Exploiting Global and Local Attentions for Heavy Rain Removal on Single Images

DAC TUNG VU, JUAN LUIS GONZALEZ, MUNCHURL KIM
Korea Advanced Institute of Science and Technology, Daejeon, Korea
Corresponding author: Munchurl Kim (e-mail: mkimee@kaist.ac.kr).

This work was supported by a grant from the Institute for Information & Communications Technology Promotion (IITP) funded by the Korean government (MSIT) (No. 2017-0-00419, Intelligent High Realistic Visual Processing for Smart Broadcasting Media). This was also partially supported by the BK21 (Brain Korea 21) Program.

ABSTRACT Heavy rain removal from a single image is the task of simultaneously eliminating rain streaks and fog, which can dramatically degrade the quality of captured images. Most existing rain removal methods do not generalize well for the heavy rain case. In this work, we propose a novel network architecture consisting of three sub-networks to remove heavy rain from a single image without estimating rain streaks and fog separately. The first sub-net, a U-net-based architecture that incorporates our Spatial Channel Attention (SCA) blocks, extracts global features that provide sufficient contextual information needed to remove atmospheric distortions caused by rain and fog. The second sub-net learns the additive residues information, which is useful in removing rain streak artifacts via our proposed Residual Inception Modules (RIM). The third sub-net, the multiplicative sub-net, adopts our Channel-attentive Inception Modules (CIM) and learns the essential brighter local features which are not effectively extracted in the SCA and additive sub-nets by modulating the local pixel intensities in the derained images. Our three clean image results are then combined via an attentive blending block to generate the final clean image. Our method with SCA, RIM, and CIM significantly outperforms the previous state-of-the-art single-image deraining methods on the synthetic datasets, shows considerably cleaner and sharper derained estimates on the real image datasets. We present extensive experiments and ablation studies supporting each of our method’s contributions on both synthetic and real image datasets.

INDEX TERMS Heavy rain removal, Single image deraining, Image restoration, Deep learning.

I. INTRODUCTION

Rain is a typical weather condition that affects the quality of images and videos, potentially degrading the performance of downstream tasks such as object tracking [2], object detection [12], and autonomous driving [14]. There are several works for rain removal on single images. However, most of them can only remove light rain efficiently and show very poor results for the heavy rain cases. Heavy rain is a much more challenging task as it entangles heavy rain streaks and haze/fog. A few works have addressed such a heavy rain removal problem, but still with lots of room for improvement.

Early rain removal methods applied hand-crafted priors [3], [10], [25], [28], [46] such as sparse learning. Even when these methods improved the overall scene, they often resulted in artifacts in the derained images, such as blur and color distortions. Recently, with the development of the deep convolutional neural networks (DCNN), learning-based methods [24], [29], [34], [40], [42], [43] have shown dramatic improvements in deraining tasks with the help of large labeled datasets. However, these methods only achieve good results in the light rain removal task. For the heavy rain removal task, most approaches fail to restore the clean background scene.

Recently, Li et al. [23] proposed a heavy rain model as:

\[ J = T \odot (I + S) + (1 - T) \odot A \]  \hspace{1cm} (1)

where \( J \) is the observed (heavy rain) image, \( I \) is the clean image, \( S \) is the rain streak layers, \( T \) is the transmission maps, and \( A \) is the atmospheric light of the scene. \( \odot \) denotes the element-wise multiplication. This model considers rain streaks \( S \) and fog \((T \text{ and } A)\) but predicts \( S \), \( A \), and \( T \) separately. In heavy rain, rain streaks and fog are mutually dependent by nature, making heavy rain removal a highly ill-posed problem. To tackle this issue, Li et al. [23] applied a generative adversarial network (GAN) [9] to refine the background clean images. However, the brightness and color contrasts were not completely corrected.

In another work, Hu et al. [11] considers the heavy rain scenes as a composition of a clean image \( I \), a rain layer \( R \),...
and a fog layer $F$ by formulating the heavy rain images $J$ as:

$$J = I \odot (1 - R - F) + R + AF$$

(2)

Also, we re-arrange (2) for the clean image $I$ as:

$$I = J + \frac{(-R - AF)}{1 - R - F} = J \odot \frac{1}{1 - R - F} - \frac{R + AF}{1 - R - F}$$

(3)

where $0 \leq R \leq 1$ represents the rain layer, $A$ denotes an atmospheric light, and $0 \leq F \leq 1$ denotes the fog layer. In [11], to obtain the training pairs of clean and rainy images, rain layers $R$ and fog layers $F$ are computed by taking into account the scene depths. Hu et al. [11] proposed a residual network to learn a low-resolution depth map and depth-guided additive residues for rain removal. But the clean image $I$ in (3) also have to be reconstructed based on the multiplicative residues term with $J$, the first term in (3), which could not be obtained from Hu et al.’s residual network. In contrast with the previous methods [11], [23], that indirectly generate the clean image by estimating the components in [11] and [2], we directly generate the clean output image in our proposed method. In this paper, we propose to model the clean image $I$ as the combination of a roughly derived and defogged estimate $I_0$, an additive residue estimate $I_1$ (given by by $(J + R_A)$, where $R_A$ is the output of an additive sub-net), and a multiplicative residue estimate $I_2$ (which is provided by $(J \odot R_M)$, where $R_M$ is the output of a multiplicative sub-net), as defined by:

$$I = I_0 \odot W_0 + (J + R_A) \odot W_1 + (J \odot R_M) \odot W_2$$

$$= I_0 \odot W_0 + I_1 \odot W_1 + I_2 \odot W_2,$$

(4)

where $W_0$, $W_1$, and $W_2$ blend pixel-wise the image components of $I$.

