Image Deraining with Frequency-Enhanced State Space Model

Shugo Yamashita* and Masaaki Ikehara

Keio University, Yokohama, Japan
{yamashita,ikehara}@tkhm.elec.keio.ac.jp

Abstract. Removing rain degradations in images is recognized as a significant issue. In this field, deep learning-based approaches, such as Convolutional Neural Networks (CNNs) and Transformers, have succeeded. Recently, State Space Models (SSMs) have exhibited superior performance across various tasks in both natural language processing and image processing due to their ability to model long-range dependencies. This study introduces SSM to image deraining with deraining-specific enhancements and proposes a Deraining Frequency-Enhanced State Space Model (DFSSM). To effectively remove rain streaks, which produce high-intensity frequency components in specific directions, we employ frequency domain processing concurrently with SSM. Additionally, we develop a novel mixed-scale gated-convolutional block, which uses convolutions with multiple kernel sizes to capture various scale degradations effectively and integrates a gating mechanism to manage the flow of information. Finally, experiments on synthetic and real-world rainy image datasets show that our method surpasses state-of-the-art methods. Code is available at https://github.com/ShugoYamashita/DFSSM.

**Keywords:** State Space Model · Deraining · Image Restoration

1 Introduction

Images captured under rainy conditions suffer from deteriorated visual quality. Rain-related degradations adversely affect the performance of downstream tasks, such as object detection, object tracking, and segmentation, as well as the overall effectiveness of vision applications that employ these algorithms. Removing rain degradations in images enhances both the performance and reliability of vision systems.

Single image deraining has been extensively studied for its usefulness. Recently, deep learning-based approaches have significantly advanced its performance. Convolutional Neural Network (CNN)-based methods [5, 6, 17, 19, 30, 37, 38, 43, 45] surpassed the previous handcrafted prior methods [9, 18, 20, 25, 47]. However, CNNs cannot capture global features because they collect features from local regions using fixed-size kernels. Subsequently developed, Transformer [36]

---

* Corresponding author

Rainy	Clear	Rainy − Clear
$|\mathcal{F}(\text{Rainy})|$	$|\mathcal{F}(\text{Clear})|$	$|\mathcal{F}(\text{Rainy} - \text{Clear})|$
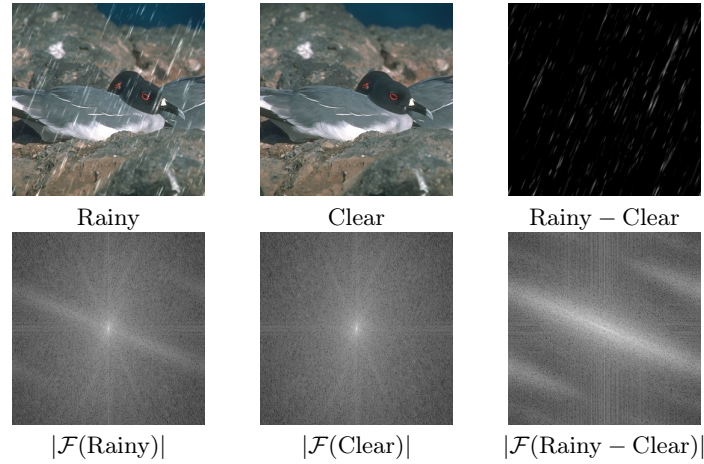
Fig. 1: Top row: a rainy image, a clear image, and their differential image. Bottom row: Corresponding 2D Fourier amplitude spectra. The amplitude of the Fourier transform is denoted as $|\mathcal{F}(\cdot)|$.

employs the Self-Attention (SA) mechanism that captures global information and enables inference based on input features. In SA, the key-query dot-product requires computational complexity of $\mathcal{O}(H^2W^2)$ for an input image with $H \times W$ pixels. Consequently, applying standard SA to image deraining in high-resolution images is computationally expensive and impractical. To alleviate this issue, SA is calculated across channels instead of spatial dimension [44], within localized window regions [41], or sparsely [1,2]. However, these methods do not necessarily achieve the effectiveness of standard SA in capturing global features.

Recently, State Space Models (SSMs) [7,8] have emerged as innovative architectures for deep neural networks. SSMs conduct sequence-to-sequence transformation using hidden states to interact with previously scanned elements, thereby modeling long-range dependencies with linear complexity. Mamba [7] enhances this architecture by treating parameters of SSMs as functions of the input, realizing content-driven inference. Mamba [7] architecture, initially designed for one-dimensional data in natural language processing, is applied for image processing by flattening two-dimensional images into one-dimensional sequences. Its effectiveness has been demonstrated in image recognition [15, 22, 29, 49], medical image segmentation [21, 26], and image restoration [10, 33]. However, in image deraining task, MambaIR [10], a generalized and simple SSM, is inferior to deraining-specific models.

In this paper, we present an effective deraining-specific SSM, taking into account the unique characteristics of this task. We develop a novel Deraining Frequency-Enhanced State Space Model (DFSSM) by integrating the SSM with frequency domain processing. To build an efficient deraining model capable of capturing long-range dependencies, we use the SSM as the primary architecture. In addition, we adopt frequency domain processing that are effective for de-

raining. As depicted in Fig. 1, Fourier transformation highlights the directional structures of rain streaks. In the spatial domain, rain streaks are dispersed across the image, but in the frequency domain, they appear as concentrated components along specific directions. Thus, frequency domain processing enables the effective removal of rain streaks as a unified spectral element.

Specifically, we propose a Frequency-Enhanced State Space Block (FSSB), which uses a Fast Fourier Transform Module (FFTM) for frequency domain processing in parallel with a Vision State Space Module (VSSM) for SSM operations. During training, a Frequency Reconstruction loss [27, 35] is adopted to promote learning based on frequency components. Furthermore, a Mixed-Scale Gated-Convolutional Block (MGCB) is used to capture local features. Convolutions with multiple kernel sizes are employed to process diverse scales of rain streaks effectively. Concurrently, adopting a gating mechanism enables the selective transmission of features.

