

Single Traffic Image Deraining via Similarity-Diversity Model

Youxing Li, Rushi Lan, Huiwen Huang, Huiyu Zhou, Zhenbing Liu, Cheng Pang and Xiaonan Luo

Abstract—Single traffic image deraining technology based on deep learning is a vital branch of image preprocessing, which is of great help to intelligent monitoring systems and driving navigation system. It is well understood that established deraining methods are derived based on one specific imaging model, neglecting the underlying correlations between different weather models and thereby limiting the applicability of these standard methods in real scenarios. To ameliorate this issue, in this work, we first explore the inherent relationship between a rain model and the haze one established up to date. We discover that these two models experience similar degradations in the low-frequency components (i.e., similarity) but diverse degradations in the high-frequency areas (i.e., diversity). Based on these observations, we develop a Similarity-Diversity model to describe these characteristics. Afterwards, we introduce a novel deep neural network to restore the rain-free background embedding the similarity-diversity model, namely deep similarity-diversity network (DSDNet). Extensive experiments have been conducted to evaluate our proposed method that outperforms the other state of the art deraining techniques. On the other hand, we deploy the proposed algorithm with Google Vision API for object recognition, which also obtains satisfactory results both qualitatively and quantitatively.

Index Terms—Deep Learning, Single Traffic Image Deraining, Imaging Model, Similarity, Diversity.

I. INTRODUCTION

ONE of the great challenges faced by Intelligent Transportation Systems is how to work efficiently under rainy conditions. However, traffic images captured in rainy days largely suffer from visual degradation effects that have a severe impact on outdoor vision tasks such as small object detection [1] and license plate recognition [2]–[4]. To alleviate this adverse outcome, removing rain streaks from the degraded images (single image deraining) become very important. Research studies on single image deraining have attracted large attention in recent years.

This work was partially supported by the the National Key Research and Development Program of China (2018AAA0100300), National Natural Science Foundation of China (Nos. 61772149, U1701267, and 61762028), Guangxi Science and Technology Project (Nos. ZY20198016, AB20238013, 2019GXNSFFA245014, and AD18216004), and Guangxi Key Laboratory of Image and Graphic Intelligent Processing (No. GIIP2001).

Y. Li, R. Lan, H. Huang, Z. Liu and C. Pang are with the Guangxi Key Laboratory of Intelligent Processing of Computer Image and Graphics, Guilin University of Electronic Technology, Guilin 541004, China. (Emails: 940756344@qq.com, rslan2016@163.com, 18814379641@163.com, zbliu2011@163.com, pangcheng3@guet.edu.cn)

H. Zhou is with School of Informatics, University of Leicester, United Kingdom. (Email: hz143@leicester.ac.uk)

X. Luo is with the National Local Joint Engineering Research Center of Satellite Navigation and Location Service, Guilin University of Electronic Technology, Guilin 541004, China. (Email: luoxn@guet.edu.cn)

In the past decade, a number of approaches have been proposed to address this challenge. Early approaches generally regard single image deraining as an optimization problem of layer separation, mainly focusing on separating the rain streak and the rain-free background layers employing technologies such as sparse coding [5], low-rank representation [6] and Gaussian mixture models [7]. In recent years, promising progress has been achieved on the single image deraining task due to the rapid growth of using deep learning, especially deep convolutional neural networks (CNNs). Several CNNs-based methods [8]–[13] have been constructed to use various feature extraction modules for learning the distinctive characteristics of rain streaks, and then restore the rain-free counterparts. For example, in [8], a multi-scale fusion network is presented, utilizing a pyramid architecture as backbone to capture rain streaks with different appearances and scales. In [9], [14], dilated convolution has been employed, which captures large-scale contextual information to facilitates rain removal and detail recovery. To handle the entanglement of rain streaks and haze, in [15], a heavy rain removal method is proposed to integrate physics-based restoration and generative adversarial learning for image deraining. In [16], an attention mechanism is utilized to highlight the effect of rain streaks and haze with different scene depths.

Although the existing deraining methods have achieved promising performance, most of them are developed based on one particular imaging model to describe the degradation process on rainy days, which cannot cover diverse situations in real world, and thereby limits the applicability of these approaches on real rainy traffic images. One solution is to design a model-free framework [17], [18] for image deraining. However, such framework lacks the interpretability on the physics level and easy to over-fit the training samples, hindering the development of the following work.

To overcome the above problem while maintaining the interpretability on the physics level, we first explore the underlying correlations between the rain and the haze models, and discover that they share similar degradations in the low-frequency components, both of which can be approximated by a mixture of the atmospheric light and the transmission map. On the contrary, they have diverse degradations in the high-frequency components, among which the rainy model generates severe shading of rain streaks, as shown in Fig. 1. Driven by this discovery, we construct a Similarity-Diversity model in this paper to describe their relationship. Furthermore, we put forward a novel network that contains three subnetworks to restore the rain-free background from the rainy images according to their similarity and diversity. To better reflect the

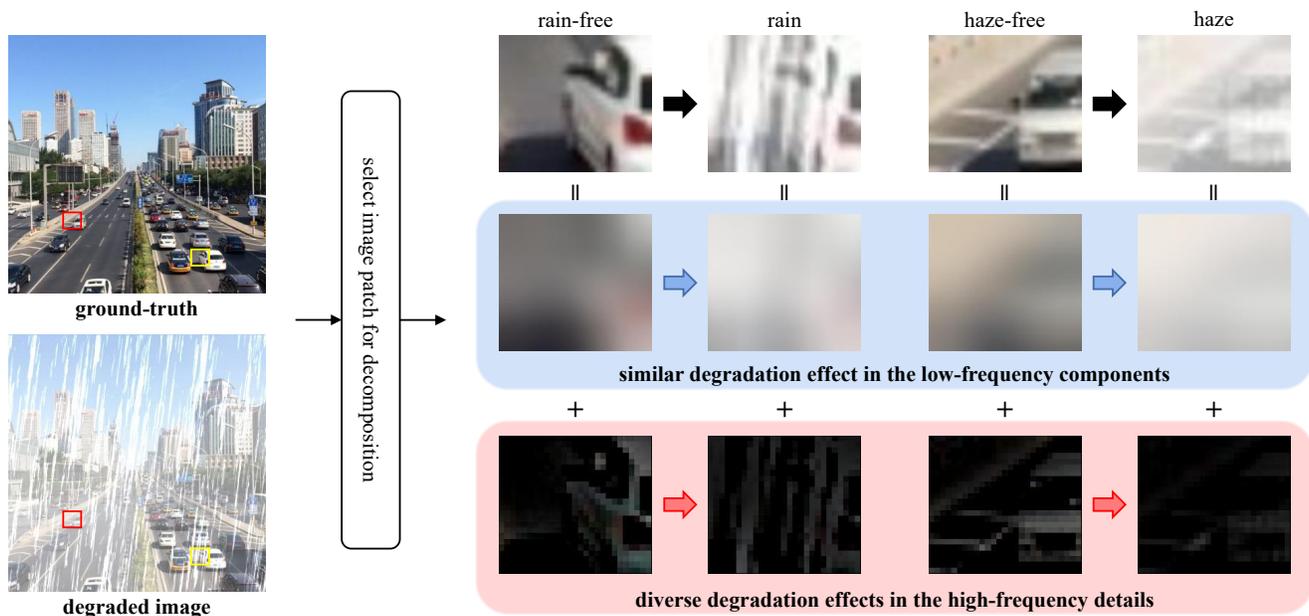


Fig. 1. Rain-free/Rainy patch pair and haze-free/haze patch pair have similarity degradation effect in the low-frequency components (blue arrow), but diversity degradation effects in the high-frequency details (red arrow).

similarity, two subnetworks TLNet and ALNet are proposed to estimate the atmospheric light and the transmission map, respectively, in the low-frequency components of the image and then restore the corresponding rain-free components. To better handle this diversity, a subnetwork HNet is proposed to directly reconstruct the high-frequency details of the image.

To summarize this work, our main contributions are:

- We discover that the rain and the haze models share similar degradations in the low-frequency components but diverse degradations in the high-frequency areas, and then develop a similarity-diversity model to describe these correlations, through which, alleviating the problem that the applicability of the specific imaging model is weak in practice.
- We propose a novel network for single traffic image deraining according to the similarity-diversity. Our network takes full advantage of the underlying correlations between the rain and the haze models, thus can deal with diverse rain streaks and their entanglements with haze.
- We undertake extensive experiments to demonstrate that our method is superior to the other state-of-the-art methods on the image deraining task. Moreover, our method effectively promote the recognition rates of vehicle and pedestrian with Google Vision API on rainy images.

II. RELATED WORK

A. Single Image Deraining

A rainy image I^r is usually simplified as the linear superposition of a rain streak layer \mathbf{R} and a rain-free background layer \mathbf{B} . Early methods rely on diverse image priors to separate the layers from the degraded image. For example, Chen *et al.* [6] take advantage of a low-rank prior to capture rain streaks from the rainy image. Luo *et al.* [5] propose discriminative

sparse coding (DSC) to separate the rain streak and the rain-free background layers from the degraded images. Li *et al.* [7] report a patch-prior based on Gaussian Mixture Models (GMM), used to model the rain streak layer and the clean background image from its rain-polluted version.

Recently, CNNs-based methods are prevailing in image restoration, including single image deraining. For instance, Fu *et al.* [19] produce a deep detail network (DDN) for image rain removal. DDN adopts the ResNet [20] structure as backbone and introduce negative residual learning to reduce the mapping range from the input to the output. Wang *et al.* [21] develop a Spatial Attentive Network (SPANet) that employs four-directional IRNN [22] for image deraining in an local-to-global manner. Jiang *et al.* [8] design a network (MSPFN) to estimate rain streaks and restore the clean background from multi-scale images and features, aiming to capture the rain streaks with different appearances and scales. Considering the veil effect caused by accumulated rain streaks in the images, a novel rain model [9] is formed as:

$$I^r = \mathbf{T} * (\mathbf{B} + \mathbf{R}) + (\mathbf{1} - \mathbf{T}) * \mathbf{A}, \quad (1)$$

where \mathbf{T} is the transmission map, \mathbf{A} is the global atmospheric light, $\mathbf{1}$ is a matrix of ones, and $*$ represents the element-wise multiplication. Li *et al.* [15] introduce a heavy rain removal method (HRGAN), where they first restore the clean background using Eq. (1), and then combine the model-free framework (i.e. conditional generation adversarial learning) to further improve the details and color of the background. To better describe the entanglement of rain streaks and haze, Hu *et al.* [16] construct a depth-attentional feature network (DAFNet) which utilizes scene depths to estimate the effect of rain streaks and haze at different positions on the degraded image. Wang *et al.* [23] investigate the rain streaks' properties, and regard rain streaks as vapor and reformulate Eq. (1) to facilitate

image deraining.

