Multi-level Memory Compensation Network for Rain Removal via Divide-and-Conquer Strategy

Kui Jiang, *Student Member, IEEE,* Zhongyuan Wang, *Member, IEEE,* Peng Yi,
Chen Chen, *Member, IEEE* Xiaofen Wang, Junjun Jiang, *Member, IEEE* Zixiang Xiong, *Fellow, IEEE*

***Abstract*—Recently an increasing number of algorithms have been proposed for rain streak removal. However, most existing methods ignore the discrepancy in removing rain streaks from background contents with different texture richness (frequencies). They adopt a unified scheme to learn the final distribution of rain streaks directly, thus sacrificing the representative accuracy of rain information. To this end, this paper leverages the *divide-and-conquer* strategy for rain streak removal by decomposing the learning task into several subproblems according to the levels (frequencies) of texture richness in background contents. Particularly, we construct a novel multi-level memory compensation network (MLMCN) for rain streak removal. It achieves a promising solution by individually handling these subproblems under the specific texture richness via several parallel subnetworks. Each subnetwork takes as input a specific sub-sampled image, sampled from the original rain ones via the Gaussian kernel, to individually learn one of the sub-distributions of the rain information. We thus produce a high-quality rain-free image by subtracting the predicted rain information from multiple subnetworks in turn. We experimentally show that our proposed MLMCN outperforms the existing deraining methods in terms of quantitative indicators, visual effects on several benchmark datasets, and the high-level object detection task.**

***Index Terms*—Rain streak removal, divide-and-conquer, background compensation, recurrent learning, residual learning.**

I. Introduction

Images or videos captured in the rain weather condition are contaminated by rain streaks, which cause significant detrimental effects on image contents. Especially for many common computer vision applications, such as human detection [1], [2], object tracking and recognition [3], as well as autonomous driving [4], they generally rely on clean and credible inputs. The poor visual effect may lead to the failure of those high-level vision tasks. Hence it is crucial to restore a clear and credible image from its rain-contaminated counterpart.

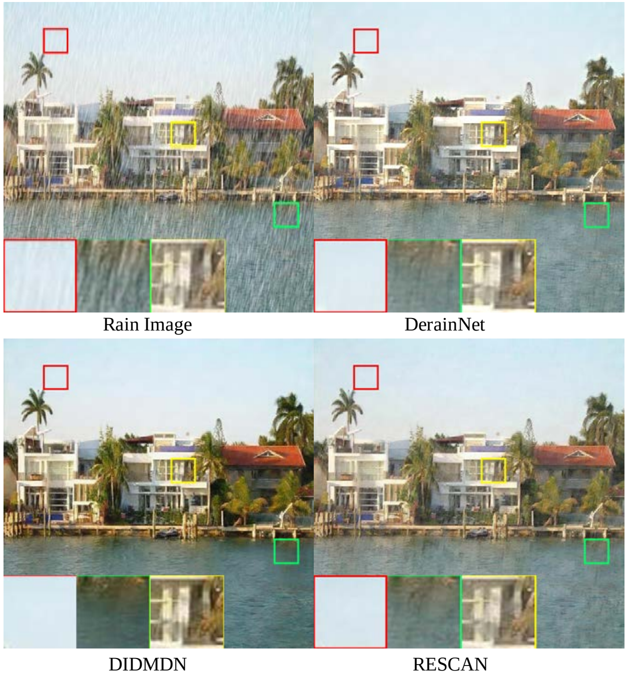


Fig. 1. An example demonstrates that a unified model cannot cope with the restoration tasks under diverse levels of texture richness in background contents. The results are obtained from DerainNet [13], DIDMDN [14], and RESCAN [15]. The components in red, green and yellow boxes represent background contents of different texture richness, respectively. These methods have high consensus levels in removing rain streaks from smooth regions (noted by the red box), but leave visible rain streaks in regions with more details (noted by the green and yellow boxes).

Past decades have made continuous progresses for delivering promising solutions on the rain streak removal task [5], [6], [7]. Previous optimization-based methods [5], [8], [9] take certain physical characteristics of rain steaks into consideration, *e.g.*, photometric appearance [10], geometrical features [11], and local structure correlations [12], and separate rain streaks from the background images using some prior knowledge. However, it is somewhat effective for these methods to learn the complex distribution of rain information in real-world scenarios with the constrained linear model.

Inspired by the recent success of deep learning on other low-level vision tasks [16], [17], many works [18], [19], [20], [21] take advantages of convolutional neural network (CNN) [22] for the deraining task. For example, Fu *et al.* [13] pioneer a deep-learning based network to remove rain streaks from

This work is supported by National Natural Science Foundation of China (U1903214, 61671332, U1736206, 62071339, 62072350, 61971165, 62072347), and Hubei Province Technological Innovation Major Project (2019AAA049, 2020BAB018). (*Corresponding author: Zhongyuan Wang*)

K. Jiang, Z. Wang, P. Yi, and X. Wang are with the National Engineering Research Center for Multimedia Software, School of Computer Science, Wuhan University, Wuhan, 430072, China. E-mail: kuijiang_1994@163.com; wzy_hope@163.com; 2017202110008@whu.edu.cn; wxiaofen@whu.edu.cn.

C. Chen is with the Department of Electrical and Computer Engineering, University of North Carolina at Charlotte. E-mail: chen.chen@uncc.edu.

J. Jiang is with the School of Computer Science and Technology, Harbin Institute of Technology, Harbin 150001, China, and is also with the Peng Cheng Laboratory, Shenzhen, China. E-mail: jiangjunjun@hit.edu.cn.

Z. Xiong is with the Department of Electrical and Computer Engineering, Texas A&M University, College Station, TX 77843 USA (E-mail: zx@ece.tamu.edu).

background images with a three-layer CNN. Limited by the number of the neurons, it produces inferior results. To promote the representation of rain streaks, Zhang *et al.* [14] consider the rain density, and construct an elaborate multi-task CNN for joint rain density estimation and deraining. However, the restoration performance is overwhelmingly dependent on the detection accuracy of rain density. More recent works [23], [24] employ encoder-decoder or pyramid frameworks to fully exploit the multi-scale features to promote modeling capability, and achieve considerable performance gains. However, the proposed multi-scale fusion schemes in these works tend to promote the deraining performance from the perspective of the feature fusion and representation, but require great memory and calculation cost. Moreover, separating rain streaks from background is essentially a complex unmixing problem due to the rain streaks acting on various texture richness levels and heavy rain condition in particular. Albeit showing significant superiority over traditional deraining algorithms [5], [12], these methods [13], [24], [25] adopt a unified model to cover all the scenarios (low-density or heavy rain conditions, simple or detailed background contents), thus hindering the flexibility and scalability for the refined representation. Taking Fig. 1 as a comparison example, these representative methods [13], [14] show high consensus levels in removing rain streaks from smooth regions (low-level texture richness), but with visible rain streaks remained on detailed regions (high-level texture richness). The phenomenon above can be explained as that the difficulty of separating rain streaks from background contents varies with the texture richness. *Thus the natural question that arises is how to effectively simulate the distribution of rain streaks according to the specifical level of texture richness.*

To solve such a complex issue, the divide-and-conquer strategy [26], [27] provides a referential scheme. As the name implies, it divides the original complex task into several subproblems, and then uses base algorithms to solve these subproblems individually [28]. Specifically, there are two methods to complete this task. One is that smooth regions and textured regions are processed separately for each rainy input. The other one decomposes the learning task into multiple levels which correspond to the image pyramid. In this study, we apply the second scheme for rain streak removal to cope with the diverse texture richness in background contents. We sample the rain input through the Gaussian sampling kernel to construct the rain image pyramid, where the sub-samples have diverse levels (frequencies) of texture richness in background contents and rain streaks. We therefore construct several parallel subnetworks to learn the individual sub-distribution of rain information in each pyramid level. The outcomes of these subnetworks can be combined to yield a final solution to the deraining task.

To further improve the performance, we construct a novel memory compensation block (MCB) by incorporating recurrent calculation and residual learning to encode the global feature correlations. Moreover, as the basic module in the subnetwork, MCB can simultaneously represent rain streaks and compensate background features through a residual memory module (RMM) and a background compensation module (BCM). More precisely, the former is utilized to characterize the rain information, where global texture correlations across the spatial dimension can be fully exploited through the recurrent processing mechanism. The latter is used to estimate the error-removing image textures in RMM and learns the compensated image details via the residual feedback. This technique contributes more to the refined representation of rain information while improving the fidelity of image textures. For simplicity, these subnetworks share the same modules, but allow for different depths to satisfy the learning tasks under background contents with specific texture richness, resulting in a novel multi-level memory compensation network (MLMCN) for single image deraining.

In summary, the major contributions of this paper are summarized as follows:

1) We propose a novel multi-level memory compensation network (MLMCN) for rain streak removal following the divide-and-conquer paradigm. We decompose the learning task into several subproblems according to the levels (frequencies) of texture richness in background contents (the rain image pyramid), and accordingly construct multiple parallel subnetworks to sovle these subproblems individually.
2) We devise a novel memory compensation block (MCB) for accurate rain information estimation and texture preserving. It combines the recurrent calculation and residual learning to fully exploit the global textural correlations across spatial dimension for feature representation. Meanwhile, a residual feedback mechanism is used for a further refinement.
3) Apart from achieving the state-of-the-art deraining performance in terms of the conventional quantitative metrics (*e.g.*, PSNR and SSIM), our method also outperforms the current representative methods on joint image deraining and object detection task.

The remainder of this paper is organized as follows. In Section II, we review the existing works related to our study, such as image/video deraining and recurrent neural network (RNN). We describe the proposed multi-level memory compensation network (MLMCN) in detail in Section III. The experimental results and discussions are given in Section IV. Finally, we conclude this paper in Section V.

II. Related Work

In the last few years, various learning-based approaches have been widely applied for rain streak removal [29], [30], [21], achieving a significant improvement in image quality. A comprehensive review is beyond the scope of this work and we discuss the most related ones in this section.

