Multi-Scale Progressive Fusion Network for Single Image Deraining

Kui Jiang\(^1\)  Zhongyuan Wang\(^1\)*  Peng Yi\(^1\)  Chen Chen\(^2\)*  Baojin Huang\(^1\)  Yimin Luo\(^3\)  Jiayi Ma\(^1\)  Junjun Jiang\(^4\)
\(^1\)Wuhan University  \(^2\)University of North Carolina at Charlotte  \(^3\)King’s College London  \(^4\)Harbin Institute of Technology

Abstract

Rain streaks in the air appear in various blurring degrees and resolutions due to different distances from their positions to the camera. Similar rain patterns are visible in a rain image as well as its multi-scale (or multi-resolution) versions, which makes it possible to exploit such complementary information for rain streak representation. In this work, we explore the multi-scale collaborative representation for rain streaks from the perspective of input image scales and hierarchical deep features in a unified framework, termed multi-scale progressive fusion network (MSPFN) for single image rain streak removal. For the similar rain streaks at different positions, we employ recurrent calculation to capture the global texture, thus allowing to explore the complementary and redundant information at the spatial dimension to characterize target rain streaks. Besides, we construct multi-scale pyramid structure, and further introduce the attention mechanism to guide the fine fusion of these correlated information from different scales. This multi-scale progressive fusion strategy not only promotes the cooperative representation, but also boosts the end-to-end training. Our proposed method is extensively evaluated on several benchmark datasets and achieves the state-of-the-art results. Moreover, we conduct experiments on joint deraining, detection, and segmentation tasks, and inspire a new research direction of vision task driven image deraining. The source code is available at https://github.com/kuihua/MSPFN.

1. Introduction

Due to substantial degradation of the image content in rain images and videos, traditional image enhancement algorithms [27] struggle to make desirable improvements on image quality. Therefore, developing specialized solutions for image deraining is imperative to a wide range of tasks [12], e.g. object detection and semantic segmentation.

*Corresponding author
rows) encode complementary or redundant information for feature representation, which would help deraining in the original image. This correlation of image contents across scales has been successfully applied to other computer vision tasks [10, 33]. Recently, authors in [8, 44] construct pyramid frameworks to exploit the multi-scale knowledge for deraining. Unfortunately, those exploitations fail to make full use of the correlations of multi-scale rain streaks (although restricted to a fixed scale-factor of 2 [10]). For example, Fu et al. [8] decompose the rain image into different pyramid levels based on its resolution, and then individually solve the restoration sub-problems at the specific scale space through several parallel sub-networks. Such decomposition strategy is the basic idea of many recurrent deraining frameworks [19]. Unlike [8] completing the deraining task from each individual resolution level, Zheng et al. [44] present a density-specific optimization for rain streak removal in a coarse-to-fine fashion, and gradually produce the rain-free image stage-by-stage [15]. However, there are no direct communications of the inter-level features across cascaded pyramid layers except for the final outputs, thus failing to take all-rounded advantages of the correlated information of rain streaks across different scales. Consequently, these methods [8, 44] are still far from producing the desirable deraining results with the limited exploitation and utilization of multi-scale rain information.

To address these limitations of the prior works, we explore the multi-scale representation from input image scales and deep neural network representations in a unified framework, and propose a multi-scale progressive fusion network (MSPFN) to exploit the correlated information of rain streaks across scales for single image deraining. Specifically, we first generate the Gaussian pyramid rain images using Gaussian kernels to down-sample the original rain image in sequence. A coarse-fusion module (CFM) (§3.1) is designed to capture the global texture information from these multi-scale rain images through recurrent calculation (Conv-LSTM), thus enabling the network to cooperatively represent the target rain streak using similar counterparts from global feature space. Meanwhile, the representation of the high-resolution pyramid layer is guided by previous outputs as well as all low-resolution pyramid layers. A fine-fusion module (FFM) (§3.2) is followed to further integrate these correlated information from different scales. By using the channel attention mechanism, the network not only discriminatively learns the scale-specific knowledge from all preceding pyramid layers, but also reduces the feature redundancy effectively. Moreover, multiple FFM’s can be cascaded to form a progressive multi-scale fusion. Finally, a reconstruction module (RM) is appended to aggregate the coarse and fine rain information extracted respectively from CFM and FFM for learning the residual rain image, which is the approximation of real rain streak distribution. The overall framework is outlined in Fig. 2. The main contributions of this paper are as follows:

