MSFA-Net: a Network for Single Image Deraining

Shun Wu\textsuperscript{a} and Jun Zhou\textsuperscript{b}

Institute of Image Communication and Network Engineering, Shanghai JiaoTong University, China.

\textsuperscript{a}wushun@sjtu.edu.cn, \textsuperscript{b}zhoujun@sjtu.edu.cn

Abstract. Rain streaks degrade the quality of image. Many methods have been proposed to solve single image rain streaks removal recently. However, some methods over-smooth the recovered image. A deep network architecture called Multi-Scale Feature Attention Network (MSFA-Net) is proposed in this paper. We propose a novel basic block structure to exploit the image features, which consists of multi-scale residual learning block and feature attention block. Several basic block structures with a local residual learning compose a group architecture. The outputs of each group architecture are concatenated for final multi-scale feature fusion. Then the features are fed into feature attention block and reconstruction module. Finally, a global residual learning module restore the clean image. Besides, the feature attention block combines channel attention with spatial attention. The proposed MSFA-Net removes the rain streaks which study a non-linear mapping relationship between the rainy and clean image from synthesized dataset. Through comparing with other state-of-the-art algorithms, our algorithm performs better for both synthesized rainy image data and real rainy image data.

1. Introduction

Images and videos with rain streaks are collected by outdoor vision systems, which have bad quality and then lead to poor performance for other computer vision tasks including image classification, person re-identification and so on [1-3]. So the rain streaks removal is a necessary research task. Nowadays, various algorithms have been proposed to remove rain streaks for both video and single image. In general, there are two categories for both video and single image rain streaks removal: (1) conventional methods, (2) methods based on deep learning.

Video rain streaks removal can take advantage of the temporal redundancy. By contrast, single image rain streaks removal is more complicated. Thus, we concentrate on removing rain streak removal for single image. The widely used rain model [4-6] for single image is described as:

\[ O = B + \tilde{S} \]  

where \( O \) is the rainy image or the input of the method, \( B \) is the clean image or the output of the method, \( \tilde{S} \) is the rain streak layer. Recovering the background \( B \) from the rainy image \( O \) and enhancing the image visibility are the goals of removing the rain streaks. In recent years, most methods have achieved certain success but have their suitability and advantage on specific situations.

Motivated by the work [7] which has multiple local residual connection, we further propose a Multi-Scale Feature Attention Network to remove single image rain streaks (Section 3). We utilize their whole framework [7]. Unlike its work, we further design a more effective basic block structure. [7] attempts to increase the number of its basic block structure to improve the ability of the network. But it results in
inadequate feature utilization. So our basic block structure includes multi-scale residual learning block and the feature attention block inspired by [8]. The multi-scale residual learning block may take maximum advantage of the rainy image features on different scales. So it can be regarded as local multi-scale features (Section 3.1). Afterwards, the different outputs of three group architecture are concatenated for final multi-scale feature fusion. In summary, our contributions are:

We present a Multi-Scale Feature Attention Network (MSFA-Net) to remove single image rain streaks. MSFA-Net achieves better performance on synthetic datasets than the other state-of-the-art algorithms. Experimental results on real-world images also demonstrated the superiority of MSFA-Net.

We design a new basic block structure that combines multi-scale residual learning block and feature attention. Multi-scale residual learning block can obtain features on different scales and to allow the information of the thin rain regions or no rain regions to be bypassed through plenty of skip connections.

We propose a feature attention merges channel attention, spatial attention into the residual learning. The channel attention gathers the average-pooled and maximum-pooled features. This module focuses on the thick rain region.

