IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received April 3, 2020, accepted April 18, 2020, date of publication April 21, 2020, date of current version May 6, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2989355

RASWNet: An Algorithm That Can Remove All Severe Weather Features from a Degraded Image

LIN GAO[1,2], (Member, IEEE), WEI LONG[1], YANYAN LI[1], HUAGUO LIU[1], XIAOHONG YU[1], AND JUN LI[2]

[1]School of Mechanical Engineering, Sichuan University, Chengdu 610065, China
[2]School of Information Engineering, Hubei Minzu University, Enshi 445000, China

Corresponding author: Yanyan Li (yyl_scu@163.com)

This work was supported by the National Natural Science Foundation of China under Grant 51875371, Grant 61562025, and Grant 61962019.

ABSTRACT The advanced driving assistant system (ADAS) is an important vehicle safety technology that can effectively reduce traffic accidents. This system can perceive information about the surrounding environment through in-vehicle cameras. However, these cameras are easily affected by severe weather conditions, such as those involving fog, rain, and snow. The quality of the images acquired by the system is degraded, and the function of the ADAS is thus weakened. In response to this problem, we propose a comprehensive imaging model that can represent the features of fog, rain streaks, raindrops and snowflakes in an image. Subsequently, an algorithm called RASWNet is proposed, which can remove all severe weather features from a degraded image. Based on the generative adversarial network, RASWNet combines the focus capture ability of a visual attention mechanism, the memory ability of the recurrent neural network and the feature extraction ability of the dense blocks approach. We verify the network structure through several ablation studies and use various synthetic and real images to test it. The results of these experiments show that our algorithm is not only better than the commonly used algorithms in terms of its clarity enhancement capacity but is also suitable for all severe weather conditions.

INDEX TERMS Generative adversarial network, remove all severe weather features, degraded image, RASWNet, visual attention mechanism.

I. INTRODUCTION

With the continuous increase in car ownership, the traffic safety situation is becoming increasingly serious. To improve driving safety, the ADAS market has been growing rapidly [1]–[4]. The four types of ADAS sensors are LIDAR, radar, cameras and ultrasonic sensors [1]. These sensors can detect the surrounding environment and obtain all types of information needed by the system. However, these sensors are costly, and they require continuous maintenance and complex synchronization for the fusion of different sources of data [1]. Because vision is the most important perception of human beings, similarly, the camera is the most important perception component of an ADAS. Thus, we use images collected by a low-cost in-vehicle monocular camera as our research object to remedy these limitations.

The associate editor coordinating the review of this manuscript and approving it for publication was Thomas Canhao Xu.

However, the ability of in-vehicle cameras to detect the surrounding environment is easily affected by severe weather conditions, such as fog, rain, and snow. For example, the presence of fog can considerably reduce the visibility and contrast of the images collected by the cameras in addition to blurring the details. Raindrops or snowflakes move and fall rapidly in the air, which will lead to partial occlusion or blurring of the images. More specifically, the raindrops that adhere to the windshield or camera lens will reflect light from other areas, thereby degrading the images [5]. Consequently, the degradation of images caused by fog, rain and snow will not only reduce the driver's response speed but also weaken the functioning of the ADAS.

Currently, there are many clarity enhancement algorithms for a single image degenerated by severe weather conditions, such as dehazing algorithms [6]–[9], rain streak removal algorithms [10]–[12], raindrop removal algorithms [5], [13] and snow removal algorithms [14], [41]. Simultaneously, there are some clarity enhancement algorithms for two types of

severe weather conditions, such as deraining and desnowing algorithms [15]–[18] and rain streaks and mist removal algorithms [33], [40]. If the algorithms are used in an ADAS, there are two challenges. The first challenge is how to recognize the current weather condition before using the algorithm. Weather recognition requires expensive equipment, which will strongly increase the cost of the car. The second challenge is that a weather recognition error will lead to a failure in the clarity enhancement. For example, if a rainy image is recognized as a foggy image, the defogging algorithm cannot remove the rain streaks and raindrops, and vice versa. In summary, we need an algorithm that can remove all of the severe weather features from a degraded image. The algorithm would be used not only in ADAS, driverless vehicles but also in intelligent monitoring, unmanned aerial vehicles (UAVs) and other fields.

In addition, for the images collected in severe weather conditions, researchers have proposed a variety of imaging models. Although these models are effective, there are still two problems: the first is the lack of uniform standards in addressing the masks of fog, rain streaks, raindrops and snowflakes; the second is the lack of a comprehensive severe weather imaging model. In summary, when we combine all types of degraded images collected in severe weather conditions together for processing, we must build a comprehensive imaging model. Hence, the contributions of our paper are as follows:

The first contribution is that we propose an algorithm called RASWNet, which can remove all severe weather features from a degraded image. Based on the generative adversarial network (GAN), it can use the dense blocks to extract the severe weather features, the visual attention mechanism to capture the regions of the features, and the recurrent neural network (RNN) to remember these regions. It can automatically locate and remove the fog, rain streaks, raindrops and snowflakes in an image, and it has excellent clarity enhancement results.

The second contribution is that we propose a comprehensive imaging model that reflects all types of severe weather features. The model unifies the masks of fog, rain streaks, raindrops and snowflakes, which not only conforms to the imaging situation of single severe weather conditions but also conforms to the imaging situation of multiple severe weather conditions.

This paper is organized as follows. Section I describes the research background and significance. Section II briefly reviews the related work. Section III describes the comprehensive imaging model. Section IV proposes RASWNet. Section V describes the datasets used. Section VI describes the experimental research, and Section VII provides the conclusions.

II. RELATED WORK

The research content of this paper involves image defogging, image deraining and desnowing, and severe weather imaging models. This section introduces the relevant research studies for these three aspects. The traditional defogging algorithm mainly uses image restoration technology based on a variety of prior studies [6]–[9], while the rain and snow removal algorithms are mainly based on image decomposition technology [19]–[21]. The results of these algorithms are generally not as good as those based on deep learning (DL). Consequently, the algorithms introduced in this section are based on DL.

A. IMAGE DEFOGGING

Image defogging algorithms based on DL involve the process of training a deep neural network to make the defogging image continuously approach the ground truth image. According to the different network structures, they can be divided into two types: defogging by using a convolutional neural network (CNN) and defogging by using a GAN.

The first is defogging by using a CNN. In 2016, Cai *et al.* [22] designed an end-to-end CNN model (DehazeNet). It took a hazy image as input and outputted its medium transmission map, which was subsequently used to recover a haze-free image via an atmospheric scattering model. In 2017, Li *et al.* [23] proposed an all-in-one network (AOD-Net) based on a CNN, which was a lightweight CNN model and could be easily embedded in an object detection algorithm. In 2018, Ren *et al.* [24] proposed an end-to-end gated fusion network (GFN), which was composed of an encoder and a decoder. The experimental results were better than those of other mainstream algorithms, but it could not remove thick fog. Ancuti *et al.* [25] collected two hazy image benchmark datasets for related research. The I-HAZE dataset contained 35 scenes corresponding to indoor domestic environments, with objects with different colors and specularities. O-HAZE contained 45 different outdoor scenes depicting the same visual content recorded in haze-free and hazy conditions under the same illumination parameters. Song *et al.* [26] proposed a novel Ranking-CNN. In this network, a novel ranking layer was proposed to extend the structure of CNN such that the statistical and structural attributes of hazy images could be simultaneously captured. In 2019, Yeh *et al.* [27] proposed a deep learning-based architecture (denoted by MSRL-DehazeNet) for single-image haze removal relying on multiscale residual learning (MSRL) and image decomposition. They reformulated the dehazing problem as restoration of the image base component.

The second is defogging by using a GAN. In 2018, Zhang *et al.* [28] proposed a densely connected pyramid dehazing network (DCPDN) that used a new edge-preserving densely connected encoder-decoder structure with a multilevel pyramid pooling module for estimating the transmission map. In 2019, Dudhane *et al.* [29] proposed a dehazing network by using a cycle-consistent GAN (CDNet), which consisted of an encoder-decoder architecture that was used to estimate the transmission map and restore the haze-free scene. Qu *et al.* [30] proposed an enhanced pix2pix dehazing network (EPDN). First, the discriminator guided the generator to create a pseudo realistic image on a coarse scale,

and then the enhancer following the generator was required to produce a realistic dehazing image on a fine scale.

In general, the defogging results of these algorithms are good. However, they can only be used to remove fog or haze from a single image, not rain or snow.

B. IMAGE DERAINING AND DESNOWING

Similar to the image defogging algorithms, the image deraining and desnowing algorithms can also be divided into two types: deraining and desnowing by using a CNN and deraining and desnowing by using a GAN.

The first type is deraining and desnowing by using a CNN. In 2017, Fu *et al.* [31] introduced a deep network architecture called DerainNet for removing rain streaks from an image. It had two characteristics: the network layers were not deep, and it was trained on a detailed (high-pass) layer. The experimental results showed that the algorithm was effective and fast. In 2018, Li *et al.* [32] proposed a nonlocally enhanced encoder-decoder network for single-image deraining, which was composed of a series of nonlocally enhanced dense blocks. It could not only remove rain streaks of various densities but also effectively preserve similar linear details. Liu *et al.* [14] proposed a context-aware deep network called DesnowNet to remove translucent and opaque snow particles. These researchers also differentiated the snow attributes of translucency and chromatic aberration for accurate estimation. In 2019, Yang *et al.* [33] proposed a joint rain detection and removal algorithm based on a CNN, which could remove a large number of rain streaks and mist via a contextual dilated network. The experimental results showed that the algorithm had a good effect on heavy rain images. Pei *et al.* [34] proposed a novel network architecture named multiweather network (MWNet), which could improve the performance of the on-board object detection system under extreme weather conditions. However, it could only recognize good weather and bad weather, and it could not enhance the clarity of the image. Ren *et al.* [35] proposed a progressive recurrent network (PReNet), which notably reduced network parameters with unsubstantial degradation in deraining performance. The experiments showed that the PReNet performed favorably on both synthetic and real rainy images. Wang *et al.* [36] proposed a novel spatial attentive network (SPANet) that could learn to identify and remove rain streaks in a local-to-global spatial attentive manner. Extensive evaluations demonstrated the superiority of the proposed method over the state-of-the-art derainers. Fu *et al.* [37] proposed a lightweight pyramid of networks (LPNet) for single-image deraining. By using the pyramid to simplify the learning problem and adopting recursive blocks to share parameters, LPNet had fewer than 8K parameters while still achieving good performance. In 2020, Jiang *et al.* [38] explored the multi-scale collaborative representation for rain streaks from the perspective of input image scales and hierarchical deep features in a unified framework, termed multi-scale progressive fusion network (MSPFN) for single image rain streak removal. Experimental results on several synthetic deraining datasets and real-world scenarios showed great superiority of their proposed MSPFN algorithm over other top-performing methods.

