

SINGLE IMAGE DERAINING USING A RECURRENT MULTI-SCALE AGGREGATION AND ENHANCEMENT NETWORK

Youzhao Yang, Hong Lu

School of Computer Science, Fudan University
{yzyang17, honglu}@fudan.edu.cn

ABSTRACT

Single image deraining is an ill-posed inverse problem due to the presence of non-uniform rain shapes, directions, and densities in images. In this paper, we propose a novel progressive single image deraining method named **R**ecurrent **M**ulti-scale **A**ggregation and **E**nhanment **N**etwork (ReMAEN). Differing from previous methods, ReMAEN contains a symmetric structure where recurrent blocks with shared channel attention are applied to select useful information collaboratively and remove rain streaks stage by stage. In ReMAEN, a **M**ulti-scale **A**ggregation and **E**nhanment **B**lock (MAEB) is constructed to detect multi-scale rain details. Moreover, to better leverage the rain details from rainy images, ReMAEN enables a symmetric skipping connection from low level to high level. Extensive experiments on synthetic and real-world datasets demonstrate that our method outperforms the state-of-the-art methods tremendously. Furthermore, ablation studies are conducted to show the improvements obtained by each module in ReMAEN. The source code is available at <https://github.com/nnyui/ReMAEN>.

Index Terms— Single Image Deraining, Dilated Convolution, Recurrent Block, Shared Channel Attention

1. INTRODUCTION

Generally, images with rain streaks will lead many applications (e.g. video surveillance, self-driving car, object tracking, and visual navigation) to fail to work since rain streaks will degrade the qualities of images. Thus it is important to remove rain streaks from rainy images. However, in many cases, the shapes, directions, and densities of rain streaks in rainy images are non-uniform. How to model the rain streaks well becomes meaningful. To address this problem, we propose a novel method to efficiently detect multi-scale rain details and remove rain streaks progressively.

In the past few years, researchers mainly focus on employing Gaussian mixture model [1], kernel [2, 3], low rank approximation [4, 5], representation learning [6], and dictionary learning [7, 8, 9] to address this problem. Most methods mentioned above require image priors to recover the underlying background while [10] introduces an automatic method

to locate rain dominated regions from rainy images and estimates the dominant directions of the rain streaks. By incorporating the joint bi-layer optimization model, [10] can iteratively separate rain streaks and background. Moreover, Deng et al. consider the intrinsic directional and structural information of rain streaks and formulate a global sparse model to efficiently remove rain streaks from rainy images [11]. Though these methods can improve the visibility of rainy images to some extent, they fail to fit the ill-posed inverse deraining procedure well. Hence, undesirable rain streaks still remain in rainy images or the intensities of background will be changed significantly.

Recently, learning based methods have achieved great success in single image deraining. [12] constructs a three-layer convolutional network to model rain streaks. The method achieves relatively low performances since it fails to model the complex distribution of rain streaks well. To efficiently exploit the rain details, [13] separates high-frequency features from rainy images using guided filter while Zhang et al. adopt an adversarial learning to enable network to learn more rain details [14]. Differently, [15] introduces a decomposition-composition network to split a rainy image into a rain-free layer rain streak layer using a decomposition net while composition net is applied to recover the rainy image from rain streak layer to make sure the preservation of rain details. And [16] proposes a dual convolutional network to learn structures and details of background. All these learning based methods above consider rain removal as a simple separation problem. However, rainy images contain complex rain details like different rain shapes, directions, and densities. Thus [17] employs a Laplacian pyramid to detect multi-scale rain details. By incorporating rain densities, [18] proposes a multi-stream densely connected deraining network to better characterize rain details with different scales. And Li et al. construct a non-local enhanced encoder-decoder network framework [19] to learn increasingly abstract feature representation while preserving the image details. Instead of taking single stage rain removal scheme, [20, 21, 22] employ a recurrent structure to make aware of multi-scale rain details and remove rain streaks stage by stage. Nonetheless, [20, 21] neglect the correlations in neighboring stages. As for [22], the direct flow of low-level rain details from hidden statuses

of previous recurrent block to current recurrent block leads to the loss of rain details in high level.

