IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received December 29, 2020, accepted January 6, 2021, date of publication January 13, 2021, date of current version January 22, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3051359

Single-Image Snow Removal Based on an Attention Mechanism and a Generative Adversarial Network

AIWEN JIA[1,2], ZHEN-HONG JIA[1,2], JIE YANG[3], (Member, IEEE), AND NIKOLA K. KASABOV[4], (Fellow, IEEE)

[1]College of Information Science and Engineering, Xinjiang University, Ürümqi 830046, China
[2]Key Laboratory of Signal Detection and Processing, Xinjiang Uygur Autonomous Region, Xinjiang University, Ürümqi 830046, China
[3]Institute of Image Processing and Pattern Recognition, Shanghai Jiao Tong University, Shanghai 200400, China
[4]Knowledge Engineering and Discovery Research Institute, Auckland University of Technology, Auckland 1020, New Zealand

Corresponding author: Zhen-Hong Jia (jzhh9009@sohu.com)

This work was supported in part by the National Science Foundation of China under Grant U1803261, and in part by the International Science and Technology Cooperation Project of the Ministry of Education of the People's Republic of China under Grant DICE 2016–2196.

ABSTRACT Bad weather, such as snowfall, can seriously decrease the quality of images and pose great challenges to computer vision algorithms. In view of the negative effect of snowfall, this paper presents a single-image snow removal method based on a generative adversarial network (GAN). Unlike previous GANs, our GAN includes an attention mechanism in the generator component. By injecting attention information, the network can pay increased attention to areas covered by snow and improve its capability to perform local repairs. At the same time, we improve the traditional U-Net network by combining it with the residual network to enhance the effect of the model when removing snowflakes from a single image. Our experiments on both synthetic and real-word images show that our method produces better results than those of other state-of-the-art methods.

INDEX TERMS Snow removal, generative adversarial networks, attention mechanisms.

I. INTRODUCTION

As a special weather phenomenon, snowflakes reduce the visibility of background scenes, affect the clarity of images, and cause useful information in the images to disappear. These issues have a tremendous negative effect on subsequent image processing tasks, such as target detection [1], scenario analysis [2], and other image processing tasks [3]. Especially for the applications of artificial intelligence, clear and clean images are needed to extract and process correct information in most cases. Therefore, removing snowflakes from a single image is of great significance in the field of computer vision.

Existing snowflake removal methods for a single image can be divided into two types: traditional model-based and deep-learning-based methods. The first type mainly uses the spatial features of snow to detect and remove it from images. Pei *et al.* [4] used saturation and visibility characteristics to remove snowflakes from images by using high frequency filtering. Xu *et al.* [5] designed a refined guidance image. First, the original image was degraded, and then it was differentiated from the original image. By using the difference between the degraded image and the original image as the guidance, the authors reduced the degradation caused by dynamic weather and maintained detailed information about local regions. Ding *et al.* [6] claimed that the rain and snow components of an image have the characteristics of ridge edges, while other components have other edge properties such as those of step edges and valley edges. A guided L0 smoothing filter combined with edge properties was used to detect and remove the rain and snow components. Zheng *et al.* [7] took advantage of the frequency characteristics of images in which the rain and snow components were in the high-frequency portion, and the low-frequency portion did not include the rain and snow components. The low-frequency components were used as the guide graph to remove the rain and snow components from the high-frequency portion. Based on morphological analysis, Rajderkar and Mohod [8] used dictionary learning and sparse representation to detect rain and snow and employed smooth filtering to repair pixels covered by rain and snow. Unfortunately, this method causes

The associate editor coordinating the review of this manuscript and approving it for publication was Yi Zhang.

image blurring. Wang *et al.* [9] proposed a hierarchical approach for rain or snow removal in a single-color image. First, they distinguished the high-frequency and low-frequency components of the image and extracted the overcomplete dictionary of rain and snow components and nondynamic components at high frequencies from a three-layer hierarchy of the image. A guided filter was then applied to restore the rain and snow pixels. Finally, the authors summed the nondynamic components to obtain images with the rain and snow removed. Lu *et al.* [10] regarded snowflakes in the atmosphere as particles. They used the maximum value of the degree of polarization and the angle of polarization obtained by global analysis of the Stokes vector to accurately estimate atmospheric air-light at infinity and the transmission map. Huang *et al.* [11] used sparsity-based regularization to reconstruct a potentially snow-free image and proposed an autotuning mechanism to seek an improved reconstruction of a snow-free image via time-varying inertia weight particle swarm optimizers. Snowflakes are removed from the image through step-by-step iteration. Model-based methods for snowflake removal only consider one or several features of snowflakes. During the detection and repair processes, some detailed information is ignored and lost, resulting in image blurring.

Unlike traditional modeling methods, algorithms based on deep learning utilize the self-learning ability of the network to extract the features from an image to detect and remove the rain and snowflakes in the image. Liu *et al.* [12] proposed a multistage network called DesnowNet, which adopts a semitransparent recovery and residual generation module to recover images blurred by snowflakes. Lin *et al.* [13] used a pyramidal hierarchical design with lateral connections across different resolutions to enrich location information and reduce computational time, which is based on DesnowNet. Li *et al.* [14] designed a composite generative adversarial network (CGAN). Unlike the previous GAN, their generator network comprises a clean background module and a snow mask estimation module to extract useful information. Based on a 3D residual network, Yan *et al.* [15] utilized both contextual information and 3D scene structure information to effectively detect snowflakes of different sizes in low frequency (LF) images. Finally, an encoder-decoder-based LF image restoration network was proposed to restore the background image. Li *et al.* [16] proposed a multiscale tacked densely connected convolutional network to detect and remove snowflakes in an image. The results of the snowflake detection network were transmitted forward to guide the snow removal network, and the results of the snowflake removal network were transmitted backward to guide the snow removal network. In this way, snowflake detection and removal were achieved. Chen *et al.* [17] proposed a joint size and transparency-aware snow removal method of joint size that can address both transparent and nontransparent snow particles by applying the modified partial convolution. Yang *et al.* [18] proposed a deep-learning-based rain streak removal method injected with self-supervision. They created a fractal band learning network based on frequency band recovery to improve the capacity to capture discriminative features for deraining. Jaw *et al.* [19] proposed a framework based on a sequential dual attention deep network to remove rain streaks in a single image. They used sequential dual attention blocks and multi-scale feature aggregation modules to improve the removal of rain streaks. Yeh *et al.* [20] proposed a method relying on multi-scale residual learning and image decomposition to remove haze from images. They employed a deep residual convolutional neural network (CNN) and a simplified U-Net to avoid color distortion. However, this algorithm leads to significant image blurring.