- We uncover the correlations of rain streaks in an image and propose a novel multi-scale progressive fusion network (MSPFN) which collaboratively represents rain streaks from multiple scales via the pyramid representation.
- To better characterize rain streaks of different scales, we devise three basic modules, coarse-fusion module (CFM), fine-fusion module (FFM) and reconstruction module (RM), to effectively extract and integrate the multi-scale information. In these modules, the complementary information of similar patterns with rain streaks, both within the same scale or across different scales (pyramid layers), is progressively fused to characterize the rain streaks distribution in a collaborative/cooperative manner.
- Apart from achieving the state-of-the-art deraining performance in terms of the conventional quantitative measurements (e.g. PSNR and SSIM), we build several synthetic rain datasets based on COCO [3] and BDD [38] datasets for joint image deraining, detection and segmentation tasks. To the best of our knowledge, we are the first to apply mainstream vision-oriented tasks (detection and segmentation) for comprehensively evaluating the deraining performance.

2. Related Work

In the last few years, substantial improvements [24, 18, 4, 17] have been observed on rain image restoration. In this work, we mainly focus on single image deraining because it is more challenging.

2.1. Single Image Deraining

Previous traditional methods for single image deraining [5, 14] fail under the complex rain conditions and produce degraded image contents due to the limited linear-mapping transformation. Very recently, deep-learning based approaches [24, 29, 39] have emerged for rain streak removal and demonstrated impressive restoration performance. For example, Fu et al. [6] introduce a three-layer convolutional neural network (CNN) to estimate and remove rain streaks from its rain-contaminated counterpart. To better represent rain streaks, Zhang et al. [40] take the rain density into account and present a multi-task CNN for joint rain density estimation and deraining. Later, Zhang et al. [41] further incorporate quantitative, visual and discriminative performance into the objective function, and propose a conditional generative adversarial network for rain streak removal. In order to alleviate the learning difficulty, recurrent frameworks [19, 36, 26] are designed to remove rain streaks in a stage-wise manner.
2.2. Multi-scale Learning

Rain streaks in the air show the apparent self-similarity, both within the same scale or across different scales, which makes it possible to exploit the correlated information across scales for rain streak representation. However, most existing deraining methods [16, 40] ignore the underlying correlations of rain streaks across different scales. Only a few attempts [8, 44] have been made to exploit the multi-scale knowledge. Fu et al. [8] decompose the restoration task into multiple subproblems and employ a set of parallel subnetworks to individually estimate the rain information in a specific pyramid scale space. However, it does not exploit and utilize the correlated information among these pyramid layers. Different from the parallel pyramid framework in [8], Zheng et al. [44] propose the cascaded pyramid network, which is similar to LapSRN [15], to iteratively remove rain streaks. However, only the high-level features are used to help the adjacent pyramid representation, which results in losing some useful hierarchical and scale features in a deep cascaded network. The significance of these features produced at different stages has been verified on image reconstruction tasks [28, 43].

Different from these methods [8, 44], in this work we introduce a novel framework MSPFN to achieve the collaborative representation of rain streaks across different scales, where the rich multi-scale rain information extracted from the Gaussian pyramid images is progressively aggregated along the pyramid layers and stages of the network. As a result, our predicted rain streak distribution is more accurate via the multi-scale collaborative representation.

3. Proposed Method

Fig. 2 shows the overall pipeline of our proposed multi-scale progressive fusion network (MSPFN) for image deraining by excavating and exploiting the inherent correlations of rain streaks across different scales. We present the details of each building block and the loss function in the following.