Comprehensive experiments are conducted to evaluate the proposed DFSSM. Across synthetic rainy image datasets such as Rain200H [42], Rain200L [42], and DID-Data [48], as well as real-world datasets including SPA-Data [39] and LHP-Rain [12], our method achieves quantitatively and qualitatively superior performance compared with state-of-the-art methods. Ablation studies demonstrate that the proposed improvements are effective in removing rain degradations.

The contributions of this paper are summarized as follows:

- We propose a Deraining Frequency-Enhanced State Space Model (DFSSM). Our method uses a Frequency-Enhanced State Space Block (FSSB) containing a Vision State Space Module (VSSM) and a Fast Fourier Transform Module (FFTM) for effective image deraining.
- We develop a Mixed-Scale Gated-Convolutional Block (MGCB), which effectively captures various scale degradations through mixed-scale convolutions and manages the flow of features using a gating mechanism.
- Experimental results show that our method achieves state-of-the-art performance on synthetic and real-world rainy image datasets.

2 Related Works

2.1 Image Deraining

Extensive research has been conducted to restore rain-degraded images. Traditional methods [9, 18, 20, 25, 47] separate a rainy image into the background and rain layers with various handcrafted priors, such as discriminative sparse coding [25] and Gaussian mixture model [20]. However, these prior-based approaches have difficulty dealing with complex rain forms.

Currently, deep learning-based methods are the dominant approaches for image deraining. Initially, CNN-based methods [5, 6, 17, 19, 30, 37, 38, 43, 45] were extensively studied. DDN [5] used a deep convolutional neural network, focusing on high-frequency detail during training. Recursive computation [17, 19, 30, 37] and multi-scale representation [17, 37, 45] were successfully deployed to remove

rain streaks. RCDNet [38] proposed interpretable network architecture with a convolutional dictionary learning model. DualGCN [6] used dual graph convolutional networks to consider long-range contextual information. SPDNet [43] employed a wavelet-based feature extraction backbone with a residue channel prior guided mechanism.

Recent studies [1, 2, 41, 44] have applied Transformer [36] to image deraining. These methods are designed to reduce the computational quadratic complexity of standard Self-Attention. Restormer [44] applied Self-Attention across channels instead of the spatial dimension. IDT [41] used a window-based Transformer and a spatial Transformer to extract local and global features. A sparse attention mechanism, which calculates attention using only highly relevant elements rather than all elements, is introduced to rain removal along with a learnable top-k selection operator [2] and an uncertainty-driven ranking strategy [1].

2.2   State Space Models

State Space Models (SSMs), recently incorporated into deep learning from classical control theory, are designed for sequence-to-sequence transformation. Structured State Spaces model (S4) [8] handles long sequences effectively with linear computation. Mamba [7] enabled content-driven inference with parameters of SSMs as functions of the input and achieved state-of-the-art performance in the natural language domain. Vim [49] and VMamba [22], which adapted Mamba to the field of image recognition, transform 2D images into 1D sequences by scanning with multiple routes and then using SSMs. LocalMamba [15] improves the scan direction and enhances the ability to capture local 2D relationships. EfficientVMamba [29] proposed a lightweight model. SSMs have also been applied in medical image segmentation [21, 26] and image restoration [10, 33]. In this study, we utilize SSM with frequency enhancement to remove rain degradations.

3   Proposed Method

3.1   Overall Pipeline

The overall architecture of the proposed Deraining Frequency-Enhanced State Space Model (DFSSM) is illustrated in Fig. 2. Given a rainy input image $I \in \mathbb{R}^{3 \times H \times W}$, our deraining network removes degradations and outputs a clear image $\hat{I} \in \mathbb{R}^{3 \times H \times W}$. Our hierarchical encoder-decoder network employs SSM with MambaIR [10] as the baseline.

We employ a U-Net [31] architecture akin to Restormer [44], comprising 8 stages across four resolution levels: $H \times W$, $\frac{H}{2} \times \frac{W}{2}$, $\frac{H}{4} \times \frac{W}{4}$ and $\frac{H}{8} \times \frac{W}{8}$. Initially, the input image I undergoes channel expansion from 3 to C via a 3×3 convolution. Then, the encoder and decoder are applied, processing features while decreasing and increasing resolution, respectively. Downsampling and upsampling of image sizes are conducted through pixel-unshuffle and pixel-shuffle techniques [32]. Skip connections facilitate the integration of features from the
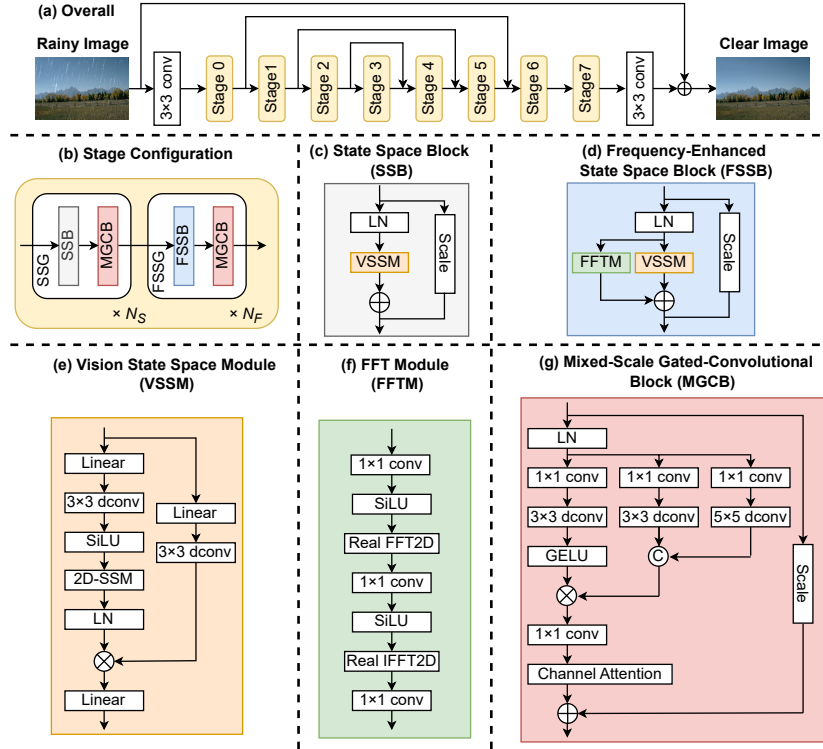
Fig. 2: The architecture of our Deraining Frequency-Enhanced State Space Model (DFSSM). The overall U-Net architecture (a) has 8 stages. Each stage (b) consists of N_S State Space Groups (SSGs) and N_F Frequency-Enhanced State Space Groups (FSSGs). (c), (d), (e), (f), and (g) illustrate the details of the components. The SSG includes a State Space Block (SSB) and a Mixed-Scale Gated-Convolutional Block (MGCB), while the FSSG includes a Frequency-Enhanced State Space Block (FSSB) and an MGCB. Both SSB and FSSB employ a Vision State Space Module (VSSM), and the FSSB also uses a Fast Fourier Transform Module (FFTM).