In this paper, we propose a novel end-to-end network architecture for heavy-rain removal. Our network, called GLA-HRRNet, is integrated by three sub-nets that can effectively remove heavy rain: (i) In the first sub-net, we obtain global features from a U-net architecture where our proposed spatial channel attention (SCA) blocks are that incorporated, called the SCA sub-net; (ii) The second sub-net analyzes a heavy rain input into a clean image and additive residues with our proposed residual inception modules (RIM); (iii) The third sub-net employs the channel-attentive inception modules (CIM) to selectively learn the informative local features by separating the heavy rain scene into a clean image and multiplicative residues. Due to heavy rains’ complex characteristics, some of the background information in the three sub-net outputs is estimated with degraded quality. To remedy this, we employed an attentive blending block to aggregate the three sub-nets’ informative features into the final clean output image. Our main contributions in this paper are the following:

- Firstly, we introduce an end-to-end network that simultaneously exploits global and local attentions to effectively remove heavy rain from single images, called GLA-HRRNet, which is a very challenging and ill-posed task in image restoration.
- Secondly, we reinterpret the heavy rain model in terms of additive and multiplicative terms in our GLA-HRRNet, which effectively solves the heavy rain removal task without directly estimating the components in heavy rain scenes such as the atmospheric light, rain streaks, and transmission map. Our approach tackles the problem that there is no compatible formula to model heavy rain due to the entanglement of rain streaks and fog in natural scenes.
- Thirdly, we propose a spatial channel attention block that is effectively integrated into our Spatial Channel Attention (SCA) sub-net.
- Finally, we compare the proposed model with state-of-the-art (SOTA) deraining methods. Our method significantly outperforms the SOTA methods in both synthetic and real image datasets by a considerable margin.

II. RELATED WORKS

In this section, we review related research in the field of image deraining. According to the type of input data, the existing works can be classified into two categories: video deraining methods and single image deraining methods.

A. VIDEO DERAINING

The traditional video deraining methods process the relative information in sequential frames to detect and remove rain streaks. In [6], the authors proposed a correlation model that captures the dynamics of rain and a physics-based motion blur. In [18], Kim et al. obtained an initial rain map by subtracting warped frames from a current frame. They then remove the detected rain streaks by applying a low-rank matrix. On the other hand, the authors in [30] divide the rainy input images into sparse ones and dense ones and model them separately based on a matrix decomposition.

Moreover, with the dramatic development of DCNN [38], learning-based methods [20], [22], [26], [39] have signifi-

---

**FIGURE 1.** Results on heavy rain removal for different SOTA methods. Our method generates a clean, derained, free-from-artifacts and realistic image.
FIGURE 2. The overall architecture of our Global and Local Attentive Heavy Rain Removal Network (GLA-HRRNet). The design of the Spatial channel attention (SCA) block is illustrated in Fig. 3. The detailed architecture of the Residual inception modules (RIM) and Channel-attentive inception modules (CIM) is shown in Fig. 5 and Fig. 6 respectively.

cantly outperformed the traditional approaches. The work in [26] firstly considers the rain removal problem, including rain occlusions. They remove rain from videos by constructing a recurrent DCNN to recover the spatial texture in rain occlusions guided by background reconstruction temporal coherence. Due to the repetitive local patterns and configuration characteristics of rain streaks, Li et al. [22] proposed to apply a multi-scale convolutional model to remove rain streaks. Besides, Yang et al. [39] showed that the adjacent frames are highly correlated, and rain streaks are randomly distributed along the temporal dimension. Based on these two observations, they proposed a “self-learned” method consisting of two stages to remove rain streaks. This method was based on temporal correlation and consistency of frames in the video.

B. SINGLE IMAGE DERAINING

Early rain removal methods for single images utilized hand-crafted priors such as Gaussian mixture models (GMMs) [25], learning dictionaries and sparse architectures [3], [10], [28]. Moreover, Zhu et al. [46] proposed to compute the gradients in a local window to extract rainy regions, as rain streaks typically span in a narrow range of directions. The authors in [1] remove rain with the assumption that rain streaks have a similar direction for time interval. Although improving the overall scene, hand-crafted priors tended to generate artifacts such as blurring or distortion.

In the deep learning era, learning-based methods [4], [5], [15], [34], [35], [37], [41], [43] have shown dramatic improvements from labeled datasets. Fu et al. [4] applied a low pass filter to decompose the rain image into two components, the base, and the detail parts, then utilized a ResNet for training their network to predict the detail component. Yang et al. [37] employed multiple dilated convolutions to remove rain. Moreover, Zang et al. [43] exploited a conditional GAN architecture, and perceptual loss [17] to train their model. Other authors have proposed using recurrent neural network (RNN) architectures to remove rain in multiple stages [24], [34]. These methods only achieve good results for light rain scenarios but fail to recover the background scene in heavy rain.

