# Uncertainty-Aware Sparse Transformer Network for Single-Image Deraindrop

Bo Fu, Yunyun Jiang, Di Wang, Jiaxin Gao, Cong Wang, and Ximing Li

***Abstract*— Image Deraindrop aims to enhance the visibility and clarity of the image by eliminating unwanted visual artifacts, such as raindrops or rain streaks. Despite remarkable advancements in image raindrop removal, the sparse distribution of raindrops and the various levels of degradation within raindrop regions are still not fully considered: 1) globally, raindrops often exhibit a unique sparse distribution in images, but the existing methods apply a uniform treatment of pixels and 2) locally, raindrops have specific degradation within raindrop regions, such as variations in shape, size, and transparency, but current approaches fail to effectively model them. To address these problems, this work proposes an uncertainty-aware sparse Transformer network (USTN) for image Deraindrop. Specifically, to characterize the sparsity of raindrops, we develop a sparse Transformers backbone in which several sparse Transformers blocks (STBs) are deployed at each scale. To ensure effective sparse feature learning, we introduce a top-k sparse attention (TSA) layer in each STB, which dynamically selects high-score attention and generates corresponding sparse feature responses. To effectively model various degrees of degradation in local raindrop regions, we incorporate image uncertainty estimation, which can explicitly show the observation that worse degradation exhibits higher uncertainty. Motivated by this, we design two decoding branches, one for estimating uncertainty maps and the other for generating raindrop-free images. We then formulate an uncertainty-driven loss to better push the USTN to remove raindrops guided by uncertainty maps. In addition, to further refine the learned sparse features, we propose a pyramid feature refinement (PFR) module to fully mine the local features under multiscale receptive fields and a residual channel–spatial attention (RCSA) module to stimulate the effective expression of the deepest features. Extensive experiments demonstrate that the proposed USTN outperforms state-of-the-art methods and is top performing. We also apply USTN to the semantic segmentation task to reveal the promising semantic-aware capability.**

Received 16 April 2024; revised 13 July 2024; accepted 26 July 2024. Date of publication 3 October 2024; date of current version 17 October 2024. This work was supported in part by the Research funding Project of Liaoning Provincial Department of Education under Grant LJKZ0986, in part by the National Natural Science Foundation of China under Grant 62276113, and in part by the Project Funded by China Postdoctoral Science Foundation under Grant 2022M721321. The Associate Editor coordinating the review process was Dr. Eduardo Cabal-Yepez. *(Corresponding author: Di Wang.)*

Bo Fu is with the School of Computer and Artificial Intelligence, Liaoning Normal University, Dalian 116081, China (e-mail: fubo@lnnu.edu.cn).

Yunyun Jiang is with the School of Computer Science and Engineering, Northeastern University, Shengyang 110169, China (e-mail: 2301884@stu.neu.edu.cn).

Di Wang and Jiaxin Gao are with the School of Software Technology, Dalian University of Technology, Dalian 116024, China (e-mail: diwang1211@mail.dlut.edu.cn; jiaxinn.gao@outlook.com).

Cong Wang is with the Department of Computing, The Hong Kong Polytechnic University, Hong Kong, China (e-mail: supercong94@gmail.com).

Ximing Li is with the College of Computer Science and Technology, Jilin University, Changchun 130012, China (e-mail: ximingli@jlu.edu.cn).

Digital Object Identifier 10.1109/TIM.2024.3472902

***Index Terms*— Image Deraindrop, pyramid feature refinement (PFR), residual channel–spatial attention (RCSA), sparse Transformers, uncertainty estimation.**

## I. Introduction

DUE to the raindrops adhering to the lens on a rainy day, the captured images often suffer serious content loss, such as edges, textures, colors, and background details [1], [2], [3]. This not only leads to declined visual quality but also adversely affects various high-level semantic perception tasks, such as object detection, semantic segmentation, autonomous driving, and so on. Therefore, the removal of raindrops becomes particularly crucial. Given an image with raindrops, the goal of raindrop removal is to restore the corresponding clear image from this degraded image. In recent years, extensive studies have been dedicated to addressing this ill-posed problem. Among them, the most popular approaches are plain CNN-based methods [3], [4], GAN-based methods [2], and Transformer-based methods [5], [6].

Early, CNN-based image Deraindrop methods [4] simply employ a network with small parameter sizes to perform raindrop removal tasks, resulting in inferior results. With the emergence of various attention mechanisms, their adaptive feature representation ability attracts considerable attention. Quan et al. [3] introduce shape-driven attention and channel recalibration techniques, leveraging mathematical modeling of raindrops, to attain better raindrop-free images. To extract more discriminative features for raindrop image restoration, GAN-based methods have emerged. Typically, Qian et al. [2] propose an attention-based generative adversarial network that incorporates channel attention to model raindrop priors, guiding the discriminator to maintain local consistency in the restored areas. These two categories of methods demonstrate favorable performance in raindrop removal with the support of attention mechanisms. However, their network architectures rely on plain convolutions, which limits their ability to characterize image information within a fixed receptive field and prevents them from effectively modeling long-range global feature dependencies. To overcome this limitation, some methods [7] introduce Transformers (e.g., ViT [6] and SwinT [8]) as the fundamental components of network structure. This is because their internal self-attention and cross-attention mechanisms excel at modeling nonlocal feature representations.

Although impressive results have been achieved in raindrop removal, the sparse distribution of raindrops and various levels


(a) Raindrop image (b) DA-CLIP [7] (c) MCW [16] (d) Uncertainty map (e) Ours (f) GT

Fig. 1. Comparison of images before and after raindrop removal. (a) Input raindrop image. (b) Raindrop removal image generated by using DA-CLIP. (c) Raindrop removal image generated by using MCW. (d) Uncertainty map generated from the raindrop image. (e) Raindrop removal image generated by using our method. (f) Clean image. Our method performs superior raindrop removal results. It effectively removes raindrops and restores sharp background details.

of degradation within raindrop regions are still not fully considered as follows.

1) Globally, raindrops often exhibit unique sparse distribution in images, but the existing methods exhibit a uniform treatment of pixels. The existing methods frequently treat each pixel of the whole image equally, resulting in the potential destruction of content in raindrop-free regions and incomplete restoration of content in regions affected by raindrops.
2) Locally, the different degradation levels of raindrop regions are not depicted, which are manifested as differences in shape, size, and transparency, but fail to effectively model them. The existing methods utilize the attention mechanism to model the raindrop map, considering it merely as a guide to facilitate the overall network convergence, without explicitly addressing the distinctions between individual local raindrop regions.

To address these issues, this article proposes an uncertainty-aware sparse Transformer network (USTN), which effectively removes raindrops and restores sharp background details in images. Given the powerful capability of Transformers in modeling long-range feature dependencies, we lean toward leveraging Transformers to extract and learn image features. However, considering the sparsity of raindrop distribution, directly using standard Transformers [6], [8] may result in capturing global information that includes redundant and irrelevant features. Therefore, we devise a sparse Transformer backbone tailored to learn the sparse feature representation of raindrop images. To be specific, we employ several sparse Transformer blocks (STBs) inspired by [9] in each scale of this backbone. In each STB, a top-k sparse attention (TSA) layer is employed instead of the standard self-attention layer. This enables the dynamic selection of the top-k highest scoring attentions, facilitating the generation of the corresponding sparse features for raindrop images.

Aiming at varying degrees of degradation in local raindrop regions, we attempt to seek a tool to explicitly extract the raindrop layer from the image. Benefit from image uncertainty [10], [11], [12] can explicitly interpret the observation that regions with more severe degradation exhibit higher uncertainty, and we build an uncertainty-driven loss, which forces the network to focus on raindrop region restoration. For this, we also design two decoded branches, one for the generation of the restored image and the other for the estimation of the uncertainty map. In addition, to further enhance the learning of sparse feature representation from raindrop images, we incorporate pyramid feature refinement (PFR) modules and residual channel–spatial attention (RCSA) modules into the backbone. The former utilizes dilated convolutions with varying dilation factors, in conjunction with shuffle attention (SA) [13], to extensively extract local features across multi-scale receptive fields. The latter utilizes channel attention [14] and spatial attention [15] to emphasize informative features in the deepest layer features, preventing them from being overlooked. We conduct extensive experiments to validate the effectiveness of the proposed USTN. As shown in Fig. 1, our USTN outperforms the two latest methods (i.e., DA-CLIP [7] and MCW [16]), performing superior raindrop removal results. It effectively removes raindrops and restores sharp background details. The main contributions of this article are as follows.

1) A USTN for image Deraindrop is introduced. It not only captures global sparse representations of raindrop images but models various local degradations in raindrop regions, restoring raindrop-free images with sharp backgrounds.
2) We introduce the TSA to customize the sparse Transformer block and apply it to the feature learning backbone. This effectively captures global sparse feature dependencies in raindrop images while suppressing redundant and irrelevant representations.
3) We employ uncertainty to model the diverse degradation of raindrop regions and build an uncertainty-driven loss to encourage the network to focus more on the restoration of raindrop regions. This effectively removes raindrops and restores their backgrounds.

The rest of this article is organized as follows. Section II summarizes the related works on Deraindrop methods and uncertainty estimation. Section III elaborates on the network architecture of the USTN and the loss functions. Section IV presents the quantitative and qualitative comparison with the SOTA methods and provides ablation analyses of each key module. Section V concludes this work.

## II. Related Works

In recent years, the research on single-image raindrop removal has received more and more attention. In this article,

we propose to use uncertainty estimation to assist the network in removing raindrops. A brief review of the most related works is as follows.

### A. Single-Image Raindrop Removal

Raindrop removal is a formidable challenge due to the complex and variable nature of raindrops. Based on the review and collation of the available studies, it is evident that only a few methods dedicated to single-image raindrop removal have been proposed [1], [2], [3], [4], [17], [18], compared with image restoration [19] and single-image rain streak removal [20], [21], [22], [23]. However, numerous image restoration works [7], [24], [25], [26], [27], [28], [29], [30] and single-image rain streak removal works [16], [31], [32], [33] offer empirical evidence of the effectiveness of their methods in raindrop removal.