Based on the motivation, a recurrent multi-scale aggregation and enhancement network (ReMAEN) is proposed for single image deraining. We apply recurrent blocks to construct a symmetric network structure and progressively remove rain streaks from rainy images by incorporating the correlations between the neighboring stages. In each recurrent block, a shared channel-attention block is applied in hidden statuses to select useful information collaboratively. Besides, a multi-scale aggregation and enhancement block (MAEB) is designed to efficiently characterize rain details with different scales. Furthermore, symmetric skipping connections are enabled in ReMAEN to keep the flow of rain details from low level to high level. During training, an edge loss is employed to preserve the texture of background. In summary, this paper makes the following contributions:

- We propose a novel single image deraining method named recurrent multi-scale aggregation and enhancement network (ReMAEN) to remove rain streaks stage by stage.
- In ReMAEN, a shared channel-attention block is applied to collaboratively select useful information from hidden statuses. And a multi-scale aggregation and enhancement block (MAEB) is designed to efficiently characterize multi-scale rain details. Moreover, ReMAEN allows symmetric skipping connection between low level and high level to make sure the flow of rain details.
- In training procedure, an edge loss is applied to preserve the texture of background. To the best of our knowledge, this is the first paper to consider the edge preservation using edge loss in deraining task.
- The proposed method obtains the superior performances compared with the state-of-the-art methods on various datasets.

2. PROPOSED METHOD

Mathematically, a rainy image I can be considered as a linear combination of background B and rain streak layer R as follows:

$$I = B + R \quad (1)$$

However, raindrops in the air have various appearances which lead to the non-uniform rain shapes, directions, and densities. Thus a simple rain streak layer fails to capture the complex distribution of rain streaks. In this case, we propose a recurrent multi-scale aggregation and enhancement network (ReMAEN) to estimate multiple rain streak layers progressively. The structure of ReMAEN is illustrated in Fig. 1.

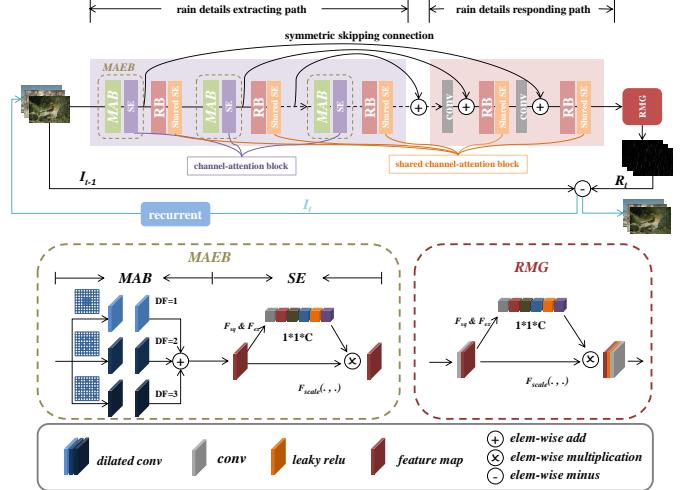


Fig. 1: The whole structure of recurrent multi-scale aggregation and enhancement network (ReMAEN).

2.1. Network Structure

By decomposing a rain streak layer into multiple layers, we can formulate it as follows:

$$\begin{aligned} R_t &= f_{ReMAEN}(I_{t-1}) \\ I_t &= I_{t-1} - R_t \\ 1 \leq t &\leq T \end{aligned} \quad (2)$$

where R_t indicates rain streak layer estimated in t -th stage, I_t indicates deraining result obtained in t -th stage and T indicates the maximum number of recurrent stages. Note that I_0 indicates the input rainy image.