The second type of algorithm is deraining and desnowing by using a GAN. In 2018, Qian *et al.* [5] proposed an attention generative network (ATT-GAN) for raindrop removal from a single image, which used the visual attention mechanism to make the generative network learn the raindrop regions and their surroundings, and the discriminative network could evaluate the local consistency of the recovery area. In 2019, Zhang *et al.* [39] proposed an image-deraining conditional GAN (ID-CGAN) algorithm, which used a new loss function to reduce the artifacts produced by the GAN, and they designed a multiscale discriminator to improve the ability to distinguish real and fake images. Li *et al.* [40] proposed a 2-stage network: a physics-based backbone followed by a depth-guided GAN refinement. Extensive experiments showed that their method outperformed the state of the art algorithms on real rain image data, recovering visually clean images with good details. Li *et al.* [41] proposed a snow removal composition GAN (SR-CGAN), which comprised a clean background module and a snow mask estimation module. The former aimed to generate a clear image from an input snowy image, and the latter was used to produce the snow mask in an input image. The experiments showed that the snow removal results of this algorithm were better than those of other similar algorithms.

In summary, some of these algorithms can remove rain streaks [31], [32], [35]–[39], some can remove raindrops [5], some can remove snowflakes [14], [41], and some can remove rain streaks and mist [33], [40], but there is no algorithm that can remove fog, rain streaks, raindrops and snowflakes from an image.

C. SEVERE WEATHER IMAGING MODELS

In view of the various image degradation types in severe weather conditions, researchers have proposed many imaging models. These models are described as follows:

The first imaging model is for foggy images, and the commonly used atmospheric scattering model [30], [42], [28] is as follows:

$$I = t \odot B + A(1 - t) \tag{II.1}$$

where I is the degraded image by fog, B is the fog-free scene image, t is the medium transmission map, and A is the global atmospheric light value. Here, $\odot$ denotes elementwise multiplication, where $t \in [0, 1]$, $t = 0$ means that the fog concentration is maximum, the scene is completely invisible, and the image shows the atmospheric light value $I = A$; $t = 1$ means that there is no fog, the scene is completely visible, and $I = B$; the values from 0 to 1 indicate changes in the medium transmission, and the larger the value of t is, the higher the visibility of the scene is.

The second imaging model is for images with rain streaks; Li *et al.* [43] proposed the following model:

$$I = B + R \tag{II.2}$$

where I is the original image with rain streaks, B is the clean background scene, and R is the component image of the rain streaks. The clean background scene B can be obtained by subtracting the rain streaks component R from the image I.

The third imaging model is for images with snowflakes; Liu *et al.* [14] proposed the following model:

$$I = (1 - M_S) \odot B + M_S \odot S \tag{II.3}$$

where I represents the original image with snowflakes, B represents the snow-free image, and S represents the component image of the snowflakes. M_S is a snowflake mask, which indicates the transparency of the snowflakes in the image. Here, $M_S \in [0, 1]$. $M_S = 0$ means no snowflakes, the scene is completely visible, and $I = B$; $M_S = 1$ means that only snowflakes can be seen, the scene is completely invisible, and $I = S$; the values from 0 to 1 mean that the snowflakes are translucent, and the smaller the value is, the higher the visibility of the scene is.

The fourth imaging model is for images with raindrops; Qian *et al.* [5] proposed the following model:

$$I = (1 - M_D) \odot B + D \tag{II.4}$$

where I represents the original image with raindrops, B represents the background image, and D is the effect brought by the raindrops, which represents the complex mixture of the background information and the light reflected by the environment and passing through the raindrops that adhere to a lens or windscreen [5]. Here, M_D is a raindrop mask that represents the binary state of the raindrop region in the image. When $M_D = 1$, the background is completely invisible, and $I = D$; when $M_D = 0$, the raindrops overlay on the completely visible background, and I=B+D.

III. COMPREHENSIVE IMAGING MODEL

The four imaging models described in Section II.C are effective when processing image degradation caused by a certain severe weather condition, but there are two challenges when processing degraded images caused by combinations of various severe weather conditions.

The first problem is the lack of uniform standards in addressing the mask problems of fog, rain streaks, raindrops and snowflakes. The mask is the transparency of the image in the fog, rain streak, raindrop and snowflake regions, which is represented by t, M_R, M_D and M_S, respectively.

First, there is no rain streak mask M_R in (II.2) when processing an image with rain streaks. The authors [43] thought that, in the rain streak region, the pixel intensity of the background image B did not decrease when overlaid on rain streaks. However, it is weakened in the actual image, as shown in Fig. 1(a). The intensity of the red exterior wall in the black box (rain streak region) of the right figure is weaker than that in the white box (no rain region). It can be seen that the rain streak has an impact on the intensity of the image background, and the impact degree varies with the location. Therefore, it is unreasonable not to set the rain streak mask M_R in (II.2).

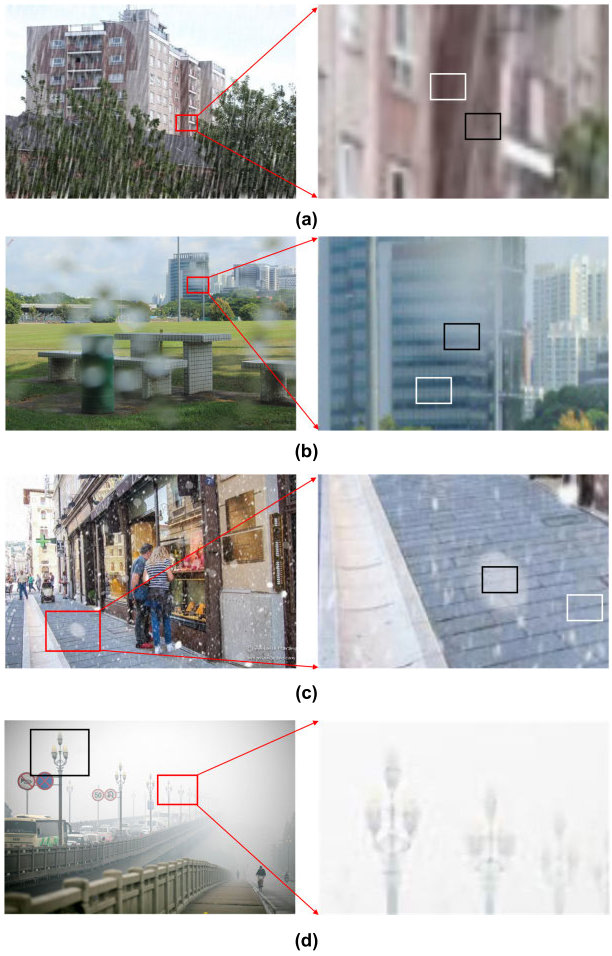


FIGURE 1. Influence of fog, rain streak, raindrop and snowflake regions on the image. (a) The details of the rain streak region, (b) The details of the raindrop region, (c) The details of the snowflake region, (d) The details of the fog region.

Second, when processing an image with raindrops, (II.4) indicates that, when the raindrop mask M_D is 0, the background is visible without attenuation. However, this expectation is not the case in the actual image, as shown in Fig. 1(b). The intensity of the blue windows in the black box (the raindrop region) is weaker than that of the blue windows in the white box (the nonraindrop region). Therefore, it can be seen that, in (II.4), it is unreasonable to set the raindrop mask M_D as a binary number.

Third, when processing an image with snowflakes, according to (II.3), when the snowflake mask M_S is between 0 and 1, there is translucency. The smaller the value is, the higher the background visibility is, as shown in Fig. 1(c). The gray floor tiles in the black box of the right figure are located in the snowflake region (the reflection of a snowflake), and the intensity value of the floor tiles is weaker than that of the snow-free region in the white box. Therefore, (II.3) is reasonable.

Finally, when processing a foggy image, according to (II.1), when the transmission map t is between 0 and 1,

there is translucency. The larger the value is, the higher the background visibility is, as shown in Fig. 1(d). The streetlight in the black box of the left figure is clearly visible, but the streetlights in the red box enlarged in the right figure are faintly visible. Because the former is nearer, t is larger; the latter is farther away, and t is smaller. Therefore, (II.1) is reasonable.

In summary, the intensity values of the background scenes in the fog, rain streak, raindrop and snowflake regions are all weakened; thus, the mask factors t, M_R, M_D and M_S must be considered when establishing the imaging model. With reference to (II.3), we change the rain streak imaging model from (II.2) to (III.1), and we change the raindrop imaging model from (II.4) to (III.2). The expressions are as follows:

$$I = (1 - M_R) \odot B + M_R \odot R \tag{III.1}$$

$$I = (1 - M_D) \odot B + M_D \odot D \tag{III.2}$$

The second problem is the lack of a comprehensive imaging model of severe weather conditions. We must build an imaging model to address the above four types of severe weather conditions together.

By combining (II.1), (II.3), (III.1) and (III.2), we can obtain a comprehensive imaging model of severe weather conditions as follows:

$$I = t \odot (1 - M_R) \odot (1 - M_S) \odot (1 - M_D) \odot B + A(1 - t) + M_R \odot R + M_S \odot S + M_D \odot D \tag{III.3}$$

If there is only one severe weather condition, (III.3) can be directly converted into (II.1), (II.3), (III.1) or (III.2). If there are multiple severe weather conditions, such as rain streaks and raindrops in an image, then the mask is M_R and M_D, and image $I = (1 - M_R) \odot (1 - M_D) \odot B + M_R \odot R + M_D \odot D$. Because of the influence of $(1\text{-}M_R)\odot(1\text{-}M_D)$, the intensity of the background scene located in both the raindrop and rain streak regions will be weaker than the single raindrop or rain streak region. If we place background B on the left side of the equal sign, (III.3) is transformed as follows:

$$B = \frac{I - A(1 - t) - M_R \odot R - M_S \odot S - M_D \odot D}{t \odot (1 - M_R) \odot (1 - M_S) \odot (1 - M_D)} \tag{III.4}$$

According to (III.4), the background scene B can be obtained to remove the severe weather features. Note that t cannot be 0 and that M_R, M_D and M_S cannot be 1. They represent that the background is completely occluded. At this time, B should be 0. Except for the image I, all of the variables on the right side of the equation are unknown, which is a typical ill-posed problem. If we estimate the atmospheric light value A, rain streak R, snowflake S, raindrop D and each mask and then combine them into a clean background scene B, there will be a large cumulative error, and the generated B will be distorted. Consequently, we take the right side of (III.4) as a whole and design a deep neural network. By training the network, the loss value tends to the minimum (i.e., the image with removal of all severe weather features is closer to the ground truth image), and a clean background scene image B can be obtained.

