Received: 28 February 2021 | Revised: 30 May 2021 | Accepted: 15 June 2021

IET Image Processing

DOI: 10.1049/ipr2.12301

The Institution of Engineering and Technology WILEY

ORIGINAL RESEARCH PAPER

An unsupervised generative adversarial network for single image deraining

Zhiying Song[1] | Yuting Guo[2] | Zifan Ma[3] | Ruocong Tang[4] | Linfeng Liu[4]

[1] School of Communication and Information Engineering, Nanjing University of Posts and Telecommunications, Nanjing, China

[2] School of Science, Nanjing University of Posts and Telecommunications, Nanjing, China

[3] School of Electronic and Optical Engineering, Nanjing University of Posts and Telecommunications, Nanjing, China

[4] School of Computer Science and Technology, Nanjing University of Posts and Telecommunications, Nanjing, China

Correspondence
Linfeng Liu, School of Computer Science and Technology, Nanjing University of Posts and Telecommunications, Nanjing 210023, China.
Email: liulf@njupt.edu.cn

Funding information
National Natural Science Foundation of China, Grant/Award Numbers: 61872191, 61872193; Six Talents Peak Project of Jiangsu Province under Grant, Grant/Award Number: 2019-XYDXX-247

Abstract

As the basis of image processing, single image deraining has always been a significant and challenging issue. Due to the lack of real rainy images and corresponding clean images, most deraining networks are trained by synthetic datasets, which makes the output images unsatisfactory in real applications. Besides, note that a heavy rainfall is typically accompanied with some fog. Although some deraining networks have been proposed to remove the rain streaks in the rainy images, the output images may still be blurred due to the accompanied fog. In this paper, these problems existing in single image deraining is comprehensively considered, and propose a Cycle-Derain network based on an unsupervised attention-guided mechanism. Specifically, the Cycle-Derain network takes advantage of generative adversarial networks with two mappings and the cycle consistency loss to train both unpaired rainy images and rain-free images. Moreover, it introduces an unsupervised attention-guided mechanism and exploits the loop-search positioning algorithm to deal with the details of rain and fog in images. Extensive experiments have been carried out, and the results show that the proposed Cycle-Derain network is preferable compared with other deraining networks, especially in term of rainy image restoration.

1 | INTRODUCTION

The images captured in rainy circumstances generally undergo some degradations, due to the impacts of rain streaks and rain drops, which affects the subsequent processing of images significantly. Therefore, single image deraining (SID) remains a vital issue. As an example, Figure 1 indicates that the rain-free images are much clearer than the rainy images, and more valuable information can be obtained from the rain-free images.

Most SID approaches typically divide a rainy image into two layers: rain layer and background layer, and then derain the images through exploring the mapping relations between rainy images and their rain layer. However, considering the fact that the heavy rain can also produce some fog in real environments [1, 2], and thus a rainy image should be taken as the combination of rain layer, fog layer and background layer, i.e. we have that:

$$Z = B + R + F, \tag{1}$$

where Z denotes a rainy image, B denotes the background layer, R denotes the rain layer and F denotes the fog layer. An example of hierarchical diagram of images is provided in Figure 2. In this paper, we will extract the rain layer from rainy images and separate the fog layer by an unsupervised attention-guided mechanism.

To remove the rain streaks and restore the details of a clear background, we focus on the SID technique based on an attention-guided mechanism of GAN (generative adversarial network) [3–5]. Especially, a Nash equilibrium is expected to be achieved for obtaining the high-quality image outputs. Most existing approaches train numerous paired images of synthetic datasets in a supervised manner, and the number of paired images is always insufficient. To this end, we propose a loop-search positioning algorithm based on CycleGAN [6] and attention-guided mechanism, which can train the unpaired images rather than paired images and output preferable images. Thus, this algorithm can reduce the computation complexity of

Input: s denotes the image of fog after deraining, T denotes the target domain, K denotes the number of epoch, S denotes the target domain of X_s, T denotes the target domain of X_T, α denotes learning rate;
Output: mapped image s';

for $m = 0$ to $K - 1$ **do**
  for $i = 0$ to $|X_s| - 1$ **do**
    a data point s from X_s is instantiated, a data point t from X_T is instantiated;
    if $m<K$ and $\lim_{n*n} s_a = 0$ **then**
      s_f is calculated according to formula (8) ;
      formula (11) and formula (12) are used for iterative updating parameters of $G_{s\text{-}t}$, $D_{s\text{-}t}$ and A_s ;
    end if
  end for
end for

FIGURE 1 Real rainy images vs. images after deraining

removing the fog layer, and obtain a clear background. Our contributions are summarised as follows:

(i) An unsupervised network termed Cycle-Derain is constructed, to realise the translation from source domain to target domain, and then back to source domain. The overall correspondence of these images is achieved through restricting the cycle consistency loss.
(ii) An attention-guided mechanism is adopted in an unsupervised way, and the pixels are endowed with different weights to carry out an element-wise product on each RGB channel. Thus, the fog layer in rainy images can be positioned, and then the rainy images can be defogged.
(iii) A loop-search positioning algorithm is designed to deal with the background and the foreground, respectively, and the removal judgements of fog layer are iteratively implemented. Especially, when the fog layer has not been removed, the foreground will be substituted into the loop. Note that the loop-search positioning algorithm can reduce the computation complexity and avoid the redundant processing of fog-free areas in images.

The rest of this paper is organised as follows: Section 2 gives some related works. Section 3 introduces the network model and details of Cycle-Derain. Section 4 reports the performance evaluation and simulation results. Finally, Section 5 concludes this paper.

2 | RELATED WORKS

2.1 | GAN and CycleGAN

Goodfellow et al. [1] propose a deep learning network named generative adversarial framework. This framework can generate more competitive images and achieve a dynamic balance between generations and discriminations. It is observed that GAN based methods are promising in image processing tasks. Recently, there are a few methods dealing with rain streaks proposed on the basis of GAN. For instance, Qian R et al. [7] introduce an attentive GAN and inject the attention map into the generative and discriminative network. The network consists of an attentive recurrent network and a contextual auto-encoder, which can handle the relatively severe presence of raindrops and the visibility degradation.

Wang Y et al. [8] propose a novel deraining framework called GRASPP-GAN. GRASPP-GAN contains a modified ResNet-18 [9] which exploits some deep features of rainy images, a revised ASPP [10, 11] structure, and a multi-scale feature fusion model. The generator and discriminator are trained simultaneously to enhance the deraining performance and visual quality. Moreover, three losses are utilised in adversarial training: 2-norm loss, gradient information guided loss, and adversarial loss. Likewise, Li R et al.[12] put forward a 2-stage network including a physics-based subnetwork and a model-free refinement subnetwork. The physics-based subnetwork decomposes the input images into high frequency components and low frequency components, respectively. The refinement subnetwork contains a conditional GAN. In [13], a GAN based deraining network is introduced and trained with mixture of two rain image composite model which can create synthesize complex rainy images. The generator and discriminator can learn the relationship between rainy images and residual images.

Although GAN is effective in generating more realistic images [4], aforementioned training of generative adversarial networks require large amounts of paired images, which are hard to be obtained. Zhu et al. [6] propose CycleGAN to deal with the difficulty of style translation among unpaired images. CycleGAN introduces the cycle consistency loss on the basis of GAN counter loss, and realises the style transformation on unpaired images through two-way transformation between source domain and target domain bilaterally. Specially, there are two branches, one is translation from the source domain X to the target domain Y and back to the source domain X, and the other is translation from the target domain Y to the source domain X and back to the target domain Y. Meanwhile, there are two constraints including the interaction of the generator and the discriminator along with the cycle consistency.[6] Moreover, the methods based on CycleGAN are widely applied in various visual tasks, such as image deblurring [14], image super-resolution [15]. The experimental results show that the CycleGAN network yield good results in image translation tasks involving the changes of colour and texture, while it could make some mistakes in the geometric changes.

