

Research Article

DR-Net: A Novel Generative Adversarial Network for Single Image Deraining

Chen Li,^{1,2} Yecai Guo ,^{1,2} Qi Liu,³ and Xiaodong Liu³

¹School of Electronic and Information Engineering, Nanjing University of Information Science & Technology, Nanjing 210023, China

²Jiangsu Collaborative Innovation Center of Atmospheric Environment and Equipment Technology, Nanjing 210023, China

³School of Computing, Edinburgh Napier University, Edinburgh EH10 5DT, UK

Correspondence should be addressed to Yecai Guo; guo-yecai@163.com

Received 31 August 2018; Revised 8 November 2018; Accepted 13 November 2018; Published 3 December 2018

Guest Editor: Weizhi Meng

Copyright © 2018 Chen Li et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Blurred vision images caused by rainy weather can negatively influence the performance of outdoor vision systems. Therefore, it is necessary to remove rain streaks from single image. In this work, a multiscale generative adversarial network- (GAN-) based model is presented, called DR-Net, for single image deraining. The proposed architecture includes two subnetworks, *i.e.*, generator subnetwork and discriminator subnetwork. We introduce a multiscale generator subnetwork which contains two convolution branches with different kernel sizes, where the smaller one captures the local rain drops information, and the larger one pays close attention to the spatial information. The discriminator subnetwork acts as a supervision signal to promote the generator subnetwork to generate more quality derained image. It is demonstrated that the proposed method yields in relatively higher performance in comparison to other state-of-the-art deraining models in terms of derained image quality and computing efficiency.

1. Introduction

In recent years, with the development of network communication, the intelligent monitoring system based on image and video processing technology has achieved promising progress. Such system plays a vital role in the maintenance of public security. Therefore, some computer vision issues related to the intelligent monitoring system have attracted a wide spread attention. Most of the computer vision algorithms proposed for addressing these issues can work well on the high visibility of video or image data. However, when the algorithms face degraded data, their performances may obviously degrade. This is because the training processes of these algorithms are based on the high visibility of the video or image dataset. Bad weather such as rainy days seriously degrades the visual quality of the captured videos or images, which may affect the performance of many computer vision algorithms like tracking, recognition, and retrieval [1]. This kind of circumstance may happen on some safely related accidents recorded by mobile phone or monitoring camera in rainy days. In such circumstance, the captured video or image data may contain a large number of fast-moving rain streaks,

which leads to the image signal distortion as well as reducing the signal-to-noise ratio and image quality. These impacts caused by rainy weather on video or image data have brought great difficulties for intelligent traffic, outdoor monitoring, military reconnaissance, and so on [2, 3]. In order to enhance the reliability of outdoor computer vision systems, there is a need to exploit effective algorithms to remove rain streaks from the degraded single image caused by rainy weather.

Mathematically, the process of the deraining procedure can be written as

$$I - O = R, \quad (1)$$

where I represents the rainy image and O and R represent rain streaks and restored clean image, respectively.

In the past few decades, some algorithms have been proposed to jointly address the rain detection and removal task. According to their concerns, these methods can be roughly classified into two categories, *i.e.*, video-based methods and single-image-based methods. Video-based deraining methods concentrate on eliminating the rain streaks from video sequence [4, 5] through exploiting frequency properties and

temporal information of rain streaks. Single-image-based methods consider the problem from two aspects:

(1) A task of blind image single decomposition: These methods mainly include morphological component analysis with sparse coding [2, 3], generalized low rank model [6], structural similarity constraints [7], and nonlocal means smooth [8].

(2) A task of learning an end-to-end projection between the rainy image and its corresponding derained clean image: Recently, due to their relatively superior ability of learning nonlinear functions, some deep learning-based methods have been proposed to address this issue, in which an end-to-end projection between the rain image and its corresponding ground truth is directly learned. These methods contain convolution neural network- (CNN-) based models [9, 10] and generative adversarial network-based models (GAN) [11, 12].

Although existing methods have achieved some degree of success, there still exist several limitations caused by the following aspects: (1) For the basic operations of many existing approaches are performed on a small receptive field or a local image patch, the spatial contextual information between the patches or receptive fields is usually ignored. (2) Since the background texture patterns and rain streaks are internally overlapping, texture details in nonrain regions are removed by most approaches, which results in the restored images containing some oversmoothness regions. (3) Some of these models introduce additional image enhancement techniques to improve the visual effect of image, which reduces the efficiency of the algorithm.

In order to alleviate these limitations mentioned above, our goal is to exploit a novel architecture with the ability of removing rain and keeping the restored clean image details jointly. More specifically, we introduce a multiscale GAN-based architecture called single image deraining generative adversarial network (DR-Net) to handle the single image deraining issue. The architecture includes two subnetworks, *i.e.*, generator network and discriminator network. Generator network acts as a feature extractor that can eliminate the rain streaks while encoding the image contents. In other words, it learns a nonlinear projection function which transfers a rainy image into a restored clean image and keeps the details of raw image simultaneously. To capture more spatial information and local rain drops, we propose a multiscale parallel convolution generator network which consists of two-branch convolution operator with different kernel sizes. Discriminator network uses the restored image generated by the generator architecture and ground truth image as inputs. It aims at differentiating the restored clean image from the real ground truth image. The function of discriminator network is to boost the generator architecture to generate more quality derained image that closely resembles the ground truth.

To sum up, the contributions of this work are as follows:

(1) We design a novel generative adversarial architecture to address the single image deraining issue. The generator architecture consists of two parallel convolution subnetworks that have different kernel sizes, specifically, one subnetwork with large kernel size that captures more spatial information

of the raw image, and the other subnetwork with relatively small kernel size which aims to acquire more local rain streaks knowledge. The multiscale operators are helpful for keeping the details of raw image and eliminating rain streaks at the same time. Additionally, with fewer feature maps, our network has the advantages of less parameters and less computing effort; thus, the training convergence and test speed are relatively faster among the comparison methods.

(2) Experiments on publicly synthesized dataset and real images show the effectiveness of the proposed network. The proposed model performs better than other recent state-of-the-art single image deraining techniques.

The remainder of this paper is organized as follows. A brief review on existing methods for image deraining is given in Section 2. Section 3 provides the detail of the proposed DR-Net architecture. Section 4 presents the experiment results on both synthetic and real images. Finally, a brief discussion is concluded in Section 5.

2. Related Works

In the past few years, a number of models have been presented to enhance the visibility of images captured with rain streaks. These models can be divided into two categories: video sequences-based methods and single-image-based methods. In this section, we give a brief review of these image deraining models.

(A) *Video Sequences-Based Methods.* Video sequences-based rainy image recovery has been extensively studied. Garg et al. [1] proposed a deraining model for the rain streak detection and removal from video sequences. The rain streaks detection has two constraints [13]: (1) First, since the rain streaks are dynamic, the changes in intensity inside their several frames are comparatively high. (2) Second, because other objects may also be dynamic, through examining whether the relations between the intensity changes along the streak and background intensity are photometrically linear, the rain streaks can be differenced. The second constraint can reduce the error alarms caused by the first. After detecting the rain streaks, the average intensity of the pixels taken from the previous and subsequent frames is used to remove the streaks. Soon after this, they further developed a postprocessing architecture for video sequences-based deraining [14]. Specifically, they first proposed a photometric method which can describe the intensities generated by individual rain. Then, a dynamic model that can capture spatiotemporal attributes of rain streaks was presented. Finally, they used these models together to describe the visual appearance of rain streaks. Zhang et al. [15] introduced another constraint named chromaticity constraint. They point out that the intensity changes in the R, G, and B channels are alike for representing rain streaks. Based on the size information and photometry properties of rain streaks, Bossu et al. [16] proposed a rain detection algorithm to fit a Gaussian distribution on rain streak histograms. They adopted a Gaussian mixture model to separate the foreground used to detect the rain streaks from the background in video sequences.