encoder into the decoder. After the skip connections at $\frac{H}{2} \times \frac{W}{2}$ and $\frac{H}{4} \times \frac{W}{4}$ resolution levels, the channels are halved through a 1×1 convolution. The features from the decoder are fed into a refinement stage operating at high resolution. Finally, applying a 3×3 convolution and adding skip connection from the input image I, we obtain the clear image $\hat{I}$.

Each encoder, decoder, and refinement stage comprises N_S State Space Groups (SSGs) and N_F Frequency-Enhanced State Space Groups (FSSGs). The SSG is a series connection of a State Space Block (SSB) and a Mixed-Scale Gated-Convolutional Block (MGCB). The FSSG is a series connection of a Frequency-

Enhanced State Space Block (FSSB) and an MGCB. The following describes the components and loss functions of our model in detail.

3.2   State Space Block (SSB)

We employ the State Space Model (SSM) to remove rain degradations. The SSM can capture global information with linear complexity. Following MambaIR [10], the State Space Block (SSB), shown in Fig. 2(c), contains a layer normalization, a Vision State Space Module (VSSM), and a skip connection using a learnable scale factor. Given the input tensor X, the SSB can be defined as:

$$X_{out} = \text{VSSM}(\text{LN}(X)) + s \cdot X, \tag{1}$$

where LN represents the layer normalization and s denotes the learnable scale factor. In the VSSM [22] (Fig. 2(e)), the 2D Selective Scan Module [10,22] scans 2D feature maps in four distinct directions, flatting them into 1D sequences. Then, each sequence is processed by SSM, and the outcomes are merged.

3.3   Frequency-Enhanced State Space Block (FSSB)

To enhance the capability of removing rain streaks, we utilize frequency domain processing. Transforming an image into the frequency domain allows the separation of high-frequency signals, representing fine details and textures, and low-frequency signals, corresponding to flat regions. Separately processing these distinct features within the frequency domain enhances the model expressiveness, as successfully demonstrated in various computer vision tasks [3, 27, 46], including image deraining [11, 18, 34]. The efficacy of frequency domain processing, particularly in deraining, is supported by the following observation. Fig. 1 shows 2D Fourier amplitude spectra of a rain image, a clear image, and their difference image. Rain streaks, which typically exhibit strong structural components in specific directions, are clearly highlighted by Fourier transformation. While rain streaks are scattered across various positions in the spatial domain, they manifest as concentrated spectral components in particular orientations in the frequency domain. Treating rain streaks as a unified spectral component facilitates effective deraining.

Motivated by these insights, we introduce the Frequency-Enhanced State Space Block (FSSB), which expands the SSB. As shown in Fig. 2(d), the FSSB uses a Fast Fourier Transform Module (FFTM) in parallel with the VSSM:

$$X_{out} = \text{VSSM}(\text{LN}(X)) + \text{FFTM}(\text{LN}(X)) + s \cdot X. \tag{2}$$

In the FFTM (Fig. 2(f)), we initially use a 1×1 convolution that halves the channel and a SiLU activation function [4] to obtain the suitable feature map for processing in the frequency domain. Subsequently, a Fast Fourier Transform (FFT) transforms the feature map from the spatial domain to the frequency domain. Given that the image features are represented as a real tensor, a 2D

Real FFT is used to reduce computational complexity by exploiting symmetry. To extract features in the frequency domain, we use a 1×1 convolution with a SiLU. Then, a 2D Real Inverted FFT (IFFT) is used to bring it back from the frequency domain to the spatial domain. Finally, a 1×1 convolution expands the channels to the original number of channels. Given the input features Z, the FFTM process is defined as:

$$\hat{Z} = \text{SiLU}(1 \times 1 \text{ Conv}(Z)), \tag{3}$$

$$Z_f = \mathcal{F}^{-1}(\text{SiLU}(1 \times 1 \text{ Conv}(\mathcal{F}(\hat{Z})))), \tag{4}$$

$$Z_{out} = 1 \times 1 \text{ Conv}(Z_f), \tag{5}$$

where $\mathcal{F}$ and $\mathcal{F}^{-1}$ denote the 2D Real FFT and IFFT.

3.4 Mixed-Scale Gated-Convolutional Block (MGCB)

SSB and FSSB can capture the spatial long-term dependency. After these components, we use the Mixed-Scale Gated-Convolutional Block (MGCB) for its proficiency in gathering local details. MGCB contains depth-wise convolutions, aggregating information from spatially neighboring pixels. To remove rain degradations of various scales, multi-scale representation [17,37,45] and convolutions [2] are effective. Hence, we adopt multiple kernel sizes for depthwise-convolution. Following [44], this convolutional block incorporates a gating mechanism. This gating mechanism allows the passage of crucial information while blocking non-essential information, thereby managing the flow of information at each hierarchical level.