2.2 | Single image deraining

There are seldom effective works for detecting and removing rain streaks from the rainy images, and it is difficult to remove rain streaks in a single image than those in a video [16]. Hence, the issue of single image deraining draws the research attention.

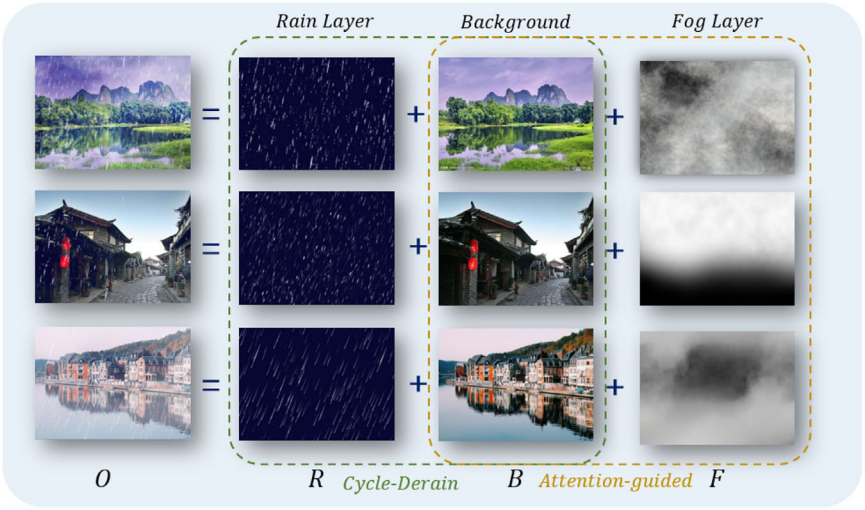


FIGURE 2 A hierarchical diagram of rainy images

For example, Luo et al. [17] propose a single image deraining method based on discriminative sparse coding. This method uses non-linear screen blend model for rainy images which are regarded as the synthesis of the rain layer and background layer. Also, a discriminative sparse coding is presented. Under the dictionary of mutually exclusive overlearning [18], the rain removal process is regularised, and thus the local patches of rain layer and background layer can be used for sparse modelling in the learning dictionary. The sparse coding learned from the dictionary has a clear distinction between the background layer and the rain layer. Besides, a variational model is used to monitor and remove the rain streaks in the input images. However, this method cannot completely solve the fuzziness in a low channel, and the rain streaks cannot be separated effectively especially when the image background is similar to the rain.

Reference [4] presents the DerainNet for single image deraining. Through a convolutional neural network, the mapping relationship between the rainy image and the clear image detail layer can be explored, and the image processing domain technique is employed to modify the objective function and improve the deraining performance. However, the generalisation ability of this method is weak in handling real rainy images.

Recently, some remarkable achievements have been made in the field of single image deraining by some machine learning methods. For example, in the aspect of full supervised learning, Yang et al. [19] propose a new rain map model including fog effect and multi-layer rain streaks to detect additional rain. Moreover, based on this rain model, a multi-task deep learning architecture is developed to detect and remove rain with full convolutional network. Ren et al. [6, 20] propose a learning network which uses a simple combination of progressive ResNet (PRN) and multi-stage recursion to exhibit a good performance of single image deraining. Furthermore, a recurrent layer is introduced to exploit the dependencies of deep features across stages to form an a progressive recurrent network (PReNet). The combination of PRN and PReNet can reduce the number of network parameters, and the network can be trained by a single loss function (SSIM or MSE), making the rain removal process simpler and more effective. However, due to the fact that the training datasets include the rainless images and the synthetic rain images, the generalisation ability in the real scenes is always weak. Wei [3] et al. add the real rainy images into the training datasets with a semi-supervised method, and consider the residual difference between the rainy images and the rain-free images. The proposed network can adapt to the real unsupervised rain models through the supervised synthesis of rain streaks. Wei et al. [4] design an unsupervised attention-guided rain streak extractor (U-ARSE), which takes the attention-guided mechanism for the spatial domain of both rainy images and rain-free images, and they combine the cycle consistency loss with CycleGAN to complete the single image deraining.

2.3 | Motivation of our work

In summary, due to the lack of paired images, the image deraining generator trained based on the synthetic datasets has a weak generalisation ability in real scenes. Besides, the deraining performance of current works is not satisfactory, or the deraining process is with low efficiency. To this end, this paper will propose an unsupervised rain removal network termed Cycle-Derain, which exploits the cycle consistency loss to train the unpaired images. Especially, an attention-guided mechanism and the convolution series structure are taken to optimise the fog generated by rain.

3 | CYCLE-DERAIN MODEL

The original supervised attention-guided deraining model learns the mapping relations by paired examples [21]. However, in the deraining tasks of real applications, the training datasets of paired images are not available. Thus, we explore an unsupervised Cycle-Derain model by unpaired images to produce ideal output images. Furthermore, the Cycle-Derain network can strengthen the defogging effect and provide a better image-to-image translation.

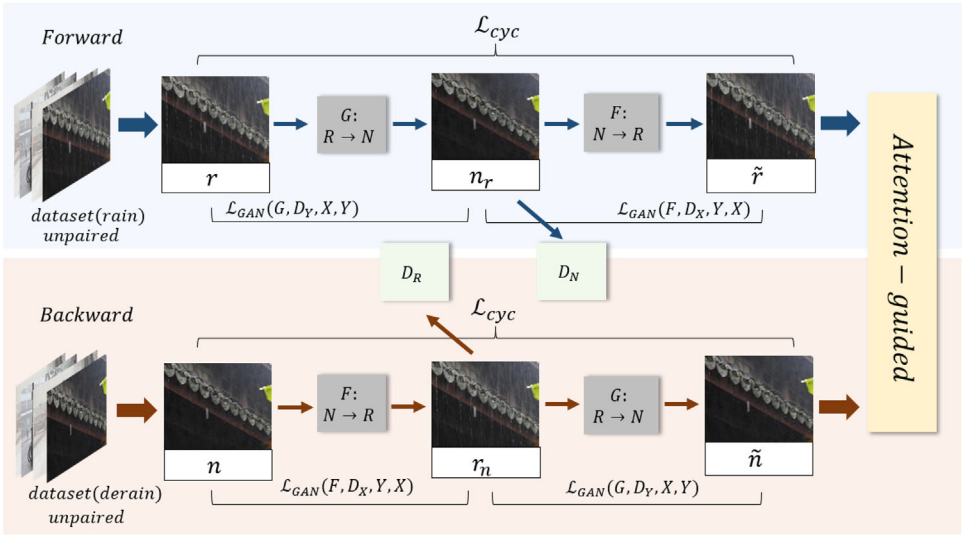


FIGURE 3 Network architecture of Cycle-Derain

The network architecture of Cycle-Derain is shown in Figure 3. The network can be divided into a forward module and a backward module: (i) With regard to the forward branches, the cycle-consistency from rainy to rainy is expressed as $r \to n_r \to \widetilde{r}$, where the input rainy image r is used to generate the rain-free image n_r, and then the reconstructed rainy image $\widetilde{r}$ is obtained by a generator. (ii) With regard to the backward module, the cycle-consistency from rain-free to rain-free is expressed as $n \to r_n \to \widetilde{n}$, where the input rain-free image n is used to generate the rainy image r_n, and then the reconstructed rainy image $\widetilde{n}$ is obtained by a generator.

In addition, we introduce an attention network to extract the fog from rainy images. Besides, three generators G_{Gr}, $G_{\widetilde{r}}$ and G_{s-t} can generate the rain-free images, rainy images and fog-free images, respectively. Three discriminators D_n, D_r and D_s are introduced to distinguish real-world images from translated images that are obtained by the aforementioned generators. This process can be expressed as:

$$m_r = \text{Cycle-Derain}(r), \tag{2}$$

where m_r denotes the output image processed by Cycle-Derain.

