ClimaWeather: Enhancing Degraded Image Restoration by Selective Kernel Attention and Local Enhance Feed Forward Block via Transformer

Yemeng Wu[a†], Dawei Guan[b†], Xinping Lin[b], Zifeng Qiu[a,c], Huihui Bai[a,d*]
[a] School of Computer Science and Technology, Beijing Jiaotong University, Beijing, 100044, China
[b] School of Electronic and Information Engineering, Beijing Jiaotong University, Beijing, 100044, China
[c] Institute of Information Science, Beijing Jiaotong University, Beijing, 100044, China
[d] Tangshan Research Institute of Beijing Jiaotong University, Tangshan, 063000, China
† These authors contributed equally to this work and should be considered co-first authors.
22722108@bjtu.edu.cn, 22721006@bjtu.edu.cn, 22721018@bjtu.edu.cn, qzf93@qq.com, hhbai@bjtu.edu.cn

2024 IEEE 17th International Conference on Signal Processing (ICSP) | 979-8-3503-8738-4/24/$31.00 ©2024 IEEE | DOI: 10.1109/ICSP62129.2024.10846015

Abstract—**Efficient and high-quality restoration of degraded images is a challenging task that has attracted much attention in recent years. Image restoration network based on Transformer provides a standard paradigm for network training process. Recently, better multi-scene image restoration methods have been developed based on Multi-head attention mechanism and larger receptive field. However, the restoration quality is still affected by the low robustness of the method. In this study, we introduce ClimaWeather, which enhances the Transformer block's feature extraction by incorporating an efficient local feature extraction network, Local Enhance Feed Forward (LEFF) Block, and Select Kernel (SK) attention block. This method reduces the loss, improves the efficiency of feature extraction and the robustness of the model. We have demonstrated the superior performance of ClimaWeather through experiments.**

Index Terms—**Image Restoration, Transformer, Feed Forward Network, Select Kernel Attention**

I. Introduction

In recent years, restoration models in degraded images have attracted significant attention [1, 2, 3, 4, 5].Among them, Transformer-based generative model is in the topical areas. We input the degraded image into the trained generating network, with the output of the clean image. The goal of the task is to produce images that are as identical as possible to the impurity-free images. Nowadays, most of the methods focus on restoring degraded images in single scenes. For example, RainFormer [6] addresses removing rain, and WaveletFormerNet [7] is designed for fog removal. Due to the limited receptive field and lack of Multi-head attention module, the generative capability of these methods is significantly affected.

In addition to image restoration with Multi-head attention, Transformer offer global feature extraction capabilities with continuously developing receptive fields. Transformer extracts image features through the self-attention mechanism and the insertion of attention blocks in convolution layers, which improves the feature extraction capability, thus increasing the generation capability and realizing multi-scene image restoration. As a representative, TransWeather [4] is based on Transformer and multi-scene data set to train the process, and can achieve multi-scene degradation image restoration. Restormer [5] is also based on Transformer and enables noise removal for multiple scenarios.

However, TransWeather still has the problems such as poor generation ability and low robustness. To address these issues, we propose our model ClimaWeather, which focuses on optimizing the local feature extraction module (LEFF Block) in the Transformer block to improve the robustness of Transformer-based image restoration. In addition, we replace the Multi-head attention mechanism with SK attention [8], enhancing the feature extraction capability of the Transformer block. Compared to previous method [4, 5, 9], our method is dedicated to optimizing image feature extraction, which allows the network to extract more necessary information, improving the restoration results. Figure 1 is the summary of our approach.

Three main contributions are summarized as follows:

- We propose a feed forward network named LEFF that optimizes feature extraction, thereby improving the robustness of the method.
- We introduce SK attention module, which can dynamically adjust its receptive field based on the inputs and capture features at various scales.
- Detailed qualitative and quantitative evaluation demonstrated the usability and multi-scenario performance of our ClimaWeather method.

II. Related Works

The degraded image restoration network based on Transformer is designed to achieve pixel-level accuracy. TransWeather [4] is a method for restoring images, which based on Transformer architecture. In this work, we chose TransWeather as our baseline approach, and it consists of several components: Transformer Encoder, TransWeather Decoder, and Convolutional Projection Block.

