Multifocal Attention-Based Cross-Scale Network for Image De-raining

Zheyu Zhang, Yurui Zhu, Xueyang Fu, Zhiwei Xiong, Zheng-Jun Zha, Feng Wu
University of Science and Technology of China
Hefei, Anhui, China

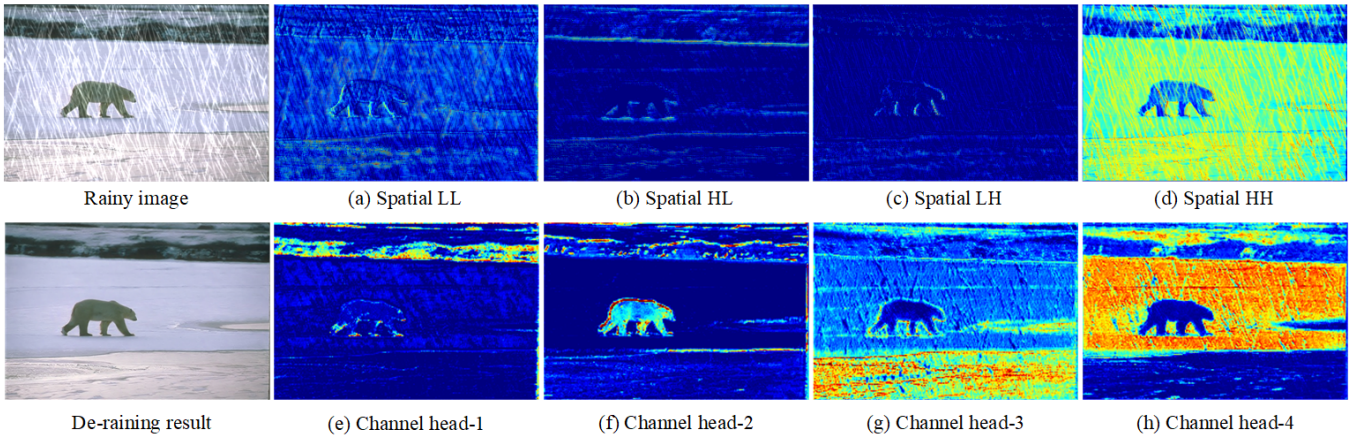


Figure 1: Visualization of learned feature maps in our Cross-Scale Similarity Attention Blocks (CSSABs). (a) (b) (c) (d) are the visualization of each group in the spatial CSSAB. {LL, HL, LH, HH} denote four wavelet bands and focus on different characteristics. LL and HL focus on the smoothed background and high-frequency rain streaks, respectively. HL and LH pay attention to horizontal and vertical characteristics, respectively. (e) (f) (g) (h) are extracted from the channel CSSAB. We incorporate the multi-head mechanism in the channel CSSAB to extract multifocal representations, e.g., bear's body, water and ice fields.

ABSTRACT

Albeit existing deep learning-based image de-raining methods have achieved promising results, most of them only extract single scale features, and neglect the fact that similar rain streaks appear repeatedly across different scales. Therefore, this paper aims to explore the cross-scale cues in a multi-scale fashion. Specifically, we first introduce an adaptive-kernel pyramid to provide effective multi-scale information. Then, we design two cross-scale similarity attention blocks (CSSABs) to search spatial and channel relationships between two scales, respectively. The spatial CSSAB explores the spatial similarity between pixels of cross-scale features, while the channel CSSAB emphasizes the interdependencies among cross-scale features. To further improve the diversity of features, we adopt the wavelet transformation and multi-head mechanism in CSSABs to generate multifocal features which focus on different areas. Finally, based on our CSSABs, we construct an effective multifocal attention-based cross-scale network, which exhaustively utilizes the cross-scale correlations of both rain streaks and background, to achieve image de-raining. Experiments show the superiority of our network over state-of-the-art image de-raining approaches both qualitatively and quantitatively. The source code and pre-trained models are available at https://github.com/zhangzheyu0/Multifocal_derain.

Corresponding author: Xueyang Fu (xyfu@ustc.edu.cn).

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.
MM '21, October 20–24, 2021, Virtual Event, China
© 2021 Association for Computing Machinery.
ACM ISBN 978-1-4503-8651-7/21/10…$15.00
https://doi.org/10.1145/3474085.3475444

CCS CONCEPTS

• **Computing methodologies** → **Reconstruction**.

KEYWORDS

Image De-raining; Cross-scale Similarity Attention; Wavelet Transform; Image Pyramid

ACM Reference Format:
Zheyu Zhang, Yurui Zhu, Xueyang Fu, Zhiwei Xiong, Zheng-Jun Zha, Feng Wu. 2021. Multifocal Attention-Based Cross-Scale Network for Image De-raining. In *Proceedings of the 29th ACM International Conference on Multimedia (MM '21), October 20–24, 2021, Virtual Event, China.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3474085.3475444

1 INTRODUCTION

Rainy days are common weather conditions in daily life. Due to the influence of rain streaks, shooting in rainy weather will cause the image to be blocked and blurred. Removing rain not only improves

visual quality, but also benefits downstream computer vision applications, such as autonomous driving and outdoor surveillance systems.

To mitigate the effect of rain, amounts of image de-raining methods have been proposed. These methods can be roughly divided into two categories: model-based and data-driven approaches. Model-based methods require prior knowledge to constrain the solution space, such as the sparsity and directions of rain streaks [37], and image patches-based similarity prior [11]. However, since these priors are manually designed, they are relatively simple and may not work under complicated rainy conditions. In recent years, data-driven deep learning (DL) methods have dominated the field of low-level vision tasks [4, 7, 16, 24, 25, 30, 32–35]. Thanks to the enormous data sets, DL methods can easily learn the rich features from the data. For the image de-raining task, most DL-based methods focus on designing elegant network architectures [27, 31]. Although these methods have achieved promising performances, there are three non-trivial factors for rain removal task which are neglected. The first factor is the global spatial information mined in the rainy images. Since rainy images usually contain long rain streaks [8], many methods expect to solve this problem by stacked convolution layers or dilated convolution. While their receptive fields are still limited and cannot deal with extreme rainy conditions. NLEDN [8] adopts the non-local module [23] to explore the global spatial information to mitigate the effect at the cost of computing burden [38]. The second factor is the cross-scale similarities between two-scale images [6, 14, 19, 20]. For the same image at two different scales, the structures of rain streaks and background object can be geometrically similar. MSPFN [6] explores the similarities through vanilla convolutions in a multi-scale fashion. Nevertheless, MSPFN can only explore cross-scale similarities in limited local areas, and cannot take global information into account. Last but not least, the global similarity for cross-scale features along the channel dimension has not been fully exploited and utilized. Therefore, it is necessary to explore the cross-scale channel correlation from a global perspective.

