SFformer: Adaptive Sparse and Frequency-Guided Transformer Network for Single Image Derain

Xinrui Wang, Hongyun Zhang(✉), Kecan Cai, Duoqian Miao, Qi Zhang, and Miao Li

Tongji University, Cao'an Road, 4800 Shanghai, China
zhanghongyun@tongji.edu.cn

Abstract. Recently transformer models have become prominent models for single image deraining (SID) task. However, these models often fail to utilize frequency knowledge and appropriate self-attention mechanisms effectively, leading to inadequate extraction of rain features and persistent artifacts. To alleviate this problem, we propose Adaptive **S**parse and **F**requency-Guided Trans**former** Network (**SFformer**) for single image derain. Specifically, we propose Adaptive Sparse Attention (ASA) module to selectively pay attention to the most useful channels for better feature aggregation. In addition, considering that rain streaks mainly correspond to the high frequency components in the image, we introduce Frequency-Guided Feedforward (FGF) module to focus on rain streaks. Integrating these proposed modules into a UNet backbone, extensive experimental results on commonly used benchmarks show that the proposed method outperforms current state-of-the-art method. The source code of our work is available at https://github.com/HuluBaba/ECEDerain.

Keywords: Single Image Derain · Sparsity · Frequency

1 Introduction

As a meaningful research topic in the field of low-level vision, single image derain plays an important role in various high-level vision tasks such as image classification, object detection, and video surveillance. Generally speaking, a rainy image can be regarded as being formed by the composition of a clean background layer

This work was supported by the National Natural Science Foundation of China under Grants 62076182, 62376198, 62376199 and 62076184. The National Key Research and Development Program of China under Grants 2022YFB3104770.

Supplementary Information The online version contains supplementary material available at https://doi.org/10.1007/978-981-97-8685-5_34.

Z. Lin et al. (Eds.): PRCV 2024, LNCS 15038, pp. 482–496, 2025.
https://doi.org/10.1007/978-981-97-8685-5_34

and a rain layer. SID aims at removing rain streaks from rainy images, and get clean background. To handle this ill-posed problem, early rain removal methods were usually based on various manually designed priors [2,9,14,15]. However, they usually require strong image priors and are susceptible to complex and changeable scenarios. To learn generalizable priors from large-scale data, convolutional neural networks (CNN) [19] emerged. With the aid of CNN, several rain removal methods have been proposed and achieved better performance [7,13,33]. Nevertheless, limited by limited receptive field and fixed kernel weights, CNN cannot handle long-range rain disturbances and changeable rainy scenes well. To deal with the CNN's shortcomings, The Transformer [20] model is introduced into field of SID to dynamically capture long-range pixel dependencies [3,22,31]. Relying on multi-head self-attention mechanism which can calculate weights specially adapted to the input and achieve effective long-range interaction between pixels, transformer-based SID methods [3,22] have shown superior performance.

In recent years, more methods have been developed to address various challenges in SID. In order to solve the difficulty in processing complex rainfall scenes, methods based on frequency prior knowledge was proposed [7,11,12]. However, [7] and [12] extract rain only from High Frequency Part (HFP), actually ignoring the minor rain present in the Low Frequency Part (LFP), while [11] is too slow because of the explicit use of Fourier transform. To overcome the lack of real rainy image-background pairs, GAN [10] was introduced and achieve realistic visual effects [12,25,34]. The SID approach utilizing CycleGAN further addressed inconsistency problem between the generated image and the original background [5,24,29]. Addressing the difficulty of processing real-world rain streaks, semi-supervised SID methods were developed [23,28]. To mitigate halo artifacts in rain removal outcomes, [1] implemented a closed-loop feedback mechanism to enhance output quality. Additionally, to adapt Transformer for high-resolution images, [31] substituted spatial attention in traditional ViT with channel attention. Furtherly, [4] dealt with noise problem in SID by sparsifying the attention module. However, its fixed percentage mask used to sparse the attention matrix could cause same retention rate across all attention heads. This can hinder the coordination among attention heads by preventing any specific head from focusing more on either local details or global context.

While current SID methods yield impressive results, they usually suffer seriously from artifacts in the results, which compromise the quality of their outputs, as shown in Fig. 1. Unfortunately, current methods to solve artifacts problem either lack adaptability or fail to integrate with Transformer. We conducted in-depth analysis and exploration of the artifact problem, and believe that the root causes behind the artifact problem lie in: (1) In classic self-attention module, each key-query pair gives an attention value, which will participate in feature aggregation. However, not all keys are related to all queries while the attention value is not 0 but only relatively small most of the time. These small values, which usually do not contain useful information, spread noise into all channels through pairwise connections in feature aggregation. (2) Traditional feedforward layer consists of two fully connected layers separated by an activation function,

without considering the frequency characteristics of rain. This is inefficient since the frequency prior knowledge of rain is not applied, and paying equal attention to all areas can easily lead to insufficient rain extraction, especially in rain-rich areas.

In this paper, we propose a comprehensive approach to address these challenges, with two key components: the Adaptive Sparse Attention (ASA) module and the Frequency-Guided Feedforward (FGF) module. The ASA module aims to selectively aggregate the most useful features, while the FGF module directs the model's focus towards rain.