Relatively, less effort has been made on heavy rain removal. Li et al. [23] proposed a model to predict rain streaks, atmospheric light, and transmission maps, which are used to compute the initial clean image estimate. Additionally, Li et al. refined the initial clean image estimate using a generative adversarial network (GAN). Wang et al. [36] applied a model considering a transmission map of rain streaks and fog. These methods do not efficiently remove heavy rain as their models struggle to represent the complex relations between heavy rain and fog in the single images. Different from the existing methods [11], [23], [36], we propose an end-to-end network to solve this problem.

In this work, we do not aim to model relative streaks, atmospheric lights, and transmission maps independently. Instead, we separate the heavy rain scene into a clean component, additive, and multiplicative residue components, which can better model the heavy rain imagery. More details are described in the next section.

III. METHODOLOGY

This section presents our proposed novel network architecture and loss functions for effective heavy rain removal in our method. Fig. 2 shows our proposed overall architecture
with three sub-nets and the attentive blending block used to combine informative features from these sub-nets to generate a final output image. We discuss the loss functions applied to the intermediate outputs and final results in Section III-B.

A. NETWORK ARCHITECTURE

Our network consists of three sub-nets: (i) a spatial channel attention (SCA) sub-net, (ii) an additive sub-net and (iii) a multiplicative sub-net, as shown in Fig. 2. The SCA sub-net takes heavy rain input images and directly yields rain-free image output. The SCA sub-net applies spatial channel attention (SCA) blocks based on a U-net architecture. The other two sub-nets predict residues from heavy rain scenes based on two operations: addition and multiplication. These two sub-nets take extracted features from the encoder part of the SCA sub-net as input. The second sub-net (additive sub-net) applies residual inception modules (RIM) to learn additive residues and adds additive residues to the input images to generate clean images. The third sub-net (multiplicative sub-net) employs channel-attentive inception modules (CIM) to produce multiplicative residues. After that, the multiplicative residues are multiplied by input heavy rainy images and generates clean images. However, some background information in the three clean image results is degraded because rain and fog are entangled and could not be easily separated. To solve this, we employ an attentive blending block to combine useful features from the three clean image estimates, $I_0$, $I_1$, $I_2$, and generate a final clean output image. We discuss more details about the three sub-nets in the following sections.

1) Spatial Channel Attention (SCA) Sub-Net

An auto-encoder based U-net [31] is a specific architecture of a feedforward neural network with skip connections from its encoder part to the decoder part, which has been popularly used for image restorations. Our SCA sub-net is based on a U-net architecture that directly estimates a clean image from a heavy rain scene via learning a direct mapping. The top left of Fig. 2 depicts the SCA sub-net. The SCA sub-net has three convolution layers at the front to extract useful features from three-channel images. After that, the SCA blocks are followed as key components in the SCA sub-net. The combination of the SCA blocks in multi-scale resolutions helps the network learn global features to remove heavy rains effectively. Finally, one convolution layer constructs a three-channel clean image output. The SCA sub-net generates a coarse derained image ($I_0$) and extracts informative features ($F_{sca}$) that are later fed into the additive and multiplicative sub-nets, which are given by:

$$I_0, F_{sca} = SCA_{subnet}(J|\phi_{sca}) \quad (5)$$

where $\phi_{sca}$ are the learnable filter parameters of our SCA sub-net.

The SCA block is illustrated in Fig. 3. In this block, we first employ a convolution layer with stride 2 to downsample the input image in the encoder part (in the decoder part, this convolution layer is replaced with a transposed convolution layer to upsample the down-sized features). After that, we apply spatial attention (SA) (a light cyan block as shown in Fig. 3) and followed by the channel attention (CA) (a flesh block in Fig. 3). SA focuses on meaningful regions which contain rain streaks and fog in the heavy rain removal task. On the other hand, CA focuses on useful channels which are sensitive to rain streaks and fog. So, we combine both SA and CA to focus on informative regions and across channels to let the network efficiently remove heavy rain in single images with diverse shapes and directions of rain streaks and fog entanglement. The combination of CA and SA allows the SCA sub-net to be a main contributor to the reconstruction of the final output, even outperforming the SOTA methods as shown later in Section V.

Moreover, we extract informative features from lower resolutions in the encoder part of the SCA sub-net and feed them into the additive and multiplicative sub-nets. It was empirically found that passing the 256-channel features from the SCA sub-net, which are at 1/4 the input resolution, yields the best performance at a lower complexity than passing those of the layers with 512 and 1024 channels. Some of these extracted features are shown in Fig. 4(a) to Fig. 4(m). Especially, Fig. 4(a) presents heavy rain input, Fig. 4(b) to Fig. 4(d) illustrate features which are active in dark and detail regions where such active features are informatively explored in the following additive sub-net. Fig. 4(e) to Fig. 4(g) depict brighter regions which are explored for the multiplicative sub-net, and Fig. 4(h) to Fig. 4(m) exhibits the features that are active in rain streaks and fog in the input image. For more details, we analyze the roles of three sub-nets in Section V-A.

2) Additive Sub-Net

Removal of heavy rain in a single image is pretty complicated, so it is difficult to separate the physical components such as rain streaks, a transmission map, and atmospheric light from a heavy rain image. Therefore, to tackle this problem, we incorporate the additive and multiplicative sub-nets that separate the heavy rain image into a clean scene...
FIGURE 4. Visualization of features extracted by SCA, Additive and Multiplicative sub-nets. The activation intensities for some interesting features maps of $F_{sca}$ are presented from (b) to (m). Three output features of the second RIM are illustrated from (n) to (p). And three output features of the second CIM are depicted from (q) to (s).