3.1 | CycleGAN

3.1.1 | Generators

We introduce two new generators G_{Gn}, G_{s-t} besides the aforementioned generators G_{Gr}, $G_{\widetilde{r}}$, $G_{\widetilde{n}}$. G_{Gr} takes the rainy image r to generate the rain-free image n_r, $G_{\widetilde{r}}$ takes the rain-free image n_r to generate the rainy image $\widetilde{r}$, $G_{\widetilde{n}}$ takes the rain-free image n to generate the rainy image r_n, and G_{Gn} takes the rainy image r_n to generate the rain-free image $\widetilde{n}$. r and $\widetilde{r}$, n and $\widetilde{n}$ must satisfy the cycle consistency, i.e. we have the following formulas:

$$\begin{cases} r \to G_{Gr}(r) \to G_{\widetilde{r}}(G_{Gr}(r)) \approx r, \\ n \to G_{\widetilde{r}}(n) \to G_{Gn}(G_{\widetilde{n}}(n)) \approx n. \end{cases} \tag{3}$$

Especially, G_{s-t} takes the images in source domain S to generate the images in target domain T. The translation is defined by:

$$\begin{cases} n_r = G_{Gr}(r), \\ \widetilde{r} = G_{\widetilde{r}}(n_r), \\ r_n = G_{\widetilde{n}}(n), \\ \widetilde{n} = G_{Gn}(r_n), \\ t = G_{s-t}(s). \end{cases} \tag{4}$$

3.1.2 | Discriminators

Likewise, we also introduce two new discriminators D_{Gn}, D_{s-t} besides the aforementioned generators D_{Gr}, $D_{\widetilde{r}}$, $D_{\widetilde{n}}$, D_{Gr}. $D_{\widetilde{n}}$ and D_{s-t} are referred to as type-I discriminators which can be utilised to distinguish rainy images, rain-free images and fog-free images from real-world images, respectively. The form of adversarial losses $\min_G \max_{D_Y} L_{\text{GAN}}(G, D_Y, X, Y)$, $\min_F \max_{D_X} L_{\text{GAN}}(F, D_X, Y, X)$ and $\min_G \max_{D_X} L_{\text{GAN}}(G, D_S, S, T)$ is expressed as:

$$\begin{aligned} L_{\text{GAN}}(G, D_Y, X, Y) = {} & E_{y \backsim p_{\text{data}}(y)}[\log D_Y(y)] \\ & + E_{x \backsim p_{\text{data}}(x)}[\log(1 - D_Y(G(x)))], \end{aligned} \tag{5}$$

where x and y denote the training samples given by the two domains X and Y, and we assume the data contributions $x \backsim p_{\text{data}}(x), y \backsim p_{\text{data}}(y)$. G attempts to make the generated images $G(x)$ similar to the images in domain Y. D distinguishes the generated images and the images in the Y domain as much as possible. Therefore, G can minimise the objective while D attempts to maximise it.

However, the adversarial losses are highly constrained, because the network is possible to map the same images to any random images in the target domain, which implies that the generated images probably do not match the target images. Therefore, we introduce type-II discriminators D_{Fr} and D_{Gn} to guarantee that the learned mapping functions are cycle-consistent, i.e. with regard to each input image x, the translation network should be able to bring it back to the original image. The difference between input images and output images is measured by a cycle consistency loss:

$$L_{\text{cyc}}(G,F) = E_{x\backsim p_{\text{data}}(x)}[||F(G(x))-x||] + E_{y\backsim p_{\text{data}}(y)}[||G(F(y))y||]. \tag{6}$$

Cycle-Derain includes two mappings: $G(F(y)) : R \to N$ and $F(G(x)) : N \to R$. In Equation (6), $F(G(x))$ and $G(F(y))$ denote the reconstructed images. R denotes the rain domain, and N denotes the rain-free domain. Hence, the forward cycle consistent loss and the backward cycle consistent loss are applied in Cycle-Derain:

$$\begin{cases} r \to G(r) \to F(G(r)) \approx r, \\ n \to F(n) \to G(F(n)) \approx n. \end{cases} \tag{7}$$

In addition, the heavy rain inevitably brings some fog which causes an adverse effect on the visual quality of rainy images. However, neither type-I discriminator nor type-II discriminator can distinguish the fog domain. Therefore, an attention network is required to locate and process the fog layers of images.

3.2 | Attention-guided mechanism

The images captured in heavy rain always include some fog layers, and thus the adverse effect must be considered. In Cycle-Derain, an unsupervised attention mechanism without alerting the background [4, 22] has been taken to address this issue.

Similar to CycleGAN, the attention mechanism in Cycle-Derain can also exploit the unpaired images for training, and the numbers of foggy images and non-foggy images are probably different, which greatly simplifies the dataset construction.

The attention network is denoted by A_s, which selects remaining fog area to be translated. The attention map s_a is induced from S and reflects the weight of each pixel. After feeding the input rain-free image to the generator G_{s-t}, the element-wise product is operated between s_a and $G_{s-t}(s)$, and then the preliminary defogging layer s_f can be obtained by:

$$s_f = s_a \odot G_{s-t}(s). \tag{8}$$

The background image s_b is expressed as:

$$s_b = (1 - s_a) \odot s. \tag{9}$$

By adding the masked output of G_{s-t} into the background, the defogged image is expressed as:

$$s' = s_f + s_b = s_a \odot G_{s-t}(s) + (1 - s_a) \odot G_{s-t}(s). \tag{10}$$

As shown in Figure 4, the rain-fog features can be extracted by two convolution layers, sigmoid function and LeakyReLu activation function.

After positioning the fog layer, we obtain s_a and apply a loop-search positioning algorithm to remove the fog layer. Considering that the learning efficiency of the generator and the discriminator is typically limited, and the training time will be extremely long. Thus, we must extract clear background from the input images. The flow chart of the loop-search positioning algorithm is illustrated in Figure 4. When the attention map s_a converges towards zero, the pixels in weight graph of images after processing cannot be added up to zero, due to the limited capacity of generators and discriminators.

When the proportion of the foggy parts where the pixel equal to 1 is small enough, indicating that the fog layer has been removed, and the image s' can be output; otherwise, s is updated by s_f, i.e. s_f is input into the attention network, and the above stages will be repeated until the attention map s_a approximates zero. Specially, the previous background image after being weighted will not be input into the cycle, and it will be added to the defogged part after obtaining a clear background layer. This mechanism can improve the efficiency of generators and discriminators, and avoid the clarity reduction of the non-foggy part. We incent this behaviour by a $L(s_a)$ loss:

$$L(s_a) = \min||s_a - 0||^2, \tag{11}$$

where min denotes the minimising operation of $||s_a||^2$. In addition, we involve the discriminator D_s, and obtain the adversarial loss to make the final image be more realistic:

$$L_{\text{GAN}}(G,D_s,S,T) = E_{t\backsim PT(t)}[logD_T(t)] + E_{s\backsim PS(s)}[log(1 - D_T(G(s')))]. \tag{12}$$

The loss function of Cycle-Derain is constituted by combining adversarial, cycle-consistency and $L(s_a)$ losses for both source domain and target domain:

$$\begin{aligned} &L(G,F,D_X,D_Y,D_S,s_a) \\ &= L_{\text{GAN}}(G,D_X,S,T) + L_{\text{GAN}}(G,D_Y,S,T) \\ &\quad + L_{\text{GAN}}(G,D_S,S,T) + \lambda_1 L_{\text{cyc}}(G,F) + \lambda_2 L(s_a). \end{aligned} \tag{13}$$

Through solving the MiniMax optimisation problem, we can obtain the optimal parameter settings of $L(\cdot)$ by the following formula:

$$G^*,F^*,D_X^*,D_Y^*,D_S^* = \arg\min_{G,F,s_a}\left\{\arg\min_{D_X,D_Y,D_S} L(G,F,D_X,D_Y,D_S,s_a)\right\}. \tag{14}$$