Firstly, we considered how to obtain higher quality aggregated features in the attention layer. Taking inspiration from DRSformer [4], we focus on the most useful features for aggregation by sparsing the dense connection in self-attention. Specifically, we mask the attention matrix with an adaptive threshold and adjust the smaller attention value to 0. Notably, we use a sub-network to calculate a most appropriate threshold for each attention head, and use soft threshold function to ensure smoother output. With the above designs, ASA module can dynamically mask irrelevant features and selectively aggregate useful features, thus enhancing the quality of the extracted features.

In addition, we also design frequency-guided feedforward module to replace the feedforward layer in vanilla Transformer. Recognizing a classic prior that rain streaks almost entirely correspond the HFP in the image while LFP contains background more, we emphasized the HFP in our model. Therefore, our FGF employs a gating mechanism that enhances the focus according to HFP. In this gating mechanism, HFP spatial distribution features are first calculated in the side branch as the gating value. The features from the classical feedforward layer are then multiplied element-wisely by the gated value. Above mechanism helps model to focus on areas with more HFP, which means higher rain density according to the prior knowledge. Therefore, the FGF module allows subsequent layers specifically pay more attention to rain-rich areas, and leads to more thorough rain removal results.

Fig. 1. Artifacts in other SID methods

The main contributions are summarized as follows:

– We propose an ASA module that choose the most useful attention values with an adaptive threshold. Selectively aggregate features, it can prevent noise from spreading through dense connections and enhance the quality of the extracted features.

– We design a novel FGF module that leverages frequency information to direct the model's focus towards areas with higher HFP. Utilizing the prior knowledge, it makes model pay more attention to rain-rich areas.
– Integrating ASA and FGF module into a U-shaped transformer backbone, our SFformer achieves SOTA results on various synthetic and real-world datasets, outperforming existing methods in both quantitative and qualitative evaluations.

2 Method

2.1 Overall Pipeline

Our proposed SFformer aims to minimize artifacts in rain removal results. Overall, the SFformer utilizes a UNet architecture and the main structure is illustrated in Fig. 2. The model consists of a set of encoders and decoders in different hierarchy, and includes projection modules at both the input and output stages.

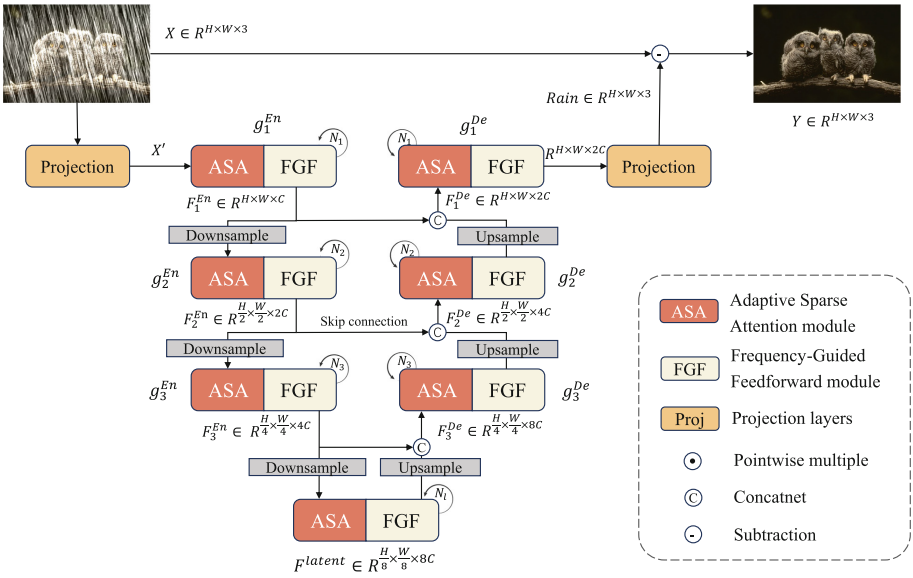

Fig. 2. Model Framework

Given a degraded image $X \in \mathbb{R}^{H\times W\times 3}$, projection module first applies several 3×3 convolution to obtain low-level visual feature embeddings $X' \in \mathbb{R}^{H\times W\times C}$, where H and W is the size of primary image and C is the channel after projection.

Then the set of symmetrical encoders and decoders will be applied on the visual embedding. Level-n encoder or decoder is composed of N_n our proposed ASA modules and FGF modules.