(B) *Single-Image-Based Methods*. Since there are no temporal information for rain streaks detection and removing, compared with the video sequences-based removal issue, single-image rain removal (SIRR) is more challenging. Some researchers regard the SIRR problem as a task of layer separation. Kang et al. [3] used a bilateral filter to decompose the rainy image into high-frequency and low-frequency parts. Then, they utilized sparse coding and dictionary learning to separate the rain component from the high-frequency part. Through analyzing the aspect ratio of elliptical kernel and the orientation angle in each pixel location, Kim et al. [8] first detected the rain streaks regions. Then, they used adaptive nonlocal means filter to these rain streak regions to remove the rain streaks. Luo et al. [17] proposed a nonlinear screen blend model to model the rainy images. Specifically, through learning a dictionary with reciprocal exclusivity, they used sparse coding to separate the rain layer and derained layer. Since the rain streaks on the imaging scene usually appear recursively with similar patterns, Chen et al. [6] proposed and generalized a low-rank rain appearance architecture to capture the spatiotemporal relationship between rain streaks. Li et al. [13] proposed a model that uses patch-based priors which are based on Gaussian mixture model for rain and background layers. Moreover, these priors could accommodate multiple scales and orientations of the rain streaks to certain degree.

Recently, deep learning-based methods have achieved outstanding performances in many domains [18–21], including image deblurring [22], image denoising [23], superresolution [24], style transfer [25, 26], and inpainting [27]. There also exists some literature that adopts the deep learning system to address the SIRR issue. These deep learning-based methods aim to learn a nonlinear mapping between the rainy image and corresponding deraining image. These methods can be mainly divided into CNN-based and GAN-based models. Fu et al. [9] designed a convolution neural network named DrainNet for removing rain streaks from single image. They first decomposed the input rainy image into its detail layer and base layer, in which the base layer keeps the structure and detail layer contains object details and rain streaks. Then, they used the detail layer as the input of deep architecture to detect and remove rain streaks. Finally, they added the output of the deep architecture to the base layer to obtain the final output. Yang et al. [28] proposed a multitask deep convolution neural network which can simultaneously learn the appearance of rain streaks, the binary rain streak, and the background. Besides, they developed a recurrent rain detecting and removing network to clear up the rain accumulation and remove rain streaks iteratively. As a popular deep learning technology, GAN [29, 30] has been adopted in many computer vision tasks. Most recently, Zhang et al. [11] proposed a conditional generative adversarial model for SIRR, in which a new refined loss function that combines perceptual loss, Euclidean loss, and adversarial loss is presented. In this paper, we also use a GAN-based method to address the SIRR problem. The architecture of the proposed generator subnetwork is different from the models mentioned above. More specifically, we design a generator with multiscale convolution operators that can

simultaneously focus on the local rain drops and the spatial information of the rainy image.

3. Proposed Method

Our purpose is to learn a nonlinear projection between the input rainy image and the output derained image through constructing a GAN-based deep architecture. The proposed GAN-based network consists of two subnetworks, i.e., generator network and discriminator network. The primary target of generator subnetwork is to remove rain streaks without missing any detail message from the rainy image. The discriminator subnetwork acts as a supervised signal to boost the quality of derained image generated by generator subnetwork. In this section, we discuss the architecture in detail.

3.1. Generative Adversarial Loss. To make the derained image generated by generator subnetwork with high quality, to fool the discriminator subnetwork, and to learn, a good discriminator has enough ability to validate whether the derained image looked real. Given a rainy image I , the optimization function of the GAN can be formulated as

$$\min_G \max_D E_{I \sim P_{data(I)}} [\log(1 - D(I, G(I)))] + E_{I \sim P_{data(I,O)}} [\log D(I, O)], \quad (2)$$

where O is the generated output image, D represents the discriminative subnetwork, and G is the generative subnetwork.

3.2. Generator Network

Architecture. As mentioned above, generator subnetwork aims to learn a mapping function from a rainy image to a derained image. The proposed generator framework is shown in the top of Figure 1. Specifically, we first implement convolution operator on the raw image with the kernel size and feature maps of 7×7 and 64, respectively. After the first convolution layer, we introduce two parallel convolution branches. The kernel size of one convolution branch is set to 3×3 , and the other is set to 5×5 . Both numbers of feature maps of each convolution layer for the two branches are set to 64. The generator subnetwork with multiscale branches has two advantages. Firstly, the small kernel size can capture more local rain information while the larger kernel size acquires more spatial information. Secondly, the multiscale convolution kernel is used to make the generator have a variety of filters, and then the learning process of weights and biases is more diverse, and thus the useful information of the image can be fully and effectively extracted. Both of the two settings are helpful for generating higher quality derained image. After five convolution layers for each branch, the outputs of the last convolution layer are concentrated by a simple addition operator. Besides, three skip connections between the front convolution layers and the several later convolution layers are introduced. As shown above, our network contains several convolution operations, which may seriously damage the details of the raw image. However,

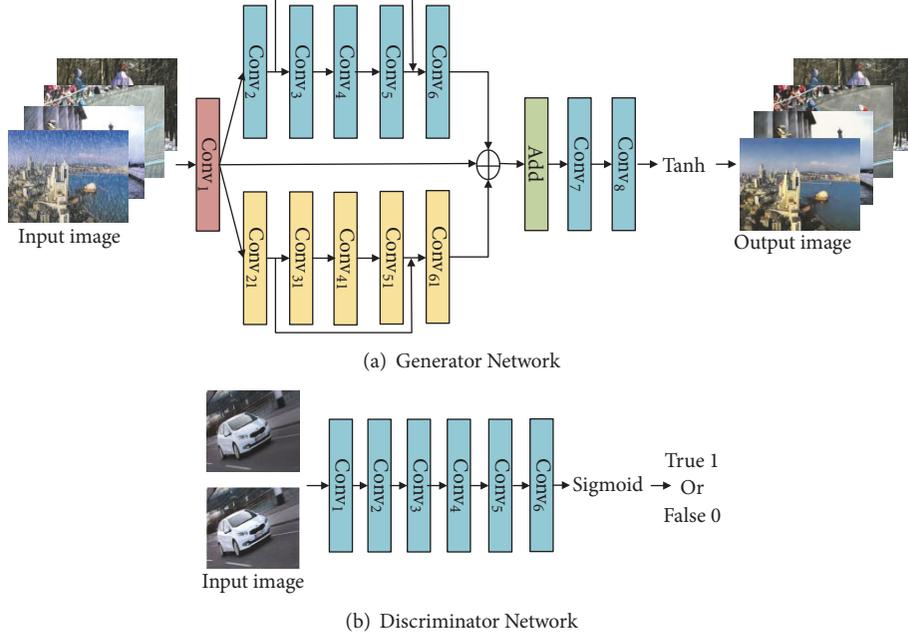


FIGURE 1: The architecture of the proposed DR-Net network. It takes a set of synthesized rainy images as inputs. First, these images are passed through a convolution layer. Then, two parallel convolution branches with five convolution layers are appended. After an additional layer and two convolution layers, the generator subnetwork outputs the derained image. Finally, the discriminator subnetwork uses the mixed derained images and the ground truths as inputs to distinguish if they are fakes or trues. Additionally, three skip connections between the front convolution layers and later convolution layers in generator subnetwork are added.

the feature maps generated by the front convolution layers contain many image details, and integrating these feature maps into the later convolution layers can help the generator to retain the image details. In addition, similar to the deep residual networks [31], the introduction of skip connections is conducive to backpropagate the gradient to the bottom layers, which makes the training phase more stable. Then, the concentrated branches are passed through two convolution layers both with 32 feature maps and 3×3 kernel size. Finally, the output layer is stacked after these convolution operators. In order to maintain the size of the raw image after the convolution operators, we set the padding to 3 pixels for the 7×7 conv layer, 1 pixel for 3×3 conv layers, and 2 pixels for 5×5 conv layers. The detailed parameters of the generative subnetwork layers are shown in Table 1.