To overcome this challenge, numerous CNN-based frameworks [3], [4], [16], [17], [26], [27], [28] have been developed to solve single-image raindrop removal. To the best of our knowledge, Eigen et al. [4] first propose the learning-based method for raindrop removal, but the image quality generated is poor due to the small network architecture. To accurately locate raindrops, Quan et al. [3] utilize the shape characteristics of raindrops to mathematically model raindrops, and they propose using shape-driven attention and channel recalibration to improve generated image quality. Furthermore, Li et al. [17] observe that there would be dual-pixel disparities in raindrop regions, but not in clean background regions. Based on this observation, they propose the first dual-pixel raindrop removal network. In addition, Chen et al. [28] propose TKLMR, a unified architecture that addresses image degradation caused by diverse adverse weather conditions, including raindrops, haze, and snow. In recent years, single-image rain streak removal methods have become increasingly popular [34], [35], [36], which, respectively, achieve superior image rain removal performance by fully mining multiscale features [35] and cross-scale relationship [36]. Moreover, several works attempt to eliminate raindrops and rain streaks from single images simultaneously [37], [38]. Furthermore, although not explicitly focused on raindrop removal, numerous image restoration works support their effectiveness with experimental evidence of raindrop removal [24], [25], [26].

In recent years, GAN-based methods have been widely popular in the field of low-level computer vision, such as image Deraindrop [1], [2], [18], waterdrop removal [39], and rain-like layer removal [40]. Qian et al. [2] propose the combination of generative adversarial network and attention mechanism to generate raindrop-free images; more importantly, they also release a real-world paired raindrop dataset, which greatly facilitates the development of single-image raindrop removal research. Yan and Loke [1] regard the single-image raindrop removal as a many-to-one image-to-image translation task; to the best of our knowledge, it is the first unsupervised learning method that effectively alleviates the challenge of the existing raindrop removal methods relying on paired raindrop datasets. In addition, UnfairGAN [18] proposes an enhanced generative adversarial network that can utilize prior high-level information to improve Deraindrop performance.

Recently, another class of neural network architectures, Transformers [5], [6], [7], [10], [29], [30], [32], has shown significant performance gains in natural language and computer vision tasks. The Transformer-based methods overcome the limitations of CNNs (i.e., local receptive field). IDT [32] is an effective and efficient Transformer-based framework for image deraining that incorporates visual task priors. DA-CLIP [7] transfers pretrained vision-language models to low-level vision tasks as a universal framework for image restoration. UDR-S2Former [10] is dedicated to restoring underlying clean images from images both raindrops and rain streaks have damaged. TransWeather [29] and GridFormer [30] are proposed to restore images degraded by all adverse weather conditions. The above methods all provide qualitative experimental results on the Deraindrop dataset [2] to demonstrate its effectiveness on single-image raindrop removal.

Although the works mentioned above demonstrate satisfactory raindrop removal performance, they ignore the arbitrary uncertainty in the input data. Specifically, the inherent noise information of the training data has not been fully mined, which limits their raindrop removal performance. Therefore, this article attempts to estimate the arbitrary uncertainty in raindrop images to guide the network to achieve better raindrop removal performance.

### B. Uncertainty Estimation

At present, some works have studied the specific performance of uncertainty in deep neural networks [12], [41], [42], [43]. Kendall and Gal [12] divide uncertainty into two categories: arbitrary uncertainty and epistemic uncertainty. The former is related to the inherent noise of the training data, and the latter is used to capture the parameter noise in the deep neural network. When a large amount of data are available, the epistemic uncertainty can be reduced.

In recent years, many computer vision tasks [11], [12], [44], [45], [46] have introduced uncertainty estimation into deep learning models, such as image super-resolution [44], image defogging [11], face recognition [45], image classification [46], and semantic segmentation [12]. Among them, image super-resolution [44], image defogging [11], and face recognition [45] model the arbitrary uncertainty in the input, which is also the focus of this article. To reduce the impact of inherent noise in the training data, Kendall and Gal [12] fix a Gaussian likelihood to model the arbitrary uncertainty loss function

$$L = \frac{1}{N}\sum_{i=1}^{N}\left(\frac{\|y_i - f(x_i)\|^2}{2\sigma_i^2} + \frac{1}{2}\log\sigma_i^2\right). \tag{1}$$

Here, $f(x_i)$ and $y_i$ are the predicted value and ground truth for pixel $i$, the variance $\sigma_i$ of pixel $i$ represents the noise scalar, and $N$ is the total number of pixels. Based on the formula, we find that the value of the loss function can be adaptively adjusted by the uncertainty of the network prediction.

For the sake of clarity, we refer to the first term of the arbitrary uncertainty loss function as the residual term and the second term as the regular term. We employ arbitrary uncertainty estimation to assist the network in raindrop removal.

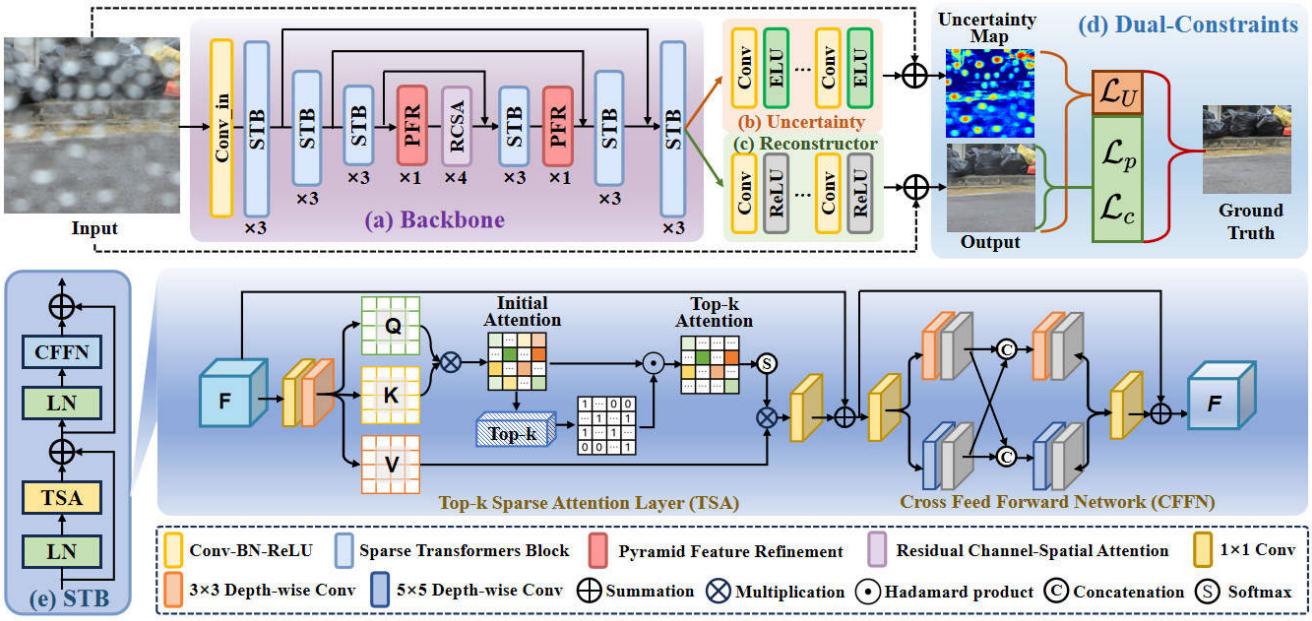



Fig. 2. Workflow of the USTN. It consists of four key parts: (a) sparse Transformers-based backbone, (b) uncertainty estimator, (c) feature reconstructor, and (d) dual constraints. The core module of the backbone, known as the STB, is depicted in (e). Among them, the loss constraints involved in (d) include content loss $\mathcal{L}_c$, uncertainty-driven loss $\mathcal{L}_U$, and perceptual loss $\mathcal{L}_p$.

In the generated raindrop-free image, the penalty to the model in areas with large prediction errors (high uncertainty) is primarily governed by the regular term of the loss function. Conversely, in areas with small prediction errors (low uncertainty), the penalty to the model is mainly determined by the residual term of the loss function. However, due to the small prediction error in this region [it is more confident that they are the correct prediction (GT)], the value of the residual term will also be small.

## III. Methodology

In this section, we begin by presenting an overview of the proposed USTN. Subsequently, we delve into the network architecture and each pivotal module. Finally, we elucidate the formulation of the loss functions governing network training.

### A. Overview

In this work, we propose a network for raindrop removal, termed USTN. Given a raindrop image $\mathcal{X} \in \mathbb{R}^{H\times W\times 3}$, the goal of USTN is to generate the corresponding clean image $\mathcal{Y} \in \mathbb{R}^{H\times W\times 3}$ with sharp textures, devoid of raindrops. As shown in Fig. 2, the USTN consists of four core parts: a sparse Transformers-based backbone with a U-like structure for encoding and decoding features, an uncertainty estimator for predicting uncertainty maps, a feature reconstructor for generating a clean image without raindrops, and dual constraints for optimizing the training process.

Specifically, the backbone comprises an encoder, two PFR layers, four RCSA layers, and a decoder. We first utilize this encoder to extract multiscale features $\mathcal{F}^i_{\text{Enc}} = \mathbb{E}(\mathcal{X}; \Theta_E)$, where $i \in \{0, 1, 2\}$ and $\Theta_E$ refers to the learnable parameters of the encoder. Then, we plug a PFR layer into the deepest STB, which is responsible for refining encoded features $\mathcal{F}_2$. Similarly, the other PFR layer is symmetrically integrated into the corresponding layer of the decoder. Positioned at the lower end of this backbone, four RCSA layers are strategically placed to promote the enhancement of discriminative features for effective decoding, thus obtaining multiscale features $\mathcal{F}^i_{\text{Dec}} = \mathbb{D}(\mathbf{x}; \Theta_D)$, where $\mathbf{x}$ represents the input of the decoder and $\Theta_D$ refers to the learnable parameters of the decoder. Subsequently, we introduce an uncertainty estimator $\mathbb{U}$ with weights $\Theta_U$ and a feature reconstructor $\mathbb{R}$ with weights $\Theta_R$ to predict uncertainty maps $\mathcal{S}$ and generate clean images $\mathcal{O}$, respectively. As depicted in Fig. 2, the $\mathbb{U}$ comprises several Conv-ELU (exponential linear unit) blocks, while the $\mathbb{R}$ consists of several Conv-ReLU (rectified linear unit) blocks. To optimize the objective $\mathcal{O} \to \mathcal{T}$, where $\mathcal{T}$ is ground truth, we design dual output-specific constraints to facilitate the training of the proposed USTN. In the forthcoming sections, we delve into a detailed exposition of the core modules within our USTN.