In this paper, we take the mentioned factors into account and design two cross-scale similarity attention blocks (CSSABs) to explore and exploit the cross-scale correlation along the spatial and channel dimensions. The spatial CSSAB searches spatial similarities from two-scale features regardless of distance. To alleviate the computation burden, we utilize the spatial pyramid pooling (SPP) [5, 38] to construct this block. The channel CSSAB absorbs multi-scale features and generates enhanced features that aggregate the cross-scale information along the channel dimension. To further improve the diversity of feature representations, we design two multifocal strategies for the spatial and channel CSSABs, respectively. For the spatial CSSAB, we design a Feature Re-Calibrated Block (FRCB) by utilizing wavelet transformation to generate multi-frequency components which can be seemed as focusing on multiple regions of interests. For the channel CSSAB, we utilize the multi-head mechanism to split features into multiple groups and process each group individually. This implicitly pushes each group to focus on content-related information. Finally, based on our proposed CSSABs, we construct a multifocal attention-based cross-scale network to achieve image de-raining. Compared with the existing DL-based techniques, our network is capable of utilizing more suitable complementary cues from different scales, and consistently performs well across various rainy images. Our work has the following main contributions.

- We propose a multifocal attention-based cross-scale network to explore and aggregate cross-scale correlations for the specific image de-raining task.
- We propose a spatial cross-scale similarity attention block to explore the spatial interdependencies. To improve the representation of features, based on the wavelet transformation, we design a feature re-calibrated block to generate multifocal spatial representations.
- We propose a channel cross-scale similarity attention block to model the global correlation along the channel dimension. The multi-head mechanism is utilized to generate multifocal channel representations.
- Experiments show that our proposed network not only achieves better performance than state-of-the-art methods both on conventional measurements, but also benefits the downstream person detection task.

2 RELATED WORK

2.1 Single Image De-raining

Traditional methods explore hand-crafted priors to constrain the ill-posed de-raining problem. For instance, Li *et.al.* [11] use Gaussian mixture models to accommodate multiple orientations and scales of rain streaks. Zhu *et.al.* [37] first estimate rain-dominated patches through rain streaks directions, and then design three image priors based on these patches. However, when the rainy image does not meet the prior assumptions, the de-rained image will be over-smoothed.

Instead, DL-based methods can extract rich features through massive data and thus achieve better de-raining performance [9, 28]. Fu *et al.* [2] adopt a residual learning network to learn rain residue in high frequency image layers. PReNet [15] shows the effectiveness of progressive recurrent network by repeating ConvLSTM layer and ResNet blocks. Li *et al.* [9] adopt a cycle mechanism and utilize a decomposition network to split the rainy image into background layer and clean layer. Then, an extra composition network maps the previous two layers back to the rainy image, which can boost the de-raining quality. NLEDN [8] utilizes non-local attention [23] and dense connection to remove long-range rain streaks. The authors of SPANet [22] propose a spatial attentive module to capture rain streaks directions for guiding the subsequent de-raining process. In DRDNet [1], the authors design two sub-nets in parallel to remove rain and recover details. In RCDNet [21], the authors integrate dictionary learning into a deep network and remove rain with optimization algorithms. Other methods estimate the clean background with extra supervised information, such as rain mask and rain density map in [27], and the rain density in [31].

2.2 Multi- and cross-scale Learning

Intuitively, rain streaks have cross-scale similarities between various scales, while only a few methods take this correlation into account. Yang *et al.* [29] embed cross-scale self-supervision in fractal band learning network to regularize the output, which ensures

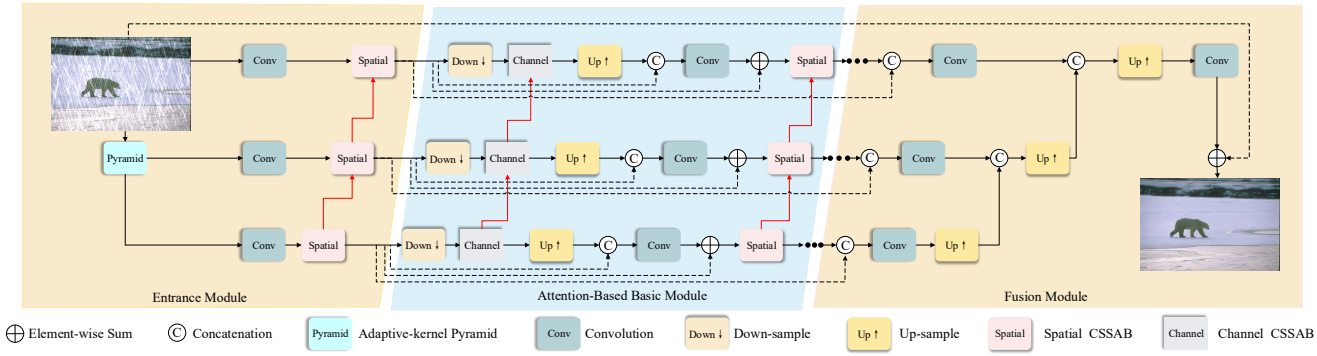


Figure 2: The overall architecture of our proposed network, which removes rain streaks in a coarse-to-fine fashion. The entrance module generates coarse features with the adaptive-kernel pyramid and the spatial CSSAB, while the attention-based basic module takes advantages of cross-scale information in both spatial and channel dimensions. The fusion module aggregates features from each scale and estimates the de-rained image. The red and dashed arrows denote cross-scale interaction and long skip connection, respectively.

features of rain streaks in different scale are equivalent. The self-supervision improves generalization for real-world data. In [6], authors first discover the similarity hidden in the multi-scale rain patterns, and absorb the cross-scale information by the concatenation of multi-scale features and convolution. This strategy is also adopted in [14, 19]. DCSFN [20] fuses the output of multi-scale subnet via gate recurrent unit. Among the mentioned approaches, the fusion strategy of two scales is not proper due to the drawback of the simple convolutional operations, which leads to limited aggregation fields. In contrast, we explicitly aggregate global cross-scale information contained in rain streaks and background. Below we detail our proposed network.