B. Single Image Dehazing

In comparison, a hazy image \mathbf{I}^h can be modeled [24], [25] as follows:

$$\mathbf{I}^h = \mathbf{T} * \mathbf{B} + (\mathbf{1} - \mathbf{T}) * \mathbf{A}. \quad (2)$$

Traditional methods usually estimate \mathbf{T} and \mathbf{A} utilizing various empirical priors such as Maximum Contrast [26], dark channel prior [27] and color attenuation [28]. Deep learning-based approaches for haze removal are somehow similar to the deraining methods, which restore the haze-free background by designing various networks according to Eq. (2). For example, Li *et al.* [29] present a new AOD-Net considering all the degradation factors, which directly generates a clean image from a hazy image. Dong *et al.* [30] design a multi-scale dehazing network MSBDN-DFF, employing a Strengthen-Operate-Subtract (SOS) boosting strategy [31] to facilitates haze removal and a dense features fusion module based on the back-projection technique [32] to effectively preserve spatial information of different scales.

Overall, established methods are based on one specific imaging model. Different from these methods, our approach focuses on the traffic image deraining task, taking full advantage of the correlation between different weather models, and therefore significantly improves the applicability of the proposed method in real scenarios.

III. SIMILARITY-DIVERSITY MODEL

We observe that rainy and hazy patches share similar degradation effects in the low-frequency components, and considerably diverse effects in the high-frequency details, as shown in Fig. 1. To explore this further, we decompose specific rain and haze models respectively from the perspective of frequency in order to explore possible correlations between them.

To begin the analysis, we decompose a degraded color (RGB) image \mathbf{I} as:

$$\mathbf{I} = \mathbf{I}_L + \mathbf{I}_H, \quad (3)$$

where $(\cdot)_L$ and $(\cdot)_H$ represent the low- and the high-frequency components of (\cdot) , respectively. Based on this form, the rain model (Eq. (1)) can be re-written as:

$$\mathbf{I}^r = \mathbf{T}_L * (\mathbf{B}_L + \mathbf{R}_L) + (\mathbf{1} - \mathbf{T}_L) * \mathbf{A}_L + \mathbf{T}_H * (\mathbf{B}_H + \mathbf{R}_H) + (\mathbf{1} - \mathbf{T}_H) * \mathbf{A}_H. \quad (4)$$

Similarly, the haze model can be split into the following form:

$$\mathbf{I}^h = \mathbf{T}_L * \mathbf{B}_L + (\mathbf{1} - \mathbf{T}_L) * \mathbf{A}_L + \mathbf{T}_H * \mathbf{B}_H + (\mathbf{1} - \mathbf{T}_H) * \mathbf{A}_H. \quad (5)$$

Using the assumptions made in the previous methods [19], [27], we understand that rain streaks mainly exist in the high-frequency details while it can be neglected in the low-frequency components (i.e., $\mathbf{R}_H \neq \mathbf{0}$ and each element in \mathbf{R}_L is approximately equal to zero, among which $\mathbf{0}$ is a matrix of zeroes). The intensities of atmospheric light are equal between any two positions through the image, therefore the corresponding change rate is equal to zero, that is, the

atmospheric light belongs to low-frequency information (i.e., $\mathbf{A}_L \neq \mathbf{0}$ and $\mathbf{A}_H = \mathbf{0}$). In the transmission map, each element value is attenuated exponentially with the scene depth. Therefore, the transmission map exists in both the low- and the high-frequency components, and each element values are greater than zero (that is, $\mathbf{T}_L \neq \mathbf{0}$ and $\mathbf{T}_H \neq \mathbf{0}$). Therefore, the rain model can be converted to the following form:

$$\mathbf{I}^r = \mathbf{T}_L * \mathbf{B}_L + (\mathbf{1} - \mathbf{T}_L) * \mathbf{A}_L + \mathbf{T}_H * (\mathbf{B}_H + \mathbf{R}_H). \quad (6)$$

The haze model can be updated as follows:

$$\mathbf{I}^h = \mathbf{T}_L * \mathbf{B}_L + (\mathbf{1} - \mathbf{T}_L) * \mathbf{A}_L + \mathbf{T}_H * \mathbf{B}_H. \quad (7)$$

We notice that the rain model (Eq. (6)) and the haze model (Eq. (7)) share similar degradations in the low-frequency components, both of which can be represented by a mixture of \mathbf{T}_L and \mathbf{A}_L (i.e., $\mathbf{T}_L * \mathbf{B}_L + (\mathbf{1} - \mathbf{T}_L) * \mathbf{A}_L$). On the contrary, they demonstrate diverse degradations in the high-frequency details, and the rain model produces certain shading of rain streaks. Based on this observation, we design a Similarity-Diversity model combining Eqs. (6) and (7) as follows:

$$\mathbf{I} = \mathbf{T}_L * \mathbf{B}_L + (\mathbf{1} - \mathbf{T}_L) * \mathbf{A}_L + \mathbf{F}^{-1}(\mathbf{B}_H), \quad (8)$$

where

$$\mathbf{F}^{-1}(\mathbf{B}_H) = \begin{cases} \mathbf{T}_H * (\mathbf{B}_H + \mathbf{R}_H), & \text{if } \mathbf{I} = \mathbf{I}^r, \\ \mathbf{T}_H * \mathbf{B}_H, & \text{if } \mathbf{I} = \mathbf{I}^h, \end{cases} \quad (9)$$

where $\mathbf{F}^{-1}(\cdot)$ represents the diverse degradation of (\cdot) . $\mathbf{F}^{-1}(\mathbf{B}_H)$ is equal to $\mathbf{T}_H * (\mathbf{B}_H + \mathbf{R}_H)$ when the degraded image is a rainy image, and $\mathbf{F}^{-1}(\mathbf{B}_H)$ is equal to $\mathbf{T}_H * \mathbf{B}_H$ when the degraded image is a hazy image.

Combining Eqs. (3) and (8), we can restore the low- and high-frequency details of the rain-free background respectively as:

$$\mathbf{B}_L = \frac{\mathbf{I}_L - (\mathbf{1} - \mathbf{T}_L) * \mathbf{A}_L}{\mathbf{T}_L}, \quad (10)$$

and

$$\mathbf{B}_H = \mathbf{F}(\mathbf{I}_H), \quad (11)$$

where

$$\mathbf{F}(\mathbf{I}_H) = \begin{cases} (\mathbf{I}_H / \mathbf{T}_H) - \mathbf{R}_H, & \text{if } \mathbf{I} = \mathbf{I}^r, \\ \mathbf{I}_H / \mathbf{T}_H, & \text{if } \mathbf{I} = \mathbf{I}^h, \end{cases} \quad (12)$$

where $\mathbf{F}(\cdot)$ stands for the diverse restoration process in the high-frequency section and $/$ is the sign of element-wise division. Finally, the clean background can be reconstructed as:

$$\mathbf{B} = \frac{\mathbf{I}_L - (\mathbf{1} - \mathbf{T}_L) * \mathbf{A}_L}{\mathbf{T}_L} + \mathbf{F}(\mathbf{I}_H). \quad (13)$$

To sum up, we construct similarity and diversity between the rain and the haze models, and develop a new background reconstruction method (Eq. (13)), which plays an important role in the subsequent network design.

IV. NETWORK DESIGN

According to the analysis of similarity-diversity in the previous section, our network is mainly composed of two-stage work. The first stage includes the similarity reconstruction

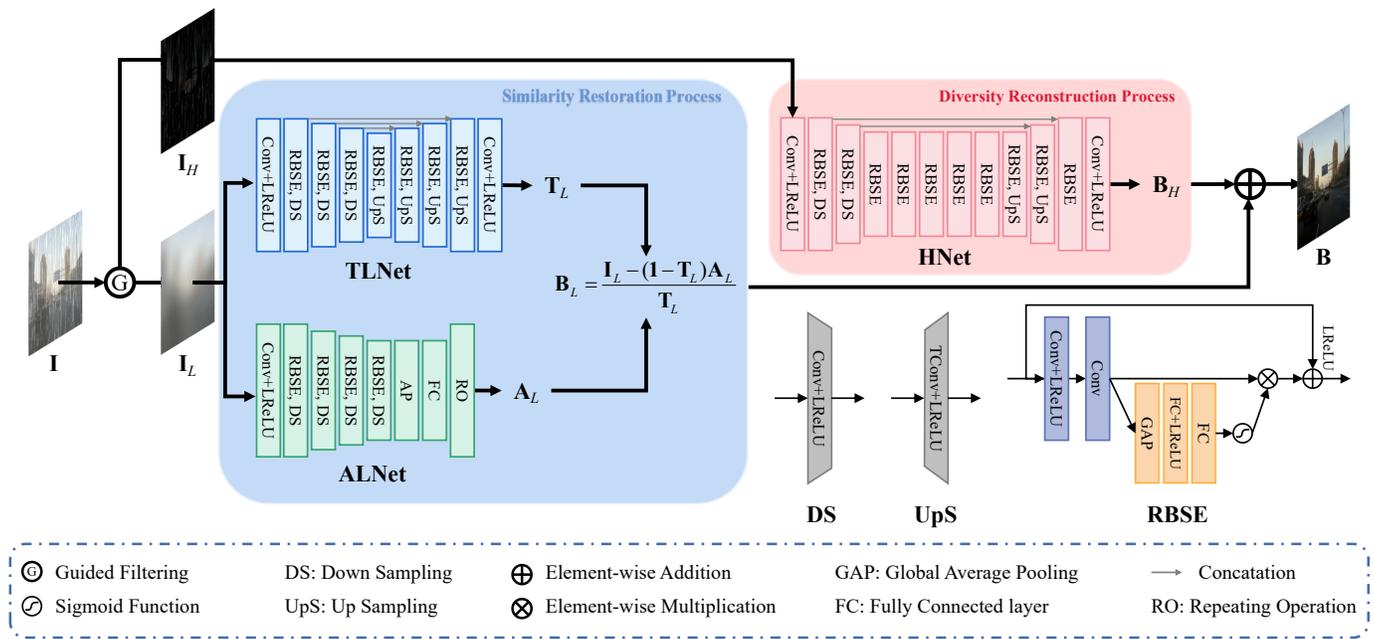


Fig. 2. The proposed network architecture. We adopt the Residual Block [20] combined with Squeeze-and-Excitation [33] (RBSE) as the basic block for feature extraction. Down and up sampling in our network are achieved with convolution (Conv) and transposed convolution (TConv) respectively. RO represents up sampling by repeating features.

process and the diversity reconstruction process is discussed in the second stage. The overall architecture of our network is shown in Fig. 2.