Generative Loss Function. As the generator subnetwork aims to generate the derained image as closer as possible to the ground truth, therefore, we adopt the Euclidean loss to supervise the generator. Given a rainy image I , the loss function can be defined as

$$L_e(G) = \frac{1}{CMN} \sum_{c=1}^C \sum_{x=1}^M \sum_{y=1}^N \|G(I) - (R)\|_2^2, \quad (3)$$

where C , M , and N represent the channel, width, and height of the images, respectively. R is the corresponding ground truth of the input image I .

Aside from the Euclidean loss, we also introduce the perceptual loss [32] which can calculate the global difference

between the features of the ground truth and those of the outputs of certain layer. The introduction of perceptual loss is helpful to improve the visual performance of generator subnetwork. The perceptual loss can be written as

$$L_p(G) = \frac{1}{SWH} \sum_{s=1}^S \sum_{w=1}^W \sum_{h=1}^H \|VGG_{16}(G(I)) - VGG_{16}(R)\|_2^2, \quad (4)$$

where S , W , and H represent the channel, width, and height of the output of a certain convolution layer, respectively. VGG_{16} is the VGG-16 model [33]. Following the work of [34], we use VGG-16 model to compute the feature loss at the additional layer.

Based on the two formulations above, we refined the generative loss function as

$$L_G = L_e(G) + \lambda L_p(G) \quad (5)$$

in which λ is the predefined weights for the perceptual loss.

3.3. Discriminator Architecture

Architecture. In our GAN-based network, discriminator architecture is designed for making the derained image synthesized by generator subnetwork much closer to the ground truth. It uses the mixed derained images and ground truths as inputs and classifies if the input is fake or real. Following the work of [11, 33], the convolution operator with

TABLE 1: Layer parameters of the generative sub-network.

| Layer Name | Parameters |
|------------|---|
| Input | Rain image |
| Layer 1 | Conv1. (7,7,64), stride=1, padding=3; Tanh |
| Layer 2 | Conv2. (3,3,64), stride=1, padding=1; Tanh |
| Layer 3 | Conv3. (3,3,64), stride=1, padding=1; Tanh |
| Layer 4 | Conv4. (3,3,64), stride=1, padding=1; Tanh |
| Layer 5 | Conv5. (3,3,64), stride=1, padding=1; Tanh |
| Add layer | Concatenate (layer 1, layer 5) |
| Layer 6 | Conv6. (3,3,64), stride=1, padding=1; Tanh |
| Layer 21 | Conv21. (5,5,64), stride=1, padding=2; Tanh |
| Layer 31 | Conv31. (5,5,64), stride=1, padding=2; Tanh |
| Layer 41 | Conv41. (5,5,64), stride=1, padding=2; Tanh |
| Layer 51 | Conv51. (5,5,64), stride=1, padding=2; Tanh |
| Add layer | Concatenate (layer 21, layer 51) |
| Layer 61 | Conv61. (5,5,64), stride=1, padding=2; Tanh |
| Add layer | Concatenate (layer 1, layer 6, layer 61) |
| Layer 7 | Conv7. (3,3,32), stride=1, padding=1; Tanh |
| Layer 8 | Conv8. (3,3,32), stride=1, padding=1; Tanh |
| Output | Rain-removal image |

PReLU activation and batch normalization is used as a basic unit throughout the whole discriminator subnetwork. The subnetwork contains five convolution layers. The numbers of feature map for each layer are set to 24, 48, 96, 192, and 384, respectively. We set the kernel size of the five convolution layers to 3×3 . After a set of convolution layers, a sigmoid function is attached at the output layer to produce a probability value that indicates the input image as fake or real. The proposed discriminator subnetwork is shown as in the bottom of Figure 1. And the detailed parameters of the discriminator subnetwork layers are shown in Table 2.

Discriminator Loss Function. Since the goal of discriminator subnetwork is to differentiate the synthesized derained image from its corresponding ground truth, we regard it as a binary classification network. Given a mixed N images set, the discriminator loss function can be expressed as

$$L_D = -\frac{1}{N} \sum_{i=1}^N (T_i \log(D(I)) - (1 - T_i) \log(1 - D(I))), \quad (6)$$

TABLE 2: Layer parameters of the discriminator sub-network.

| Layer Name | Parameters |
|------------|--|
| Input | Rain image |
| Layer 1 | Conv1. (3,3,24), stride=2, padding=1; |
| Layer 2 | Conv2. (3, 3, 48), stride=2, padding=1; PReLU |
| Layer 3 | Conv3. (3, 3, 96), stride=2, padding=1; PReLU |
| Layer 4 | Conv4. (3, 3, 192), stride=2, padding=1; PReLU |
| Layer 5 | Conv5. (3, 3, 384), stride=2, padding=1; PReLU |
| Layer 6 | Conv6. (3,3,1), stride=2, padding=1; Sigmoid |
| Output | 1 or 0 |

where T_i is the label of input image I and $T_i=1$ indicates that I is a real ground truth while $T_i=0$ indicates that I is a fake.

4. Experiment and Results

In this section, we first introduce the dataset and evaluation protocols used in this work. Then, the details of the experiments implemented to evaluate the proposed DR-Net model are presented. Finally, some comparison experiments results are discussed.

4.1. Dataset and Evaluation Protocols

Synthetic Dataset. We use the synthesized dataset created by [11] as the training and testing data. The training set of this dataset contains 700 paired images, in which 200 images are selected from BSD training set [35] and the rest of the 500 images are chosen from the UCID dataset [36]. The test set contains a total of 100 paired images, in which 50 images were selected from BSD dataset and the rest of the 50 were chosen from UCID dataset. Besides, we also use the test set created by [9] to evaluate the proposed model.

Real-World Rainy Image. To validate the effectiveness of the proposed DR-Net, we also test it on the real-world rainy image. Specifically, we use the real-world dataset created by [11] and some traffic images downloaded from the Internet to evaluate the performance of our model.

Evaluation Protocols. We adopt the structural similarity index (SSIM) [37] and the visual information fidelity (VIF) [38] to evaluate the performance of our model and the compared state-of-the-art methods as well. The higher SSIM value is, the closer to ground truth the derained image is. For the clean image, the SSIM value is 1. Similarly, higher VIF indicates higher quality of the derained result.