3 METHODOLOGY

In general, the observed rainy image $\mathbf{O}$ can be decomposed into the clean background layer $\mathbf{B}$ and the residue layer $\mathbf{R}$:

$$\mathbf{O} = \mathbf{B} + \mathbf{R}. \tag{1}$$

As shown in Figure 2, our network directly predicts $\mathbf{R}$ instead of $\mathbf{B}$. The reason is that the residual layer $\mathbf{R}$ is sparser than $\mathbf{B}$, which is easier for network convergence.

3.1 Adaptive-kernel Pyramid

To obtain multi-scale information, we first design an adaptive-kernel pyramid as the basic components of our network architecture. The process of pyramid generation is:

$$\mathbf{I}_s = \mathbf{K} * \mathbf{I}_{s-1}, \tag{2}$$

where $\mathbf{I}_s$ is the smaller-scale image, $\mathbf{I}_{s-1}$ is the larger-scale image, and $\mathbf{K}$ denotes image filter. Different from existing methods that use fixed-kernel, e.g., Gaussian kernel in Gaussian image pyramid, to construct the image pyramid, we allow $\mathbf{K}$ to be learnable to improve the network flexibility. Since our kernels adaptively learn the weights from specific training data, they perform better than the fixed-kernel whose parameter is manually designed.

3.2 Spatial Cross-Scale Similarity Attention Block (Spatial CSSAB)

Attention mechanism maps a query with each key and responds to a weighted sum of corresponding values. The mapping function can be seen as a similarity search function, which is widely used in high-level vision tasks. However, existing methods based on similarity mechanism only design attention block in a single scale for image de-raining, which neglect the cross-scale correlations. To mitigate this problem, we introduce two types of CSSABs to explore correlation along the spatial and channel dimensions, respectively.

The spatial CSSAB is designed to model the spatial global correlation in cross-scale interaction. When given the features from the two scales, the keys and values are obtained from the small-scale features, and the queries are obtained from the large-scale features. An intuitive way for aggregating global cross-scale information is to apply the non-local block with the three inputs. Concretely, the keys and values are $\mathbf{K}, \mathbf{V} \in \mathbb{R}^{H_2 W_2 \times C}$ and queries are $\mathbf{Q} \in \mathbb{R}^{H_1 W_1 \times C}$, where $H_1 \times W_1 \times C$ is the size of large-scale feature maps and $H_2 \times W_2 \times C$ is the small-scales. Then the attention output can be obtained from:

$$\mathbf{F}_{out} = \mathbf{F}_{in} + \text{softmax}\left(\mathbf{Q} \otimes \mathbf{K}^T\right) \otimes \mathbf{V}, \tag{3}$$

where $\mathbf{F}_{in}$ and $\mathbf{F}_{out}$ are the large-scale feature maps, $\otimes$ denotes matrix multiplication.

Since features from all channels are transformed in the same fashion, they may focus on one specific signal band [12]. Therefore, we introduce a new Feature Re-Calibrated Block (FRCB) (Figure 3) to generate multifocal features from different signal bands. Specifically, we first utilize the wavelet transformation as guidance to re-calibrate the features and push them to focus on different signal bands in different channels. For simplicity, we only take the queries $\mathbf{Q} \in \mathbb{R}^{H_1 \times W_1 \times C}$ for illustration. We adopt the simple and fast Haar wavelet [13, 26] to decompose the features into four-frequency bands $\{\mathbf{Q}_{LL}, \mathbf{Q}_{LH}, \mathbf{Q}_{HL}, \mathbf{Q}_{HH}\} \in \mathbb{R}^{H_1/2 \times W_1/2 \times C}$. Then, we utilize the pixel-shuffle operation [17] to rearrange features $\mathbb{R}^{H_1/2 \times W_1/2 \times C}$

to $\mathbb{R}^{H_1 \times W_1 \times C/4}$. Finally, we concatenate the features of four groups to get the re-calibrated features. After queries, keys and values are re-calibrated, we search similarities across scales in these four-frequency groups, individually. As shown in Figure 1, the learned features are multifocal, e.g., horizontal or vertical characteristics, and smoothed or high-frequency characteristics.

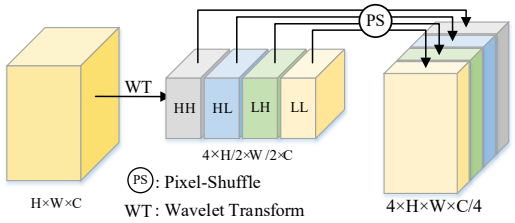


Figure 3: Feature Re-Calibrated Block (FRCB). The FRCB adopts wavelet transformation and pixel-shuffle to generate multifocal features.

To further strengthen the practicality of our spatial CSSAB, we decrease the computational complexity in matrix multiplication $O(H_1W_1H_2W_2C)$ with the spatial pyramid pooling (SPP). In SPP, the feature maps are sent to 4 different adaptive average pooling layers and transformed into 4 feature maps with sizes of 1×1, 3×3, 6×6 and 8×8. Then the four maps are reshaped and concatenated into a fixed-length anchor with a fixed length of S, $S = 1 \times 1 + 3 \times 3 + 6 \times 6 + 8 \times 8 = 110$. Therefore, the SPP not only decreases the computational complexity to $O(H_1W_1SC)$ (where $S \ll H_2W_2$), but also provides a sparse multi-scale representation which is proved useful in many computer vision tasks. With SPP, the similarity attention map $\mathbf{M} \in \mathbb{R}^{4 \times H_1W_1 \times S}$ is calculated as:

$$\mathbf{M} = \text{softmax}\left[\mathbf{Q}_{LL}\mathbf{K}_{LL}^T, \mathbf{Q}_{LH}\mathbf{K}_{LH}^T, \mathbf{Q}_{HL}\mathbf{K}_{HL}^T, \mathbf{Q}_{HH}\mathbf{K}_{HH}^T\right], \quad (4)$$

where $\mathbf{K} \in \mathbb{R}^{S \times C}$ are different-frequency components in keys processed by SPP, $[\cdot]$ denotes concatenation. At last, an enhanced spatial feature $\mathbf{F}_{out}$ is:

$$\mathbf{F}_{out} = \mathbf{F}_{in} + [\mathbf{M}_{LL}\mathbf{V}_{LL}, \mathbf{M}_{LH}\mathbf{V}_{LH}, \mathbf{M}_{HL}\mathbf{V}_{HL}, \mathbf{M}_{HH}\mathbf{V}_{HH},], \quad (5)$$

where $\mathbf{M} \in \mathbb{R}^{H_1W_1 \times C}$ are different-frequency components in similarity attention map, and $\mathbf{V} \in \mathbb{R}^{S \times C}$ are different-frequency components in values processed by SPP.