The encoding side expands channel capacity while reduces spatial size, because we need features to aggregate more global information as the hierarchy becomes deeper. For each level in encoding side, Feature extraction will be performed utilizing the level-n encoder g_n^{En}, on the previous layer features F_{n-1}^{En} after downsample, as shown in Eq. 1.

$$F_n^{En} = g_n^{En}\left(\text{ Downsample}(F_{n-1}^{En}) \mid \theta_n^{En}\right) \tag{1}$$

where θ_n^{En} is the parameter in level-n encoder g_n^{En}.

$$F_{n-1}^{De} = \text{concat}\left(F_{n-1}^{En}, \text{Upsample}\left(g_n^{De}(F_n^{De} \mid \theta_n^{De})\right)\right) \tag{2}$$

After passing the hierarchical encoder, latent features $F^{latent} \in \mathbb{R}^{\frac{H}{8} \times \frac{W}{8} \times 8C}$, containing abstract semantic information, are extracted from the high-resolution input. Correspondingly, with the guidance of F^{latent}, the decoding side restore detailed visual features hierarchically as shown in Eq. 2. Notably, θ_n^{De} represents the parameters in the level-n encoder. It is worth noting that before feeding into the level-n decoder, a skip connection transfers features from encoder in corresponding level to concatenate with input, which carries high-resolution detailed information.

Following the decoding process, several convolution layers are applied to the output feature F_1^{De} of the last decoder g_1^{De} to generate the rain image $Rain$, which is then subtracted from the degraded image, as shown in Eq. 3.

$$Y = X - Rain \tag{3}$$

2.2 Adaptive Sparse Attention

In classical channel attention modules, connections between each pair of channels lead to the transmission and accumulation of noise through dense computations. In order to alleviate this problem, our key point is to sparsify the attention matrix using a dynamically generated threshold, as shown in Fig. 3.

Given an input vector $Input \in \mathbb{R}^{H \times W \times C}$, depthwise separable convolution is first applied after layer normalization to aggregate local context across both channel and spatial dimensions. $Q', K', V' \in \mathbb{R}^{T \times \frac{C}{T} \times HW}$ are obtained after depthwise separable convolution and reshaping, where T is the number of attention head. The cross-covariance across channels $A = Q'K'^{\top} \in \mathbb{R}^{T \times \frac{C}{T} \times \frac{C}{T}}$, represents the weights for feature aggregation. Instead of directly using the attention matrix A for aggregating V' as in traditional channel attention, we first apply a soft threshold to sparsify A. This method efficiently filters out irrelevant information while retaining crucial features for aggregation. The threshold τ is set by a simple sub-net, and can vary across different attention heads to determine the most effective threshold for each. After sparsification and softmax computing, the obtained attention matrix A' is then utilized to aggregate V', as shown in Eq. 4.

$$Output = \text{PointwiseConv}(\text{Reshape}(A'V')) + Input \tag{4}$$

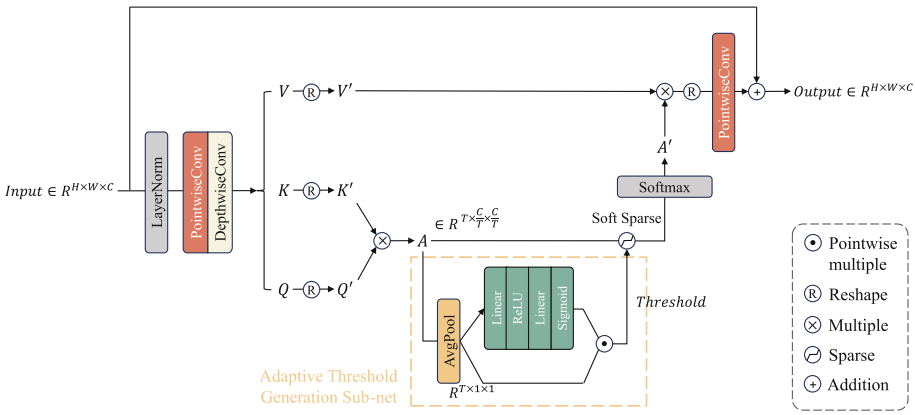

Fig. 3. Structure of ASA module.

Adaptive Threshold Generation Sub-Net. To generate appropriate thresholds, a simple adaptive threshold generation sub-net is used to estimate the thresholds. We want the threshold of each attention head be expressed in the following form:

$$Threshold = \mathrm{Avg}_{\mathrm{head}}(A) \odot \mathrm{MLP}(\mathrm{Avg}_{\mathrm{head}}(A)) \in \mathbb{R}^{T\times 1\times 1} \tag{5}$$

where $\mathrm{Avg}_{\mathrm{head}}(A)$ denotes the average value across a specific attention head. Each attention head is associated with a scalar threshold value stored in $Threshold$. This threshold value is calculated based on the mean attention value of this attention head, multiplied by a coefficient. We use the a simple MLP to compute this coefficient rather than the threshold, to enable the network to focus on the relative value in attention head, rather than roughly discarding or retaining the entire attention head. Specifically, the MLP employ a bottleneck structure, in which the dimension reduce firstly and expand then, to mitigate the risk of overfitting. By generating coefficients according to the mean values across all attention heads, this sub-net allows each head to optimize collaboratively and maintain its unique focal points.

Soft Threshold Sparse. Figure 4 illustrates the soft threshold function and its derivatives. For inputs with an absolute value smaller than the threshold τ, the corresponding output is zero. The output increase uniformly from zero when input exceeds the threshold τ. It is worth noting that derivatives $\frac{\partial y}{\partial x}$ for inputs near zero is zero. This property indicates that minor noise signals are prevented from propagating through the network during the backward process. In addition, unlike ordinary thresholds that jump directly from 0 to 1, you can observe that the transition of the output in this module is gradual rather than abrupt. This smooth change in output enhances the module's robustness while sparsifying the attention.