### B. Sparse Transformers Block

Considering the natural sparsity of raindrop distributions in images, we intend to employ the STB [9] as a fundamental module for feature encoding and decoding. This preference stems from that standard Transformers [6] introduce redundant relations across all query–key pairs and irrelevant representations. As shown in Fig. 2(e), similar to [9], the STB includes two residual steps. Given the encoded features $\mathcal{F}^{j-1}_{\text{Enc}}$ at the $(j-1)$th scale, the current encoding of the STB can be formulated as follows:

$$\begin{aligned} \mathbf{x}'_j &= \mathbf{x}_{j-1} + \text{TSA}\big(\text{LN}\big(\mathbf{x}_{j-1}\big)\big) \\ \mathbf{x}_j &= \mathbf{x}'_j + \text{CFFN}\big(\text{LN}\big(\mathbf{x}'_j\big)\big) \end{aligned} \tag{2}$$

where LN(·) denotes the layer normalization and CFFN(·) denotes the cross feedforward network. $\mathbf{x}'_j$ is the output of the first residual step. The TSA(·) represents the TSA layer, which replaces the self-attention layer in the standard Transformers and plays a pivotal role in modeling sparse correlations across query–key pairs of tokens in images. The bottom of Fig. 2 shows the detailed schematics of TSA and CFFN.

*1) TSA Layer:* Taking the features $\mathcal{F}_{\text{Enc}}^{j-1}$ with a dimension of $H/2^{j-1} \times W/2^{j-1} \times 2^{j-1}C$ as input of the current STB, the TSA layer first splits these features into query **Q**, key **K**, and value **V** with dimension of $\mathbb{R}^{L\times d}$, $L = H/2^{j-1} \times W/2^{j-1}$. The initial attention is obtained by

$$\mathbf{Att}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \text{Softmax}\left(\frac{\mathbf{Q}\mathbf{K}^{\top}}{\tau}\right)\mathbf{V}. \tag{3}$$

Here, $\tau$ denotes temperature factor defined by $\tau = (d)^{1/2}$, and $d$ refers to feature channels. Subsequently, we introduce an adaptive selection operation where the highest scoring top-k elements are extracted from **Att** while discarding the remaining elements, thus preserving strong correlations for the query–key pairs. This dense-to-sparse attention selection process can be formulated as follows:

$$\text{TSA}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \text{Softmax}\left(\text{T}_{\text{k}}\left(\frac{\mathbf{Q}\mathbf{K}^{T}}{\tau}\right)\right)\mathbf{V} \tag{4}$$

where $\text{T}_{\text{k}}(\cdot)$ denotes top-k selection operation, which is dynamically learnable. Specifically, the top-k attention can be defined by

$$[\text{T}_{\text{k}}(A)]_{ij} = \begin{cases} A_{ij}, & A_{ij} \geq s_i, \\ 0, & \text{otherwise}. \end{cases} \tag{5}$$

Here, $s_i$ denotes the $k$th largest value in the $j$th row of $(\mathbf{Q}\mathbf{K}^T/\tau)$. Any element $A_{ij}$ that does not satisfy $A_{ij} \geq s_i$ is set to zero. Then, the sparse attention $[\text{T}_{\text{k}}(A)]$ is utilized to recalibrate **V**, resulting in enhanced features $\mathbf{x}'_j$ at the $j$th scale as (2).

*2) Cross Feedforward Network:* At the second residual step, given features $\text{LN}(\mathbf{x}'_j)$ after layer normalization, we send them into a parallel structure composed of $3\times3$ and $5\times5$ depthwise convolutions [47] for local information extraction. In this way, the local feature extraction can be formulated as follows:

$$\begin{aligned}
\mathbf{x}_j^m &= \text{Conv}_{1\times1}\left(\text{LN}\left(\mathbf{x}'_j\right)\right) \\
\mathbf{x}_j^{m-t1} &= \delta\left(\text{DWConv}_{3\times3}\left(\mathbf{x}_j^m\right)\right) \\
\mathbf{x}_j^{m-b1} &= \delta\left(\text{DWConv}_{5\times5}\left(\mathbf{x}_j^m\right)\right) \\
\mathbf{x}_j^{m-t2} &= \delta\left(\text{DWConv}_{3\times3}\left[\mathbf{x}_j^{m-t1}, \mathbf{x}_j^{m-b1}\right]\right) \\
\mathbf{x}_j^{m-b2} &= \delta\left(\text{DWConv}_{5\times5}\left[\mathbf{x}_j^{m-b1}, \mathbf{x}_j^{m-t1}\right]\right) \\
\mathbf{x}_j &= \text{Conv}_{1\times1}\left[\mathbf{x}_j^{m-t2}, \mathbf{x}_j^{m-b2}\right] + \mathbf{x}'_j
\end{aligned} \tag{6}$$

where $\delta(\cdot)$ is a ReLU, $\text{Conv}_{1\times1}$ represents $1 \times 1$ convolution, $\text{DWConv}_{3\times3}$ and $\text{DWConv}_{5\times5}$ refer to $3 \times 3$ and $5 \times 5$ depthwise convolutions, respectively, and $[\cdot]$ is concatenation.

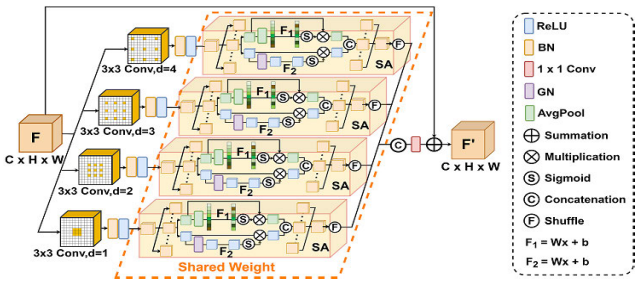



Fig. 3. Workflow of the PFR module. Note that the SA layer [13] enhances the feature representation ability of the CNN network by grouping features in the channel dimension and processing them in parallel.

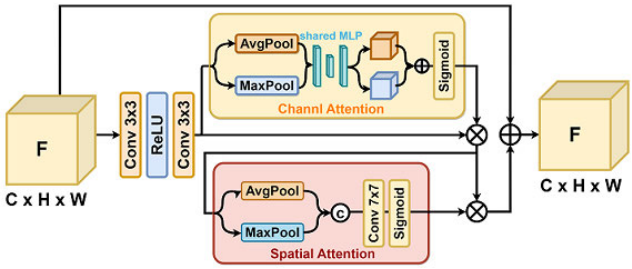



Fig. 4. Workflow of the RCSA module. It optimizes feature representation by using channel attention and spatial attention to process the extracted features.

### C. Pyramid Feature Refinement

To further refine the features encoded through STBs, PFR modules are embedded after the highest level STBs in both the encoder and decoder, as shown in Fig. 2. The detailed structure of the PFR is depicted in Fig. 3. The PFR is a parallel module composed of dilated convolutions with different dilation coefficients and SA layers attention mechanism [13]. As we all know, dilated convolution can improve the global representation of features by expanding the receptive field, but its essence is still convolution to extract local features within the receptive field. Also, different dilation coefficients are set for the dilated convolution, which also plays a sparse selection effect to a certain extent. The SA layers attention mechanism [13] groups features in the channel dimension and processes the grouped features in parallel, thereby enhancing the ability of the network to model local information. Note that the PFR does not change the size of the features, so it has good migration capabilities and is a plug-and-play module. As mentioned above, the entire PFR feature extraction operation process can be formulated as follows:

$$\begin{aligned}
\mathbf{x}_j^d &= SA\left(\delta\left(\text{BN}\left(\text{DConv}_{3\times3}^d\left(\mathbf{x}_j\right)\right)\right)\right) \\
\hat{\mathbf{x}}_j &= \text{Conv}_{1\times1}\left(\left[\mathbf{x}_j^d\right]\right) + \mathbf{x}_j
\end{aligned} \tag{7}$$

where $SA(\cdot)$ represents the SA layers attention mechanism, $\delta(\cdot)$ represents ReLU activation, BN(·) represents batch normalization [48], $\text{Conv}_{1\times1}$ represents $1 \times 1$ convolution, and $\text{DConv}_{3\times3}^d$ refers to the dilated convolution with a kernel size of $3 \times 3$ and a dilation coefficient of $d$. Note that the value range of $d$ is $[0, 1, 2, 3]$.

### D. Residual Channel–Spatial Attention

At the bottom of this network, we embed 4 RCSA modules. Their structure is shown in Fig. 4, The module has two key

components, i.e., channel attention ($\text{Att}_\text{c}$) [14] and spatial attention ($\text{Att}_\text{s}$) [15]. The intermediate features are refined through the RCSA. Therefore, it is used to assist the PFR module in further processing the extracted features to ensure effective representation of useful features in the deepest layers of the network. Given features $\hat{\mathbf{x}}_{j=2}$ with a dimension of $H/4 \times W/4 \times 4C$, the RCSA can be formulated as follows:

$$\begin{aligned} \bar{\mathrm{x}} &= \text{Conv}_{3\times3}\big(\delta\big(\text{Conv}_{3\times3}\big(\hat{\mathbf{x}}_{j=2}\big)\big)\big) \\ \bar{\mathrm{x}}_c &= \text{Att}_\text{c}\big(\bar{\mathrm{x}}\big) \otimes \bar{\mathrm{x}} \\ \bar{\mathrm{x}}_s &= \text{Att}_\text{s}\big(\bar{\mathrm{x}}_c\big) \otimes \bar{\mathrm{x}}_c \\ \mathrm{x}_\text{out} &= \bar{\mathrm{x}}_s + \hat{\mathbf{x}}_{j=2}. \end{aligned} \tag{8}$$

Here, $\text{Conv}_{3\times3}$ represents $3 \times 3$ convolution, $\delta(\cdot)$ denotes a ReLU activation, and $\otimes$ represents elementwise multiplication. $\text{Att}_\text{c}$ and $\text{Att}_\text{s}$ present the channel and spatial attention, respectively, and they are formulated as follows:

$$\begin{aligned} \text{Att}_\text{c}\big(\bar{\mathrm{x}}\big) &= \sigma\big(\text{MLP}\big(\text{AP}\big(\bar{\mathrm{x}}\big)\big) + \text{MLP}\big(\text{MP}\big(\bar{\mathrm{x}}\big)\big)\big) \\ \text{Att}_\text{s}\big(\bar{\mathrm{x}}_c\big) &= \sigma\big(\text{Conv}_{7\times7}\big[\text{AP}\big(\bar{\mathrm{x}}_c\big), \text{MP}\big(\bar{\mathrm{x}}_c\big)\big]\big). \end{aligned} \tag{9}$$

where $\sigma(\cdot)$ denotes the sigmoid function and $\text{Conv}_{7\times7}$ denotes $7\times7$ convolution. $\text{AP}(\cdot)$ and $\text{MP}(\cdot)$ denote the average pooling and max pooling operations, respectively. $[\cdot]$ is concatenation, and $\text{MLP}(\cdot)$ represents a multilayer perceptron.