A. Single Image Deraining

Over the years, a growing number of studies [31], [32], [33], [34] have been developed to promote the visibility and readability of image contents captured under rain conditions, including single-image-based and video-based methods [35], [36], [37]. We focus on the single image rain removal within this work since it is more challenging – only relying on limited spatial information to separate rain streaks from the complex

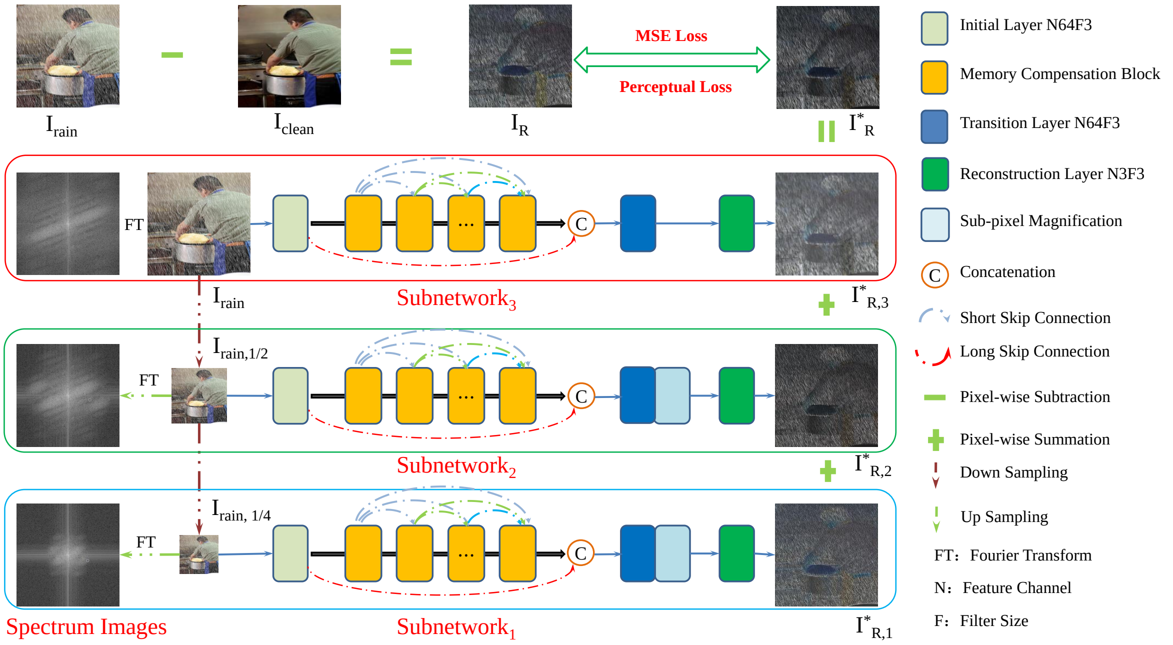


Fig. 2. Outline of the proposed multi-level memory compensation network (MLMCN). We set N to 3 as an example, corresponding to a three-level image pyramid with different texture richness in background contents and rain streaks (please refer to the spectrum images, where the lower frequencies concentrate on the center). We therefore divide the learning task into multiple subproblems according to the levels of texture richness (the number of the subsamples), and construct several parallel subnetworks (Subnetwork$_1$, Subnetwork$_2$, Subnetwork$_3$) to individually estimate the sub-distribution of rain information in each pyramid level. The solutions of subnetworks ($I^*_{R,1}$, $I^*_{R,2}$, $I^*_{R,3}$) can be combined to yield a final solution (I^*_R).

background image. For the early shallow-learning based works [12], [8], they usually produce results visually vulnerable and inconsistent with real scenes using linear transformation model and hand-crafted priors. With more powerful modeling capabilities than shallow models, CNN-based algorithms gradually emerge with promising performances in recent years. For example, Fu *et al.* [13] firstly propose a shallow CNN to estimate and remove rain streaks from rain-contaminated images. Next, Yang *et al.* [19] present a multi-task deep learning framework by combining dilated convolution and recurrent learning strategy to simultaneously detect and eliminate the rain streaks. Moreover, Zhang *et al.* [38] learn a set of generic sparsity-based and low-rank representation-based convolutional filters for efficiently representing background and rain streaks in an image. To bridge the gap of the deraining performance between synthetic data and real-world data, researchers [39], [40], [41] propose to construct new paired rain image datasets by considering the visual effects and naturalness. Although these aforementioned methods have gained significant improvements over the conventional deraining algorithms [12], they ignore the discrepancy of separating rain streaks from background contents with different texture richness. Overall, using a unified learning scheme to cover all scenarios, complex textured regions in particular, inevitably sacrifices the accuracy.

B. Recurrent Learning

Over the past few years, RNN and its variants [42] have been widely applied to sequential tasks [43], [44], [45] due to their powerful ability of processing historical information with the recursive memory. By modeling the information flow of context textures, RNN [46] is especially suitable for processing strongly correlated information in time or spatial series. Hence, it is useful to integrate recurrent learning into deraining models for rain streak removal to fully exploit the global correlation of features. For example, Qian *et al.* [32] inject visual attention into both the generative and discriminative networks to guide the network to focus on the rain-contaminated regions. In particular, they use a convolutional LSTM unit to integrate the features from preceding layers and generate the 2D attention maps. Combining the recurrent calculation and squeeze-and-excitation (SE) strategy, Li *et al.* [15] present a REcurrent SE Context Aggregation Net (RESCAN) for rain streak removal in a stage-wise manner. Moreover, the dilated convolutional layer is also integrated to acquire more contextual information for representing rain streaks accurately. Recently, Ren *et al.* [47] propose a progressive recurrent network to take advantage of recursive and recurrent computation for a better restoration performance by repeatedly unfolding a shallow residual network. Albeit achieving impressive results, these deraining methods [32], [48] adopt a unified mapping model to cope with all conditions and ignore the discrepancy in removing rain streaks from background contents with different texture richness. Moreover, the repeated residual learning paradigm naturally causes cumulative errors (removing background details), thus producing over-smoothing deraining results. To overcome these problems, we introduce the divide-and-conquer paradigm for rain streak removal by decomposing the learning task into several subproblems according to the level of texture richness in background contents. We further construct a novel multi-level memory compensation network (MLMCN), in which we achieve the promising solution by individually handling these subproblems under the specific texture richness via several parallel subnetworks.

III. Proposed Method

In this section, we present the multi-level memory compensation network (MLMCN) in detail. Meanwhile, the base modules in MLMCN are also described in this part, including the memory compensation block (MCB) and its basic components: residual memory module (RMM) and background compensation module (BCM). As shown in Fig. 2, MLMCN takes a group of sub-sampled rain images (rain image pyramid) as inputs, and individually solves the restoration subproblems regarding background contents with specific texture richness through several parallel subnetworks.

A. Architecture and Model Optimization

An observed rain image I_{rain} can be roughly decomposed into the clean background image I_{clean} and its corresponding residual rain information I_R using the following formula

$$I_R = I_{rain} - I_{clean}. \tag{1}$$

Our goal is to train an effective and robust model to simulate the rain distribution I_R^*, similar to the real pattern I_R. We then obtain the resulting high-quality rain-free image I_{derain} by subtracting I_R^* from the rainy input I_{rain}.

As mentioned before, we adopt the divide-and-conquer strategy for rain streak removal by decomposing the learning task into N subproblems (N is set to 3 in the example of Fig. 2.) based on the levels of texture richness in the background contents (the layers of rain image pyramid). More specifically, given a rainy image with the size of $W \times H \times C$, we define an initial Gaussian kernel first, such as $[[1/k, 0, 0], [0, 1/k, 0], [0, 0, 1/k]]$, where k denotes the amplification factor. And then, the Gaussian kernel serves as the convolution kernel to sample the input image with the special stride k, returning a low-resolution image with the size of $W/k \times H/k \times C$. Thus we generate multiple sub-sampled images from the original rainy input via different k values, which contain the same image contents but with different texture richness. To solve these subproblems independently, our proposed MLMCN employs N parallel subnetworks, each taking one sub-sampled image as input to learn the sub-distribution of rain information under the specific texture richness. For convenience, these subnetworks have the same structure but with different depths of background compensation module (BCM) to tackle the restoration tasks with different texture richness. Therefore, for simplicity, we use subnetwork$_3$ as a representative example to present the framework in detail. As shown in Fig. 2, subnetwork$_3$ can be roughly decomposed into three parts: initial feature extraction, rain information modeling, and residual rain image reconstruction. In particular, except the subnetwork$_3$ taking the raw rain image as input, other subnetworks adopt the sub-sampled rain images as input. Motivated by [49], we adopt one initial convolutional layer to extract the shallow features F_{ini} from the rain input I_{rain} in the first part. This process can be formulated as

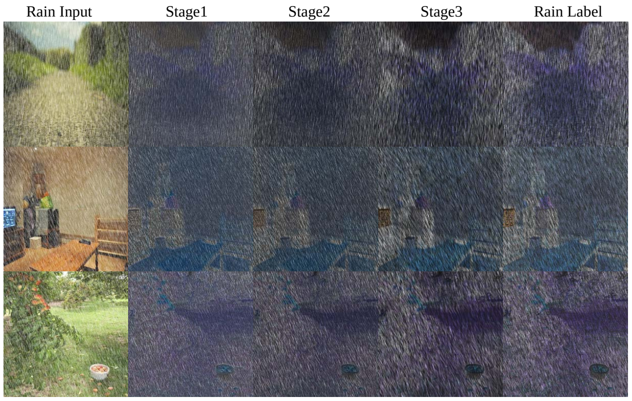


Fig. 3. The investigation of the cumulative residual rain images from multiple subnetworks. By subtracting the clean background image I_{clean} from its contaminated input I_{rain}, we obtain the real rain label I_R. Stage1 denotes the rain information ($I_{R,1}^*$) estimated from subnetwork$_1$. By cumulating the results of subnetwork$_1$ and subnetwork$_2$, we obtain the cumulated residual rain image ($I_{R,1}^*$, $I_{R,2}^*$) in Stage2. The result in Stage3 refers to the final regressed residual rain image (I_R^*) cumulated via three subnetworks.

$$F_{ini} = H_{ini}(I_{rain}), \tag{2}$$

where $H_{ini}(\cdot)$ denotes the initial convolutional function used to obtain the initial estimation of rain information and project the input from image space into feature space. Then, F_{ini} is transmitted into the memory compensation block (MCB) for a deep representation of rain streaks. The feature extraction in the first MCB can be described as

$$F_{MCB,1} = H_{MCB,1}(F_{ini}), \tag{3}$$

where $F_{MCB,1}$ denotes the rain information extracted via $H_{MCB,1}(\cdot)$ in the first MCB.