4.2. Training Setting. In this study, we use the torch framework [39] to implement our network. The batch size is

TABLE 3: SSIM results compared with three baseline networks and one extended network on synthesized test images.

| Images | 3×3 Single | 5×5 Single | GEN | DGAN | Ours |
|--------|------------|------------|--------|---------------|---------------|
| a | 0.6461 | 0.7452 | 0.7737 | 0.8592 | 0.8571 |
| b | 0.8213 | 0.7915 | 0.7866 | 0.8476 | 0.8525 |
| c | 0.6345 | 0.6647 | 0.6466 | 0.8859 | 0.8826 |
| d | 0.8007 | 0.7586 | 0.7880 | 0.8112 | 0.8176 |
| e | 0.6331 | 0.6469 | 0.7919 | 0.9196 | 0.9287 |

TABLE 4: SSIM results on synthesized test image.

| Images | Ground truth | Rainy image | DSC [17] | GMM-LP [13] | DrainNet [9] | ID-CGAN [11] | Ours |
|--------|--------------|-------------|----------|-------------|--------------|--------------|---------------|
| f | 1 | 0.7270 | 0.7550 | 0.8067 | 0.8448 | 0.8465 | 0.8604 |
| g | 1 | 0.8569 | 0.8836 | 0.8865 | 0.8995 | 0.8573 | 0.9186 |
| h | 1 | 0.6857 | 0.6673 | 0.7663 | 0.8042 | 0.7720 | 0.8607 |
| i | 1 | 0.7707 | 0.7837 | 0.8532 | 0.8620 | 0.8762 | 0.9101 |
| j | 1 | 0.7093 | 0.7687 | 0.7637 | 0.8009 | 0.8071 | 0.8064 |

set to 9. We use the Adaptive Moment Estimation (Adam) algorithm with the learning rate of 0.0002 to optimize the network. All the training images are resized to 480×480 pixels. The training process converges in roughly 4-5 hours with NVIDIA GTX Taitan-xp GPU.

4.3. Comparison with Baseline Networks. In this section, we evaluate the performance of the proposed DR-Net with the following four baseline architectures:

- (i) 3×3 Single: Single-scale network is trained only using 3×3 convolution branches.
- (ii) 5×5 Single: Single-scale network is trained only using 5×5 convolution branches.
- (iii) GEN: Only generator subnetwork is used, which equates to a traditional CNN architecture.
- (iv) DGAN: The depth and kernel’s numbers of network are increased.

We train these four networks as well as DR-Net on the synthetic training dataset. Table 3 shows the SSIM results of DR-Net compared with the four baseline architectures on synthesized test images. From Table 3, we can observe that the proposed DR-Net achieves the highest SSIM values among the five configurations. Compared with the DR-Net, the generator with single-scale network (i.e., 3×3 or 5×5 convolution branches) achieves the lower SSIM values. When we discard the discriminator subnetwork, the performance of the architecture is also decreased. The DGAN achieves comparable SSIM values with DR-Net. However, increasing the depth and the kernel’s number leads to more running time whether in the training or testing phases. Sample results of the proposed method compared with the four networks on synthesized test images are shown in Figure 2. From Figure 2, it can be seen that the four baseline networks can improve the quality of the rain image but have obvious chromatic aberration and blurred background. The proposed DR-Net achieves the best visual effect in terms of the quality of derained image.

4.4. Comparison with State-of-the-Art Methods. We compare the proposed method with the following representative single-image deraining methods:

DSC: discriminative sparse coding-based method.

GMM-LP: layer prior-based model.

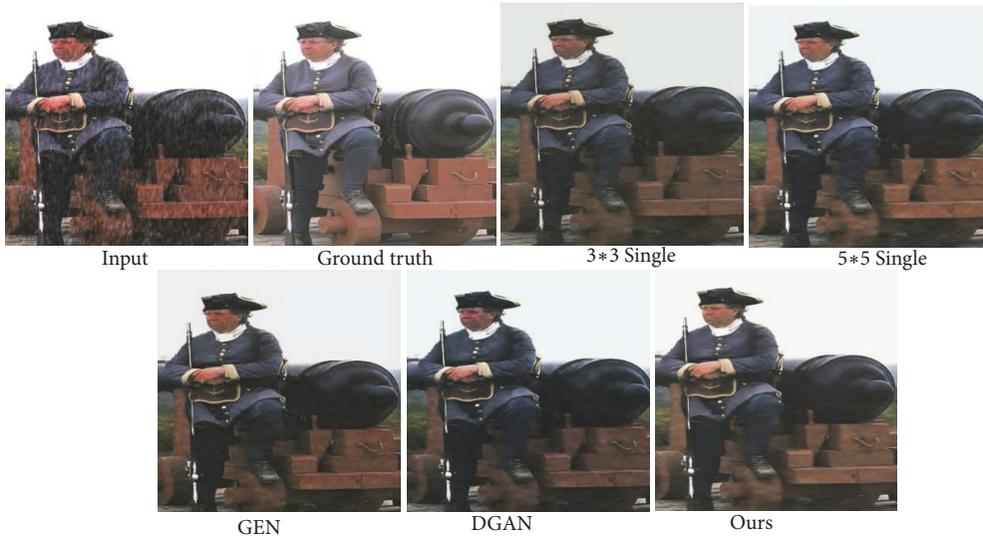
DrainNet: convolution neural network-based method.

ID-CGAN: conditional general adversarial network-based method.

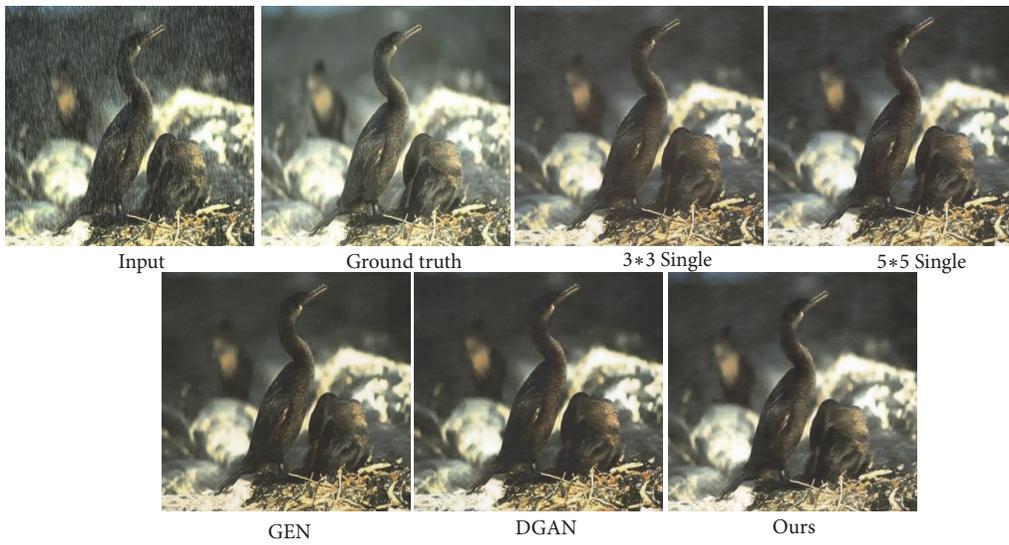
Results on Synthesized Data. In this set of experiments, we implement the comparisons between the proposed model and the four compared methods on the newly synthesized image data. For the ground truths of these test images are known, the structure similarity index (SSIM) and the visual information fidelity (VIF) for quantitative measure can be calculated. From Tables 4 and 5, we can observe that the proposed model achieves the highest SSIM and VIF values.

The visual comparisons for five synthesized images with different intensity and orientations are shown as Figure 3. As can be seen, DSC can remove the partial streaks and reduce the dense degree of rain streaks, but they cannot completely remove the rain streaks. The same situation also takes place in GMM-LP algorithm. Among these four compared algorithms, DrainNet and ID-CGAN get better visual expression. However, compared to the two models, the derained results of the proposed model are better. Moreover, our SSIM and VIF values of five derained images are higher than theirs. Both the experiment results demonstrate the effectiveness of the proposed method.