As shown in Eq. 2, ReMAEN removes rain streaks stage by stage by incorporating information from previous stage. Taking I_{t-1} as input, ReMAEN estimates the rain streak layer R_t between I_t in current stage and I_{t-1} in previous stage.

Specifically, ReMAEN is a symmetric structure with two paths: a rain details extracting path and a rain details responding path. In rain details extracting path, we firstly apply a multi-scale aggregation and enhancement block (MAEB) to extract multi-scale rain details. Then a recurrent block (RB) with memory unit is adopted following the MAEB to keep useful information and well guide the later stages. By stacking multiple MAEBs and RBs iteratively, various rain details in different hierarchies can be efficiently obtained. As for rain details responding path, features of rain details extracted by MAEB in rain details extracting path have access to the corresponding symmetric layers in rain details responding path through symmetric skipping connection. Subsequently, a recurrent block (RB) is used to keep useful information and generate more efficient features for the following layers. Remarkably, a channel-attention block is applied behind each RB of the two paths. All the channel-attention blocks share the same parameters which collaboratively help RBs to select

useful information. Finally, features obtained by rain details responding path are fed into a rainy map generator (RMG) to estimate the rain streak layer efficiently.

2.2. Multi-Scale Aggregation and Enhancement Block

The structure of multi-scale aggregation and enhancement block (MAEB) is shown in Fig. 1. As can be observed, MAEB contains two basic blocks. The first block is multi-scale aggregation block (MAB). There are P paths in MAB. Each path consists of two layers. Each layer contains a convolution operation and an activation function. To preserve the rain details with negative feature values, leaky relu is adopted as the activation function. As for convolution operation in each path, dilated convolution with different dilated factors is used to focus on multi-scale rain details. All the convolution operations in each path share the same parameters and keep the same dimensions between input and output. Furthermore, dilated factor in p -th path is set as p . We add all the output features extracted by each path to obtain rain details with different scales.

To better model the inter dependencies of the multi-scale rain details, a squeeze-and-excitation (SE) [23] block is applied. In squeeze step, a global information embedding operation is used to exploit contextual information in the whole feature maps. Then the excitation operation makes full use of the aggregated information in squeeze step and captures feature dependencies efficiently.

2.3. Shared Channel-Attention Block

Though hidden statuses obtained by different recurrent blocks contain different rain details, they are highly correlative. The input of current recurrent block comes from the output of previous recurrent blocks with or without transformation. Thus it is meaningful to model the inter-dependencies among the recurrent blocks. Based on the motivation, we apply a shared channel-attention block following all the recurrent blocks to capture the correlations. Specifically, a shared squeeze-and-excitation (SE) block is used. As shown in Fig. 1, all SE blocks behind the recurrent blocks share the same parameters and the hidden statuses are collaborative to guide the shared SE block to efficiently model dependencies among the whole recurrent blocks. And shared SE block helps to select useful information of hidden statuses.

2.4. Loss Function

Generally, the output of the ReMAEN in the last stage should be equal to the clean images in pixel level. Thus a pixel-wise mean square error (MSE) loss is leveraged as follows:

$$L_{mse} = \frac{1}{HWC} \sum_{x=1}^H \sum_{y=1}^W \sum_{z=1}^C \|I_T^{x,y,z} - I_{gt}^{x,y,z}\|^2 \quad (3)$$

where H , W , and C indicate height, width, and number of channels respectively. I_T indicates the final deraining result.

Furthermore, motivated by the observation that a model with MSE loss tends to obtain a blurred reconstruction [24, 25], we adopt an edge loss to preserve texture of background while removing rain streaks efficiently. To the best of our knowledge, this is the first paper to consider the edge preservation using edge loss in deraining task. The edge loss can be formulated as follows:

$$L_{edge} = \frac{1}{HWC} \sum_{x=1}^H \sum_{y=1}^W \sum_{z=1}^C \|f(I_T)^{x,y,z} - f(I_{gt})^{x,y,z}\|^2 \quad (4)$$

where f indicates convolution with a kernel K as defined in Eq. (5).

$$K = \begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix} \quad (5)$$

By leveraging a weighted combination of MSE loss and edge loss, the final loss function is defined as follows:

$$L_{final} = L_{mse} + \lambda L_{edge} \quad (6)$$

where λ is the weight of L_{edge} .

