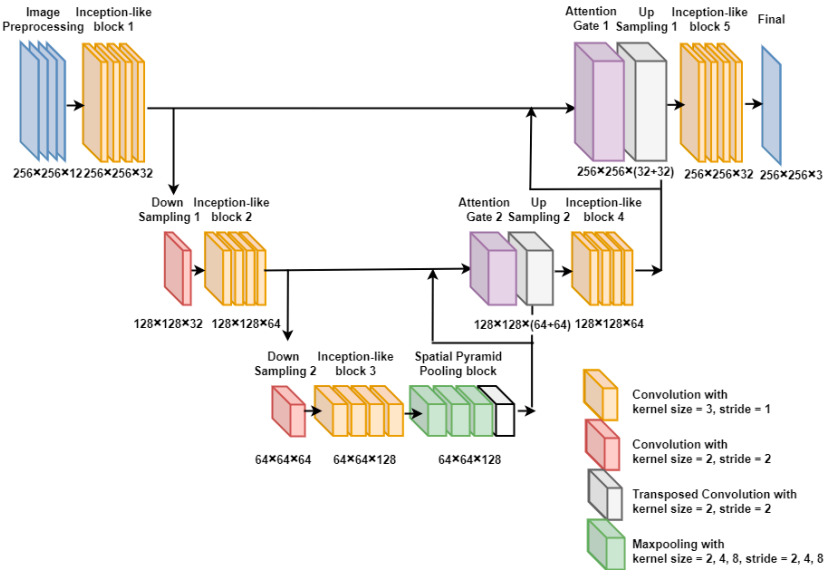

Figure 1. The proposed image-dehazing CNN architecture.

3.1. Image Pre-Processing Block

Haze will make the overall image whiter, that is, three RGB channel values will concentrate on 255 for an 8-bit image and result in a blurred and unclear object structure. Therefore, a normalization procedure, as shown in Equation (4), is applied to stretch the original RGB value linearly and make the image clearer.

$$f_{norm}(x_{i,c}) = \frac{x_{i,c} - \min(x_c)}{\max(x_c) - \min(x_c)}, \quad (4)$$

where $x_{i,c}$ is the input, i is the spatial dimension, c is the channel dimension referring to the RGB channels, and $\max(x_c)$ and $\min(x_c)$ denote the maximum and minimum values, respectively.

By mapping the normalized RGB image to HSV, YCbCr, and Lab, 12-channel images are formed. It is beneficial to improve the quality of image dehazing. The mapping operations to different color spaces only require simple mathematical operations without additional trainable parameters, thus saving storage space. Equation (5) denotes the output of the image preprocessing block.

$$I_{12} = \left[R_{h,w}, G_{h,w}, B_{h,w}, H_{h,w}, S_{h,w}, V_{h,w}, Y_{h,w}, Cb_{h,w}, Cr_{h,w}, L_{h,w}, a_{h,w}, b_{h,w}\right] \quad (5)$$

where h and w are the dimensions of image height and width, respectively, and I_{12} is the normalized image comprising 12 channels. RGB denotes red, green, and blue values. HSV denotes hue, saturation, and brightness values. YCbCr denotes the luminance, blue difference, and red difference. Lab denotes lightness, red/green values, and blue/yellow values.

3.2. Inception-like Block

Convolution was used to extract image features to form feature maps. These feature maps represent the high-dimensional images. Feature maps obtained through convolution filters usually go through nonlinear operators after convolution, such as a rectified linear unit (ReLU) or sigmoid function, to increase the nonlinear characteristics of the neural network.

Szegedy et al. [24] proposed that the receptive fields of two 3×3 convolutions are equivalent to the receptive field of one 5×5 convolution. Similarly, the receptive fields of the three 3×3 convolutions are equivalent to the receptive field of one 7×7 convolution. One 7×7 convolution is 5.44 (49/9) times more computationally expensive than one 3×3 convolution, and three 3×3 convolutions are 3 (27/9) times more computationally expensive than one 3×3 convolution. Therefore, to increase the receptive field while also reducing the computational complexity, we use two 3×3 convolutions instead of one 5×5 convolution, and the same applies to 7×7 and 9×9 convolutions which inspired by MultiResUnet [25]. Figure 2 shows the overall inception-like block that includes four convolutions in series and one residual connection. The outputs of these convolutions are concatenated, and to increase the effectiveness of training, a 1×1 convolution is added as the residual connection.

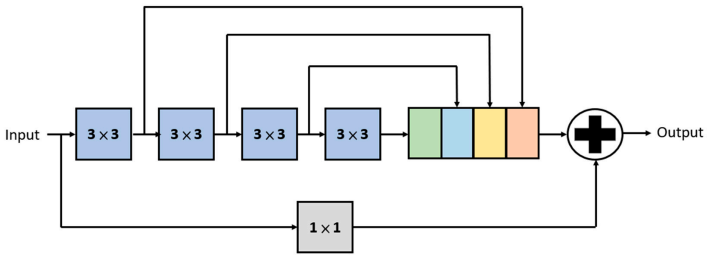

Figure 2. An illustration of an inception-like block.

3.3. SPP Block

Maximum pooling was used to extract the obvious feature values in the feature maps. He et al. [26] used different pooling kernel sizes to extract feature values from the same feature map, from coarse to fine, to generate feature maps with different resolutions. In this way, without increasing the trainable parameters, it helps the shallower network obtain deeper network layers, thereby enhancing the effect of deep learning training. The SPP block in this study uses multi-scale maximum pooling with kernel sizes of 8×8, 4×4, and 2×2 to extract features from coarse to fine. Because the resolution of the feature map shrinks after the maximum pooling, the results of each maximum pooling need to be upsampled by bilinear interpolation to the same resolution as the original feature map before they can be concatenated. Figure 3 shows the overall SPP block, which includes multiscale maximum pooling operations to extract features from coarse to fine.