We first decompose the input I using the guided filter [34] to obtain the low-frequency information I_L , where the guidance image is the residual channel result [35] of I . I_L is then subtracted from I to get the corresponding high-frequency details I_H . We use this kind of residual channel images for guided filtering, aiming to decompose the rain streaks into the high-frequency details as much as possible. We further perform the similarity restoration process to reconstruct the rain-free component B_L from I_L , and then implement the diversity reconstruction process to reconstruct the clean high-frequency details B_H from I_H . Finally, the restored B_L and B_H are combined to generate the clean background B .

A. Similarity Restoration Process

In this stage, we restore the low-frequency content of the clean background B_L . To reflect the similarity between the rain and the haze models, we introduce two subnetworks TLNet and ALNet to estimate the low-frequency components of transmission map T_L and atmospheric light A_L respectively from input I_L , and then restore B_L by Eq. (10). A crucial problem is how we can produce T_L and A_L in the presence of only the input I_L . In this paper, we utilize the pre-training strategy reported in [23] to address the above problem. Therefore, there are two options to derive T_L and A_L : (1) Pre-train ALNet to estimate A_L and then train TLNet to estimate T_L ; or (2) Pre-train TLNet to estimate T_L and later train ALNet to estimate A_L . It should be noted that we need to calculate a priori result to approximate T_L when pre-training ALNet in option (1) or to approximate A_L when pre-training

TLNet in option (2). To minimize the priori errors during the pre-training, we choose the latter option because A_L , which is constant through the image, can be calculated easier than scene depth-related T_L .

1) *TLNet*: As shown in Fig. 2, TLNet uses the standard U-NET structure [36] as backbone and the Residual Block [20] is combined with Squeeze-and-Excitation [33] (named RBSE) to form the basic feature extraction module because of its simplicity and efficiency. We do not apply any complex module (e.g. dilated convolution [37], self-calibrated convolutions [38], or multi-scale fusion [39]) to the network due to the computational overhead. To accurately estimate T_L , we first calculate the average value A_p using top 0.1 percentage of the bright pixels in the input I_L to approximate A_L (i.e., $A_L \approx A_p$):

$$A_p = \frac{1}{0.001N(I_L)} \sum Top_{0.1\%}(I_L), \quad (14)$$

where $N(\cdot)$ is equal to the total number of elements in (\cdot) . We then estimate T_L through TLNet:

$$T_L = TLNet(I_L). \quad (15)$$

where $TLNet(\cdot)$ represents the trainable TLNet. The coarse low-frequency components of the clean background B_{L-c} is then restored by:

$$B_{L-c} = \frac{I_L - (1 - T_L) * A_p}{T_L}. \quad (16)$$

In the pre-training phase, we use the negative SSIM loss [40] to optimize $TLNet(\cdot)$:

$$\mathcal{L}_T = 1 - SSIM(B_{L-c}, B_{L-gt}), \quad (17)$$

where $(\cdot)_{-gt}$ is the ground-truth of (\cdot) .

2) *ALNet*: We further put forward a subnetwork *ALNet* to improve the estimation of \mathbf{A}_L , composed of the RBSE and FC layers. \mathbf{A}_L is then estimated as follows:

$$\mathbf{A}_L = \text{ALNet}(\mathbf{I}_L). \quad (18)$$

where $\text{ALNet}(\cdot)$ represents the trainable *ALNet*. At the same time, we use the pre-trained *TLNet* to predict \mathbf{T}_L and then restore the fine low-frequency components of the clean background:

$$\mathbf{B}_L = \frac{\mathbf{I}_L - (\mathbf{1} - \mathbf{T}_L) * \mathbf{A}_L}{\mathbf{T}_L}. \quad (19)$$

The loss function we used to train $\text{ALNet}(\cdot)$ is described as:

$$\mathcal{L}_{TA} = 1 - \text{SSIM}(\mathbf{B}_L, \mathbf{B}_{L_{gt}}). \quad (20)$$

In the meantime, we fine-tune the pre-trained $\text{TLNet}(\cdot)$ to maintain the accuracy of the predicted \mathbf{T}_L . We understand that *ALNet* further reduce the error, introduced by Eq. (14), in an attempt to restore the background color. Comparison details will be presented in the ablation study of Section 5.

B. Diversity Reconstruction Process

In this stage, we focus on the restoration of the high-frequency details of the clean background \mathbf{B}_H . To effectively handle the diversity between the rain and the haze models while avoiding the priori errors about the high-frequency details, we introduce a subnetwork *HNet* to directly restore \mathbf{B}_H by adaptively learning the diverse restoration process $\mathbf{F}(\cdot)$ in Eq. 11.

1) *HNet*: *HNet* also utilizes the U-Net structure as backbone and RBSE as basic feature extraction block but it is deeper than *TLNet*, which can extract more high-level semantic information to facilitate the detail restoration of the image. We then restore the clean high-frequency details as:

$$\mathbf{B}_H = \text{HNet}(\mathbf{I}_H). \quad (21)$$

where $\text{HNet}(\cdot)$ stands for the trainable *HNet*. In the training phase, we use the L1 loss to optimize $\text{HNet}(\cdot)$:

$$\mathcal{L}_H = \|\mathbf{B}_H, \mathbf{B}_{H_{gt}}\|_1. \quad (22)$$

At the same time, we fine-tune the trained $\text{TLNet}(\cdot)$ and $\text{ALNet}(\cdot)$ by Eq. 20 to maintain the accuracy of the restored \mathbf{B}_L .

C. Deep Similarity-Diversity Network

We add the reconstructed \mathbf{B}_L and \mathbf{B}_H to generate \mathbf{B} . In order to ensure the quality of the final prediction, we use the hybrid loss function [40], L1 loss plus negative SSIM Loss, to jointly optimize all subnetworks in *DSDNet*:

$$\mathcal{L}_{TAH} = \mathcal{L}_1(\mathbf{B}, \mathbf{B}_{gt}) + \mathcal{L}_{SSIM}(\mathbf{B}, \mathbf{B}_{gt}). \quad (23)$$

The overall training strategy of our proposed network is shown in Alg. 1.

D. Discussion

The established methods such as [15], [16], [23], which attempt to handle rain streaks and haze simultaneously, are

Algorithm 1 *DSDNet* training algorithm

Data: Degraded color image \mathbf{I} , ground-truth \mathbf{B}_{gt} , decompose \mathbf{I} into \mathbf{I}_L and \mathbf{I}_H , \mathbf{B}_{gt} into $\mathbf{B}_{L_{gt}}$ and $\mathbf{B}_{H_{gt}}$ by the guided filtering [34], iteration numbers $iters$.

Train *TLNet*:

- 1: Initialize $\text{TLNet}(\cdot)$ using Glorot initialization [41].
- 2: **for** $i=1$ to $iters$ **do**
- 3: Calculate \mathbf{A}_p by Eq. (14) to approximate \mathbf{A}_L .
- 4: Calculate \mathbf{T}_L by Eq. (15).
- 5: Reconstruct \mathbf{B}_{L_c} by Eq. (16).
- 6: Update $\text{TLNet}(\cdot)$ by Eq. (17).
- 7: **end for**
- 8: Saving the params from $\text{TLNet}(\cdot)$ as \mathbf{W}_{TL} .

Train *ALNet* and fine-tune *TLNet*:

- 1: Initialize $\text{ALNet}(\cdot)$ as in [41], $\text{TLNet}(\cdot)$ using \mathbf{W}_{TL} .
- 2: **for** $i=1$ to $iters$ **do**
- 3: Calculate \mathbf{A}_L by Eq. (18).
- 4: Calculate \mathbf{T}_L by Eq. (15).
- 5: Reconstruct \mathbf{B}_L by Eq. (19).
- 6: Update $\text{ALNet}(\cdot)$ and fine-tune $\text{TLNet}(\cdot)$ by Eq. (20).
- 7: **end for**
- 8: Saving the params from $\text{TLNet}(\cdot)$ and $\text{ALNet}(\cdot)$ as \mathbf{W}_{TL} and \mathbf{W}_{AL} respectively.

Train *HNet*, fine-tune *TLNet* and *ALNet*:

- 1: Initialize $\text{HNet}(\cdot)$ as in [41], $\text{TLNet}(\cdot)$ using \mathbf{W}_{TL} , $\text{ALNet}(\cdot)$ using \mathbf{W}_{AL} .
- 2: **for** $i=1$ to $iters$ **do**
- 3: Calculate \mathbf{B}_H by Eq. (21).
- 4: Calculate \mathbf{T}_L by Eq. (15).
- 5: Calculate \mathbf{A}_L by Eq. (18).
- 6: Reconstruct \mathbf{B} by Eq. (13).
- 7: Update $\text{HNet}(\cdot)$ by Eqs. (22) and (23), fine-tune $\text{TLNet}(\cdot)$ and $\text{ALNet}(\cdot)$ by Eqs. (20) and (23).
- 8: **end for**
- 9: Saving the params from $\text{TLNet}(\cdot)$, $\text{ALNet}(\cdot)$, and $\text{HNet}(\cdot)$ as \mathbf{W}_{TL} , \mathbf{W}_{AL} , and \mathbf{W}_H , respectively.