FIGURE 5. The architecture of our Residual inception module (RIM) in the additive sub-net.

and residue components based on two operations: addition and multiplication. In the additive sub-net, we first extract features ($F_{sca}$) from the SCA sub-net’s encoder and then incorporate two residual inception modules (RIM) to predict the additive residues. Fig. 4(n) to Fig. 4(p) visualize some output features of the second RIM. As can be observed, these features are more active in dark and detail regions, confirming that the additive sub-net helps recover high-frequency details and remove rain streaks in the dark regions. After processing the SCA’s features with two RIMs, two convolutional layers are applied to generate the additive residues for each of the three channels of the input images. The additive residues are combined with the heavy rain image input to generate an intermediate clean image estimate $I_1$ as followings:

$$I_1(p) = J(p) + R_A(p)$$  \hspace{1cm} (6)

where $J$ is the heavy rainy input image, $p$ is the pixel location, and $R_A$ denotes additive residues that are learned to decompose input $J$ following (4). $R_A$ is obtained from our additive sub-net by:

$$R_A = add_{subnet}(F_{sca}|\phi_{add})$$  \hspace{1cm} (7)

where $\phi_{add}$ are the learnable filter parameters of our additive sub-net.

Fig. 5 shows the structure of the RIM used in our additive sub-net. This block, based on the inception block [32], consists of three different convolution layers with kernel sizes of 3, 5, and 7, and then concatenates the results to learn additive residues. The three convolution layers with different kernel sizes allow for increased receptive fields and efficiently learn residues in heavy rain images with diverse shapes and various rain directions. With large receptive fields, the RIM block can effectively remove heavy rain streaks and artifacts that often appear in the output of the SCA sub-net.

3) Multiplicative Sub-Net

The deraining work [11] that only employs additive residues learning has a limitation in removing heavy rains according
primary purpose of our separated models is the removal of

Why only Two Multiplicative and Additive Sub-nets:
The primary purpose of our separated models is the removal of rain streaks and fog entanglement that resides spread over the heavy rain images. The rain streaks are formulated as:

\[ J_s = I + S \]  

where \( S \) indicates rain streaks, \( J, I \) are the rainy input image and clean image, respectively. Besides, the fog image is modeled as:

\[ J_f = T \odot I + (1 - T) \odot A \]  

where \( T \) is transmission map, \( A \) is atmospheric light and \( J, I \) are the fog input image and clean output image, respectively. Meanwhile, the previous work \[23\] separates the heavy rainy input image into the rain streaks, an atmospheric light, a transmission map, and background scene. The model is not effective because the rain streaks and fog are entangled in heavy rain scenes. Besides, the fog model and the rain streak model consist of only two arithmetic components: addition and multiplication according to (10) and (11). Consequently, if generating heavy rain from rain streaks and fog in any formula, only two operators are employed: addition and multiplication. This has motivated us to propose two separator sub-nets (see Fig. 2) containing two components: the additive term \( J + R_A \) and the multiplicative term \( J * R_M \) in (4).

Moreover, to effectively combine the three clean images, which are provided by the SCA, additive and multiplicative sub-nets, we use an attentive blending block to create weight maps that linearly combine the three clean estimates as the final rain-free output image. The weight maps are optimized when trained with a loss function of the network. Our superior performance shows the effectiveness of our sub-nets for heavy rain removal. More details of the attentive blending block are discussed in the following sub-section.

Attentive Blending Block: After receiving the three clean estimates from the SCA, additive, and multiplicative sub-nets, we concatenate and feed them as input to the attention mechanism \[21, 45\] to estimate the final clean image. Expressly, three 3 x 3 convolution layers, two 1 x 1 convolution layers, a global average pooling layer, and a sigmoid function are incorporated. As expected, the background scene’s information is automatically highlighted, leading to better recovery of a rain-free image. Furthermore, because there is complementary information in the three clean image results, the attentive blending block can selectively fuse the three intermediate outputs. Hence, our method combines these three sub-nets using these learned attention maps, thus outperforming the SOTA methods in image deraining. The architecture of the attentive blending block is shown in Fig. 7. The adaptive blending weights \( (W_0, W_1, W_2) \) are then given by:

\[ W_0, W_1, W_2 = ATT\_BB(I_0, I_1, I_2|\phi_{ATT\_BB}) \]  

where \( \phi_{ATT\_BB} \) are the learnable filter parameters of our attentive blending block.
Attentive Blending Block

FIGURE 7. The architecture of the Attentive blending block.

B. LOSS FUNCTIONS

We apply an intermediate loss for the three clean image estimates to effectively learn useful features and efficiently remove heavy rain in the input images. As a result, the total loss used to train our end-to-end network architecture consists of two-loss functions measured at the intermediate and the final outputs, respectively. The intermediate loss includes three mean squared error (MSE) terms between the ground truth (GT) clean image and each of the clean estimates of the SCA, additive, and multiplicative sub-nets. The resulting intermediate loss \( L_{\text{inter}} \) is defined as follows:

\[
L_{\text{inter}} = \lambda_0 L_{\text{SCA}} + \lambda_1 L_{\text{Add}} + \lambda_2 L_{\text{Mul}} \tag{13}
\]

\[
L_{\text{SCA}} = L^2(I_{GT}, I_0) \tag{14}
\]

\[
L_{\text{Add}} = L^2(I_{GT}, I_1) \tag{15}
\]

\[
L_{\text{Mul}} = L^2(I_{GT}, I_2) \tag{16}
\]

where \( I_{GT} \) is the ground truth clean image. \( I_0, I_1, \) and \( I_2 \) are the clean estimates of the SCA, additive, and multiplicative sub-nets, respectively. \( \lambda_0, \lambda_1 \) and \( \lambda_2 \) are hyper-parameters that are determined empirically to have \( \lambda_0 = \lambda_1 = \lambda_2 = 1 \).