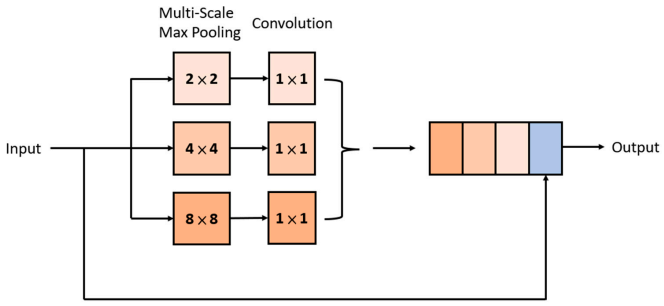

Figure 3. An illustration of an SPP block.

3.4. Attention Gate

The original U-Net feature fusion involves directly concatenating the output of the transposed convolution and the output of the encoder with the same resolution, while Oktay et al. [27] proposed Attention U-Net to replace the direct concatenation part with the attention gate; that is, the same spatial dimension of the feature maps needs to be multiplied by the same attention coefficient between 0 and 1, as in Equation (6). This attention mechanism is beneficial for automatically highlighting salient regions for semantic segmentation.

$$\hat{x}_{i,c}^{l} = \alpha_i^l \times x_{i,c}^{l}, 1 \geq \alpha_i^l \geq 0, \tag{6}$$

where i is the spatial dimension, c is the channel dimension, l is the layer, $x_{i,c}^{l}$ is the input, α_i^l is the attention coefficient, and $\hat{x}_{i,c}^{l}$ is the output.

Due to different task orientations, to pay more attention to the unclear region of the hazy image, the values of the same spatial dimension are multiplied by different attention coefficients, as follows:

$$\hat{x}_{i,c}^{l} = \alpha_{i,c}^{l} \times x_{i,c}^{l}, \ \alpha_{i,c}^{l} \geq 0, \tag{7}$$

where $\alpha_{i,c}^{l}$ is positive, which differs from Attention U-Net [27].

The details of the attention gate are shown in Figure 4, where the attention coefficients are used to pay more attention to the unclear region of the hazy image during back-propagation. The difference between the original attention coefficient and the proposed attention coefficient is shown in Figure 5, where the first half was proposed by Attention U-net [27] for semantic segmentation. The second half is proposed in this study for single-image dehazing. The attention coefficient is determined by the gate signal and the output of a certain layer of the encoder. The gate signal is the result of the transposed convolution of the decoder. For example, in Figure 1, the output of inception-like block 1 is the input of the attention gate, and inception-like block 4, after the transposed convolution, is the gate signal.

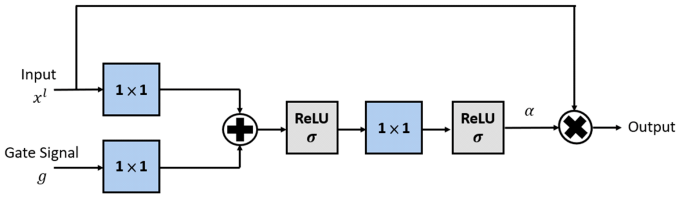

Figure 4. An illustration of an attention gate.

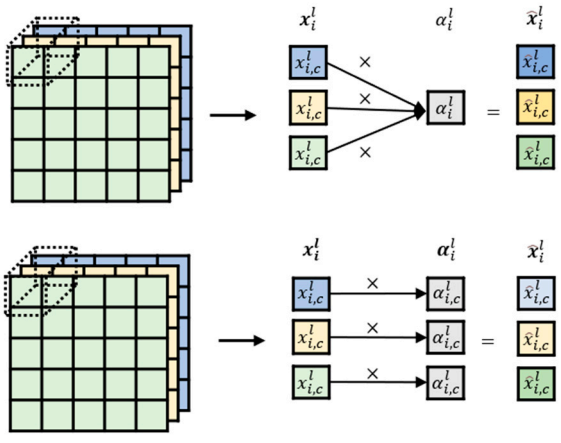

Figure 5. Schematic diagram of an attention gate with single and multiple attention coefficients.

The attention coefficient is calculated using Equation (8). The activation function adopts the ReLU function to ensure that the attention coefficient is greater than 0.

$$\alpha_i^l, = \sigma(\psi^T(\sigma(W_x^T x_i^l + W_g^T g_i^l + b_g)) + b_\psi), \tag{8}$$

where x_i^l is the input, σ is the ReLU function, $W_x \in \mathbb{R}^{C_l \times C_{int}}$, $W_g \in \mathbb{R}^{C_g \times C_{int}}$, C_l is the number of channels in the input feature map, C_g is the number of channels in output feature map, C_{int} is the number of channels in the intermediate feature map, $\psi \in \mathbb{R}^{C_{int} \times C_l}$, and b_g and b_ψ are biases.

Figure 6 shows that the attention gate calls attention to objects of distant buildings, ground textures, trees, and pedestrians that are blurred due to haze. The feature map after the attention gate changes the gain by the attention coefficient to achieve the effect of focusing on objects with complex structures and suppressing objects with relatively simple structures, such as focusing on the characteristics of pedestrians and suppressing the characteristics of the sky.