A deeper model constructed with the skip or dense connections can improve the representation of image contents [49], [50]. Therefore, we construct a deep CNN for a fine representation of rain streaks by stacking **M** MCBs with skip connections, where information can be effectively propagated among layers and memory stages during training. Thus the final MCB benefits from the multi-scale outputs of previous blocks, and generates the high-level rain features $F_{MCB,M}$. Following that, $F_{MCB,M}$ along with the shallow features F_{ini} are concatenated to obtain the fused features via a transition layer, and then projected into the image space to regress the residual rain information $I_{R,3}^*$ in the current texture richness level. The aforementioned procedures in subnetwork$_3$ are described as follows

$$I_{R,3}^* = H_R(H_C(F_{MCB,M}, F_{ini})), \tag{4}$$

where $H_C(\cdot)$ and $H_R(\cdot)$ denote the functions of concatenation operation and rain image reconstruction, respectively.

Inspired by [51], there are additional sub-pixel magnification layers $H_{SP}(\cdot)$ for the remaining $N-1$ subnetworks (subnetwork$_1$ and subnetwork$_2$ in Fig. 2), which learn an array of upscaling filters to upscale the final LR feature maps into the HR output layer via a rearrangement operation. Thus the

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JSTSP.2021.3052648, IEEE Journal of Selected Topics in Signal Processing

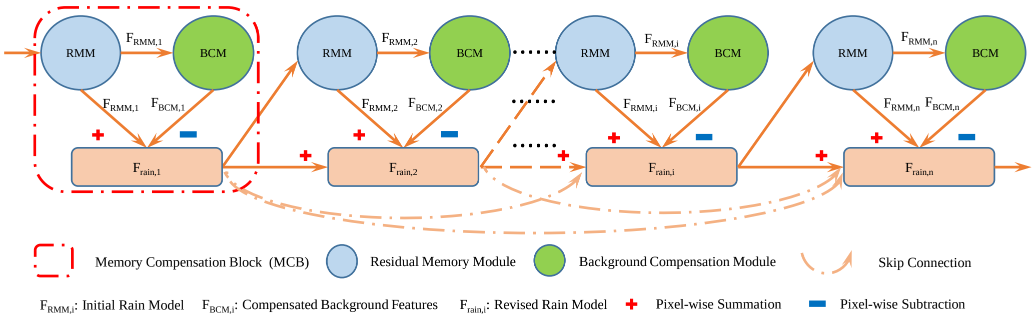


Fig. 4. The pipeline of the constructed residual dense memory framework and the proposed memory compensation block (MCB). The component in the red box denotes a single MCB, including a residual memory module (RMM) and a background compensation module (BCM) which are used to estimate and rectify the rain model respectively. By combining the recursive memory with the residual learning strategy, we form a novel residual dense memory framework for rain information estimation.

reconstruction modules in other subnetworks are represented as

$$I^*_{R,j} = H_R(H_{SP}(H_C(F_{MCB,M}, F_{ini}))), 1 \le j<N. \quad (5)$$

Finally, the entire restoration procedure for rain image in Fig. 2 can be defined as

$$I_{derain} = I_{rain} - I^*_R, I^*_R = \sum_{j=1}^{N} I^*_{R,j}. \quad (6)$$

In Eq. (6), I^*_R denotes the predicted residual rain image cumulated via N subnetworks. $I^*_{R,j}$ refers to the j_{th} element in I^*_R, estimated by subnetwork$_j$. Thus we generate the rain-free result I_{derain} by subtracting I^*_R from the rain image I_{rain}. We can approach the optimal solution to real rain streak distribution by individually dealing with all subproblems via multiple parallel subnetworks. Fig. 3 exhibits the predicted residual rain images cumulated by multiple subnetworks.

In order to obtain the estimation I^*_R of real rain streaks distribution and generate the corresponding high-quality rain-free image I_{derain} from the rain input I_{rain}, we use a robust loss function [52] for training. In particular, we adopt the holistic label (noted as the examples I_R in Fig. 2) to constrain these three subnetworks in order to generate the residual rain image I^*_R that is consistent with the rain label I_R. The loss function in our method is formulated as

$$L_{derain} = \sqrt{(I_R - I^*_R)^2 + \varepsilon^2}. \quad (7)$$

In Eq. (7), we empirically set the penalty coefficient ε as 10^{-3}.

To further improve the restoration quality, we introduce the perceptual loss to enforce the consistency of textures and luminance between I_{derain} and I_{clean}, producing results more faithful to the ground truth. The perceptual loss function is defined as

$$L_{per} = \sqrt{(F_{clean,i} - F_{derain,i})^2 + \varepsilon^2}, \quad (8)$$

where $F_{derain,i}$ and $F_{clean,i}$ refer to the extracted features from the predicted rain-free image and the ground truth by VGG19 [53], respectively. Finally, the total objective function is given by

$$L = \lambda \times L_{per} + L_{derain}. \quad (9)$$

We use the weight parameter λ to balance the loss terms. It is empirically set to 0.05 to guide the rain-free image generation in global appearance and high-level feature representation.

B. Memory Compensation Block

We hereby present the memory compensation block (MCB) in detail. Our idea of designing this module is to promote the estimation accuracy of rain information while keeping the image fidelity. As shown in Fig. 4, we combine the recurrent calculation and residual learning, and propose a novel memory compensation block (MCB) to simulate the rain streak distribution. MCB is composed of a residual memory module (RMM) and a background compensation module (BCM). The former is applied to represent the rain information via the global residual learning, and the latter is designed to rectify the estimated rain streak model from RMM by learning the remaining background details as compensation. This feedback scheme allows the network to simultaneously model rain streaks and compensate background features. These procedures above in the i_{th} MCB are formally expressed as

$$\begin{aligned} F_{RMM,i} &= H_{RMM}(F_{MCB,i-1}), \\ F_{BCM,i} &= H_{BCM}(F_{RMM,i}), \\ F_{rain,i} &= F_{RMM,i} - F_{BCM,i}, \\ F_{MCB,i} &= F_{rain,i} + F_{MCB,i-1}. \end{aligned} \quad (10)$$

$F_{RMM,i}$ denotes the estimated rain information via $H_{RMM}(\cdot)$ in RMM. In order to stabilize training and improve the fidelity of background textures, we adopt BCM to estimate the remaining background features in RMM, leading to the compensated components $F_{BCM,i}$. Thus we can produce the rectified rain information $F_{rain,i}$ and obtain the output $F_{MCB,i}$ of the current MCB by adding its input $F_{MCB,i-1}$ in a pixel-wise manner.

Thanks to the powerful capability of dense networks [15], [50] on feature extraction and representation, we construct the subnetworks with the residual dense architecture as shown in Fig. 4. By stacking MCBs in a cascaded manner via short and long skip connections, the information can be propagated among layers and memory stages, thereby helping the gradient descent and optimization.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JSTSP.2021.3052648, IEEE Journal of Selected Topics in Signal Processing

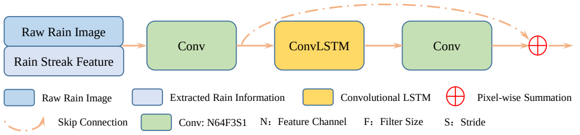


Fig. 5. The pipeline of the proposed residual memory module (RMM). By integrating ConvLSTM into the residual block, we construct the residual memory module to exploit the long-range spatial interdependence.

*1) **Residual Memory Module**:* RNN-based methods have recently achieved impressive performance in capturing long-range interdependence from sequential samples [54], [55]. Thus, researchers incorporate recurrent computation into image processing tasks to exploit the textural correlations across spatial series [47].

To achieve the similar goal, we construct a novel residual memory module (RMM) to exploit the global rain information (*e.g.*, similar appearance and patterns across spatial resolution) based on ConvLSTM. As shown in Fig. 5, the proposed RMM is essentially a residual ConvLSTM unit, including a ConvLSTM and two convolutional layers. The feature extraction procedures in RMM can be formulated as

$$X_t = H_{cat}(I_{rain}, X), \tag{11}$$

$$\begin{aligned} i_t &= \sigma\left(W_{xi} \otimes X_t + W_{hi} \otimes H_{t-1} + W_{ci} \circ C_{t-1} + b_i\right), \\ f_t &= \sigma\left(W_{xf} \otimes X_t + W_{hf} \otimes H_{t-1} + W_{cf} \circ C_{t-1} + b_f\right), \\ C_t &= f_t \circ C_{t-1} + i_t \circ tanh\left(W_{xc} \otimes X_t + W_{hc} \otimes H_{t-1} + b_c\right), \\ o_t &= \sigma\left(W_{xo} \otimes X_t + W_{ho} \otimes H_{t-1} + W_{co} C_t + b_o\right), \\ H_t &= o_t \circ tanh\left(C_t\right) \end{aligned} \tag{12}$$

$$F_{RMM} = H_{res}(o_t) + X_t. \tag{13}$$

In Eq. (11), $H_{cat}(\cdot)$ denotes the transition layer, which takes the preceding rain estimation information along with the rain image I_{rain} as input in a concatenated manner. It is practical and effective to perform the re-extraction of raw rain input I_{rain} at different stages to obtain the fused features X_t [47]. Then given the input X_t in Eq (12), we recurrently revise the estimation of the rain streak distribution in current RMM with the collaboration of the gate units (the input gate i_t, the forget gate f_t, and the output gate o_t), the cell body C_t, and the hidden state in the t_{th} state via Hadamard product ($\circ$) and convolution operation ($\otimes$). W and b are the weight parameters and bias. The estimated rain information o_t and the updated cell C_t can be determined by the preceding features X_t and the previous hidden states H_{t-1}.

*2) **Background Compensation Module**:* Although previous deraining methods [15], [25] have significantly improved the restoration performance over the traditional approaches [6], they tend to generate over-smoothing results. Specifically, they mistakenly remove the background details while wiping out the rain streaks from rain-contaminated images, thus causing the loss of high-frequency information.

We construct a background compensation module (BCM) to estimate the error-removing background details and rectify the estimated rainy model with compensated components to promote image texture fidelity. As shown in Fig. 6, our proposed BCM consists of G residual units along with a group of skip connections, thus allowing the network to share the feature maps among layers and different units. Given the predicted rain streak distribution F_{RMM} by RMM, we first adopt a strided convolutional layer to extract features and project them onto the LR space to reduce computation and memory footprint by using the following formulation:

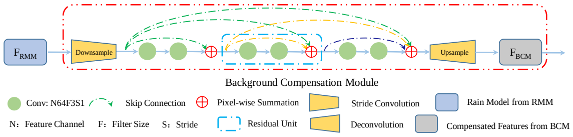


Fig. 6. The pipeline of the proposed background compensation module (BCM). We can effectively estimate the error-removing image details and return the compensated features through the successive residual dense framework. Besides, the up-down structure can greatly alleviate the computation burden.

$$F_{down} = H_{down}(F_{RMM}). \tag{14}$$

Then the output $F_{down,i}$ is transmitted to the residual units for a deep extraction to learn the compensated components. The aforementioned procedure can be expressed as

$$F_{BCM} = H_{up}(H_{res,G}(F_{down})), \tag{15}$$

where $H_{res,G}(\cdot)$ denote the feature extraction functions of G residual units for estimating the lost high-frequency details. We then apply a deconvolutional layer $H_{up}(\cdot)$ to integrate these features and project them into the original resolution space to generate the compensated components F_{BCM}.