The loss function in the final output image includes the MSE loss and the edge loss between the clean image output and the GT as follows:

\[
L_{\text{final}} = \lambda_3 L_{\text{MSE}} + \lambda_4 L_{\text{edge}} \tag{17}
\]

\[
L_{\text{MSE}} = L^2(I_{GT}, I) \tag{18}
\]

\[
L_{\text{edge}} = ||H_{x}(I_{GT}) - H_{x}(I)||_1 + ||H_{y}(I_{GT}) - H_{y}(I)||_1 \tag{19}
\]

where \( I \) is the final output of our GLA-HRRNet. \( H_x(\cdot), H_y(\cdot) \) are operators that calculate gradients along the horizontal and vertical directions of the images, respectively. \( \lambda_3 \) and \( \lambda_4 \) are weighting hyper-parameters, empirically set to \( \lambda_3 = \lambda_4 = 1 \). The total loss function used to train our GLA-HRRNet can be expressed as:

\[
L_{\text{total}} = L_{\text{inter}} + L_{\text{final}} \tag{20}
\]

IV. IMPLEMENTATION DETAILS

Training strategy: First, we initialize the network parameters by the Xavier initialization [7]. Then, all training images are randomly cropped into patches with a size of 200 × 300 pixels. We train our network for 200 epochs with a batch size of 4 using the Adam optimizer [19] with the default settings. The learning rate is adjusted following the poly policy in [27] with an initial learning rate of 0.0001, which is halved after 100 epochs. We implemented our proposed GLA-HRRNet on PyTorch and performed our experiments on two NVIDIA TITAN X GPUs.

Training datasets: There are several large-scale synthetic datasets publicly available for training deraining models. However, most datasets do not contain the rain accumulation effects that are often present in the heavy rain scenarios. We selected the Outdoor-Rain dataset [23] which is generated on a set of clean outdoor images. This dataset renders proper rain streaks and rain accumulation effects based on scene depths, which are estimated by a pre-trained monocular depth estimation [8]. The Outdoor-Rain dataset contains 9k training samples and 1.5k test samples.

V. EXPERIMENTS AND RESULTS

In this section, we compare our model, the GLA-HRRNet, with the state-of-the-art methods on both synthetic and real datasets. For the real dataset case, we take the real rainy images published in [23] and images collected from the internet with the keyword “heavy rain photo.” The synthetic rain data is evaluated on the Outdoor-Rain dataset [23] as mentioned in Section IV. For a fair comparison, we download the publicly available source codes and re-train the previous works to generate clean image results from heavy rain images. If the SOTA networks do not converge, we use the best pre-trained models in the corresponding released source codes.

A. ANALYSIS OF INTERMEDIATE RESULTS

As discussed above, our proposed GLA-HRRNet integrates three intermediate sub-nets to generate a final clean output image. This section illustrates the three clean estimates obtained from the three sub-nets: SCA, additive, and multiplicative sub-nets. Fig. 8 shows three clean estimates, \((I_0, I_1, I_2)\) and their corresponding weight maps \((W_0, W_1, W_2)\) from the SCA, additive and multiplicative sub-nets, respectively. We also illustrate the output of additive sub-net, \(R_A\) and

| Method          | PSNR | SSIM |
|-----------------|------|------|
| GLA-HRRNet (ours)| 25.58 | 0.880 |
| DAFNet [11]     | 23.38 | 0.790 |
| HeavyRain [23]  | 20.68 | 0.783 |
| SPANet [34]     | 19.12 | 0.746 |
| PreNet [29]     | 17.82 | 0.719 |
| RESCAN [24]     | 21.77 | 0.709 |
| DID-MDN [42]    | 22.64 | 0.807 |
| UMRL [40]       | 21.58 | 0.816 |
| RCDNet [33]     | 22.43 | 0.814 |
| MSPFN [16]      | 18.25 | 0.726 |
| JCAS [19]       | 16.59 | 0.669 |
FIGURE 8. Visualization of the components in our proposed GLA-HRRNet.

Overall, we can observe that $I_0$ yields a better visual quality in terms of deraining than those of $I_1$ and $I_2$ for each input. This is because the SCA sub-net extracts global features, which help to capture scene geometry information. Nevertheless, it can be noted that $I_0$ still contains artifacts as some features are damaged when removing heavy rain. The additive sub-net learns the additive residues which are then added to the heavy rain input to generate a clean image free of rain streaks. The clean estimate $I_1$ of the additive sub-net contains higher frequency information than that of $I_0$. For more details, Fig. 8(b, c) and Fig. 8(l, m) show some regions where their structures (vertical lines) are damaged in the outputs ($I_0$) of the SCA sub-net, and the clean estimate of the additive sub-net ($I_1$) contains greater details. The multiplicative sub-net learns the multiplicative residues, which selectively extract brighter local features that are not learned effectively in the SCA and additive sub-nets. Such brighter local features are highlighted in the weight map $W_2$ as shown in red boxes in Fig. 8(j, t). Our attentive blending block adaptively combines the informative features from the three intermediate results to yield a clean and sharp final output image $I$.