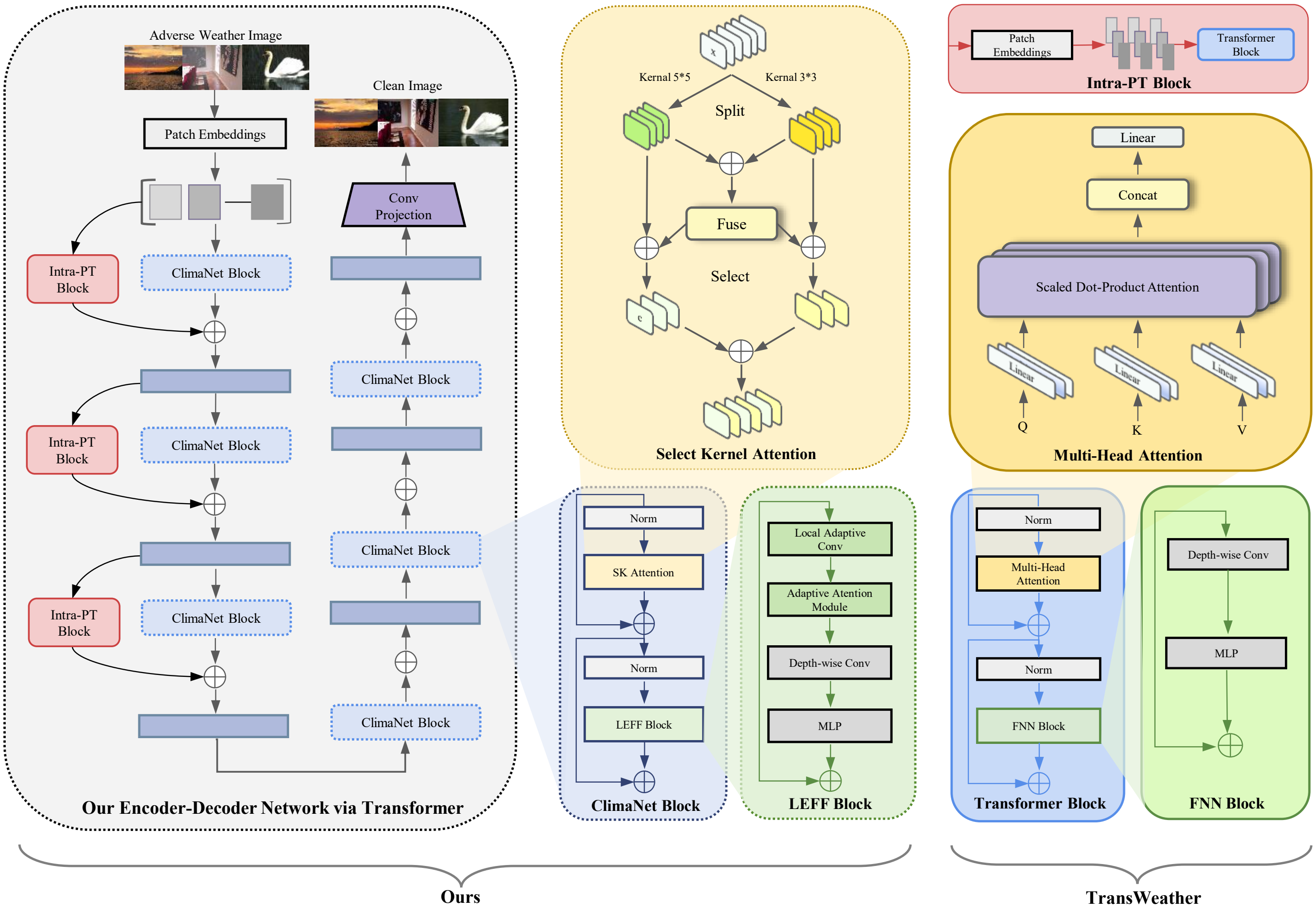

Figure 1: Overview of ClimaWeather, compared with TransWeather. Our network on the left uses specialized modules, ClimaNet with SK module and LEFF blocks. The network outperforms our baseline TransWeather on the right, which relies on standard Transformer blocks, FNN block, and Multi-head attention.

A. *Transformer Coder*

We extract multi-level features from the Transformer encoder to create a hierarchical feature representation of the input image. Both high-level and low-level feature extraction are made easier by the encoder's ability to extract features at various levels. We execute overlapped patch merging at each step. Before moving the features to the next step, we use this to combine overlapping feature patches to acquire features of the same size as non-overlapped patches. In Transformer Coders, it may include two important structures: Transformer Block and Intra-Patch Transformer Block, as shown in **Figure 1**.