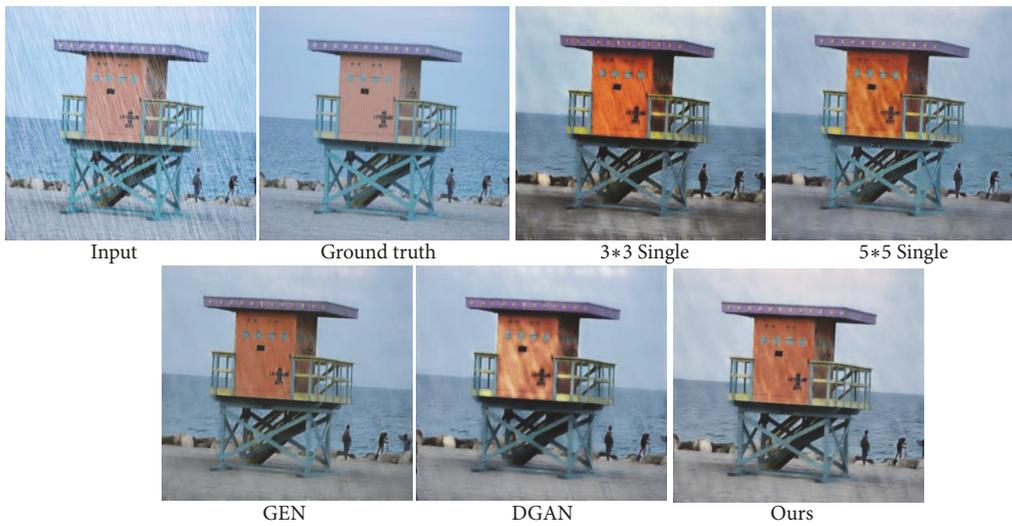
Results on Real-World Data. We also evaluate the proposed model on several real-world rainy images. Figure 4 presents the testing results on four real-world rainy images. From Figure 4, we can observe that both GMM-LP and DSC fail to completely remove rain streaks. Among them, the deraining



(a)



(b)



(c)

FIGURE 2: Continued.

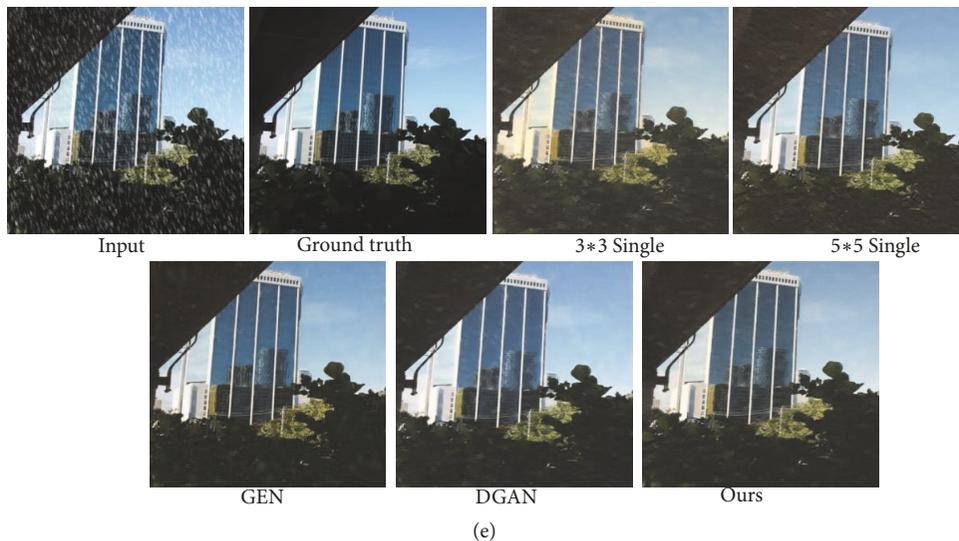
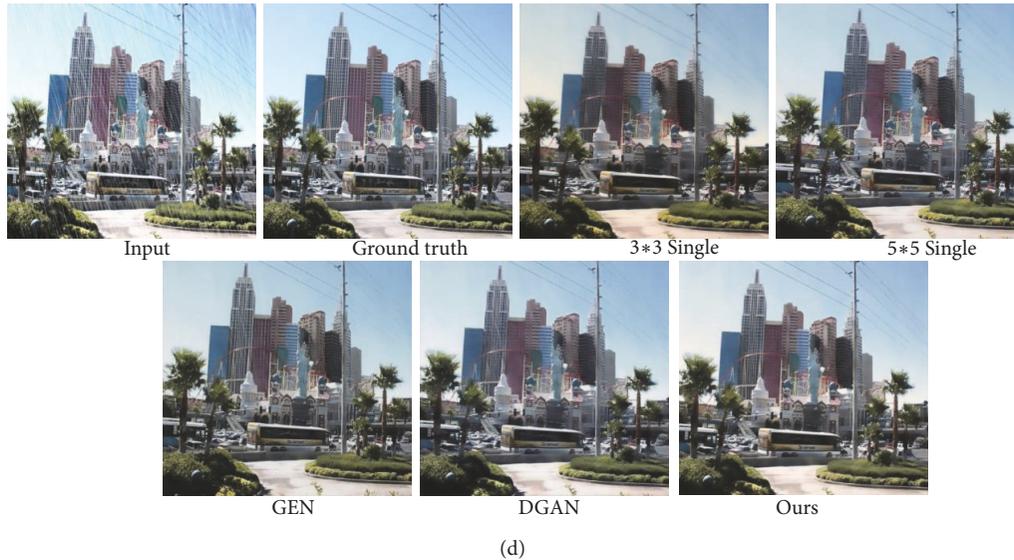


FIGURE 2: Results on five synthesized rainy images.

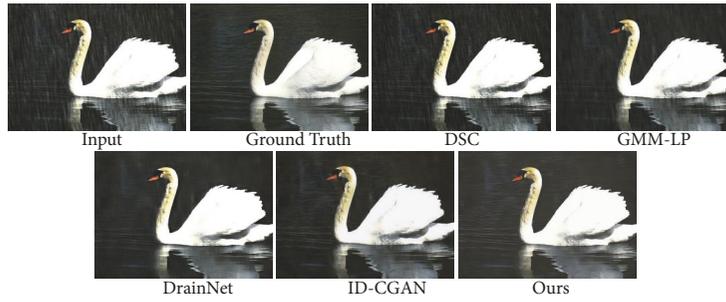
TABLE 5: VIF results on synthesized test image.

| Images | DSC [17] | GMM-LP [13] | DrainNet [9] | ID-CGAN [11] | Ours |
|--------|-------------|----------------|-----------------|-----------------|---------------|
| f | 0.3260 | 0.4015 | 0.3899 | 0.3211 | 0.4750 |
| g | 0.4103 | 0.4389 | 0.4036 | 0.4187 | 0.5058 |
| h | 0.2144 | 0.2819 | 0.2514 | 0.3316 | 0.3760 |
| i | 0.3272 | 0.3909 | 0.3471 | 0.4224 | 0.4580 |
| j | 0.2836 | 0.3093 | 0.3076 | 0.0131 | 0.4577 |

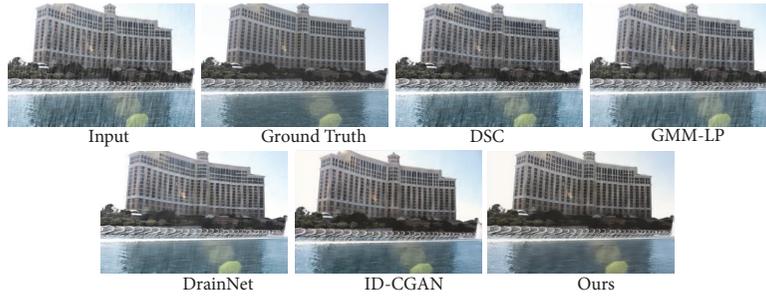
effects of DrainNet and ID-CGAN are on a par with the proposed model from the visual perspective. However, compared with DrainNet and ID-CGAN, our derained results are able to maintain more details of the raw input images. In order to have a better comparison, we show one specific region of interest for each derained result of the five algorithms. From the regions of interests, we can see that, compared with other four algorithms, our model provides the best visual

performance on jointly removing rain streaks and retaining details, which further verifies the validity of the proposed method.