3. EXPERIMENTS

In this section, we present the training details and experimental results. To evaluate the performances of the proposed method, we choose five synthetic datasets with different sizes and complexities. Furthermore, a real-world dataset is obtained to assess the generalization ability of ReMAEN. Structural similarity (SSIM) [26] and peak signal-to-noise ratio (PSNR) are used on synthetic datasets. Since clean images are not available in real-world dataset, we evaluate it visually.

3.1. Training Details

Table 1: Descriptions of synthetic and real-world datasets. Values in each column of #train and #test indicate the number of clean and rainy image pairs except that real-world dataset only contains rainy images.

Datasets	#train	#test	label
Rain100L	200	100	rain mask/rain map
Rain100H	1800	100	rain mask/rain map
Rain800	700	100	-
Rain12000	12000	1200	rain-density labels
Rain14000	12600	1400	-
Real-world	-	67	-

As tabulated in Table 1, five synthetic datasets and a real-world dataset are chosen for comparison. Rain100L and

Rain100H are from [21], Rain800 is from [14], Rain12000 is obtained from [18], and Rain14000 is provided by [13]. Moreover, real-world dataset is collected from [14, 21]. Remarkably, during the training procedure, only clean and rainy image pairs are used without additional labels.

Settings. In ReMAEN, the rain details extracting path stacks 4 MAEBs and 4 RBs iteratively and 3 RBs are stacked symmetrically in rain details responding path. Moreover, we apply a shared SE block following all the RBs. Furthermore, we set $P = 3$ which indicates 3 paths are set in each MAB. Maximum number of stages T is set as 3. All the convolutional layers have filter size $f_s = 3$, stride $s = 1$, and number of filters $c = 32$. The learning rate is started with 0.001 and decreased to 0.0001 after 30000 iterations. The weight of the loss function is set as $\lambda = 0.1$. During training, 32 clean and rainy patch pairs with a size of 64×64 are randomly generated from input image pairs as a mini-batch per iteration. Adam optimizer is adopted with $\beta_1 = 0.9$ and $\beta_2 = 0.999$ respectively. We implement ReMAEN with Tensorflow library [27] and train it on a NVIDIA Geforce GTX 1070 GPU for 40000 iterations.

3.2. Ablation Study

Table 2: Investigation of recurrent block (RB), multi-scale aggregation block (MAB), symmetric skipping connection (SSC), squeeze-and-excitation (SE) block, and shared SE block. Performances are evaluated on Rain100L dataset.

Modules	Different combinations of modules				
RB	✓	✓	✓	✓	✓
MAB	✗	✓	✓	✓	✓
SSC	✗	✗	✓	✓	✓
SE	✗	✗	✗	✓	✓
Shared SE	✗	✗	✗	✗	✓
PSNR	35.62	35.96	36.48	36.69	36.75
SSIM	0.969	0.971	0.975	0.976	0.977

Ablation study on module. We conduct an ablation study to demonstrate the efficiency of each module. Simplistically, all variants in Table 2 are assessed using single stage rain removal scheme. As shown in Table 2. Simply stacking recurrent blocks (RB) will obtain relatively low performances. Applying multi-scale aggregation block (MAB) in rain details extracting path to extract multi-scale rain details can help to remove rain streaks efficiently. Moreover, features obtained in low level are transmitted to high level through symmetric skipping connection (SSC) which can make full use of rain details in low level and enhance the performances considerably. We can further observe that a squeeze-and-excitation (SE) block applied behind MAB can better model the inter-dependencies of features from MAB. To investigate the dependencies among the whole RBs, a shared SE block

is adopted. The shared SE block in ReMAEN can collaboratively select useful information from hidden statuses and improve the deraining performances.