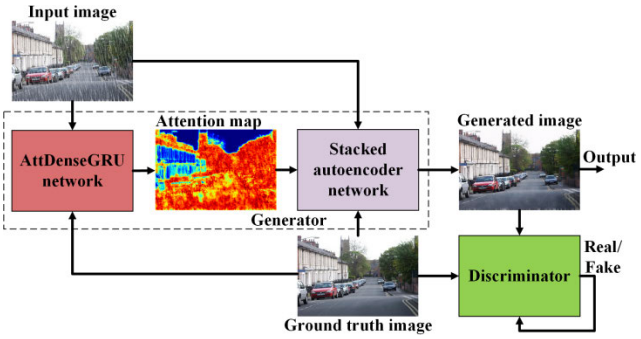


FIGURE 2. Overall structure of RASWNet.

IV. RASWNET

To obtain a clear image in any severe weather condition, we propose an algorithm called RASWNet that can remove all severe weather features from a degraded image. It is based on the GAN, and it uses the technology of the visual attention mechanism, the RNN and the CNN. The overall structure of RASWNet is shown in Fig. 2. It can be seen that RASWNet consists of a generator and a discriminator. The generator consists of an AttDenseGRU network and a Stacked autoencoder network. The input severe weather image is sent to the AttDenseGRU, which uses dense blocks and gated recurrent units (GRUs) to generate attention maps, and it outputs the last attention map to the Stacked autoencoder. The Stacked autoencoder sends the generated image to the discriminator, and the discriminator can distinguish whether the image is real or fake. The generated image also participates in the calculation of the loss function. As a label, the ground truth image is sent to the AttDenseGRU, the Stacked autoencoder and the discriminator. In addition, it is used to calculate the loss function.

A. ATTDENSEGRU NETWORK

The AttDenseGRU network is a structure that combines the CNN and the RNN. Its purpose is to locate the regions of severe weather features from the input image (including fog, raindrop, rain streak or snowflake) that need to be removed and the pixels around them. It can generate the attention maps that highlight the regions that must be removed in the image. On the one hand, the attention map is the reference of the Stacked autoencoder to remove fog, rain streaks, raindrops and snowflakes. On the other hand, it is one of the parameters of the loss function of the Stacked autoencoder and the discriminator. The overall structure of the AttDenseGRU is shown in Fig. 3.

It can be seen that an AttDenseGRU consists of three network parts with the same structure, and each of them is composed of five dense blocks, one GRU and one Conv layer. The input image is sent to the first network part to generate attention map 1. Subsequently, attention map 1 and the input image are concatenated into the second network part to generate attention map 2. Finally, attention map 2 and the input image are concatenated into the third network part to

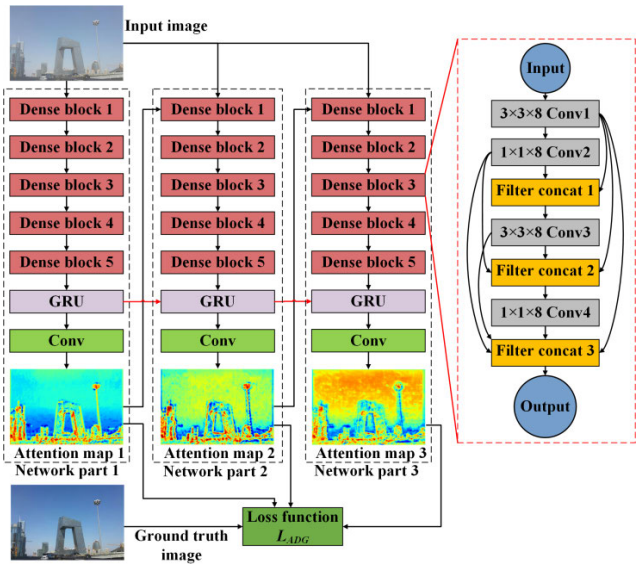


FIGURE 3. Overall structure of the AttDenseGRU network. Conv is a convolution layer. Filter concat is a filter concatenation layer. The parameters of the Conv layer are (Width of filters)×(Height of filters)×(The number of filters). The structure of the AttDenseGRU is on the left, and the internal structure of a dense block is on the right.

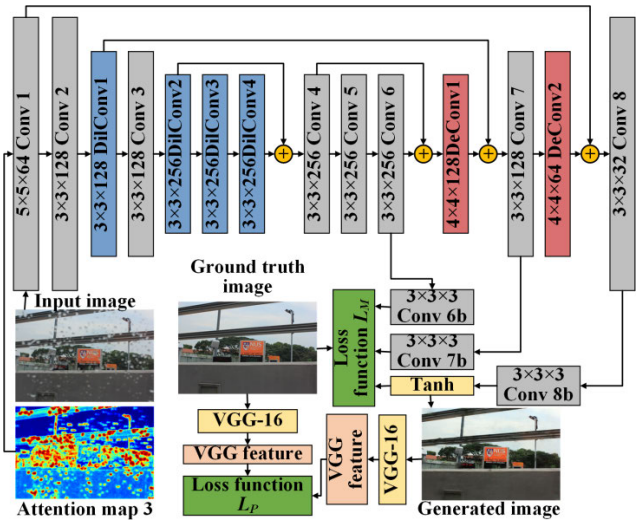


FIGURE 4. Structure of the Stacked autoencoder network. Conv is a convolution layer. DilConv is a dilated convolution layer. DeConv is a deconvolution layer. The parameters of the Conv, DilConv and DeConv layers are (Width of filters)×(Height of filters)×(The number of filters).

generate attention map 3. In the network, three GRUs are also connected in sequence, thus forming an RNN structure. The most commonly used RNN unit is the long short-term memory (LSTM) unit [44], but its structure is complex. Thus, the GRU, which is simpler than the LSTM, is adopted as a unit of the RNN. The memory ability of the GRU will gradually improve the attention level through each network part. The regions of fog, raindrops, rain streaks and snowflakes to be removed become more and more highlighted in the attention map. The changes of the highlighted fog regions of attention maps 1 to 3 can be seen in Fig. 3. The function of a Conv layer is to generate an attention map. Before training the network, the initial value of the attention map is set to 0.5.

The internal function of the convolution GRU [45] is realized by (IV.1). One GRU consists of an update gate z_t, a reset gate r_t, a new hidden state $\tilde{H}_t$ and a hidden state H_t. In this instance, $*$ is the convolution operation, σ is the Sigmoid function, tanh is the Hyperbolic tangent function, and b is the bias value. The first expression is the update gate z_t; it executes convolutions of the input X_t and the previous hidden state H_{t-1} in sequence, and then it performs nonlinear processing with a Sigmoid function, where $z_t \in (0,1)$. It can be seen from the fourth expression that, the smaller the value of z_t is, the smaller the proportion of H_{t-1} is, and the larger the proportion of $\tilde{H}_t$ is. The second expression is the reset gate r_t; its calculation is similar to the update gate, where $r_t \in (0,1)$. It can be seen from the third expression that, in the calculation of $\tilde{H}_t$, r_t determines the proportion of the previous hidden state H_{t-1}. The smaller the value of r_t is, the smaller the proportion of H_{t-1} is.

$$z_t = \sigma(W_z * X_t + U_z * H_{t-1} + b_z)$$
$$r_t = \sigma(W_r * X_t + U_r * H_{t-1} + b_r)$$
$$\tilde{H}_t = \tanh(W_h * X_t + U_h * (r_t \odot H_{t-1}) + b_H)$$
$$H_t = (1 - z_t) \odot \tilde{H}_t + z_t \odot H_{t-1} \quad \text{(IV.1)}$$

The right side of Fig. 3 shows the internal structure of a dense block. Each block consists of two $3 \times 3 \times 8$ Conv layers, two $1 \times 1 \times 8$ Conv layers and three Filter concat layers. The dense block can not only reduce the number of network parameters and calculations but also ensure the ability of feature extraction.

The loss function $L_{ADG}(A, C)$ of the AttDenseGRU is shown in Fig. 3 and (IV.2). In this instance, A is the attention map, C is the ground truth image, $n = 3$ is the number of attention maps, $\phi = 0.9$ is the base number of the coefficient, and A_t (t = 1,2,3) denotes the attention maps 1 through 3. It can be seen that the loss function $L_{ADG}(A, C)$ calculates the mean square error (MSE) between each attention map and the ground truth image, multiplies them by 0.9^{3-t} and sums the three products. When t changes from 1 to 3, the 0.9^{3-t} coefficient changes from 0.9^2, 0.9^1 to 1, which is the weight of each attention map, which indicates that the preceding attention map has less influence on the loss function and that the subsequent attention map has more influence.

$$L_{ADG}(A, C) = \sum_{t=1}^{n} \varphi^{n-t} L_{MSE}(A_t, C) \quad \text{(IV.2)}$$

B. STACKED AUTOENCODER NETWORK

The input of the Stacked autoencoder network concatenates the input image and attention map 3, and the output is the generated image, as shown in Fig. 4. A Stacked autoencoder consists of eight Conv layers, four DilConv (dilated convolution) layers and two DeConv (deconvolution) layers. The initial size of the filters of DilConv layers is 3×3, and the dilation rate [46] is 2, 4, 8 and 16 in successive order.

The DilConv layers can make the receptive field expand exponentially and do not affect the resolution of the image. According to [47], [48], the two DeConv layers can double the width and height of the input feature maps. To make the output feature maps smoother, the average pooling with unchanged size is used after each DeConv layer. To ensure that the output image is not distorted, four shortcuts are used in the network. In other words, the outputs of Conv 1 and DeConv 2 are added as the input of Conv 8; the outputs of DilConv 1 and DeConv 1 are added as the input of Conv 7; the outputs of DilConv 2 and DilConv 4 are added as the input of Conv 4; and the outputs of Conv 4 and Conv 6 are added as the input of DeConv 1.

The Stacked autoencoder has two loss functions, which are the multiscale loss function L_M and perceptual loss function L_P, as shown in Fig. 4. In this figure, L_M compares the output of $3 \times 3 \times 3$ Conv layers 6b, 7b and 8b with the ground truth image. Because the sizes of the three Conv layers are different (they are 25%, 50% and 100% of Conv 8, respectively), the loss function is called the multiscale loss function. The expression is as follows:

$$L_M(\{Y\}, C) = 0.8L_{MSE}(Y_6, \frac{C}{4}) + 0.9L_{MSE}(Y_7, \frac{C}{2}) + 1.0L_{MSE}(Y_8, C) \quad \text{(IV.3)}$$

where Y_6 and Y_7 represent the outputs of Conv 6b and 7b, respectively, and Y_8 is the output of Conv 8b through the function tanh. Here, C is the ground truth image, and $\frac{C}{4}$ and $\frac{C}{2}$ represent that their sizes are $\frac{1}{4}$ and $\frac{1}{2}$ of C, which is to remain consistent with the sizes of Y_6 and Y_7. L_{MSE} indicates the mean square error. Here, 0.8, 0.9 and 1.0 are three weight values, which indicate that the subsequent network layer has more influence on the loss function.