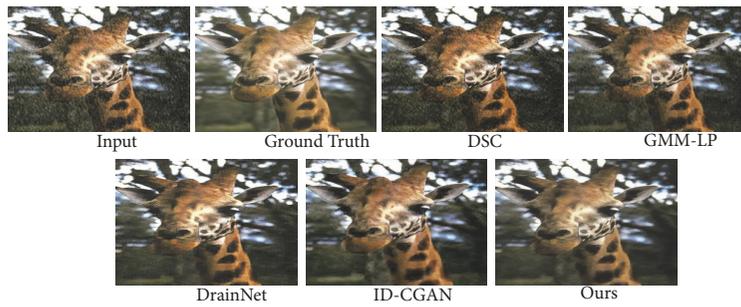
User Study Comparisons. Since there is no ground truth for real-world data, we constructed a user study to supply real feedback and quantify the subjective evaluation of the proposed model. We selected 10 real-world rainy images from



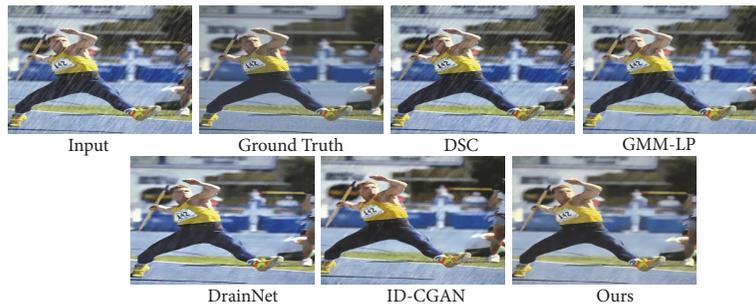
(f)



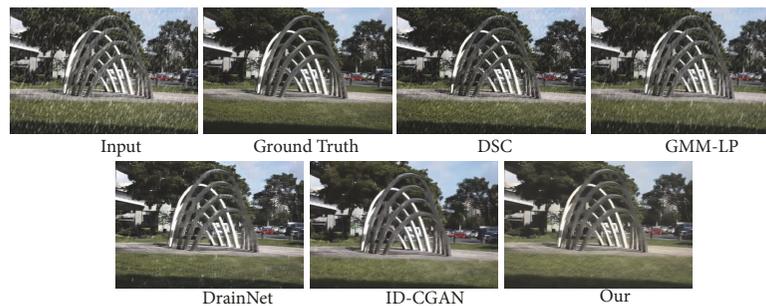
(g)



(h)



(i)



(j)

FIGURE 3: Results on five synthesized rainy images.

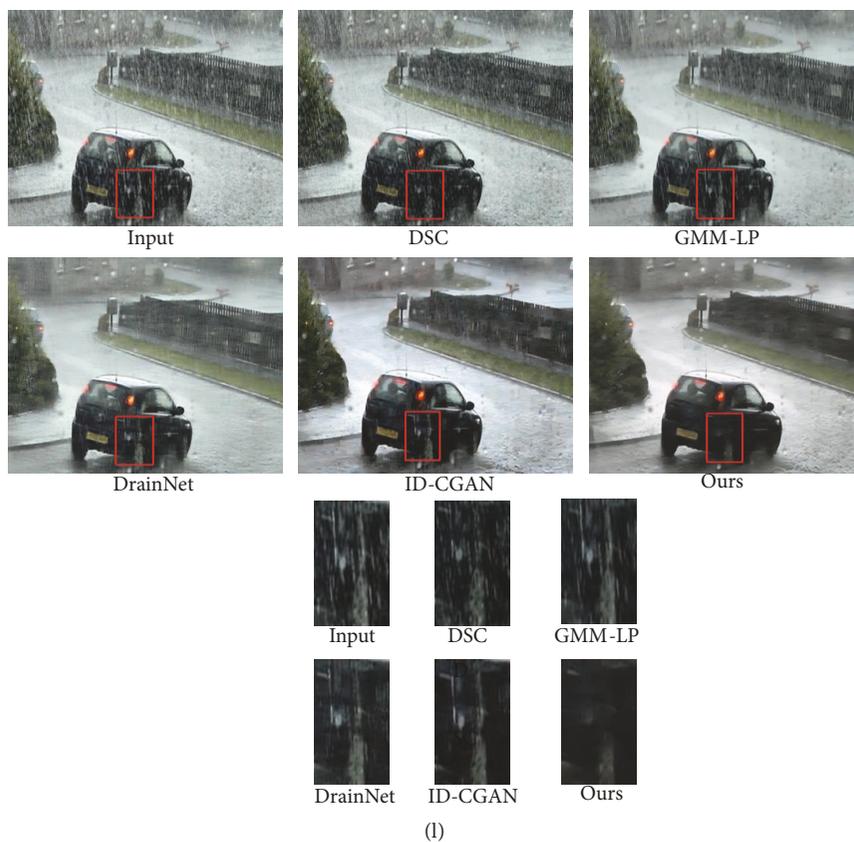
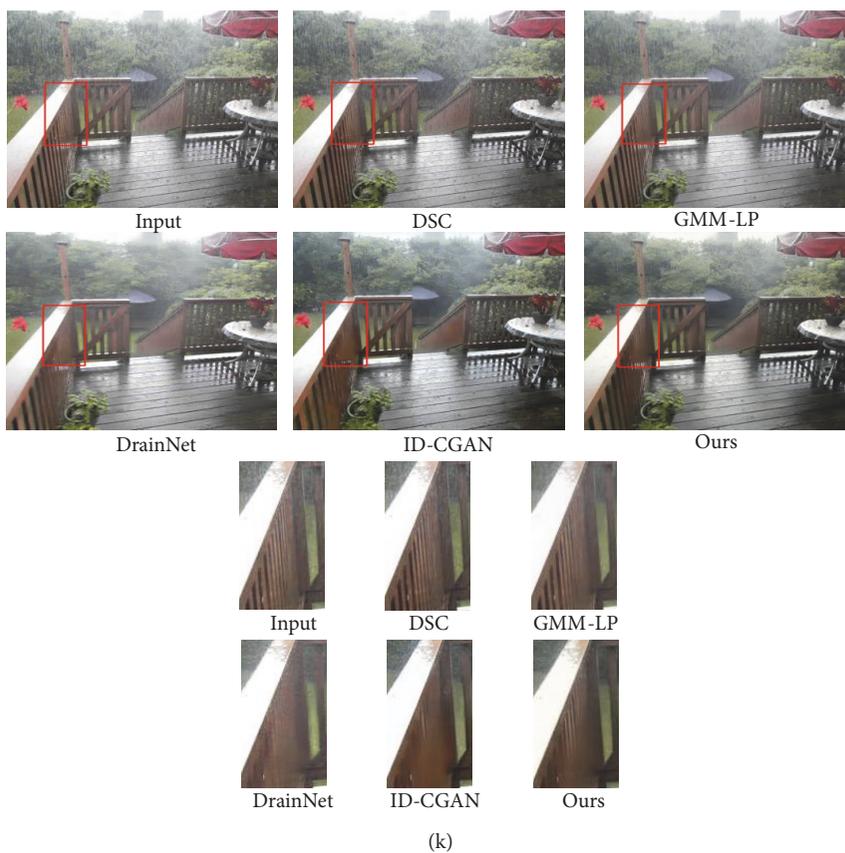
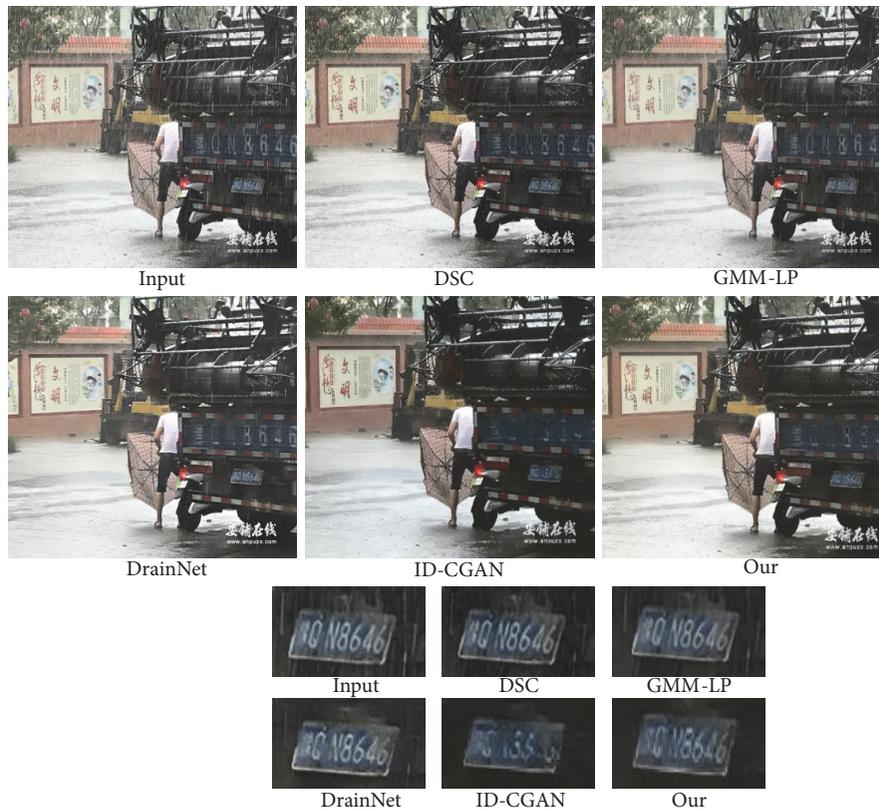