IV. Experiments and Discussions

This section evaluates the restoration performance of our proposed MLMCN qualitatively and quantitatively, including ablation studies on basic modules and comparison experiments with current top-performing deraining methods.

A. Implementation Details

1) Comparison Methods: The performance of the proposed MLMCN method is compared against several representative deraining methods on both synthetic and real-world datasets, including the previous deep-learning based methods (DerainNet [13] and JORDER [19]), the high-performance deraining approaches (RESCAN [15] and DIDMDN [14], as well as more recent deraining methods (UMRL [25], SEMI [56] and PreNet [47]). For the sake of fairness, we retrain these models on the unified dataset with publicly available codes provided by authors except for JORDER [19]. We directly use the pre-trained model in [19] for testing due to the lack of public training codes. In particular, we retrain and evaluate UMRL [25] without adopting the cycle spinning strategy for fairness since other comparison methods do not use this procedure.

2) Data Collection: In this study, we collect about 13712 clean/rain image pairs with different rain conditions from [57], [30], [6] to construct a mixed dataset for training our network as well as other competing methods for a fair comparison since there is no unified training datasets for all competing methods. For example, JORDER [19] uses 1254 pairs for training while UMRL follows [14] and uses 12700 images for training. Meanwhile, several widely used synthetic rain datasets, including Test100 [57], Rain100H and Rain100L [19], Test1200 [14], and Test2800 [30], are used for evaluation, which are of diverse contents, intensities, and scales of rain streaks.

3) Experimental Setup: In our baseline MLMCN, the restoration task is decomposed into N subproblems (N is empirically set to 3 in this work to balance the restoration performance and efficiency) corresponding to the level of texture richness in background contents (the level of the rain image pyramid). In particular, the depths M of MCB in these three subnetworks are set to 3, 5, and 8, respectively, in order to cope with the restoration tasks with different texture richness in background contents. In addition, the number G of the residual units in BCM is empirically set to 3 by considering both the deraining performance and efficiency. For the training samples, before packing them into our proposed model, we coarsely crop them into small image patches with a size of 96×96 pixels without overlapping to obtain the sample pairs (about 137,000 image pairs). We set the batch size to 8, and the learning rate is initialized as 5×10^{-4} and reduced by half every 20000 steps till 1×10^{-5}. After 50 epochs less than 850K iterations on the training dataset, we obtain the optimal solution of our baseline MLMCN with the above settings. The implementation is based on an NVIDIA Titan Xp GPU and an Intel I7-8700 CPU.

4) Evaluation Criteria: The widely used evaluation metrics, including Peak Signal to Noise Ratio (PSNR), Feature Similarity (FSIM) [58], Structural Similarity (SSIM) [59]), and Perceptual Similarity (PSIM) [60], are utilized to evaluate the restoration performance of our proposed MLMCN and other top-performing deraining methods on synthetic images. Moreover, we introduce two additional reference-free quantitative indicators, Naturalness Image Quality Evaluator (NIQE) [61] and Spatial-Spectral Entropy-based Quality (SSEQ) index [62], as well as the precision of down-stream object detection task to comprehensively validate the deraining performance on real-world scenarios.

B. Ablation Study

1) Validation of RMM and BCM: Since our proposed MLMCN integrates the residual memory module (RMM) and background compensation module (BCM) into a unified framework and constructs the memory compensation block (MCB), we carry out ablation studies to validate the contributions of individual components to the final deraining performance. Based on our final model MLMCN, we design another two comparison models, MLMCN$_{w/o\ BCM}$ and MLMCN$_{w/o\ RMM}$, by simply removing the basic modules RMM and BCM respectively while keeping similar computational complexity and parameter size with MLMCN. For a fair comparison, we maintain the same settings during training.

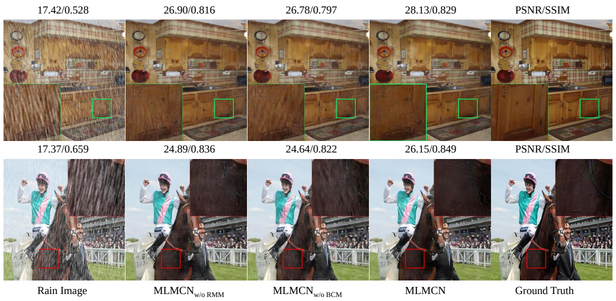


Fig. 7. Evaluation results of two basic modules (RMM and BCM) on **Test1200** dataset. MLMCN$_{w/o\ RMM}$ and MLMCN$_{w/o\ BCM}$ denote the comparison models by removing RMM and BCM respectively from our baseline MLMCN while keeping the similar parameters and computational complexity. In particular, taking the samples in the first row as an example, we can conclude that by combining residual recurrent learning and background compensation strategy, the complete MLMCN model exhibits better restoration performance in terms of less rain streaks and clearer background textures. Please zoom in to see more details.

TABLE I

EVALUATION OF THE BASIC MODULES RMM AND BCM ON **TEST1200** DATASET. MLMCN$_{w/o\ BCM}$ AND MLMCN$_{w/o\ RMM}$ DENOTE THE COMPARISON MODELS BY REMOVING RMM AND BCM RESPECTIVELY FROM MLMCN WHILE KEEPING THE SIMILAR COMPUTATIONAL COMPLEXITY AND PARAMETER SIZE.

Model	Rain Image	MLMCN$_{w/o\ BCM}$	MLMCN$_{w/o\ RMM}$	MLMCN
PSNR	22.15	30.27	30.43	**30.86**
SSIM	0.732	0.898	0.900	**0.901**
FSIM	0.881	0.948	0.949	**0.949**
Par.(Million)	–	6.008	6.005	6.001

Quantitative results are tabulated in Table I. From these scores, we can observe that the complete MLMCN method shows great superiority over its incomplete versions (MLMCN$_{w/o\ BCM}$ and MLMCN$_{w/o\ RMM}$), surpassing them by 0.59dB and 0.43dB on PSNR, respectively. We may attribute these advantages to the effective complementary of these two basic modules, which enable the network to represent rain streaks and compensate background features simultaneously. Visual comparison results are shown in Fig. 7. As expected, MLMCN shows better restoration performance, mainly reflecting on visual results with more credible and clear image details than that of MLMCN$_{w/o\ BCM}$ and MLMCN$_{w/o\ RMM}$. Although MLMCN$_{w/o\ BCM}$ can take advantage of the global information, it still fails to recover credible image textures due to the lack of error compensation, leading to the blurry visual effect. As for MLMCN$_{w/o\ RMM}$, visible rain streaks are observed because of the limited modeling capacity. In summary, the elaborate framework contributes more to the restoration performance, which provides positive clues for subsequent high-level computer vision tasks.

2) Validation of Divide-and-Conquer Strategy: As shown in Fig. 1, the difficulties of removing rain streaks from background image vary with the texture richness. Therefore, we introduce a novel layer-wise learning paradigm (*i.e.*, using the divide-and-conquer strategy), which decomposes the learning task into multiple subproblems and solves them individually. As such, we construct a new multi-level memory

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JSTSP.2021.3052648, IEEE Journal of Selected Topics in Signal Processing

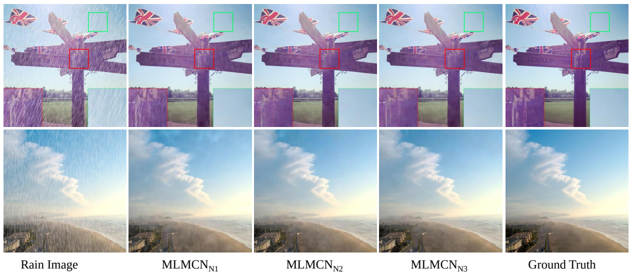


Fig. 8. The investigation results of the divide-and-conquer strategy on **Test1200** dataset. MLMCN$_{N1}$ denotes the deraining model only including subnetwork$_3$ in Fig. 2, similar to the single-level restoration scheme in previous works [13], [14]. MLMCN$_{N2}$ and MLMCN$_{N3}$ refer to the comparison models with two and three subnetworks, respectively. Taking the samples in the first row as an example, we can see that these three deraining models have high consensus levels in removing rain streaks from smooth regions (noted as the green box). But MLMCN$_{N3}$ demonstrates better restoration quality on the detailed regions (noted as the red box) by generating results with clearer and more credible image textures.

compensation network (MLMCN) for rain streaks removal, which contains N subnetworks (subnetwork$_1$, subnetwork$_2$, $\cdots$, subnetwork$_N$) to individually separate rain streaks from the sub-sampled rain images of specific texture richness. In this subsection, we investigate the influence of the divide-and-conquer strategy on the deraining performance. Conceptually, a large N means that the learning task can be divided into more subproblems to refine the representation of rain information. However, we set N to no more than 3 for the consideration of balance between efficiency and performance.

When N is set to 1, we obtain the first comparison model MLMCN$_{N1}$, which only has one subnetwork (noted as the subnetwork$_3$ in Fig. 2), similar to the single-level restoration scheme in previous works [13], [14]. Through a sub-sampled operation on the raw rain image I_{rain} with Gaussian kernel by the scale of 2, we can obtain the input of subnetwork$_2$, which has the same image contents with I_{rain} but different texture richness. By combining subnetwork$_2$ and subnetwork$_3$, we obtain the second comparison model MLMCN$_{N2}$, which performs the restoration task by considering two-level texture richness in background contents. Based on MLMCN$_{N2}$, we construct our baseline MLMCN by adding an additional branch subnetwork$_1$, which takes the sub-sampled image with the factor of 4 as input. Table II presents the comparison results on **Test1200** dataset, including the widely used objective indicators (PSNR, SSIM, etc.), runtime, and model parameter size. It is obvious that our baseline MLMCN using the divide-and-conquer strategy gains the best scores over the other two comparison models with acceptable model parameter size and runtime. The visual comparison results in Fig. 8 reveal that removing the rain streaks from background contents in a stage-wise manner produces better restoration results, enjoying clearer image contents and richer details. These results further demonstrate the effectiveness of the divide-and-conquer strategy in removing rain streaks.