To remove snowflakes from a single image, we developed a novel single-image snowflake removal method employing an attention mechanism and an improved U-Net on the basis of a GAN.

Compared to previous studies on snow removal from a single image, our method offers the following contributions.

(1) Our method takes advantage of an attention mechanism. The attention diagram of snowflakes is employed as the guide for improving the sensitivity of the network model to snowflakes, thereby improving the snowflake removal ability of the model.
(2) We combine a U-Net with a residual network (ResNet) to enhance the quality of recovered snow-free images.

This paper is organized as follows. Section 2 introduces the deep learning network framework related to our work. Section 3 presents our proposed network model and the loss function. Section 4 presents the experimental results and analysis, and the paper is briefly summarized in Section 5.

II. RELATED WORKS

In this section, we briefly review the basic model for snow removal from a single image and related deep learning network frameworks.

A. SINGLE-IMAGE SNOW REMOVAL MODEL

Snowy images can be seen as combinations of clean background pixels and snowflake-contaminated pixels, and these images can be expressed as follows:

$$I = B \otimes (1 - M) + S \otimes M \tag{1}$$

where I represents the input image, which is corrupted by snow, B represents clean background pixels, S represents snow pixels, M denotes a characteristic graph of the approximate snowflake position, and the operator $\otimes$ represents elementwise multiplication.

B. GENERATIVE ADVERSARIAL NETWORK (GAN)

In 2014, Goodfellow *et al.* [21] proposed the GAN framework for the first time. The structure of the GAN framework is shown in Fig. 1. As shown in the figure, this framework trains two models simultaneously: a generator network G and a discriminator network D. The former is trained to learn the true distribution of given data and create a generated sample,

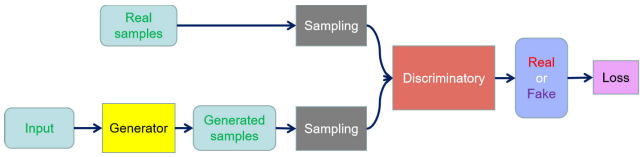


FIGURE 1. Architecture of a generative adversarial network (GAN).

while the latter is used to determine if a sample is a true sample. The training process of G involves trying to force D to make as many mistakes as possible, while the training process of D involves improving its ability to determine if a sample is a real sample or a sample generated by the generator network. With constant training, the generator is able to generate a fake sample that is sufficiently similar to the real sample. The loss function of a GAN is defined as follows:

$$\min_{G}\max_{D}\ \mathbb{E}_{x\subset p_{data}(x)}\left[\log\left(D(x)\right)\right] + \mathbb{E}_{z\subset p_z(z)}\left[1-\log\left(D(G(z))\right)\right] \quad (2)$$

where G represents the generator, D represents the discriminator, x is a sample from the real data (its label is known), and z is the sample produced by the generator (its label is unknown).

However, the traditional GAN has some problems, such as training instability, gradient disappearance and mode collapse. Arjovsky *et al.* [22] proposed the WGAN model, which includes the Wasserstein distance with superior smoothness. They solved the vanishing gradient and mode collapse problems faced by the GAN. Mao *et al.* [23] adopted the least squares loss function for the discriminator and proposed the LSGAN model, which increases the quality of the images generated by the network and stabilizes the training process at the same time.

Recently, GANs have been applied in many fields, such as image enhancement [24], image segmentation [25], target detection [26], image repair [27] and other applications [28]–[30].

C. RESIDUAL NETWORK (ResNet)

The main problems encountered by deep learning algorithms with network depth are vanishing gradients and exploding gradients. The general corresponding solutions to these problems are the initialization and batch normalization of data. However, these solutions cause other problems, such as the degradation of the performance of the network and increases in the network depth and error rate. Tulyakov *et al.* [28] proposed the ResNet in 2015. This network solves the problems of network degradation and gradient problems, thus improving the performance of the network.

ResNet includes a method for fitting the residual mapping, that is, the convolution result is not directly taken as the output, but the identity mapping is used for the calculation. Assume that the network has a hidden layer F(x) that satisfies the mapping relation F(x)=H(x)-x. If multiple nonlinear layers are combined, we can consider them as a complex

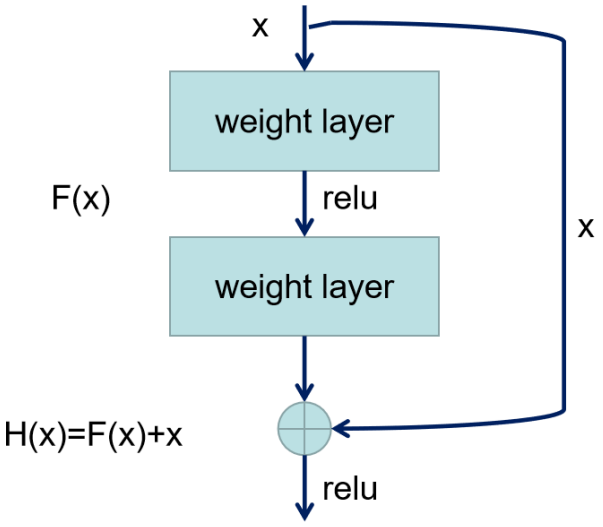


FIGURE 2. Architecture of the residual network.

network. Similarly, we can assume that the residual mapping of the hidden layer approximates a complex function: H(x)=F(x)+x. The structure of ResNet is shown in Fig. 2.

As shown in Fig. 2, ResNet performs feature extraction on the image by adding the outputs and inputs of multiple convolution hierarchies, thus reducing the number of training parameters used. Compared with other networks, ResNet is relatively simple with fewer training parameters and a shorter training time, thereby solving the performance degradation problem of deep CNNs. Consequently, ResNet has been widely used in computer vision.