FIGURE 4: Continued.



(m)



(n)

FIGURE 4: Results on real-world rainy images.

TABLE 6: Average scores of user study.

| Images | Input | DSC | GMM-LP | DrainNet | ID-CGAN | ours |
|--------|-------|------|--------|----------|---------|------|
| Scores | 1.34 | 1.87 | 2.06 | 3.55 | 3.98 | 4.07 |

TABLE 7: Running time (in seconds) of DR-GAN compared with State-of-the-art methods.

| | DSC | GMM-LP | DrainNet | ID-CGAN | Ours |
|---------|--------|--------|----------|---------|------|
| 480×480 | 277.8s | 566.1s | 2.1s | 1.2s | 1.4s |

TABLE 8: Parameter number for three deep learning based deraining methods.

| DrainNet | ID-CGAN | ours |
|----------|---------|-------|
| 165888 | 18136 | 22714 |

the real-world dataset created by [11] and some rainy traffic images downloaded from the Internet. Figure 4 shows some derained results. For user study, we first used all methods to generate deraining images and randomly ordered the deraining results. Then, we displayed the ordered results on the screen and asked 10 participants with computer vision expertise to rank the results from 1 to 5, with 1 being the worst and 5 being the best. Table 6 shows the average subjective scores of the five methods on the real-world rainy images. We can see that our model achieves the highest average score among the five methods, which indicates that the proposed method can generate better deraining results on real-world rainy images from the subjective perspective.

Running Time and Parameter Number Comparisons. To estimate the efficiency of the proposed method, we computed the running time of the compared state-of-the-art methods as well as the proposed method. All the evaluations are implemented on the 480×480 rainy image. DSC and GMM-LP are non-deep learning methods that are implemented on CPU according to the provided code. DrainNet, ID-CGAN, and our methods are implemented on GPU. Table 7 presents the comparison results. In general, our multiscale network requires only 1.4 seconds to process a 480×480 rainy image, which is the same as the existing single-scale deep learning methods. Table 8 compares the parameter numbers of DrainNet, ID-CGAN, and the proposed network. Although our generator has two branches, it has less feature maps. It can be seen that the parameter number of our multiscale rain-removal network is slightly more than ID-CGAN, but less than DrainNet.

5. Conclusion

In this study, we have proposed a generative adversarial network-based architecture for single rainy image removal. The presented architecture consists of two subnetworks, *i.e.*, generation and discrimination subnetworks. The generation subnetwork with multiscale convolution operators can capture local rain drops and spatial information of rainy images simultaneously. To reserve more detail of the

raw background of the rainy images and to improve the steadiness of the training process, three skip connections between the front convolution layers and later convolution layers are introduced. Acting as a supervisory signal, the discrimination subnetwork can be helpful in improving the quality of derained images generated by generation model. Experiments on synthetic and real-world images show that the proposed architecture outperforms other state-of-the-art methods. In the future, we will consider how to use the superresolution and attention mechanism to further improve the ability of deraining for the network.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

This work was supported by the National Natural Science Foundation of China [Grant no. 61673222], the Jiangsu Universities Natural Science Research Project [Grant no. 13KJA510001], and the European Union’s Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie [Grant no. 701697].

References

- [1] K. Garg and S. K. Nayar, “Detection and removal of rain from videos,” in *Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR 2004*, pp. 1528–1535, USA, July 2004.
- [2] D. Huang, L. Kang, Y. Wang, and C. Lin, “Self-learning based image decomposition with applications to single image denoising,” *IEEE Transactions on Multimedia*, vol. 16, pp. 83–93, 2014.
- [3] L.-W. Kang, C.-W. Lin, and Y.-H. Fu, “Automatic single-image-based rain streaks removal via image decomposition,” *IEEE Transactions on Image Processing*, vol. 21, no. 4, pp. 1742–1755, 2012.
- [4] K. Garg and S. K. Nayar, “Photorealistic rendering of rain streaks,” *Acm Transactions on Graphics*, vol. 25, pp. 996–1002, 2006.