TABLE II
EVALUATION OF THE NUMBER OF SUBNETWORKS (N) ON **TEST1200** DATASET. WE ALSO REPORT COMPARISON RESULTS OF THE AVERAGE INFERENCE TIME AND MODEL PARAMETERS ON TEST SAMPLES WITH THE SIZE OF 512×512.

Model	MLMCN$_{N1}$	MLMCN$_{N2}$	MLMCN
PSNR	30.46	30.68	**30.86**
SSIM	0.9012	0.9019	**0.9042**
FSIM	0.9489	0.9491	**0.9496**
PSIM	0.9989	0.9989	**0.9990**
Runtime (s)	**0.41**	0.50	0.54
Par. (Million)	**2.731**	4.614	6.001

TABLE III
INVESTIGATIONS OF THE DEPTH M IN THREE SUBNETWORKS ON **TEST1200** DATASET. MLMCN$_{a,b,c}$ DENOTES THE COMPARISON MODEL WITH THE BLOCK DEPTHS OF THREE SUBNETWORKS SET TO a, b AND c, RESPECTIVELY.

Index	PSNR	SSIM	FSIM	PSIM
MLMCN$_{3,5,8}$	**30.86**	**0.9042**	**0.9496**	**0.9990**
MLMCN$_{5,5,5}$	30.58	0.9013	0.9491	0.9990
MLMCN$_{3,8,5}$	30.64	0.9021	0.9493	0.9990
MLMCN$_{8,3,5}$	30.34	0.8963	0.9485	0.9989
MLMCN$_{8,5,3}$	30.12	0.8917	0.9436	0.9983

*3) **Validation of Model Depth M:*** The difficulty of removing rain streaks from contaminated background image increases with the texture richness. As mentioned before, our proposed MLMCN consists of three subnetworks to independently accomplish the restoration subtasks by removing rain streaks from background images with the specific texture richness. These subnetworks have the same architecture but different depths M of MCB to cope with different learning tasks. For example, it is enough to set M to 3 in subnetwork$_1$ to separate rain streaks from smooth image contents (low-level texture richness). However, a larger depth is necessary for subnetwork$_3$ for complex texture regions (high-level texture richness). In our final model, the depths M in three subnetworks are set to 3, 5, and 8, respectively, denoted by MLMCN$_{3,5,8}$. To investigate the effect of model depth M on the restoration performance, we construct four comparison models, namely MLMCN$_{a,b,c}$, corresponding to the depth M of [3,8,5], [8,3,5], [8,5,3] and [5,5,5] for these three subnetworks while keeping similar amount of model parameters. The evaluation results are tabulated in Table III. When applying the unified depth for three subnetworks or reducing the depth of subnetwork$_3$, the performance decreases sharply (please refer to the evaluated scores of MLMCN$_{3,5,8}$, MLMCN$_{5,5,5}$ and MLMCN$_{8,5,3}$). The reason might be that it is redundant for subnetwork$_1$ with a larger depth to separate rain streaks from background contents with low-level texture richness. However, there is a limited representative capability for subnetwork$_3$ to recover clear texture details from heavily rain-contaminated image regions with a shallow depth. Therefore, the depths of MCB in three subnetworks are set to 3, 5, and 8 to meet the restoration tasks under different texture richness.

C. Comparison with State-of-the-art Methods

*1) **Synthesized Data:*** To further evaluate the deraining performance of our proposed MLMCN method, we conduct comparison experiments on several synthesized rainy datasets, including **Rain100H** [19], **Rain100L** [19], **Test100** [57],

TABLE IV
COMPARISON RESULTS OF AVERAGE PSNR, SSIM AS WELL AS FSIM ON **RAIN100H** AND **RAIN100L** DATASETS. SINCE THESE TWO DATASETS ARE ADOPTED AS THE TRAINING SAMPLES IN [19], JORDER [19] IS NOT CONSIDERED FOR COMPARISON. MLMCN$_{w\ SSIM}$ DENOTES OUR DERAINING MODEL TRAINED WITH SSIM LOSS.

Methods	DerainNet [13]	RESCAN [15]	DIDMDN [14]	UMRL [25]	SEMI [56]	PreNet [47]	MLMCN (Ours)	MLMCN$_{w\ SSIM}$ (Ours)
Dataset				**Rain100H/Rain100L**				
$PSNR$	14.92/27.03	26.36/29.80	17.35/25.23	26.01/29.18	16.56/25.03	26.77/**32.44**	26.43/29.86	**27.32**/31.25
$SSIM$	0.592/0.884	0.786/0.881	0.524/0.741	0.832/0.923	0.486/0.842	**0.858/0.950**	0.815/0.907	0.840/0.925
$FSIM$	0.755/0.904	0.864/0.919	0.726/0.861	0.876/0.940	0.692/0.893	0.890/**0.956**	0.866/0.923	**0.892**/0.938

TABLE V
THE COMPARISON RESULTS OF AVERAGE PSNR, SSIM AS WELL AS FSIM ON **TEST100**, **TEST2800** AND **TEST1200** DATASETS. MLMCN$_{w\ SSIM}$ DENOTES OUR DERAINING MODEL TRAINED WITH SSIM LOSS.

Methods	DerainNet [13]	JORDER [19]	RESCAN [15]	DIDMDN [14]	UMRL [25]	SEMI [56]	PreNet [47]	MLMCN (Ours)	MLMCN$_{w\ SSIM}$ (Ours)
Dataset					**Test100/Test2800/Test1200**				
$PSNR$	22.77/24.31/23.38	22.40/26.47/26.84	25.00/31.29/30.51	22.56/28.13/29.65	24.41/29.97/30.55	22.35/24.43/26.05	24.81/31.75/31.36	24.98/31.49/30.86	**25.65/32.15/31.38**
$SSIM$	0.810/0.861/0.835	0.798/0.851/0.840	0.835/0.904/0.882	0.818/0.867/0.901	0.829/0.905/0.910	0.788/0.782/0.822	0.851/0.916/0.911	0.850/0.916/0.904	**0.867/0.920/0.911**
$FSIM$	0.884/0.930/0.924	0.883/0.909/0.901	0.909/0.952/0.944	0.899/0.943/0.950	0.910/0.955/0.955	0.887/0.897/0.917	0.916/0.956/0.955	0.916/0.957/0.949	**0.922/0.959/0.955**

Test2800 [30], and **Test1200** [14]. Qualitative results are presented in Tables IV and V. In Table V, it is obvious that our proposed MLMCN with SSIM loss (MLMCN$_{w\ SSIM}$) achieves significant improvements over other state-of-the-art methods on three synthetic datasets. For **Test1200** dataset, MLMCN improves 7.49dB and 4.03dB on PSNR than the early deep-learning based methods DerainNet [13] and JORDER [19], respectively. When compared with RESCAN [15] and DIDMDN [14], MLMCN also outperforms these representative deraining methods, surpassing them by a large margin among all evaluation indexes. Furthermore, when compared with more recent deraining methods UMRL [25], SEMI [56] and PreNet [47], our proposed method is still very competitive in terms of PSNR. Meanwhile, Table IV gives the results under heavy and low-density rain conditions. The results indicate that these algorithms have high consensus levels for restoration tasks under low-density rain conditions. However, only our proposed MLMCN and PreNet still give impressive scores under complex heavy rain scenarios.

Visual comparison results on these synthetic rain datasets in Fig. 9, 10 and 11 show that previous CNN-based methods, such as DerainNet [13] and DIDMDN [14], exhibit poor restoration performance – generating results with visible rain streaks. In particular, these methods generate obvious artifacts in the rain-contaminated regions. The obvious performance decreasing may result from the error estimation of rain information due to the limited modeling capability. As for the semi-supervision method, SEMI fails to remove the rain streaks because there are great uncertainty and difficulty in training the model by adding a relatively hard constraint between both synthetic and real rainy image domains in a single network. Our proposed MLMCN method can produce results with cleaner and clearer texture details over the competing methods.

We further compare our MLMCN model with a representative multi-scale deraining method (DDC [23]) and a pyramid-based deraining algorithm (LPNet [63]) on several synthetic rain datasets. These two methods are also retrained with the unified dataset used in our study. Quantitative results of PSNR and SSIM are provided in Table VI. Our method is still competitive when compared with both the multi-scale based and pyramid-based deraining approaches.

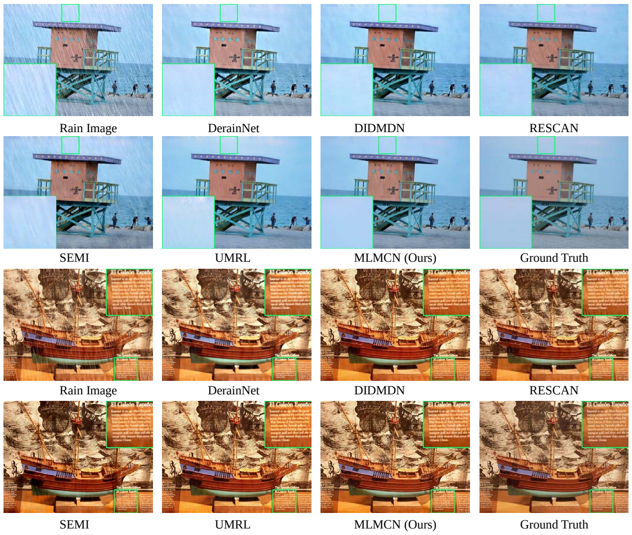


Fig. 9. Restoration results on the Test100 dataset containing samples of different densities and directions.

TABLE VI
THE COMPARISON RESULTS OF AVERAGE PSNR AND SSIM ON **TEST1200** AND **TEST100** DATASETS. MLMCN$_{w\ SSIM}$ DENOTES OUR DERAINING MODEL TRAINED WITH ADDITIONAL SSIM LOSS.