D. U-NET

Ronneberger *et al.* [32] proposed the U-Net structure in 2015, by forming a symmetrical U-shaped structure for image feature extraction through an encoding network and a decoding network. The encoding network is mainly responsible for downsampling and extracting high-dimensional feature information. Each downsampling iteration contains two convolution operations and one pooling operation. Employing a rectified linear unit (ReLu) as the activation function halves the size of the sampling and doubles the number of features. The decoding network is mainly used for the upsampling. Each upsampling iteration contains two convolution operations, and the ReLu is modified as the activation function. With each upsampling step, the size of the image is twice that of the input, and the number of features is halved. During the upsampling processes, the output features of each iteration are combined with the features of the corresponding encoding network to complete the missing boundary information.

III. PROPOSED METHOD

In this section, we introduce our single-image snow removal model based on the GAN in detail, including the generator and discriminator. The architecture of our model is shown in Fig. 3. Additional details for each block are shown in Fig. 4 ∼Fig. 7.

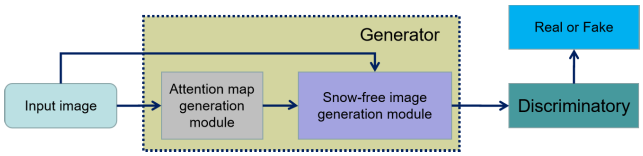


FIGURE 3. Architecture of our model.

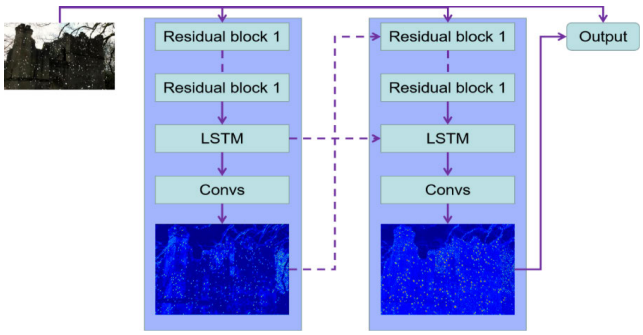


FIGURE 4. Architecture of the attention map estimation module.

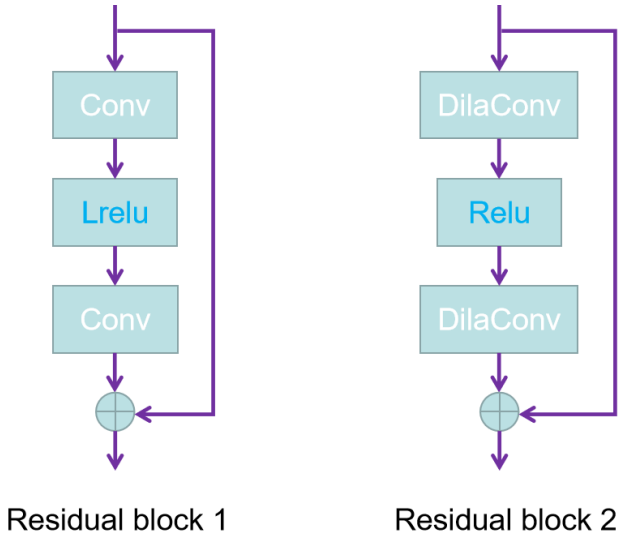


FIGURE 5. Architectures of residual blocks.

A. GENERATOR NETWORK

The generator network in our model consists of two portions: the attention map estimation module and the snow-free image generation module. The function of the attention map estimation module is to discover the snow-covered area by learning between the clean image and the image polluted by snowflakes. The snow-free image generation module can repair the snowy image by referring to the attention map.

1) ATTENTION MAP ESTIMATION MODULE

Inspired by Mnih *et al.* [35] and Qian *et al.* [36], we utilize a recurrent network to generate attention maps. The recurrent network consists of four blocks, and each block consists of five ResNet layers, one long-short-term memory (LSTM) layer and one convolutional layer. The structure of the network is shown in Fig. 4. In the network training phase, the input of this module consists of a snowy image, a snow-free image, and a binary mask of snow. After being processed by this module, the attention map of snowflakes is obtained and merged with the snowy image before being injected into the next module.

The series of ResNet layer is used to extract features from the input image, and its structure is shown in Fig. 5. The convolutional layer is used to generate a 2D attention map. The attention map generated by each block is also merged with the input images and injected into the next block at the same time. The final attention map is obtained through the learning processes of all four blocks

The LSTM unit contains an input gate i_t, a forgetting gate f_t, an output gate o_t, and a cell state gate C_t. The interactions between gates are defined as follows:

$$\begin{aligned} i_t &= \sigma(W_{xi} * X_t + W_{hi} * H_{t-1} + W_{ci} \otimes C_{t-1} + b_i) \\ f_t &= \sigma(W_{xf} * X_t + W_{hf} * H_{t-1} + W_{cf} \otimes C_{t-1} + b_f) \\ C_t &= f_t \otimes C_{t-1} + i_t \otimes \tanh(W_{xc} * X_t + W_{hc} * H_{t-1} + b_c) \\ o_t &= \sigma(W_{xo} * X_t + W_{ho} * H_{t-1} + W_{co} \otimes C_{t-1} + b_o) \\ H_t &= o_t \otimes \tanh(C_t) \end{aligned} \tag{3}$$

where X_t represents the features generated by ResNet, H_t denotes the final output features extracted by the LSTM unit, C_t is the unit state provided to the next LSTM, b_i, b_f, b_c, and b_o represent the biases of the input gate, forgetting gate, cell state gate and output gate, respectively, and the operator $*$ stands for the convolution operation.