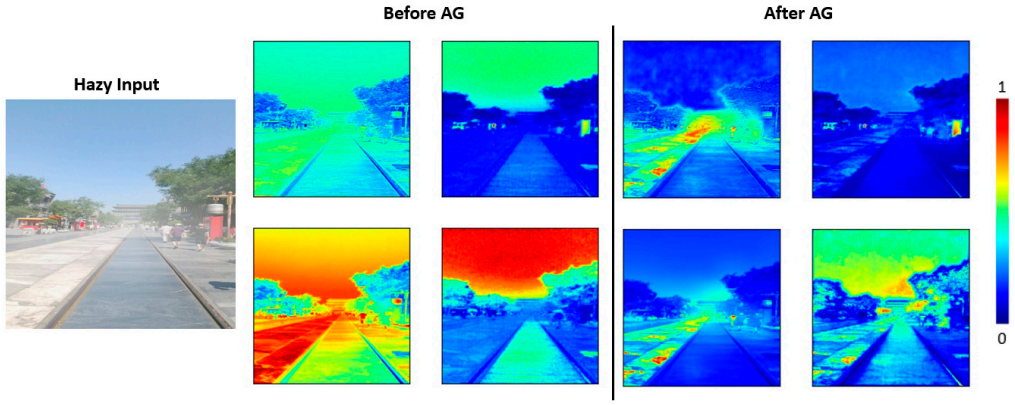

Figure 6. The effect of selecting four random feature maps before and after passing through AG 1.

3.5. Loss Function

The loss function setting of the image-dehazing CNN usually adopts the mean square error (*MSE*), as follows:

$$L_{MSE}(P) = \frac{1}{N}\sum_{p \in P}(x(p) - y(p))^2, \tag{9}$$

Treating *MSE* as a loss function means that only the difference in pixel values between the dehazing image and clear image is considered during training. The *MSE* is related to the peak signal-to-noise ratio (*PSNR*).

$$PSNR = 20\log_{10}(\frac{MAX}{\sqrt{MSE}}), \tag{10}$$

where MAX is the maximum possible pixel value. In the case of an 8-bit image, MAX is 255.

Considering that using only *MSE* may lead to visible speckle artifacts, this study introduces the structural similarity (*SSIM*) to reveal the degree of similarity between dehazing and clear images, including luminance, contrast, and structure, which can improve the learning efficiency and quality of image dehazing. When the *SSIM* is closer to 1, the two images are more similar.

$$SSIM(P) = l(x,y)^\alpha \cdot c(x,y)^\beta \cdot s(x,y)^\gamma, \tag{11}$$

where $\alpha = \beta = \gamma = 1$, $l(x,y)$ is the similarity of luminance, $c(x,y)$ is the similarity of contrast, and $s(x,y)$ is the similarity of structures. The details of $l(x,y)$, $c(x,y)$, and $s(x,y)$ are as shown in Equation (12).

$$\begin{cases} l(p) = \dfrac{2\mu_x\mu_y + C_1}{\mu_x^2 + \mu_y^2 + C_1} \\ c(p) = \dfrac{2\sigma_x\sigma_y + C_2}{\sigma_x^2 + \sigma_y^2 + C_2}, \\ s(p) = \dfrac{\sigma_{xy} + C_3}{\sigma_x\sigma_y + C_3} \end{cases} \tag{12}$$

where μ_x and μ_y are the mean values of the signals x and y, respectively, σ_x and σ_y are the standard deviations of the signals x and y, and C_1, C_2, and C_3 are constant values.

In this study, the loss function is used in combination with the *MSE* and *SSIM*, as follows:

$$Loss = 0.2 \cdot L_{MSE} + 0.8 \cdot L_{SSIM}, \tag{13}$$

In addition to the pixel value difference between the image generated by the proposed network and the ground truth, the difference in the similarity between image patches is also considered.

4. Experiments and Analysis

In this section, image-dehazing experiments are conducted. First, we determine the trainable parameters of the image-dehazing CNN. We then introduce the image-dehazing dataset used for training and testing the image-dehazing CNN. A comparison of the resulting performance with that of other state-of-the-art models by using the image quality metrics is provided. Figure 7 shows the workflow of the proposed-dehazing model generation.

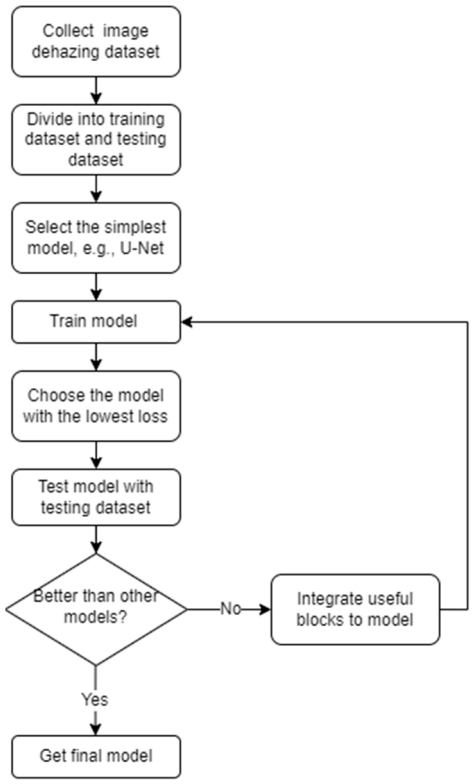

Figure 7. The workflow of the proposed dehazing-model generation.

4.1. Parameters of Neural Network

The input image specification of the dehazing CNN is $12 \times 256 \times 256$ after resizing the original image. An image pre-processing block is used to obtain a pre-processed image with 12 channels. The internal parameters of the proposed image-dehazing CNN are listed in Table 1, including five inception-like blocks, two attention gates, two downsamplings, and two upsamplings. The dehazing output image is an RGB image; therefore, there are three convolutions at the end of the neural network.

Table 1. Parameters of proposed network.