### *E. Uncertainty Estimator*

The existing methods for raindrop removal apply uniform constraints to raindrop regions, disregarding the inherent variations in shape, position, size, transparency, and distribution of raindrops. This limitation results in suboptimal restoration performance. To tackle this challenge, we aim to develop a discriminative method that effectively distinguishes and treats raindrop regions exhibiting diverse levels of degradation. Motivated by the uncertainty estimation [12], which indicates that regions with greater degradation in an image exhibit higher levels of uncertainty, we construct an uncertainty estimator $\mathbb{U}$ [as shown in Fig. 2(b)] to predict an uncertainty map $\mathcal{S} \in \mathbb{R}^{H\times W\times 3}$ for representing the degradation information of each pixel given a raindrop image $\mathcal{X} \in \mathbb{R}^{H\times W\times 3}$. Specifically, the uncertainty estimator is composed of 3 Conv-ELU blocks. Given the output features $\mathcal{F}^2_{\text{Dec}}$ of the backbone [Fig. 2(a)], the estimation of uncertainty map is formulated as follows:

$$\mathcal{S} = \mathbb{U}\big(\mathcal{F}^2_{\text{Dec}}; \Theta_U\big) + \mathcal{X} \tag{10}$$

where $\Theta_U$ represents the weights of the uncertainty estimator.

Subsequently, a key of this work is to leverage the estimated uncertainty map to constrain the network in generating raindrop-free images. We assume that the Deraindrop image $\hat{\mathcal{Y}}$ follows a Gaussian distribution, and we define a likelihood function as follows:

$$p\big(\mathcal{Y}|\hat{\mathcal{Y}}\big) = \mathcal{N}\big(\hat{\mathcal{Y}}, \sigma^2\big) \tag{11}$$

where the mean of the Gaussian distribution $\mathcal{N}$ is generated raindrop-free image $\hat{\mathcal{Y}}$, $\mathcal{Y}$ represents its corresponding ground truth, and $\sigma^2$ is the observation noise. To model arbitrary uncertainty, [12] introduces pixel-specific observation noise $\sigma_i$. We follow [12] and specify the likelihood function as follows:

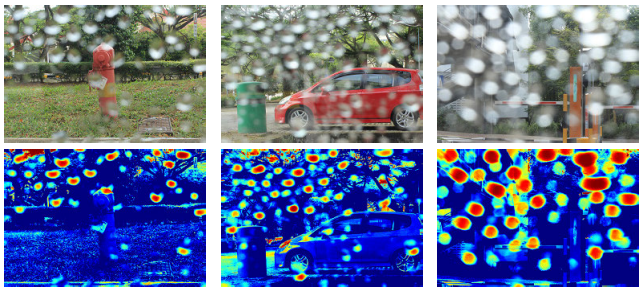


Fig. 5. Raindrop images and their corresponding uncertainty maps. The top row shows raindrop images, and the bottom row shows their uncertainty maps. These uncertainty maps depict the sparsity of raindrop distribution and different degrees of degradation within raindrop regions, i.e., the variations in shape, size, and transparency.

$$p\big(\mathcal{Y}_i|\hat{\mathcal{Y}}_i\big) = \frac{1}{\sqrt{2\pi}\sigma_i}\exp\left(\frac{\big(\mathcal{Y}_i - \hat{\mathcal{Y}}_i\big)^2}{2\sigma_i^2}\right) \tag{12}$$

where $\hat{\mathcal{Y}}_i$ and $\mathcal{Y}_i$ are the predicted value and ground truth for pixel $i$ and the variance $\sigma_i$ of pixel $i$ represents the noise scalar. We take the negative logarithm of the likelihood function as follows:

$$-lnP\big(\mathcal{Y}_i|\hat{\mathcal{Y}}_i\big) = \frac{\big(\mathcal{Y}_i - \hat{\mathcal{Y}}_i\big)^2}{2\sigma_i^2} + \frac{1}{2}ln\sigma_i^2 + ln\sqrt{2\pi}. \tag{13}$$

For numerical stability, we train the network to learn the logarithmic variance $\mathcal{S}_i = ln\sigma_i^2$. We assume that $N$ is the total number of pixels. Finally, we simplify and minimize the negative log-likelihood function to obtain the final uncertainty-driven loss function by

$$\mathcal{L}_U = \frac{1}{2N}\sum_{i=1}^{N}\big(exp(-\mathcal{S}_i)\|\mathcal{Y}_i - \hat{\mathcal{Y}}_i\|^2 + \mathcal{S}_i\big). \tag{14}$$

The first term is the residual term, and the second term represents the regular term.

We argue that uncertainty estimation can facilitate the network to better focus on raindrop degradation regions. Here, we attempt to explain the uncertainty-driven loss for the raindrop removal task. Observing the uncertainty maps in Fig. 5, we find that the more severe the degradation, the higher the uncertainty value. For regions with severe degradation, the residual term in (14) can be ignored, and the loss is mainly controlled by the regular term. Conversely, in regions with a lower uncertainty value, indicating a milder level of degradation, the uncertainty-driven loss is primarily governed by its residual term. Hence, the uncertainty-driven loss can dynamically adjust the penalty imposed by the loss function on the network based on the degree of image degradation.

### *F. Overall Losses*

As shown in Fig. 2, the overall objective function consists of three parts: content loss $\mathcal{L}_c$, uncertainty-driven loss $\mathcal{L}_U$, and perceptual loss $\mathcal{L}_p$. We multiply each loss function by its corresponding weight to obtain the final loss function of the network as follows:

$$\mathcal{L}_\text{over} = \lambda_c\mathcal{L}_c + \lambda_p\mathcal{L}_p + \lambda_U\mathcal{L}_U \tag{15}$$

where weight coefficients $\lambda_c$ and $\lambda_U$ are set to 1.0 and $\lambda_p$ is set to 0.1. Given raindrop removal images $\hat{\mathcal{Y}}_{k=1}^{N}$ and corresponding clean images $\mathcal{Y}_{k=1}^{N}$, the content loss $\mathcal{L}_c$ is calculated by employing the $L$-1 norm, which quantifies the pixelwise disparities between the Deraindrop image $\hat{\mathcal{Y}}_k$ and the ground truth $\mathcal{Y}_k$. It is defined as follows:

$$\mathcal{L}_c = \frac{1}{N}\sum_{k=1}^{N} \|\hat{\mathcal{Y}}_k - \mathcal{Y}_k\|_1. \tag{16}$$

Note that, we use the VGG-19 [49] pretrained by the ImageNet dataset to calculate the perceptual loss at the feature level, which is defined as follows:

$$\mathcal{L}_p = \frac{1}{CHW}\|\Phi_i(\hat{\mathcal{Y}}_k) - \Phi_i(\mathcal{Y}_k)\|_2 \tag{17}$$

where $C$, $H$, and $W$ are the number of channels, height, and width of feature maps. $\Phi_i$ represents the $i$th layer of the VGG-19 model. The uncertainty-driven loss is defined by (14).

## IV. Experiments

### A. Dataset

To demonstrate the effectiveness of the proposed method, we evaluate the raindrop removal results on the Deraindrop dataset [2]. The Deraindrop dataset is obtained by taking photographs before/after spraying water on the glass and contains 1110 image pairs. The dataset consists of three parts: training set (861 image pairs), testA (58 image pairs), and testB (249 image pairs), where testA is a subset of testB. Following the same strategy as [2], we use 861 image pairs for training and testA for testing.

### B. Evaluation Metrics

We utilize reference-based evaluation metrics (SSIM, PSNR, FID, LPIPS, and FSIM) and nonreference metrics (NIQE, PIQE, and BRISQUE) to evaluate the performance of our method on the image raindrop removal task.

*1) Reference-Based Metrics:* We utilize PSNR, SSIM, FID, LPIPS, and FSIM to evaluate the performance of our method. The details of these reference-based metrics are described below.

1) Peak signal-to-noise ratio (PSNR) [50] mainly measures the proximity of the corresponding pixels between the generated image $\mathbf{x}$ and the reference image $\mathbf{y}$. The higher its value, the higher the quality of the generated image. PSNR-L denotes the PSNR on the $Y$ channel in the YCbCr space

$$\text{PSNR}(\mathbf{x},\mathbf{y}) = 10\log_{10}\left[255^2/MSE(\mathbf{x},\mathbf{y})\right]. \tag{18}$$

2) Structural similarity index (SSIM) [51] measures the similarity between the generated image $\mathbf{x}$ and its reference image $\mathbf{y}$ from three aspects of luminance, contrast, and structure, and its value range is [0,1]. Similar to the above PSNR, a higher value indicates a higher quality image

$$\text{SSIM}(\mathbf{x},\mathbf{y}) = \left(\frac{2\mu_{\mathbf{x}}\mu_{\mathbf{y}} + c_1}{\mu_{\mathbf{x}}^2 + \mu_{\mathbf{y}}^2 + c_1}\right)\left(\frac{2\sigma_{\mathbf{xy}} + c_2}{\sigma_{\mathbf{x}}^2 + \sigma_{\mathbf{y}}^2 + c_2}\right) \tag{19}$$

where $\mu_{\mathbf{x}}$ and $\mu_{\mathbf{y}}$ are the mean values of images $\mathbf{x}$ and $\mathbf{y}$. $\sigma_{\mathbf{x}}^2$ and $\sigma_{\mathbf{y}}^2$ are the variances of images $\mathbf{x}$ and $\mathbf{y}$. $c_1$ and $c_2$ are two constant positive values that are used to prevent numerical instability.

3) Fréchet inception distance (FID) [52] aims to measure the difference between generated and real image distributions. A lower FID value indicates that the generated image is closer to the real image distribution.
4) Learned perceptual image patch similarity (LPIPS) [53] is used to measure the perceptual similarity between two images. A lower value indicates a higher quality image.
5) Feature similarity index measure (FSIM) [54] is a variant of SSIM that considers not all pixels in an image have the same importance. The higher the value, the better the image quality.