When training the attention map estimation network, we use pairs of images polluted by snow and snow-free images, with both having the same background. During each training process, the loss function of the network is defined as the mean squared error (MSE) between the output attention map and the binary mask. The loss function is expressed as follows:

$$\ell_{\mathrm{AML}}(\{A\}, M) = \sum_{t=1}^{N} \alpha^{N-t} \ell_{MSE}(A_t, M) \tag{4}$$

where A_t represents the attention map generated by the attention map estimation network at time step t, $A_t = AML_t(F_{t-1}, H_{t-1}, C_{t-1})$, F_{t-1} is the splicing of the input image and the attention map generated by the previous training process, and M represents the binary mask of snow, which can be obtained when the snowy image is synthesized. In our model, $N = 4$ and $\alpha = 0.7$. We set the initial attention map with values of 0.5, and as the number of training steps increases, the values of the pixels covered by snow increase gradually.

2) SNOW-FREE IMAGE GENERATION MODULE

In our proposed model, we use a U-Net-based network to generate snow-free images. The generator module consists of an encoding component and a decoding component. Taking advantage of ResNet, we combine ResNet with U-Net to improve the quality of the recovered snow-free images with a small increase in network complexity. The structure of this network is shown in Fig.6. In the network training phase,

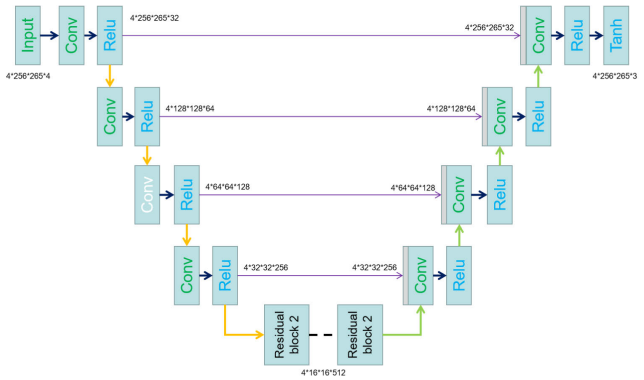


FIGURE 6. Architecture of the snow-free image generation module.

the attention map and snowy image are input into this module, and a snow-free image is output after processing.

The encoding component is based on the downsampling operation while the decoding component employs the upsampling operation. Skip connections are utilized to retain the details of the image. In the fifth layer of U-Net, we employ five ResNet blocks containing dilated convolutions. The structure of this ResNet is shown in Fig. 5. U-Net has deeper levels and more training parameters than ResNet, which extracts more features for restoring the image. At the same time, this method avoids training times that are too long and overfitting [31]. The function of dilated convolution is to enlarge the receptive field of the network without increasing the complexity of the parameters.

As shown in Fig. 6, the input of the first convolutional layer is a 4D array. The first dimension represents the batch size, and the last dimension represents the number of feature channels. In the encoding component, we set 32 first-level feature channels and then double the number of feature channels step by step until reaching a total of 512. Accordingly, in the decoding component, we gradually reduce the number of feature channels in the upsampling portion until the color snow-free image is generated. We employ the mean absolute error (MAE) to express the difference between the generated image and the original image, with different scales generated by different levels. The loss function of the network is defined as follows:

$$\ell_{\text{Unet}}(\{\text{R}\},\{\text{T}\}) = \sum_{i=1}^{\text{I}} \beta_i \ell_{MAE}(\text{R}_i, \text{T}_i) \tag{5}$$

where R_i represents the i-th output image of the decoder, T_i represents the corresponding snow-free image, and β_i represents the weights of the loss at different scales. Theβ values of the 1st, 2nd, 3rd, and 4th layers from the end of the structure are set to 1,0.8,0.6, and 0.5, respectively. In addition, we use a perceptual loss [34] to calculate the global difference between the output of the snow-free image generation module and the clean image. A trained CNN, such as the VGG16 network trained on the ImageNet dataset, is employed to extract this discrepancy. The loss function can be rewritten as

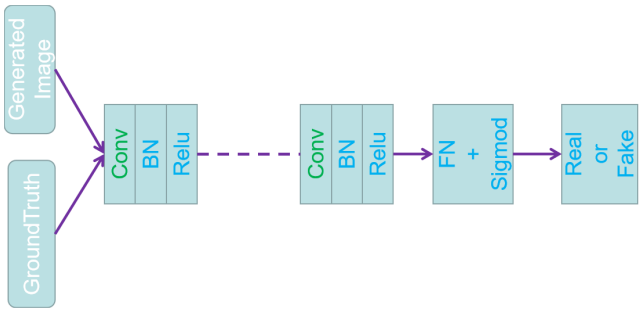


FIGURE 7. Architecture of the discriminator network.

follows:

$$\ell_{PL}(\{\text{O}\},\{\text{T}\}) = \sum_{i=1}^{\text{I}} \ell_{MAE}(\text{VGG}(\text{O}_i), \text{VGG}(\text{T}_i)) \tag{6}$$

where O_i represents the output image of the generation module, T_i denotes the corresponding clean image, and VGG is a pretrained CNN used to extract image features.

To sum up, the loss function of the generator network can be expressed as follows:

$$\ell_{\text{G}} = \ell_{\text{AMI}}(\{\text{A}\}, \text{M}) + \ell_{\text{Unet}}(\{\text{R}\}, \{\text{T}\}) + \ell_{PL}(\{\text{O}\}, \{\text{T}\}) + \ell_{\text{GAN}}(\text{O}) \tag{7}$$

where $l_{GAN}(O) = \log(1 - D(O))$ and O represents the generator of the final output image.

B. DISCRIMINATOR NETWORK

The discriminator network is used to classify the input image as real or fake. As shown in Fig. 7, the discriminator contains four groups of convolutional layers. For each layer, there is a convolutional layer followed by a batch normalization layer and a ReLu activation layer. A fully connected layer and a single neuron with a sigmoid activation operation are placed in the last layer for the output.

The loss function of the discriminator network is defined as follows:

$$\ell_{\text{D}} = -\log(\text{D}(\text{T})) - \log(1 - \text{D}(\text{G}(\text{I}))) \tag{8}$$

where T represents the real snow-free image, and I denotes the input snowy image.

IV. EXPERIMENTAL ANALYSIS

In this section, we introduce the data and details of the training process. In the following subsections, the effects of different methods are evaluated from various aspects in detail.