Specifically, the MGCB is depicted in Fig. 2(g). The input tensor X is applied a layer normalization and fed into three parallel branches: a gate branch, a 3×3 dconv branch, and a 5×5 dconv branch. Within these branches, the channels are initially expanded by factors of γ, $\frac{\gamma}{2}$, and $\frac{\gamma}{2}$, respectively, through 1×1 convolutions. γ is a hyperparameter of the expansion rate. In the gate branch, a 3×3 depth-wise convolution followed by a GELU activation function [13] is conducted to get X_{gate}. In the 3×3 and 5×5 dconv branches, 3×3 and 5×5 depth-wise convolutions are performed with output $X_{3 \times 3}$ and $X_{5 \times 5}$, respectively. In the process of integrating these branches, $X_{3 \times 3}$ and $X_{5 \times 5}$ are concatenated on a channel-wise basis, subsequently taking the element-wise product with X_{gate}. Then, a 1×1 convolution is conducted to restore the channel dimension. Furthermore, we use the channel attention [14] to consider the global context in this block. Last, a skip connection with a learnable scale factor is added to obtain X_{out}. In conclusion, the MGCB can be formulated as:

$$X_{gate} = \text{GELU}(3 \times 3 \text{ DW-Conv}(\text{LN}(X))), \tag{6}$$

$$X_{3 \times 3} = 3 \times 3 \text{ DW-Conv}(\text{LN}(X)), \tag{7}$$

$$X_{5\times5} = 5 \times 5 \text{ DW-Conv}(\text{LN}(X)), \tag{8}$$

$$X_{out} = \text{CA}(X_{gate} \odot [X_{3\times3}, X_{5\times5}]) + s \cdot X, \tag{9}$$

where DW-Conv denotes depth-wise convolution, CA stands for the channel attention, and $[\cdot]$ represents the channel-wise concatenation.

3.5 Loss Functions

The loss function of our method is calculated based on the model output $\hat{I}$ and the ground truth G. We use an L1-loss and a Frequency Reconstruction loss [27, 35]. The L1-loss is formulated as:

$$\mathcal{L}_{L1} = \|\hat{I} - G\|_1, \tag{10}$$

where $\|\cdot\|_1$ is the $L1$ norm. The Frequency Reconstruction loss [27,35] computes the loss at each frequency component:

$$\mathcal{L}_{Freq} = \|\mathcal{F}(\hat{I}) - \mathcal{F}(G)\|_1 \tag{11}$$

where $\mathcal{F}(\cdot)$ denotes the 2D Real Fast Fourier Transform. The total loss is:

$$\mathcal{L}_{total} = \mathcal{L}_{L1} + \lambda_f \mathcal{L}_{Freq}, \tag{12}$$

where λ_f are coefficients to adjust the ratio of the losses.

4   Experiments

4.1   Experimental settings

Datasets. In our experiment, existing synthetic and real-world datasets are used to validate our proposed method. Each dataset contains image pairs; one is degraded by rain, and the other is the same scene without rain. Rain200H and Rain200L [42] include images of synthetic heavy and light rain streaks, respectively. Both datasets have $1,800$ training image pairs and 200 testing image pairs. DID-Data [48] consists of $12,000$ synthetic training image pairs and $1,200$ testing image pairs with different rain-density levels. SPA-Data [39] and LHP-Rain [12] are real-world rainy image datasets comprising $638,492$ and $758,997$ training image pairs, respectively. Each dataset includes $1,000$ testing image pairs.

Table 1: Quantitative results on the Rain200H, Rain200L [42], DID-Data [48], SPA-Data [39], and LHP-Rain [12].

Architecture	Method	Rain200H [42]		Rain200L [42]		DID-Data [48]		SPA-Data [39]		LHP-Rain [12]	
		PSNR↑	SSIM↑	PSNR↑	SSIM↑	PSNR↑	SSIM↑	PSNR↑	SSIM↑	PSNR↑	SSIM↑
Prior	DSC [25]	14.73	0.3815	27.16	0.8663	24.24	0.8279	34.95	0.9416	-	-
	GMM [20]	14.50	0.4164	28.66	0.8652	25.81	0.8344	34.30	0.9428	-	-
CNN	DDN [5]	26.05	0.8056	34.68	0.9671	30.97	0.9116	36.16	0.9457	32.61	0.9037
	RESCAN [19]	26.75	0.8353	36.09	0.9697	33.38	0.9417	38.11	0.9707	32.72	0.9058
	PReNet [30]	29.37	0.9029	37.95	0.9816	33.17	0.9481	40.16	0.9816	33.17	0.9177
	MSPFN [17]	29.36	0.9034	38.58	0.9827	33.72	0.9550	43.43	0.9843	32.20	0.8955
	RCDNet [38]	30.24	0.9048	39.17	0.9885	34.08	0.9532	43.36	0.9831	32.98	0.9119
	MPRNet [45]	30.67	0.9110	39.47	0.9825	33.99	0.9590	43.64	0.9844	32.43	0.8925
	DualGCN [6]	31.15	0.9125	40.73	0.9886	34.41	0.9624	44.18	0.9902	33.38	0.9200
	SPDNet [43]	31.28	0.9207	40.50	0.9875	34.57	0.9560	43.20	0.9871	32.98	0.9122
Transformer	Restormer [44]	32.00	0.9329	40.99	0.9890	35.29	0.9641	47.98	0.9921	33.77	0.9290
	IDT [41]	32.10	0.9344	40.74	0.9884	34.95	0.9627	47.34	0.9930	33.43	0.9246
	DRSformer [2]	32.17	0.9326	41.23	0.9894	35.35	0.9646	48.54	0.9924	33.55	0.9227
SSM	**DFSSM**	**32.99**	**0.9403**	**41.81**	**0.9905**	**35.66**	**0.9671**	**49.55**	**0.9939**	**33.99**	**0.9322**

Implementation Details. Our proposed DFSSM is implemented on PyTorch framework [28]. Each stage has $N_S = 1$ SSG and $N_F = 3$ FSSGs, as empirically determined in Sec. 4.3. The initial channel C is 32. In the MGCB, the expansion rate γ is 2.0. The coefficient λ_f of the loss functions is set to 0.01.

Our DFSSM is trained for $600,000$ iterations on LHP-Rain [12] and for $300,000$ iterations on the other datasets. The initial learning rate 3×10^{-4} is gradually decreased to 1×10^{-6} with the cosine annealing strategy [23]. Our model is optimized by AdamW [24] with $\beta_1 = 0.9$, $\beta_2 = 0.999$, and weight decay $= 1 \times 10^{-4}$. The batch size is 4. The patch size in training is set to 128×128 on Rain200H and Rain200L [42], whereas it is set to 256×256 on the other datasets. We apply random cropping along with horizontal and vertical flipping for data augmentation.