Methods	DDC [23]	LPNet [63]	MLMCN (Ours)	MLMCN$_{w\ SSIM}$ (Ours)
Dataset		**Test100/Test1200**		
$PSNR$	23.47/28.65	23.39/25.00	24.98/30.86	**25.65/31.38**
$SSIM$	0.806/0.854	0.743/0.782	0.850/0.904	**0.867/0.911**

*2) **Real-world Data:*** To better verify the generalization capability of our MLMCN algorithm, we conduct experiments on real-world rain images provided by [14], [19]. The images are diverse in contents, intensities and scales. Since the ground truth is unavailable for these test samples, we introduce NIQE [61] and SSEQ [62] to distinguish from pixel-based evaluation fashion. Smaller scores of SSEQ and NIQE indicate

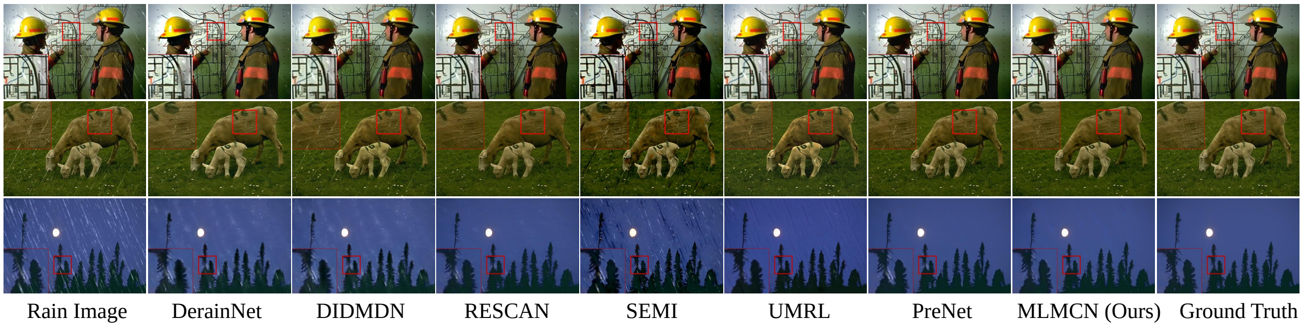


Fig. 10. Restoration results on the Rain100H and Rain100L datasets containing rain streaks with different directions and densities.

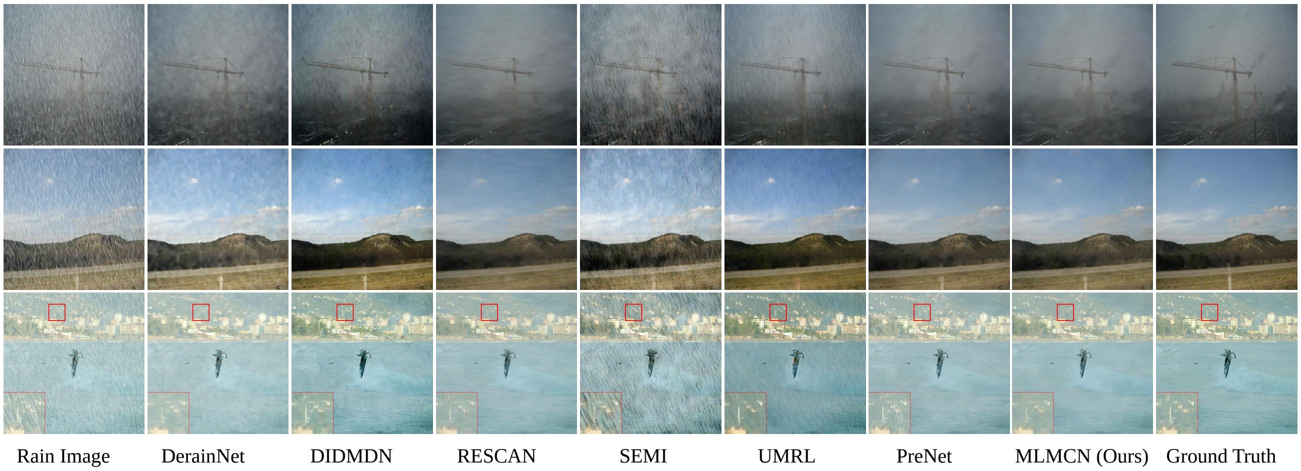


Fig. 11. The restoration examples on the Test1200 dataset, varying in different orientations and scales of rain streaks.

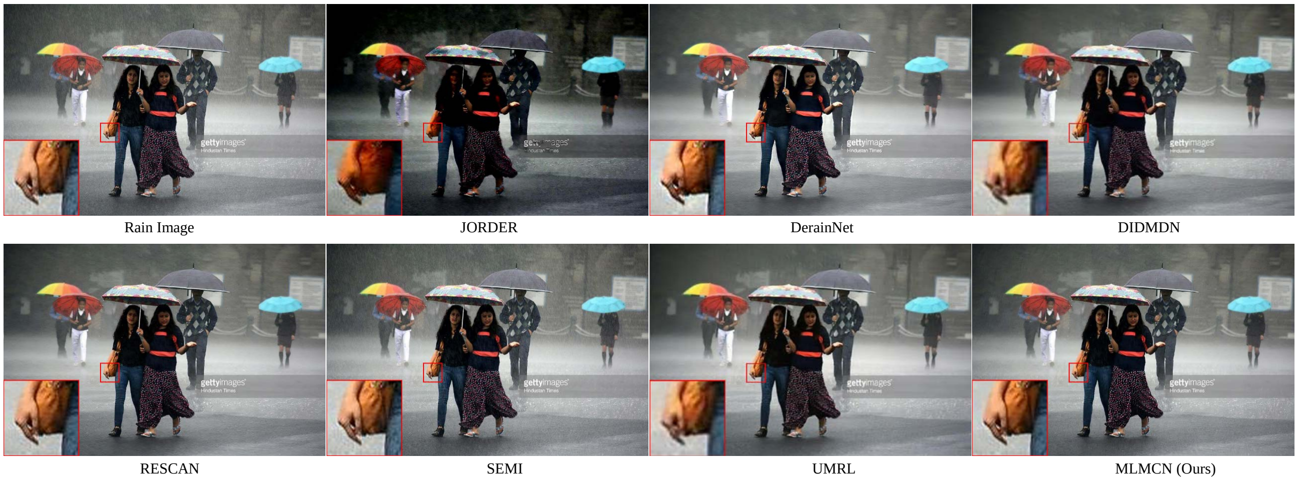


Fig. 12. Restoration results on real-world scenarios, demonstrating the fidelity of image details.

TABLE VII
EVALUATION RESULTS OF AVERAGE NIQE AND SSEQ ON 127 REAL-WORLD SCENARIOS. THESE TWO METRICS CAN EFFECTIVELY MEASURE THE RESTORATION QUALITY WITHOUT REFERENCE. A LOWER VALUE OF NIQE AND SSEQ INDICATES A HIGHER QUALITY IMAGE.

Methods	DerainNet [13]	JORDER [19]	RESCAN [15]	DIDMDN [14]	UMRL [25]	SEMI [56]	PreNet [47]	MLMCN (Ours)
$NIQE$	4.078	4.414	3.852	3.929	3.984	4.262	**3.835**	3.918
$SSEQ$	30.53	34.82	30.09	32.42	29.48	29.35	29.61	**29.19**

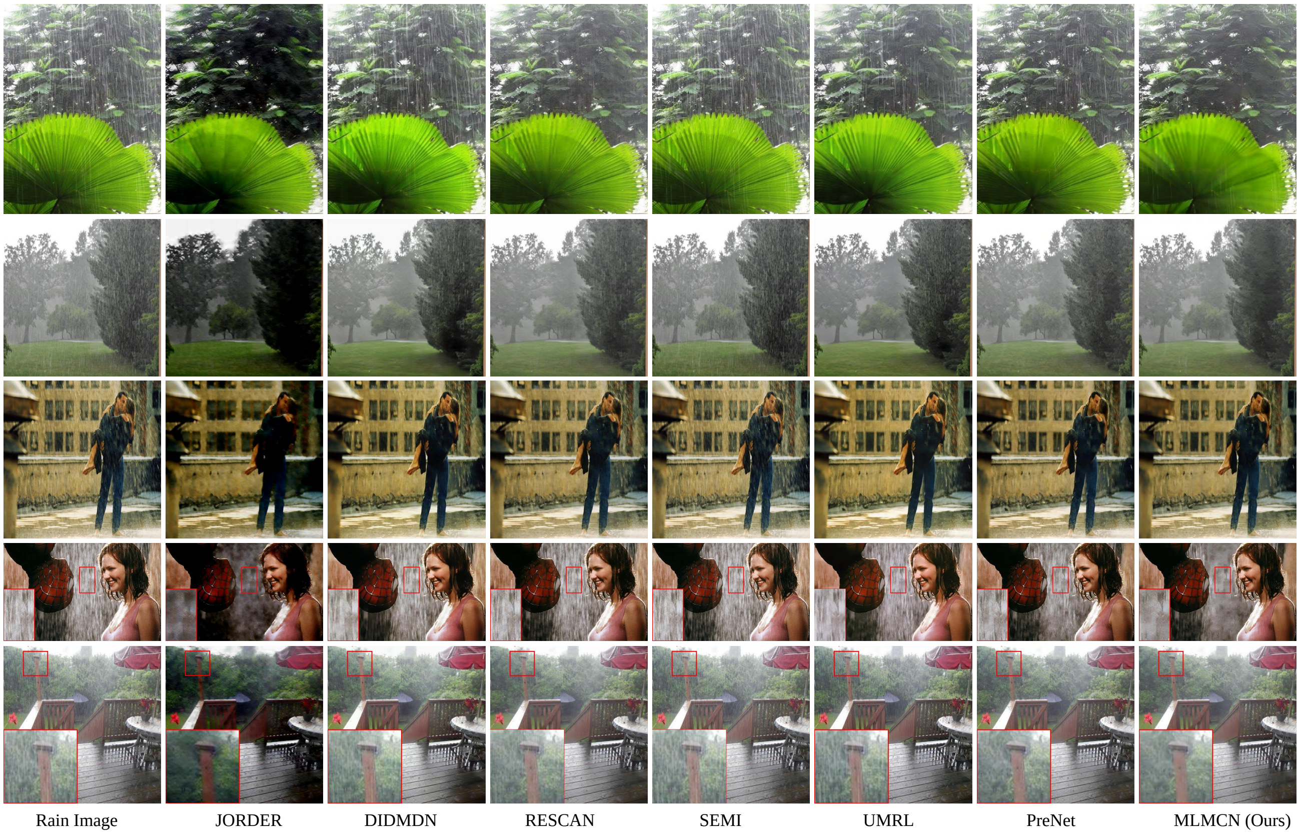


Fig. 13. Deraining results on real-world rainy images under different rain conditions for restoration performance comparison.