Table 3: Investigation of recurrent rain removal stages. Performances are evaluated on Rain100L dataset.

Metrics	ReMAEN		
	stage-1	stage-2	stage-3
PSNR	36.75	36.82	37.80
SSIM	0.977	0.978	0.982

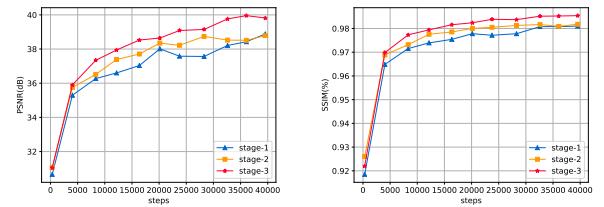


Fig. 2: Training convergence analysis on PSNR/SSIM of ReMAEN with different number of recurrent rain removal stages. Performances are evaluated on Rain100L dataset.

Ablation study on stage. To investigate the number of recurrent rain removal stages, we implement an ablation study on stage. As shown in Fig. 2, the increased recurrent stages can lead to better training performances. And the testing results on Rain100L are tabulated in Table 3. We can observe that multiple stages can better model the complex distribution of rain streaks and remove rain streaks efficiently by incorporating the correlations between neighboring stages. To balance the performances and memory, we remove rain streaks in 3 stages.

Table 4: Investigation of loss items (MSE loss and edge loss). Performances are evaluated on Rain100L dataset.

ReMAEN (stage-3)	PSNR	SSIM
w/o edge loss	37.08	0.978
with edge loss	37.80	0.982

Ablation study on loss. We further investigate the improvements obtained by edge loss. As can be seen in the yellow rectangle boxes of Fig. 3, training procedure with MSE loss and edge loss can well preserve the texture of background. However, training with MSE loss only is difficult to distinguish the texture of background and rain streaks with similar attributes (e.g. shape or color) and it leads to the failure of preserving texture of background. And Table 4 shows that training with MSE loss and edge loss can largely boost the performances while simply using MSE loss for training obtains relatively low performances.

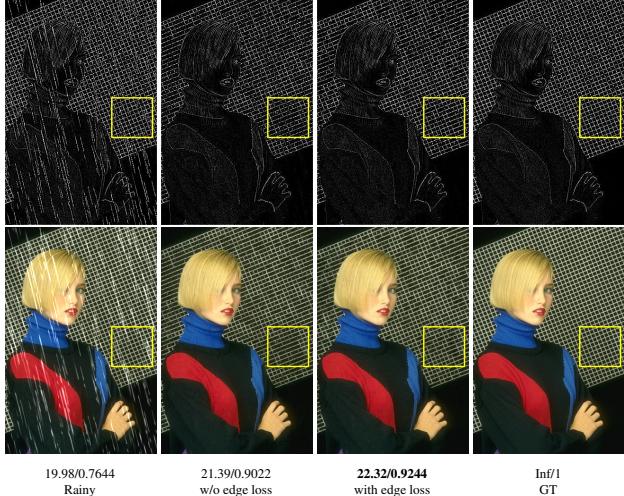


Fig. 3: Deraining results (without or with edge loss). Results are evaluated on Rain100L. The first row shows the results of convolution using kernel in Eq. (5) and the second row shows the deraining results. Values at the bottom of each image indicate PSNR/SSIM.

3.3. Quantitative and Qualitative Results

Table 5: Average PSNR and SSIM values on five synthetic datasets. The value with **red bold font** indicates ranking the first place in this column while value with **blue font** is the second place.