Return: \mathbf{W}_{TL} , \mathbf{W}_{AL} , and \mathbf{W}_H .

mainly designed on the basis of Eq. (1) or its variants. Assuming rain streaks and haze are entangled each with other, they mainly focus on how to describe this entanglement with individual imaging models. However, this entanglement is affected by different factors such as scene depth, environment brightness, and the resolution and motion state of imaging equipment. Therefore, it is difficult to describe such complicated process with a simple set-up.

Different from these methods, we analyze the rain imaging process and the haze imaging process respectively, and integrate them into our proposed similarity-diversity model to guide the network design, thus enable our network not only to effectively deal with rain streaks, haze and their mixing, but also to maintain highly interpretability on the physics level.

TABLE I
QUANTITATIVE COMPARISON ON TWO SYNTHETIC RAINY DATASETS (**RAIN100H** AND **RAIN100L**) AND ONE REAL DATASET (**SPA-DATASET**). THE BEST AND THE SECOND BEST PERFORMANCE VALUES ARE SHOWN IN **HIGHLIGHTED** AND UNDERLINED RESPECTIVELY.

Dataset	Methods	DSC	GMM	DDN	RESCAN	SPANet	JORDER-E	BRN	MSPFN	KPN	DSDNet (Ours)
Rain100H	PSNR	14.01	14.52	25.93	26.84	27.08	<u>29.08</u>	28.63	27.97	27.93	32.82
	SSIM	0.4474	0.5609	0.7932	0.8195	0.8579	<u>0.8716</u>	<u>0.8869</u>	0.8501	0.8632	0.9442
Rain100L	PSNR	25.81	28.05	35.39	36.52	35.46	37.42	<u>37.82</u>	37.77	36.28	41.89
	SSIM	0.8726	0.8978	0.9638	0.9695	0.9642	0.9771	<u>0.9803</u>	0.9760	0.9712	0.9833
SPA-dataset	PSNR	32.15	31.62	35.30	35.90	36.06	38.41	36.10	35.93	<u>40.28</u>	41.15
	SSIM	0.9245	0.9180	0.9449	0.9506	0.9510	0.9626	0.9594	0.9512	<u>0.9784</u>	0.9818

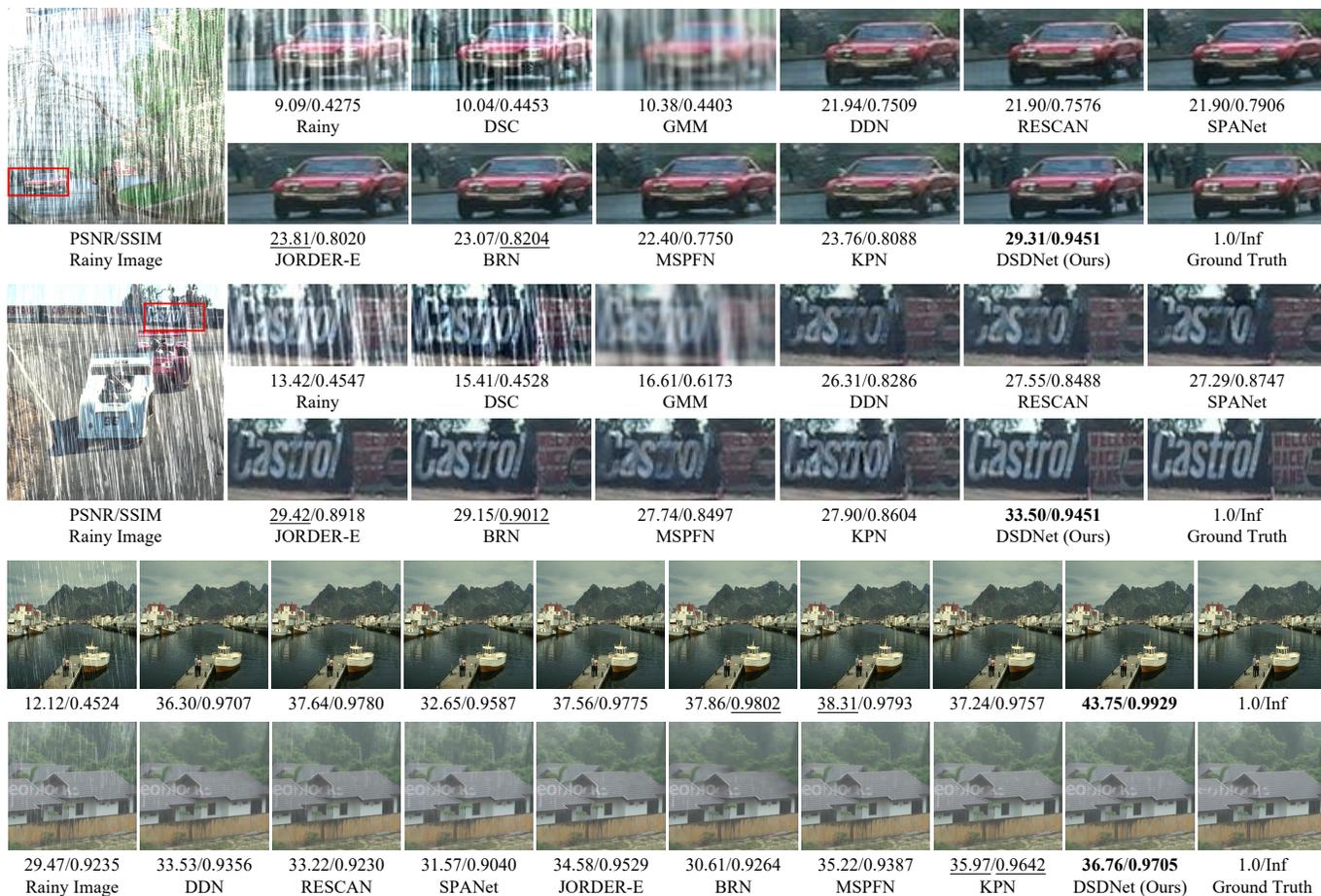


Fig. 3. The qualitative deraining results on four common scenes from **Rain100H** (the first and second scenes), **Rain100L** (the third scene), and **SPA-dataset** (the fourth scene).

V. EXPERIMENTAL WORK

A. Settings

Datasets and Evaluation. We perform experiments on a diverse range of datasets, including synthetic rainy and rain-haze images, real rainy images, heavy-rain images, and rain-haze images, to comprehensively evaluate the rain and haze removal effectiveness of our DSDNet. We first conduct experiments on two synthetic rainy datasets Rain100H and Rain100L [9], and one real dataset SPA-dataset [21]. Both of Rain100H and Rain100L contain 1800 pairs of training samples (synthetic rainy image/ground-truth) and 100 pairs of testing ones. SPA-datasets includes 638492 pairs of train-

ing samples (real rainy image/ground-truth) and 1000 pairs of testing ones. Moreover, we randomly select 4200 hazy images from [42] and superimpose them with the existing rain streaks noise [18] to synthesize a rain-haze dataset called Rain-Haze200 for experiment, which contains 4000 pairs of training samples (synthetic rain-haze image/ground-truth) and 200 pairs of testing ones. On labeled datasets, PSNR and SSIM [43] are used as our performance evaluation metrics. In addition, we collect unlabeled real traffic rainy images of various scenes for qualitative evaluation, from monitored overpasses, crossroads, sidewalk images to driving shots, and use Perceptual Index (PI) [44] as blind quantitative evaluation

TABLE II
QUANTITATIVE COMPARISON ON **RAIN-HAZE200** DATASET. “*” REPRESENT JOINT DEHAZING METHOD MSBDN-DFE [30] FOR COMPARISON.

Methods	DDN	DDN*	RESCAN	RESCAN*	JORDER-E	JORDER-E*	BRN	BRN*	HRGAN	KPN	DSDNet (Ours)
PSNR	23.52	24.84	25.10	25.24	26.67	26.47	26.07	25.51	15.17	<u>27.31</u>	28.05
SSIM	0.7882	0.8124	0.8195	0.8425	0.8806	0.8748	<u>0.8986</u>	0.8935	0.4227	0.8933	0.9030