3.1. Multi-scale Coarse Fusion

For a given rain image, our method first generates the Gaussian pyramid rain images using Gaussian kernels to down-sample the original rain image into different scales, e.g., 1/2 and 1/4. The network takes as input the pyramid rain images and extracts the shallow features through multiple parallel initial convolution layers (see the first block of “initial layer” in Fig. 2). Based on the initial features from each scale, the coarse-fusion module (CFM) then performs the deep extraction and fusion of multi-scale rain information through several parallel residual recurrent units (RRU), as shown in Fig. 3. The reasons for designing CFM are three folds: (a) To exploit the repetition of rain streaks under the same scale, we apply the recurrent calculation and residual learning to capture the global texture information, making it possible to cooperatively represent target rain streaks. More accurately, we introduce Conv-LSTM to model the information flow of context textures at spatial dimension with the recursive memory, where the contextual texture correlations are transformed into structured cyclic dependencies to capture the complementary or redundant rain information (e.g. the solid arrows in Fig. 1). (b) The multi-scale structure provides an alternative solution to greatly increase the receptive filed to cover more contents while maintaining a
shallow depth. (c) The high-resolution representations benefit from the outputs of previous stages as well as all low-resolution pyramid layers via iterative sampling and fusion.

### 3.2. Multi-scale Fine Fusion

The outputs of CFM go through the fine-fusion module (FFM) to refine the correlated information from different scales. As shown in Fig. 2, FFM enjoys the similar multi-scale structure with CFM for convenience. Unlike CFM, we introduce the channel attention unit (CAU) to enhance the discriminative learning ability of the network through focusing on the most informative scale-specific knowledge, making the cooperative representation more efficient. To alleviate the computation burden, we apply the strided convolution to reduce the spatial dimension of features, and alleviate the computation burden, we apply the strided convolution to reduce the spatial dimension of features, and finally utilize the deconvolution layer to increase the resolution to avoid losing resolution information, resulting in the U-shaped residual attention block (URAB). As depicted in Fig. 4, URAB is composed of several CAUs, along with the short skip connections to help the fine representation of multi-scale rain information. Moreover, long skip connections are used between cascaded FFMs to achieve progressive fusion of multi-scale rain information as well as to facilitate the effective backward propagation of the gradient.

### 3.3. Rain Streak Reconstruction

To learn the final residual rain image, we further integrate both low- and high-level multi-scale features respectively from CFM and FFM via a reconstruction module (RM), schematically depicted in Fig. 2. Specifically, the outputs from CFM are concatenated with the outputs from the last FFM, and then a convolution layer is used to learn the channel interdependence and rescale the feature values from the two modules. Similarly, the iterative sampling and fusion of rain information across different pyramid layers are implemented to estimate the residual rain image.

### 3.4. Loss Function

Mean squared error (MSE) is the commonly used loss to train the network [40, 34]. However, it usually produces blurry and over-smoothed visual effect with the loss of high-frequency textures due to the squared penalty. In this work, we perform the successive approximation to the real rain streak distribution $I_R$ with the guidance of the Charbonnier penalty function [15], which is more tolerant of small errors and holds better convergence during training. The function is expressed as

$$L_{con} = \sqrt{(I_R^* - I_R)^2 + \varepsilon^2}. \tag{1}$$

In Equation (1), $I_R^*$ denotes the predicted residual rain image. The predicted rain-free image $I_{Dera}^\text{clean}$ is generated by subtracting $I_R^*$ from its rain-contaminated counterpart $I^\text{rain}$. The penalty coefficient $\varepsilon$ is empirically set to $10^{-3}$.

In order to further improve the fidelity and authenticity of high-frequency details while removing rain streaks, we propose the additional edge loss to constrain the high-frequency components between the ground truth $I_{Clean}^\text{con}$ and the predicted rain-free image $I_{Dera}^\text{clean}$. The edge loss is defined as

$$L_{edge} = \sqrt{(\text{Lap}(I_{Clean}^\text{con}) - \text{Lap}(I_{Dera}^\text{clean}))^2 + \varepsilon^2}. \tag{2}$$

In Equation (2), $\text{Lap}(I_{Clean}^\text{con})$ and $\text{Lap}(I_{Dera}^\text{clean})$ denote the edge maps respectively extracted from $I_{Clean}^\text{con}$ and $I_{Dera}^\text{clean}$ via the Laplacian operator [13]. Then, the total loss function is given by

$$L = L_{con} + \lambda \times L_{edge}, \tag{3}$$

where the weight parameter $\lambda$ is empirically set to 0.05 to balance the loss terms.