	Rain100L	Rain100H	Rain800	Rain12000	Rain14000
Methods	PSNR/SSIM	PSNR/SSIM	PSNR/SSIM	PSNR/SSIM	PSNR/SSIM
Rainy	25.52/0.825	12.13/0.349	21.16/0.652	21.15/0.778	23.69/0.757
CNN [12]	23.70/0.810	11.44/0.304	19.12/0.601	18.94/0.664	18.52/0.672
DSC [8]	24.16/0.870	15.66/0.544	18.56/0.599	21.44/0.789	22.03/0.799
LP [1]	29.11/0.880	14.26/0.423	22.27/0.741	22.75/0.835	25.64/0.836
DDN [13]	32.04/0.938	24.95/0.781	21.16/0.732	27.33/0.898	27.61/ 0.901
ID.CGAN [14]	25.88/0.891	14.16/0.607	22.73/0.817	23.32/0.803	21.93/0.784
JCAS [6]	28.40/0.881	13.65/0.459	22.19/0.766	27.91/0.778	28.77 /0.819
JORDER [21]	36.11/0.970	22.15/0.674	22.24/0.776	24.32/0.862	27.55/0.853
UGSM [11]	28.59/0.877	14.89/0.467	23.11/0.755	24.72/0.781	26.38/0.842
DID-MDN [18]	25.70/0.858	17.39/0.612	21.89/0.793	27.95/ 0.908	27.99/0.869
DualCNN [16]	26.87/0.860	14.23/0.468	24.11 /0.821	23.38/0.787	24.98/0.838
RESCAN [22]	36.64 / 0.975	26.45 / 0.846	24.09/ 0.841	29.95 /0.884	28.57/0.891
Ours	37.80 / 0.982	28.97 / 0.884	26.86 / 0.854	32.50 / 0.911	32.31 / 0.916

To demonstrate that the proposed method outperforms the state-of-the-art methods tremendously, we compare quantitative performances of different methods on five synthetic datasets with different sizes and complexities. As shown in Table 5, our method surpasses other methods on Rain100L, Rain100H, Rain800, and Rain12000, Rain14000 with the improvements of 1.16dB, 2.52dB, 2.75dB, 2.55dB, and 3.54dB on PSNR metric and 0.7%, 3.8%, 1.3%, 0.3%, and 1.5% on SSIM metric respectively.

Moreover, to visually demonstrate that the proposed method obtains favorable qualitative performance, ReMAEN is evaluated on both synthetic datasets and real-world dataset and the deraining results are presented in Fig. 4. Since the

limitation of the space, we only show the deraining results of recent state-of-the-art methods. As can be seen, deraining results on synthetic datasets obtained by JORDER [21] and DID-MDN [18] still remain many rain streaks. Moreover, JORDE [21] tends to dim and blur deraining images. Though RESACN [22] can remove rain streaks well, our method achieves the best performance. Furthermore, compared with other methods, our method can better remove rain streaks of rainy images from real-world datasets.

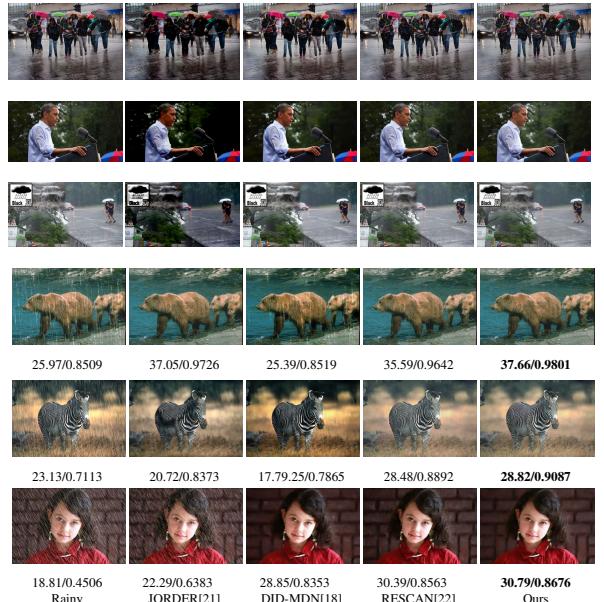


Fig. 4: Deraining results on synthetic and real-world datasets. The first three images are evaluated on real-world dataset. And the last three images are assessed on synthetic datasets. Values at the bottom of each image indicate PSNR/SSIM.