![Figure 1. The network of MSFA-Net.](image)

2. Related work
Removing rain streaks is an ill-posed and challenging question for single image. For conventional methods, they mainly employ a model-driven methodology which utilize prior knowledge or physical properties. [2] proposed a method decomposing the rainy image into low-frequency and high-frequency layers. In high-frequency layer, they set apart the rain streaks from background. However, in this process, they lost some details of the background. Li et al. [5] exploited GMM based patch prior to adapt variable orientations and scales of rain streaks. They achieved good performance, but also slightly smoothed the background. Gu et al. [9] designed a joint convolutional analysis and synthesis (JCAS) sparse representation model which integrate Analysis sparse representation (ASR) and synthesis sparse representation (SSR). However, their methods generally needed many iterations of computation resulting in their inefficiency. In [10], they proposed an unrolling strategy to remove the rain whose conventional numerical iterations involve data-dependent information. [11] indicated that this is a good effort to integrate model-driven and data-driven methodologies.

For deep learning methods, they mainly adopt a data-driven manner which design specific network. [12] proposed a density-aware image deraining approach (DID-MDN). By utilizing the residual-aware classifier process, they can adaptively determine the density information of rain. However, the real rain image can not judge the density only according to these three degree in fact. [13] utilized a recurrent squeeze-and-excitation (SE) block to recover the background. In [14-15], their network regarded the
image detail layer as input so they had an advantage in keeping texture details. But they did not deal with removing rain streaks in heavy rain cases. Yang et al. [16] proposed multi-task deep learning architecture. They further designed an enhanced version JORDER-E to get a better performance [17]. In contrast to these previous algorithms based on deep learning which treat channel-wise feature equally, we concentrate on treating the channel-wise and spatial-wise features unequally.

3. Method
The input of the MSFA-Net is a rainy image in Figure 1. The whole network includes multiple skip connections for both local residual connection and global residual connection which enable the less important information to be bypassed as same as [7]. The input image is fed into a $3 \times 3$ convolution layer. And then it is sent to 3 Group Architectures. The different output features of 3 Group Architectures are concatenated and then fed into a feature attention. The final feature is fed into the restoration part and the global residual learning to attain the no-rain image. Besides, every Group Architectures include 13 Basic Block Structures. Figure 2 shows the Basic Block Structure which combines the multi-scale residual learning block and the feature attention block.

3.1. Multi-scale residual learning block
The rain degradation is complex. [17] pointed out that numerous methods were performed in a restricted receptive field. So we adopt different convolutional kernel size to attain different receptive field and then get different scale features based on their work [8]. The multi-scale residual learning block includes local multi-scale features fusion and local residual learning as shown in Figure 2. Like [8], it is a local two-bypass network. The input feature maps are fed into diverse convolutional layers. Its kernel size is $3 \times 3$, $5 \times 5$ respectively. And then they are sent to the ReLU function to improve the power of the network. Finally all these feature maps are concatenated and passed into a $1 \times 1$ convolutional layer. The detailed operation can be described as:

\[
S_1 = \delta(Conv_{3 \times 3}(M_{in}))
\]

\[
R_1 = \delta(Conv_{5 \times 5}(M_{in}))
\]

\[
S_2 = \delta(Conv_{3 \times 3}([S_1, R_1]))
\]

\[
R_2 = \delta(Conv_{5 \times 5}([R_1, S_1]))
\]

\[
M'_{out} = Conv_{1 \times 1}([S_2, R_2])
\]
\[ M_{\text{out}} = M'_{\text{out}} + M_{\text{in}} \]  

(7)

Where \( M_{\text{in}} \) denotes the input of this module, \( M_{\text{out}} \) denotes the output of this module. \( \delta(\cdot) \) represents the ReLU function. While the subscripts of \( \text{Conv} \) denotes the size of the convolutional kernel. And \([S_1, R_1], [R_1, S_1], [S_2, R_2]\) denote the concatenation operation. This module can detect the input image features adaptively.

### 3.2. Feature attention

[18] indicated that the edge or texture area involves more high-frequency information. However, most image deraining networks ignored the channel and spatial attention and their work always tended to remove texture details more or less and then resulted in over-smoothing effect for the recovered background. Hence, our network should concentrate on the more important regions to restore the high-frequency details or the thick rain regions. So the feature attention block learns what and where to concern or suppress through combining the channel attention and spatial attention in Figure 3.