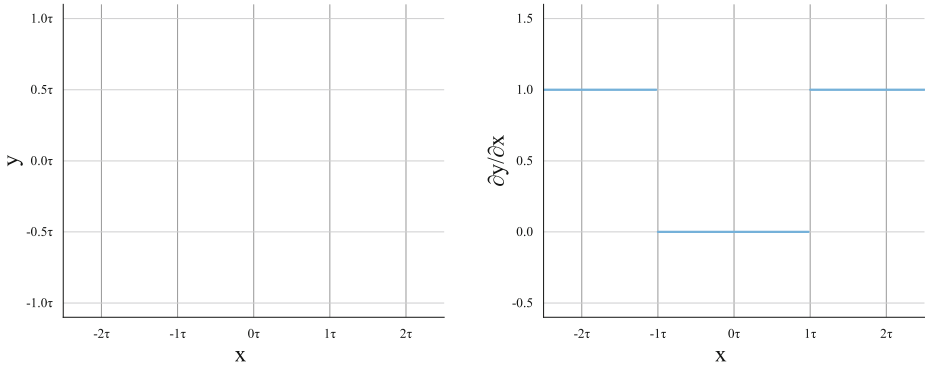

Fig. 4. Soft threshold function and its derivatives

With the above-mentioned designs, ASA module offers following three advantages: (1) It dynamically adapts its threshold based on the input, making the module more adaptable. (2) It smoothens the sparsification process for enhanced model robustness. (3) It concentrates on the most useful features when aggregating features, without being interfered by much useless features.

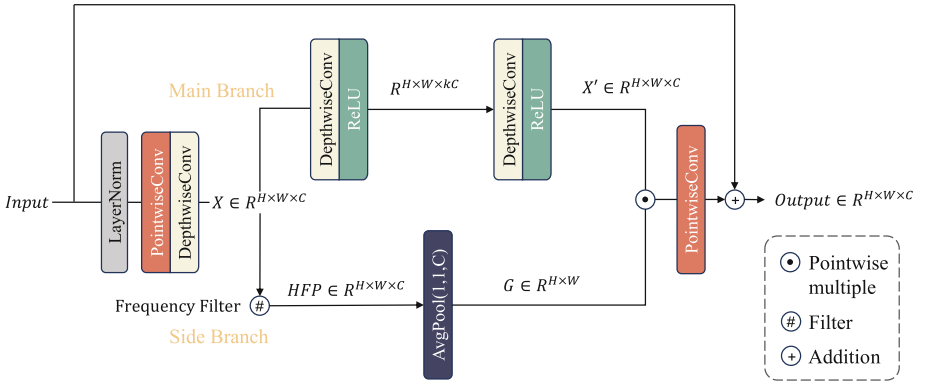

Fig. 5. Structure of FGF module.

2.3 Frequency-Guided Feedforward

To apply the frequency prior knowledge appropriately and efficiently, we propose the FGF module. Given $Input \in \mathbb{R}^{H\times W\times C}$, a depthwise separable convolution is utilized first after layer normalization to combine channel and spatial context. Following prior research [6], we employ a specific convolution kernel as fast low-pass filter to separate LFP and HFP. Then the HFP is utilized to calculate gating

values in side branch. And spatial context aggregation is performed using 5*5 depthwise convolution for larger receptive field in the main branch, as in Eq. 6.

$$X' = f(f(X)),\ f : \mathrm{ReLU}(\mathrm{DepthwiseConv}(\cdot\,;\theta)) \rightarrow \mathbb{R}^{H\times W\times ch} \tag{6}$$

where $ch = kC > C$ for the first layer while $ch = C$ for the second layer.

At the same time, the side branch will estimate the gating value. We can obtain $G \in \mathbb{R}^{H\times W}$ utilizing positional averaging on HFP.

$$G = \mathrm{Average}(HFP, \mathrm{dim} = 3) \tag{7}$$

According to the priori, the variable G, which reflects the local HFP strength, can be interpreted as an indicator of the spatial rain density. Consequently, the gating mechanism employs G to decide the fraction of features in the main branch, denoted as $X' \in \mathbb{R}^{H\times W\times C}$, that should be forwarded to the subsequent layers of the network.

$$Output = \mathrm{PointwiseConv}(X' \odot G) + Input \tag{8}$$

Since the guide feature G calculated from HFP represents spatial rain density, the gating mechanism based on it can highlight the rain-rich area in the image features. By concentrating on the rain rather than the background, the FGF module allows subsequent layers to remove rain more precisely.

3 Experimental Results and Discussion

3.1 Experimental Settings

Datasets. We conduct experiments on five challenging synthetic benchmark datasets including Rain200H [27], Rain200L [27], Rain1200 [33], DDN [7] and SPA-DATA [21]. In addition, Ren et al. [18] provide a widely-used real-world rainy images dataset PReNetReal. More details about the datasets are provided in supplementary material.