Evaluation metrics. Model performance is evaluated using peak signal-to-noise ratio (PSNR) [16] and structural similarity (SSIM) [40]. Following previous studies [2,41,44], PSNR and SSIM are measured based on the luminance channel Y of the YCbCr color space.

4.2   Comparison with state-of-the-art methods

Compared Methods. We compare our DFSSM with state-of-the-art single image deraining methods, including prior-based methods (DSC [25] and GMM [20]), CNN-based methods (DDN [5], RESCAN [19], PReNet [30], MSPFN [17], RCD-Net [38], MPRNet [45], DualGCN [6], and SPDNet [43]), and Transformer-based methods (Restormer [44], IDT [41], and DRSformer [2]).

Quantitative Comparison. Tab. 1 reports the quantitative results in terms of PSNR and SSIM for single image deraining. Our DFSSM achieves the best results across all evaluation metrics for synthetic rain and real-world rain. In
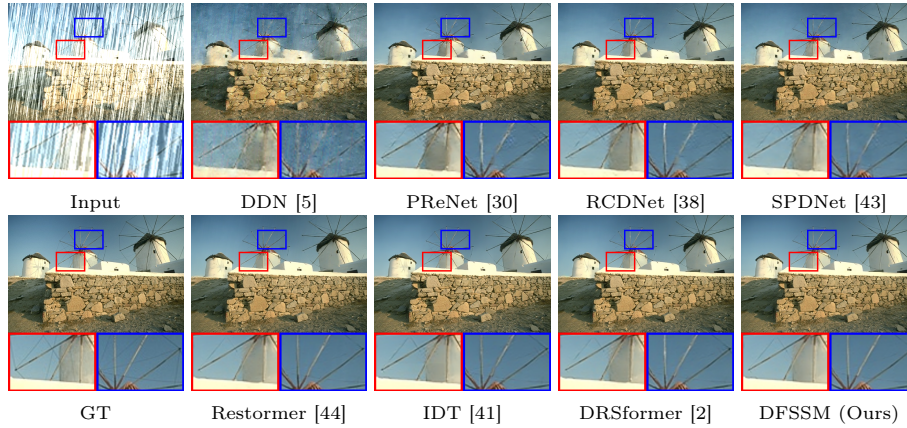
Fig. 3: Visual comparison on the synthetic rainy image of Rain200H [42]. Red and blue boxes correspond to the zoomed-in patches.

Table 2: Model efficiency comparison on the Rain200H [42]. FLOPs are calculated using a 256×256 resolution image.

Method	SPDNet [43]	Restormer [44]	IDT [41]	DRS-former [2]	**DFSSM-S**	**DFSSM**
PSNR↑ /SSIM↑	31.28/0.9207	32.00/0.9329	32.10/0.9344	32.17/0.9326	32.11/0.9313	32.99/0.9403
Parameters (M)↓	3.3	26.1	16.4	33.7	7.0	19.0
FLOPs (G)↓	89.3	156.3	73.5	244.6	87.9	212.4

particular, our DFSSM outperforms the state-of-the-art model DRSformer [2] in PSNR by 0.82dB on the Rain200H [42], 0.58dB on the Rain200L [42], 0.31dB on the DID-Data [48], 1.01dB on the SPA-Data [39], and 0.44dB on the LHP-Rain [12].

Qualitative Comparison. We conduct the qualitative comparison of deraining methods. Our proposed DFSSM achieves remarkable performance in synthetic and real-world rain scenarios. Fig. 3 visualizes the deraining results of the synthetic heavy rainy image. CNN-based methods, including DDN [5], PReNet [30], RCDNet [38], and SPDNet [43] fail to restore the background in rainy conditions, resulting in unnatural textures, particularly around windmills. Compared with Transformer-based IDT [41] and DRSformer [2], our proposed DFSSM generates a clearer image, especially in the area within the red box. As shown in Fig. 4, the results on real-world images indicate that our DFSSM achieves superior deraining outputs. Notably, our DFSSM most effectively removes the rain streaks in the sky within the red box, and the rain splashes on the ground within the blue box.

Model Efficiency Comparison. Tab. 2 depicts the model efficiency comparison with conventional high-performing methods [2, 41, 43, 44]. We calculate
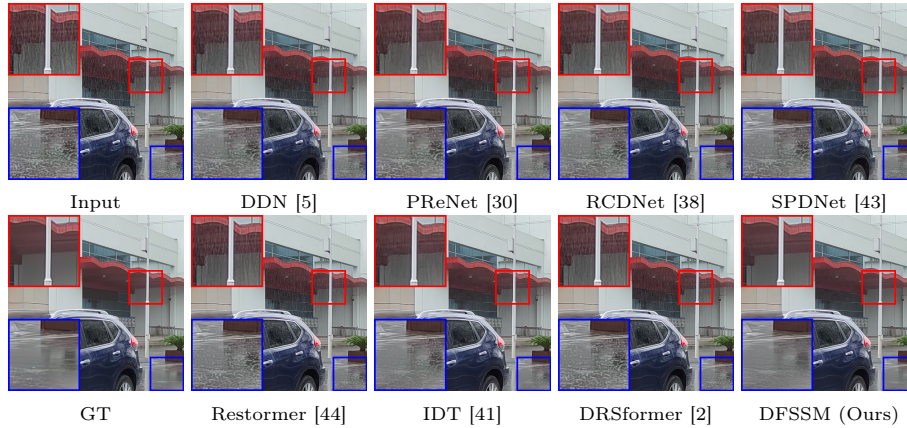
Fig. 4: Visual comparison on the synthetic rainy image of LHP-Rain [12]. Red and blue boxes correspond to the zoomed-in patches.

Table 3: Ablation study of the overall architecture.