*2) Nonreference Metrics:* We utilize NIQE, PIQE, and BRISQUE to evaluate the performance of our method. The details of these nonreference metrics are described below.

1) Natural image quality evaluator (NIQE) [55] measures deviations from statistical regularities observed in natural images. The lower the value, the better the image quality.
2) Perceptual-based image quality evaluator (PIQE) [56] performs block-level analysis to assess image distortion, with lower values indicating higher image quality.
3) Blind/referenceless image spatial quality evaluator (BRISQUE) [57] measures image quality by fitting image coefficients to a Gaussian distribution and using SVM for evaluation. The lower the value, the better the image quality.

### C. Implementation Details

The proposed method is implemented on the PyTorch framework. The whole training process takes 300 epochs and requires an NVIDIA RTX2080Ti GPU. We use Adam optimizer [58] ($\beta_1 = 0.5$ and $\beta_2 = 0.999$) to optimize the model with an initial rate of $1\times10^{-4}$. After the first 150 epochs, we linearly decay the rate to zero over the next 150 epochs. The batch size is set to 1, and the patch size is set to 256.

### D. Comparison With State-of-the-Art Methods

*1) Quantitative Comparison:* We quantitatively compare our method with the existing state-of-the-art methods on the above nine evaluation metrics. As shown in Table I, it is clear that our method outperforms other state-of-the-art methods on PSNR-L, PSNR, FSIM, FID, LPIPS, and BRISQUE. It demonstrates that images generated using our method have better restoration performance regarding image perceptual quality and naturalness. In addition, our method ranks second in raindrop removal performance in terms of SSIM and PIQE. Although DuRN [26] achieves the best performance on SSIM, AttenGAN [2] performs best on NIQE, and DA-CLIP [7] demonstrates superior performance in PIQE; they significantly lag behind our method in other metrics.

*2) Qualitative Comparison:* We implement a qualitative comparison of various raindrop removal methods on the Deraindrop dataset [2]. We show two sets of images to

TABLE I

QUANTITATIVE RAINDROP REMOVAL RESULTS. ↑/↓ FOR A METRIC DENOTES THAT A HIGHER/LOWER VALUE IS BETTER. THE BEST RESULTS ARE **BOLDED**, AND THE SECOND BEST RESULTS ARE HIGHLIGHTED IN UNDERLINE

| | Public. | Reference Metric | | | | | | Non-reference Metric | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | PSNR-L↑ | SSIM↑ | PSNR↑ | FSIM↑ | FID↓ | LPIPS↓ | NIQE↓ | PIQE↓ | BRISQUE↓ |
| AttenGAN [2] | CVPR'18 | 31.52 | 0.902 | 30.55 | 0.972 | 31.55 | 0.048 | **8.235** | 8.353 | 13.80 |
| Quan's [3] | ICCV'19 | 31.37 | 0.918 | 30.85 | 0.972 | 30.68 | 0.064 | 9.899 | 12.05 | 20.81 |
| DuRN [26] | CVPR'19 | 31.24 | **0.926** | 30.70 | 0.973 | 27.59 | 0.059 | 8.956 | 7.929 | 16.23 |
| BPP [27] | ICIP'21 | 30.79 | 0.918 | 30.24 | 0.969 | 34.97 | 0.078 | 8.881 | 8.523 | 17.44 |
| TKLMR [28] | CVPR'22 | 30.45 | 0.912 | 29.81 | 0.967 | 34.01 | 0.056 | 9.206 | 7.317 | 13.38 |
| MCW [16] | SPIC'22 | 30.25 | 0.919 | 29.76 | 0.967 | 35.08 | 0.068 | 10.43 | 10.67 | 19.63 |
| DA-CLIP [7] | ICLR'24 | 29.92 | 0.870 | 29.31 | 0.969 | 24.05 | 0.061 | 8.357 | **6.863** | 17.94 |
| Ours | None | **31.62** | 0.924 | **30.96** | **0.976** | **21.85** | **0.042** | 9.563 | 7.020 | **12.05** |

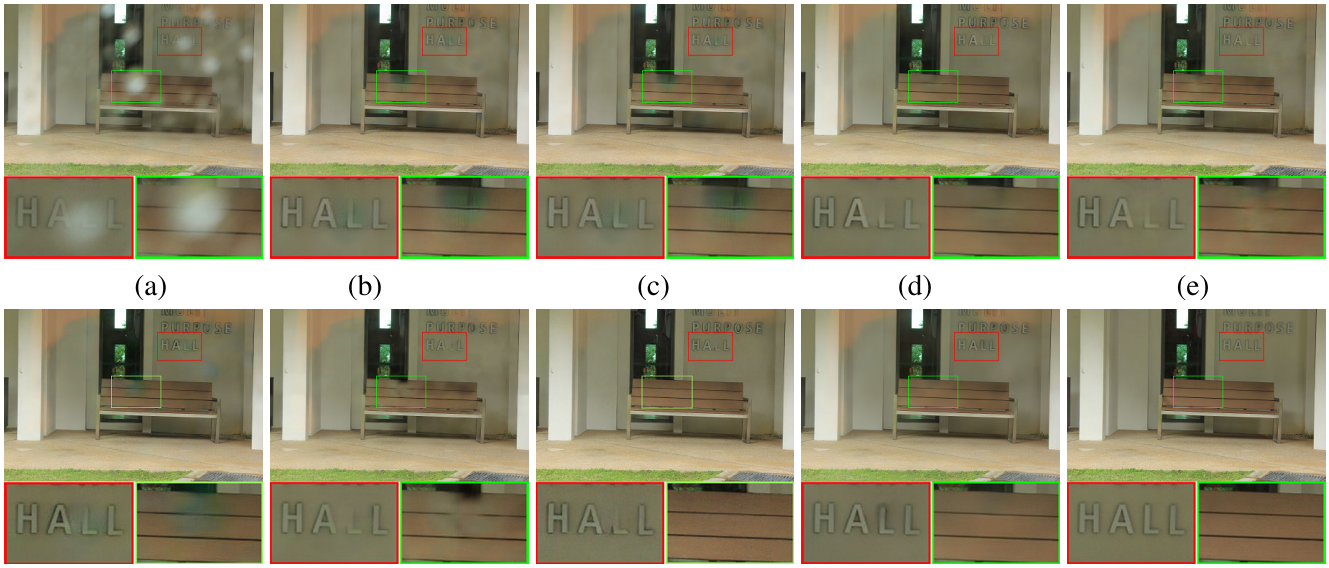



Fig. 6. Qualitative comparisons on the Deraindrop dataset [2]. The raindrop removal results by our method can protect the original edge details and textures in the image to a greater extent while removing raindrops. (a) Input. (b) AttenGAN [2]. (c) Quan's [3]. (d) DuRN [26]. (e) BPP [27]. (f) TKLMR [28]. (g) MCW [16]. (h) DA-CLIP [7]. (i) Ours. (j) GT.

intuitively demonstrate that the USTN has a more effective and robust raindrop removal performance. As shown in Fig. 6, it is clear that our method can protect the original edge details and textures in the image while removing raindrops. In Fig. 7, we demonstrate the advantages of our approach by magnifying the local region of the image. In addition, by visualizing the pixel distribution of a certain column in the generated image, we can more intuitively highlight that our method can achieve better rain removal effects.

### *E. High-Level Task Application*

Deep learning-based methods [59], [60], [61], [62], [63] have demonstrated superior performance in recent years. However, images are usually disturbed by various factors (such as raindrops), so these data-based methods cannot be effectively applied to real-world environments. Considering the negative impact of raindrop occlusion on semantic segmentation, we obtain the semantic segmentation results of generated images based on the Segmenter [64] pretrained on the ADE20K dataset. We propose that the performance of raindrop removal methods can be evaluated from the perspective of semantic segmentation. We perform semantic segmentation on testA of the Deraindrop dataset, as shown in Fig. 8. Compared with the existing state-of-the-art methods, we can intuitively observe that the semantic segmentation results of Fig. 8(i) and (j) are the most similar. Therefore, we conclude that the images generated by our method achieve better semantic segmentation results.

### *F. Efficiency Analysis*

To more comprehensively evaluate the model performance of our method, we analyze the efficiency of our method compared with the existing state-of-the-art methods. Specifically, we measure the model parameters and inference time, comparing them with the existing state-of-the-art methods. As shown in Table II, our method does not achieve the fewest