### 4. Experiments and Discussions

We conduct extensive experiments on several synthetic and real-world rain image datasets [7, 41, 29] to evaluate the restoration performance of our proposed MSPFN as well as six state-of-the-art deraining methods. These representative methods include DerainNet [6], RESCAN [19], DIDMDN [40], UMRL [37], SEMI [31] and PreNet [26]. There is no unified training datasets for all competing methods in this paper, e.g. PreNet refers to JORDER [35] and uses 1254 pairs for training. UMRL refers to [40] and uses 12700 images for training. Therefore, directly taking the results from their papers is unfair and meaningless. To this end, we collect about 13700 clean/rain image pairs from [41, 7] for training our network as well as other competing methods for a fair comparison. In particular, these competing methods are retrained in the experiments with their publicly released codes and follow their original settings under the unified training dataset. Separately, the
Table 1. Dataset description. A total of 13712 clean/rain image pairs are used for training. There are additional 4300 labeled reference samples as well as 200 real-world scenarios for testing.

<table>
<thead>
<tr>
<th>Datasets</th>
<th>Training Samples</th>
<th>Testing Samples</th>
<th>Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rain1400 [7]</td>
<td>11200</td>
<td>2800</td>
<td>Test2800</td>
</tr>
<tr>
<td>Rain1800 [35]</td>
<td>1800</td>
<td>0</td>
<td>Rain1800</td>
</tr>
<tr>
<td>Rain2000 [41]</td>
<td>700</td>
<td>100</td>
<td>Test100</td>
</tr>
<tr>
<td>Rain100H [35]</td>
<td>0</td>
<td>100</td>
<td>Rain100H</td>
</tr>
<tr>
<td>Rain100L [35]</td>
<td>0</td>
<td>1200</td>
<td>Test1200</td>
</tr>
<tr>
<td>Rain12 [20]</td>
<td>12</td>
<td>0</td>
<td>Rain12</td>
</tr>
<tr>
<td>Real200 [25, 31]</td>
<td>0</td>
<td>200</td>
<td>Real200</td>
</tr>
<tr>
<td>RDR/RIS [15]</td>
<td>0</td>
<td>2395/2348</td>
<td>RDR/RIS</td>
</tr>
<tr>
<td>Total Count</td>
<td>13712</td>
<td>9343</td>
<td>-</td>
</tr>
</tbody>
</table>

Table 2. Evaluation of the basic components in our baseline MSPFN on Test100 dataset. We obtain the average inference time of deraining on images with size of $512 \times 384$.

<table>
<thead>
<tr>
<th>Model</th>
<th>Model</th>
<th>Model</th>
<th>Model</th>
<th>Model</th>
<th>Model</th>
<th>MSPFN</th>
</tr>
</thead>
<tbody>
<tr>
<td>PSNR</td>
<td>51.28</td>
<td>52.82</td>
<td>53.42</td>
<td>54.01</td>
<td>54.59</td>
<td>55.22</td>
</tr>
<tr>
<td>FSIM</td>
<td>0.676</td>
<td>0.684</td>
<td>0.691</td>
<td>0.704</td>
<td>0.710</td>
<td>0.720</td>
</tr>
<tr>
<td>SSIM</td>
<td>0.895</td>
<td>0.905</td>
<td>0.911</td>
<td>0.921</td>
<td>0.924</td>
<td>0.925</td>
</tr>
<tr>
<td>Ave. inf. time (s)</td>
<td>0.193</td>
<td>0.224</td>
<td>0.213</td>
<td>0.250</td>
<td>0.241</td>
<td>0.180</td>
</tr>
<tr>
<td>Par. (Millions)</td>
<td>5.53</td>
<td>11.70</td>
<td>2.29</td>
<td>11.78</td>
<td>5.60</td>
<td>8.45</td>
</tr>
<tr>
<td>15.22</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

detailed descriptions of the used datasets are tabulated in Table 1. In order to quantitatively evaluate the restoration quality, we adopt the commonly used evaluation metrics, such as Peak Signal to Noise Ratio (PSNR), Feature Similarity (FSIM) [42], and Structural Similarity (SSIM) [30].

4.1. Implementation Details

In our baseline, the pyramid levels are set to 3, i.e. the original scale, 1/2 scale and 1/4 scale. In CFM, the filter numbers of each recurrent Conv-LSTM are respectively set to 32, 64, and 128, corresponding to the gradually increasing resolution. The depths/numbers of FFM (M) and CAU (N) are set to 10 and 3, respectively. We use Adam optimizer with batch size of 8 for training on one NVIDIA Titan Xp GPU. The learning rate is initialized to $2 \times 10^{-4}$ and reduced by half at every 20000 steps till $1 \times 10^{-6}$. We train the network for 30 epochs with the above settings.