Setting	FreqLoss	FSSB	MGCB	PSNR↑	SSIM↑	Param. (M)↓	FLOPs (G)↓
(a)				32.00	0.9273	17.1	198.3
(b)	✓			32.36	0.9323	17.1	198.3
(c)	✓	✓		32.87	0.9385	18.7	207.9
(d)	✓		✓	32.82	0.9384	17.4	202.8
(e)	✓	✓	✓	**32.99**	**0.9403**	19.0	212.4

FLOPs using a 256×256 resolution image. To compare computational efficiency, we build a lightweight DFSSM model, DFSSM-S. In the default DFSSM, the embedding dimension is set to $C = 48$, with the number of blocks per stage being $[N_S, N_F] = [1, 3]$. Meanwhile, in DFSSM-S, these hyperparameters are set to $C = 32$, $[N_S, N_F] = [1, 2]$. Although DFSSM demonstrates superior performance compared with existing methods, it entails high FLOPs. Conversely, DFSSM-S is an efficient model that achieves competitive performance with relatively fewer parameters and lower FLOPs.

4.3 Ablation study

In ablation studies, we train all models using the Rain200H [42] dataset under the same conditions.

Ablation study of the overall architecture. To evaluate the components of the proposed DFSSM, we conduct an ablation study across five settings. This result is presented in Tab. 3. We adopt MambaIR [10], a generalized and simple SSM, as the baseline. In setting (a), the baseline performs worse than deraining-specific Transformers in Tab. 1. The baseline uses only the L1-Loss

Table 4: Varying the Number of SSGs and FSSGs based on the setting (c) (Tab. 3). In the default setting, each stage comprises 1 SSG and 3 FSSGs.

SSG	FSSG	PSNR↑	SSIM↑	Param.(M)↓	FLOPs (G)↓
4	0	32.36	0.9323	17.1	198.3
3	1	32.55	0.9359	17.7	201.5
2	2	32.78	0.9379	18.2	204.7
1	3	**32.87**	0.9385	18.7	207.9
0	4	32.78	**0.9386**	19.2	211.1

Table 5: Ablation study of our Fast Fourier Transform Module (FFTM) based on the setting (c) (Tab. 3).

Method	PSNR ↑	SSIM↑
FFTM	**32.87**	**0.9385**
FFTM w/o FFT	32.58	0.9354
FFTM w/o 1×1 conv in spatial domain	32.33	0.9322

in its loss function, while setting (b) also uses Frequency Reconstruction Loss (FreqLoss) [27, 35], resulting in improved accuracy. In setting (c), integrating FSSB into the setting (b) to incorporate frequency domain processing enhances the efficacy of rain removal. Additionally, setting (d) alters the convolutional block in the baseline to the MGCB, enabling it to effectively address multi-scale rain degradations and manage the flow of information, thereby improving performance. Setting (e), including all these improvements, surpasses the baseline with a gain of 0.99 dB in PSNR and 0.013 in SSIM.

Varying the Number of SSG and FSSG. We develop the FSSB by incorporating FFTM in parallel with VSSM in the SSB. The SSG contains the SSB, while the FSSG contains the FSSB. Each encoder, decoder, and refinement stage has N_S SSGs and N_F FSSGs. Tab. 4 demonstrates the impact of varying the number of SSGs and FSSGs while maintaining their total count at 4. Increasing the number of FSSGs enhances performance, but increasing from 3 to 4 FSSGs does not yield further improvement. Thus, in the default setting, each stage comprises $N_S = 1$ SSG and $N_F = 3$ FSSGs.

Ablation study of FFTM. Tab. 5 presents the results of an ablation study concerning the configuration of the Fast Fourier Transform Module (FFTM). Initially, to verify the effectiveness of adopting FFT, we compare it against a module without FFT. The results indicate that removing FFT degrades performance, confirming that transforming and processing in the frequency domain via FFT is effective for image deraining. Likewise, removing the spatial domain processing by 1×1 convolution with SiLU activation function before the FFT and 1×1 convolution after the IFFT deteriorates performance. Thus, these spatial domain processes are necessary.

Table 6: Comparison of our Mixed-Scale Gated-Convolutional Block (MGCB) with previous convolutional blocks based on the setting (d) (Tab. 3). CA denotes the channel attention [14].

Method	CA	PSNR↑	SSIM↑	Param. (M)↓	FLOPs (G)↓
GDFN [44]		32.56	0.9345	17.2	201.3
MSFN [2]		32.67	0.9363	20.5	237.7
GDFN [44] + CA	✓	32.71	0.9376	17.3	201.5
MSFN [2] + CA	✓	32.61	0.9361	20.5	237.9
ConvLayer in MambaIR [10]	✓	32.36	0.9323	17.1	198.3
MGCB	✓	**32.82**	**0.9384**	17.4	202.8

Table 7: Comparison of State Space Model (SSM) [7] with Self-Attention (SA) [36] and Multi-Dconv Head Transposed Attention (MDTA) [44].

Method	Param. (M)	Cropped Images (128×128)			Original Images (321×421)		
		PSNR↑	SSIM↑	FLOPs (G)↓	PSNR↑	SSIM↑	FLOPs (G)↓
SA	26.1	**31.58**	0.9288	865.2	out of memory		
MDTA	26.3	31.01	0.9240	56.5	32.57	0.9356	551.4
SSM	**19.0**	31.43	**0.9292**	**53.1**	**32.99**	**0.9403**	**518.8**

Effectiveness of MGCB. We compare the Mixed-Scale Gated-Convolutional Block (MGCB) with existing convolutional blocks. In the baseline [10], the convolution layer (ConvLayer) comprises two 3×3 convolutions, an activation function between them, and the channel attention [14]. The Gated-Dconv Feed-Forward Network (GDFN) [44] comprises a gating mechanism and two 3×3 depth-wise convolutions. The Mixed-Scale Feed-Forward Network (MSFN) [2] incorporates two sets of 3×3 and 5×5 depth-wise convolutions, though it does not use a gating mechanism. Since GDFN and MSFN do not employ the channel attention, versions with the channel attention are also included in the comparison. As indicated in Tab. 6, the proposed MGCB outperforms all other methods, with slightly more parameters and FLOPs than GDFN [44] and ConvLayer in MambaIR [10], and 3.1M parameters and 35 GFLOPs fewer than MSFN [2]. In our MGCB, the mixed-scale convolutions and the gating mechanism work effectively.