The perceptual loss function L_P, proposed by Johnson *et al.* [49], can compare the difference in the feature maps between the generated image $G(\mathbf{I})$ and the ground truth image C. Its expression is as follows [49]:

$$L_P(G(I), C) = L_{MSE}(VGG(G(I)), VGG(C)) \quad \text{(IV.4)}$$

The two images $G(\mathbf{I})$ and C are sent to the VGGNet-16 [50] pretrained model for forward propagation, and then, the first seven Conv feature maps are extracted. Finally, the MSE of these feature maps of $G(\mathbf{I})$ and C is calculated.

Combining (IV.2) through (IV.4), the expression of the generator total loss function L_G is obtained as follows [5]:

$$L_G = 0.01 \times \log(1 - D(G(I)) + L_{ADG}(A, C) + L_M(\{Y\}, C) + L_P(G(I), C) \quad \text{(IV.5)}$$

The first item on the right side of the equation is the original loss function of the GAN generator [51], which is multiplied by 0.01 to reduce its weight and enhance the role of the next three loss functions.

C. DISCRIMINATOR

The input of the discriminator is the generated image $O(O = G(\mathbf{I}))$ or the ground truth image C. It consists of six Conv layers, one discriminator map, three Filter concat layers, one global average pooling layer and one FC+Sigmoid layer. The network structure of the discriminator is shown in Fig. 5, in which the Conv layers and the Filter concat layers in the dotted box form a dense block to extract the features of the input image. The output of the dense block constitutes a discriminator map. Then, the output of the discriminator map and the Filter concat layer 3 are multiplied to highlight the regions of the severe weather features in the feature maps. Subsequently, the sizes of the feature maps are reduced by Conv layers 5 and 6, and then the network is flattened by the global average pooling layer. Finally, regardless of whether the image is real or fake is discriminated by going through the FC+Sigmoid layer.

It can be seen from Fig. 5 that the loss function of the discriminator consists of the output of the ground truth image or the generated image through the discriminator, the discriminator map and attention map 3. Its expression L_D is as follows:

$$L_D = -\log(D(C)) - \log(1 - D(O)) + L_{Dmap}(O, C, A_3) \quad \text{(IV.6)}$$

The first two items on the right side of the equation are the original loss function of the GAN discriminator [51]. The third item is called the discriminator map loss function $L_{Dmap}(O, C, A_3)$, which is related to the generated image O, the ground truth image C and attention map 3. The expression of this loss function is as follows:

$$L_{Dmap}(O, C, A_3) = L_{MSE}(D_{map}(C), D_{map}(O)) + L_{MSE}(D_{map}(O), A_3) \quad \text{(IV.7)}$$

where $D_{map}(O)$ or $D_{map}(C)$ represents the discriminator map produced by the generated image O or the ground truth image C going through the dense block. The first item on the right side of the equation represents the MSE of $D_{map}(C)$ and $D_{map}(O)$, and the second item represents the MSE of $D_{map}(O)$ and attention map 3. The smaller the MSE values of these two items are, the smaller the difference between the generated image, the ground truth image and attention map 3 is.

V. DATASET

The severe weather images mainly include degraded images caused by fog, rain streaks, raindrops or snowflakes, and thus, we must collect the corresponding synthetic image dataset. After collection and arrangement, the foggy images come from the realistic single image dehazing (RESIDE) benchmark dataset established by Li *et al.* [52]. The rain streak images come from the dataset of Fu *et al.* [18]; the raindrop images come from the dataset of Qian *et al.* [5]; and the snowflake images come from the Snow100K dataset of Liu *et al.* [14].

We select road traffic scene images from the datasets for the training, validation and testing of the network. Our severe weather image dataset is shown in Table 1. Note that a pair of images represents one severe weather image

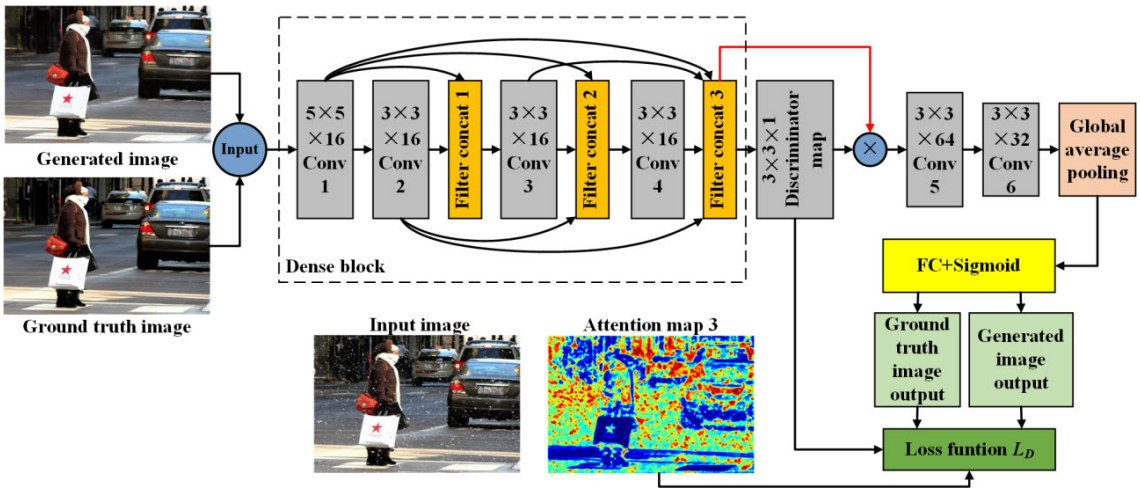


FIGURE 5. Network structure of the discriminator. Conv is a convolution layer. Filter concat is a filter concatenation layer. FC is a fully connected layer. The parameters of the Conv layer are (Width of filters) × (Height of filters) × (The number of filters).

TABLE 1. Our severe weather image dataset. Here, "-" indicates that the number does not need to be counted.

Weather condition	Name of the synthetic image dataset	Total number of images in the dataset	Number of severe weather images we selected	Number of ground truth images we selected
Foggy	RESIDE Outdoor Training Set [52]	72135	700	700
	RESIDE Indoor Training Set [52]	13990	300	300
Rainy	Synthetic rain streak image dataset [18]	15000	1000	1000
	Raindrop image dataset [5]	2336	1000	1000
Snowy	Snow100K testing dataset [14]	150000	1000	1000
Total	Our severe weather image dataset	-	4000	4000

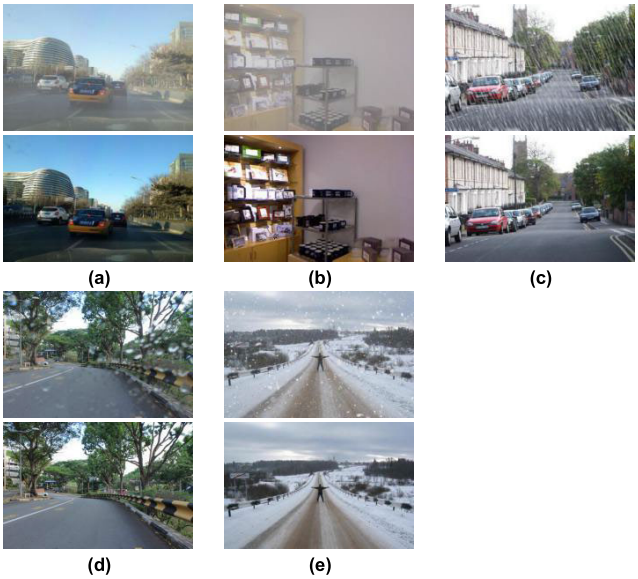


FIGURE 6. Five pairs of samples in our severe weather image dataset. The top is the severe weather image, and the bottom is the ground truth image. (a) Outdoor foggy image, (b) Indoor foggy image, (c) Rain streak image, (d) Raindrop image, (e) Snowflake image.

and one corresponding ground truth image. First, we select 700 pairs of outdoor foggy images from the RESIDE outdoor training set and 300 pairs of indoor foggy images from the RESIDE indoor training set. Second, we select 1000 pairs of rain streak images from the synthetic rain streak image dataset and 1000 pairs of raindrop images from the raindrop image dataset. Third, we select 422, 289 and 289 (1000 in total) pairs of snowflake images from the large, medium and small snowflakes of the Snow100k testing dataset. Finally, we establish our dataset by these 4000 pairs of images.

To speed up the training and validation of the network, it is necessary to convert the images into TFRecord (including images and labels), which is the binary data format of TensorFlow. The image sequence is shuffled by the program. Subsequently, 3400 pairs of images are randomly selected to train, 200 pairs to validate and 400 pairs to test. Among them, the original images are all adjusted to the JPG format with 720 × 480 size. When the TFRecord files are generated, the images are resized to 448 × 308. During training and validation, the images are automatically cropped to 420 × 280 at random. Finally, the image size can be set independently during testing, which is 420 × 316 by default. Five pairs of samples in our severe weather image dataset are shown in Fig. 6.

VI. EXPERIMENTS

After the severe weather image dataset has been collected, we must perform experimental research on RASWNet.

The experiments consist of six parts. Section A is the study of the Stacked autoencoder to optimize its network structure. Section B is the ablation study of RASWNet to verify whether our proposed network model is effective. Section C shows how RASWNet compares with the other algorithms regarding the clarity enhancement results of synthetic images. Section D shows how RASWNet compares with other algorithms regarding the clarity enhancement results of real images. Section E shows the running time of RASWNet compared with other algorithms. Section F shows the improvement in object detection results after using image clarity enhancement algorithms.

The settings of the training and validation hyperparameters are as follows. The initial learning rate is set to 0.0002. Because the network model is divided into two parts, the discriminator and the generator, the optimizer is different. The Adam is used in discriminator optimization, and the SGD with a momentum of 0.9 is used in generator optimization. To achieve the best training effect, the number of iteration steps is set to 200,000. The batch size is set to 1, which is actually a pair of images. The GPU memory fraction is 81% during training and 75% during validation because Windows 10 takes up more GPU memory. The PSNR and SSIM are used to evaluate the image quality during the training and validation.

A. STUDY OF THE STACKED AUTOENCODER

Before the study, we set the AttDenseGRU as structure A, the Stacked autoencoder as structure B and the discriminator as structure C. Among them, structure B is necessary to make the generated image. Therefore, to eliminate the interference, only structure B is used in this experiment. The study of the Stacked autoencoder mainly includes three aspects: the first is to change the number of DilConv layers, the second is to change the type of activation functions, and the third is to change the number of feature maps. We choose 5 images (20 images in total) from the test set of fog, rain streaks, raindrops and snowflakes, respectively. Through these three aspects of study, we verify their impact on the clarity enhancement results and obtain the best network model of structure B. Their experimental contents are described as follows:

The first study is to change the number of DilConv layers in structure B, from 0, 2, 4 to 6. Here, 0 represents that all DilConv layers in Fig. 4 are replaced by Conv layers; 2 represents that DilConv layers 3 and 4 in Fig. 4 are replaced by two Conv layers; 4 is consistent with the network in Fig. 4; 6 represents that Conv layers 4 and 5 in Fig. 4 are replaced by two DilConv layers, and the dilation rate is 2, 2, 4, 8, 16 and 32 in successive order. The average PSNR and SSIM values of the output images are shown in Table 2. It can be seen that the PSNR and SSIM values of the four DilConv layers are the highest, and the clarity enhancement results are not good when there is no DilConv layer or too many DilConv layers. Consequently, it is reasonable to use four DilConv layers in structure B.