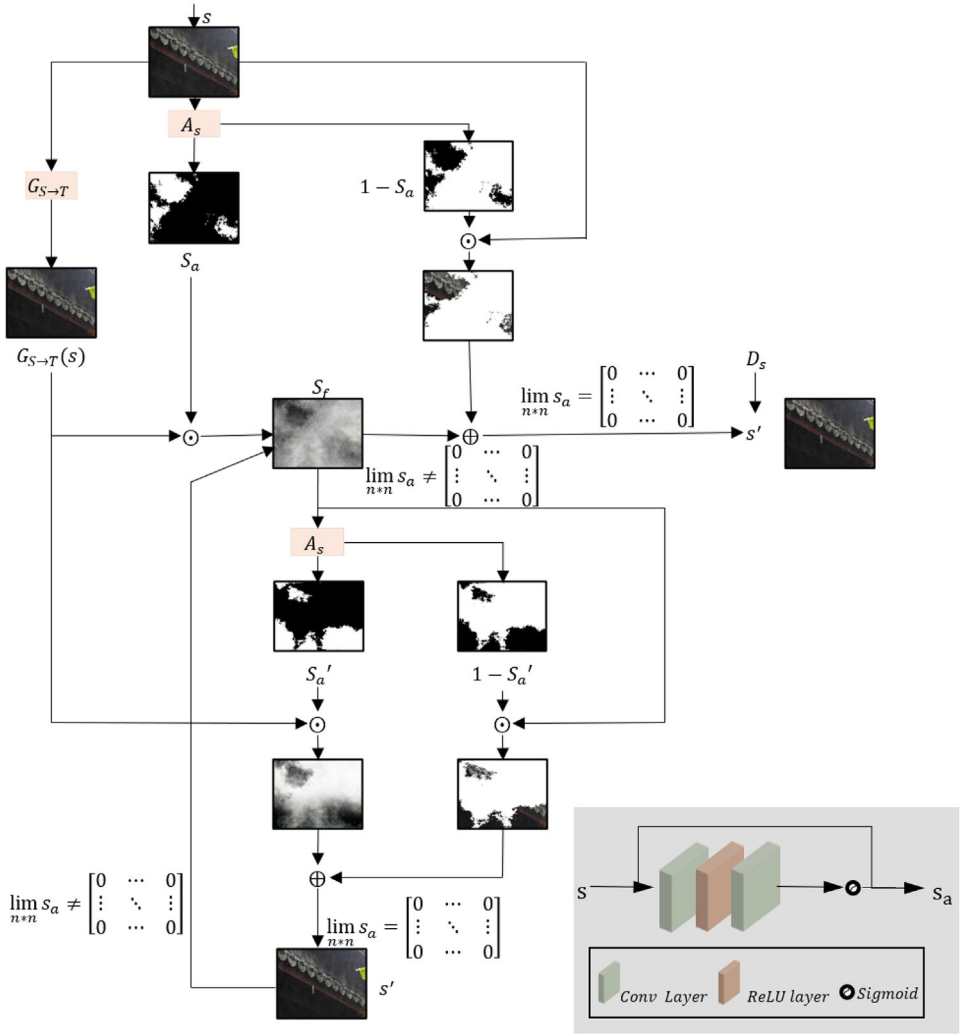


FIGURE 4 Network model with an attention-guided mechanism

The above details describe the network architecture of the loop-search positioning algorithm that we introduce into the unsupervised attention-guided mechanism. Moreover, Algorithm 1 summarises the main procedure of the loop-search positioning algorithm.

3.3 | Complexity analysis

Different from other related algorithms, there are two main factors that affect the running time of Cycle-Derain: (a) the learning rate of the generators and discriminators; (b) the computation complexity. The learning rate determines when the objective function will converge to the local minimum value, and it is exponentially decreased with the increase of the number of training epochs. Therefore, the computation complexity is mainly applied to measure the running time.

Suppose m denotes the number of images, in the best case the fog and rain in each image has been removed after finishing

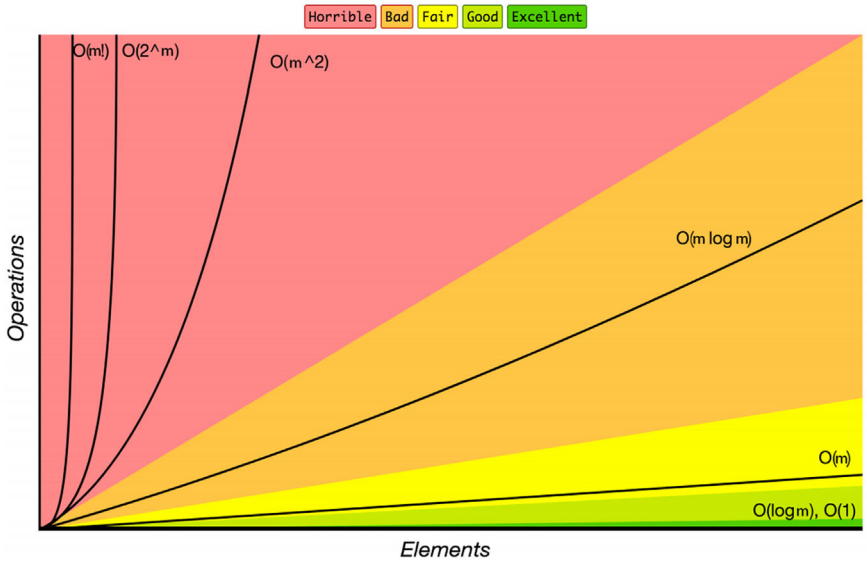
ALGORITHM 1 The attention-guided mechanism is added to the loop-search positioning algorithm for defogging

Input: s denotes the image of fog after deraining, T denotes the target domain, K denotes the number of epoch, S denotes the target domain of X_s, T denotes the target domain of X_T, α denotes learning rate;

Output: mapped image s';

for $m = 0$ to $K - 1$ **do**
  for $i = 0$ to $|X_s| - 1$ **do**
    a data point s from X_s is instantiated, a data point t from X_T is instantiated;
    if $m < K$ and $\lim_{n*n} s_a = 0$ **then**
      s_f is calculated according to formula (8) ;
      formula (11) and formula (12) are used for iterative updating parameters of G_{s-t}, D_{s-t} and A_s ;
    end if
  end for
end for

FIGURE 5 Computation complexity



a CycleGAN process. Each step will be executed for each image, and the computation complexity reaches $\mathcal{O}(m)$. However, in the worst case, each image is unable to finish a CycleGAN process completely, and the attention mechanism repeats the operations of loop-search positioning algorithm several times. The processing time needed for each image is linearly related to the number of repetitions. Thus, the largest computation complexity is equal to $\mathcal{O}(m^2)$, and the average computation complexity is equal to $\mathcal{O}(m)$.

As shown in Figure 5, the computation complexity of Cycle-Derain remains an acceptable level, which indicates that Cycle-Derain can achieve an ideal deraining and defogging effect within a limited running time.

4 | EXPERIMENTS AND DISCUSSIONS

The learning network architecture has adopted our proposed Cycle-Derain model. The datasets are trained unsupervisedly by a Pytorch framework, with a NVIDIA GeForce GTX 2080 Ti GPU. Especially, the trained datasets include rainy-foggy images and background images with different styles and captured in different environments. The original images are first filtered, and then the patch-level discriminator architecture enables Cycle-Derain to work on images with random sizes and resolutions in a fully convolution.