Transformer Block: We employ feed forward networks and Multi-head self-attention layers in each Transformer block to compute the self-attention features. The following is a summary of the computation:

$$\mathbf{T}_i\left(\mathbf{I}_i\right) = FFN\left(MSA\left(\mathbf{I}_i\right) + \mathbf{I}_i\right) \tag{1}$$

When **I** is the input, ***T()*** stands for the Transformer block, $FFN()$ for the feed forward network block, $MSA()$ for Multi-head self-attention, and i for the encoder stage. The heads of queries (Q), keys (K), and values (V) have the same dimensions as the original self-attention network and are computed as follows:

$$\operatorname{Attn}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \operatorname{softmax}\left(\frac{\mathbf{Q}\mathbf{K}^T}{\sqrt{d}}\right)\mathbf{V} \tag{2}$$

where the dimension is denoted by d. Observe that each Transformer block use Multi-Head attention, and that number is a hyper-parameter that we change for every Transformer encoder step. since then, an FFN block receives the self-attention features. Here, the FFN block is slightly different from ViT [10]. The use of Depth-wise convolution aids in ring locality information and provides Transformers with positioning information. The following is a summary of the computation performed in the FFN block:

$$FFN_i\left(\mathbf{X}_i\right) = MLP\left(GELU\left(DWC\left(MLP\left(\mathbf{X}_i\right)\right)\right)\right) + \mathbf{X}_i \tag{3}$$

Depth-wise convolution is represented by DWC, while X stands for self-attention characteristics, GELU stands for

Gaussian error linear units [11], and MLP stands for multi-layer perception [12, 13, 14].

Due to the insufficient feature extraction capability of Multi-head attention in the face of complex scene degradation leading to poor repair results, we employ SK attention for optimisation to improve the robustness and feasibility of multi-scene repair. Meanwhile, due to the insufficient feature extraction capability of the depth image caused by the relatively simple structure of the feed-forward network, in order to solve the problem of feature loss and error in the process of restoration generation, we adopt a new feature extraction network as a part of the feed-forward network in order to improve the effectiveness of the network. Specific details are described in Chapter 3.

Intra-Patch Transformer Block: The Transformer encoder's Intra-patch Transformer (Intra-PT) blocks are integrated in between each stage. Sub-patches, which are created by breaking up the original patches into smaller pieces, are applied to these blocks. More detailed analysis is possible because each sub-patch is half the height and width of the main patch. The Intra-PT block optimizes processing within these smaller sub-patches by utilizing a Transformer design akin to that previously described. Because this design works on a more detailed scale, it is especially good at retrieving minute information, which is important for resolving small degradation in the data. The formulation of feed forward process in our Transformer encoder can be summarized as follows:

$$\mathbf{Y}_i = MT_i\left(\mathbf{X}_i\right) + \mathrm{IntraPT}_i\left(P\left(\mathbf{X}_i\right)\right) \tag{4}$$

B. Encoder and Decoder

Funnel-shaped structures consisting of encoders and decoders are usually represented and extracted due to features. The network is also composed of these two parts, as shown in Figure 1. In the encoder part, it is made up of alternating Transformer blocks and Intra-Patch Transformer blocks. Intra-Patch Transformer Block concatenate the output features of each Transformer Block part. In the decoder part, it is simply accumulated from Transformer blocks.

III. Methodology

Although encoder and decoder can implement multi-scale feature learning, we may loss some feature information in this process. To recover the details, our ClimaNet block contains a Selective Kernel (SK) attention module and a Local Enhance Feed Forward block (LEFF). Besides, we use a specific loss functions when training the network.

A. ClimaNet Blocks

Each ClimaNet Block uses SK attention and LEFF to capture local information and features across different scales. The SK attention module selects the specific features from multiple receptive fields. It handles various types of image degradation. The LEFF block use attention mechanism to maintain local structures. The design of the blocks represents a balance between local feature extraction and global information processing.

B. Select Kernel (SK) Attention

Since Multi-head attention is calculated based on fixed parts, SK attention mechanism can be self-adapted to image content information. To solve this problem, we use SK attention module to calculate the weights dynamically, and fuse the results of each kernel. This module has three steps: **Split**, **Fuse** and **Select**, as shown in Figure 1.

- **Split:** In this step, we use different kernels to perform convolutions, and get various features. For any given feature map $X \in \mathbb{R}^{H\times W\times C}$, two convolution operations are applied, $F : X \to \tilde{U} \in \mathbb{R}^{H\times W\times C}$ and $\hat{F} : X \to \hat{U} \in \mathbb{R}^{H\times W\times C}$, with kernel sizes of 3×3 and 5×5, respectively.
- **Fuse:** The results are first fused from multiple branches via an element-wise summation:

  $$U = \tilde{U} + \hat{U} \tag{5}$$

  Then we use global average pooling to generate channel-wise statistics as $s \in \mathbb{R}^{C}$ and calculate the c-th element by shrinking U through spatial dimensions $H \times W$:

  $$s_c = F_{gp}\left(U_c\right) = \frac{1}{H\times W}\sum_{i=1}^{H}\sum_{j=1}^{W} U_c(i,j) \tag{6}$$