3.3 Channel Cross-Scale Similarity Attention Block (Channel CSSAB)

To complement the spatial CSSAB, we further design the channel attention to emphasize the global content-related information. Different from existing methods that perform channel attention in a single scale, our channel CSSAB can provide channel interaction across two adjacent scales. Since wavelet transform and pixel-shuffle will change the inherent appearance and order of channels, we discard FRCB in our channel CSSAB. SPP is retained which extracts hierarchical global information in each channel making similarity search more accurate. After processed by SPP, queries from large-scale features are expressed as $\mathbf{Q} \in \mathbb{R}^{S \times C}$. Keys from small-scale features are expressed as $\mathbf{K} \in \mathbb{R}^{S \times C}$. Values from large-scale remain unchanged $\mathbf{V} \in \mathbb{R}^{H_1W_1 \times C}$. The final output from the channel CSSAB is expressed:

$$\mathbf{F}_{out} = \mathbf{F}_{in} + \mathbf{V} \otimes \left(\text{softmax}\left(\mathbf{K}^T \otimes \mathbf{Q}\right)\right), \quad (6)$$

Similar to the spatial CSSAB, the features in the channel CSSAB can also focus on different areas. Inspired by the transformer [18] in nature language processing, we adopt the multi-head mechanism to generate multifocal features implicitly compared with explicit guidance in the spatial CSSAB. We choose four heads as our baseline, and as shown in Figure 1, our channel CSSAB can highlight different focal areas, such as the bear's body, ice fields and water.

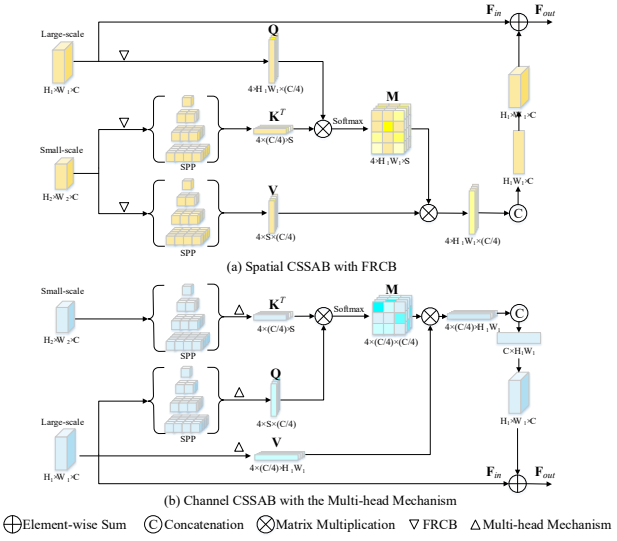


Figure 4: Architectures of CSSABs. (a) is the spatial CSSAB and (b) is the channel CSSAB. We omit the convolution and reshape functions for visualization.

3.4 De-raining Network

As shown in Figure 2, our network consists of entrance module, attention-based basic module and fusion module. In the entrance module, we first apply adaptive-kernel pyramid to generate multi-scale images. Three parallel convolutions are applied to extract shallow features from the image pyramid. Then, we utilize spatial CSSAB for enhancing coarse features. With the spatial CSSAB, the coarse features contain cross-scale and long-distance information which is better for subsequent processing. Specifically, we inject two adjacent scale features into our spatial CSSAB, and enhance large-scale features from small-scale features. Notable, our CSSAB at the bottom of the pyramid only absorbs single-scale features.

In the attention-based basic module, we embed both the spatial and channel CSSAB in each scale. To alleviate the computation burden, we down-sample the spatial dimension of features in each scale before sending them into the channel CSSAB. After fusing

Table 1: Ablation study on the effect of the components in proposed network on *Rain200H* [27].

Models	Pyramid			Cross-scale interaction (CSI)				PSNR↑	SSIM↑
	Fixed		Adaptive	w/o CSI	w/ CSI				
	Gaussian	Downsampling	Adaptive	w/o CSI	Conv	Spatial-CSSAB	Channel-CSSAB		
Model-1	✓					✓	✓	29.32	0.893
Model-2		✓				✓	✓	28.73	0.874
Ours			✓			✓	✓	**30.17**	**0.906**
Model-3			✓	✓				29.12	0.884
Model-4			✓		✓			29.21	0.890
Model-5			✓			✓		30.01	0.900
Model-6			✓				✓	29.97	0.900

the channel information of different scales, we use pixel-shuffle to up-sample the spatial dimension. Long skip connections are also adopted to propagate the gradient. Then the pyramidal features are fed into spatial CSSAB to aggregate the global spatial information between two adjacent scales. We deploy 8 basic modules to explore cross-scale features in a coarse-to-fine fashion. In the fusion module, we up-sample small-scale features and concatenate them with large-scale features. In this way, different-scale features are fully applied to estimate a rain residual layer.

3.5 Loss Function

We train our network using the mean absolute error (MAE), which can preserve edges and details compared with mean square error (MSE) [36]. The MAE loss is:

$$\mathcal{L} = \left\| \hat{\mathbf{B}} - \mathbf{B} \right\|_1, \tag{7}$$

where $\hat{\mathbf{B}}$ and $\mathbf{B}$ are the estimated de-rained image and corresponding ground-truth background image, respectively.

4 EXPERIMENTS

We compare our network with several SOTA methods on three data sets, i.e., *Rain200H* [27], *Rain200L* [27] and *DID* [31]. *Rain200H* and *Rain200L* both have 1,800 paired training images and 200 paired testing images. We also perform ablation study on *Rain200H* to validate the effectiveness of our network. *DID* is synthesized with three rain-density levels and each level has 4,000 paired images. The testing data set contains 1,200 paired images with three levels mixed.

We select one real-world data set with ground truth, i.e., *SPAdata* [22], for evaluating the generalization of our network. This testing data set consists of 1,000 pairs captured from real rain videos. The ground-truth images are generated with temporal priors and human supervision. We also conduct experiments on several Internet data.

4.1 Training Details

Our network is implemented on PyTorch and trained on an Nvidia 1080Ti GPU for 700 epochs. We adopt Adam optimizer with a mini-batch size of 4. We input 128×128 image pairs without any data augmentation. The initial learning rate is 2×10^{-4}, and divided by 2 every 100 epochs.

Table 2: Ablation study on the effect of the head number on *Rain200H* [27].

head number	1	2	4	8	16
PSNR↑	29.50	29.77	**30.17**	29.46	29.64
SSIM↑	0.895	0.899	**0.906**	0.894	0.896

Table 3: Ablation study on the effect of the module number on *Rain200H* [27].