A. DATASET

In this paper, a snow dataset named Snow100K[2] [8] is utilized for training and testing, and it contains synthesized snowy images, relevant clean images and snow masks. We employ 8000 snow masks of disparate scales and 10000 clean background images to generate 18620 synthesized snowy images. The dataset is divided into a training set and a test set at a ratio of 8:2 to improve the performance of the network.

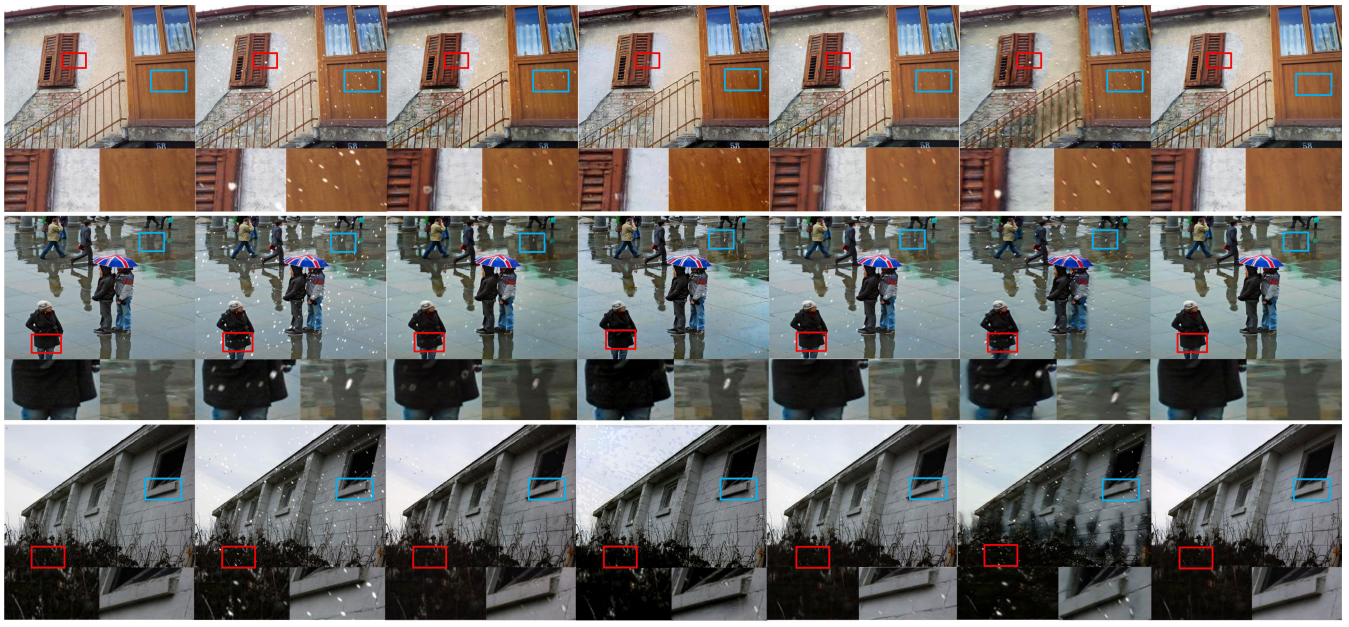


FIGURE 8. Example synthetic image results. (a) Ground truth, (b) Snowy image, (c) Li et al. [9], (d) Chen et al. [17], (e)Qian et al. [36], (f) Yang et al. [38], (f) our method.

B. TRAINING DETAILS

We train the network on an NVIDIA Tesla V100 GPU. Our proposed method is implemented using TensorFlow 1.12.0 and Python 3.6.0. The parameters of the learning rate and batch size are set to 0.0004 and 4, respectively. All training images are resized to 256*256. In addition, the generator network and discriminator network of the GAN are trained at the same time, and their parameters are updated accordingly.

C. RESULTS ANALYSIS

In this section, we show the experimental results of our method along with those of other state-of-art algorithms in terms of removing snowflakes from a single image: the algorithms of Wang *et al.* [9], Liu *et al.* [12], Qian *et al.* [36], and Yang *et al.* [38]. We analyze the experimental results from different perspectives. In this paper, we present six synthesized snowy images, as shown in Fig. 8 and Fig. 10, which are included in these test set. We also employ real-world snowy images obtained from YouTube, was shown in Fig. 9.

1) QUANTITATIVE EVALUATION

Table 1 shows quantitative comparisons between our method and other existing methods using the PSNR [39] and SSIM [40] metrics, which are based on images in Fig. 8. As shown in the table, compared with those of other methods, the PSNR and SSIM values obtained by our method are higher. This indicates that the snow-free image generated by our method is closer to the real snow-free image than the images generated by the other algorithms.

We also compare the network complexity and the run time complexity of our method and the other methods. We utilize processing time per image to present the run time complexity. As shown in Table 3, the method of Yang *et al.* [38] has the fastest processing time, while the method of Chen *et al.* [17] has the slowest. Our method is slower than the methods of Wang *et al.* [9] and Yang *et al.* [38], but faster than other competing algorithms. To compute the network complexity, we employ floating point operations (FLOPs), which represents the calculated amount. In Table 4, we show the FLOPs of our method and the methods of Wang *et al.* [9],

TABLE 1. Performance comparison of all competing methods on synthetic snow videos in terms of the PSNR, and SSIM metrics in figure 8.

	1st image		2nd image		3rd image	
	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
Li *et al.* [9]	23.95	0.9198	24.96	0.9214	28.05	0.9389
Chen *et al.* [17]	24.56	0.8834	26.19	0.8995	25.47	0.8847
Qian *et al.* [36]	25.36	0.9362	26.45	0.9366	28.24	0.9442
Yang *et al.* [38]	23.64	0.8633	24.85	0.8712	25.39	0.8381
Our	28.26	0.9780	29.71	0.9791	29.15	0.9597

TABLE 2. Performance comparison of loss function ablation experiments in terms of the PSNR and SSIM metrics.

	1st image		2nd image	
	PSNR	SSIM	PSNR	SSIM
c	28.5508	0.9091	27.6322	0.8864
d	20.6313	0.6771	21.6481	0.7012
e	24.2146	0.7879	23.9517	0.7956
f	29.8812	0.9493	28.5334	0.9233

TABLE 3. Processing time comparison.