Blocks	Layers	Num
Inception-like block 1 and 5	Conv2D(3,3) [1]	8
	Conv2D(3,3)	8
	Conv2D(3,3)	8
	Conv2D(3,3)	8
	Conv2D(1,1)	32
Inception-like block 2 and 4	Conv2D(3,3)	16
	Conv2D(3,3)	16
	Conv2D(3,3)	16
	Conv2D(3,3)	16
	Conv2D(1,1)	64
Inception-like block 3	Conv2D(3,3)	32
	Conv2D(3,3)	32
	Conv2D(3,3)	32
	Conv2D(3,3)	32
	Conv2D(1,1)	128
Pyramid pooling block	Conv2D(1,1)	32
	Conv2D(1,1)	32
	Conv2D(1,1)	32
	Conv2D(1,1)	32
Attention gate 1	Conv2D(1,1)	32
	Conv2D(1,1)	32
	Conv2D(1,1)	32
	Conv2D(1,1)	32
Attention gate 2	Conv2D(1,1)	64
	Conv2D(1,1)	64
	Conv2D(1,1)	64
	Conv2D(1,1)	64
Downsampling 1	Conv2D(2,2)	32
Downsampling 2	Conv2D(2,2)	64
Upsampling 1	ConvT2D(2,2) [2]	32
Upsampling 2	ConvT2D(2,2)	64
Final convolution	Conv2D(1,1)	3

[1] Convolution with kernel size = 3 and stride = 1. [2] Transposed convolutions with kernel size = 2 and stride = 2.

The size of the trainable parameters of the proposed image-dehazing CNN affects the time required for training and inference speed. Although a larger number of trainable parameters can improve the image-dehazing quality, the number of trainable parameters should not be too large for real-time dehazing. Table 2 shows a comparison of the number

of trainable parameters with the state-of-the-art model. Although FFA-Net [12] has outstanding dehazing quality, it has too many trainable parameters to meet a lightweight model and will not be applied in the proposed approach.

Table 2. Comparison of trainable parameters.

Methods	Trainable Parameters	Model Size
DehazeNet [9]	8.305 K	0.035 MB
AOD-Net [10]	1.76 K	0.008 MB
ReViewNet [28]	399.7 K	5 MB
FFA-Net [12]	4455.9 K	21.3 MB
TheiaNet [20]	157.9 K	1.98 MB
Proposed Method	207.3 K	0.86 MB

4.2. Image-Dehazing Datasets

Constructing a large-scale real image-dehazing dataset is not easy because concurrently acquiring a hazy image and a haze-free image in the same place is not possible. Therefore, a large-scale dataset named RESIDE (REalistic Single-Image DEhazing) was proposed by Li et al. [29]. In RESIDE, data are presented as synthetic and real hazy images that correspond to clear images. The RESIDE Indoor Training Set (ITS) contains 1399 clear images, each of which contains 10 images with varying degrees of haze, for a total of 13,990 hazy images. The RESIDE Outdoor Training Set (OTS) contains 2061 clear images, and each clear image synthesizes 35 images with different degrees of haze to obtain a total of 72,135 hazy images. In addition, the testing dataset of RESIDE includes a synthetic objective testing set (SOTS) and a hybrid subjective testing set (HSTS).

In this study, RESIDE ITS and RESIDE OTS were used to train indoor and outdoor image-dehazing CNNs, while RESIDE SOTS and RESIDE HSTS were used to test the image-dehazing quality. The distribution of the dataset is shown in Table 3.

Table 3. Distribution of the dataset.

Datasets	Training Dataset		Testing Dataset	
Categories	**Indoor**	**Outdoor**	**Indoor**	**Outdoor**
RESIDE ITS	13,990	-	-	-
RESIDE OTS	-	72,135	-	-
RESIDE SOTS	-	-	500	500
RESIDE HSTS	-	-	-	10

4.3. Ablation Study

It is crucial to discuss the influence of each neural network block on the dehazing quality; therefore, we trained six different models to observe the performance of these blocks. Table 4 lists the details of the six models.

Table 4. Description of ablation analysis.

Blocks	Model 1	Model 2	Model 3	Model 4	Model 5	Model 6 (Proposed)
Image pre-processing block	-	-	-	-	O	O
Inception-like block	-	O	O	O	O	O
Pyramid pooling block	-	-	O	O	O	O
Original attention gate	-	-	-	O	O	-
Proposed attention gate	-	-	-	-	-	O

Table 5 shows the dehazing quality metrics of the six models and their corresponding numbers of trainable parameters. From model 2 and model 3, the inception-like block and the pyramid pooling improve the dehazing quality. Model 4 shows that an attention gate can increase *SSIM* and *PSNR* without adding too many trainable parameters, and model 5 uses the image preprocessing block without adding trainable parameters. Model 6 uses the proposed attention gate instead of the original attention gate, resulting the best dehazing quality among these approaches.

Table 5. Result of ablation analysis on RESIDE SOTS outdoor dataset.

Methods	*SSIM*	*PSNR*	Trainable Parameters
Model 1	0.9587	25.04	123.539 K
Model 2	0.9602	26.01	175.107 K
Model 3	0.9644	26.50	191.619 K
Model 4	0.9673	27.31	196.885 K
Model 5	0.9718	28.64	196.885 K
Model 6	0.9736	29.15	207.267 K

4.4. Performance and Discussion

In the study, PyTorch was applied for developing the deep learning architecture with a central processing unit (CPU) of a 3.00 GHz Intel Core i5-8500 CPU. The GPU processor used a GeForce GTX 1660 Super 6G for training, the batch size was set to 8, and the loss function of Equation (13) was adopted. A learning rate of 0.0001 and 150 epochs were used for the Adam optimizer [30]. A total of 90% of the training dataset was used for training, and 10% was used for verification, which can be used to confirm the loss trend and ensure that the model does not overfit.