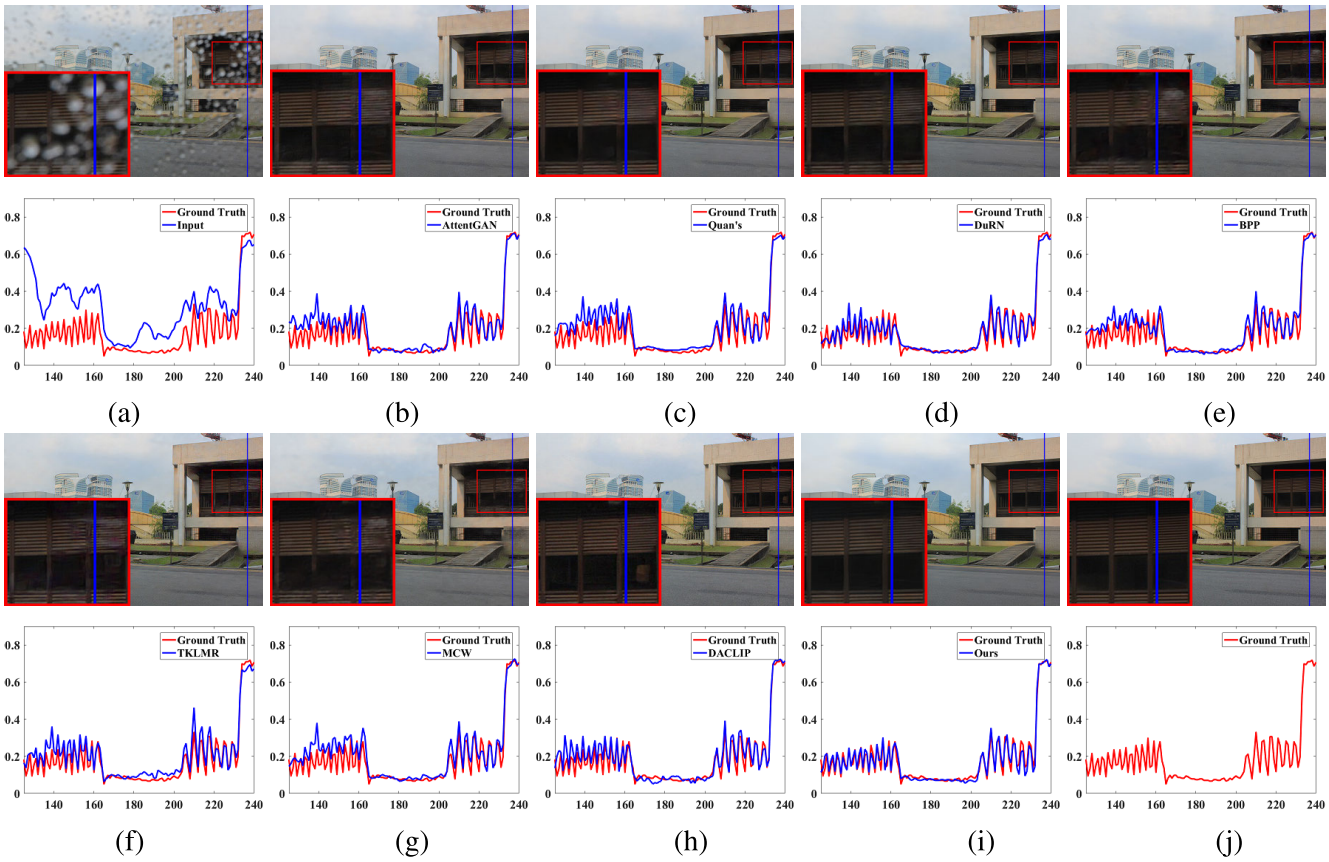



Fig. 7. Qualitative comparisons on the Deraindrop dataset [2]. The raindrop removal images generated by the existing state-of-the-art methods are shown. We visualize the pixel distribution of the generated images in a column marked with blue lines. (a) Input. (b) AttenGAN [2]. (c) Quan's [3]. (d) DuRN [26]. (e) BPP [27]. (f) TKLMR [28]. (g) MCW [16]. (h) DA-CLIP [7]. (i) Ours. (j) GT.

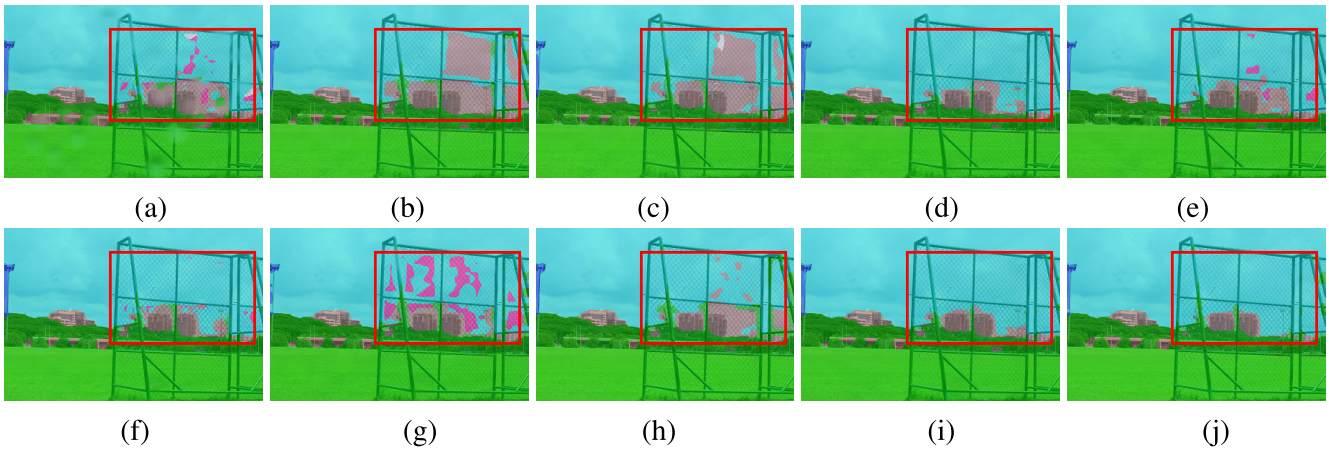
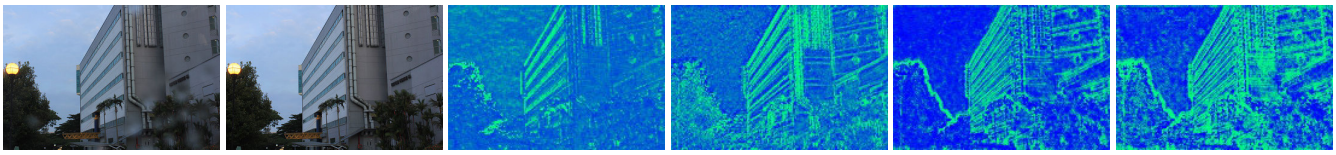



Fig. 8. Semantic segmentation results obtained before and after raindrop removal based on pretrained Segmenter [64]. The raindrop removal results by our method achieve better semantic segmentation results. (a) Input. (b) AttenGAN [2]. (c) Quan's [3]. (d) DuRN [26]. (e) BPP [27]. (f) TKLMR [28]. (g) MCW [16]. (h) DA-CLIP [7]. (i) Ours. (j) GT.

model parameters and least inference time. The main reason is that our network involves Transformer blocks, known to have large model parameters and high computational resource consumption. Nevertheless, our network parameters and inference time are still significantly lower than TKLMR [28] and DA-CLIP [7]. Through comprehensive analysis of the experimental results in Tables I and II, it can be proved that, although our method does not achieve the fewest model parameters and least inference time, our method outperforms the existing state-of-the-art methods in image generation quality and achieves the optimal raindrop removal performance. Overall, our method has advantages in the field of raindrop removal.

### G. Ablation Studies

*1) Effectiveness of PFR:* To evaluate the effectiveness of PFR-1 and PFR-2, we design three model architectures. As shown in "M-1," "M-2," and "M-3" in Table III, when



(a) Input (b) GT (c) w/o PFR-1&2 (d) w/o PFR-1 (e) w/o PFR-2 (f) w/ PFR-1&2

Fig. 9. Effectiveness of PFR module. (a) Input image. (b) Ground truth. (c) Without PFR-1 and PFR-2 modules. (d) Without PFR-1 module. (e) Without PFR-2 module. (f) With PFR-1 and PFR-2 modules. The PFR module effectively models the global and local features of images.

TABLE II

EFFICIENCY ANALYSIS OF THE EXISTING RAINDROP REMOVAL METHODS. THE BEST RESULTS ARE **BOLDED**, AND THE SECOND BEST RESULTS ARE HIGHLIGHTED IN UNDERLINE

| | Public. | Param. (Mb) | Time (s) |
|---|---|---|---|
| AttenGAN [2] | CVPR'18 | 7.27 | 0.085 |
| Quan's [3] | ICCV'19 | 6.24 | 0.136 |
| DuRN [26] | CVPR'19 | 10.18 | **0.047** |
| TKLMR [28] | CVPR'22 | 28.71 | 1.966 |
| MCW [16] | SPIC'22 | **2.16** | 0.235 |
| DA-CLIP [7] | ICLR'24 | 48.98 | 23.48 |
| Ours | None | 19.54 | 0.827 |

TABLE III

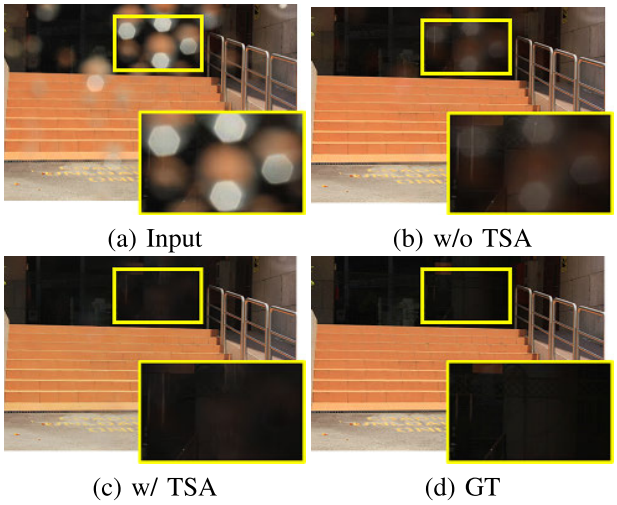
ABLATION STUDY FOR DIFFERENT VARIANTS OF OUR USTN ON THE DERAINDROP DATASET [2]. PFR-1, PFR-2, TSA, AND $\mathcal{L}_U$ DENOTE THE FIRST PFR OF USTN, THE SECOND PFR OF USTN, THE TSA LAYER, AND THE UNCERTAINTY-DRIVEN LOSS, RESPECTIVELY. THE BEST RESULTS ARE **BOLDED**

| | Module | | | | Metric | | | |
|---|---|---|---|---|---|---|---|---|
| | PFR-1 | PFR-2 | TSA | $\mathcal{L}_U$ | PSNR-L↑ | SSIM↑ | PSNR↑ | FSIM↑ |
| M-1 | ✗ | ✗ | ✓ | ✓ | 29.64 | 0.919 | 30.15 | 0.973 |
| M-2 | ✗ | ✓ | ✓ | ✓ | 31.28 | 0.922 | 30.69 | **0.976** |
| M-3 | ✓ | ✗ | ✓ | ✓ | 31.30 | 0.921 | 30.69 | 0.975 |
| M-4 | ✓ | ✓ | ✗ | ✓ | 31.30 | 0.921 | 30.70 | 0.975 |
| M-5 | ✓ | ✓ | ✓ | ✗ | 31.08 | 0.908 | 30.45 | 0.975 |
| Ours | ✓ | ✓ | ✓ | ✓ | **31.62** | **0.924** | **30.96** | **0.976** |