	Fig. 8			Fig. 9	
	1st image	2nd image	3rd image	1st image	2nd image
Li *et al.* [9]	14.22s	14.56s	15.96s	14.61s	14.13s
Chen *et al.* [17]	91.19s	90.13s	94.01s	92.03s	91.34s
Qian *et al.* [36]	15.41s	15.48s	17.40s	15.98s	15.13s
Yang *et al.* [38]	3.81s	3.08s	3.62s	3.70s	3.95s
our	14.63s	14.54s	16.76s	14.83s	14.61s

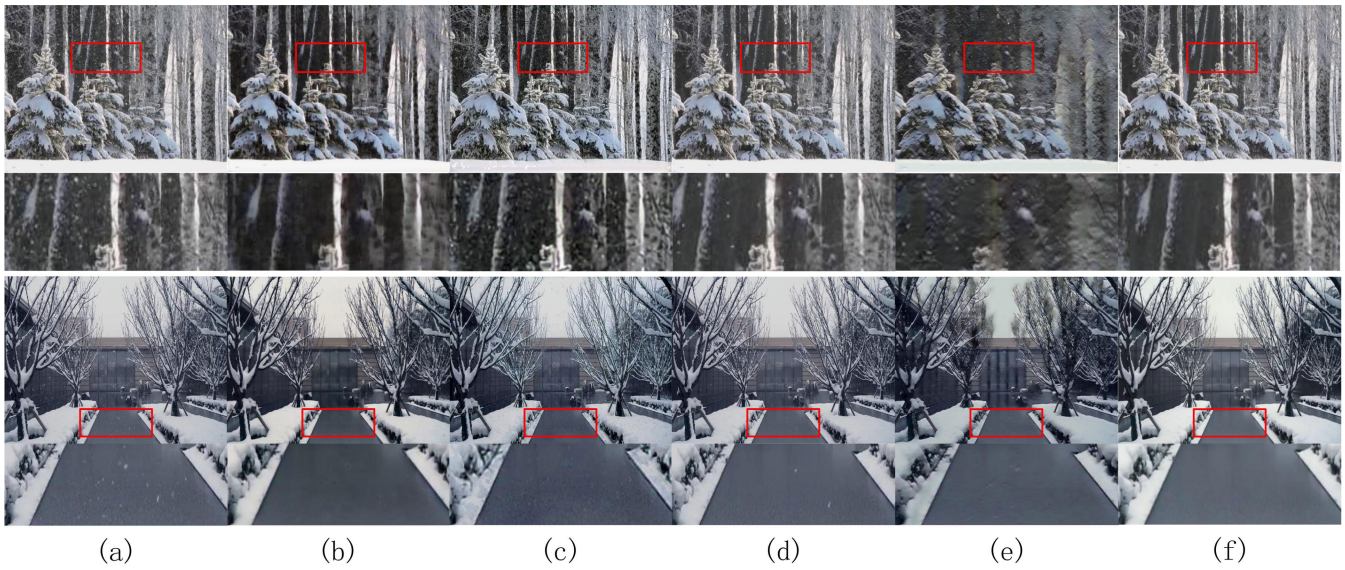


FIGURE 9. Example real-world images results. (a) Snowy image, (b) Li et al. [9], (c) Chen et al. [17], (d) Qian et al. [36], (e) Yang et al. [38], (f) our method.

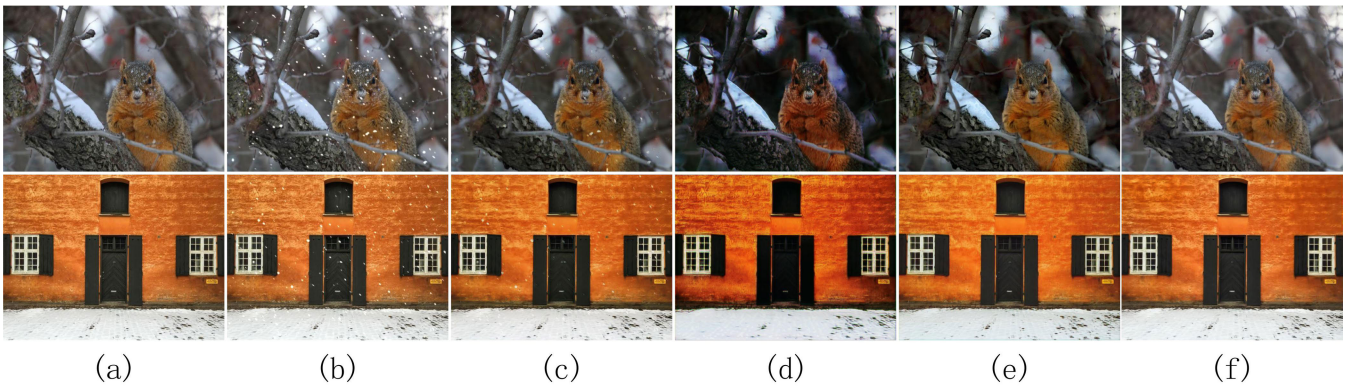


FIGURE 10. Example results of the ablation study. (a) Ground truth, (b) snowy image.

TABLE 4. The network complexity of our method compared with state-of-art methods in terms of flops.

	FLOPs
Li *et al.* [9]	5.06E85
Qian *et al.* [36]	4.11E86
Yang *et al.* [38]	2.33E73
our	7.21E85

Qian *et al.* [36], Yang *et al.* [38]. As shown in the table, our framework is less complex than that of Qian *et al.* [36] but more so than the others.