As shown in Fig. 8, $W_0$ is active in most image regions, which indicates that the SCA sub-net is the main contributor in generating a clean image. This is reasonable, as the SCA is the unique sub-net with receptive fields large enough to reason about the scene geometries and to remove most rain streaks and haze. On the other hand, it can be observed that $W_1$ is the most active in the darker and detailed regions where rain streaks can be the most prominent in the input images. For this reason, the additive sub-net can effectively help in yielding a final clean image with minimal rain streak artifacts and greater details. We highlight a dark region in $W_1$ where the contributions of $I_1$ can be clearly observed in a yellow box in Fig. 8(h, r). Finally, $W_2$, as discussed above, shows higher activations in brighter regions (as can be seen in two red boxes in Fig. 8(j, t)), confirming that the multiplicative residues help modulating the brighter local feature intensities in the final derained output images. Zoom-in Fig. 8 for better visualization.
B. RESULTS ON THE OUTDOOR-RAIN SYNTHETIC DATASET

We use the peak signal to noise ratio (PSNR) [13] and the structural similarity (SSIM) [44] index metrics to compare our GLA-HRRNet with the SOTA methods. Table 1 presents a comparison of our model with the following state-of-the-art methods: JCAS [10], RESCAN [24], DIF-MDN [42], SPANet [34], PreNet [29], UMRL [40], DAFNet [11], HeavyRain [23], MSPFN [16] and RCDNet [33]. Except the JCAS, all methods in Table 1 utilize deep learning-based approaches. It can be observed that the methods trained for plain rain removal [10], [16], [24], [29], [33] perform very poorly on the heavy rain scenario. Meanwhile, the quantitative performance of our GLA-HRRNet is considerably superior to those of the others, outperforming the SOTA methods by more than 2dB in PSNR and 0.06 in SSIM. Table 1 demonstrates the effectiveness of our proposed GLA-HRRNet, which exploits global and local attentions to remove rain and fog entangled in the heavy rains. Moreover, Fig. 9 depicts the qualitative results for our GLA-HRRNet and the SOTA methods. It can be seen that our GLA-HRRNet can effectively remove heavy rains in degraded images and presents very sharp and clean results, while the other methods show unremoved rain streaks, blur, and haze.

C. RESULTS ON THE RAIN100H SYNTHETIC DATASET

In addition to the Outdoor-Rain dataset, we compare our GLA-HRRNet with the SOTA method on the Rain100H dataset [37]. Table 2 presents the performance comparison of our network against the SOTA methods: JCAS [10], RESCAN [24], DIF-MDN [42], UMRL [40], PreNet [29], DAFNet [11], MSPFN [16], and RCDNet [33]. For the Rain100H dataset, our GLA-HRRNet still outperform all methods under comparison by more than 1dB in PSNR and 0.05 in SSIM.

D. RESULTS ON REAL IMAGE DATASETS

Fig. 10 shows the qualitative comparisons for real heavy rain images among the methods in Table 1. In real scenes, the SOTA methods fail to completely remove heavy rain due to the presence of strong rain accumulation, various shapes, and directions of rain streaks. HeavyRain [23] and the DAFNet [11], which are designed to solve the rain and fog entanglement, do not remove heavy rain efficiently, and show color and contrast that are not realistic. One can easily see the artifacts in the results produced by the SOTA methods. Meanwhile, our method still shows realistic color, cleaner and sharper derained images than the other methods under comparison. Zoom-in Fig. 10 for better visualization.

For a more comprehensive comparison, we performed a user study to evaluate the results of SOTA methods on the real image datasets. To do so, we first selected 20 heavy rain images collected from internet with the key word “heavy rain photo”. For a user study with subjective tests, we invited 20 participants, aged from 20 to 31. Then, we showed the outputs (20 derained images) of the SOTA methods and our GLA-HRRNet to each participant and asked to rate their outputs (20 derained images) of the SOTA methods and our GLA-HRRNet to each participant and asked to rate their outputs (20 derained images) of the SOTA methods and our GLA-HRRNet to each participant and asked to rate their outputs (20 derained images) of the SOTA methods and our GLA-HRRNet to each participant and asked to rate their outputs (20 derained images) of the SOTA methods and our GLA-HRRNet.

E. COMPLEXITY COMPARISON WITH NUMBERS OF LEARNABLE PARAMETERS AND INference TIMES

Table 4 compares the computation complexities of rain removal methods in terms of the numbers of parameters and inference times. It can be seen in Table 4 that the inference with our proposed GLA-HRRNet is much faster than the other SOTA methods under comparison. For comparisons, the methods are implemented in PyTorch codes (except the JCAS that is implemented in MATLAB) and were tested on a PC platform with two NVIDIA Titan X GPUs and an Intel Core i7-7700K CPU with 64GB RAM.