TABLE 2. Average PSNR and SSIM values of the output images after we change the number of DilConv layers in structure B. The red numbers indicate the best result.

Number of DilConv layers	0	2	4	6
Average PSNR	18.913	19.365	**25.326**	23.542
Average SSIM	0.804	0.814	**0.870**	0.794

TABLE 3. Average PSNR and SSIM values of the output images after we change the type of activation functions in structure B. The red numbers indicate the best result.

Activation function	ReLU	LeakyReLU	Tanh
Average PSNR	**25.326**	24.482	21.848
Average SSIM	0.870	**0.884**	0.863

TABLE 4. Average PSNR and SSIM values of the output images after we change the number of feature maps in structure B. The red numbers indicate the best result.

Number of feature maps	1/8	1/4	1/2	original
Average PSNR	22.343	23.697	22.474	**25.326**
Average SSIM	0.847	0.863	0.863	**0.870**

The second study is to change the type of activation functions in structure B. In addition to the Conv 8b layer, which keeps the Tanh unchanged, the activation functions in the network are compared by ReLU, LeakyReLU and Tanh. The slope of the negative part of LeakyReLU is 0.2. The average PSNR and SSIM values of the output images are shown in Table 3. It can be seen that the result of using LeakyReLU is basically the same as that of using ReLU. The SSIM value of the former is slightly higher, and the PSNR value of the latter is slightly higher, while the latter is better because the operation of the former is more complex. Tanh is the worst among the three. Consequently, it is reasonable to choose the ReLU function in structure B.

The third study is to change the number of feature maps in structure B, from 1/8 (one eighth of the original), 1/4 (a quarter of the original), 1/2 (a half of the original) to the original. The average PSNR and SSIM values of the output images are shown in Table 4. It can be seen that the result of the original number of feature maps is the best, the 1/4 number of feature maps is second, and the other two are worse. Consequently, it is reasonable to use the original number of feature maps in structure B.

B. ABLATION STUDY OF RASWNET

To verify the effectiveness of our proposed RASWNet model, it is necessary to conduct an ablation study. This study mainly includes two aspects: the first is to change the number of network parts that contain GRU in structure A to verify the impact of the attention map on the clarity enhancement result; the second is to change the different combinations of structures A and B and C to verify the impact of each part on the clarity enhancement result. Their experimental contents are described as follows:

FIGURE 7. Result of clarity enhancement after the number of network parts is changed. The four images are fog, rain streak, raindrop and snowflake, from top to bottom. These images are not used for training. The red numbers indicate the best result. AVE means average. (a) Input, (b) n=2, (c) n=3, (d) n=4, (e) n=5.

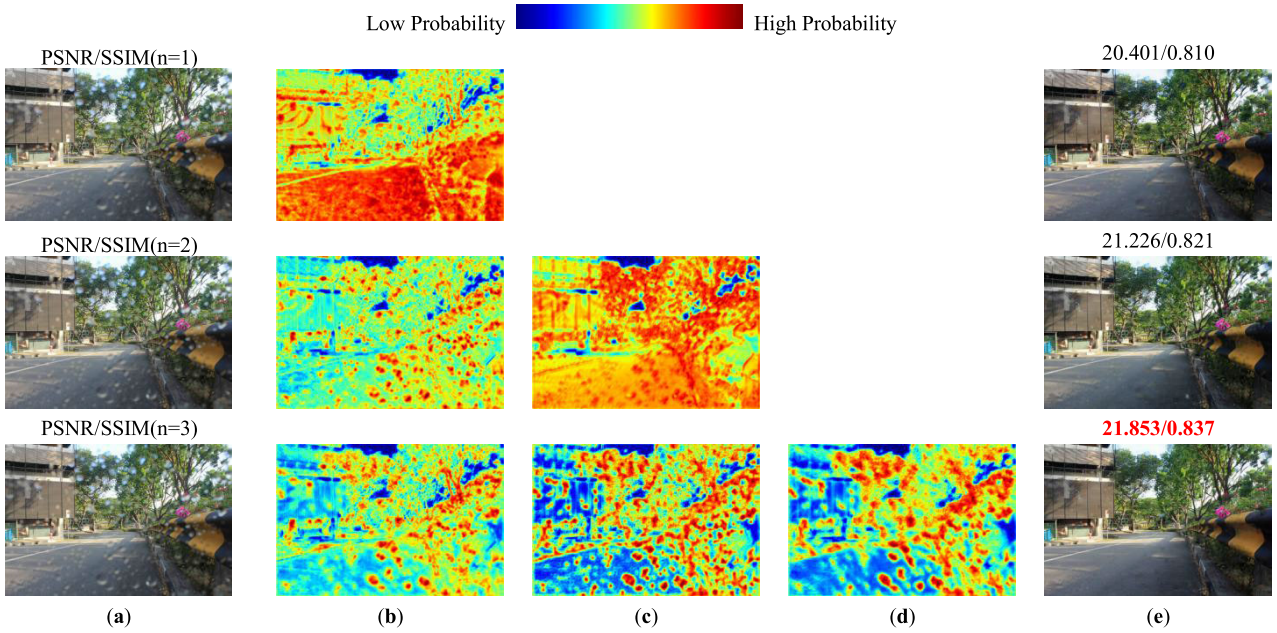


FIGURE 8. Effect of attention maps on the result of clarity enhancement. This is an image of raindrops, and it is not used for training. The red numbers indicate the best result. (a) Input, (b) Attention map 1, (c) Attention map 2, (d) Attention map 3, (e) Output.

The first ablation study is to change the number of network parts in structure A. The structure of each network part is shown in Fig. 3. Each network part generates an attention map. We use four models to train, validate and test, and the number of network parts of each model ranges from 2 to 5 (n=2, 3, 4, 5). The generated images and the PSNR and SSIM values of each model are shown in Fig. 7. It can be seen that the clarity enhancement result is the best when n = 3.

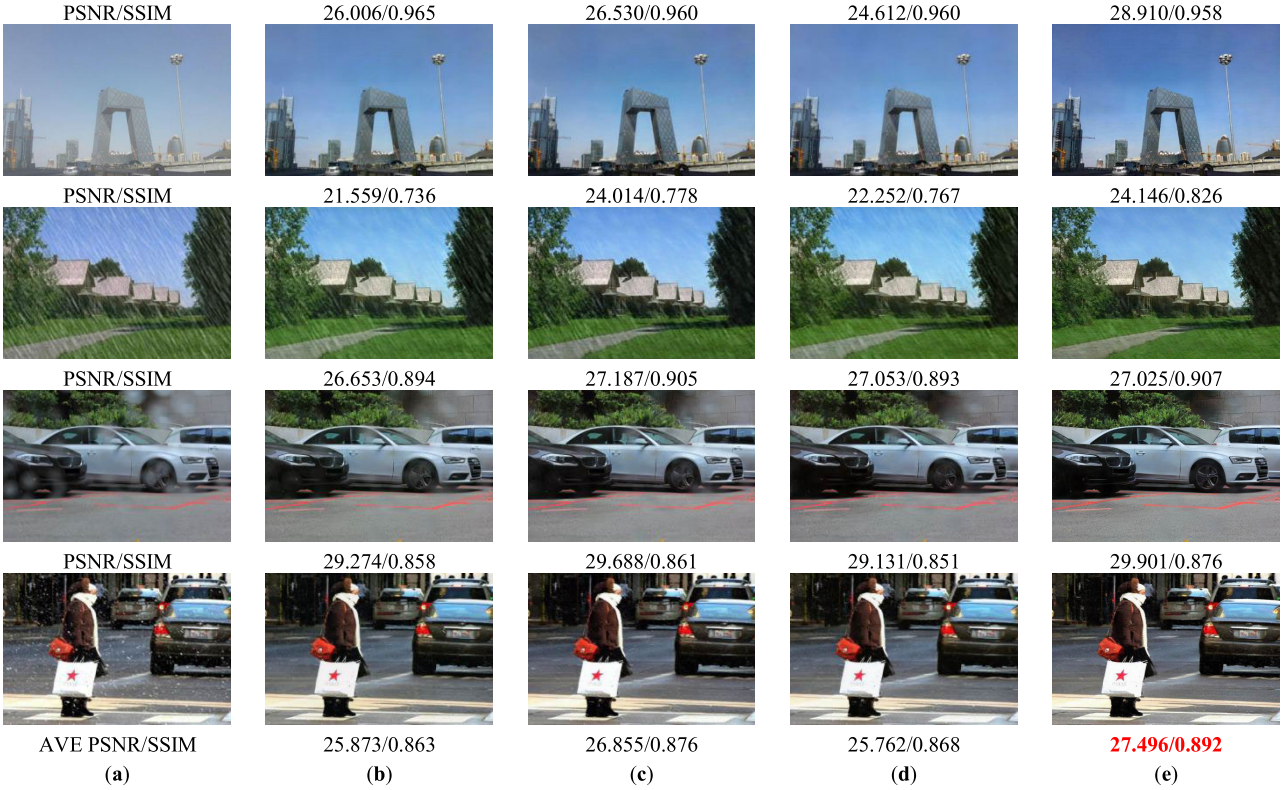


FIGURE 9. Result of clarity enhancement after the combination of the three structures is changed. The four images are the same as in Figure 7. The red numbers indicate the best result. AVE means average. (a) Input, (b) B, (c) A+B, (d) B+C, (e) A+B+C.

FIGURE 10. First comparison of the defogging results of the synthetic images. We selected three foggy images of road traffic scenes from the RESIDE Outdoor Training Set [52]. They are not used for training. The red numbers indicate the best result. AVE means the average. (a) Input, (b) DCP [8], (c) BCCR [9], (d) AOD-Net [23], (e) RASWNet.

On this basis, we continue to study the effect of attention maps on the results of clarity enhancement. We also trained a model of only one network part (n=1) in structure A. Subsequently, we compare the three network models of n = 1 through 3. When n = 1 to 3, structure A outputs attention maps 1 through 3 to structure B, respectively.