The image-dehazing CNN results were analyzed and compared using two metrics, namely, *PSNR* and *SSIM*, introduced in the previous section. Table 6 compares the image quality metrics of the different methods applied to the RESIDE SOTS indoor and outdoor testing datasets. The datasets contain images of various scenes with different haze concentrations. From Table 6, the proposed image-dehazing CNN leads to better results than the prior-based methods and learning-based methods that have been proposed in recent years. Compared to the latest approach, TheiaNet [20], the average *SSIM* of the indoor dehazing results increased by 5.65%, and the average *PSNR* increased by 7.73%, whereas the average *SSIM* of the outdoor dehazing results increased by 2.83%, and the average *PSNR* increased by 14.09%.

Table 6. Comparison of dehazing results on RESIDE SOTS dataset.

Categories	Indoor		Outdoor	
Method	***SSIM* (% inc)**	***PSNR* (%inc)**	***SSIM* (% inc)**	***PSNR* (% inc)**
DCP [4]	0.8179 (17.75)	16.62 (66.79)	0.8148 (19.49)	19.13 (52.38)
CAP [5]	0.8364 (15.15)	19.05 (45.51)	0.8514 (14.35)	22.46 (29.79)
DehazeNet [9]	0.8472 (13.68)	21.14 (31.13)	0.8630 (12.82)	22.57 (29.15)
AOD-Net [10]	0.8504 (13.25)	19.06 (45.44)	0.8765 (11.07)	20.29 (43.67)
HIDeGAN [17]	0.8680 (10.96)	24.71 (12.18)	0.8780 (10.89)	25.54 (14.13)
ReViewNet [28]	0.8946 (7.66)	23.61 (17.41)	0.9137 (6.56)	23.64 (23.31)
TheiaNet [20]	0.9116 (5.65)	25.73 (7.73)	0.9468 (2.83)	25.55 (14.09)
Proposed Method	0.9631 (best)	27.72 (best)	0.9736 (best)	29.15 (best)

Table 7 compares the image quality metrics of the different methods applied to the RESIDE HSTS testing dataset. Compared to TheiaNet [20], the average SSIM of the dehazing results increased by 2.47%, and the average PSNR increased by 8.09%.

Table 7. Comparison of dehazing results on RESIDE HSTS dataset.

Model	*SSIM* (% inc)	PSNR (% inc)
DCP [4]	0.7609 (29.36)	14.84 (120.69)
CAP [5]	0.8726 (12.75)	21.53 (52.11)
HIDeGAN [17]	0.8940 (10.10)	28.04 (16.80)
AOD-Net [10]	0.8973 (9.70)	20.55 (59.37)
DehazeNet [9]	0.9153 (7.54)	24.48 (33.78)
ReViewNet [28]	0.9582 (2.72)	27.50 (19.09)
TheiaNet [20]	0.9606 (2.47)	30.30 (8.09)
Proposed Method	0.9843 (best)	32.75 (best)

To estimate the average execution time per frame, 500 hazy images with the original resolution of 620×460 were processed by the trained dehazing CNN. It only took 0.018 s per frame, on average, through the GPU, which fulfills the real-time application requirement, and it took only 0.24 s, on average, through the CPU. Tables 8 and 9 compare the average execution time per frame of the dehazing models by ReViewNet [28] and TheiaNet [20]. Since DehazeNet [9] and AOD-Net [10] did not produce satisfactory dehazing quality, as listed in Table 6, the corresponding average execution times of both methods are not included in Tables 8 and 9.

Table 8. Comparison of average execution time per frame on CPU.

Model	ReViewNet [28]	TheiaNet [20]	Proposed Method
CPU	3.6 GHz	3.6 GHz	3.0 GHz
AET [1]	0.28 s	0.247 s	0.244 s

[1] Average execution time.

Table 9. Comparison of average execution time per frame on GPU.

Model	ReViewNet [28]	TheiaNet [20]	Proposed Method
GPU	NVIDIA TITAN V 12 GB	NVIDIA TITAN V 12 GB	GTX 1660 Super 6 GB
AET	0.025 s	0.018 s	0.018 s

Some comparisons between the dehazing results of the RESIDE SOTS indoor and outdoor testing datasets and ground truth are demonstrated as follows. Figure 8 illustrates the indoor dehazing results. The box indicates the key recovery area where the image is not clear due to the haze. It can be seen from the results of the image dehazing that people and furniture in the distance were affected by color deviation and low contrast were well-restored and were almost the same as the ground truth. Similarly, Figure 9 shows the restoration of traffic signs, cars, and sky regions blurred by haze in the outdoor environment. AOD-Net [10] and DehazeNet [9] cannot recover the color deviation or the contrast well. ReViewNet [28] generates a darker image than ground truth and cannot restore the appearance of the no parking sign. The proposed method effectively removes the haze on buildings, sky, and traffic signs. The same dehazing effect was also achieved for images with buildings and trees, as shown in Figure 10. It shows that ReViewNet [28] generates a darker dehazing result due to a color deviation problem. As illustrated in Figure 11, removing haze for objects far in the distance is a limitation of TheiaNet [20]. Obviously, the proposed method generates a clearer dehazing result for distant buildings than the other models. For a real hazy image, the proposed method achieves a visually pleasing result compared to those of the other models, as shown in Figure 12.