Comparison Methods. Various kinds of SID methods are included in our comparison, as shown in Table 1. We utilize pre-trained models for evaluation when available. For method without pre-trained models, we train them using the hyper-parameters specified by the original authors. For situations where parameters are not fully reported or source code is unavailable, we refer to results from the original publications.

Evaluation Metrics. To assess restoration performance, we employ the commonly-used PSNR and SSIM metrics on paired datasets. Following the previous work [4,6], we convert images into YCbCr space and then evaluate PSNR for the Y channel to focus on the luminance. For real-world datasets lacking ground truth, we utilize no-reference quality metrics NIQE [17] and BRISQUE [16] to conduct quantitative comparisons. These metrics do not refer to ground truth but directly evaluate image naturalness to the human eyes, based on learning-derived evaluators. Therefore, they can reflect the quality of the rain-removed image.

Implementation Details. From the surface layer to the deepest layer, the number of Transformer blocks and attention heads arranged as [(4, 1), (6, 2), (6, 4), (8, 8)], and channel counts at [48, 96, 192, 384]. 3 additional transformer blocks are placed at the end of the model to directly enhance final rain removal result. Before processing, each image is cropped to a 128 × 128 patch, with random vertical and horizontal flips applied. The training utilizes the AdamW optimizer, with momentum set at (0.9, 0.999) and weight decay set at 0.0001. The initial learning rate of 3×10^{-4} is gradually reduced to 1×10^{-6} using a cosine annealing with restarts mechanism. We train models for 3×10^5 iterations across all datasets with a batch size of 4, using 2 NVIDIA RTX 3090 GPUs.

Table 1. Quantitative results on synthetic datasets. The **optimal results** on the same metric are in bold, and the suboptimal results are underlined.

Datasets		Rain200H		Rain200L		DDN		Rain1200		SPA-DATA	
Metrics		PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
Traditional based	DSC [14]	14.73	0.3815	27.16	0.8663	27.31	0.8373	24.24	0.8279	34.95	0.9416
	GMM [15]	14.50	0.4164	28.66	0.8625	27.55	0.8479	25.81	0.8344	34.30	0.9428
Convolution based	DDN [7]	26.05	0.8056	34.64	0.9671	30.00	0.9041	30.97	0.9116	36.16	0.9428
	PReNet [18]	29.04	0.8991	37.80	0.9814	32.60	0.9459	33.17	0.9481	40.16	0.9816
	RESCAN [13]	26.75	0.8353	36.09	0.9697	31.94	0.9345	33.38	0.9417	38.11	0.9707
	DualGCN [8]	31.15	0.9125	40.73	0.9886	33.01	0.9489	34.37	0.9620	44.18	0.9902
	SPDNet [30]	31.28	0.9207	40.50	0.9875	33.15	0.9457	34.57	0.9560	43.20	0.9871
	MPRNet [32]	30.67	0.9110	39.47	0.9825	33.10	0.9347	33.99	0.9590	43064	0.9844
Transformer based	Uformer [22]	30.80	0.9105	40.20	0.9860	33.95	0.9545	35.02	0.9621	46.13	0.9913
	Restormer [31]	32.00	0.9329	40.97	0.9890	34.20	0.9571	35.29	0.9641	47.98	0.9921
	IPT [3]	32.10	**0.9433**	40.74	0.9890	33.84	0.9549	34.89	0.9623	47.35	**0.9930**
	DRSformer [4]	32.20	0.9326	41.23	0.9894	34.35	**0.9588**	35.35	0.9646	48.54	0.9924
	Ours	**32.47**	0.9375	**41.51**	**0.9910**	**34.41**	0.9586	**35.37**	**0.9660**	**48.62**	0.9923

3.2 Comparison Results

For Synthetic Dataset. As is evident from the data in Table 1, SFformer achieves consistent and significant performance improvement over existing methods on the PSNR and SSIM metrics when evaluating on the five mostly used datasets. The datasets include rainy scenes of various sizes and densities, and SFformer's consistently excellent performance across these scenarios demonstrates its superiority. Notably, our method surpasses the recently acclaimed DRSformer, achieving an enhancement of 0.14dB in PSNR and 0.0016 in SSIM on average across these datasets.

In addition to these quantitative achievements, the visual outcomes depicted in Fig. 6 further underscore the superiority of our proposed method.

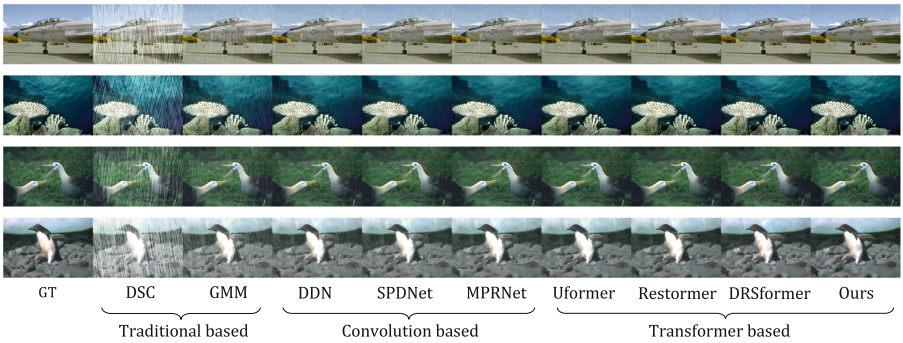

Fig. 6. Subjective comparison on synthetic datasets between methods.