![Feature Attention](image)

**Figure 3.** The Feature Attention.

We apply the average pooling and maximum pooling feature for the channel attention to increase the effectiveness of the network. They can modulate the feature representations more adaptively. Before entering the feature attention block, we feed the input \( M_{\text{out}} \) into a convolutional layer whose kernel size is \( 3 \times 3 \).

\[ FA_{\text{in}} = \text{Conv}_{3 \times 3}(M_{\text{out}}) \]  

(8)

Where \( FA_{\text{in}} \) denotes the input of feature attention. The channel attention focuses on the inter-channel features. For many works, average-pooling has been used so far. However, maximum-pooling features can aggregate another important clue to acquire finer channel-wise attention. [19] thought max-pooled feature could compensate for the missing encoding of the average-pooled feature. So we use average pooling and maximum pooling as feature descriptors simultaneously. Then the features enter into two \( 1 \times 1 \) convolution layers, a ReLU and a sigmoid activation function.

\[ \text{avg}_c = \text{AvgPool}(FA_{\text{in}}) \]  

(9)

\[ \text{max}_c = \text{MaxPool}(FA_{\text{in}}) \]  

(10)

\[ CA_{\text{out}}^* = \sigma(\text{Conv}_{1 \times 1}(\delta(\text{Conv}_{1 \times 1}(\text{avg}_c))) + \text{Conv}_{1 \times 1}(\delta(\text{Conv}_{1 \times 1}(\text{max}_c)))) \]  

(11)
Where \( \text{avg}_c, \text{max}_c \) is the result after average pooling and maximum pooling respectively. \( \sigma(\cdot) \) denotes a Sigmoid function, \( \delta(\cdot) \) represents a ReLU function. Then we use element-wise multiplication for the input \( FA_{in} \) and \( CA_{out}^* \). \( CA_{out} \) is the final output of the channel attention.

\[
CA_{out} = FA_{in} \odot CA_{out}^*
\] (12)

[10], [17] indicated that spatial contextual information demonstrated to be helpful for single image rain removal. Considering aforementioned discussion, we adopt the spatial attention after the channel attention. The features pass into the two \( 1 \times 1 \) and a ReLU, sigmoid activation function.

\[
SA_{out}^* = \sigma(Conv_{1 \times 1}(\delta(Conv_{1 \times 1}(CA_{out}))))
\] (13)

Then we element-wise multiply the input \( CA_{out} \) and the weights of the spatial attention \( SA_{out}^* \). \( SA_{out} \) is the result of spatial attention.

\[
FA_{out} = SA_{out} \odot CA_{out} \odot SA_{out}^*
\] (14)

We visualize the spatial and channel attention weight map of three Group Architectures to visually demonstrate the effectiveness. We can see clearly that the features of rainy streaks are given less weight from Figure 4. As for channel attention, we reveals a \( 3 \times 64 \) sized map which every row represents corresponding Group Architecture output. Different features are assigned to different weights as shown in Figure 5. The subsequent experiments results can demonstrate the effectiveness of our method.

![Spatial Attention Map](image)

**Figure 4.** Spatial Attention Map.

![Channel Attention Map](image)

**Figure 5.** Channel Attention Map.

| Table 1. PSNR/SSIM results among different algorithms |
|----------------------------------------------------|
| Baseline | Rain100L | Rain100H |
|----------|----------|----------|
| DSC[4]   | 24.16/0.8633 | 15.66/0.5444 |
| LP[5]    | 29.11/0.8812 | 14.26/0.4225 |
| DDN[14]  | 33.50/0.9444 | 20.12/0.6531 |
| UGSM[20] | 28.83/0.8823 | 13.40/0.5089 |
| JCAS[9]  | 29.91/0.9041 | 14.26/0.4837 |
| DID-MDN[12] | 28.27/0.8569 | 13.85/0.3748 |
| ID-CGAN[21] | 23.39/0.8186 | 16.86/0.4921 |
| JORDER-E[17] | 37.10/0.9795 | 24.54/0.8024 |
| FFA-Net[7] | 38.10/0.98 | 28.88/0.8932 |
4. Experiments