4.2. Ablation Studies

Validation on Basic Components. Using our baseline model ($M = 10, N = 3$), we design six comparison models to analyze the effects of the proposed basic modules (CFM and FFM), multi-scale pyramid framework, and multi-scale progressive fusion scheme on deraining performance. Quantitative results on Test100 dataset are listed in Table 2. From the results, our baseline MSPFN exhibits great superiority over its incomplete versions, including Model1 (single-scale framework with only the original input), Model2 (removing CFM from MSPFN), and Model3 (removing all FFMs from MSPFN), surpassing them by 0.73dB, 0.28dB, and 3.60dB (PSNR), respectively. Moreover, we construct Model4 by applying the fusion strategy in [8] to verify the effectiveness of the proposed multi-scale progressive fusion scheme. It is evident that MSPFN gains a significant improvement over Model4 by 0.54dB with an acceptable complexity increase. Model5 ($M = 5, N = 1$) and Model6 ($M = 6, N = 3$) are the simplified variants of MSPFN with smaller depths. When compared with the single-scale framework (Model1), Model5 has the approximately equal amount of parameters but achieves faster inference speed with the multi-scale pyramid framework. Model6 has the similar computation complexity but more parameters as compared with Model1. The results show that Model5 achieves the comparable performance while it’s a quarter more efficient. Model6 gains the better scores over Model1 by 0.32dB while keeping the similar computation complexity. We attribute these advantages to the effective cooperative representation of rain streaks among different pyramid layers and stages of the network.

Parameter Analysis on $M$ and $N$. We assess the influence of the depth of FFM (M) and the number of CAU (N) on deraining performance. Based on our baseline ($M = 10, N = 3$), we construct three comparison models, $i.e.$ MSPFN$_{M30N1}$, MSPFN$_{M33N2}$ and MSPFN$_{M50N5}$, while keeping approximately the same number of parameters. As shown in Table 3, the performance declines with the reduction of M. This indicates the important role of FFM for exploiting the multi-scale rain information in a progressive fashion. When increasing the number of CAU (MSPFN$_{M17N2}$), it yields a slight improvement (0.13dB), but with additional 30% of the parameters. We also add two models MSPFN$_{M30N1}$ and MSPFN$_{M50N1}$ for comparison. The former is designed to pursue a better deraining performance with more FFMs to enhance multi-scale fusion, while the latter is a lightweight model with smaller depth ($M = 5, N = 1$) and width (all filter channels = 32). Meanwhile, the strided convolution and deconvolution are employed twice in our proposed U-shaped residual attention block (URAB) of MSPFN$_{M50N1}$ to further alleviate the computation burden. As we expected, MSPFN$_{M30N1}$ achieves the best scores for all the metrics. MSPFN$_{M50N1}$ still obtains the acceptable performance, although being a much lighter network. Considering the tradeoff between efficiency and deraining performance, we set $M$ and $N$ to 17 and 1 respectively in the following experiments.

4.3. Comparisons with State-of-the-arts

4.3.1 Synthesized Data

We compare our MSPFN ($M = 17, N = 1$) with other six top-performing deraining methods [6, 19, 40, 37, 31, 26] on five synthetic datasets. Quantitative results are shown in Table 4. One can see that MSPFN achieves remarkable improvements over these state-of-the-art methods. For example, MSPFN surpasses DerainNet [6] and DIDMDN [40]...
Table 3. Evaluation of the depth of FFM (M), the number of CAU (N), as well as the model parameters on Test100 dataset. MSPFN_{M=N} denotes the model with M = a and N = b.