For Real-World Dataset. To further substantiate the effectiveness of the proposed method, we employed the real-world dataset PReNetReal to benchmark SFformer against other leading methods. As indicated in Table 2, our method, SFformer, achieves lower NIQE and BRISQUE scores compared to all other evaluated methods. That signifies superior perceptual quality in SFformer's rain removal results.

Table 2. Quantitative results on real-world datasets. The **optimal results** are in bold.

Method	GMM	MPRNet	Uformer	Restormer	DRSformer	Ours
NIQE↓	6.291	4.637	4.928	5.310	4.301	**4.019**
BRISQUE↓	49.19	27.03	27.25	33.46	26.24	**24.78**

Illustrated in Fig. 7, it is observed that most models leave some apparent rain streaks in their rain removal results, whereas SFformer, MPRNet and DRSformer produce relatively cleaner images. However, MPRNet and DRSformer tend to compromise structural details or induce blurriness around edges in their rain removal results. In contrast, our SFformer maintains high background quality while effectively removing rain streaks, demonstrating its advanced performance in preserving image quality during real rain removal.

3.3 Ablation Study

In this section, several ablation study on each component of the SFformer is conducted to assess the effectiveness of our module designs. For ablation experiments, all SID models were trained on Rain200H for 3×10^5 iterations with batch size set to 4. As detailed in Table 3(a)(d)(i)(j), Simply removing each of our proposed modules will cause performance decreases to different extents, revealing

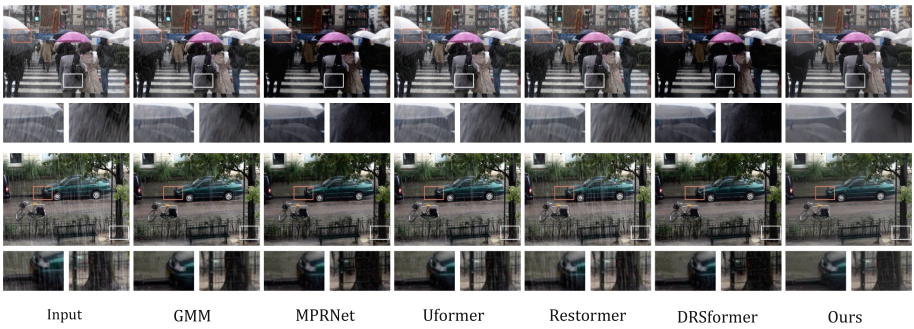

Fig. 7. Comparison on real-world dataset between methods.

that our specially designed ASA and FGF modules significantly enhance performance. We will conduct more detailed ablation studies then. Due to space limitations, details of the model variants in the ablation experiments are given in the supplementary material.

Table 3. Ablation study results of ASA module and FGF module.

	PSNR↑	SSIM↑	Params(M)
a. Baseline	31.77	0.9297	31.24
b. Baseline+soft threshold	32.06	0.9299	31.67
c. TKSA [4]	32.01	0.9326	31.85
d. ASA(ours)	32.26	0.9331	32.12
e. Baseline+5*5Dconv	32.05	0.9301	35.85
f. Baseline+frequency guide	32.04	0.9327	31.25
g. GDFN [31]	31.81	0.9304	31.64
h. MSFN [4]	31.97	0.9286	34.40
i. FGF(ours)	32.14	0.9335	35.87
j. Full SFformer	32.47	0.9375	36.68

Effictiveness of ASA Module. In Table 3, a quantitative ablation study illustrates the impact of the ASA module. In Table 3(a)(b)(d), we sequentially remove the adaptive threshold generation sub-net and the soft threshold sparse mechanism from the complete ASA module. The observed stepwise performance decrease clearly demonstrates the critical roles that both the above two play in enhancing the module's effectiveness. In addition, the experimental results in Table 3(c)(d) proves that our sparsification strategy is better than that in DRSformer [4]. The results confirm that ASA's sparsification and selective aggregation significantly contribute to the overall performance boost.

Effictiveness of FGF Module. To evaluate the benefits of our proposed design, we compare FGF module with several possiple alternatives. Normal 3*3 kernel size depth-wise feedforward is used as a baseline in Table 3(a). Firstly, Table 3(a)(e)(f)(i) has proved that the performance improvement does not come from the use of larger convolution kernels or frequency guidance mechanisms independently, but from their synergistic effect. In addition, in order to compare between different alternatives to vanilla feedforward layers, we replaced our frequency-guided feedforward blocks with GDFN (utilized in Restormer) [31] or MSFN (employed in DRSformer) [4] blocks, resulting in a performance decline, as shown in Table 3(g)(h)(i). The results confirm that utilizing HFP features to implement gating mechanism to guide model focus on rain significantly increases PSNR and SSIM, which validates the FGF module's superiority.