2) QUALITATIVE EVALUATION

In this section, we show the qualitative evaluation of the performances of the proposed method and the methods of Wang *et al.* [9], Chen *et al.* [17], Qian *et al.* [36], and Yang *et al.* [38]. We conduct experiments on both the synthetic images and the real image to provide convincing results. Fig. 8 shows the results obtained by the algorithms of Wang *et al.* [9], Chen *et al.* [17], and Yang *et al.* [38] in comparison with our results. As seen from Fig. 8, the algorithm of Yang *et al.* [38] removes only a few snowflakes from the image, and it causes considerable blurring of the image at the same time. By comparison, the method of Wang *et al.* [9] removes more snow; however, distortion is generated when the snowflake pixels are repaired. Some snowflake pixels have not been completely fixed, as the second group image shows. The algorithm of Chen *et al.* [17] removes some snowflakes while some snow remains. Unfortunately, images processed by the method of Chen *et al.* [17] lose the details and produce some artifacts, which are obvious in the third group image. The method of Qian *et al.* [36] removes most of the snowflakes in the image, while some still remained. In contrast, our method removes snowflakes and produces a snow-free image that most closely resembles the real background

Fig. 9 shows the resulting real-world snowy images generated by different methods. As shown in Fig. 9, the algorithm of Yang *et al.* [38] only removes some of the snowflakes from the image and loses background information while repairing pixels covered by snowflakes. As shown in the first group image, the method of Chen *et al.* [17] fails to remove snow from the real-world image. A large number of snowflakes

remained after the treatment. The algorithm of Qian *et al.* [36] removes some of the snowflakes and saves more background information. However, there still some noticeable snowflakes remain. As can been seen clearly in Fig. 9, the method of Wang *et al.* [9] removes some of the snowflakes but causes a substantial amount of blurring in the image. Compared to the previous algorithm, our method removes snowflakes thoroughly and preserves background information integrally. By comparison, the images generated by our method are clearer than the images generated by the other algorithms.

D. ABLATION STUDY

To study the effectiveness of each module and the loss function in our proposed network, we conducted an ablation study, and the results are shown in Table. 2 and Fig. 10. Subfigures (c), (d), and (e) represent the loss function without ℓ_{AML}, ℓ_{PL}, and ℓ_{Unet}, respectively. Subfigure (f) denotes the completed loss function. As seen from Fig. 10, without ℓ_{AML},the attention map estimation module is not trained, so some snowflakes will remain. Without ℓ_{PL}, there will be color distortion in the images. The snow-free image generation module is not well trained without ℓ_{Unet}, which causes blurring in the generated images. The completed loss function performs better than the partial loss function.

V. CONCLUSION

In this paper, we propose a single-image snow removal model based on an attention mechanism and a GAN. We use the attention mechanism to detect snowflakes in a single image and make the snow-free image generation module pay increased attention to the pixels covered by snowflakes when repairing the image by using an attention map. To obtain high-quality snow-free images, we improve U-Net to increase the amount of available information. Experiments on synthetic images and real-world images show that our method has advantages over other snow removal methods. In future work, we will focus on the problem of misjudgments of snow and improve the ability of our model to deal with real-world snow scenes.

REFERENCES

[1] O. L. Junior, D. Delgado, V. Goncalves, and U. Nunes, "Trainable classifier-fusion schemes: An application to pedestrian detection," in *Proc. 12th Int. IEEE Conf. Intell. Transp. Syst.*, Oct. 2009, pp. 1–6.

[2] L. Itti, C. Koch, and E. Niebur, "A model of saliency-based visual attention for rapid scene analysis," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 20, no. 11, pp. 1254–1259, Nov. 1998.

[3] M. Farenzena, L. Bazzani, A. Perina, V. Murino, and M. Cristani, "Person re-identification by symmetry-driven accumulation of local features," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, Jun. 2010, pp. 2360–2367.

[4] S.-C. Pei, Y.-T. Tsai, and C.-Y. Lee, "Removing rain and snow in a single image using saturation and visibility features," in *Proc. IEEE Int. Conf. Multimedia Expo Workshops (ICMEW)*, Jul. 2014, pp. 1–6.

[5] J. Xu, W. Zhao, P. Liu, and X. Tang, "An improved guidance image based method to remove rain and snow in a single image," *Comput. Inf. Sci.*, vol. 5, no. 3, p. 49, Apr. 2012.

[6] X. Ding, L. Chen, X. Zheng, Y. Huang, and D. Zeng, "Single image rain and snow removal via guided l0 smoothing filter," *Multimedia Tools Appl.*, vol. 75, no. 5, pp. 2697–2712, Mar. 2016.

[7] X. Zheng, Y. Liao, W. Guo, X. Fu, and X. Ding, "Single-image-based rain and snow removal using multi-guided filter," in *Proc. Int. Conf. Neural Inf. Process.*, 2013, pp. 258–265.

[8] D. Rajderkar and P. S. Mohod, "Removing snow from an image via image decomposition," in *Proc. IEEE Int. Conf. Emerg. Trends Comput., Commun. Nanotechnol. (ICECCN)*, Mar. 2013, pp. 576–579.

[9] Y. Wang, S. Liu, C. Chen, and B. Zeng, "A hierarchical approach for rain or snow removing in a single color image," *IEEE Trans. Image Process.*, vol. 26, no. 8, pp. 3936–3950, Aug. 2017.

[10] Y. Lu, M. Xu, S. Jia, and W. Huang, "Fast snow removal algorithm based on the maximum value of the degree of polarization and angle of polarization," *Phys. Scripta*, vol. 94, no. 4, Apr. 2019, Art. no. 045501.

[11] S.-C. Huang, D.-W. Jaw, B.-H. Chen, and S.-Y. Kuo, "Single image snow removal using sparse representation and particle swarm optimizer," *ACM Trans. Intell. Syst. Technol.*, vol. 11, no. 2, pp. 1–15, Mar. 2020.

[12] 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.

[13] C.-Y. Lin, Z. Tao, A.-S. Xu, L.-W. Kang, and F. Akhyar, "Sequential dual attention network for rain streak removal in a single image," *IEEE Trans. Image Process.*, vol. 29, pp. 9250–9265, 2020.

[14] 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.

[15] T. Yan, Y. Ding, F. Zhang, N. Xie, W. Liu, Z. Wu, and Y. Liu, "Snow removal from light field images," *IEEE Access*, vol. 7, pp. 164203–164215, 2019.

[16] P. Li, M. Yun, J. Tian, Y. Tang, G. Wang, and C. Wu, "Stacked dense networks for single-image snow removal," *Neurocomputing*, vol. 367, pp. 152–163, Nov. 2019.