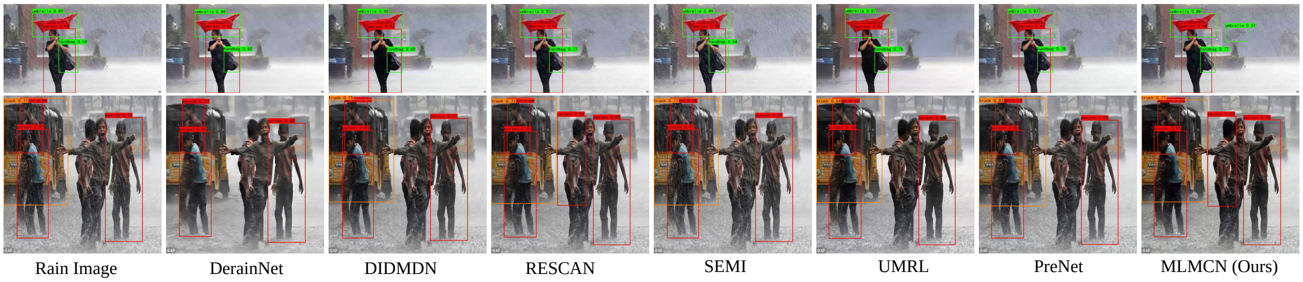


Fig. 14. Examples of joint image deraining and object detection tasks on real-world scenarios.

better perceptual quality and clearer image. The qualitative results are listed in Table VII. Again, our proposed MLMCN has the lowest SSEQ value on 127 real-world rain samples, surpassing the top-performing deraining methods (UMRL [25] and PreNet [47]) by a large margin. For NIQE index, MLMCN obtains the third-best score. Moreover, we conduct comparison experiments on real-world rain images to demonstrate restoration performance in terms of image fidelity. The visual comparison results are shown in Fig. 12. Only our proposed MLMCN recovers the clear and distinguishable hand contour while removing the main rain streaks. Other competing methods generate results with more artifacts and tend to blur the image. The ability to recover the detailed contents further validates the efficacy of the MCB structure. In another group of real-world samples, as shown in Fig. 13, MLMCN arguably exhibits the best visual performance by generating results with more and clearer image details but fewer rain streaks. Overall, the substantial improvements both on visual quality and evaluation indicators on real-world scenarios further verify the effectiveness of our proposed method.

D. Evaluation via Downstream Vision Task

Removing rain streaks from the image under complex rain conditions while recovering credible textural details is meaningful for many high-level vision applications such as object detection and pedestrian recognition. This motivates us to investigate the effect of deraining performance on object detection accuracy based on the popular object detection algorithms (*e.g.*, YOLOv4 [64]). By using our proposed MLMCN and several representative deraining methods, the restoration procedures are directly applied to these rainy images to generate the predicted rain-free outputs. We then apply the publicly available pre-trained models of YOLOv4 for the detection task. Visual comparison results and the corresponding detection precision on two scenarios are shown in Fig. 14. It is evident that rain streaks can greatly degrade the detection accuracy by missing targets and producing a low detection precision. When compared with other deraining models, the deraining results generated by MLMCN facilitate better object detection performance. We attribute the considerable performance improvements of both deraining and down-stream object detection to the effective global spatial feature representation and background compensation strategy, which leads to more faithful recovery of textural details.

V. Conclusions

In this paper, we propose a novel multi-level memory compensation network (MLMCN) for rain streak removal using the divide-and-conquer strategy. Specifically, it decomposes the learning task into multiple subproblems according to the diverse levels of texture richness in background contents (the layer of rain image pyramid). MLMCN contains several parallel subnetworks, which are designed to independently learn the sub-distribution of rain information in sub-samples with specific texture richness. We integrate the recurrent calculation and residual learning to fully exploit the global textual information and design a novel memory compensation block (MCB) for rain streak representation. Meanwhile, to improve the fidelity of background details, we introduce feedback representation in MCB by learning the compensated features of background details to refine the estimation of rain streaks. Experimental results on several rain datasets and the joint deraining and object detection task validate the superiority of our method over the existing state-of-the-art approaches.

References

[1] S. Wu, S. Wang, R. Laganiere, C. Liu, H.-S. Wong, and Y. Xu, "Exploiting target data to learn deep convolutional networks for scene-adapted human detection," *IEEE Trans. Image Process.*, vol. 27, no. 3, pp. 1418–1432, 2017.

[2] G. Chen, J. Lu, M. Yang, and J. Zhou, "Spatial-temporal attention-aware learning for video-based person re-identification," *IEEE Trans. Image Process.*, 2019.

[3] T.-H. Chan, K. Jia, S. Gao, J. Lu, Z. Zeng, and Y. Ma, "Pcanet: A simple deep learning baseline for image classification?" *IEEE Trans. Image Process.*, vol. 24, no. 12, pp. 5017–5032, 2015.

[4] M. Teichmann, M. Weber, M. Zoellner, R. Cipolla, and R. Urtasun, "Multinet: Real-time joint semantic reasoning for autonomous driving," in *IV*, 2018, pp. 1013–1020.

[5] X. Zhang, H. Li, Y. Qi, W. K. Leow, and T. K. Ng, "Rain removal in video by combining temporal and chromatic properties," in *ICME*, 2006, pp. 461–464.

[6] Y. Li, R. T. Tan, X. Guo, J. Lu, and M. S. Brown, "Rain streak removal using layer priors," in *CVPR*, 2016, pp. 2736–2744.

[7] H. Wang, Y. Wu, Q. Xie, Q. Zhao, Y. Liang, and D. Meng, "Structural residual learning for single image rain removal," *KNOWLEDGE-BASED SYSTEMS*, 2020.

[8] L.-W. Kang, C.-W. Lin, and Y.-H. Fu, "Automatic single-image-based rain streaks removal via image decomposition," *IEEE Trans. Image Process.*, vol. 21, no. 4, pp. 1742–1755, 2011.

[9] Y. Luo, Y. Xu, and H. Ji, "Removing rain from a single image via discriminative sparse coding," in *ICCV*, 2015, pp. 3397–3405.

[10] K. Garg and S. K. Nayar, "When does a camera see rain?" in *ICCV*, vol. 2, 2005, pp. 1067–1074.

[11] W. Ren, J. Tian, Z. Han, A. Chan, and Y. Tang, "Video desnowing and deraining based on matrix decomposition," in *CVPR*, 2017, pp. 4210–4219.

[12] Y.-L. Chen and C.-T. Hsu, "A generalized low-rank appearance model for spatio-temporally correlated rain streaks," in *ICCV*, 2013, pp. 1968–1975.

[13] X. Fu, J. Huang, X. Ding, Y. Liao, and J. Paisley, "Clearing the skies: A deep network architecture for single-image rain removal," *IEEE Trans. Image Process.*, vol. 26, no. 6, pp. 2944–2956, 2017.

[14] H. Zhang and V. M. Patel, "Density-aware single image de-raining using a multi-stream dense network," in *CVPR*, 2018, pp. 695–704.

[15] X. Li, J. Wu, Z. Lin, H. Liu, and H. Zha, "Recurrent squeeze-and-excitation context aggregation net for single image deraining," in *ECCV*, 2018, pp. 254–269.

[16] P. Yi, Z. Wang, K. Jiang, Z. Shao, and J. Ma, "Multi-temporal ultra dense memory network for video super-resolution," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 30, no. 8, pp. 2503–2516, Aug 2020.

[17] P. Yi, Z. Wang, K. Jiang, J. Jiang, T. Lu, and J. Ma, "A progressive fusion generative adversarial network for realistic and consistent video super-resolution," *IEEE Trans. Pattern Anal. Mach. Intell.*, pp. 1–1, 2020.

[18] J. Liu, W. Yang, S. Yang, and Z. Guo, "D3r-net: Dynamic routing residue recurrent network for video rain removal," *IEEE Trans. Image Process.*, vol. 28, no. 2, pp. 699–712, 2018.

[19] W. Yang, R. T. Tan, J. Feng, J. Liu, Z. Guo, and S. Yan, "Deep joint rain detection and removal from a single image," in *CVPR*, 2017, pp. 1357–1366.

[20] M. Li, X. Cao, Q. Zhao, L. Zhang, C. Gao, and D. Meng, "Video rain/snow removal by transformed online multiscale convolutional sparse coding," *IEEE Trans. Pattern Anal. Mach. Intell.*, 2020.

[21] W. Yang, R. T. Tan, J. Feng, J. Liu, S. Yan, and Z. Guo, "Joint rain detection and removal from a single image with contextualized deep networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 6, pp. 1377–1393, 2020.

[22] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *CVPR*, 2016, pp. 770–778.

[23] S. Li, W. Ren, J. Zhang, J. Yu, and X. Guo, "Single image rain removal via a deep decomposition–composition network," *Computer Vision and Image Understanding*, vol. 186, pp. 48–57, 2019.

[24] K. Jiang, Z. Wang, P. Yi, C. Chen, B. Huang, Y. Luo, J. Ma, and J. Jiang, "Multi-scale progressive fusion network for single image deraining," in *CVPR*, 2020, pp. 8346–8355.

[25] R. Yasarla and V. M. Patel, "Uncertainty guided multi-scale residual learning-using a cycle spinning cnn for single image de-raining," in *CVPR*, 2019, pp. 8405–8414.

[26] R. A. Jacobs, M. I. Jordan, S. J. Nowlan, G. E. Hinton *et al.*, "Adaptive mixtures of local experts." *Neural computation*, vol. 3, no. 1, pp. 79–87, 1991.

[27] M. I. Jordan and R. A. Jacobs, "Hierarchical mixtures of experts and the em algorithm," *Neural computation*, vol. 6, no. 2, pp. 181–214, 1994.

[28] Cheng-Chin Chiang and Hsin-Chia Fu, "A divide-and-conquer methodology for modular supervised neural network design," in *IEEE International Conference on Neural Networks*, vol. 1, 1994, pp. 119–124 vol.1.

[29] H. Wang, Y. Wu, M. Li, Q. Zhao, and D. Meng, "A survey on rain removal from video and single image," *arXiv preprint arXiv:1909.08326*, 2019.

[30] X. Fu, J. Huang, D. Zeng, Y. Huang, X. Ding, and J. Paisley, "Removing rain from single images via a deep detail network," in *CVPR*, 2017, pp. 3855–3863.

[31] K. Jiang, Z. Wang, P. Yi, C. Chen, Z. Han, T. Lu, B. Huang, and J. Jiang, "Decomposition makes better rain removal: An improved attention-guided deraining network," *IEEE Transactions on Circuits and Systems for Video Technology*, pp. 1–1, 2020.

[32] R. Qian, R. T. Tan, W. Yang, J. Su, and J. Liu, "Attentive generative adversarial network for raindrop removal from a single image," in *CVPR*, 2018, pp. 2482–2491.

[33] W. Yang, J. Liu, and J. Feng, "Frame-consistent recurrent video deraining with dual-level flow," in *CVPR*, 2019, pp. 1661–1670.