<table>
<thead>
<tr>
<th>Models</th>
<th>MSPFN_{M=30,N=1}</th>
<th>MSPFN_{M=17,N=2}</th>
<th>MSPFN_{M=13,N=2}</th>
<th>MSPFN_{M=10,N=3}</th>
<th>MSPFN_{M=8,N=5}</th>
<th>MSPFN_{M=5,N=1}</th>
</tr>
</thead>
<tbody>
<tr>
<td>PSNR</td>
<td>27.91</td>
<td>27.50</td>
<td>27.63</td>
<td>27.42</td>
<td>27.29</td>
<td>27.13</td>
</tr>
<tr>
<td>SSIM</td>
<td>0.879</td>
<td>0.876</td>
<td>0.877</td>
<td>0.874</td>
<td>0.869</td>
<td>0.867</td>
</tr>
<tr>
<td>SSIM</td>
<td>0.929</td>
<td>0.928</td>
<td>0.928</td>
<td>0.927</td>
<td>0.925</td>
<td>0.923</td>
</tr>
<tr>
<td>Par. (Millions)</td>
<td>21.81</td>
<td>13.35</td>
<td>17.20</td>
<td>13.63</td>
<td>13.22</td>
<td>14.56</td>
</tr>
</tbody>
</table>

Figure 5. Restoration results on synthetic datasets, including Rain100H, Rain100L, Test100, and Test1200.

by 9.01dB and 2.74dB, respectively, in terms of PSNR on Test1200 dataset. Visual results on different rain conditions (diverse rain streak orientations and magnitudes) are presented in Fig. 5. MSPFN exhibits impressive restoration performance on all scenarios, generating results with rich and credible image textures while removing main rain streaks. For other comparison methods, they tend to blur the image contents, or still leave some visible rain streaks. For example, only our MSPFN restores the clear and credible image details in the “Giraffe” image, while the competing methods fail to remove rain streaks and their results have obvious color distortion.

4.3.2 Real-world Data

We conduct additional comparisons on three real-world datasets, including Real200 [40], Rain in Driving (RID) and Rain in Surveillance (RIS) datasets [18], to further verify the generalization capability of MSPFN. RID and RIS cover 2495 and 2348 samples, collected from car-mounted cameras and networked traffic surveillance cameras in rainy days respectively. Moreover, we use another two quantitative indicators, Naturalness Image Quality Evaluator (NIQE) [23] and Spatial-Spectral Entropy-based Quality (SSEQ) [22], to quantitatively evaluate the reference-free restoration performance. The smaller scores of SSEQ and NIQE indicate better perceptual quality and clearer contents. The results are listed in Table 5. As expected, our proposed MSPFN has the best average scores on 200 real-world samples, outperforming the state-of-the-art deraining methods [19, 37, 26] by a large margin. Moreover, we show four representative deraining examples in Fig. 6 for visual comparison. In the last image, obvious rain streaks are observed in the results of other deraining methods, but our
### Table 4. Comparison results of average PSNR, SSIM and FSIM on several widely used rain datasets, including Rain100H, Rain100L, Test100, Test2800, and Test1200. **MSPFN (Ours)** denotes our model without the edge constraint in the loss function.