The attention maps, the generated images and the PSNR and SSIM values of each model are shown in Fig. 8. It can be seen that the result of clarity enhancement is the best when n = 3 and the worst when n = 1. It can be seen from the attention maps that, when n = 1, the attention of the model is mainly on the road surface, resulting in incomplete removal of raindrops in the position of the road surface. When n = 2, the attention of the model is mainly on the trees, and the removal of raindrops in the position of the road surface is better than the model of n = 1. When n = 3, the attention of the model is basically on the raindrops, so the result is the best.

The second ablation study is to change the combination of structures A and B and C. Therefore, there are four combinations of network models: B, A+B, B+C and A+B+C. We use four models to train, validate and test. The generated images and the PSNR and SSIM values of each model are shown in Fig. 9. It can be seen that the clarity enhancement result of the B+C structure is slightly better than that of B, the A+B structure is better than the former two, and the result of the A+B+C structure is the best. Consequently, our proposed network model is effective.

C. SYNTHETIC IMAGES

Because the ground truth images are available for reference, the clarity enhancement results of the synthetic severe weather images can be verified from objective aspects. Our proposed RASWNet will be compared with the image defogging algorithms DCP [8], BCCR [9], DehazeNet [22] and AOD-Net [23], the rain streak removal algorithms GSM [17], DDN [18], and ID-CGAN [39], and the raindrop removal algorithm ATT-GAN [5].

The first comparison of the defogging results of the synthetic images and the corresponding PSNR and SSIM values are shown in Fig. 10. It can be seen that RASWNet has the best defogging quality, regardless of the detail or color. The defogging quality of AOD-Net is common; the removal of the fog is uneven, but the image color has no obvious distortion. The blue sky in the defogging images of BCCR is obviously distorted, the color is too saturated, and the white area is larger than normal. The defogging images of DCP are better than those of BCCR in the color of the blue sky, but the road color is darker. From the average values of the PSNR and SSIM, we can see that RASWNet is the best, BCCR is second in the PSNR, and DCP is second in SSIM.

To further test the effectiveness of the image defogging function of RASWNet, we also need to test it on the other dataset. The selected dataset is the O-HAZE dataset of the NTIRE 2018 Challenge on Image Dehazing [25]. The second comparison of the defogging results of the synthetic images and the corresponding PSNR and SSIM values are shown in Fig. 11. It can be seen that RASWNet is also the best on this dataset. The defogging results of DCP and BCCR are similar to that of Fig. 10. DehazeNet's result is not completely defogged, and the color is darker.

The comparison of the deraining and desnowing results of the synthetic images and the corresponding PSNR and SSIM values are shown in Fig. 12. RASWNet basically removes the rain streaks. The small and medium raindrops have been removed, but there are some black artifacts after removing large raindrops. It can remove most of the small and medium snowflakes, and only some of the larger snowflakes have residues (see Fig. 12(f)). ID-CGAN can remove most of the rain streaks, and it has the best result of the other three rain streak removal algorithms. It cannot remove raindrops and snowflakes, basically (see Fig. 12(e)). ATT-GAN can remove raindrops slightly better than RASWNet. However, it cannot remove rain streaks. The ability to remove snowflakes is worse than that of RASWNet (see Fig. 12(d)). DDN can remove most of the rain streaks, but the result is not as good as that of RASWNet and ID-CGAN. It cannot remove raindrops and snowflakes, basically (see Fig. 12(c)). GSM can only remove some of the finer rain streaks. It cannot remove raindrops and snowflakes (see Fig. 12(b)). From the average value of the PSNR and SSIM, it can be seen that RASWNet is the best, and ATT-GAN is second.

D. REAL IMAGES

The evaluation of the clarity enhancement results of the real images is different from that of synthetic images because there are no ground truth images. We use DCP [8], DDN [18] and ATT-GAN [5] to compare with RASWNet. Comparison of the clarity enhancement results of the real severe weather images is shown in Fig. 13. From the foggy images, RASWNet is the best because it has strong defogging ability and no color distortion; DCP is the second best except that the sky color is distorted and the brightness is darker; ATT-GAN can remove a thin layer of fog; and DDN is basically useless. From the rain streak images, DDN is the best, RASWNet is the second best, ATT-GAN is the third best, and DCP cannot remove the rain streaks. From the raindrop images, RASWNet is better than ATT-GAN, while DCP and DDN cannot remove the raindrops. From the snowflake images, RASWNet is the best, and ATT-GAN can remove the smaller snowflakes, while DCP and DDN are basically useless.

E. RUNNING TIME COMPARISON

The average running times of the clarity enhancement algorithms are shown in Table 5. We select 20 images (foggy, rain streaks, raindrops and snowflakes, 5 images each) from the test set for the algorithms to run on the same machine (Intel Core-i7 7700K CPU, 16 GB of memory and a Nvidia GeForce GTX 1070 GPU). Experimental results show that the running speed of RASWNet is in the middle of the seven algorithms: slower than AOD-Net [23], GSM [17], and ATT-GAN [5] but faster than DCP [8], DDN [18] and BCCR [9]. Of course, the result is obtained by GPU acceleration.

F. EVALUATION ON OBJECT DETECTION

Image defogging, deraining and desnowing algorithms can be used as a preprocessing step to improve the performance

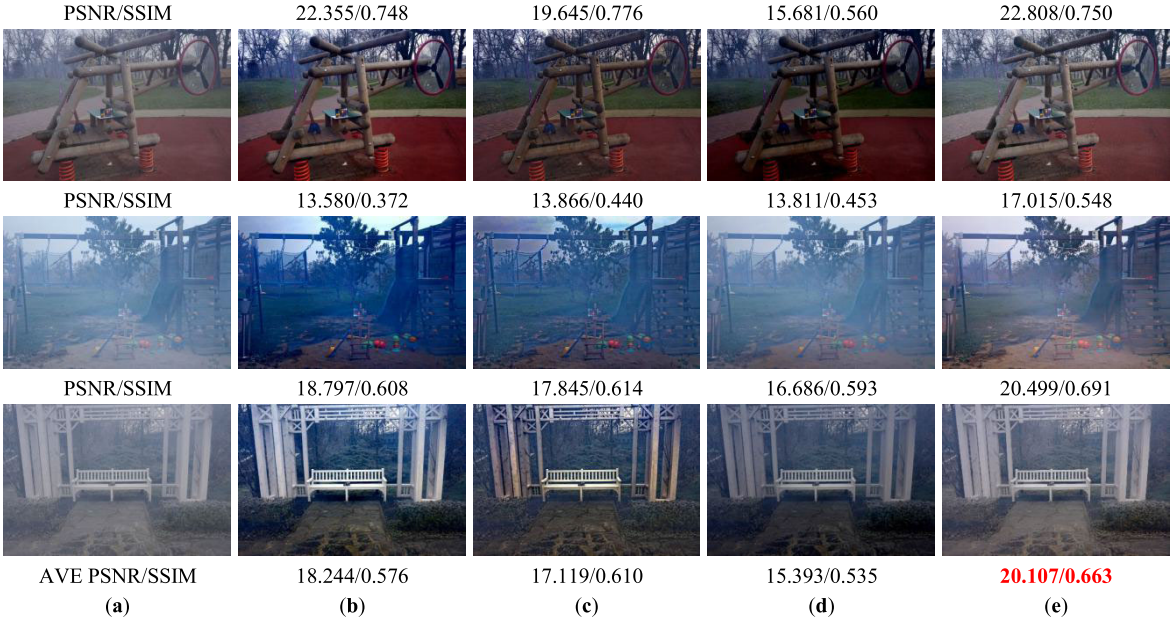


FIGURE 11. Second comparison of the defogging results of the synthetic images. We selected three foggy images from the O-HAZE Dataset of the NTIRE 2018 Challenge on Image Dehazing [25]. The red numbers indicate the best result. AVE means the average. (a) Input, (b) DCP [8], (c) BCCR [9], (d) DehazeNet [22], (e) RASWNet.

FIGURE 12. Comparison of the deraining and desnowing results of the synthetic images. We selected three rainy and snowy images of road traffic scenes. From the top to the bottom, the first image is from the synthetic rain streak image dataset [18], the second image is from the raindrop image dataset [5], and the third image is from the Snow100K testing dataset [14]. They are not used for training. The red numbers indicate the best result. AVE means the average. (a) Input, (b) GSM [17], (c) DDN [18], (d) ATT-GAN [5], (e) ID-CGAN [39], (f) RASWNet.

of other high-level vision tasks, such as face recognition and object detection [5], [6], [14], [23], [26]. However, the above algorithms can only handle one or two severe weather conditions and cannot be used in all weather conditions. Consequently, to demonstrate the performance improvement obtained after clarity enhancement using RASWNet, we evaluated Faster-RCNN [54] on the VOC 2012 dataset.

First, we selected 102 images of road traffic scenes from the VOC 2012 test set as ground truth images. Using the Weather function of CorelDRAW, these images were made into an equal number of foggy, rain streak and snowflake images. Subsequently, we used RASWNet to process the degraded images and used the pretrained Faster-RCNN model to detect the objects. In this process, the defogging algorithm DCP [8] and the deraining algorithm DDN [18] were also used to compare with the RASWNet. The mean average precision (mAP) and F1-measure values of the object detection results are shown in Table 6. It can be seen that

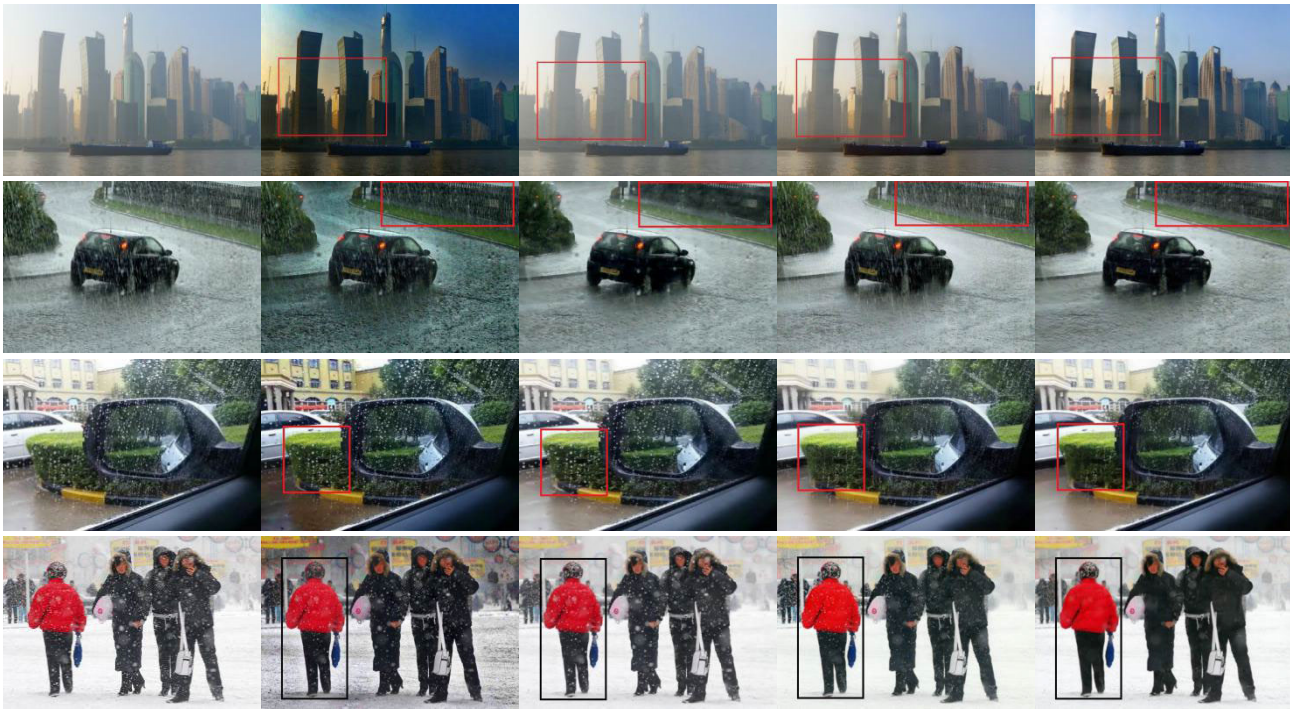


FIGURE 13. Comparison of the clarity enhancement results of the real severe weather images. We selected four images from the multiclass weather image (MWI) dataset [53] collected by Zhang *et al.* From top to bottom, they are fog, rain streak, raindrop and snowflake images. (a) Input, (b) DCP [8], (c) DDN [18], (d) ATT-GAN [5], (e) RASWNet.