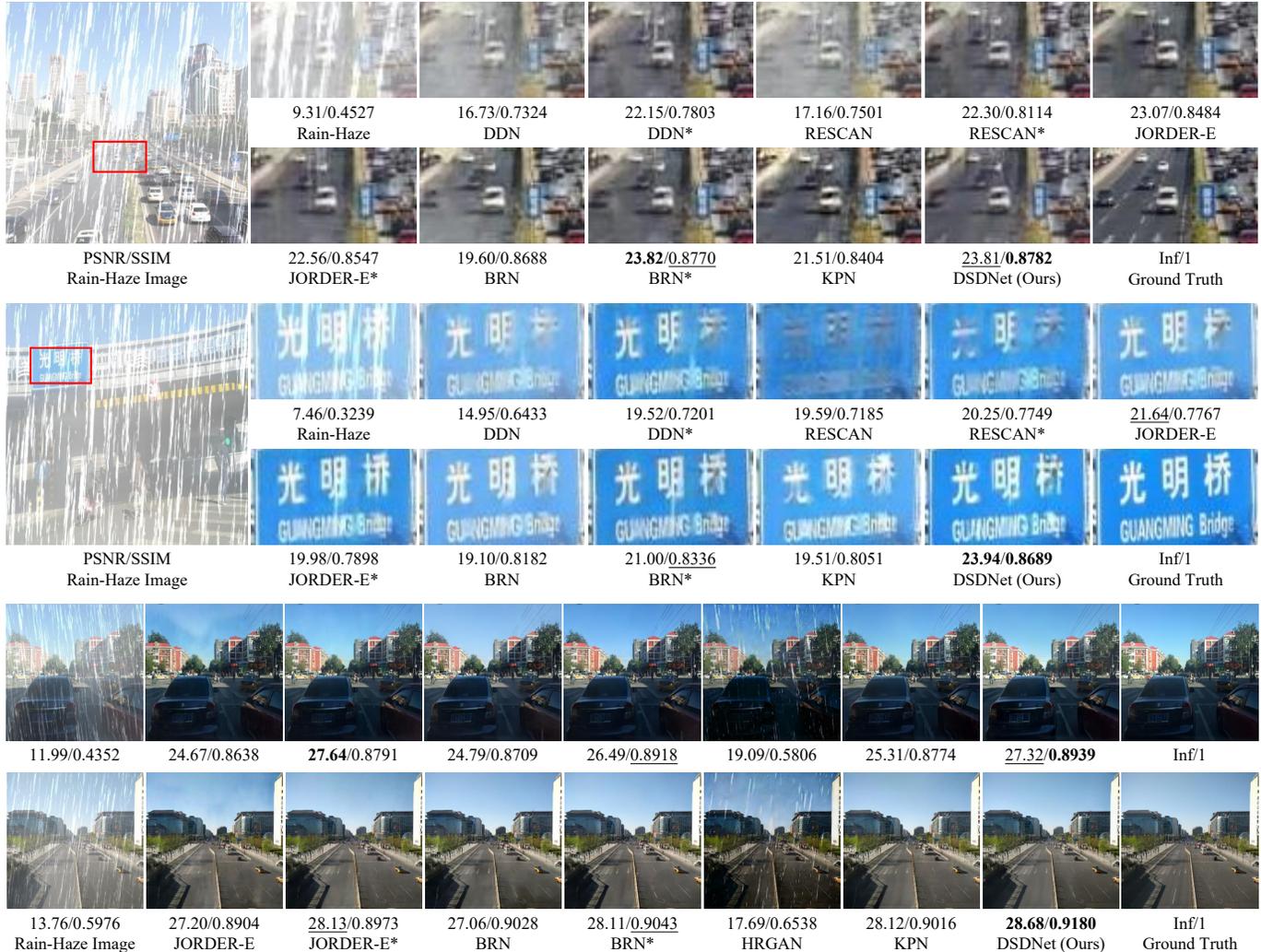


Fig. 4. The qualitative deraining results on four common scenes derived from **Rain-Haze200**.

metrics. We also collect 50 images of traffic transportation and pedestrians on rainy days, respectively, and joint Google API for traffic target recognition to reflect the effectiveness of our method.

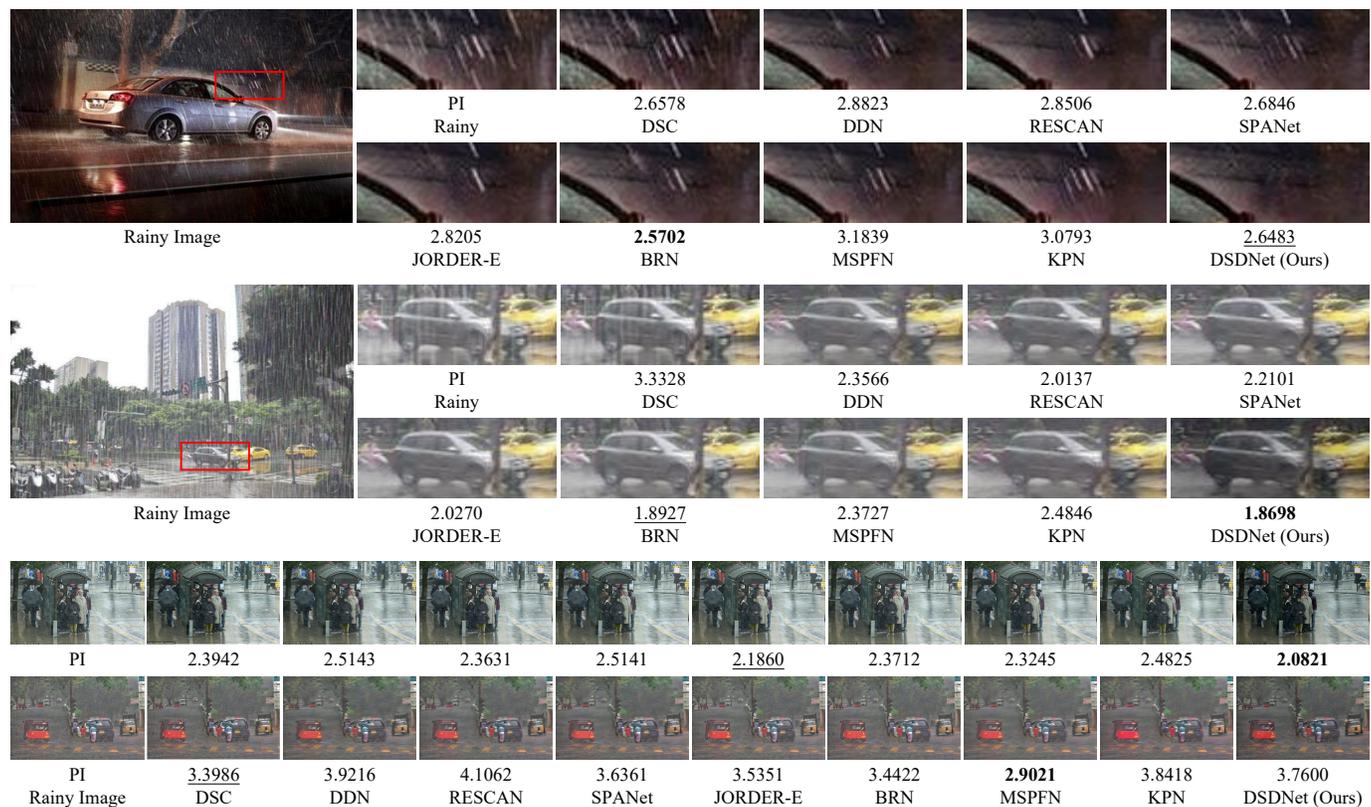
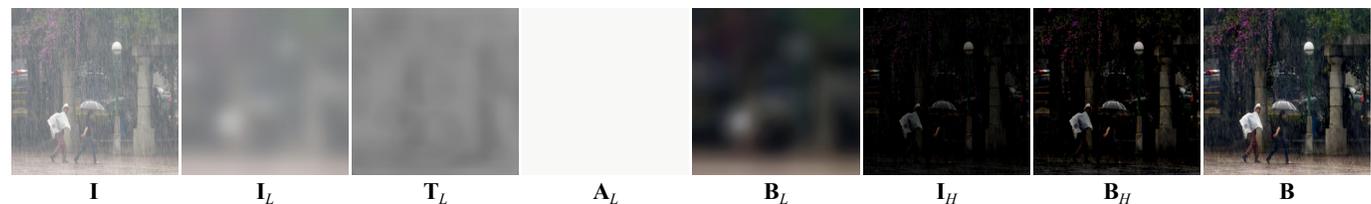
Training and Fine-tuning Details. Our method is performed using the Pytorch framework on NVIDIA GeForce RTX 2080Ti GPU. All the subnetworks are trained using Adam optimizer [45] with the batch size of 6 and all the training datasets are resized to 384×384 as input. The initial learning rate is set as 1×10^{-4} for training and the cosine annealing strategy [46] is adopted with a total of 6 cycles for training with 43200 iterations per cycle. The learning rate in the fine-tuning is 1×10^{-6} without any other adjustment.

Network Details. In the proposed DSDNet, the kernel

radius r and regularization ϵ of guided filter [34] are set as 30 and 1 respectively. Furthermore, we adopt 4×4 convolution with stride of 2 for down sampling and 4×4 transposed convolution with stride of 2 for up sampling. The LReLU [47] with a negative slope of 0.2 is used as activation function in all subnetworks. In addition, TLNet, ALNet and HNet include 7, 4 and 10 RBSE respectively.

B. Experiment on common scene

We compare our proposed DSDNet with a total of ten state-of-the-art image deraining methods, including eight rain streaks removal methods (DSC [5], GMM [7], DDN [19], RESCAN [48], SPANet [21], JORDER-E [9], BRN [49],

Fig. 5. Image deraining results on **unlabeled** real rainy images.Fig. 6. Visualized the intermediate results of our method, where A_L is a three-dimensional matrix with the same channel and spatial dimension as the input rainy image I , and each element value in this matrix is equal to 248.

MSPFN [8]), one heavy rain removal method (HRGAN [15]) and one model-free method (KPN [18]).

TABLE III
COMPARISONS OF MODEL PARAMETERS AND TEST RUNNING TIME. "M"
REPRESENTS MILLION. IMAGE SIZE INDICATES THE RESOLUTION OF
TESTING IMAGE.

Methods	HRGAN	KPN	DSDNet (Ours)
Image Size\Params	40.6M	27.4M	14.2M
256 × 256	0.0502s	0.0076s	0.0243s
384 × 384	0.0911s	0.0161s	0.0438s
512 × 512	0.1418s	0.0271s	0.0748s

Quantitative Comparison. The quantitative results are shown in Tables I and II. Compared with the other state of the art deraining methods, our approach has a significant improvement both in PSNR and SSIM, where the average PSNR by our method boosts 3.70+dB than the second best performance on the Rain100H and Rain100L datasets, and

0.80+dB on SPA-dataset. Moreover, the average PSNR by our DSDNet is also 0.70+dB higher than the model-free method KPN [18] on the Rain-Haze200 dataset.

Qualitative Comparison. The qualitative results are shown in Figs. 3 and 4. We observe that the competing methods tend to under-derain and blur the details of distant background, especially on the extremely rainy image dataset Rain100H. Furthermore, these competing methods can hardly handle image areas with dense haze in the Rain-Haze200 dataset as they fail to incorporate the degradation process of hazy imaging into their physical models. Although combining the dehazing methods enhance the ability of haze removal, it is easy to cause severe artifacts. In contrast, our DSDNet not only effectively removes rain streaks and haze but also successfully restores details and colors in the background. This result is attributed to the guidance of the similarity-diversity physical model, which enables our DSDNet to distinguish rain streaks and haze effectively, and then combine corresponding inverse degradation processes to remove these noises. This advantage improves

TABLE IV
ABLATION STUDIES WITH DIFFERENT SUBNETWORKS, INCLUDING COMPARISONS OF MODEL PARAMETERS, TEST RUNNING TIME, AND DERAINING PERFORMANCE.