Effectiveness of SSM. We evaluate the effectiveness of the SSM adopted in our DFSSM. We compare three methods with global receptive fields: State Space Model (SSM) [7], standard Self-Attention (SA) [36], and Multi-Dconv Head Transposed Attention (MDTA) [44], which calculates SA across channels rather than spatial dimension. Models, where the SSM in DFSSM is replaced with SA or MDTA, are trained under the same experimental conditions. As shown in Fig. 5, when the input image size increases, computational cost (FLOPs) of SA increases quadratically, while that of MDTA and SSM increases linearly. Tab. 7 shows the performance for both the 128×128 cropped images and the original size images. We use an Nvidia RTX A6000 (48GB GPU memory), but SA cannot process the original image size due to its high computational cost, resulting
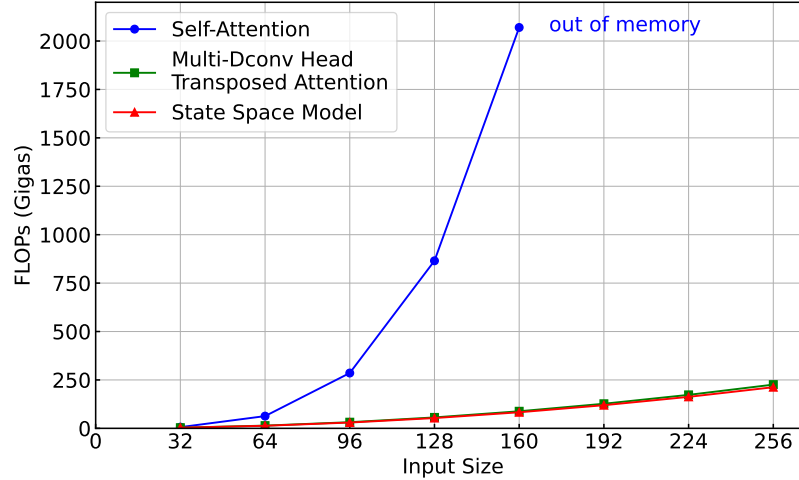
Fig. 5: Comparison of computational complexity among State Space Model (SSM) [7], standard Self-Attention (SA) [36], and multi-Dconv Head Transposed Attention (MDTA) [44]. FLOPs are measured for input image sizes ranging from 32×32 to 256×256. SA can not be measured beyond 192×192 due to being out of memory.

in out-of-memory. For 128×128 cropped images, SSM exhibits similar performance with only 6.1% lower computational costs compared with SA. Whether for cropped or original images, SSM achieves higher performance with slightly lower computational cost than MDTA.

5   Conclusion

In this paper, we present a novel Deraining Frequency-Enhanced State Space Model (DFSSM). To effectively remove rain degradations, we design a Frequency-Enhanced State Space Block (FSSB), which employs a State Space Block (SSB) to model long-range dependencies and a Fast Fourier Transform Module (FFTM) for frequency domain processing in parallel. Consequently, we propose a Mixed-Scale Gated-Convolutional Block (MGCB) to capture various scale rain degradations and control the flow of features. In synthetic and real-world rain scenarios, our proposed DFSSM performs favorably against state-of-the-art algorithms quantitatively and qualitatively.

References

1. Chen, S., Ye, T., Bai, J., Chen, E., Shi, J., Zhu, L.: Sparse sampling transformer with uncertainty-driven ranking for unified removal of raindrops and rain streaks. In: ICCV. pp. 13106–13117 (2023)

2. Chen, X., Li, H., Li, M., Pan, J.: Learning a sparse transformer network for effective image deraining. In: CVPR. pp. 5896–5905 (2023)
3. Chi, L., Jiang, B., Mu, Y.: Fast fourier convolution. NeurIPS **33**, 4479–4488 (2020)
4. Elfwing, S., Uchibe, E., Doya, K.: Sigmoid-weighted linear units for neural network function approximation in reinforcement learning. Neural networks **107**, 3–11 (2018)
5. Fu, X., Huang, J., Zeng, D., Huang, Y., Ding, X., Paisley, J.: Removing rain from single images via a deep detail network. In: CVPR. pp. 3855–3863 (2017)
6. Fu, X., Qi, Q., Zha, Z.J., Zhu, Y., Ding, X.: Rain streak removal via dual graph convolutional network. In: AAAI. pp. 1352–1360 (2021)
7. Gu, A., Dao, T.: Mamba: Linear-time sequence modeling with selective state spaces. arXiv preprint arXiv:2312.00752 (2023)
8. Gu, A., Goel, K., Re, C.: Efficiently modeling long sequences with structured state spaces. In: ICLR (2022)
9. Gu, S., Meng, D., Zuo, W., Zhang, L.: Joint convolutional analysis and synthesis sparse representation for single image layer separation. In: ICCV. pp. 1708–1716 (2017)
10. Guo, H., Li, J., Dai, T., Ouyang, Z., Ren, X., Xia, S.T.: Mambair: A simple baseline for image restoration with state-space model. In: ECCV (2024)
11. Guo, X., Fu, X., Zhou, M., Huang, Z., Peng, J., Zha, Z.J.: Exploring fourier prior for single image rain removal. In: IJCAI. pp. 935–941 (2022)
12. Guo, Y., Xiao, X., Chang, Y., Deng, S., Yan, L.: From sky to the ground: A large-scale benchmark and simple baseline towards real rain removal. In: ICCV. pp. 12097–12107 (2023)
13. Hendrycks, D., Gimpel, K.: Gaussian error linear units (gelus). arXiv preprint arXiv:1606.08415 (2016)
14. Hu, J., Shen, L., Sun, G.: Squeeze-and-excitation networks. In: CVPR. pp. 7132–7141 (2018)
15. Huang, T., Pei, X., You, S., Wang, F., Qian, C., Xu, C.: Localmamba: Visual state space model with windowed selective scan. arXiv preprint arXiv:2403.09338 (2024)
16. Huynh-Thu, Q., Ghanbari, M.: Scope of validity of psnr in image/video quality assessment. Electronics letters **44**(13), 800–801 (2008)
17. Jiang, K., Wang, Z., Yi, P., Chen, C., Huang, B., Luo, Y., Ma, J., Jiang, J.: Multi-scale progressive fusion network for single image deraining. In: CVPR. pp. 8346–8355 (2020)
18. Kang, L.W., Lin, C.W., Fu, Y.H.: Automatic single-image-based rain streaks removal via image decomposition. IEEE TIP **21**(4), 1742–1755 (2011)
19. Li, X., Wu, J., Lin, Z., Liu, H., Zha, H.: Recurrent squeeze-and-excitation context aggregation net for single image deraining. In: ECCV. pp. 254–269 (2018)
20. Li, Y., Tan, R.T., Guo, X., Lu, J., Brown, M.S.: Rain streak removal using layer priors. In: CVPR. pp. 2736–2744 (2016)
21. Liu, J., Yang, H., Zhou, H.Y., Xi, Y., Yu, L., Li, C., Liang, Y., Shi, G., Yu, Y., Zhang, S., et al.: Swin-umamba: Mamba-based unet with imagenet-based pretraining. In: MICCAI. pp. 615–625 (2024)
22. Liu, Y., Tian, Y., Zhao, Y., Yu, H., Xie, L., Wang, Y., Ye, Q., Liu, Y.: Vmamba: Visual state space model. arXiv preprint arXiv:2401.10166 (2024)
23. Loshchilov, I., Hutter, F.: SGDR: Stochastic gradient descent with warm restarts. In: ICLR (2017)
24. Loshchilov, I., Hutter, F.: Decoupled weight decay regularization. In: ICLR (2019)
25. Luo, Y., Xu, Y., Ji, H.: Removing rain from a single image via discriminative sparse coding. In: ICCV. pp. 3397–3405 (2015)