[34] H. Wang, Q. Xie, Q. Zhao, and D. Meng, "A model-driven deep neural network for single image rain removal," in *CVPR*, 2020, pp. 3103–3112.

[35] P. C. Barnum, S. Narasimhan, and T. Kanade, "Analysis of rain and snow in frequency space," *IJCV*, vol. 86, no. 2-3, p. 256, 2010.

[36] J. Bossu, N. Hautière, and J.-P. Tarel, "Rain or snow detection in image sequences through use of a histogram of orientation of streaks," *IJCV*, vol. 93, no. 3, pp. 348–367, 2011.

[37] J. Chen, C.-H. Tan, J. Hou, L.-P. Chau, and H. Li, "Robust video content alignment and compensation for rain removal in a cnn framework," in *CVPR*, 2018, pp. 6286–6295.

[38] H. Zhang and V. M. Patel, "Convolutional sparse and low-rank coding-based rain streak removal," in *WACV*, 2017, pp. 1259–1267.

[39] X. Hu, C.-W. Fu, L. Zhu, and P.-A. Heng, "Depth-attentional features for single-image rain removal," in *CVPR*, 2019, pp. 8022–8031.

[40] T. Wang, X. Yang, K. Xu, S. Chen, Q. Zhang, and R. W. Lau, "Spatial attentive single-image deraining with a high quality real rain dataset," in *CVPR*, 2019, pp. 12 270–12 279.

[41] S. Li, I. B. Araujo, W. Ren, Z. Wang, E. K. Tokuda, R. H. Junior, R. Cesar-Junior, J. Zhang, X. Guo, and X. Cao, "Single image deraining: A comprehensive benchmark analysis," in *CVPR*, 2019, pp. 3838–3847.

[42] S. Xingjian, Z. Chen, H. Wang, D.-Y. Yeung, W.-K. Wong, and W.-c. Woo, "Convolutional lstm network: A machine learning approach for precipitation nowcasting," in *Advances in neural information processing systems*, 2015, pp. 802–810.

[43] W.-N. Hsu, Y. Zhang, and J. Glass, "A prioritized grid long short-term memory rnn for speech recognition," in *2016 IEEE Spoken Language Technology Workshop (SLT)*, 2016, pp. 467–473.

[44] C. Liu, Y. Wang, K. Kumar, and Y. Gong, "Investigations on speaker adaptation of lstm rnn models for speech recognition," in *ICASSP*, 2016, pp. 5020–5024.

[45] M. A. Zaman and S. Z. Mishu, "Convolutional recurrent neural network for question answering," in *EICT*, 2017, pp. 1–6.

[46] Z. Wang, P. Yi, K. Jiang, J. Jiang, Z. Han, T. Lu, and J. Ma, "Multi-memory convolutional neural network for video super-resolution," *IEEE Trans. Image Process.*, vol. 28, no. 5, pp. 2530–2544, 2018.

[47] D. Ren, W. Zuo, Q. Hu, P. Zhu, and D. Meng, "Progressive image deraining networks: a better and simpler baseline," in *CVPR*, 2019, pp. 3937–3946.

[48] Y. Yang and H. Lu, "Single image deraining using a recurrent multi-scale aggregation and enhancement network," in *ICME*, 2019, pp. 1378–1383.

[49] Y. Zhang, Y. Tian, Y. Kong, B. Zhong, and Y. Fu, "Residual dense network for image super-resolution," in *CVPR*, 2018, pp. 2472–2481.

[50] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *CVPR*, 2017, pp. 4700–4708.

[51] W. Shi, J. Caballero, F. Huszr, J. Totz, A. P. Aitken, R. Bishop, D. Rueckert, and Z. Wang, "Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network," in *CVPR*, 2016, pp. 1874–1883.

[52] W.-S. Lai, J.-B. Huang, N. Ahuja, and M.-H. Yang, "Fast and accurate image super-resolution with deep laplacian pyramid networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 11, pp. 2599–2613, Nov 2019.

[53] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *ICLR*, 2015.

[54] A. Graves, A.-r. Mohamed, and G. Hinton, "Speech recognition with deep recurrent neural networks," in *ICASSP*, 2013, pp. 6645–6649.

[55] A. Kumar, O. Irsoy, P. Ondruska, M. Iyyer, J. Bradbury, I. Gulrajani, V. Zhong, R. Paulus, and R. Socher, "Ask me anything: Dynamic memory networks for natural language processing," in *ICML*, 2016, pp. 1378–1387.

[56] W. Wei, D. Meng, Q. Zhao, Z. Xu, and Y. Wu, "Semi-supervised transfer learning for image rain removal," in *CVPR*, 2019, pp. 3877–3886.

[57] H. Zhang, V. Sindagi, and V. M. Patel, "Image de-raining using a conditional generative adversarial network," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 30, no. 11, pp. 3943–3956, 2020.

[58] L. Zhang, L. Zhang, X. Mou, and D. Zhang, "Fsim: A feature similarity index for image quality assessment," *IEEE Trans. Image Process.*, vol. 20, no. 8, pp. 2378–2386, 2011.

[59] Z. Wang, A. C. Bovik, H. R. Sheikh, E. P. Simoncelli *et al.*, "Image quality assessment: from error visibility to structural similarity," *IEEE Trans. Image Process.*, vol. 13, no. 4, pp. 600–612, 2004.

[60] K. Gu, L. Li, H. Lu, X. Min, and W. Lin, "A fast reliable image quality predictor by fusing micro-and macro-structures," *IEEE Trans. Ind. Electron.*, vol. 64, no. 5, pp. 3903–3912, 2017.

[61] A. Mittal, R. Soundararajan, and A. C. Bovik, "Making a "completely blind " image quality analyzer," *IEEE Signal Process. Lett*, vol. 20, no. 3, pp. 209–212, 2013.

[62] L. Liu, B. Liu, H. Huang, and A. C. Bovik, "No-reference image quality assessment based on spatial and spectral entropies," *Signal Processing: Image Communication*, vol. 29, no. 8, pp. 856–863, 2014.

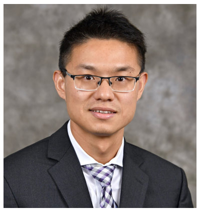
[63] X. Fu, B. Liang, Y. Huang, X. Ding, and J. Paisley, "Lightweight pyramid networks for image deraining," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 31, no. 6, pp. 1794–1807, 2020.

[64] A. Bochkovskiy, C. Wang, and H. M. Liao, "Yolov4: Optimal speed and accuracy of object detection," *CoRR*, vol. abs/2004.10934, 2020.

Kui Jiang received the M.S. degree in the School of Computer Science, Wuhan University, Wuhan, China, in 2019. He is currently working toward the Ph.D. degree under the supervision of Prof. Zhongyuan Wang in the School of Computer Science, Wuhan University, Wuhan, China. His research interests include image/video processing and computer vision.

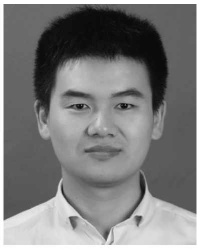
Zhongyuan Wang (M'13) received the Ph.D. degree in communication and information system from Wuhan University, Wuhan, China, in 2008. Dr. Wang is now an associate professor with School of Computer Science, Wuhan University, Wuhan, China. He is currently directing three projects funded by the National Natural Science Foundation Program of China. His research interests include video compression, image processing, and multimedia communications, etc.

Peng Yi received the B.S. degree in Faculty of Electronic Information and Electrical Engineering from Dalian University of Technology, Dalian, China, in 2017. He is currently working toward the Ph.D. degree under the supervision of Prof. Zhongyuan Wang in the School of Computer, Wuhan University. His research interests include image/video processing and computer vision.



Chen Chen received the B.E. degree in automation from Beijing Forestry University, Beijing, China, in 2009, the M.S. degree in electrical engineering from Mississippi State University, Starkville, MS, USA, in 2012, and the Ph.D. degree from the Department of Electrical Engineering, University of Texas at Dallas, Richardson, TX, USA, in 2016. He held a Postdoctoral Research Associate position at the Center for Research in Computer Vision (CRCV), University of Central Florida, FL, USA, from July 2016 to June 2018. He is an Assistant Professor with the Department of Electrical and Computer Engineering, University of North Carolina at Charlotte, Charlotte, NC, USA. His research interests include signal and image processing, computer vision, and deep learning. Dr. Chen is an Associate Editor of Journal of Real-Time Image Processing and the IEEE JOURNAL ON MINIATURIZATION FOR AIR AND SPACE SYSTEMS.

Xiaofen Wang received the B.S degree from Yangtze University in 2008 and M.S degree in Guilin University of Electronic Technology in 2011. From 2012, she worked at Hubei University of Science and Technology. She is currently pursuing her Ph.D degree in School of Computer Science, Wuhan University. Her research interests focus on computer vision, deep learning and machine learning.



Junjun Jiang (M'15) received the B.S. degree from the Department of Mathematics, Huaqiao University, Quanzhou, China, in 2009, and the Ph.D. degree from the School of Computer, Wuhan University, Wuhan, China, in 2014. From 2015 to 2018, he was an Associate Professor with China University of Geosciences, Wuhan. He was a Project Researcher with the National Institute of Informatics, Tokyo, Japan, from 2016 to 2018. He is currently a Professor with the School of Computer Science and Technology, Harbin Institute of Technology, Harbin, China. He won the Finalist of the World's FIRST 10K Best Paper Award at ICME 2017, the Best Student Paper Runner-up Award at MMM 2015, the Best Paper Award at IFTC 2018. He received the 2016 China Computer Federation (CCF) Outstanding Doctoral Dissertation Award and 2015 ACM Wuhan Doctoral Dissertation Award. His research interests include image processing and computer vision.

Zixiang Xiong (F'07) received the Ph.D. degree in electrical engineering from the University of Illinois at UrbanaCChampaign, Urbana-Champaign, IL, USA, in 1996. Since 1999, he has been with the Department of Electrical and Computer Engineering, Texas A&M University, College Station, TX, USA, where he is currently a Professor. Dr. Xiong served as an Associate Editor for the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY from 1999 to 2005, the IEEE TRANSACTIONS ON IMAGE PROCESSING from 2002 to 2005, the IEEE TRANSACTIONS ON SIGNAL PROCESSING from 2002 to 2006, the IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS (PART B) from 2005 to 2009, and the IEEE TRANSACTIONS ON COMMUNICATIONS from 2008 to 2013. He is currently an Associate Editor for the IEEE TRANSACTIONS MULTIMEDIA.