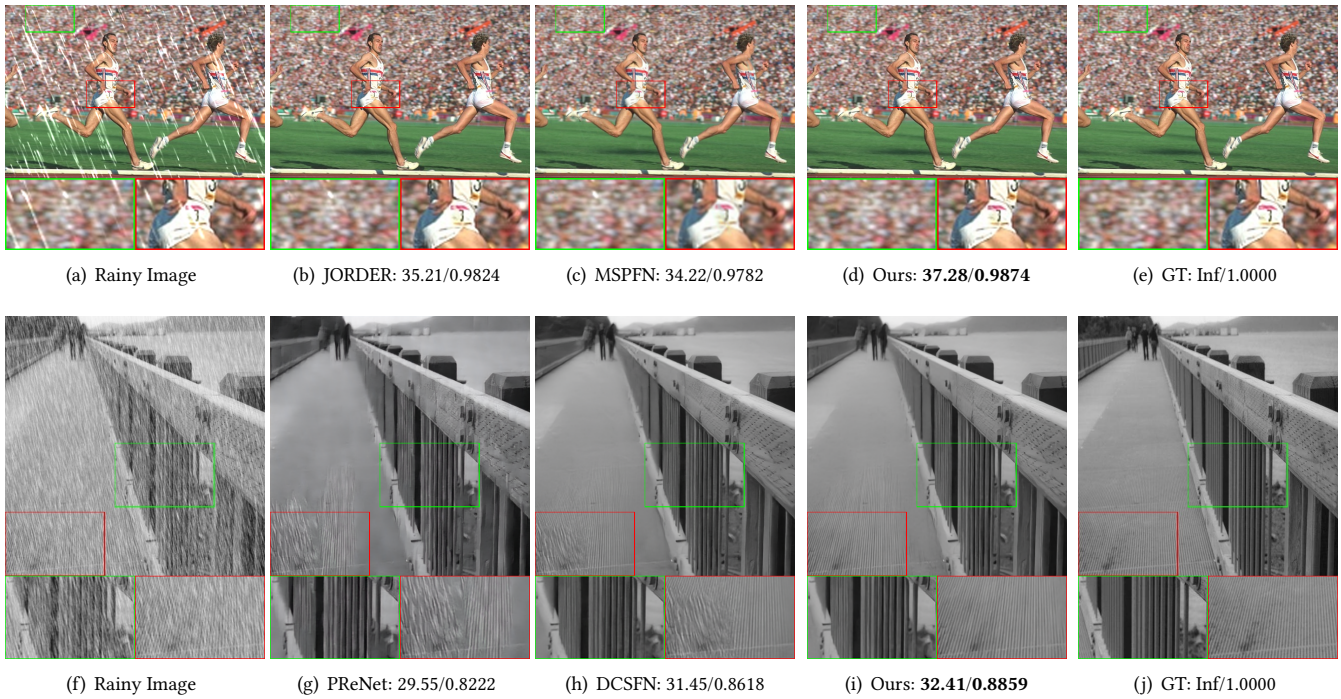
Module number	6	7	8	9	10
PSNR↑	29.34	29.70	**30.17**	29.84	29.87
SSIM↑	0.893	0.897	**0.906**	0.901	0.900

4.2 Ablation Study

4.2.1 Ablation Study on Different Components. We design seven models to test the effect of each component on the *Rain200H* data set in Table 1. We separate the experiments into two categories, i.e. Pyramid and Cross-Scale Interaction (CSI). The pyramid consists of fixed-kernel pyramid and adaptive-kernel pyramid. In the experiments, we set Gaussian kernel and downsampling kernel as our fixed-kernel. For the experiments of CSI, we first design a model without CSI, i.e., a model with three single-scale sub-nets, which deal with each scale image individually. Meanwhile, we compare different ways of CSI within three sub-classes, i.e., convolution (Conv), spatial CSSAB and channel CSSAB.

As shown in Table 1, adaptive-kernel pyramid (Ours) achieves about an improvement of 1.0dB on average PSNR compared with fixed-kernel pyramid (Model-1 and Model-2). CSI can also benefit the rain removal especially with our CSSABs. Compared with the simplest CSI (Model-4), both the spatial CSSAB and channel CSSAB lead to better results. The visual performance is provided in Figure 6.

4.2.2 Ablation Study on hyper parameters. In the channel CSSAB, we introduce the multi-head mechanism to improve the diversity of the generated features. Is the performance better with more head? So in Table 2, we test the influence of head number. The performance is improved as the head number increases but begins



(a) Rainy Image (b) JORDER: 35.21/0.9824 (c) MSPFN: 34.22/0.9782 (d) Ours: **37.28**/**0.9874** (e) GT: Inf/1.0000

(f) Rainy Image (g) PReNet: 29.55/0.8222 (h) DCSFN: 31.45/0.8618 (i) Ours: **32.41**/**0.8859** (j) GT: Inf/1.0000

Figure 5: Comparison results on *Rain200L* [27] and *DID* [31]. PSNR/SSIM on the luminance (Y) channel for reference.

Figure 6: Results on the effect of each component.

to decay when the head number exceeds 4. For example, when *head* = 16, the PSNR decreases 0.53dB compared with *head* = 4.

Table 4: Ablation study on the effect of the kernel number on *Rain200H* [27].