4.1 | Datasets

Besides collecting considerable rainy images and rain-free images, we construct a rainy-foggy image dataset termed Cycle-Derain dataset through adding the fog layers into some rainy images. The Cycle-Derain dataset includes many objects with different backgrounds and different sizes, making the transformations between images become more comprehensive. Firstly, we input paired images in the beginning to obtain the parameter settings of Cycle-Derain. After that, we will input the unpaired images for training rather than the paired images, and thus the training time can be greatly reduced.

4.2 | Network model architecture

In the procedure of building the deraining network of Cycle-Derain, after resizing input images we obtain the image size of 256*256*3. Then, we apply 9 residual blocks to deal with the resized images. The convolution [23] with step size 2 is used for instance regularisation, and all elements in a single sample and a single channel are utilised to improve the clarity of input images. After the preliminary processing of deraining generators, the images are further processed by the ReLU activation function [24], and then the inverse convolution network [25] with step size 2 is utilised to generate the rain-free images which are with size of 256*256*3.

For the mapping process from rain-free region to rain-free region, the generated rain-free image is processed by a convolution network with the step size 2, and then tan h function [26] is taken as the activation function. Finally, the inverse convolution network is applied to generate the rainy images with the same size. The four generators in this procedure adopt the least square method to restrict and minimise the value of loss function. We set λ_1 to 10 in the loss function and employ the Adam decoder to achieve a better deraining effect.

Besides, the four deraining discriminators form a patch level discriminator architecture containing 70*70 PatchGAN, and adopt the least square method to constrain the loss function and guarantee the authenticity of images. The deraining discriminators judge whether the cycle consistency can be satisfied or not, and hence the number of parameters is much fewer than that of the full image discriminator. These discriminators are able to take any size of images in a fully convolution.

The images generated by the deraining generator are first processed through the convolution network, and then the

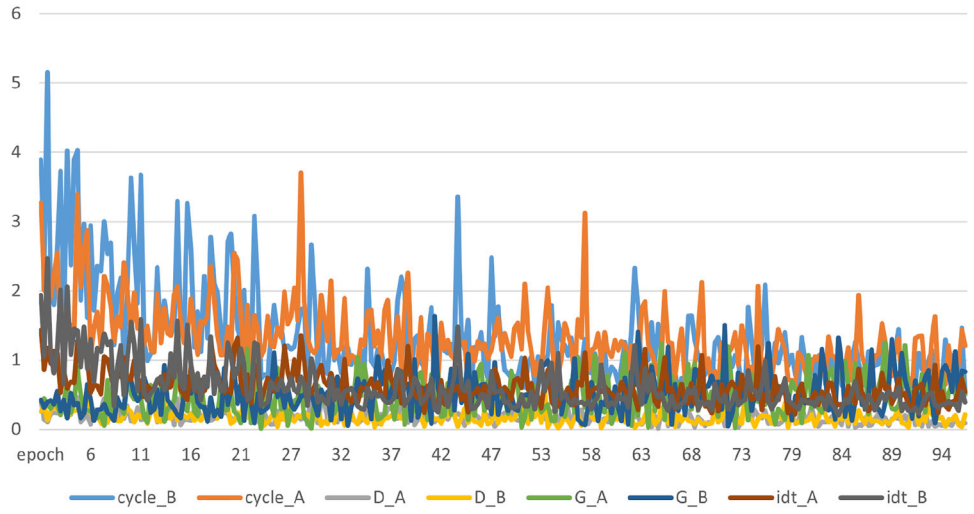


FIGURE 6 Translations-I from rainy images to rain-free images

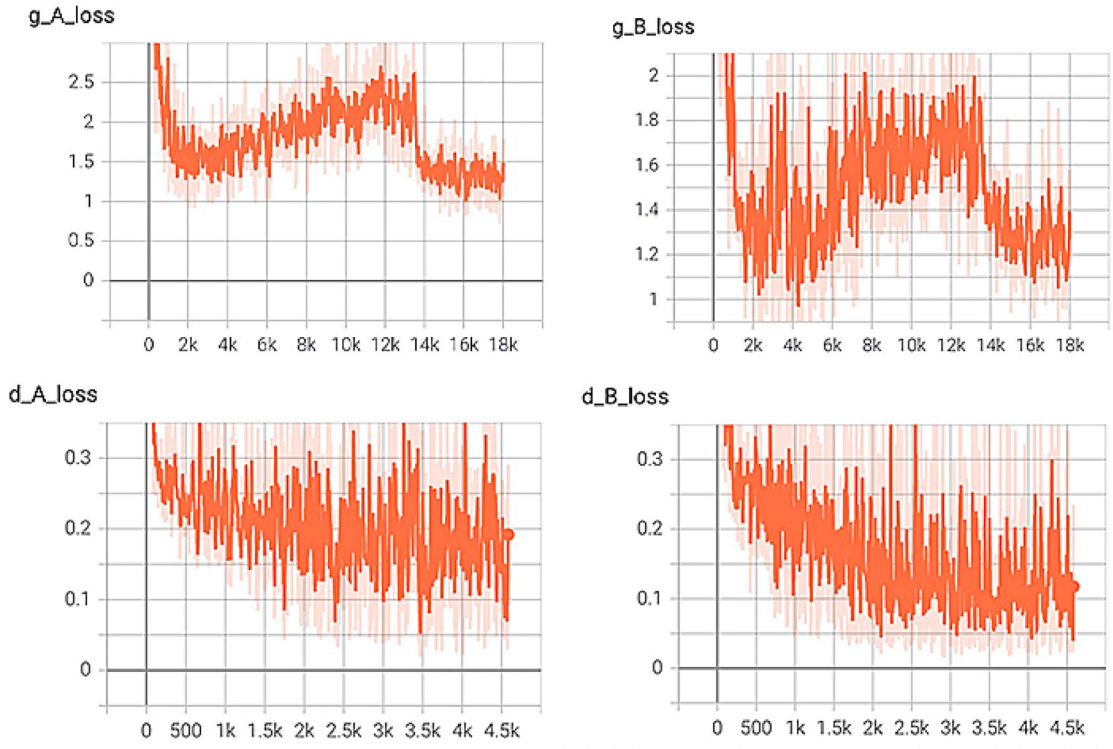


FIGURE 7 Translations-II from rainy images to rain-free images

convolution network is inserted into the Leaky ReLU activation function [27] for further calculation. After that, the image is input into the convolution network again for instance regularisation, and then the image with the size of 16*16*3 is generated by the Leaky ReLU.

InceptionV3 [28] is taken as the feature extractor in the defogging network of Cycle-Derain, which contains two convolution networks and a ReLU activation function. Specifically, the original deraining input images are processed by the convolution network, and the results are inserted into the ReLU activation function for the calculation of loss function. Then, a layer of convolution network is utilised to yield the weight graph. The feature extractor structure is shown in Figure 4.

Moreover, note that the defogging generator and defogging discriminator are trained simultaneously, and the KID [29]