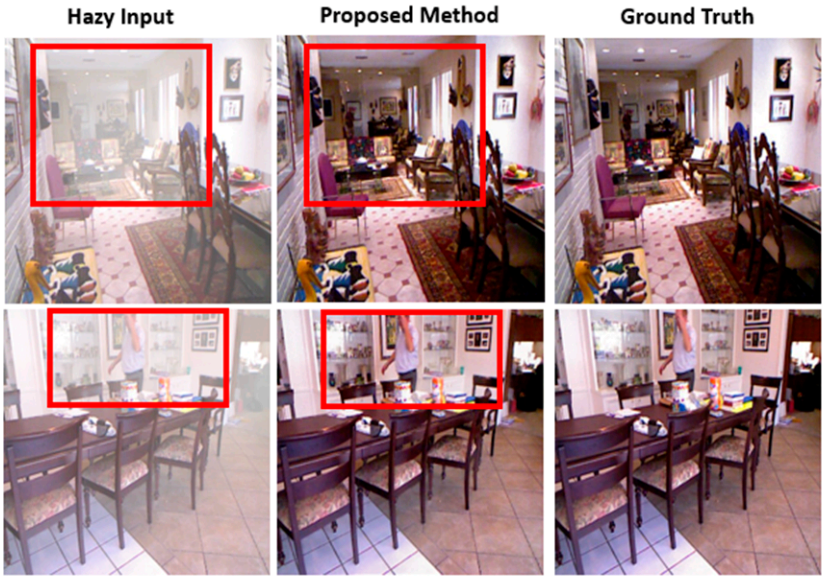

Figure 8. Comparison of dehazing results from RESIDE SOTS indoor testing dataset with ground truth.

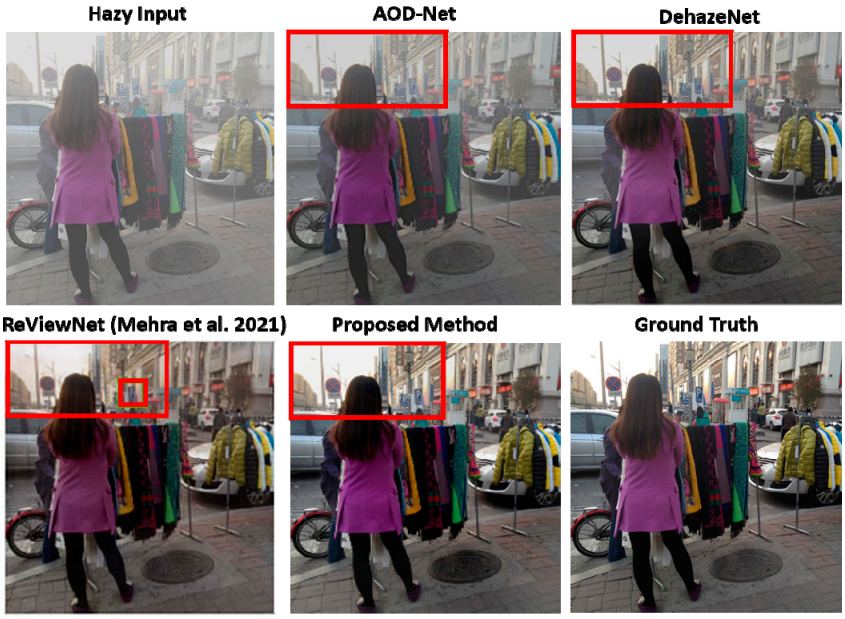

Figure 9. Comparison of dehazing results of traffic signs, cars, and sky region [28].

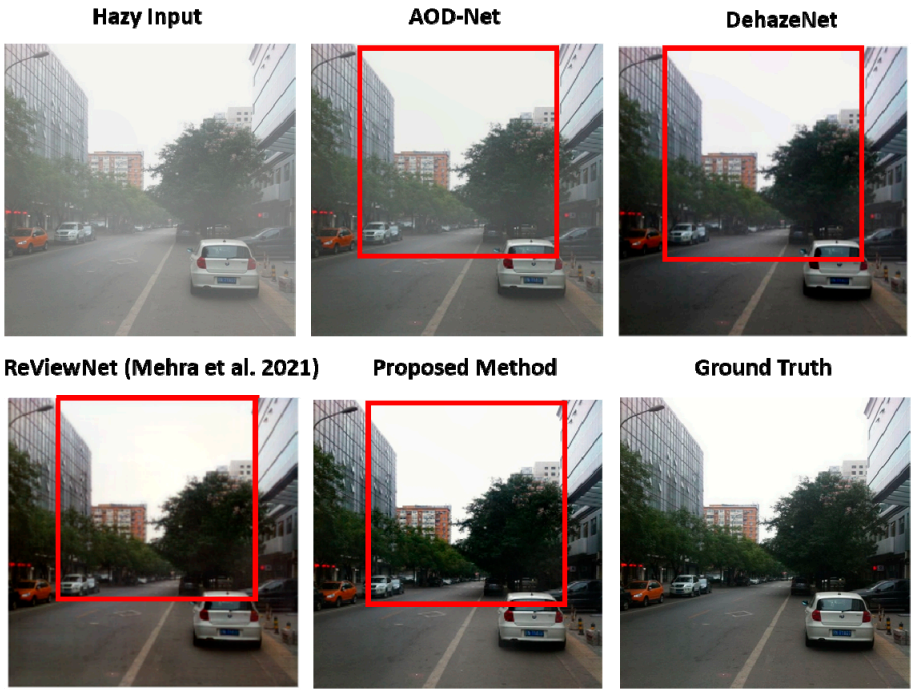

Figure 10. Comparison of dehazing results of urban road [28].