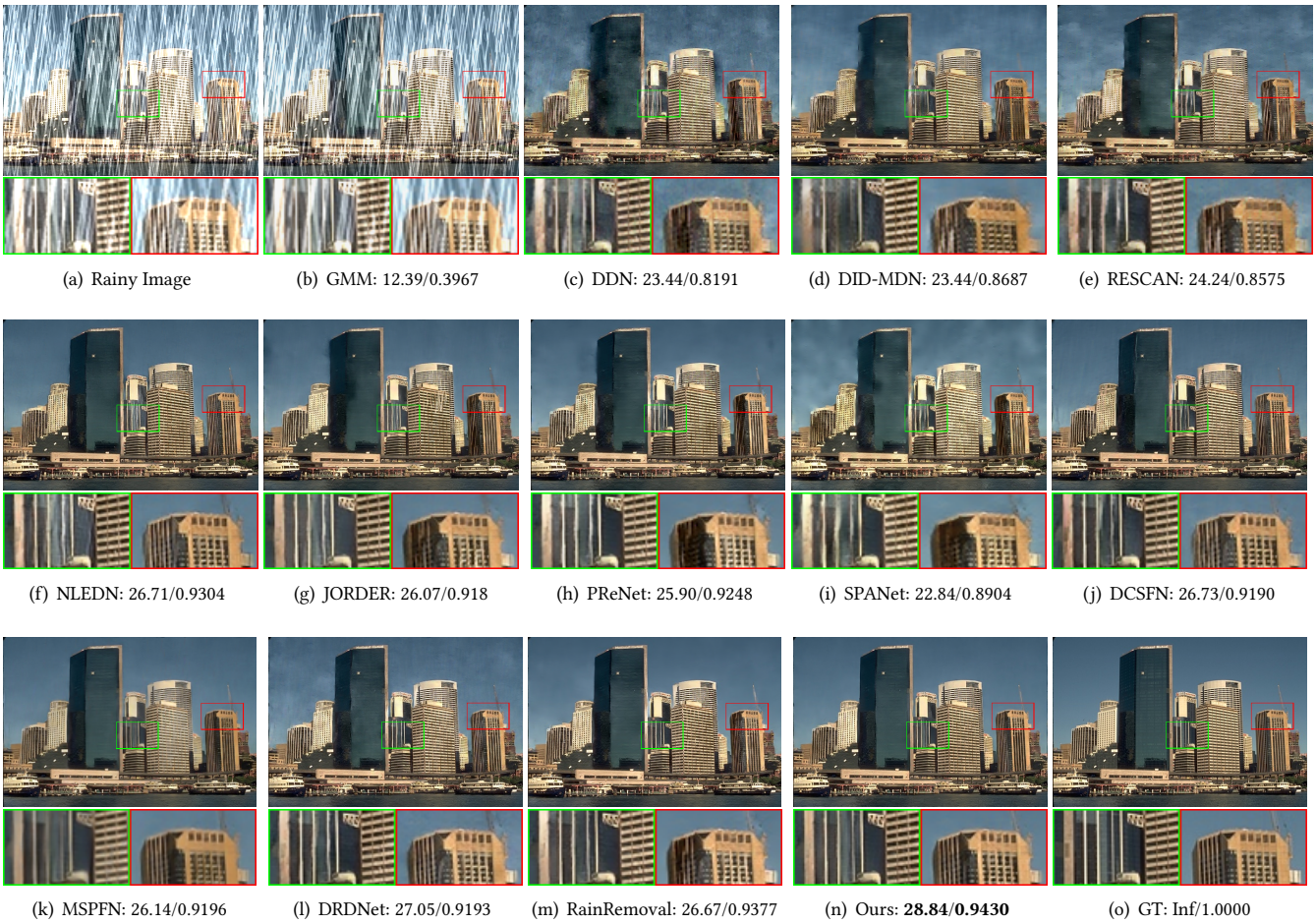
Kernel numbers	32	64	96
PSNR↑	28.82	**30.17**	29.01
SSIM↑	0.882	**0.906**	0.884

Table 5: Comparison results on *SPAdata* [22]. The compared models are trained on *Rain200H* [27] and tested on *SPAdata* for validating generalization on real rainy condition.

	MSPFN	DRDNet	DCSFN	RainRemoval	Ours
PSNR↑	31.01	33.56	34.53	34.45	**35.25**
SSIM↑	0.936	0.948	0.952	0.953	**0.955**

It is that each group only has four feature maps since the total number of feature maps is 64. The enhanced feature in each group is generated on the weighted average of the four features. If one of them is inaccurate, the result is easily affected. While it is difficult to be affected in a group with more features. Therefore, we use 4 heads as the default setting.

We also test the impact of attention-based basic module number. From Table 3, we can observe that the results almost increase with more modules. However, when the training data is not enough, it is harder to converge to a better result with redundant modules. Consequently, we set 8 attention-based basic modules in our network.



(a) Rainy Image (b) GMM: 12.39/0.3967 (c) DDN: 23.44/0.8191 (d) DID-MDN: 23.44/0.8687 (e) RESCAN: 24.24/0.8575

(f) NLEDN: 26.71/0.9304 (g) JORDER: 26.07/0.918 (h) PReNet: 25.90/0.9248 (i) SPANet: 22.84/0.8904 (j) DCSFN: 26.73/0.9190

(k) MSPFN: 26.14/0.9196 (l) DRDNet: 27.05/0.9193 (m) RainRemoval: 26.67/0.9377 (n) Ours: **28.84/0.9430** (o) GT: Inf/1.0000

Figure 7: Comparison results on *Rain200H* [27]. PSNR/SSIM on the luminance (Y) channel for reference.

Table 6: Average PSNR and SSIM results on three benchmark data sets. Boldfaced and underlined indicate top 1^{st} and 2^{nd} rank, respectively.

Data sets	Rain200H PSNR↑/SSIM↑	Rain200L PSNR↑/SSIM↑	DID PSNR↑/SSIM↑
GMM [11] (CVPR'16)	14.50/0.418	28.56/0.870	25.81/0.796
JCAS [3] (ICCV'17)	14.69/0.502	29.79/0.898	25.16/0.822
DDN [2] (CVPR'17)	26.05/0.805	34.68/0.967	30.97/0.885
DID-MDN [31] (CVPR'18)	25.02/0.842	35.40/0.962	29.66/0.901
RESCAN [10] (ECCV'18)	26.75/0.835	36.09/0.970	33.38/0.918
NLEDN [8] (MM'18)	29.85/0.899	39.83/0.984	34.68/0.936
JORDER [27] (TPAMI'19)	29.35/0.890	39.12/0.984	34.06/0.931
PReNet [15] (CVPR'19)	29.04/0.899	37.25/0.979	33.17/0.928
SPANet [22] (CVPR'19)	26.27/0.866	35.79/0.965	33.04/0.928
MSPFN [6] (CVPR'20	28.99/0.888	39.03/0.982	34.51/0.935
DRDNet [1] (CVPR'20)	29.03/0.885	38.84/0.982	33.73/0.923
DCSFN [20] (MM'20)	29.46/0.893	38.86/0.983	34.77/0.938
RainRemoval [14] (MM'20)	29.32/0.900	39.78/0.985	33.48/0.921
Ours	**30.17/0.906**	**40.38/0.987**	**35.00/0.940**

In Table 4, we test the impact of kernel number. The increasing kernel number cannot always improve the performance of de-raining, since the gradient is more difficult to propagate with redundant parameters caused by the increasing kernel number.