TABLE IV

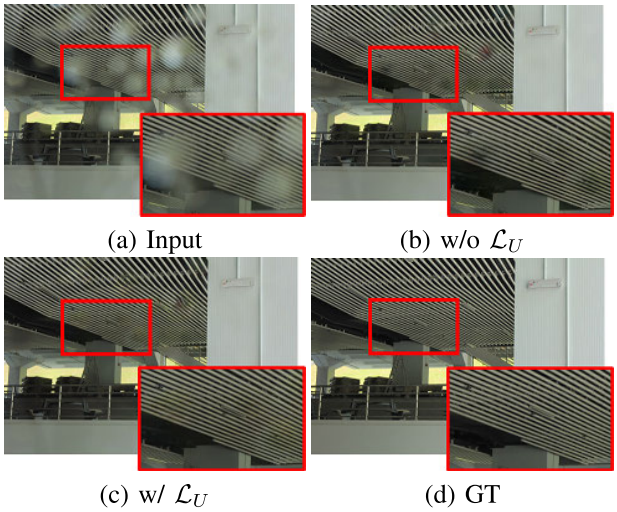
ABLATION STUDY FOR THE NUMBER OF STBS IN OUR METHOD ON THE DERAINDROP DATASET [2]. STB-1, STB-2, AND STB-4 DENOTE STB IN 1, 1/2, AND 1/4 SCALES. THE BEST RESULTS ARE **BOLDED**

| | Module | | | Metric | | | |
|---|---|---|---|---|---|---|---|
| | STB-1 | STB-2 | STB-4 | PSNR-L↑ | SSIM↑ | PSNR↑ | LPIPS↓ |
| M-6 | 1 | 1 | 1 | 31.06 | 0.919 | 30.44 | 0.044 |
| M-7 | 2 | 2 | 2 | 31.34 | 0.921 | 30.73 | **0.040** |
| Ours | 3 | 3 | 3 | **31.62** | **0.924** | **30.96** | 0.042 |



(a) Input (b) w/o TSA (c) w/ TSA (d) GT

Fig. 10. Ablation analysis of TSA in STB. (a) Input image. (b) Without the TSA layer. (c) With the TSA layer. (d) Ground truth. The TSA layer can effectively reduce artifacts and restore high-quality background images.



(a) Input (b) w/o $\mathcal{L}_U$ (c) w/ $\mathcal{L}_U$ (d) GT

Fig. 11. Ablation analysis of uncertainty-driven loss $\mathcal{L}_U$. (a) Input image. (b) Without $\mathcal{L}_U$. (c) With $\mathcal{L}_U$. (d) Ground truth. Using $\mathcal{L}_U$ contributes to effectively removing raindrops and reconstructing clear background structures.

PFR-1 and PFR-2 are removed, PSNR-L and PSNR decrease by 1.98 and 0.81 dB, respectively. When only PFR-1 is removed, PSNR-L and PSNR decrease by 0.34 and 0.27 dB, respectively. When only PFR-2 is removed, PSNR-L decreases by 0.32 dB. As shown in Fig. 9, we visualize the features of the input image before and after being processed by PFR-1 and PFR-2. The comparison between Fig. 9(c) and (d) demonstrates the effective extraction of local and global features by PFR-1, including building edges and leaf textures. A comparison of Fig. 9(e) and (f) reveals that PFR-2 further extracts detailed information from the image. The above experimental results demonstrate that PFR effectively models the global and local features of images.

*2) Effectiveness of the TSA:* We investigate the effectiveness of the TSA by replacing TSA with the self-attention layer in the standard Transformers. Comparing the quantitative results for "M-4" with ours in Table III, we find that the lack of TSA results in the raindrop removal performance of the network decreasing (e.g., PSNR-L decreased

(a) Input (b) M-6 (c) M-7 (d) Ours (e) GT

Fig. 12. Effectiveness of the number of STBs. (a) Input image. (b) "M-6" represents that the number of STB-1/2/4 is 1. (c) "M-7" represents that the number of STB-1/2/4 is 2. (d) Number of STB-1/2/4 is 3. (e) Ground truth. With the number of STBs increasing, the raindrop removal performance of USTN consistently improves.

by 0.32 dB). Comparing (b) and (c) in Fig. 10, when using the self-attention layer in the standard Transformers instead of TSA, we observe a significant decrease in the capability to restore the area heavily damaged by raindrops. The experimental results above demonstrate that TSA is better suited for modeling raindrop images due to their sparsity, and it significantly improves the raindrop removal capabilities of the network.

*3) Effectiveness of Uncertainty-Driven Loss:* To investigate the impact of uncertainty-driven loss, we exclude it and continue to utilize the Deraindrop dataset [2] for network training. Upon comparing the quantitative results for "M-5" with ours in Table III, we observe significant performance enhancements attributable to the uncertainty-driven loss in our network. Specifically, we note a 0.54-dB improvement in PSNR-L and a 0.51-dB improvement in PSNR. As depicted in Fig. 11(b), the absence of uncertainty-driven loss results in a diminished restoration effect on the edges and textures. This observation implicitly validates the adaptive raindrop removal capability facilitated by the uncertainty-driven loss.

*4) Effect of the Number of STBs:* To investigate the influence of the number of STBs on raindrop removal, we vary the number of STBs across different scales. As presented in Table IV, we implement "M-6" and "M-7" and compare them with our approach. The table data demonstrate that the PSNR-L, SSIM, and PSNR increase as the number of STBs increases. For instance, "M-7" exhibits a 0.28-dB improvement in PSNR-L compared with "M-6," while our method shows a 0.28-dB improvement in PSNR-L compared with "M-7." The qualitative comparison results are depicted in Fig. 12(b)–(d). Notably, in (b), with the network structure being "M-6," the restoration effect of heavy rain regions in the input image is notably poor, and raindrops persist in the generated image. (c) demonstrates that with the network structure being "M-7," the remaining raindrops in the generated image are substantially reduced, and (d) represents the generated image obtained using our final network structure. Fig. 12 clearly illustrates that as the number of STBs increases, the restoration effect of the areas affected by raindrops consistently improves.

## V. Conclusion

This article presents an uncertainty-aware sparse Transformer network called USTN for image Deraindrop. During training, USTN utilizes uncertainty estimation to adaptively adjust the degree of penalty to the network according to the degree of raindrop degradation, improving the restoration quality in uncertain areas. In addition, we introduce sparse Transformer blocks into USTN. This module replaces the self-attention layer in the standard Transformers with the TSA layer, which can retain the most useful attention values for modeling global features. The experimental results illustrate that the proposed method outperforms other state-of-the-art methods by effectively preserving original image features while removing raindrops. This characteristic enhances its applicability in real-world scenarios.

## References


[1] X. Yan and Y. R. Loke, "RainGAN: Unsupervised raindrop removal via decomposition and composition," in *Proc. IEEE/CVF Winter Conf. Appl. Comput. Vis. Workshops (WACVW)*, Jan. 2022, pp. 14–23.

[2] R. Qian, R. T. Tan, W. Yang, J. Su, and J. Liu, "Attentive generative adversarial network for raindrop removal from a single image," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 2482–2491.

[3] Y. Quan, S. Deng, Y. Chen, and H. Ji, "Deep learning for seeing through window with raindrops," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 2463–2471.

[4] D. Eigen, D. Krishnan, and R. Fergus, "Restoring an image taken through a window covered with dirt or rain," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2013, pp. 633–640.

[5] A. Vaswani et al., "Attention is all you need," in *Proc. Adv. Neural Inform. Process. Syst. (NIPS)*, 2017, pp. 5998–6008.

[6] A. Dosovitskiy et al., "An image is worth 16×16 words: Transformers for image recognition at scale," in *Proc. ICLR*, 2021, pp. 1–12.

[7] Z. Luo, F. K. Gustafsson, Z. Zhao, J. Sjölund, and T. B. Schön, "Controlling vision-language models for multi-task image restoration," 2023, *arXiv:2310.01018*.

[8] Z. Liu et al., "Swin transformer: Hierarchical vision transformer using shifted windows," in *Proc. ICCV*, 2021, pp. 9992–10002.

[9] X. Chen, H. Li, M. Li, and J. Pan, "Learning a sparse transformer network for effective image deraining," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2023, pp. 5896–5905.

[10] S. Chen, T. Ye, J. Bai, E. Chen, J. Shi, and L. Zhu, "Sparse sampling transformer with uncertainty-driven ranking for unified removal of raindrops and rain streaks," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2023, pp. 13106–13117.

[11] Y. Jin, W. Yan, W. Yang, and R. T. Tan, "Structure representation network and uncertainty feedback learning for dense non-uniform fog removal," in *Proc. ACCV*, vol. 13843, 2022, pp. 155–172.

[12] A. Kendall and Y. Gal, "What uncertainties do we need in Bayesian deep learning for computer vision?" in *Proc. NeurIPS*, vol. 30, 2017, pp. 5574–5584.

[13] Q.-L. Zhang and Y.-B. Yang, "SA-Net: Shuffle attention for deep convolutional neural networks," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Jun. 2021, pp. 2235–2239.

[14] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *Proc. CVPR*, 2018, pp. 7132–7141.

[15] S. Woo, J. Park, J.-Y. Lee, and I. S. Kweon, "CBAM: Convolutional block attention module," in *Proc. Eur. Conf. Comput. Vis.*, Sep. 2018, pp. 3–19.

[16] Y. Park, M. Jeon, J. Lee, and M. Kang, "MCW-Net: Single image deraining with multi-level connections and wide regional non-local blocks," *Signal Process., Image Commun.*, vol. 105, Jul. 2022, Art. no. 116701.

[17] Y. Li, Y. Monno, and M. Okutomi, "Dual-pixel raindrop removal," in *Proc. BMVC*, 2022, p. 439.

[18] D. M. Nguyen, T. P. Le, D. M. Vo, and S.-W. Lee, "UnfairGAN: An enhanced generative adversarial network for raindrop removal from a single image," *Expert Syst. Appl.*, vol. 210, Dec. 2022, Art. no. 118232.

[19] Z. Jiang, S. Yang, J. Liu, X. Fan, and R. Liu, "Multiscale synergism ensemble progressive and contrastive investigation for image restoration," *IEEE Trans. Instrum. Meas.*, vol. 73, pp. 1–14, 2024.

[20] W. Tao, X. Yan, Y. Wang, and M. Wei, "MFFDNet: Single image deraining via dual-channel mixed feature fusion," *IEEE Trans. Instrum. Meas.*, vol. 73, pp. 1–13, 2024.

[21] X. Jiao, Y. Liu, J. Gao, X. Chu, X. Fan, and R. Liu, "PEARL: Preprocessing enhanced adversarial robust learning of image deraining for semantic segmentation," in *Proc. 31st ACM Int. Conf. Multimedia*, vol. 35, Oct. 2023, pp. 8185–8194.