<table>
<thead>
<tr>
<th>Methods</th>
<th>Rain100H</th>
<th>Rain100L</th>
<th>Test100</th>
<th>Test2800</th>
<th>Test1200</th>
<th>Average</th>
</tr>
</thead>
<tbody>
<tr>
<td>DerainNet [6]</td>
<td>PSNR</td>
<td>SSIM</td>
<td>FSIM</td>
<td>PSNR</td>
<td>SSIM</td>
<td>FSIM</td>
</tr>
<tr>
<td></td>
<td>27.77/0.810/0.884</td>
<td>14.92/0.592/0.755</td>
<td>27.03/0.884/0.904</td>
<td>24.31/0.861/0.930</td>
<td>23.38/0.835/0.924</td>
<td>22.40/0.796/0.879</td>
</tr>
<tr>
<td>RESCAN [19]</td>
<td>25.00/0.8350/0.909</td>
<td>26.36/0.7860/0.864</td>
<td>29.80/0.881/0.919</td>
<td>31.29/0.904/0.952</td>
<td>50.51/0.882/0.944</td>
<td>28.50/0.857/0.917</td>
</tr>
<tr>
<td>DDMON [40]</td>
<td>22.56/0.8180/0.899</td>
<td>17.35/0.5240/0.746</td>
<td>25.23/0.7410/0.861</td>
<td>28.13/0.867/0.943</td>
<td>29.65/0.901/0.950</td>
<td>24.58/0.770/0.876</td>
</tr>
<tr>
<td>UMLR [39]</td>
<td>24.41/0.8290/0.910</td>
<td>26.01/0.8320/0.876</td>
<td>29.18/0.9230/0.940</td>
<td>29.97/0.905/0.955</td>
<td>50.55/0.916/0.955</td>
<td>28.02/0.880/0.927</td>
</tr>
<tr>
<td>SEMI [31]</td>
<td>22.35/0.7880/0.887</td>
<td>16.56/0.4860/0.692</td>
<td>25.03/0.8420/0.893</td>
<td>24.43/0.782/0.897</td>
<td>26.05/0.8220/0.917</td>
<td>22.86/0.7440/0.857</td>
</tr>
<tr>
<td>PreNet [26]</td>
<td>24.81/0.8510/0.916</td>
<td>26.77/0.8580/0.890</td>
<td>32.44/0.9500/0.956</td>
<td>31.75/0.916/0.956</td>
<td>31.36/0.911/0.955</td>
<td>29.42/0.897/0.934</td>
</tr>
<tr>
<td><strong>MSPFN (Ours)</strong></td>
<td>26.93/0.8650/0.924</td>
<td>28.33/0.8420/0.893</td>
<td>32.18/0.9280/0.939</td>
<td>32.70/0.9280/0.946</td>
<td>32.22/0.9140/0.958</td>
<td>30.51/0.895/0.934</td>
</tr>
<tr>
<td><strong>MSPFN</strong></td>
<td>27.50/0.8760/0.928</td>
<td>28.66/0.8600/0.890</td>
<td>32.40/0.9330/0.943</td>
<td>32.82/0.9300/0.966</td>
<td>32.39/0.9160/0.960</td>
<td>30.75/0.9030/0.957</td>
</tr>
</tbody>
</table>

**MSPFN** can well preserve more realistic and credible image details while effectively removing main rain streaks.

### Table 5. Comparison results of average NIQE/SSEQ on real-world datasets (**Real200**, **RID**, and **RIS**). The smaller scores indicate better perceptual quality.

<table>
<thead>
<tr>
<th>Methods</th>
<th>Rain100L</th>
<th>Rain100H</th>
<th>Test100</th>
<th>Test2800</th>
<th>Test1200</th>
<th>Average</th>
</tr>
</thead>
<tbody>
<tr>
<td>DerainNet [6]</td>
<td>PSNR</td>
<td>SSIM</td>
<td>FSIM</td>
<td>PSNR</td>
<td>SSIM</td>
<td>FSIM</td>
</tr>
<tr>
<td></td>
<td>14.92/0.494/0.671</td>
<td>17.35/0.524/0.746</td>
<td>25.23/0.741/0.861</td>
<td>28.13/0.867/0.943</td>
<td>29.65/0.901/0.950</td>
<td>24.58/0.770/0.876</td>
</tr>
<tr>
<td>RESCAN [19]</td>
<td>25.00/0.8350/0.909</td>
<td>26.36/0.7860/0.864</td>
<td>29.80/0.881/0.919</td>
<td>31.29/0.904/0.952</td>
<td>50.51/0.882/0.944</td>
<td>28.50/0.857/0.917</td>
</tr>
<tr>
<td>DDMON [40]</td>
<td>22.56/0.8180/0.899</td>
<td>17.35/0.5240/0.746</td>
<td>25.23/0.7410/0.861</td>
<td>28.13/0.867/0.943</td>
<td>29.65/0.901/0.950</td>
<td>24.58/0.770/0.876</td>
</tr>
<tr>
<td>UMLR [39]</td>
<td>24.41/0.8290/0.910</td>
<td>26.01/0.8320/0.876</td>
<td>29.18/0.9230/0.940</td>
<td>29.97/0.905/0.955</td>
<td>50.55/0.916/0.955</td>
<td>28.02/0.880/0.927</td>
</tr>
<tr>
<td>SEMI [31]</td>
<td>22.35/0.7880/0.887</td>
<td>16.56/0.4860/0.692</td>
<td>25.03/0.8420/0.893</td>
<td>24.43/0.782/0.897</td>
<td>26.05/0.8220/0.917</td>
<td>22.86/0.7440/0.857</td>
</tr>
<tr>
<td>PreNet [26]</td>
<td>24.81/0.8510/0.916</td>
<td>26.77/0.8580/0.890</td>
<td>32.44/0.9500/0.956</td>
<td>31.75/0.916/0.956</td>
<td>31.36/0.911/0.955</td>
<td>29.42/0.897/0.934</td>
</tr>
<tr>
<td><strong>MSPFN (Ours)</strong></td>
<td>26.93/0.8650/0.924</td>
<td>28.33/0.8420/0.893</td>
<td>32.18/0.9280/0.939</td>
<td>32.70/0.9280/0.946</td>
<td>32.22/0.9140/0.958</td>
<td>30.51/0.895/0.934</td>
</tr>
<tr>
<td><strong>MSPFN</strong></td>
<td>27.50/0.8760/0.928</td>
<td>28.66/0.8600/0.890</td>
<td>32.40/0.9330/0.943</td>
<td>32.82/0.9300/0.966</td>
<td>32.39/0.9160/0.960</td>
<td>30.75/0.9030/0.957</td>
</tr>
</tbody>
</table>