4.3 Comparison with SOTA methods

We compare with 2 model-based methods, i.e., GMM [11], and JCAS [3], and 11 DL-based methods, i.e., DDN [2], DID-MDN [31], RESCAN [10], NLEDN [8], JORDER [27], PreNet [15], SPANet [22], MSPFN [6], DRDNet [1], DCSFN [20] and RainRemoval [14]. All the compared methods are tested on the same platform for a fair comparison. The PSNR and SSIM are calculated on the luminance (Y) channel, and the results are shown in Table 6. It is clear that our network achieves the best overall performance. In synthetic data set, we present visual results in Figure 5 and Figure 7. All compared DL-based methods can remove obvious rain streaks, but tend to over-smooth the details of background or introduce artifacts. We validate the generalization on *SPAdata* and the Internet data. Table 5 shows that our method can also work well in real rainy condition.

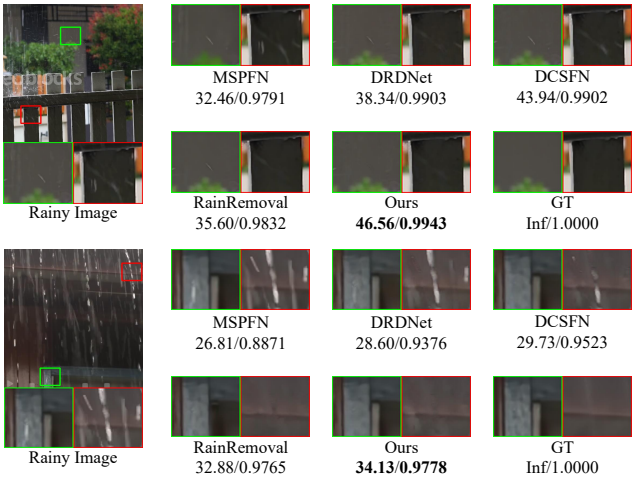


Figure 8: Generalization results on the *SPAdata* [22]. The weights of models are trained on *Rain200H* [27].

Figure 9: Generalization results on the Internet data. The weights of models are trained on *Rain200H* [27].

In Figure 8 and Figure 9, our method removes rain streaks without losing texture of background.

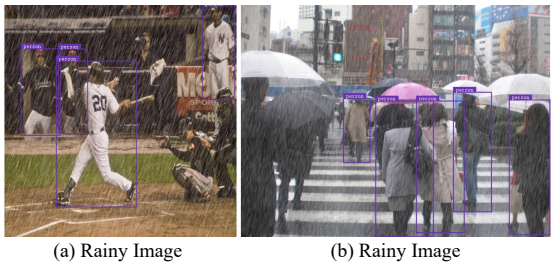
4.4 Application for High-level Tasks

Rain removal task usually plays a pre-processing role for high-level vision tasks. In our experiments, we evaluate our model in person detection which is reported on a commercial detection system [1]. In Figure 10, eight persons are detected compared with five persons in the rainy image. Compared with other rain removal methods, our method can detect more persons with higher probability scores on the likelihood.

5 CONCLUSION

We introduce a multifocal attention-based cross-scale network to explore the cross-scale correlations of rain streaks and background, which is critical for the image de-raining. Our network is constructed based on the adaptive-kernel pyramid and two cross-scale similarity attention blocks. The adaptive-kernel pyramid generates more appropriate hierarchical features for post-processing. The spatial CSSAB explores global cross-scale information along the

[1]Clarifai: https://www.clarifai.com/



(a) Rainy Image (b) Rainy Image

(c) Ours (d) Ours

	MSPFN	DRDNet	DCSFN	Ours
PERSON	0.91	0.91	0.91	0.91
PERSON	0.69	0.68	0.69	**0.74**
PERSON	0.62	0.62	0.62	**0.65**
PERSON	0.62	0.58	0.62	0.62
PERSON	0.55	0.54	0.55	**0.57**
PERSON	0.52	0.52	0.52	**0.53**

(e) Probability Score↑

	MSPFN	DRDNet	DCSFN	Ours
PERSON	0.86	0.86	0.88	**0.89**
PERSON	**0.85**	0.81	0.80	0.79
PERSON	0.74	**0.77**	0.75	**0.77**
PERSON	0.78	**0.79**	0.76	0.75
PERSON	0.64	0.64	0.67	**0.68**
PERSON	0.53	-	**0.63**	0.61
PERSON	-	-	-	**0.54**
PERSON	-	-	**0.52**	**0.52**

(f) Probability Score↑

Figure 10: Pre-processing for person detection.'-' indicates that the method cannot detect this person.

spatial dimension while the channel CSSAB explores the cross-scale correlation in channel-wise. The feature re-calibrated block and the multi-head mechanism in CSSABs push the enhanced features to be multifocal. With the effective components, our network achieves promising de-raining results on several benchmarks.

ACKNOWLEDGMENTS

This work was supported in part by the National Natural Science Foundation of China (NSFC) under Grant 61901433; in part by the University Synergy Innovation Program of Anhui Province under Grant GXXT-2019-025; and in part by the USTC Research Funds of the Double First-Class Initiative under Grant YD2100002003.

REFERENCES

[1] Sen Deng, Mingqiang Wei, Jun Wang, Luming Liang, Haoran Xie, and Meng Wang. 2020. DRD-Net: Detail-recovery Image Deraining via Context Aggregation Networks. In *CVPR*.
[2] Xueyang Fu, Jiabin Huang, Xinghao Ding, Yinghao Liao, and John Paisley. 2017. Clearing the skies: A deep network architecture for single-image rain removal. *IEEE Transactions on Image Processing* 26, 6 (2017), 2944–2956.
[3] Shuhang Gu, Deyu Meng, Wangmeng Zuo, and Lei Zhang. 2017. Joint convolutional analysis and synthesis sparse representation for single image layer separation. In *ICCV*. 1708–1716.

[4] Chunle Guo, Chongyi Li, Jichang Guo, Chen Change Loy, Junhui Hou, Sam Kwong, and Runmin Cong. 2020. Zero-reference deep curve estimation for low-light image enhancement. In *CVPR*. 1780–1789.

[5] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2015. Spatial pyramid pooling in deep convolutional networks for visual recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 37, 9 (2015), 1904–1916.

[6] Kui Jiang, Zhongyuan Wang, Peng Yi, Chen Chen, Baojin Huang, Yimin Luo, Jiayi Ma, and Junjun Jiang. 2020. Multi-scale progressive fusion network for single image deraining. In *CVPR*. 8346–8355.

[7] Chongyi Li, Huazhu Fu, Runmin Cong, Zechao Li, and Qianqian Xu. 2020. NuI-Go: Recursive Non-Local Encoder-Decoder Network for Retinal Image Non-Uniform Illumination Removal. In *ACM International Conference on Multimedia*. 1478–1487.