FIGURE 14. Samples of object detection (Faster-RCNN [54]) results after using clarity enhancement algorithms. From top to bottom, they are fog, rain streak and snowflake images. (a) Degraded images, (b) Processed by DDN [18], (c) Processed by DCP [8], (d) Processed by RASWNet, (e) Ground truth images.

Faster-RCNN can only achieve a very low average precision for degraded images. DDN slightly improved the average precision of rainy images, DCP improved the average precision of foggy images, and RASWNet improved the average precision of all images by approximately 47%.

The samples of object detection results after using clarity enhancement algorithms are shown in Fig. 14. It can be seen that Faster-RCNN has poor detection ability in degraded images and can only detect one person from the snowy image. After the deraining process by DDN [18], the same

TABLE 5. Comparison of average running times of the algorithms (unit: second, image size: 480 × 360). The red numbers indicate the best result.

Algorithms	Platform	Running time
DCP [8]	MATLAB	2.09
BCCR [9]	MATLAB	4.37
AOD-Net [23] (GPU)	PyCharm	**0.09**
GSM [17]	MATLAB	1.17
DDN [18]	MATLAB	4.02
ATT-GAN [5] (GPU)	PyCharm	1.28
RASWNet (GPU)	PyCharm	1.35

TABLE 6. Object detection performance using Faster-RCNN [54] on the VOC 2012 dataset. The red numbers indicate the best result.

Conditions	mAP	F1-measure
Degraded images	0.288	0.344
Processed by the DDN [18]	0.314	0.378
Processed by the DCP [8]	0.400	0.437
Processed by the RASWNet	**0.424**	**0.465**
Ground truth images	0.559	0.555

detection model can detect two targets from the rainy image and one more person from both foggy and snowy images. After the defogging process by DCP [8], the same detection model detects the same number of persons from the foggy image as the ground truth image and detects one more person from the rainy image. After the clarity enhancement process by RASWNet, the detection results of all images can be improved. The detection result of foggy and snowy images is basically the same as that of the ground truth images, and the detection result of the rainy image is better than that of DDN.

VII. CONCLUSION

In this paper, we build a comprehensive severe weather imaging model that can represent the features of fog, rain streaks, raindrops and snowflakes in an image. Subsequently, we propose an algorithm called RASWNet that can remove all of the severe weather features from a degraded image. Based on the GAN, it uses the visual attention mechanism to locate the regions of fog, rain streaks, raindrops and snowflakes; it uses the GRUs to memorize these regions; and it uses dense blocks to extract features from the image. We verify the effectiveness of each structure in the algorithm model by performing a study of the Stacked autoencoder and an ablation study of RASWNet. We also use various synthetic images and real images to test our algorithm, and we compare it with some commonly used defogging, desnowing and deraining algorithms. The experimental results show that RASWNet is not only better than the commonly used algorithms in its clarity enhancement capacity but also useful in any severe weather condition, and it is suitable for ADAS and monitoring systems. However, RASWNet runs at a slower speed, and in the future, we will increase the speed while maintaining the clarity enhancement ability of the algorithm.

REFERENCES

[1] A. Dairi, F. Harrou, M. Senouci, and Y. Sun, "Unsupervised obstacle detection in driving environments using deep-learning-based stereovision," *Robot. Auto. Syst.*, vol. 100, pp. 287–301, Feb. 2018, doi: 10.1016/j.robot.2017.11.014.
[2] H. Liu, T. Taniguchi, Y. Tanaka, K. Takenaka, and T. Bando, "Visualization of driving behavior based on hidden feature extraction by using deep learning," *IEEE Trans. Intell. Transp. Syst.*, vol. 18, no. 9, pp. 2477–2489, Sep. 2017, doi: 10.1109/TITS.2017.2649541.
[3] H. Yoo, J. Son, B. Ham, and K. Sohn, "Real-time rear obstacle detection using reliable disparity for driver assistance," *Expert Syst. Appl.*, vol. 56, pp. 186–196, Sep. 2016, doi: 10.1016/j.eswa.2016.02.049.
[4] A. Asvadi, C. Premebida, P. Peixoto, and U. Nunes, "3D lidar-based static and moving obstacle detection in driving environments: An approach based on voxels and multi-region ground planes," *Robot. Auto. Syst.*, vol. 83, pp. 299–311, Sep. 2016, doi: 10.1016/j.robot.2016.06.007.
[5] R. Qian, R. T. Tan, W. Yang, J. Su, and J. Liu, "Attentive generative adversarial network for raindrop removal from a single image," in *Proc. CVPR*, Salt Lake City, UT, USA, Jun. 2018, pp. 2482–2491, doi: 10.1109/CVPR.2018.00263.
[6] Q. Zhu, J. Mai, and L. Shao, "A fast single image haze removal algorithm using color attenuation prior," *IEEE Trans. Image Process.*, vol. 24, no. 11, pp. 3522–3533, Nov. 2015, doi: 10.1109/TIP.2015.2446191.
[7] C. Qu, D.-Y. Bi, P. Sui, A.-N. Chao, and Y.-F. Wang, "Robust dehaze algorithm for degraded image of CMOS image sensors," *Sensors*, vol. 17, no. 10, p. 2175, Sep. 2017, doi: 10.3390/s17102175.
[8] K. He, J. Sun, and X. Tang, "Single image haze removal using dark channel prior," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 12, pp. 2341–2353, Dec. 2011, doi: 10.1109/TPAMI.2010.168.
[9] G. Meng, Y. Wang, J. Duan, S. Xiang, and C. Pan, "Efficient image dehazing with boundary constraint and contextual regularization," in *Proc. ICCV*, Sydney, NSW, Australia, Dec. 2013, pp. 617–624, doi: 10.1109/ICCV.2013.82.
[10] D. Hao, Q. Li, and C. Li, "Single-image-based rain streak removal using multidimensional variational mode decomposition and bilateral filter," *J. Electron. Imag.*, vol. 26, no. 1, Feb. 2017, Art. no. 013020, doi: 10.1117/1.JEI.26.1.013020.
[11] L. Zhu, C.-W. Fu, D. Lischinski, and P.-A. Heng, "Joint bi-layer optimization for single-image rain streak removal," in *Proc. ICCV*, Venice, Italy, Oct. 2017, pp. 2545–2553, doi: 10.1109/ICCV.2017.276.
[12] Y. Li, R. T. Tan, X. Guo, J. Lu, and M. S. Brown, "Rain streak removal using layer priors," in *Proc. CVPR*, Seattle, WA, USA, Jun. 2016, pp. 2736–2744, doi: 10.1109/CVPR.2016.299.
[13] D. Eigen, D. Krishnan, and R. Fergus, "Restoring an image taken through a window covered with dirt or rain," in *Proc. ICCV*, Sydney, NSW, Australia, Dec. 2013, pp. 633–640, doi: 10.1109/ICCV.2013.84.
[14] Y.-F. Liu, D.-W. Jaw, S.-C. Huang, and J.-N. Hwang, "DesnowNet: Context-aware deep network for snow removal," *IEEE Trans. Image Process.*, vol. 27, no. 6, pp. 3064–3073, Jun. 2018, doi: 10.1109/TIP.2018.2806202.
[15] W. Ren, J. Tian, Z. Han, A. Chan, and Y. Tang, "Video desnowing and deraining based on matrix decomposition," in *Proc. CVPR*, Honolulu, HI, USA, Jul. 2017, pp. 2838–2847, doi: 10.1109/CVPR.2017.303.
[16] J.-H. Kim, J.-Y. Sim, and C.-S. Kim, "Video deraining and desnowing using temporal correlation and low-rank matrix completion," *IEEE Trans. Image Process.*, vol. 24, no. 9, pp. 2658–2670, Sep. 2015, doi: 10.1109/TIP.2015.2428933.
[17] L.-J. Deng, T.-Z. Huang, X.-L. Zhao, and T.-X. Jiang, "A directional global sparse model for single image rain removal," *Appl. Math. Model.*, vol. 59, pp. 662–679, Jul. 2018, doi: 10.1016/j.apm.2018.03.001.
[18] X. Fu, J. Huang, D. Zeng, Y. Huang, X. Ding, and J. Paisley, "Removing rain from single images via a deep detail network," in *Proc. CVPR*, Honolulu, HI, USA, Jul. 2017, pp. 1715–1723, doi: 10.1109/CVPR.2017.186.
[19] P. C. Barnum, S. Narasimhan, and T. Kanade, "Analysis of rain and snow in frequency space," *Int. J. Comput. Vis.*, vol. 86, nos. 2–3, pp. 256–274, Jan. 2010, doi: 10.1007/s11263-008-0200-2.
[20] L.-W. Kang, C.-W. Lin, and Y.-H. Fu, "Automatic single-image-based rain streaks removal via image decomposition," *IEEE Trans. Image Process.*, vol. 21, no. 4, pp. 1742–1755, Apr. 2012, doi: 10.1109/TIP.2011.2179057.
[21] D.-A. Huang, L.-W. Kang, Y.-C.-F. Wang, and C.-W. Lin, "Self-learning based image decomposition with applications to single image denoising," *IEEE Trans. Multimedia*, vol. 16, no. 1, pp. 83–93, Jan. 2014, doi: 10.1109/TMM.2013.2284759.