[22] D. Wang, H. Tang, J. Pan, and J. Tang, "Learning a tree-structured channel-wise refinement network for efficient image deraining," in *Proc. IEEE Int. Conf. Multimedia Expo (ICME)*, Jul. 2021, pp. 1–6.

[23] D. Wang, J.-S. Pan, and J.-H. Tang, "Single image deraining using residual channel attention networks," *J. Comput. Sci. Technol.*, vol. 38, no. 2, pp. 439–454, Apr. 2023.

[24] Z. Tu et al., "MAXIM: Multi-axis MLP for image processing," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 5769–5780.

[25] C.-M. Fan, T.-J. Liu, and K.-H. Liu, "Compound multi-branch feature fusion for image deraindrop," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Oct. 2023, pp. 3399–3403.

[26] X. Liu, M. Suganuma, Z. Sun, and T. Okatani, "Dual residual networks leveraging the potential of paired operations for image restoration," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 7000–7009.

[27] P. N. Michelini, H. Liu, Y. Lu, and X. Jiang, "Back—Projection pipeline," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Sep. 2021, pp. 1949–1953.

[28] W.-T. Chen, Z.-K. Huang, C.-C. Tsai, H.-H. Yang, J.-J. Ding, and S.-Y. Kuo, "Learning multiple adverse weather removal via two-stage knowledge learning and multi-contrastive regularization: Toward a unified model," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 17632–17641.

[29] J. M. J. Valanarasu, R. Yasarla, and V. M. Patel, "TransWeather: Transformer-based restoration of images degraded by adverse weather conditions," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 2353–2363.

[30] T. Wang et al., "GridFormer: Residual dense transformer with grid structure for image restoration in adverse weather conditions," 2023, *arXiv:2305.17863*.

[31] G. Wang, C. Sun, and A. Sowmya, "ERL-Net: Entangled representation learning for single image de-raining," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 5643–5651.

[32] J. Xiao, X. Fu, A. Liu, F. Wu, and Z.-J. Zha, "Image de-raining transformer," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 11, pp. 12978–12995, Nov. 2023.

[33] Q. Guo et al., "EfficientDeRain: Learning pixel-wise dilation filtering for high-efficiency single-image deraining," in *Proc. AAAI Conf. Artif. Intell.*, May 2021, pp. 1487–1495.

[34] C. Wang, J. Pan, and X.-M. Wu, "Online-updated high-order collaborative networks for single image deraining," in *Proc. AAAI*, vol. 36, no. 2, 2022, pp. 2406–2413.

[35] C. Wang, Y. Wu, Z. Su, and J. Chen, "Joint self-attention and scale-aggregation for self-calibrated deraining network," in *Proc. 28th ACM Int. Conf. Multimedia*, Oct. 2020, pp. 2517–2525.

[36] C. Wang, X. Xing, Y. Wu, Z. Su, and J. Chen, "DCSFN: Deep cross-scale fusion network for single image rain removal," in *Proc. 28th ACM Int. Conf. Multimedia*, Oct. 2020, pp. 1643–1651.

[37] R. Quan, X. Yu, Y. Liang, and Y. Yang, "Removing raindrops and rain streaks in one go," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 9143–9152.

[38] K. Zhang, D. Li, W. Luo, and W. Ren, "Dual attention-in-attention model for joint rain streak and raindrop removal," *IEEE Trans. Image Process.*, vol. 30, pp. 7608–7619, 2021.

[39] Q. Luo et al., "Waterdrop removal from hot-rolled steel strip surfaces based on progressive recurrent generative adversarial networks," *IEEE Trans. Instrum. Meas.*, vol. 70, pp. 1–11, 2021.

[40] Q. Luo, H. He, K. Liu, C. Yang, O. Silvén, and L. Liu, "Rain-like layer removal from hot-rolled steel strip based on attentive dual residual generative adversarial network," *IEEE Trans. Instrum. Meas.*, vol. 72, pp. 1–15, 2023.

[41] C. Blundell, J. Cornebise, K. Kavukcuoglu, and D. Wierstra, "Weight uncertainty in neural network," in *Proc. Int. Conf. Mach. Learn.*, 2015, pp. 1613–1622.

[42] Y. Gal and Z. Ghahramani, "Dropout as a Bayesian approximation: Representing model uncertainty in deep learning," in *Proc. 33rd Int. Conf. Mach. Learn.*, vol. 48, Jun. 2016, pp. 1050–1059.

[43] X. Cao and K. Peng, "Stochastic uncertain degradation modeling and remaining useful life prediction considering aleatory and epistemic uncertainty," *IEEE Trans. Instrum. Meas.*, vol. 72, pp. 1–12, 2023.

[44] Q. Ning, W. Dong, X. Li, J. Wu, and G. Shi, "Uncertainty-driven loss for single image super-resolution," in *Proc. NeurIPS*, vol. 34, 2021, pp. 16398–16409.

[45] J. Chang, Z. Lan, C. Cheng, and Y. Wei, "Data uncertainty learning in face recognition," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 5709–5718.

[46] Y. Gu, Z. Jin, and S. C. Chiu, "Active learning combining uncertainty and diversity for multi-class image classification," *IET Comput. Vis.*, vol. 9, no. 3, pp. 400–407, Jun. 2015.

[47] Z. Wang, X. Cun, J. Bao, W. Zhou, J. Liu, and H. Li, "Uformer: A general U-shaped transformer for image restoration," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 17662–17672.

[48] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *Proc. Int. Conf. Mach. Learn.*, 2015, pp. 448–456.

[49] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *Proc. ICLR*, 2015, pp. 1–10.

[50] A. Tanchenko, "Visual-PSNR measure of image quality," *J. Vis. Commun. Image Represent.*, vol. 25, no. 5, pp. 874–878, Jul. 2014.

[51] W. Zhou, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: From error visibility to structural similarity," *IEEE Trans. Image Process.*, vol. 13, no. 4, pp. 600–612, Apr. 2004.

[52] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, and S. Hochreiter, "Gans trained by a two time-scale update rule converge to a local Nash equilibrium," in *Proc. NIPS*, 2017, pp. 6626–6637.

[53] R. Zhang, P. Isola, A. A. Efros, E. Shechtman, and O. Wang, "The unreasonable effectiveness of deep features as a perceptual metric," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 586–595.

[54] 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, Aug. 2011.

[55] 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, Mar. 2013.

[56] N. Venkatanath, D. Venkatanath, M. Chandrasekhar, S. S. Channappayya, and S. S. Medasani, "Blind image quality evaluation using perception based features," in *Proc. 21st Nat. Conf. Commun. (NCC)*, Feb. 2015, pp. 1–6.

[57] A. Mittal, A. K. Moorthy, and A. C. Bovik, "No-reference image quality assessment in the spatial domain," *IEEE Trans. Image Process.*, vol. 21, no. 12, pp. 4695–4708, Dec. 2012.

[58] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proc. ICLR*, 2014, pp. 1–14.

[59] D. Wang, J. Liu, L. Ma, R. Liu, and X. Fan, "Improving misaligned multi-modality image fusion with one-stage progressive dense registration," *IEEE Trans. Circuits Syst. Video Technol.*, early access, Jun. 11, 2024, doi: 10.1109/TCSVT.2024.3412743.

[60] D. Wang, J. Liu, X. Fan, and R. Liu, "Unsupervised misaligned infrared and visible image fusion via cross-modality image generation and registration," in *Proc. Int. Joint Conf. Artif. Intell.*, Jul. 2022, pp. 3508–3515.

[61] D. Wang, J. Liu, R. Liu, and X. Fan, "An interactively reinforced paradigm for joint infrared-visible image fusion and saliency object detection," *Inf. Fusion*, vol. 98, Oct. 2023, Art. no. 101828.

[62] J. Liu, J. Shang, R. Liu, and X. Fan, "Attention-guided global-local adversarial learning for detail-preserving multi-exposure image fusion," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 32, no. 8, pp. 5026–5040, Aug. 2022.

[63] J. Liu, R. Lin, G. Wu, R. Liu, Z. Luo, and X. Fan, "CoCoNet: Coupled contrastive learning network with multi-level feature ensemble for multi-modality image fusion," *Int. J. Comput. Vis.*, vol. 132, no. 5, pp. 1748–1775, May 2024.

[64] R. Strudel, R. Garcia, I. Laptev, and C. Schmid, "Segmenter: Transformer for semantic segmentation," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 7262–7272.

**Bo Fu** received the B.S. and M.S. degrees in computer science and technology from Liaoning Normal University, Dalian, China, in 2006 and 2009, respectively, and the Ph.D. degree in computer application technology from Jilin University, Changchun, China, in 2013.

He is currently an Associate Professor with the School of Computer and Artificial Intelligence, Liaoning Normal University. His research interests include image processing and computer vision.

**Yunyun Jiang** received the B.S. degree in digital media technology from Liaoning Normal University, Dalian, China, in 2023. She is currently pursuing the master's degree with the School of Computer Science and Engineering, Northeastern University, Shenyang, China.

Her research interests include computer vision and deep learning.

**Di Wang** received the M.S. degree in computer science and engineering from Nanjing University of Science and Technology, Nanjing, China, in 2021. She is currently pursuing the Ph.D. degree in software engineering with Dalian University of Technology (DUT), Dalian, China.

Her research interests include computer vision, image fusion, and deep learning.

**Jiaxin Gao** received the B.S. degree in applied mathematics from Dalian University of Technology (DUT), Dalian, China, in 2018, where she is currently pursuing the Ph.D. degree in software engineering.

She is with the Key Laboratory for Ubiquitous Network and Service Software of Liaoning Province, DUT. Her research interests include computer vision, machine learning, and optimization.

**Cong Wang** received the bachelor's degree in mathematics and applied mathematics from Inner Mongolia University, Hohhot, China, in 2017, and the master's degree in computational mathematics from Dalian University of Technology (DUT), Dalian, China, in 2020. He is currently pursuing the Ph.D. degree with the Department of Computing, The Hong Kong Polytechnic University, Hong Kong.

His research interests include computer vision and deep learning.

**Ximing Li** received the Ph.D. degree from Jilin University, Changchun, China, in 2015.

He is currently a Professor with Jilin University. His research interests include machine learning and natural language processing.