[8] Guanbin Li, Xiang He, Wei Zhang, Huiyou Chang, Le Dong, and Liang Lin. 2018. Non-locally enhanced encoder-decoder network for single image de-raining. In *ACM International Conference on Multimedia*. 1056–1064.

[9] Siyuan Li, Wenqi Ren, Jiawan Zhang, Jinke Yu, and Xiaojie Guo. 2019. Single image rain removal via a deep decomposition–composition network. *Computer Vision and Image Understanding* 186 (2019), 48–57.

[10] Xia Li, Jianlong Wu, Zhouchen Lin, Hong Liu, and Hongbin Zha. 2018. Recurrent squeeze-and-excitation context aggregation net for single image deraining. In *ECCV*. 254–269.

[11] Yu Li, Robby T Tan, Xiaojie Guo, Jiangbo Lu, and Michael S Brown. 2016. Rain streak removal using layer priors. In *CVPR*. 2736–2744.

[12] Jiang-Jiang Liu, Qibin Hou, Ming-Ming Cheng, Changhu Wang, and Jiashi Feng. 2020. Improving convolutional networks with self-calibrated convolutions. In *CVPR*. 10096–10105.

[13] Stephane G Mallat. 1989. A theory for multiresolution signal decomposition: the wavelet representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 11, 7 (1989), 674–693.

[14] Bo Pang, Deming Zhai, Junjun Jiang, and Xianming Liu. 2020. Single Image Deraining via Scale-space Invariant Attention Neural Network. In *ACM International Conference on Multimedia*. 375–383.

[15] Dongwei Ren, Wangmeng Zuo, Qinghua Hu, Pengfei Zhu, and Deyu Meng. 2019. Progressive image deraining networks: A better and simpler baseline. In *CVPR*. 3937–3946.

[16] Wenqi Ren, Jinshan Pan, Hua Zhang, Xiaochun Cao, and Ming-Hsuan Yang. 2020. Single image dehazing via multi-scale convolutional neural networks with holistic edges. *International Journal of Computer Vision* 128, 1 (2020), 240–259.

[17] Wenzhe Shi, Jose Caballero, Ferenc Huszár, Johannes Totz, Andrew P Aitken, Rob Bishop, Daniel Rueckert, and Zehan Wang. 2016. Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network. In *CVPR*. 1874–1883.

[18] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *NeurIPS*.

[19] Cong Wang, Yutong Wu, Zhixun Su, and Junyang Chen. 2020. Joint Self-Attention and Scale-Aggregation for Self-Calibrated Deraining Network. In *ACM International Conference on Multimedia*. 2517–2525.

[20] Cong Wang, Xiaoying Xing, Yutong Wu, Zhixun Su, and Junyang Chen. 2020. DCSFN: Deep Cross-scale Fusion Network for Single Image Rain Removal. In *ACM International Conference on Multimedia*. 1643–1651.

[21] Hong Wang, Qi Xie, Qian Zhao, and Deyu Meng. 2020. A model-driven deep neural network for single image rain removal. In *CVPR*. 3103–3112.

[22] Tianyu Wang, Xin Yang, Ke Xu, Shaozhe Chen, Qiang Zhang, and Rynson WH Lau. 2019. Spatial attentive single-image deraining with a high quality real rain dataset. In *CVPR*. 12270–12279.

[23] Xiaolong Wang, Ross Girshick, Abhinav Gupta, and Kaiming He. 2018. Non-local neural networks. In *CVPR*. 7794–7803.

[24] Zeyu Xiao, Xueyang Fu, Jie Huang, Zhen Cheng, and Zhiwei Xiong. 2021. Space-time distillation for video super-resolution. In *CVPR*.

[25] Zeyu Xiao, Zhiwei Xiong, Xueyang Fu, Dong Liu, and Zheng-Jun Zha. 2020. Space-Time Video Super-Resolution Using Temporal Profiles. In *ACM MM*.

[26] Wenhan Yang, Jiaying Liu, Shuai Yang, and Zongming Guo. 2019. Scale-free single image deraining via visibility-enhanced recurrent wavelet learning. *IEEE Transactions on Image Processing* 28, 6 (2019), 2948–2961.

[27] Wenhan Yang, Robby T Tan, Jiashi Feng, Zongming Guo, Shuicheng Yan, and Jiaying Liu. 2019. Joint rain detection and removal from a single image with contextualized deep networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 42, 6 (2019), 1377–1393.

[28] Wenhan Yang, Robby T Tan, Shiqi Wang, Yuming Fang, and Jiaying Liu. 2020. Single image deraining: From model-based to data-driven and beyond. *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2020).

[29] Wenhan Yang, Shiqi Wang, Dejia Xu, Xiaodong Wang, and Jiaying Liu. 2020. Towards scale-free rain streak removal via self-supervised fractal band learning. In *AAAI*, Vol. 34. 12629–12636.

[30] He Zhang and Vishal M Patel. 2018. Densely connected pyramid dehazing network. In *CVPR*. 3194–3203.

[31] He Zhang and Vishal M Patel. 2018. Density-aware single image de-raining using a multi-stream dense network. In *CVPR*. 695–704.

[32] He Zhang, Vishwanath Sindagi, and Vishal M Patel. 2019. Image de-raining using a conditional generative adversarial network. *IEEE Transactions on Circuits and Systems for Video Technology* 30, 11 (2019), 3943–3956.

[33] Yulun Zhang, Kunpeng Li, Kai Li, Lichen Wang, Bineng Zhong, and Yun Fu. 2018. Image super-resolution using very deep residual channel attention networks. In *ECCV*. 286–301.

[34] Yulun Zhang, Yapeng Tian, Yu Kong, Bineng Zhong, and Yun Fu. 2018. Residual dense network for image super-resolution. In *CVPR*. 2472–2481.

[35] Yulun Zhang, Yapeng Tian, Yu Kong, Bineng Zhong, and Yun Fu. 2020. Residual dense network for image restoration. *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2020).

[36] Hang Zhao, Orazio Gallo, Iuri Frosio, and Jan Kautz. 2016. Loss functions for image restoration with neural networks. *IEEE Transactions on Computational Imaging* 3, 1 (2016), 47–57.

[37] Lei Zhu, Chi-Wing Fu, Dani Lischinski, and Pheng-Ann Heng. 2017. Joint bi-layer optimization for single-image rain streak removal. In *ICCV*. 2526–2534.

[38] Zhen Zhu, Mengde Xu, Song Bai, Tengteng Huang, and Xiang Bai. 2019. Asymmetric non-local neural networks for semantic segmentation. In *ICCV*. 593–602.