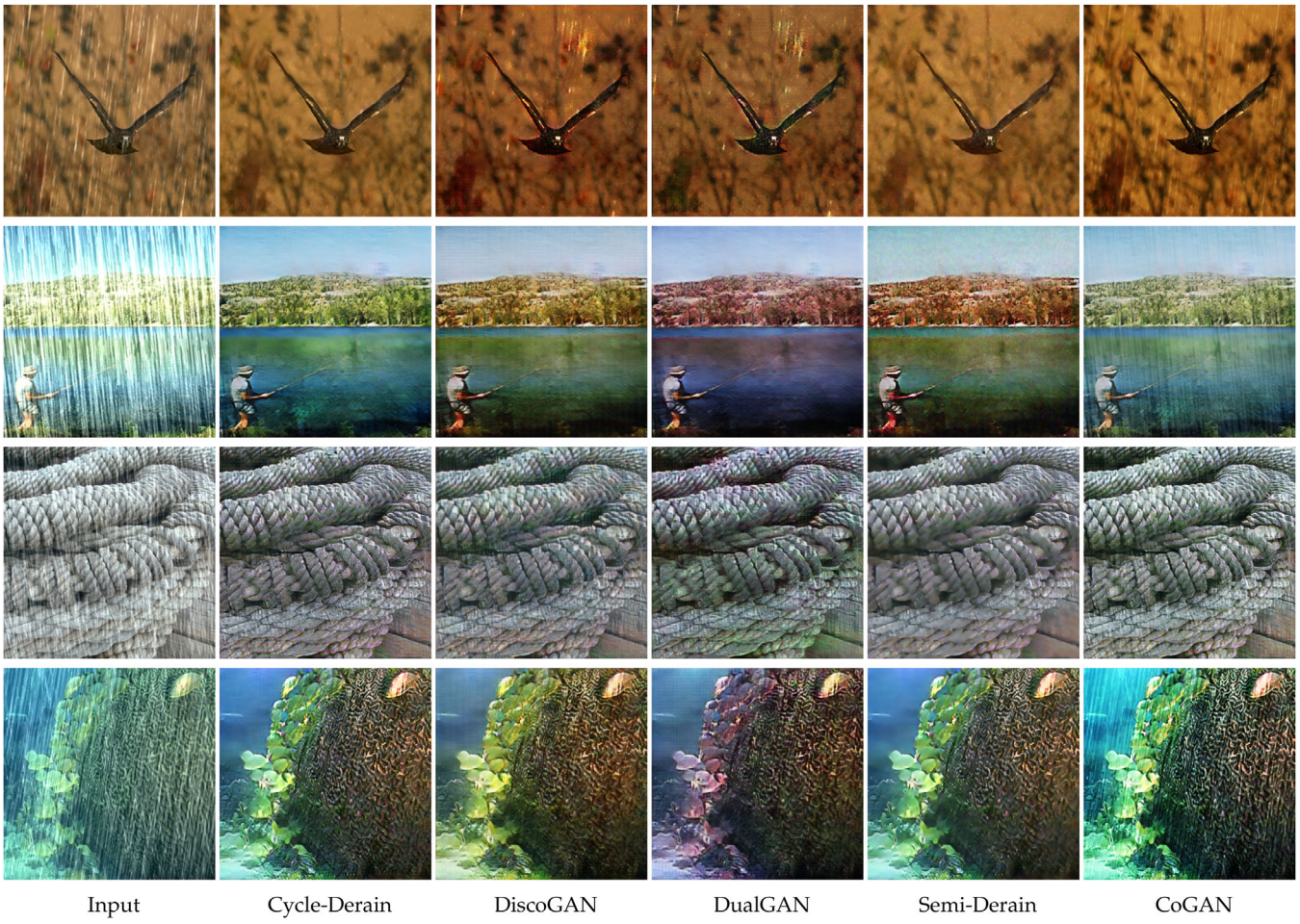


FIGURE 8 Loss function values with different epochs

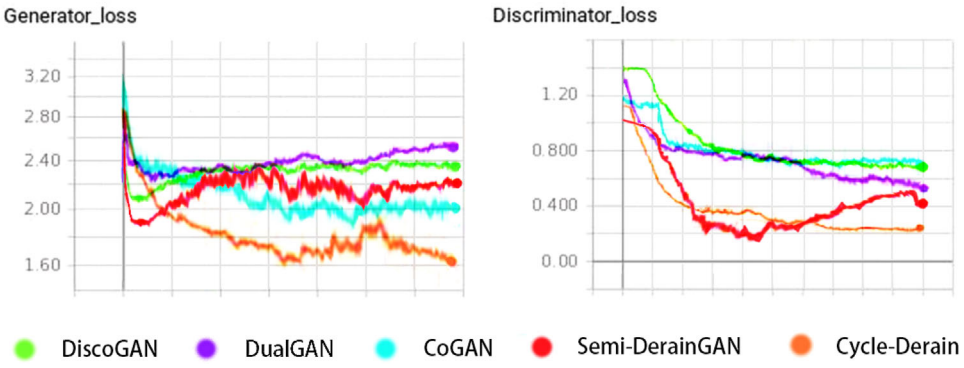


FIGURE 9 Loss function values of defogging part

algorithm with unbiased estimator is employed to constrain the loss function and enhance the reliability of Cycle-Derain, due to the following facts: the KID algorithm can quantify the characteristics of the original deraining images and the generated defogging images; the KID algorithm can measure the difference between the original deraining images and the generated defogging images.

4.3 | Training details

In deraining part, 1,096 images are utilised to conduct 400 training epochs, and the loss function results of the first 100 epochs are visually exhibited in Figure 8. In the beginning, the cycle consistency loss is quite large, and then it is rapidly reduced especially after 73 epochs. The generator and discriminator are

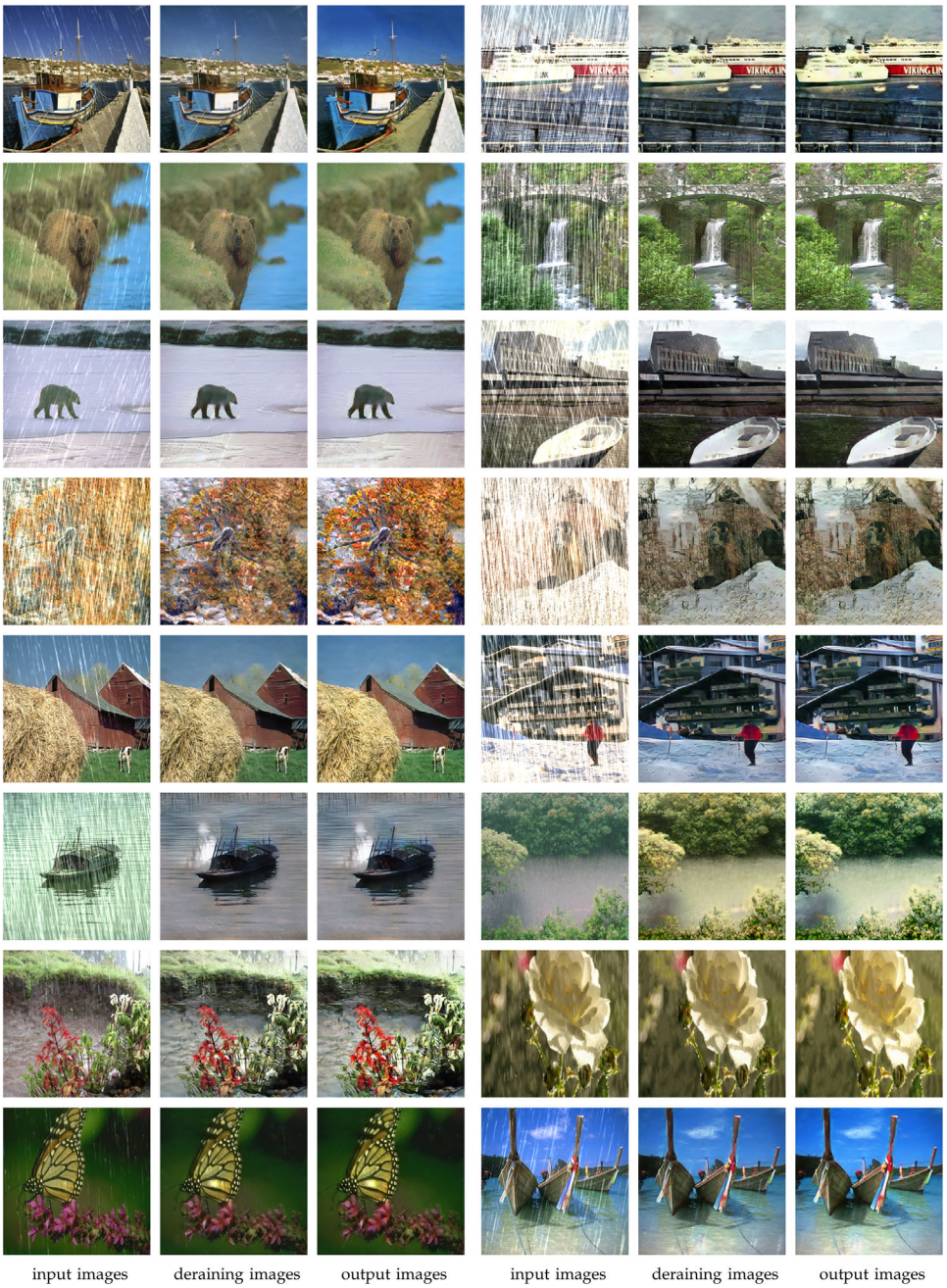


FIGURE 10 Comparison of deraining results by different methods

trained as expected, i.e. the value of loss function becomes smaller. The defogging training process utilises 18,000 images, and every 4 images produced by the generator will be delivered to the discriminator, by which the authenticity of generated images can be determined.