[17] W.-T. Chen, H.-Y. Fang, J.-J. Ding, C.-C. Tsai, and S.-Y. Kuo, "JSTASR: Joint size and transparency-aware snow removal algorithm based on modified partial convolution and veiling effect removal," in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 754–770.

[18] W. Yang, S. Wang, and J. Liu, "Removing arbitrary-scale rain streaks via fractal band learning with self-supervision," *IEEE Trans. Image Process.*, vol. 29, pp. 6759–6772, 2020.

[19] D.-W. Jaw, S.-C. Huang, and S.-Y. Kuo, "DesnowGAN: An efficient single image snow removal framework using cross-resolution lateral connection and GANs," *IEEE Trans. Circuits Syst. Video Technol.*, early access, Jun. 17, 2020, doi: 10.1109/TCSVT.2020.3003025.

[20] 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.

[21] I. Goodfellow, J. Pouget-Abadie, and M. Mirza, "Generative adversarial nets," in *Proc. Adv. Neural Inf. Process. Syst.*, 2014, pp. 2672–2680.

[22] M. Arjovsky, S. Chintala, and L. Bottou, "Wasserstein GAN," 2017, *arXiv:1701.07875*. [Online]. Available: http://arxiv.org/abs/1701.07875

[23] X. Mao, Q. Li, H. Xie, R. Y. K. Lau, Z. Wang, and S. P. Smolley, "Least squares generative adversarial networks," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 2794–2802.

[24] X. Wang, K. Yu, S. Wu, J. Gu, Y. Liu, C. Dong, Y. Qiao, and C. C. Loy, "ESRGAN: Enhanced super-resolution generative adversarial networks," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 63–79.

[25] A. Kumar Mondal, J. Dolz, and C. Desrosiers, "Few-shot 3D multi-modal medical image segmentation using generative adversarial learning," 2018, *arXiv:1810.12241*. [Online]. Available: http://arxiv.org/abs/1810.12241

[26] J. Li, X. Liang, Y. Wei, T. Xu, J. Feng, and S. Yan, "Perceptual generative adversarial networks for small object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 1222–1230.

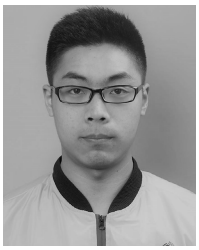
[27] C. Zheng, T.-J. Cham, and J. Cai, "Pluralistic image completion," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 1438–1447.

[28] S. Tulyakov, M.-Y. Liu, X. Yang, and J. Kautz, "MoCoGAN: Decomposing motion and content for video generation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 1526–1535.

[29] O. Press, A. Bar, B. Bogin, J. Berant, and L. Wolf, "Language generation with recurrent generative adversarial networks without pre-training," 2017, *arXiv:1706.01399*. [Online]. Available: http://arxiv.org/abs/1706.01399

[30] H. Shi, J. Dong, W. Wang, Y. Qian, and X. Zhang, "SSGAN: Secure steganography based on generative adversarial networks," in *Proc. Pacific Rim Conf. Multimedia*, 2017, pp. 534–544.

[31] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.
[32] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional networks for biomedical image segmentation," in *Proc. Int. Conf. Med. Image Comput. Comput.-Assist. Intervent.*, 2015, pp. 234–241.
[33] Y. Song, C. Yang, Z. Lin, X. Liu, Q. Huang, H. Li, and C.-C. J. Kuo, "Contextual-based image inpainting: Infer, match, and translate," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 3–19.
[34] H. Liu, B. Jiang, Y. Xiao, and C. Yang, "Coherent semantic attention for image inpainting," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 4170–4179.
[35] V. Mnih, N. Heess, and A. Graves, "Recurrent models of visual attention," in *Proc. Adv. Neural Inf. Process. Syst.*, 2014, pp. 2204–2212.
[36] 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. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 2482–2491.
[37] J. Johnson, A. Alahi, and L. Fei-Fei, "Perceptual losses for real-time style transfer and super-resolution," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 694–711.
[38] W. Yang, R. T. Tan, J. Feng, Z. Guo, S. Yan, and J. Liu, "Joint rain detection and removal from a single image with contextualized deep networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 6, pp. 1377–1393, Jun. 2020.
[39] Q. Huynh-Thu and M. Ghanbari, "Scope of validity of PSNR in image/video quality assessment," *Electron. Lett.*, vol. 44, no. 13, pp. 800–801, Jun. 2008.
[40] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: From error visibility to structural similarity," *IEEE Trans. Image Process.*, vol. 13, no. 4, pp. 600–612, Apr. 2004.



AIWEN JIA received the bachelor's degree in electronic information engineering from the School of Information Science and Engineering, Liaocheng University, China, in 2018. He is currently pursuing the master's degree from the School of Information Science and Engineering, Xinjiang University, China.

His research interest includes the clarity of surveillance videos and images in snow environments.

ZHEN-HONG JIA received the B.S. degree from Beijing Normal University, Beijing, China, in 1985, and the M.S. and Ph.D. degrees from Shanghai Jiao Tong University, Shanghai, China, in 1987 and 1995, respectively.

He is currently a Professor with the Key Laboratory of Signal Detection and Processing, Xinjiang Uygur Autonomous Region, Xinjiang University, China. His research interests include digital image processing, photoelectric information detection, and sensor.

JIE YANG (Member, IEEE) received the Ph.D. degree from the Department of Computer Science, Hamburg University, Germany, in 1994.

He is currently a Professor with the Institute of Image Processing and Pattern Recognition, Shanghai Jiao Tong University, China. His major research interests include object detection and recognition, data fusion and data mining, and medical image processing.

NIKOLA K. KASABOV (Fellow, IEEE) received the M.S. degree in computing and electrical engineering and the Ph.D. degree in mathematical sciences from the Technical University of Sofia, Sofia, Bulgaria, in 1971 and 1975, respectively.

He is currently the Director and the Founder of the Knowledge Engineering and Discovery Research Institute, and a Professor of Knowledge Engineering with the School of Computing and Mathematical Sciences, Auckland University of Technology, Auckland, New Zealand. His major research interests include information science, computational intelligence, neural networks, bioinformatics, neuroinformatics, speech and image processing. He has published more than 650 works in these areas.