26. Ma, J., Li, F., Wang, B.: U-mamba: Enhancing long-range dependency for biomedical image segmentation. arXiv preprint arXiv:2401.04722 (2024)
27. Mao, X., Liu, Y., Liu, F., Li, Q., Shen, W., Wang, Y.: Intriguing findings of frequency selection for image deblurring. In: AAAI. pp. 1905–1913 (2023)
28. Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., et al.: Pytorch: An imperative style, high-performance deep learning library. NeurIPS **32** (2019)
29. Pei, X., Huang, T., Xu, C.: Efficientvmamba: Atrous selective scan for light weight visual mamba. arXiv preprint arXiv:2403.09977 (2024)
30. Ren, D., Zuo, W., Hu, Q., Zhu, P., Meng, D.: Progressive image deraining networks: A better and simpler baseline. In: CVPR. pp. 3937–3946 (2019)
31. Ronneberger, O., Fischer, P., Brox, T.: U-net: Convolutional networks for biomedical image segmentation. In: MICCAI. pp. 234–241 (2015)
32. Shi, W., Caballero, J., Huszár, F., Totz, J., Aitken, A.P., Bishop, R., Rueckert, D., Wang, Z.: Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network. In: CVPR. pp. 1874–1883 (2016)
33. Shi, Y., Xia, B., Jin, X., Wang, X., Zhao, T., Xia, X., Xiao, X., Yang, W.: Vmambair: Visual state space model for image restoration. arXiv preprint arXiv:2403.11423 (2024)
34. Song, T., Li, P., Jin, G., Jin, J., Fan, S., Chen, X.: Image deraining transformer with sparsity and frequency guidance. In: ICME. pp. 1889–1894 (2023)
35. Tu, Z., Talebi, H., Zhang, H., Yang, F., Milanfar, P., Bovik, A., Li, Y.: Maxim: Multi-axis mlp for image processing. In: CVPR. pp. 5769–5780 (2022)
36. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I.: Attention is all you need. NeurIPS **30** (2017)
37. Wang, C., Xing, X., Wu, Y., Su, Z., Chen, J.: Dcsfn: Deep cross-scale fusion network for single image rain removal. In: ACM MM. pp. 1643–1651 (2020)
38. Wang, H., Xie, Q., Zhao, Q., Meng, D.: A model-driven deep neural network for single image rain removal. In: CVPR. pp. 3103–3112 (2020)
39. 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: CVPR. pp. 12270–12279 (2019)
40. Wang, Z., Bovik, A.C., Sheikh, H.R., Simoncelli, E.P.: Image quality assessment: from error visibility to structural similarity. IEEE TIP **13**(4), 600–612 (2004)
41. Xiao, J., Fu, X., Liu, A., Wu, F., Zha, Z.J.: Image de-raining transformer. IEEE TPAMI **45**(11), 12978–12995 (2023)
42. Yang, W., Tan, R.T., Feng, J., Liu, J., Guo, Z., Yan, S.: Deep joint rain detection and removal from a single image. In: CVPR. pp. 1357–1366 (2017)
43. Yi, Q., Li, J., Dai, Q., Fang, F., Zhang, G., Zeng, T.: Structure-preserving deraining with residue channel prior guidance. In: ICCV. pp. 4238–4247 (2021)
44. Zamir, S.W., Arora, A., Khan, S., Hayat, M., Khan, F.S., Yang, M.H.: Restormer: Efficient transformer for high-resolution image restoration. In: CVPR. pp. 5728–5739 (2022)
45. Zamir, S.W., Arora, A., Khan, S., Hayat, M., Khan, F.S., Yang, M.H., Shao, L.: Multi-stage progressive image restoration. In: CVPR. pp. 14821–14831 (2021)
46. Zhang, D., Huang, F., Liu, S., Wang, X., Jin, Z.: Swinfir: Revisiting the swinir with fast fourier convolution and improved training for image super-resolution. arXiv preprint arXiv:2208.11247 (2022)
47. Zhang, H., Patel, V.M.: Convolutional sparse and low-rank coding-based rain streak removal. In: WACV. pp. 1259–1267 (2017)

48. Zhang, H., Patel, V.M.: Density-aware single image de-raining using a multi-stream dense network. In: CVPR. pp. 695–704 (2018)
49. Zhu, L., Liao, B., Zhang, Q., Wang, X., Liu, W., Wang, X.: Vision mamba: Efficient visual representation learning with bidirectional state space model. In: ICML (2024)