Figure 9 shows that after training 2,000 images, the loss function value is reduced to be very small. Furthermore, after training 10,000 images, the loss function value of generator is generally increased with some fluctuations, while the loss function value of discriminator is decreased with some fluctuations. This is because the generator and the discriminator are mutually restricted, and a balance will be achieved after a long-term iteration. After processing the next 6,000 images, the loss function values of both generators and discriminators are decreased slowly. Note that the loss function values fluctuate slightly, and they finally coverage to a very small value. This phenomenon also indicates that the generator and discriminator can achieve a Nash equilibrium. The learning rate of both the generator and discriminator are decreased accordingly.

When the number of input images is large enough, it is difficult to achieve better deraining effect. The results indicate that Cycle-Derain achieves a preferable tradeoff when 18,000

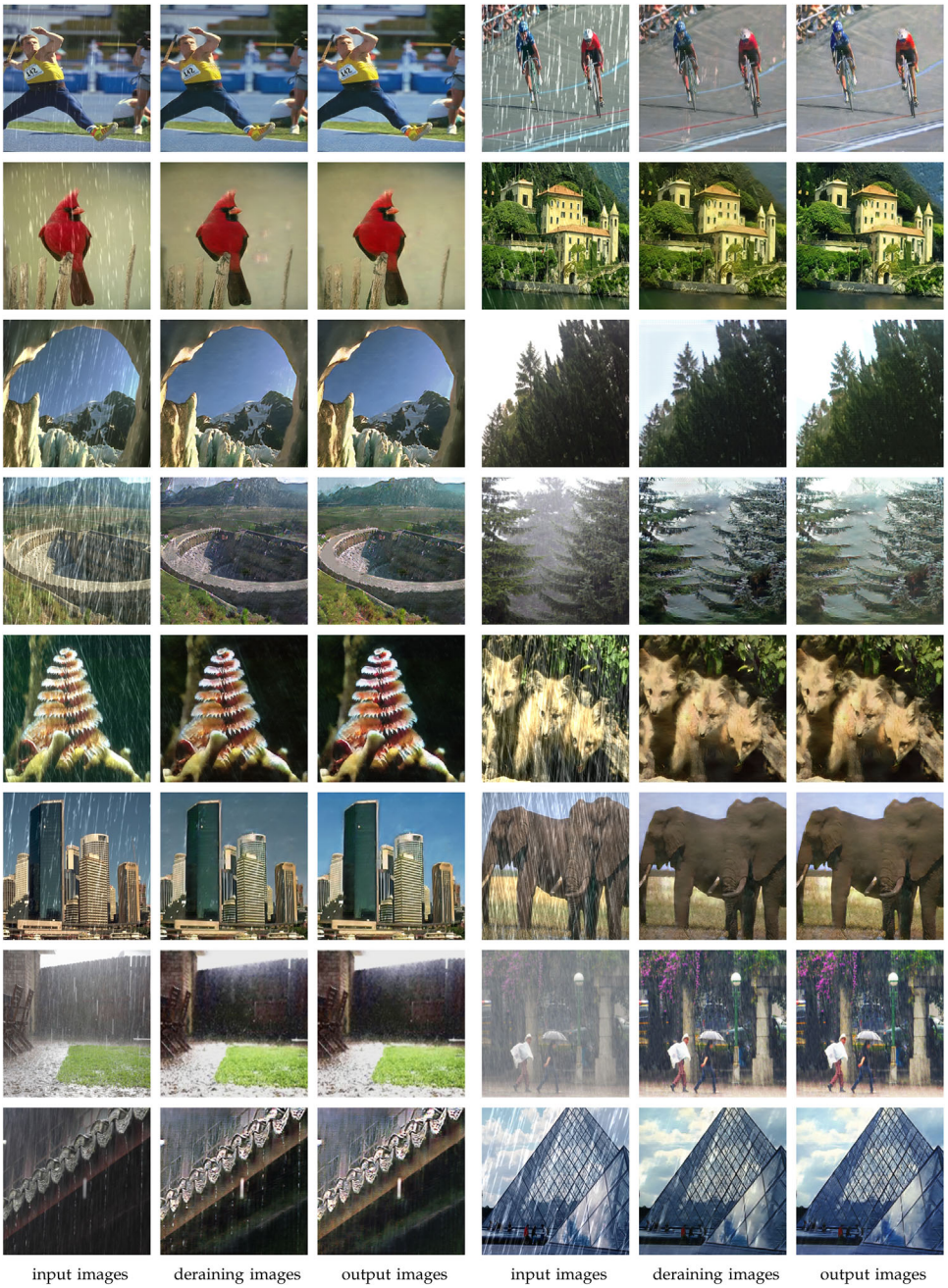


FIGURE 11 Comparison of loss function values of generators and discriminators

images are used for training. Experimental results are provided in Figures 6 and 7.

4.4 | Comparisons with state-of-the-arts

In this section, we compare Cycle-Derain with DisoGAN [30], DualGAN [31], Semi-DerainGAN[32] and CoGAN [33]. DiscoGAN and DualGAN have the similar architectures in image processing, while their losses are quite different: DiscoGAN adopts a standard GAN loss [30], and DualGAN adopts a Wasserstein GAN loss [31]. Firstly, in terms of image processing, 36,000 rainy images in different models are used for training on 2,080Ti GPU.

By comparing with the images provided in Figure 10, we find that Cycle-Derain can successfully derain the images and restore clear backgrounds. Moreover, Cycle-Derain exhibits a preferable effect on restoring the original colours. DualGAN removes most of the rain in images, but the colour of images is severely distorted. Semi-DerainGAN perform well in removing rain, but the output seem vague in details. Although CoGAN is able to restore the original colours, it cannot completely derain images

TABLE 1 Comparison of experimental results

Name	SSIM	PSNR/dB	FLOPS	Loss of generator	Loss of discriminator	Time processing each image /s
DiscoGAN	0.823	30.55	557	2.387	0.698	2.916
DualGAN	0.581	26.78	595	2.539	0.536	2.671
CoGAN	0.735	32.89	502	2.006	0.702	1.032
Semi-DerainGAN	0.602	27.91	570	2.210	0.409	3.287
Cycle-Derain	0.901	36.90	598	1.618	0.220	4.333

when CoGAN are trained by the same images. DiscoGAN performs better than DualGAN and CoGAN in term of deraining effect. However, the colour restoration is unsatisfactory, and the image sharpness is significantly damaged.

In terms of the loss function, as illustrated in Figure 11, when DiscoGAN and DualGAN adopt different losses, both the generator and the discriminator have similar loss function values. The loss function value of the final generator of CoGAN is obviously lower than those of DiscoGAN and DualGAN. Besides, the loss function value of the discriminator is similar to that of DiscoGAN. These results also indicate that Semi-DerainGAN perform well in removing the rain, while the output images seem vague in details. Especially, when the number of input images is large enough (the model is sufficiently trained), the performance of Semi-DearinGAN is worse than that of our proposed method.