4.1. Datasets and baseline methods
There are a few benchmark datasets to evaluate our method. These datasets are: (1) Rain100L, which consists of 200 rainy/clear images for training, 100 rainy/clear images for testing and has one type of rain streaks. (2) Rain100H, which includes 1800 rainy/clear images for training, 100 rainy/clear images for testing and has five types of rain streaks. This paper make a comparison among following methods: DSC [4], LP [5], DDN [14], UGSM [20], JCAS [9], DID-MDN [12], ID-CGAN [21], JORDER-E [17], PReNet [3], FFA-Net [7]. We employ Peak Signal-to-Noise Ratio (PSNR) and Structure Similarity Index (SSIM) to assess the performance of these algorithms on synthesized datasets.

4.2. Implementation details
We adopt $L_1$ loss function.

$$L(\theta)=\frac{1}{N}\sum_{i=1}^{N}||I_{gt}-MSF^\theta(I_{rain})||$$

$\theta$ denotes the parameters of this network. $I_{gt}$ signifies the ground truth and $I_{rain}$ signifies the corresponding rainy image. We use ADAM optimizer. Our initial learning rate is 0.0001. While our learning rate adjusts from 0.0001 to 0 according to the cosine annealing strategy [7], [22]. The Rain100L and the Rain100H is trained for $3\times10^5$, $2\times10^5$ respectively. Every Group Architecture has 64 filters. We implement our model on Pytorch using two NVIDIA TITIAN XP GPU.

4.3. Experiment results
Table 1 displays the PSNR/SSIM results of different algorithms on Rain100L and Rain100H. Note that FFA-NET [7] is trained from scratch on these datasets and the initial conditions of the experiment are the same as ours. Other approaches are available online. As observed, our approach is evidently superior over other state-of-the-art methods. Such good performance proves that combining the multi-scale residual learning block and the feature attention boosts the performance on these synthesized rainy datasets.

| Algorithm       | PSNR  | SSIM  |
|-----------------|-------|-------|
| PReNet[3]       | 37.48 | 0.979 |
| MSFA-Net(ours)  | 39.78 | 0.988 |

Figure 6. Comparison of MSFA-Net with other algorithms on Rain100H dataset.

Figure 6 and Figure 7 show some results of synthesized images. Five competing methods are considered including LP [5], JCAS [9], DDN [14], UGSM [20] and PReNet [3]. Enlarging the images can demonstrate our approach is superior. For the images on Rain100H, the rain streaks effect the image...
quality seriously. As observed, LP [5], JCAS [9], DDN [14], UGSM [20] can not remove most of the rain streaks. PReNet [3] and our method are superior over them for rain streaks removal. However, our method behaves better than PReNet [3] in preserving the details such as the second line of Figure 6 on the zebra markings recovery. For the images on Rain100L whose rain streaks are light, LP [5], JCAS [9], DDN [14], UGSM [20] remove most of the rain streaks but over-smooth the background resulting in lacking of details. Although PReNet [3] remove all the rain streaks, it sometimes smooths the background like the second line of Figure 7 on the buildings recovery. In summary, MSFA-Net (ours) is better in removing the rain streaks.

Figure 8 shows some results of rainy images in the real world. All these images were taken in real rainy situations. The ground truth of these images do not exist so we compare with each other. As observed, our method also behaves better than other approaches through zooming in these recovered images.