4. CONCLUSION

In this paper, we propose a recurrent multi-scale aggregation and enhancement network (ReMAEN) to progressively remove rain streaks from a rainy image. In ReMAEN, a multi-scale aggregation and enhancement block (MAEB) is constructed to model complex distribution of rain streaks efficiently. To better characterize the inter-dependencies among recurrent blocks, we adopt a shared channel-attention block to collaboratively select useful information of hidden statuses. Moreover, ReMAEN allows a symmetric skipping connection from low level to high level to fully exploit rain details. And an edge loss is incorporated with MSE loss to well preserve the texture of background. Experimental results show that the proposed method outperforms other methods considerably.

5. ACKNOWLEDGEMENT

This work was supported in part by National Natural Science Foundation of China (No. U1509206).

6. REFERENCES

- [1] Yu Li, Robby T. Tan, Xiaojie Guo, Jiangbo Lu, and Michael S. Brown, “Rain streak removal using layer priors,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 2736–2744.
- [2] Jin-Hwan Kim, Chul Lee, Jae-Young Sim, and Chang-Su Kim, “Single-image deraining using an adaptive nonlocal means filter,” in *Proceedings of the IEEE International Conference on Image Processing*, 2013, pp. 914–917.
- [3] Ye-Tao Wang, Xi-Le Zhao, Tai-Xiang Jiang, Liang-Jian Deng, Yi Chang, and Ting-Zhu Huang, “Rain streak removal for single image via kernel guided cnn,” *arXiv preprint arXiv:1808.08545*, 2018.
- [4] Yi-Lei Chen and Chiou-Ting Hsu, “A generalized low-rank appearance model for spatio-temporally correlated rain streaks,” in *Proceedings of the IEEE International Conference on Computer Vision*, 2013, pp. 1968–1975.
- [5] Jin-Hwan Kim, Jae-Young Sim, and Chang-Su Kim, “Video de-raining and desnowing using temporal correlation and low-rank matrix completion,” *IEEE Transactions on Image Processing*, vol. 24, no. 9, pp. 2658–2670, 2015.
- [6] Shuhang Gu, Deyu Meng, Wangmeng Zuo, and Lei Zhang, “Joint convolutional analysis and synthesis sparse representation for single image layer separation,” in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 1717–1725.
- [7] Li-Wei Kang, Chia-Wen Lin, and Yu-Hsiang Fu, “Automatic single-image-based rain streaks removal via image decomposition,” *IEEE Transactions on Image Processing*, vol. 21, no. 4, pp. 1742–1755, 2012.
- [8] Yu Luo, Yong Xu, and Hui Ji, “Removing rain from a single image via discriminative sparse coding,” in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 3397–3405.
- [9] Yinglong Wang, Shuaicheng Liu, Chen Chen, and Bing Zeng, “A hierarchical approach for rain or snow removing in a single color image,” *IEEE Transactions on Image Processing*, vol. 26, no. 8, pp. 3936–3950, 2017.
- [10] Lei Zhu, Chi-Wing Fu, Dani Lischinski, and Pheng-Ann Heng, “Joint bilayer optimization for single-image rain streak removal,” in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 2526–2534.
- [11] Liang-Jian Deng, Ting-Zhu Huang, Xi-Le Zhao, and Tai-Xiang Jiang, “A directional global sparse model for single image rain removal,” *Applied Mathematical Modelling*, vol. 59, pp. 662–679, 2018.
- [12] David Eigen, Dilip Krishnan, and Rob Fergus, “Restoring an image taken through a window covered with dirt or rain,” in *Proceedings of the IEEE International Conference on Computer Vision*, 2013, pp. 633–640.