Despite the initial loss function value of the Cycle-Derain generator exhibits very high, after training about 5,000 images, the loss function values of both generators and discriminators are much lower than those of others.

As shown in Table 1, since Cycle-Derain has introduced the loop-search positioning algorithm, the computation complexity is slightly higher than others. Since Cycle-Derain has introduced the loop-search positioning algorithm, the computation complexity for training the model is slightly higher than others. Because Cycle-Derain processes both the rain layer and the fog layer, and thus it consumes more time. However, after the training process, the computation complexity for deraining the images of Cycle-Derain is very close to others, which can be demonstrated by the results of FLOPS (floating-point operations per second) given in TABLE 1. Therefore, the average training time is longer, while the loss function values of Cycle-Derain are evidently lower than other methods. More importantly, by comparing with them in SSIM (Structural Similarity) and PSNR (peak signal to noise ratio), it is obvious that the quality of images processed by Cycle-Derain is much higher than that of other methods.

5 | CONCLUSION

Single image deraining has become a vital issue in the field of computer vision processing. However, the paired image datasets may be not available in real applications. To this end, we propose the Cycle-Derain model to process the rainy images in an unsupervised manner. Moreover, most existing approaches have ignored the fog layers in the rainy images, and they always process the rainy images by using some physical models, making the processing models become much complex and with large computation complexity. In our proposed model, a loop-search positioning algorithm is specially introduced to deal with the background and the foreground of images, and the removal of fog layers is iteratively implemented and judged to output better images.

ACKNOWLEDGMENTS

This research is supported by National Natural Science Foundation of China under Grant Nos. 61872191, 61872193; Six Talents Peak Project of Jiangsu Province under Grant No. 2019-XYDXX-247.

ORCID

Linfeng Liu https://orcid.org/0000-0002-0824-6203

REFERENCES

1. Goodfellow, I., Pouget-Abadie, J., Mirza, M.: Generative adversarial nets. In: Advances in Neural Information Processing Systems, pp. 2672–2680. Curran Associates, Red Hook, NY (2014)
2. Fu, X., Huang, J., Ding, X.: Clearing the skies: a deep network architecture for single-image rain removal. IEEE Trans. Image Process. 26(6), 2944–2956 (2017)
3. Wei, W., et al.: Semi-supervised CNN for single image rain removal. arXiv:1807.11078, (2018)
4. Wei, Y., Zhang, Z., Fan, J.: DerainCycleGAN: an attention-guided unsupervised benchmark for single image deraining and rainmaking. arXiv:1912.07015, (2019)
5. Mejjati, Y.A., Richardt, C., Tompkin, J.: Unsupervised attention-guided image-to-image translation. In: Advances in Neural Information Processing Systems, pp. 3693–3703. Curran Associates, Red Hook, NY (2018)
6. Zhu, J.Y., Park, T., Isola, P.: Unpaired image-to-image translation using cycle-consistent adversarial networks. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 2223–2232. IEEE, Piscataway, NJ (2017)
7. Qian, R., et al.: Attentive generative adversarial network for raindrop removal from a single image. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, pp. 2482–2491. IEEE, Piscataway, NJ (2018)
8. Wang, Y., et al.: Gradient information guided deraining with a novel network and adversarial training. arXiv:1910.03839, (2019)
9. He, K., et al.: Deep residual learning for image recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 770–778. IEEE, Piscataway, NJ (2016)
10. Chen, L.C., et al.: Encoder-decoder with atrous separable convolution for semantic image segmentation. In: Proceedings of the European

Conference on Computer Vision (ECCV), pp. 801–818. Springer, Berlin, Heidelberg (2018)
11. Fu, H., et al.: Deep ordinal regression network for monocular depth estimation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 2002–2011. IEEE, Piscataway, NJ (2018)
12. Li, R., Cheong, L.F., Tan, R.T.: Heavy rain image restoration: Integrating physics model and conditional adversarial learning. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 1633–1642. IEEE, Piscataway, NJ (2019)
13. Matsui, T., Ikehara, M.: GAN-based rain noise removal from single-image considering rain composite models. IEEE Access 8, 40892–40900 (2020)
14. Lu, B., Chen, J.C., Chellappa, R.: Unsupervised domain-specific deblurring via disentangled representations. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 10225–10234. IEEE, Piscataway, NJ (2019)
15. Yuan, Y., Liu, S., Zhang, J.: Unsupervised image super-resolution using cycle-in-cycle generative adversarial networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, pp. 701–710. IEEE, Piscataway, NJ (2018)
16. Wei, W., Meng, D., Zhao, Q.: Semi-supervised transfer learning for image rain removal. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 3877–3886. IEEE, Piscataway, NJ (2019)
17. Luo, Y., Xu, Y., Ji, H.: Removing rain from a single image via discriminative sparse coding. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 3397–3405. IEEE, Piscataway, NJ (2015)
18. Jiang, K., Wang, Z., Yi, P.: Multi-scale progressive fusion network for single image deraining. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 8346–8355. IEEE, Piscataway, NJ (2020)
19. Yang, W., Tan, R.T., Feng, J.: Deep joint rain detection and removal from a single image. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1357–1366. IEEE, Piscataway, NJ (2017)
20. Ren, D., Zuo, W., Hu, Q.: Progressive image deraining networks: a better and simpler baseline. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 3937–3946. IEEE, Piscataway, NJ (2019)
21. Wang, T., Yang, X., Xu, K.: Spatial attentive single-image deraining with a high quality real rain dataset. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 12270–12279. IEEE, Piscataway, NJ (2019)
22. Shen, Y., Feng, Y., Deng, S.: MBA-RainGAN: multi-branch attention generative adversarial network for mixture of rain removal from single images. arXiv:2005.10582, (2020)
23. Wang, H., Xie, Q., Zhao, Q.: A model-driven deep neural network for single image rain removal. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 3103–3112. IEEE, Piscataway, NJ (2020)
24. Liu, M.Y., Tuzel, O.: Coupled generative adversarial networks. In: Advances in Neural Information Processing Systems, pp. 469–477. Curran Associates, Red Hook, NY (2016)
25. Deng, L.J., Huang, T.Z., Zhao, X.L.: A directional global sparse model for single image rain removal. Appl. Math. Modell. 59, 662–679 (2018)
26. Engin, D., Gen, A., Ekenel, H.: Cycle-dehaze: Enhanced CycleGAN for single image dehazing. In: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), pp. 938–9388. IEEE, Piscataway, NJ (2018)
27. Wang, H., et al.: Meng. A survey on rain removal from video and single image. arXiv:1909.08326, (2019)
28. Szegedy, C., Vanhoucke, V., Ioffe, S.: Rethinking the inception architecture for computer vision. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2818–2826. IEEE, Piscataway, NJ (2016)
29. Kurach, K., Lucic, M., Zhai, X.: A large-scale study on regularization and normalization in GANs. In: International Conference on Machine Learning, pp. 3581–3590. Microtome Publishing, Brookline, MA (2019)
30. Kim, T., Cha, M., Kim, H.: Learning to discover cross-domain relations with generative adversarial networks. In: International Conference on Machine Learning, pp. 1857–1865, ACM Press, New York (2017)
31. Yi, Z., Zhang, H., Tan, P.: Dualgan: unsupervised dual learning for image-to-image translation. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 2849–2857. IEEE, Piscataway, NJ (2017)
32. Wei, Y., Zhang, Z., Zhang, H.: Semi-DerainGAN: a new semi-supervised single image deraining network. arXiv:2001.08388, (2020)
33. Liu, M.Y., Tuzel, O.: Coupled generative adversarial networks. arXiv:1606.07536, (2016)

How to cite this article: Song, Z., et al.: An unsupervised generative adversarial network for single image deraining. IET Image Process. 15, 3105–3117 (2021). https://doi.org/10.1049/ipr2.12301