### Table 2

| Method          | PSNR  | SSIM  |
|-----------------|-------|-------|
| JCAS [10]       | 17.69 | 0.683 |
| RESCAN [24]     | 26.45 | 0.845 |
| DIF-MDN [42]    | 25.00 | 0.754 |
| UMRL [40]       | 26.01 | 0.832 |
| PreNet [29]     | 26.77 | 0.858 |
| DAFNet [11]     | 28.44 | 0.874 |
| MSPFN [16]      | 28.66 | 0.860 |
| DID-MDN [42]    | 27.86 | 0.835 |
| GLA-HRRNet (ours) | 29.88 | 0.927 |

### Table 3

| Method          | Rating scores (mean ± std.) |
|-----------------|-----------------------------|
| JCAS [10]       | 6.07 ± 1.87                 |
| DAFNet [11]     | 6.07 ± 1.87                 |
| DIF-MDN [42]    | 5.89 ± 1.67                 |
| HeavyRain [23]  | 5.07 ± 1.87                 |
| RCDNet [33]     | 6.62 ± 1.64                 |
| GLA-HRRNet (ours) | 6.62 ± 1.64             |

### Table 4

| Method          | No. parameters | Inference times (sec.) |
|-----------------|----------------|------------------------|
| JCAS [10]       | x              | 608.9                  |
| DAFNet [11]     | 4.0056M        | 0.86                   |
| DIF-MDN [42]    | 66.0056M       | 0.07                   |
| HeavyRain [23]  | 40.0027M       | 0.75                   |
| RCDNet [33]     | 3.1668M        | 1.96                   |
| GLA-HRRNet (ours) | 23.4875M      | 0.02                   |
FIGURE 9. Comparison of the proposed model with the SOTA methods on the synthetic Outdoor-Rain dataset [23].
FIGURE 10. The comparison results of proposed methods with SOTAs on real datasets.
In the ablation study, we evaluated

### TABLE 5. Ablations studies of the proposed sub-nets.

| Network          | SCA | Add Mul | PSNR | SSIM |
|------------------|-----|---------|------|------|
| SCA              | √   |         | 24.46| 0.848|
| Add              | √   |         | 23.81| 0.859|
| Mul              | √   |         | 22.96| 0.814|
| SCA + Add        | √   | √       | 24.95| 0.873|
| SCA + Mul        | √   | √       | 24.85| 0.872|
| Add + Mul        | √   | √       | 24.31| 0.867|
| Full GLA-HRRNet  | √   | √       | 25.58| 0.880|

### TABLE 6. Ablations studies of our proposed SCA block.

| Method          | SA  | CA  | PSNR | SSIM |
|-----------------|-----|-----|------|------|
| No CA           | √   |     | 24.36| 0.875|
| No SA           |     | √   | 25.14| 0.876|
| No CA-SA        |     | √   | 24.18| 0.870|
| Full GLA-HRRNet | √   | √   | 25.58| 0.880|

### VI. CONCLUSION

In this paper, we have proposed an end-to-end network that integrates three sub-nets, called GLA-HRRNet. Our proposed network exploiting global and local attention is capable of effectively removing heavy rain in single images where the strong rain streaks and fog are entangled. The first sub-net, called the spatial channel attention (SCA) sub-net, is designed based on a U-net architecture to extract global features that aid in predicting an initial rain-and-fog-free image. The second *additive* sub-net based on our proposed Residual inception modules (RIM) learns to predict additive residues. And finally, the third *multiplicative* sub-net based on our channel-attentive inception modules (CIM) learns informative brighter local features to modulate the local pixel intensities. The final result is produced by adaptively blending our three clean image estimates in our attentive blending block. With the SCA, additive, and multiplicative sub-nets integrated into our full network, the GLA-HRRNet has shown significantly better performance of heavy rain removal than the recent SOTA methods quantitatively and qualitatively throughout comprehensive experiments for both synthetic and real datasets.

### REFERENCES

[1] Y. Chen and C. Hsu. A generalized low-rank appearance model for spatio-temporally correlated rain streaks. In 2013 IEEE International Conference on Computer Vision (ICCV), pages 1968–1975, 2013.
[2] D. Comaniciu, V. Ramesh, and P. Meer. Kernel-based object tracking. IEEE Transactions on Pattern Analysis and Machine Intelligence, 25(5):564–577, 2003.
[3] Liang-Jian Deng, Ting-Zhu Huang, Xi-Le Zhao, and Tai-Xiang Jiang. A directional global sparse model for single image rain removal. Applied Mathematical Modelling, 39:662 – 679, 2018.
[4] X. Fu, J. Huang, D. Zeng, Y. Huang, X. Ding, and J. Paisley. Removing rain from single images via a deep detail network. In 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 1715–1723, 2017.
[5] 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, 31(6):1794–1807, 2020.
[6] K. Garg and S. K. Nayar. Detection and removal of rain from videos. In Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pages I–I, 2004.
[7] Xavier Glorot and Yoshua Bengio. Understanding the difficulty of training deep feedforward neural networks. In AISTATS‘10, pages 249–256, Chia Laguna Resort, Sardinia, Italy, 13–15 May 2010.
[8] C. Godard, O. M. Aodha, and G. J. Brostow. Unsupervised monocular depth estimation with left-right consistency. In 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 6602–6611, 2017.
[9] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In Advances in Neural Information Processing Systems, pages 2672–2680, 2014.
[10] S. Gu, D. Meng, W. Zuo, and L. Zhang. Joint convolutional analysis and synthesis sparse representation for single image layer separation. In 2017 IEEE International Conference on Computer Vision (ICCV), pages 1717–1725, 2017.
[11] X. Hu, C. Fu, L. Zhu, and P. Heng. Depth-attentional features for single-image image rain removal. In 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pages 8014–8023, 2019.
[12] X. Hu, X. Xu, Y. Xiao, H. Chen, S. He, J. Qin, and P. Heng. Sinet: A scale-insensitive convolutional neural network for fast vehicle detection. IEEE Transactions on Intelligent Transportation Systems, 20(3):1010–1019, 2019.
[13] Q. Huynh-Thu and M. Ghanbari. Scope of validity of psnr in image/video quality assessment. Electronics Letters, 44(13):800–801, 2008.
[14] Joel Janai, Fatma Güney, Aseem Behl, and Andreas Geiger. Computer vision for autonomous vehicles: Problems, datasets and state of the art. Foundations and Trends® in Computer Graphics and Visualization, 12(1–3):1–308, 2020.
[15] 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, pages 1–1, 2020.
[16] K. Jiang, Z. Wang, P. Yi, C. Chen, B. Huang, Y. Lao, J. Ma, and J. Jiang. Multi-scale progressive fusion network for single image deraining. In 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pages 8343–8352, 2020.
[17] Justin Johnson, Alexandre Alahi, and Li Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. In European Conference on Computer Vision (ECCV), 2016.
correlation and low-rank matrix completion. IEEE Transactions on Image Processing, 24(9):2658–2670, 2015.