- [13] Xueyang Fu, Jiabin Huang, Delu Zeng, Yue Huang, Xinghao Ding, and John Paisley, “Removing rain from single images via a deep detail network,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 1715–1723.
- [14] He Zhang, Vishwanath Sindagi, and Vishal M. Patel, “Image de-raining using a conditional generative adversarial network,” *arXiv preprint arXiv:1701.05957*, 2017.
- [15] Siyuan Li, Wenqi Ren, Jiawan Zhang, Jinke Yu, and Xiaoje Guo, “Fast single image rain removal via a deep decomposition-composition network,” *arXiv preprint arXiv:1804.02688*, 2018.
- [16] Jinshan Pan, Sifei Liu, Deqing Sun, Jiawei Zhang, Yang Liu, Jimmy Ren, Zechao Li, Jinhui Tang, Huchuan Lu, and Yu-Wing Tai, “Learning dual convolutional neural networks for low-level vision,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 3070–3079.
- [17] Xueyang Fu, Borong Liang, Yue Huang, Xinghao Ding, and John Paisley, “Lightweight pyramid networks for image de-raining,” *arXiv preprint arXiv:1805.06173*, 2018.
- [18] He Zhang and Vishal M. Patel, “Density-aware single image de-raining using a multi-stream dense network,” in *The IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 695–704.
- [19] Guanbin Li, Xiang He, Wei Zhang, Huiyou Chang, Le Dong, and Liang Lin, “Non-locally enhanced encoder-decoder network for single image de-raining,” *Proceedings of the 26th ACM International Conference on Multimedia*, pp. 1056–1064, 2018.
- [20] Ruoteng Li, Loong-Fah Cheong, and Robby T. Tan, “Single image de-raining using scale-aware multi-stage recurrent network,” *arXiv preprint arXiv:1712.06830*, 2017.
- [21] Wenhao Yang, Robby T. Tan, Jiashi Feng, Jiaying Liu, Zongming Guo, and Shuicheng Yan, “Deep joint rain detection and removal from a single image,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 1357–1366.
- [22] Xia Li, Jianlong Wu, Zhouchen Lin, Hong Liu, and Hongbin Zha, “Recurrent squeeze-and-excitation context aggregation net for single image deraining,” in *The European Conference on Computer Vision (ECCV)*, 2018, pp. 262–277.
- [23] Jie Hu, Li Shen, and Gang Sun, “Squeeze-and-excitation networks,” *arXiv preprint arXiv:1709.01507*, 2017.
- [24] Deepak Pathak, Philipp Krahenbuhl, Jeff Donahue, Trevor Darrell, and Alexei A Efros, “Context encoders: Feature learning by inpainting,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 2536–2544.
- [25] Christian Ledig, Lucas Theis, Ferenc Huszar, Jose Caballero, Andrew Cunningham, Alejandro Acosta, Andrew P. Aitken, Alykhan Tejani, Johannes Totz, Zehan Wang, and Wenzhe Shi, “Photo-realistic single image super-resolution using a generative adversarial network.,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 105–114.
- [26] Zhou Wang, Alan C. Bovik, Hamid R. Sheikh, and Eero P. Simoncelli, “Image quality assessment: from error visibility to structural similarity,” *IEEE Transactions on Image Processing*, vol. 13, no. 4, pp. 600–612, 2004.
- [27] Martín Abadi, Paul Barham, Jianmin Chen, Zhifeng Chen, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Geoffrey Irving, Michael Isard, Manjunath Kudlur, Josh Levenberg, Rajat Monga, Sherry Moore, Derek Gordon Murray, Benoit Steiner, Paul A. Tucker, Vijay Vasudevan, Pete Warden, Martin Wicke, Yuan Yu, and Xiaoqiang Zhang, “Tensorflow: a system for large-scale machine learning.,” in *OSDI*, 2016, vol. 16, pp. 265–283.