	HNet	TLNet	ALNet	Params\Image Size	256 × 256	384 × 384	512 × 512	Dataset	Rain100H	Rain-Haze200
V_1	✓			9.78M	0.0138s	0.0223s	0.0243s	PSNR SSIM	30.66 0.9062	27.06 0.9012
V_2	✓	✓		13.6M	0.0302s	0.0428s	0.0438s	PSNR SSIM	29.22 0.9279	27.61 0.8993
DSDNet	✓	✓	✓	14.2M	0.0532s	0.0722s	0.0748s	PSNR SSIM	32.82 0.9442	28.05 0.9030

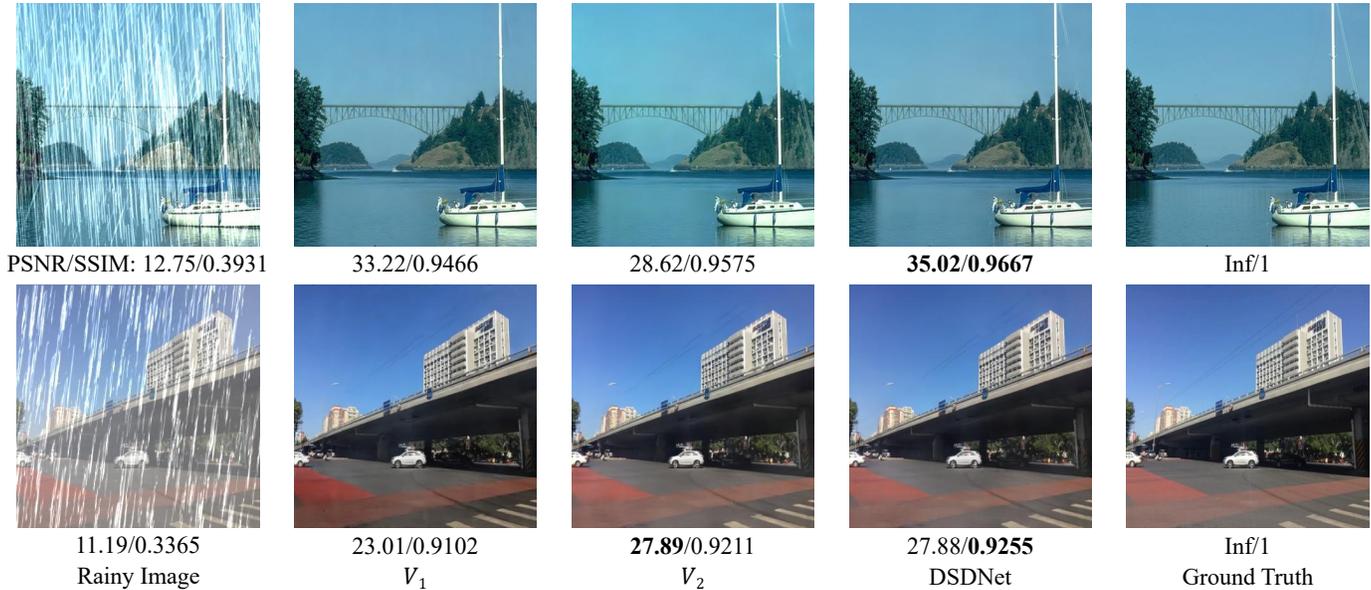


Fig. 7. Image deraining results with different subnetworks on two common scenes.

TABLE V
ABLATION WITH DIFFERENT LOSS ON **RAIN100** DATASET.

	S $\mathcal{L}_T, \mathcal{L}_{TA}$	D \mathcal{L}_H		B_L	B_H	B
V_3	\mathcal{L}_{SSIM}	\mathcal{L}_{SSIM}	PSNR SSIM	33.00 0.9857	36.31 0.9434	30.59 0.9352
V_4	\mathcal{L}_1	\mathcal{L}_1	PSNR SSIM	30.19 0.9785	34.49 0.8925	31.72 0.9390
DSDNet	\mathcal{L}_{SSIM}	\mathcal{L}_1	PSNR SSIM	32.75 0.9856	35.65 0.9171	32.82 0.9442

TABLE VI
PI SCORES OF DERAINING RESULTS ON **REAL TRAFFIC RAINY IMAGES**.

Dataset\Methods	RESCAN	JORDER-E	MSPFN	KPN	DSDNet (Ours)
Rainy Vehicle	3.2632	3.0108	3.4624	3.6602	<u>3.0208</u>
Rainy Pedestrian	2.7987	<u>2.6241</u>	3.0194	3.0739	2.5644

the efficiency of subsequent tasks in intelligent transportation systems, making them more reliable and accurate. Therefore, our approach has significant potential in advancing the field of computer vision for intelligent transportation systems.

We further perform comparisons on the unlabeled real rainy images and visualize the deraining results of four typical traffic

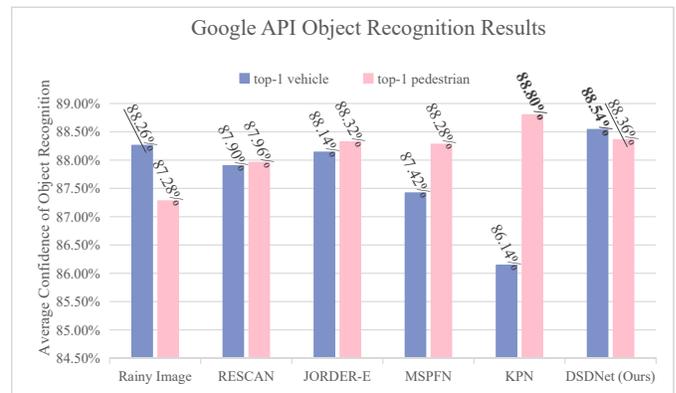


Fig. 8. The average confidence of top-1 vehicle and pedestrian recognition tested on the Google Vision API. We test 50 sets of traffic and pedestrian images in real rainy days respectively, as well as deraining images of ours and four deep learning-based methods. Note that the confidence of mis-recognized or un-recognized target is set to 0.

images in Fig. 5. The corresponding PI [44] is below each visualized result, and a lower PI represents better subjective perceptual quality. It can be seen that our DSDNet can effectively remove rain streaks while maintaining a lower PI as much as possible. **One fact is that rain streaks in unlabeled**

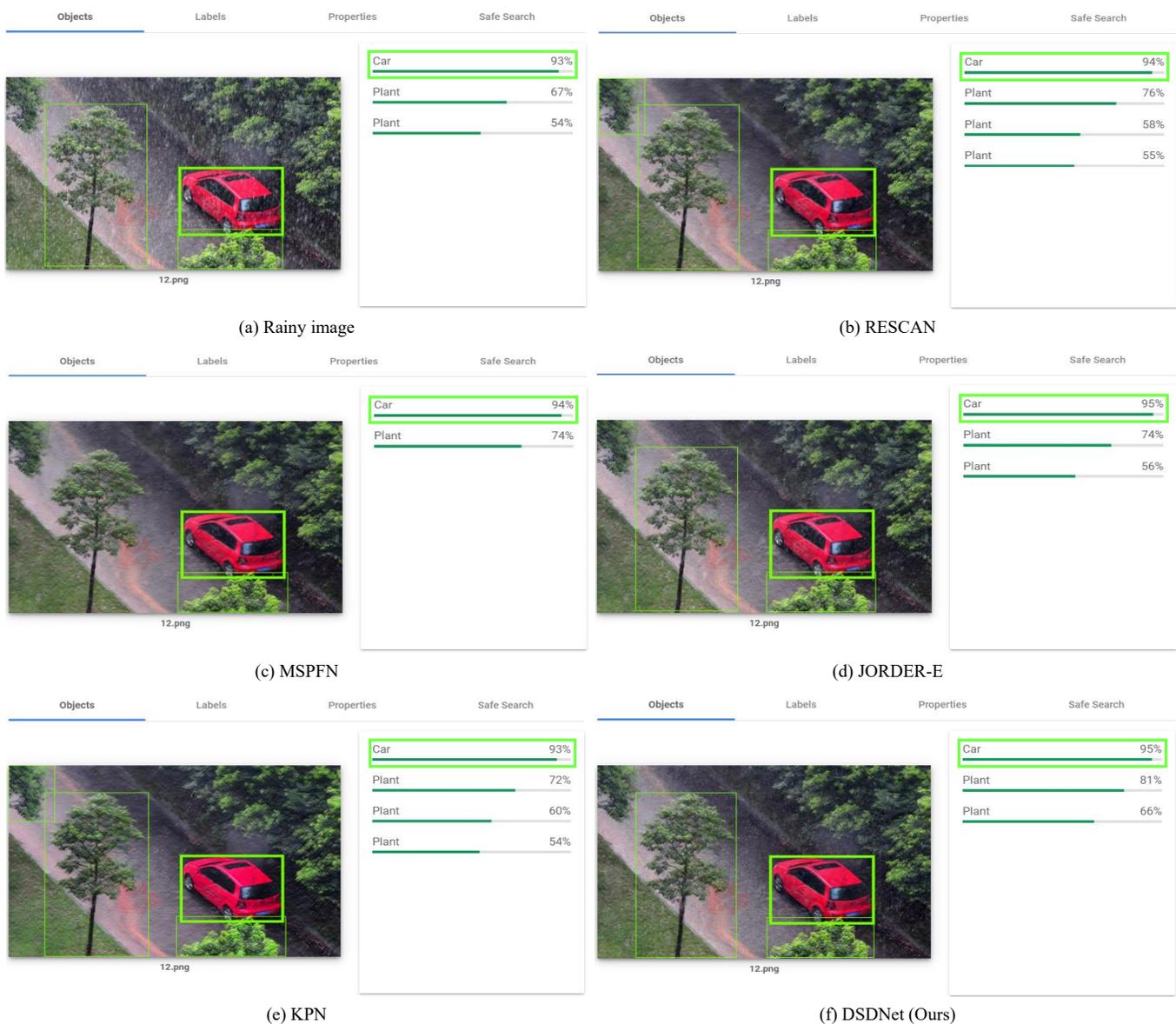


Fig. 9. The recognition results of Case 1 by the Google Vision API. (a) is the recognition result on the rainy image; (b)-(e) represent the recognition result on the deraining image of RESCAN, JORDER-E, MSPFN, and KPN, respectively; (f) is the recognition result on the deraining image of our DSDNet. The left of each result is the visualization of detected objects, and the right is a list contains the types of detected objects and corresponding recognition confidence. The bold box indicates detected vehicle, which have the highest confidence in the corresponding classification.

real rainy images have diverse forms, including heavy rain streaks, mist accumulated by rain streaks, and a mixture of rain and haze. Therefore, it is difficult to accurately describe the degradation process of these rainy images using a single physical model, thus affecting the generalization ability of the competing methods. In contrast, our similarity-diversity model combines multiple physical degradation processes, enabling the proposed DSDNet to adaptively handle various degradation phenomena on these unlabeled real rainy images. Moreover, we display the intermediate results of our method in Fig. 6 to understand what predict in each subnetwork.