Qualitative Analysis of ASA. To dive deep into how the ASA module boost the performance, it's crucial to investigate whether the blocked channel connections genuinely not contribute to restoration quality. The sparsification of the attention matrix can be likened to pruning operation. By employing a well-established parameter importance evaluation method from the model pruning domain [26], we can assess the contribution of the attention matrix $Attn^{(n,h)} = \{a_{km}^{(n,h)}\}$. Note that k and m denote the indices of elements in the h-th attention head of level-n layer $Attn^{(n,h)}$.

Considering that the attention matrix influences the end result by modulating the aggregation weight of the different channels.

$$Out^{(n,h)} = Attn^{(n,h)} V^{(n,h)} \tag{9}$$

To quantify the contribution of a specific element in the attention matrix, we assess the impact on performance resulting from its removal, akin to the method used in pruning, which is represented in Eq. 10.

$$\mathcal{I}_{km}^{(n,h)} = \mathcal{L}(V^{(n,h)}, Attn^{(n,h)} | a_{km}^{(n,h)} = 0) - \mathcal{L}(V^{(n,h)}, Attn^{(n,h)}) \tag{10}$$

For an efficient evaluation of $\mathcal{I}_{km}^{(n,h)}$ in Eq. 10, A first-order Taylor expansion is employed, as follows, as illustrated in Eq. 11.

$$\mathcal{L}(V^{(n,h)}, Attn^{(n,h)} | a_{km}^{(n,h)} = 0) = \mathcal{L}(V^{(n,h)}, Attn^{(n,h)}) + Grad_{a_{km}} a_{km}^{(n,h)} \tag{11}$$

Substituting this approximation into Eq. 10, we obtain the gain:

$$\mathcal{I}^{(n,h)} = Grad_{Attn}^{(n,h)} Attn^{(n,h)} \tag{12}$$

We calculate the connection gains before and after sparsification, as shown in Fig. 8. In these visualizations, blue represents a strong positive gain, whereas red is used to indicate weak positive or even negative gain. It can be found that after sparsification, the map's color shifts predominantly to blue, which indicates an improvement. This shift confirms that the majority of masked connections were indeed negative-gain, while the preserved connections predominantly exhibit high positive gain, aligning with our initial hypothesis.

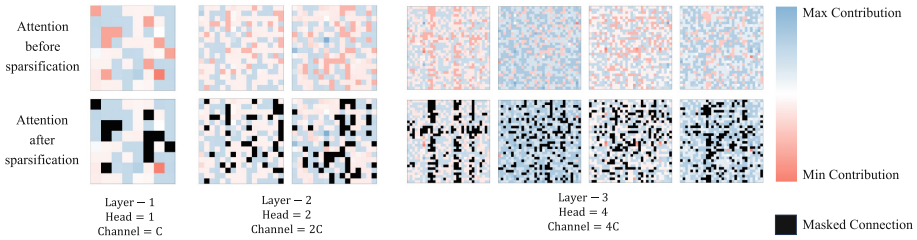

Fig. 8. Gain of attention before and after mask

Qualitative Analysis of FGF. Due to the space limitations, the qualitative analysis of FGF is provided in the supplementary material.

4 Conclusion

We present the SFformer, an adaptive sparse and frequency-guided transformer network for single image derain, featuring the innovative ASA and FGF modules. Specifically, the adaptive sparse attention module (ASA) module dynamically determines an appropriate threshold for sparsification. It enables selective feature aggregation from the most relevant channels, thereby improving the quality of the aggregated features. Concurrently, frequency-guided feedforward module (FGF) module leverages prior knowledge to effectively direct the model's focus towards areas with high HFP density, helping model pay more attention to rain in the image. Equipped with these novel modules, SFformer achieves state-of-the-art results across current SID datasets.

Limitations. Since the frequency prior knowledge we utilized is for the rain streaks, which is the derain targets most of the time, this method cannot handle raindrops attached to the lens well. In future research, it would be beneficial to explore how to remove various forms of rain, as well as other adverse weather conditions such as fog and snow. Moreover, although the size of the model is comparable to current transformer-based methods, it still limits the application on some edge devices.

References

1. Chen, C., Li, H.: Robust representation learning with feedback for single image deraining. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 7738–7747 (2021)
2. Chen, D.Y., Chen, C.C., Kang, L.W.: Visual depth guided color image rain streaks removal using sparse coding. IEEE Trans. Circuits Syst. Video Technol. **24**(8), 1430–1455 (2014). https://doi.org/10.1109/TCSVT.2014.2308627