- [5] J.-H. Kim, J.-Y. Sim, and C.-S. Kim, "Video deraining and desnowing using temporal correlation and low-rank matrix completion," *IEEE Transactions on Image Processing*, vol. 24, no. 9, pp. 2658–2670, 2015.
- [6] Y.-L. Chen and C.-T. Hsu, "A generalized low-rank appearance model for spatio-temporally correlated rain streaks," in *Proceedings of the 14th IEEE International Conference on Computer Vision, ICCV 2013*, pp. 1968–1975, Australia, December 2013.
- [7] S.-H. Sun, S.-P. Fan, and Y.-C. F. Wang, "Exploiting image structural similarity for single image rain removal," in *Proceedings of the IEEE International Conference on Image Processing*, pp. 4482–4486, 2015.
- [8] J.-H. Kim, C. Lee, J.-Y. Sim, and C.-S. Kim, "Single-image deraining using an adaptive nonlocal means filter," in *Proceedings of the 20th IEEE International Conference on Image Processing, ICIP 2013*, pp. 914–917, Australia, September 2013.
- [9] X. Fu, J. Huang, X. Ding, Y. Liao, and J. Paisley, "Clearing the skies: a deep network architecture for single-image rain removal," *IEEE Transactions on Image Processing*, vol. 26, no. 6, pp. 2944–2956, 2017.
- [10] X. Fu, J. Huang, D. Zeng, Y. Huang, X. Ding, and J. Paisley, "Removing rain from single images via a deep detail network," in *Proceedings of the 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017*, pp. 1715–1723, USA, July 2017.
- [11] H. Zhang, V. Sindagi, and V. M. Patel, *Image De-Raining Using a Conditional Generative Adversarial Network*, arXiv preprint, arXiv, 1701.05957, 2017, arXiv:1701.05957.
- [12] R. Qian, R. T. Tan, W. Yang, J. Su, and J. Liu, "Attentive Generative Adversarial Network for Raindrop Removal from a Single Image," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2482–2491, 2018.
- [13] Y. Li, R. T. Tan, X. Guo, J. Lu, and M. S. Brown, "Rain Streak Removal Using Layer Priors," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2736–2744, 2016.
- [14] K. Garg and S. K. Nayar, "Vision and rain," *International Journal of Computer Vision*, vol. 75, no. 1, pp. 3–27, 2007.
- [15] Z. Xiaopeng, L. Hao, Q. Yingyi, K. L. Wee, and K. N. Teck, "Rain removal in video by combining temporal and chromatic properties," in *Proceedings of the IEEE International Conference on Multimedia and Expo, ICME 2006*, pp. 461–464, Canada, July 2006.
- [16] J. Bossu, N. Hautière, and J.-P. Tarel, "Rain or snow detection in image sequences through use of a histogram of orientation of streaks," *International Journal of Computer Vision*, vol. 93, no. 3, pp. 348–367, 2011.
- [17] Y. Luo, Y. Xu, and H. Ji, "Removing Rain from a Single Image Via Discriminative Sparse Coding," in *Proceedings of the IEEE International Conference on Computer Vision*, pp. 3397–3405, 2015.
- [18] H. Zhang, B. Chen, Z. Wang, and H. Liu, "Deep Max-Margin Discriminant Projection," *IEEE Transactions on Cybernetics*, pp. 1–13, 2018.
- [19] H. Zhang, B. Chen, D. Guo, and M. Zhou, "WHAI, "Weibull hybrid autoencoding inference for deep topic modeling," in *Proceedings of the in International Conference on Learning Representations*, 2018.
- [20] Y. Cong, B. Chen, H. Liu, and M. Zhou, "Deep latent Dirichlet allocation with topic-layer-adaptive stochastic gradient Riemannian MCMC," in *Proceedings of the International Conference on Machine Learning*, vol. 70, pp. 864–873, 2017.
- [21] B. Chen, G. Polatkan, G. Sapiro, D. Blei, D. Dunson, and L. Carin, "Deep learning with hierarchical convolutional factor analysis," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 8, pp. 1887–1901, 2013.
- [22] C. J. Schuler, M. Hirsch, S. Harmeling, and B. Schölkopf, "Learning to Deblur," *IEEE Transactions on Pattern Analysis Machine Intelligence*, vol. 38, pp. 1439–1451, 2014.
- [23] K. Zhang, W. Zuo, S. Gu, and L. Zhang, "Learning deep CNN denoiser prior for image restoration," in *Proceedings of the 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017*, pp. 2808–2817, USA, July 2017.
- [24] H. Huang, R. He, Z. Sun, and T. Tan, "Wavelet-SRNet: A Wavelet-Based CNN for Multi-scale Face Super Resolution," in *Proceedings of the 16th IEEE International Conference on Computer Vision, ICCV 2017*, pp. 1698–1706, Italy, October 2017.
- [25] L. Gatys, A. Ecker, and M. Bethge, "A Neural Algorithm of Artistic Style," *Journal of Vision*, vol. 16, no. 12, p. 326, 2016.
- [26] Z. Yan, H. Zhang, B. Wang, S. Paris, and Y. Yu, "Automatic Photo Adjustment Using Deep Neural Networks," *ACM Transactions on Graphics*, vol. 35, no. 2, pp. 1–15, 2016.
- [27] J. Xie, L. Xu, and E. Chen, "Image denoising and inpainting with deep neural networks," in *Proceedings of the Proceedings of the 25th International Conference on Neural Information Processing Systems (NIPS '12)*, vol. 1, pp. 341–349, December 2012.
- [28] W. Yang, R. T. Tan, J. Feng, J. Liu, Z. Guo, and S. Yan, "Joint Rain Detection and Removal Via Iterative Region Dependent Multi-Task Learning," *CoRR*, vol. abs/1609.07769, 2016.
- [29] X. Mao, Q. Li, H. Xie, R. Y. K. Lau, Z. Wang, and S. P. Smolley, "Least Squares Generative Adversarial Networks," in *Proceedings of the 16th IEEE International Conference on Computer Vision, ICCV 2017*, pp. 2813–2821, Italy, October 2017.
- [30] M. Arjovsky, S. Chintala, and L. Bottou, "Wasserstein gan," <https://arxiv.org/abs/1701.07875>.
- [31] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016*, pp. 770–778, July 2016.
- [32] J. Johnson, A. Alahi, and L. Fei-Fei, "Perceptual Losses for Real-Time Style Transfer and Super-Resolution," in *Computer Vision – ECCV 2016*, vol. 9906 of *Lecture Notes in Computer Science*, pp. 694–711, Springer International Publishing, Cham, 2016.
- [33] A. Radford, L. Metz, and S. Chintala, "Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks," <https://arxiv.org/abs/1511.06434>.
- [34] K. Simonyan and A. Zisserman, "Very Deep Convolutional Networks for Large-Scale Image Recognition," <https://arxiv.org/abs/1409.1556>.
- [35] P. Arbeláez, M. Maire, C. Fowlkes, and J. Malik, "Contour detection and hierarchical image segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 5, pp. 898–916, 2011.
- [36] G. Schaefer and M. Stich, "UCID - An uncompressed colour image database," in *Proceedings of the Storage and Retrieval Methods and Applications for Multimedia 2004*, pp. 472–480, USA, January 2004.
- [37] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: from error visibility to structural similarity," *IEEE Transactions on Image Processing*, vol. 13, no. 4, pp. 600–612, 2004.
- [38] Z. Wang and A. C. Bovik, "A universal image quality index," *IEEE Signal Processing Letters*, vol. 9, no. 3, pp. 81–84, 2002.

- [39] R. Collobert, K. Kavukcuoglu, and C. Farabet, *Torch7: A Matlab-Like Environment for Machine Learning*, NIPS Workshop, BigLearn, 2012.



Hindawi

Submit your manuscripts at
www.hindawi.com