Overall, our method can not only remove rain streaks of different scales, motion states and densities, but also well restore natural color and brightness, compared to the competing deraining methods, which indicates the effectiveness of our

method in real scenes.

Parameters and Test Running Time. Furthermore, we compare the parameters and test running time of the proposed network DSDNet against one representative heavy rain removal method HRGAN [15] and one model-free method KPN [18], the details are shown in Table III. In terms of network parameters, our DSDNet contains fewer parameters than HRGAN and KPN. Moreover, our DSDNet takes less running time than HRGAN when removing rain streaks from rainy images with different resolutions. Although the time cost is higher than that of KPN, our DSDNet joint three subnetworks to simulate the inverse process of imaging, therefore it has better physics interpretability. These analyses indicate that our approach is more practical.



Fig. 10. The recognition results of Case 2 by the Google Vision API. (a) is the recognition result on the rainy image; (b)-(e) represent the recognition result on the deraining image of RESCAN, JORDER-E, MSPFN, and KPN, respectively; (f) is the recognition result on the deraining image of our DSDNet. The left of each result is the visualization of detected objects, and the right is a list contains the types of detected objects and corresponding recognition confidence. The bold box indicates detected vehicle, which have the highest confidence in the corresponding classification.

C. Ablation Studies

Network Architectures. We comprehensively analyze each subnetwork to demonstrate the effectiveness of the proposed DSDNet in terms of model parameters, test running time, physics interpretability, and deraining effect.

- V_1 : HNet for deraining.
- V_2 : Joint TLNet and HNet for deraining, among which A_L is approximated by Eq. (14).
- Ours: Joint TLNet, ALNet and HNet for deraining.

The comparison results are shown in Table IV and Fig. 7. It can be found that the deraining results obtained by HNet (V_1) suffer from severe detail distortion, although this network architecture is friendly in terms of test running time. Joint TLNet and HNet (V_2) for deraining can restore better details and have better physics interpretability but tends to generate

color distortion. In contrast, our DSDNet, embedding a subnetwork (ALNet) based on V_2 to predict A_L , restore excellent details and color, indicating the importance of ALNet and the effectiveness of our network.

Loss Function. We also perform ablation experiments on the L1 and the negative SSIM losses to demonstrate the selection of the loss functions in this paper.

- V_3 : Apply the negative SSIM loss both in similarity (S) and diversity (D) reconstruction processes.
- V_4 : Apply the L1 loss both in similarity (S) and diversity (D) reconstruction processes.
- Ours: Apply the negative SSIM loss in similarity (S) process and the L1 loss in diversity (D) reconstruction process.

The results are shown in Table V. We notice that applying the negative SSIM loss both in similarity and diversity recon-

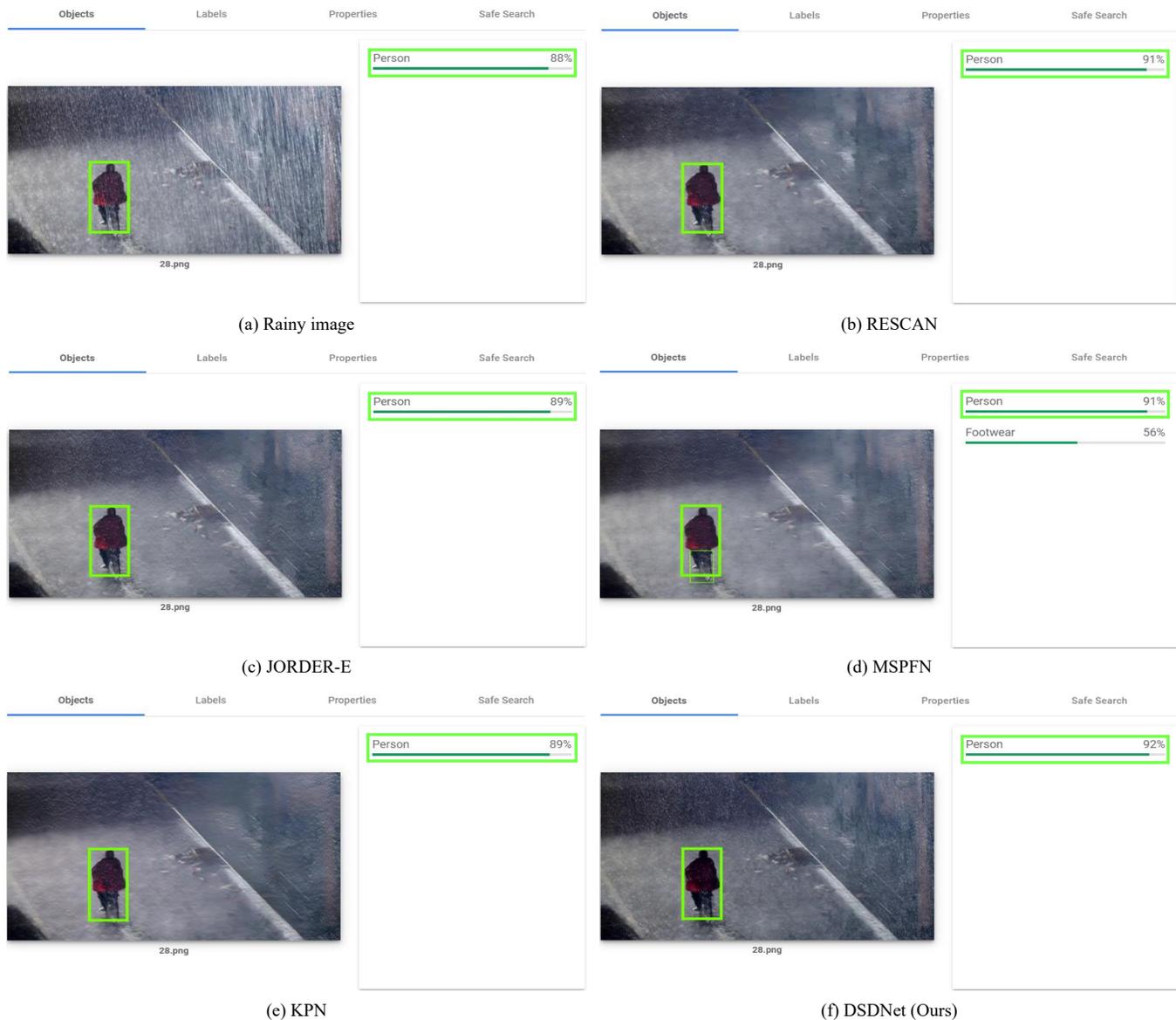


Fig. 11. The recognition results of Case 3 by the Google Vision API. (a) is the recognition result on the rainy image; (b)-(e) represent the recognition result on the deraining image of RESCAN, JORDER-E, MSPFN, and KPN, respectively; (f) is the recognition result on the deraining image of our DSDNet. The left of each result is the visualization of detected objects, and the right is a list contains the types of detected objects and corresponding recognition confidence. The bold box indicates detected pedestrian, which have the highest confidence in the corresponding classification.

struction process (V_3) over-restores the low and high frequency contents, resulting in color distortion of the background. On the contrary, applying the L1 loss both in similarity and diversity reconstruction processes (V_4) under-restores the low and high frequency components, although this better restores the final background than the previous case (V_3). In comparison, the selection in this paper helps us to properly restore the low and high frequency components and the final background, indicating the effectiveness of the loss function that we select.

D. Traffic Application

An important task of Intelligent Transportation System is to recognize cars and pedestrians in rainy images. However, the rain streaks in real scenarios have various shapes and accompanied with diverse motion states. These rain streaks will blur

the details and texture of background and create distortion in image brightness, color and contrast, thus reducing the recognition efficiency of the intelligent system. To demonstrate that our approach can facilitate the execution of outdoor traffic tasks, we utilize the Google Vision API for outdoor target recognition in order to evaluate the rain removal results.

Specifically, we examine 50 sets of traffic and pedestrian images acquired in rainy days [50], respectively, and the deraining images by our DSDNet and four deep learning-based methods. We evaluate PI for these deraining images and record the object recognition results of vehicles and pedestrians, which have the highest confidence values in the corresponding classification on Google Vision API, respectively, namely top-1 vehicle and top-1 pedestrian. The comparison results of PI and object recognition are shown in Table VI and Fig. 8,

respectively. It can be seen that our DSDNet improves the average confidence of Google Vision API for top-1 vehicle and pedestrian recognition on rainy images to 88.54% and 88.36%. The average confidence of Google Vision API for top-1 vehicle and pedestrian recognition on the deraining images of our DSDNet boost 0.28% and 1.08%, respectively, than the performance directly on the rainy images. In contrast, the competing methods tend to impair the average confidence of Google Vision API for top-1 vehicle recognition on rainy images. In addition, our DSDNet achieves better performance in terms of PI than those of competing methods, where the average PI by our DSDNet lower 0.0597 than the second best performance on rainy pedestrian dataset and comparable with the best performance on rainy vehicle dataset. These comparison results illustrate that our DSDNet promotes average recognition confidence by the Google Vision API better than other comparison approaches over top-1 vehicle. Although the improvement over top-1 pedestrian is less than that of KPN, our DSDNet achieves more satisfactory performance in terms of PI, indicating that the restored images by our DSDNet have better perceptual quality. Furthermore, we visualize the deraining images of three real scenarios and the corresponding recognition results, as shown in Figs. 9, 10 and 11. It can be seen that our DSDNet can effectively remove rain and mist from the real traffic rainy image while preserving important object information that plays a crucial role in the recognition by Google Vision API.

VI. CONCLUSION

In order to generalize the applicability of a physics-based deep deraining architecture, we first explored the underlying relationships between the rain and haze models, and then incorporate them into a Similarity-Diversity model to guide the development of the subsequent work. Furthermore, we introduced a novel deep network DSDNet for image deraining based on the similarity-diversity changes. Our network takes into account the underlying correlations between the rain and the haze models, thus can effectively remove rain streaks and their entanglement with haze. Extensive experiments have shown that our method achieved superior performance against the other state of the art deraining methods both on synthetic and real traffic rainy images.