![Figure 7. Comparison of MSFA-Net with other algorithms on Rain100H dataset.](image)

![Figure 8. Comparison of MSFA-Net with other algorithms on some real-world rainy images.](image)
5. Conclusion
We present a Multi-Scale Feature Attention Network to remove rain streaks. We gather the multi-scale residual learning block and the feature attention in a novel basic block structure. The multi-scale residual learning block combines local multi-scale features which improves the network performance greatly. The feature attention gives different weights to different features. The different outputs of three group architecture are concatenated for final multi-scale feature fusion. Evaluations on the synthesized images and real-world images proved our algorithm outperforms state-of-the-art algorithms. Like most data-driven methods, to achieve better results on real images, we need many training rainy in real world which is time-consuming and difficult to collect. So we will consider the self-supervised method to improve the network in the following work. Putting the real-world rainy images without clean images into the network will enhance the ability of the network in real world.

References
[1] Fu X, Liang B, Huang Y, Ding X and Paisley J 2019 IEEE Transactions on Neural Networks and Learning Systems 1–14
[2] Kang L, Lin C and Fu Y 2012 IEEE Transactions on Image Processing 21 1742–1755
[3] Ren D, Zuo W, Hu Q, Zhu P and Meng D 2019 Progressive image deraining networks: A better and simpler baseline 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) pp 3932–3941
[4] Luo Y, Xu Y and Ji H 2015 Removing rain from a single image via discriminative sparse coding 2015 IEEE International Conference on Computer Vision (ICCV) pp 3397–3405
[5] Li Y, Tan R T, Guo X, Lu J and Brown M S 2016 Rain streak removal using layer priors 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR) pp 2736–2744
[6] Huang D, Kang L, Yang M, Lin C and Wang Y F 2012 Context-aware single image rain removal 2012 IEEE International Conference on Multimedia and Expo pp 164–169
[7] Qin X, Wang Z, Bai Y, Xie X and Jia H 2019 Ffa-net: Feature fusion attention network for single image dehazing (Preprint 1911.07559)
[8] Li J, Fang F, Mei K and Zhang G 2018 Multi-scale residual network for image super-resolution Computer Vision – ECCV 2018 ed Ferrari V, Hebert M, Sminchisescu C and Weiss Y (Cham: Springer International Publishing) pp 527–542
[9] Gu S, Meng D, Zuo W and Zhang L 2017 Joint convolutional analysis and synthesis sparse representation for single image layer separation 2017 IEEE International Conference on Computer Vision (ICCV) pp 1717–1725
[10] Mu P, Chen J, Liu R, Fan X and Luo Z 2019 IEEE Signal Processing Letters 26 307–311
[11] Wang H, Wu Y, Li M, Zhao Q and Meng D 2019 A survey on rain removal from video and single image (Preprint 1909.08326)
[12] Zhang H and Patel V M 2018 Density-aware single image de-raining using a multi-stream dense network 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition pp 695–704
[13] Li X, Wu J, Lin Z, Liu H and Zha H 2018 Lecture Notes in Computer Science 262C277
[14] Fu X, Huang J, Zeng D, Huang Y, Ding X and Paisley J 2017 Removing rain from single images via a deep detail network 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR) pp 1715–1723
[15] Fu X, Huang J, Ding X, Liao Y and Paisley J 2017 IEEE Transactions on Image Processing 26 2944–2956
[16] Yang W, Tan R T, Feng J, Liu J, Guo Z and Yan S 2017 Deep joint rain detection and removal from a single image 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR) pp 1685–1694
[17] Yang W, Tan R T, Feng J, Liu J, Yan S and Guo Z 2019 IEEE Transactions on Pattern Analysis and Machine Intelligence 1–1
[18] Hu Y, Li J, Huang Y and Gao X 2019 IEEE Transactions on Circuits and Systems for Video Technology 1–1
[19] Woo S, Park J, Lee J Y and Kweon I S 2018 *Lecture Notes in Computer Science* 3C19
[20] Deng L J, Huang T Z, Zhao X L and Jiang T X 2018 *Applied Mathematical Modelling* 59
[21] Zhang H, Sindagi V and Patel V M 2019 *IEEE Transactions on Circuits and Systems for Video Technology* 1–1
[22] He T, Zhang Z, Zhang H, Zhang Z, Xie J and Li M 2019 Bag of tricks for image classification with convolutional neural networks *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* pp 558–567