**4.3.3 Other Applications**

Image deraining under complex weather conditions can be considered as an effective enhancement of image content. It can potentially be incorporated into other high-level vision systems for applications such as object detection and segmentation. This motivates us to investigate the effect of restoration performance on the accuracy of object detection and segmentation based on some popular algorithms, e.g., YOLOv3 [25], Mask R-CNN [12], and RefineNet [21]. To this end, we randomly select a total of 850 samples from COCO [3] and BDD [38] datasets to create three new synthetic rain datasets COCO350 (for detection), BDD350 (for detection), and BDD150 (for segmentation) through Photoshop. These rain images are of diverse streak orientations and magnitudes, and at the same time have complex...

[Figure 6. Comparison results on four real-world scenarios with RESCAN [19], UMRL [37] and PreNet [26].]
imaging conditions such as night scenes. By using our proposed deraining algorithm MSPFN as well as other top-performing deraining methods [19, 26], the restoration procedures are directly implemented on these three datasets to produce the rain-free images. And then we apply the public available pre-trained models of YOLOv3 (for detection), Mask R-CNN (for instance segmentation), and RefineNet (for semantic segmentation) to perform the downstream tasks. Qualitative results, including the deraining performance as well as the precision of the subsequent detection and segmentation tasks, are tabulated in Table 6. In addition, visual comparisons are shown in Fig. 7.

It is obvious that rain streaks can greatly degrade the detection accuracy and segmentation precision, night scenarios in particular, i.e. by missing targets and producing low detection or segmentation confidence (mean pixel accuracy (mPA) and mean Intersection of Union (mIoU)). In addition, the detection precision of the produced rain-free images by MSPFN shows a notable improvement over that of original rain inputs by nearly 10%, and MSPFN achieves the best results of 52.96% mPA as well as 35.90% mIoU for semantic segmentation task on BDD150. When compared with other top-performing deraining models, the rain-free images generated by MSPFN show more credible contents with more details, which effectively promote the detection and segmentation performance. Moreover, we also evaluate our lightweight deraining model MSPFN∗ with lighter depth (M = 5, N = 1) and width (with all filter channels of 32) since computation efficiency is crucial for mobile devices and applications require real-time throughput such as autonomous driving. MSPFN∗ still achieves competitive performance compared with other models [19, 26] while it’s a half more efficient in terms of inference time.

5. Conclusion

In this paper, we propose a novel multi-scale progressive fusion network (MSPFN) to exploit the multi-scale rain information to cooperatively represent rain streaks based on the pyramid framework. To achieve this goal, we design several basic modules (CFM, FFM and RM) along with our proposed multi-scale progressive fusion mechanism to explore the inherent correlations of the similar rain patterns among multi-scale rain streaks. Consequently, our predicted rain streak distribution is potentially more correct due to the collaborative representation of rain streaks across different scales. Experimental results on several synthetic deraining datasets and real-world scenarios, as well as several downstream vision tasks (i.e. object detection and segmentation) have shown great superiority of our proposed MSPFN algorithm over other top-performing methods.

6. Acknowledgement

This work is supported by National Key R&D Project (2016YFE0202300) and National Natural Science Foundation of China (U1903214, 61671332, U1736206, 41771452, 41771454, 61971165), and Hubei Province Technological Innovation Major Project (2019AAA049, 2018CFA024).
References