Our future works will focus on improving the efficiency of these physics-based deep deraining networks from following perspectives, designing lightweight feature extraction modules, simplifying the physical process on the single image deraining, and conducting the physical process on the features with low spatial resolution.

ACKNOWLEDGEMENTS

The authors would like to sincerely thank the anonymous reviewers for their insightful comments and constructive suggestions that significantly improved the quality of this paper.

REFERENCES

- [1] G. Cheng, P. Zhou, and J. Han, "Rifd-cnn: Rotation-invariant and fisher discriminative convolutional neural networks for object detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.
- [2] S. M. Silva and C. R. Jung, "License plate detection and recognition in unconstrained scenarios," in *Proceedings of the European Conference on Computer Vision (ECCV)*, September 2018.
- [3] L. Zhang, P. Wang, H. Li, Z. Li, C. Shen, and Y. Zhang, "A robust attentional framework for license plate recognition in the wild," *IEEE Transactions on Intelligent Transportation Systems*, 2020.
- [4] Y. Wang, Z.-P. Bian, Y. Zhou, and L.-P. Chau, "Rethinking and designing a high-performing automatic license plate recognition approach," *IEEE Transactions on Intelligent Transportation Systems*, 2021.
- [5] Y. Luo, Y. Xu, and H. Ji, "Removing rain from a single image via discriminative sparse coding," in *The IEEE International Conference on Computer Vision (ICCV)*, December 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 IEEE International Conference on Computer Vision (ICCV)*, December 2013.
- [7] Y. Li, R. T. Tan, X. Guo, J. Lu, and M. S. Brown, "Rain streak removal using layer priors," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.
- [8] K. Jiang, Z. Wang, P. Yi, C. Chen, B. Huang, Y. Luo, J. Ma, and J. Jiang, "Multi-scale progressive fusion network for single image deraining," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 8346–8355.
- [9] 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 transactions on pattern analysis and machine intelligence*, vol. 42, no. 6, pp. 1377–1393, 2019.
- [10] C. Wang, Y. Wu, Z. Su, and J. Chen, "Joint self-attention and scale-aggregation for self-calibrated deraining network," in *Proceedings of the 28th ACM International Conference on Multimedia*, 2020, pp. 2517–2525.
- [11] H. Wang, Q. Xie, Q. Zhao, and D. Meng, "A model-driven deep neural network for single image rain removal," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 3103–3112.
- [12] C. Wang, X. Xing, Y. Wu, Z. Su, and J. Chen, "Dcsfn: Deep cross-scale fusion network for single image rain removal," in *Proceedings of the 28th ACM International Conference on Multimedia*, 2020, pp. 1643–1651.
- [13] S. W. Zamir, A. Arora, S. Khan, M. Hayat, F. S. Khan, M.-H. Yang, and L. Shao, "Multi-stage progressive image restoration," in *CVPR*, 2021.
- [14] S. Deng, M. Wei, J. Wang, Y. Feng, L. Liang, H. Xie, F. L. Wang, and M. Wang, "Detail-recovery image deraining via context aggregation networks," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 14 560–14 569.
- [15] R. Li, L.-F. Cheong, and R. T. Tan, "Heavy rain image restoration: Integrating physics model and conditional adversarial learning," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 1633–1642.
- [16] X. Hu, C.-W. Fu, L. Zhu, and P.-A. Heng, "Depth-attentional features for single-image rain removal," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 8022–8031.
- [17] H. Zhang, V. Sindagi, and V. M. Patel, "Image de-raining using a conditional generative adversarial network," *IEEE transactions on circuits and systems for video technology*, vol. 30, no. 11, pp. 3943–3956, 2019.
- [18] Q. Guo, J. Sun, F. Juefei-Xu, L. Ma, X. Xie, W. Feng, and Y. Liu, "Efficientderain: Learning pixel-wise dilation filtering for high-efficiency single-image deraining," in *AAAI*, 2021.
- [19] X. Fu, J. Huang, D. Zeng, Y. Huang, X. Ding, and J. Paisley, "Removing rain from single images via a deep detail network," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.
- [20] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.
- [21] T. Wang, X. Yang, K. Xu, S. Chen, Q. Zhang, and R. W. Lau, "Spatial attentive single-image deraining with a high quality real rain dataset," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.
- [22] S. Bell, C. L. Zitnick, K. Bala, and R. Girshick, "Inside-outside net: Detecting objects in context with skip pooling and recurrent neural networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 2874–2883.
- [23] Y. Wang, Y. Song, C. Ma, and B. Zeng, "Rethinking image deraining via rain streaks and vapors," in *European Conference on Computer Vision*. Springer, 2020, pp. 367–382.
- [24] S. G. Narasimhan and S. K. Nayar, "Chromatic framework for vision in bad weather," in *Proceedings IEEE Conference on Computer Vision and*

- 1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
- Pattern Recognition. CVPR 2000 (Cat. No. PR00662)*, vol. 1. IEEE, 2000, pp. 598–605.
- [25] S. G. Narasimhan and S. K. Nayar, “Vision and the atmosphere,” *International journal of computer vision*, vol. 48, no. 3, pp. 233–254, 2002.
- [26] R. T. Tan, “Visibility in bad weather from a single image,” in *2008 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2008, pp. 1–8.
- [27] K. He, J. Sun, and X. Tang, “Single image haze removal using dark channel prior,” *IEEE transactions on pattern analysis and machine intelligence*, vol. 33, no. 12, pp. 2341–2353, 2010.
- [28] Q. Zhu, J. Mai, and L. Shao, “A fast single image haze removal algorithm using color attenuation prior,” *IEEE transactions on image processing*, vol. 24, no. 11, pp. 3522–3533, 2015.
- [29] B. Li, X. Peng, Z. Wang, J. Xu, and D. Feng, “Aod-net: All-in-one dehazing network,” in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 4770–4778.
- [30] H. Dong, J. Pan, L. Xiang, Z. Hu, X. Zhang, F. Wang, and M.-H. Yang, “Multi-scale boosted dehazing network with dense feature fusion,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 2157–2167.
- [31] Y. Romano and M. Elad, “Boosting of image denoising algorithms,” *SIAM Journal on Imaging Sciences*, vol. 8, no. 2, pp. 1187–1219, 2015.
- [32] M. Haris, G. Shakhnarovich, and N. Ukita, “Deep back-projection networks for super-resolution,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 1664–1673.
- [33] J. Hu, L. Shen, and G. Sun, “Squeeze-and-excitation networks,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 7132–7141.
- [34] K. He, J. Sun, and X. Tang, “Guided image filtering,” in *European conference on computer vision*. Springer, 2010, pp. 1–14.
- [35] R. Li, R. T. Tan, and L.-F. Cheong, “Robust optical flow in rainy scenes,” in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 288–304.
- [36] O. Ronneberger, P. Fischer, and T. Brox, “U-net: Convolutional networks for biomedical image segmentation,” in *International Conference on Medical image computing and computer-assisted intervention*. Springer, 2015, pp. 234–241.
- [37] F. Yu and V. Koltun, “Multi-scale context aggregation by dilated convolutions,” in *ICLR*, 2016.
- [38] J.-J. Liu, Q. Hou, M.-M. Cheng, C. Wang, and J. Feng, “Improving convolutional networks with self-calibrated convolutions,” in *IEEE CVPR*, 2020.
- [39] W.-S. Lai, J.-B. Huang, N. Ahuja, and M.-H. Yang, “Deep laplacian pyramid networks for fast and accurate super-resolution,” in *IEEE Conferene on Computer Vision and Pattern Recognition*, 2017.
- [40] H. Zhao, O. Gallo, I. Frosio, and J. Kautz, “Loss functions for image restoration with neural networks,” *IEEE Transactions on computational imaging*, vol. 3, no. 1, pp. 47–57, 2016.
- [41] X. Glorot and Y. Bengio, “Understanding the difficulty of training deep feedforward neural networks,” in *Proceedings of the thirteenth international conference on artificial intelligence and statistics*. JMLR Workshop and Conference Proceedings, 2010, pp. 249–256.
- [42] B. Li, W. Ren, D. Fu, D. Tao, D. Feng, W. Zeng, and Z. Wang, “Benchmarking single-image dehazing and beyond,” *IEEE Transactions on Image Processing*, vol. 28, no. 1, pp. 492–505, 2019.
- [43] A. Hore and D. Ziou, “Image quality metrics: Psnr vs. ssim,” in *2010 20th International Conference on Pattern Recognition*. IEEE, 2010, pp. 2366–2369.
- [44] Y. Blau, R. Mechrez, R. Timofte, T. Michaeli, and L. Zelnik-Manor, “The 2018 pirm challenge on perceptual image super-resolution,” in *Proceedings of the European Conference on Computer Vision (ECCV) Workshops*, 2018, pp. 0–0.
- [45] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” *arXiv preprint arXiv:1412.6980*, 2014.
- [46] T. He, Z. Zhang, H. Zhang, Z. Zhang, J. Xie, and M. Li, “Bag of tricks for image classification with convolutional neural networks,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 558–567.
- [47] A. L. Maas, A. Y. Hannun, and A. Y. Ng, “Rectifier nonlinearities improve neural network acoustic models,” in *Proc. icml*, vol. 30, no. 1, 2013, p. 3.
- [48] X. Li, J. Wu, Z. Lin, H. Liu, and H. Zha, “Recurrent squeeze-and-excitation context aggregation net for single image deraining,” in *Proceedings of the European Conference on Computer Vision (ECCV)*, September 2018.
- [49] D. Ren, W. Shang, P. Zhu, Q. Hu, D. Meng, and W. Zuo, “Single image deraining using bilateral recurrent network,” *IEEE Transactions on Image Processing*, 2020.
- [50] S. Li, I. B. Araujo, W. Ren, Z. Wang, E. K. Tokuda, R. H. Junior, R. Cesar-Junior, J. Zhang, X. Guo, and X. Cao, “Single image deraining: A comprehensive benchmark analysis,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 3838–3847.