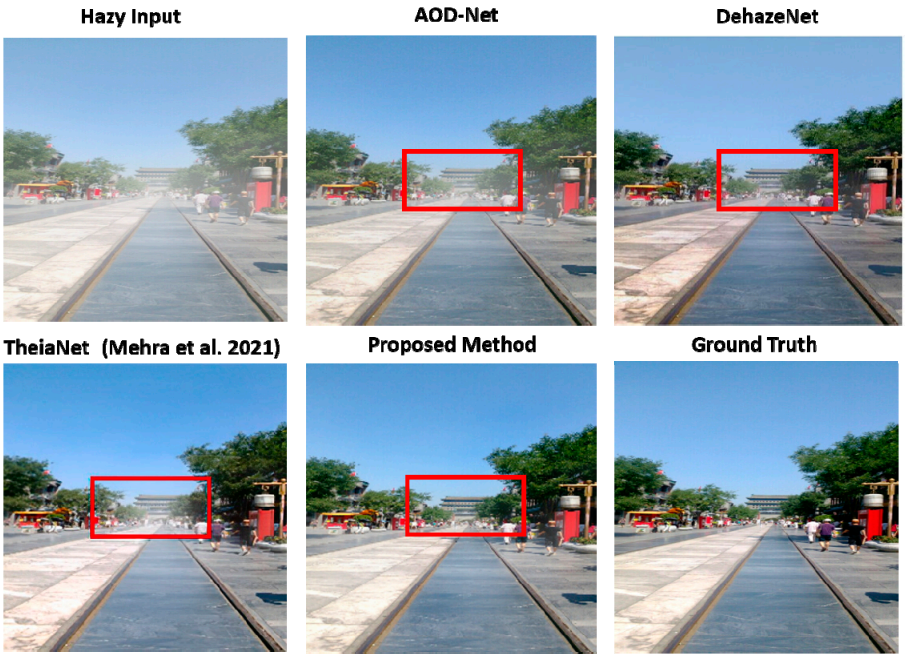

Figure 11. Comparison of dehazing results of distant haze [20].

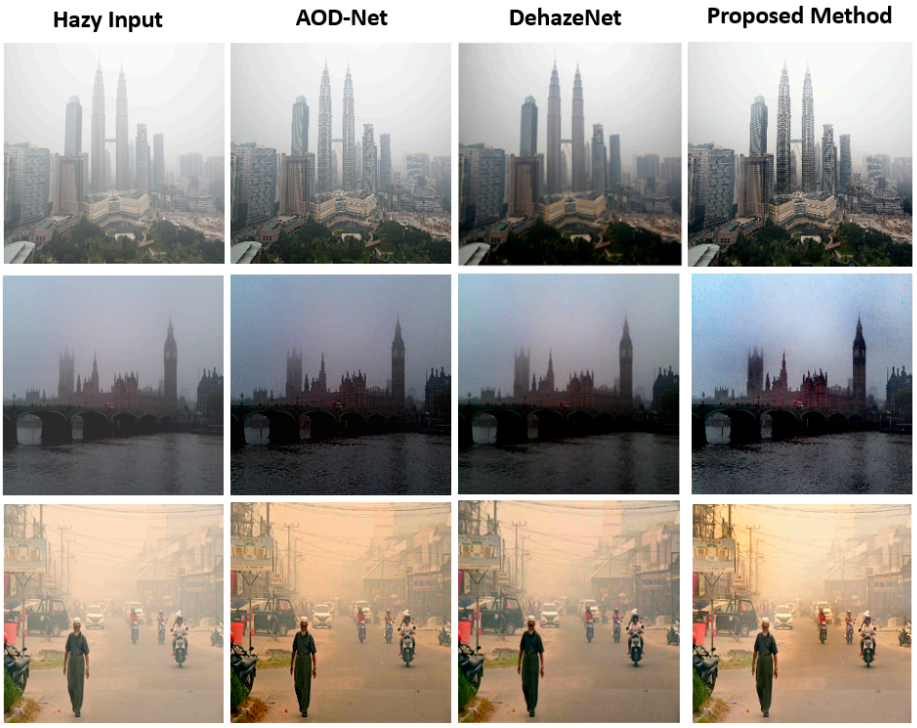

Figure 12. Comparison of dehazing results of real hazy image.

5. Conclusions

The pros and cons of the image-dehazing methods in recent years as well as the proposed method are summarized as follows. The DCP and CAP dehazing methods based on statistical experience are training-free methods. They obtain a certain degree of image-dehazing quality, but the resulting quality is poor compared to other approaches. AOD-Net uses a simple CNN architecture to learn a function composed of the transmission map and atmospheric light. Although the model is lightweight, efforts are still required to improve its dehazing quality. Without requiring knowledge of the physical model, ReViewNet and TheiaNet use the U-Net architecture as a prototype and integrate the appropriate CNN blocks to produce quality dehazing images. However, the dehazing quality for objects far in the distance may become a drawback of using TheiaNet. Based on the U-Net architecture, the proposed approach integrates image preprocessing blocks, inception-like blocks, SPP blocks, and AGs to achieve a better image-dehazing quality, even for images with sky and objects far in the distance. For a real hazy image, it also produces clearer dehazing results. The proposed model can be considered as a lightweight and real-time model for dehazing applications.

Author Contributions: Conceptualization, C.-Y.T. and C.-L.C.; methodology, C.-Y.T. and C.-L.C.; software, C.-Y.T.; validation, C.-L.C.; formal analysis, C.-Y.T.; data curation, C.-Y.T.; writing—original draft preparation, C.-Y.T.; writing—review and editing, C.-L.C.; supervision, C.-L.C.; funding acquisition, C.-L.C. All authors have read and agreed to the published version of the manuscript.

Funding: This study was funded by the Ministry of Science and Technology, Taiwan, under grant No. MOST 107-2221-E-006-119.