[22] B. Cai, X. Xu, K. Jia, C. Qing, and D. Tao, "DehazeNet: An end-to-end system for single image haze removal," *IEEE Trans. Image Process.*, vol. 25, no. 11, pp. 5187–5198, Nov. 2016, doi: 10.1109/TIP.2016.2598681.

[23] B. Li, X. Peng, Z. Wang, J. Xu, and D. Feng, "AOD-net: All-in-one dehazing network," in *Proc. ICCV*, Venice, Italy, Oct. 2017, pp. 4780–4788, doi: 10.1109/ICCV.2017.511.

[24] W. Ren, L. Ma, J. Zhang, J. Pan, X. Cao, W. Liu, and M.-H. Yang, "Gated fusion network for single image dehazing," in *Proc. CVPR*, Salt Lake City, UT, USA, Jun. 2018, pp. 3253–3261, doi: 10.1109/CVPR.2018.00343.

[25] C. Ancuti, C. O. Ancuti, and R. Timofte, "Ntire 2018 challenge on image dehazing: Methods and results," in *Proc. CVPR*, Salt Lake City, UT, USA, Jun. 2018, pp. 1004–1014.

[26] Y. Song, J. Li, X. Wang, and X. Chen, "Single image dehazing using ranking convolutional neural network," *IEEE Trans. Multimedia*, vol. 20, no. 6, pp. 1548–1560, Jun. 2018, doi: 10.1109/TMM.2017.2771472.

[27] C.-H. Yeh, C.-H. Huang, and L.-W. Kang, "Multi-scale deep residual learning-based single image haze removal via image decomposition," *IEEE Trans. Image Process.*, vol. 29, pp. 3153–3167, 2020, doi: 10.1109/TIP.2019.2957929.

[28] H. Zhang and V. M. Patel, "Densely connected pyramid dehazing network," in *Proc. CVPR*, Salt Lake City, UT, USA, Jun. 2018, pp. 3194–3203, doi: 10.1109/CVPR.2018.00337.

[29] A. Dudhane and S. Murala, "CDNet: Single image de-hazing using unpaired adversarial training," in *Proc. WACV*, Waikoloa Village, HI, USA, Jan. 2019, pp. 1147–1155, doi: 10.1109/WACV.2019.00127.

[30] Y. Qu, Y. Chen, J. Huang, and Y. Xie, "Enhanced Pix2pix dehazing network," in *Proc. CVPR*, Long Beach, CA, USA, Jun. 2019, pp. 8160–8168.

[31] X. Fu, J. Huang, X. Ding, Y. Liao, and J. Paisley, "Clearing the skies: A deep network architecture for single-image rain removal," *IEEE Trans. Image Process.*, vol. 26, no. 6, pp. 2944–2956, Jun. 2017, doi: 10.1109/TIP.2017.2691802.

[32] G. Li, X. He, W. Zhang, H. Chang, L. Dong, and L. Lin, "Non-locally enhanced encoder-decoder network for single image de-raining," in *Proc. ACM MM*, Seoul, South Korea, 2018, pp. 1056–1064, doi: 10.1145/3240508.3240636.

[33] W. Yang, R. T. Tan, J. Feng, J. Liu, S. Yan, and Z. Guo, "Joint rain detection and removal from a single image with contextualized deep networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, Jan. 28, 2019, doi: 10.1109/TPAMI.2019.2895793.

[34] L. Pei, X. Yuan, and X. Dai, "MWNet: Object detection network applicable for different weather conditions," *IET Intell. Transp. Syst.*, vol. 13, no. 9, pp. 1394–1400, Sep. 2019, doi: 10.1049/iet-its.2019.0023.

[35] D. Ren, W. Zuo, Q. Hu, P. Zhu, and D. Meng, "Progressive image deraining networks: A better and simpler baseline," in *Proc. CVPR*, Long Beach, CA, USA, Jun. 2019, pp. 3937–3946.

[36] T. Wang, X. Yang, K. Xu, S. Chen, Q. Zhang, and R. W. H. Lau, "Spatial attentive single-image deraining with a high quality real rain dataset," in *Proc. CVPR*, Long Beach, CA, USA, Jun. 2019, pp. 12270–12279.

[37] X. Fu, B. Liang, Y. Huang, X. Ding, and J. Paisley, "Lightweight pyramid networks for image deraining," 2018, *arXiv:1805.06173*. [Online]. Available: http://arxiv.org/abs/1805.06173

[38] K. Jiang, Z. Wang, P. Yi, C. Chen, B. Huang, Y. Luo, J. Ma, and J. Jiang, "Multi-scale progressive fusion network for single image deraining," 2020, *arXiv:2003.10985*. [Online]. Available: http://arxiv.org/abs/2003.10985

[39] H. Zhang, V. Sindagi, and V. M. Patel, "Image de-raining using a conditional generative adversarial network," 2017, *arXiv:1701.05957*. [Online]. Available: http://arxiv.org/abs/1701.05957

[40] R. Li, L.-F. Cheong, and R. T. Tan, "Heavy rain image restoration: Integrating physics model and conditional adversarial learning," in *Proc. CVPR*, Long Beach, CA, USA, Jun. 2019, pp. 1633–1642.

[41] Z. Li, J. Zhang, Z. Fang, B. Huang, X. Jiang, Y. Gao, and J.-N. Hwang, "Single image snow removal via composition generative adversarial networks," *IEEE Access*, vol. 7, pp. 25016–25025, 2019, doi: 10.1109/ACCESS.2019.2900323.

[42] T. Dong, G. Zhao, J. Wu, Y. Ye, and Y. Shen, "Efficient traffic video dehazing using adaptive dark channel prior and spatial–temporal correlations," *Sensors*, vol. 19, no. 7, p. 1593, Apr. 2019, doi: 10.3390/s19071593.

[43] S. Li, X. Cao, I. B. Araujo, W. Ren, Z. Wang, E. K. Tokuda, R. H. Junior, R. Cesar-Junior, J. Zhang, and X. Guo, "Single image deraining: A comprehensive benchmark analysis," in *Proc. CVPR*, Long Beach, CA, USA, Jun. 2019, pp. 3838–3847.

[44] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, Nov. 1997, doi: 10.1162/neco.1997.9.8.1735.

[45] K. Cho, B. van Merrienboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, "Learning phrase representations using RNN encoder-decoder for statistical machine translation," 2014, *arXiv:1406.1078*. [Online]. Available: http://arxiv.org/abs/1406.1078

[46] F. Yu and V. Koltun, "Multi-scale context aggregation by dilated convolutions," 2015, *arXiv:1511.07122*. [Online]. Available: http://arxiv.org/abs/1511.07122

[47] M. D. Zeiler and R. Fergus, "Visualizing and understanding convolutional networks," in *Proc. ECCV*, Zürich, Switzerland, 2014, pp. 818–833.

[48] M. D. Zeiler, G. W. Taylor, and R. Fergus, "Adaptive deconvolutional networks for mid and high level feature learning," in *Proc. ICCV*, Barcelona, Spain, Nov. 2011, pp. 2018–2025.

[49] J. Johnson, A. Alahi, and F. Li, "Perceptual Losses for real-time style transfer and super-resolution," in *Proc. ECCV*, Amsterdam, The Netherlands, 2016, pp. 694–711, doi: 10.1007/978-3-319-46475-6_43.

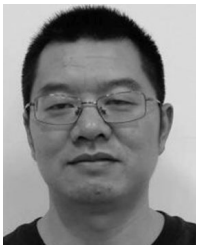
[50] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014, *arXiv:1409.1556*. [Online]. Available: http://arxiv.org/abs/1409.1556

[51] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Proc. NIPS*, Montreal, QC, Canada, 2014, pp. 2672–2680.

[52] B. Li, W. Ren, D. Fu, D. Tao, D. Feng, W. Zeng, and Z. Wang, "Benchmarking single-image dehazing and beyond," *IEEE Trans. Image Process.*, vol. 28, no. 1, pp. 492–505, Jan. 2019, doi: 10.1109/TIP.2018.2867951.

[53] Z. Zhang, H. Ma, H. Fu, and C. Zhang, "Scene-free multi-class weather classification on single images," *Neurocomputing*, vol. 207, pp. 365–373, Sep. 2016, doi: 10.1016/j.neucom.2016.05.015.

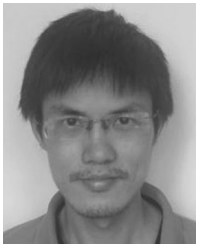
[54] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 6, pp. 1137–1149, Jun. 2017, doi: 10.1109/TPAMI.2016.2577031.



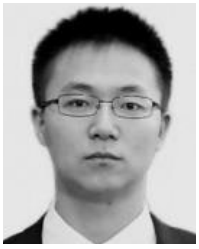
LIN GAO (Member, IEEE) received the B.S. degree in material forming and control engineering from the Huazhong University of Science and Technology, Wuhan, China, in 2002, and the M.S. degree in control engineering from the Wuhan University of Technology, Wuhan, in 2007. He is currently pursuing the Ph.D. degree with Sichuan University, Chengdu, China. He is also an Associate Professor with Hubei Minzu University, Enshi, China. His research interests include digital image processing, deep learning, and embedded systems.

WEI LONG received the B.S. and M.S. degrees in aeroengine control engineering from Northwestern Polytechnical University, and the Ph.D. degree in mechanical manufacturing and automation from Sichuan University, Chengdu, China, in 1998. He is currently a Professor with Sichuan University. He has published about 130 articles in domestic and international journals and conferences. He has presided over or participated in more than 50 scientific research projects, published eight academic works and obtained five invention patents. His research interests include industrial equipment automation and numerical control technology, mechanical equipment safety evaluation and reliability analysis, enterprise information control, vehicle/traffic intelligence technology, and manufacturing logistics and planning control.

YANYAN LI received the Ph.D. degree from Sichuan University. She has published about 16 articles in intelligence transportation system and obtained five invention patents. Her research interests include machine vision, vehicle/traffic intelligence technology, and deep learning manufacturing logistics and planning control.



HUAGUO LIU received the B.S. degree in mechanical engineering from West China University, Chengdu, China, in 2006, and the M.S. degree in mechanical engineering from Sichuan University, Chengdu, China, in 2009. He is currently pursuing the Ph.D. degree with Sichuan University, Chengdu, China. His research interests include e-government and safety evaluation of large equipment.



XIAOHONG YU received the B.S. degree in power machinery and engineering from the Hebei University of Technology, Tianjin, China, in 2015. He is currently pursuing the Ph.D. degree with Sichuan University, Chengdu, China. His research interests include machine vision, vehicle/traffic intelligence technology, and deep learning manufacturing logistics and planning control.

JUN LI received the M.S. degree in computer science from the Huazhong University of Science and Technology, in 1999. He is currently a Professor with the School of Information and Engineering, Hubei Minzu University. His research interests include image processing, computer graphics, and digital protection of intangible cultural heritages.