3. Chen, H., et al.: Pre-trained image processing transformer. In: 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 12299–12310 (2021)
4. Chen, X., Li, H., Li, M., Pan, J.: Learning a sparse transformer network for effective image deraining. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 5896–5905 (2023)
5. Chen, Y., Yan, Z., Ma, L.: New insights on the generation of rain streaks: generating-removing united unpaired image deraining network. In: Chinese Conference on Pattern Recognition and Computer Vision (PRCV), pp. 390–402. Springer (2023)
6. Cui, Y., et al.: Selective frequency network for image restoration. In: The Eleventh International Conference on Learning Representations (2023)
7. Fu, X., Huang, J., Zeng, D., Huang, Y., Ding, X., Paisley, J.: Removing rain from single images via a deep detail network. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 3855–3863 (2017)
8. Fu, X., Qi, Q., Zha, Z.J., Zhu, Y., Ding, X.: Rain streak removal via dual graph convolutional network. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 35, pp. 1352–1360 (2021)
9. Fu, Y.H., Kang, L.W., Lin, C.W., Hsu, C.T.: Single-frame-based rain removal via image decomposition. In: 2011 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 1453–1456 (2011). https://doi.org/10.1109/ICASSP.2011.5946766
10. Goodfellow, I., et al.: Generative adversarial networks. Commun. ACM **63**(11), 139–144 (2020)
11. He, Y., Jiang, A., Jiang, L., Wang, Z., Wang, L.: Dual-path coupled image deraining network via spatial-frequency interaction. arXiv:2402.04855 (2024)
12. Li, R., Cheong, L.F., Tan, R.T.: Heavy rain image restoration: Integrating physics model and conditional adversarial learning. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 1633–1642 (2019)
13. Li, X., Wu, J., Lin, Z., Liu, H., Zha, H.: Recurrent squeeze-and-excitation context aggregation net for single image deraining. In: Proceedings of the European Conference on Computer Vision (ECCV), pp. 254–269 (2018)
14. Li, Y., Tan, R.T., Guo, X., Lu, J., Brown, M.S.: Rain streak removal using layer priors. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2736–2744 (2016)
15. Luo, Y., Xu, Y., Ji, H.: Removing rain from a single image via discriminative sparse coding. In: 2015 IEEE International Conference on Computer Vision (ICCV), pp. 3397–3405 (2015). https://doi.org/10.1109/ICCV.2015.388
16. Mittal, A., Moorthy, A.K., Bovik, A.C.: No-reference image quality assessment in the spatial domain. IEEE Trans. Image Process. **21**(12), 4695–4708 (2012)
17. Mittal, A., Soundararajan, R., Bovik, A.C.: Making a "completely blind" image quality analyzer. IEEE Signal Process. Lett. **20**(3), 209–212 (2012)
18. Ren, D., Zuo, W., Hu, Q., Zhu, P., Meng, D.: Progressive image deraining networks: a better and simpler baseline. In: 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 3932–3941 (2019)
19. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. arXiv:1409.1556 (2014)
20. Vaswani, A., et al.: Attention is all you need. In: Advances in Neural Information Processing Systems, vol. 30 (2017)

21. Wang, T., Yang, X., Xu, K., Chen, S., Zhang, Q., Lau, R.W.: Spatial attentive single-image deraining with a high quality real rain dataset. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 12270–12279 (2019)
22. Wang, Z., Cun, X., Bao, J., Zhou, W., Liu, J., Li, H.: Uformer: a general U-shaped transformer for image restoration. In: 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 17662–17672 (2022)
23. Wei, W., Meng, D., Zhao, Q., Xu, Z., Wu, Y.: Semi-supervised transfer learning for image rain removal. In: 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 3872–3881 (2019)
24. Wei, Y., et al.: DerainCycleGAN: rain attentive cycleGAN for single image deraining and rainmaking. IEEE Trans. Image Process. **30**, 4788–4801 (2021)
25. Wei, Y., et al.: Semi-derainGAN: a new semi-supervised single image deraining. In: 2021 IEEE International Conference on Multimedia and Expo (ICME), pp. 1–6. IEEE (2021)
26. Yang, H., Yin, H., Molchanov, P., Li, H., Kautz, J.: NViT: vision transformer compression and parameter redistribution. arXiv:2110.04869 (2021)
27. Yang, W., Tan, R.T., Feng, J., Liu, J., Guo, Z., Yan, S.: Deep joint rain detection and removal from a single image. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1357–1366 (2017)
28. Yasarla, R., Sindagi, V.A., Patel, V.M.: Syn2Real transfer learning for image deraining using Gaussian processes. In: 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 2723–2733 (2020)
29. Ye, Y., Chang, Y., Zhou, H., Yan, L.: Closing the loop: joint rain generation and removal via disentangled image translation. In: 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 2053–2062 (2021)
30. Yi, Q., Li, J., Dai, Q., Fang, F., Zhang, G., Zeng, T.: Structure-preserving deraining with residue channel prior guidance. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 4238–4247 (2021)
31. Zamir, S.W., Arora, A., Khan, S., Hayat, M., Khan, F.S., Yang, M.: Restormer: efficient transformer for high-resolution image restoration. In: 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 5718–5729 (2022)
32. Zamir, S.W., et al.: Multi-stage progressive image restoration. In: 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 14816–14826 (2021)
33. Zhang, H., Patel, V.M.: Density-aware single image de-raining using a multi-stream dense network. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 695–704 (2018)
34. Zhang, H., Sindagi, V., Patel, V.M.: Image de-raining using a conditional generative adversarial network. IEEE Trans. Circuits Syst. Video Technol. **30**(11), 3943–3956 (2019)