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. Gui, J.; Cong, X.; Cao, Y.; Ren, W.; Zhang, J.; Zhang, J.; Tao, D. A Comprehensive Survey on Image Dehazing Based on Deep Learning. *arXiv* **2021**, arXiv:2106.03323.
2. Cozman, F.; Krotkov, E. Depth from scattering. In Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, San Juan, PR, USA, 17–19 June 1997.
3. Tan, R.T. Visibility in bad weather from a single image. In Proceedings of the 2008 IEEE Conference on Computer Vision and Pattern Recognition, Anchorage, AK, USA, 23–28 June 2008.
4. He, K.; Sun, J.; Tang, X. Single Image Haze Removal Using Dark Channel Prior. *IEEE Trans. Pattern Anal. Mach. Intell.* **2011**, *33*, 2341–2353.
5. Zhu, Q.; Mai, J.; Shao, L. A Fast Single Image Haze Removal Algorithm Using Color Attenuation Prior. *IEEE Trans. Image Process.* **2015**, *24*, 3522–3533.
6. Berman, D.; Treibitz, T.; Avidan, S. Non-local Image Dehazing. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016.
7. Ju, M.; Ding, C.; Ren, W.; Yang, Y.; Zhang, D.; Guo, Y.J. IDE: Image Dehazing and Exposure Using an Enhanced Atmospheric Scattering Model. *IEEE Trans. Image Process.* **2021**, *30*, 2180–2192.
8. Yang, G.; Evans, A.N. Improved single image dehazing methods for resource-constrained platforms. *J. Real-Time Image Process.* **2021**, *18*, 2511–2525.
9. Cai, B.; Xu, X.; Jia, K.; Qing, C.; Tao, D. Dehazenet: An end-to-end system for single image haze removal. *IEEE Trans. Image Process.* **2016**, *25*, 5187–5198.
10. Li, B.; Peng, X.; Wang, Z.; Xu, J.; Feng, D. AOD-Net: All-in-One Dehazing Network. In Proceedings of the 2017 IEEE International Conference on Computer Vision (ICCV), Venice, Italy, 22–29 October 2017.
11. Zhang, J.; Tao, D. FAMED-Net: A fast and accurate multi-scale end-to-end dehazing network. *IEEE Trans. Image Process.* **2019**, *29*, 72–84.
12. Qin, X.; Wang, Z.; Bai, Y.; Xie, X.; Jia, H. FFA-Net: Feature fusion attention network for single image dehazing. In Proceedings of the AAAI Conference on Artificial Intelligence, New York, NJ, USA, 7–12 February 2020.
13. Liu, Z.; Xiao, B.; Alrabeiah, M.; Wang, K.; Chen, J. Single Image Dehazing with a Generic Model-Agnostic Convolutional Neural Network. *IEEE Signal Process. Lett.* **2019**, *26*, 833–837.
14. Engin, D.; Genç, A.; Ekenel, H.K. Cycle-dehaze: Enhanced cyclegan for single image dehazing. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, Salt Lake City, UT, USA, 18–22 June 2018.
15. Zhu, J.-Y.; Park, T.; Isola, P.; Efros, A.A. Unpaired image-to-image translation using cycle-consistent adversarial networks. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017.
16. Qu, Y.; Chen, Y.; Huang, J.; Xie, Y. Enhanced Pix2pix Dehazing Network. In Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 15–20 June 2019.
17. Mehta, A.; Sinha, H.; Narang, P.; Mandal, M. HIDeGan: A Hyperspectral-guided Image Dehazing GAN. In Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), Seattle, WA, USA, 14–19 June 2020.
18. Mirza, M.; Osindero, S. Conditional generative adversarial nets. *arXiv* **2014**, arXiv:1411.1784.
19. Ronneberger, O.; Fischer, P.; Brox, T. U-Net: Convolutional Networks for Biomedical Image Segmentation. In Medical Image Computing and Computer-Assisted Intervention—MICCAI 2015, Munich, Germany, 5–9 October 2015; Springer International Publishing: Cham, Switzerland.
20. Mehra, A.; Narang, P.; Mandal, M. TheiaNet: Towards fast and inexpensive CNN design choices for image dehazing. *J. Vis. Commun. Image Represent.* **2021**, *77*, 103137.
21. Ren, W.; Ma, L.; Zhang, J.; Pan, J.; Cao, X.; Liu, W.; Yang, M.-H. Gated fusion network for single image dehazing. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018.
22. Wan, Y.; Chen, Q. Joint image dehazing and contrast enhancement using the HSV color space. In Proceedings of the 2015 Visual Communications and Image Processing (VCIP), Singapore, 13–16 December 2015.
23. Tufail, Z.; Khurshid, K.; Salman, A.; Nizami, I.F.; Khurshid, K.; Jeon, B. Improved Dark Channel Prior for Image Defogging Using RGB and YCbCr Color Space. *IEEE Access* **2018**, *6*, 32576–32587.
24. Szegedy, C.; et al. Rethinking the inception architecture for computer vision. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016.
25. Ibtehaz, N.; Rahman, M.S. MultiResUNet: Rethinking the U-Net architecture for multimodal biomedical image segmentation. *Neural Netw.* **2020**, *121*, 74–87.
26. He, K.; Zhang, X.; Ren, S.; Sun, J. Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **2015**, *37*, 1904–1916.
27. Oktay, O.; Schlemper, J.; Folgoc, L.L.; Lee, M.; Heinrich, M.; Misawa, K.; Mori, K.; McDonagh, S.; Hammerla, N.Y.; Kainz, B.; et al. Attention u-net: Learning where to look for the pancreas. *arXiv* **2018**, arXiv:1804.03999.
28. Mehra, A.; Mandal, M.; Narang, P.; Chamola, V. ReViewNet: A Fast and Resource Optimized Network for Enabling Safe Autonomous Driving in Hazy Weather Conditions. *IEEE Trans. Intell. Transp. Syst.* **2021**, *22*, 4256–4266.

29. Li, B.; Ren, W.; Fu, D.; Tao, D.; Feng, D.; Zeng, W.; Wang, Z. RESIDE: A Benchmark for Single Image Dehazing. *arXiv* **2017**, arXiv:1712.04143.
30. Kingma, D.P.; Ba, J. Adam: A method for stochastic optimization. *arXiv* **2014**, arXiv:1412.6980.