[19] Diederik Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In ICLR, 2015.

[20] K. H. Lee, E. Ryu, and J. O. Kim. Progressive rain removal via a recurrent convolutional network for real rain videos. IEEE Access, 8:203134–203145, 2020.

[21] G. Li, Y. Xie, L. Lin, and Y. Yu. Instance-level salient object segmentation. In 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 247–256, 2017.

[22] M. Li, Q. Xie, Q. Zhao, W. Wei, S. Gu, J. Tao, and D. Meng. Video rain streak removal by multiscale convolutional sparse coding. In 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 6644–6653, 2018.

[23] R. Li, L. Cheong, and R. T. Tan. Heavy rain image restoration: Integrating physics model and conditional adversarial learning. In 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pages 1633–1642, 2019.

[24] Xia Li, Jianlong Wu, Zhouchen Lin, Hong Liu, and Hongbin Zha. Recurrent squeeze-and-excitation context aggregation net for single image deraining. In European Conference on Computer Vision (ECCV), pages 262–277, 2018.

[25] Y. Li, R. T. Tan, X. Guo, J. Lu, and M. S. Brown. Rain streak removal using layer priors. In 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 2736–2744, 2016.

[26] J. Liu, W. Yang, S. Yang, and Z. Guo. Erase or fill? deep joint recurrent rain removal and reconstruction in videos. In 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 3233–3242, 2018.

[27] Wei Liu, Andrew Rabinovich, and Alexander C. Berg. Parsenet: Looking wider to see better. CoRR, 2015.

[28] Y. Luo, Y. Xu, and H. Ji. Removing rain from a single image via discriminative sparse coding. In 2015 IEEE International Conference on Computer Vision (ICCV), pages 3397–3405, 2015.

[29] D. Ren, W. Zuo, Q. Hu, P. Zhu, and D. Meng. Progressive image deraining networks: A better and simpler baseline. In 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pages 3932–3941, 2019.

[30] W. Ren, J. Tian, Z. Han, A. Chan, and Y. Tang. Video desnowing and deraining based on matrix decomposition. In 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 2838–2847, 2017.

[31] O. Ronneberger, P. Fischer, and T. Brox. U-net: Convolutional networks for biomedical image segmentation. In Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015, pages 234–241, 2015.

[32] C. Szegedy, Wei Liu, Yangqing Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. Going deeper with convolutions. In 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 1–9, 2015.

[33] H. Wang, Q. Xie, Q. Zhao, and D. Meng. A model-driven deep neural network for single image rain removal. In 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pages 3100–3109, 2020.

[34] T. Wang, X. Yang, K. Xu, S. Chen, Q. Zhang, and R. W. H. Lau. Spatial attentive single-image deraining with a high quality real rain dataset. In 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pages 12262–12271, 2019.

[35] X. Wang, Z. Li, H. Shan, Z. Tian, Y. Ren, and W. Zhou. Fastderainnet: A deep learning algorithm for single image deraining. IEEE Access, 8:127622–127630, 2020.

[36] Yinglong Wang, Yibing Song, Chao Ma, and Bing Zeng. Rethinking image deraining via rain streaks and vapers, 2020.

[37] 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 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 1685–1694, 2017.

[38] W. Yang, R. T. Tan, S. Wang, Y. Fang, and J. Liu. Single image deraining: From model-based to data-driven and beyond. IEEE Transactions on Pattern Analysis and Machine Intelligence, pages 1–1, 2020.

[39] W. Yang, R. T. Tan, S. Wang, and J. Liu. Self-learning video rain streak removal: When cyclic consistency meets temporal correspondence. In 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pages 1717–1726, 2020.

[40] R. Yasarla and V. M. Patel. Uncertainty guided multi-scale residual learning using a cycle spinning cnn for single image de-raining. In 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pages 8397–8406, 2019.