  Next, we create a compact feature $z \in \mathbb{R}^{d\times 1}$ to guide the precise and adaptive selections:

  $$z = F_{fc}(s) = \delta(B(Ws)) \tag{7}$$

  where δ is the ReLU function, B denotes the Batch Normalization, and $W \in \mathbb{R}^{d\times C}$. To study the impact on the efficiency of the model, we use a reduction ratio r to control its value:

  $$d = \max\left(\frac{C}{r}, L\right) \tag{8}$$

  where L denotes the minimal value of d.
- **Select:** We use a soft attention across channels to select different spatial scales of information, which is guided by the compact feature descriptor z:

  $$a_c = \frac{e^{A_c z}}{e^{A_c z} + e^{B_c z}}, \quad b_c = \frac{e^{B_c z}}{e^{A_c z} + e^{B_c z}} \tag{9}$$

  where $A_c, B_c \in \mathbb{R}^{C\times d}$ and a, b denote the soft attention vectors for $\tilde{U}$ and $\hat{U}$, respectively. Note that $A_c \in \mathbb{R}^{1\times d}$ is the c-th row of A, and a_c is the c-th element of a, likewise for B_c and b_c. In the case of two branches, the matrix B is redundant because $a_c + b_c = 1$.
  The final feature map V is obtained through the attention weights on various kernels:

  $$V_c = a_c \cdot \tilde{U}_c + b_c \cdot \hat{U}_c \tag{10}$$

  where $V = \left[V_1, V_2, \ldots, V_C\right], \quad V_c \in \mathbb{R}^{H\times W}$.

Particularly, these three steps on all features have the same parameters, thus we select the feature from different scales dynamically.

C. Local Enhance Feed Forward (LEFF) Block

In each LEFF, we use Local Adaptive Convolution (LAC) to capture local information. Then we use Adaptive Attention Module (AAM) to enhance these local features by selectively focusing on the most important details and adjusting attention weights based on the input. Next, a Depth-wise Convolution (DWConv) processes these features, and a GELU activation function introduces non-linear transformations. To maintain the information, we choose Multi-Layer Perception (MLP) to map the features, and add a residual connection with the output back to the original input. The process is defined as:

$$\mathrm{LEFF}\left(X_i\right)=\mathrm{MLP}\left(\mathrm{GELU}\left(\mathrm{DWConv}\left(\mathrm{AAM}\left(\mathrm{LAC}\left(X_i\right)\right)\right)\right)\right)+X_i \tag{11}$$

LAC [15] is the Local Adaptive Convolution, AAM [16] is the Adaptive Attention Module, DWConv [17] represents the Depth-wise Convolution, GELU [11] is the activation function, MLP [18] is the Multi-Layer Perception, and X_i represents the input feature.

LEFF use Local Adaptive Convolution (LAC) layer to select local features. It can capture textures and small-scale structures. The AAM detects important features in local regions by adjusting attention weights based on input features. The DWConv, with the GELU activation function, processes the features in the previous stages by adding non-linear transformations. This step enhances the feature representation and capture more complex patterns that linear operations might miss. We choose GELU function to help us maintain gradient flow and reduce the risk of vanishing gradients. The MLP maps the processed features into a higher-dimensional space, and enhances the network's ability to learn complex pattern. Finally, We use residual connection to prevent feature degradation across layers and allow the network to learn deeper representations.

D. Loss Function

Our network uses a composite loss function with three components: reconstruction loss, content preservation loss, and edge consistency loss.

Reconstruction Loss (L_{rec}): This loss is the L1-norm of the pixel-wise difference between the restored image (I^{hq}) and the ground truth image (I^{gt}):

$$L_{\text{rec}}=\left\|I^{hq}-I^{gt}\right\| \tag{12}$$

Content Preservation Loss ($L_{content}$): This loss measures the difference between deep features of the restored and ground truth images, obtained from a pre-trained VGG19 network [19]:

$$L_{\text{content}}=\sum_l\left\|\phi_l\left(I^{hq}\right)-\phi_l\left(I^{gt}\right)\right\|^2 \tag{13}$$

where $\phi_l(\cdot)$ represents the feature map from the $l-th$ layer of VGG19.

Edge Consistency Loss (L_{edge}): This loss preserves edge sharpness and coherence. It is the $L1-norm$ of the difference between the edge maps of the restored and ground truth images, computed using the Sobel operator [20]:

$$L_{\text{edge}}=\left\|\mathrm{Sobel}\left(I^{hq}\right)-\mathrm{Sobel}\left(I^{gt}\right)\right\| \tag{14}$$

The total loss function combines these three losses, weighted by hyper-parameters α, β, and γ:

$$L_{\text{total}}=\alpha L_{\text{rec}}+\beta L_{\text{content}}+\gamma L_{\text{edge}} \tag{15}$$

IV. Experiment

In Chapter 4, we conduct experiments to verify the effectiveness of our method. We explain the training parameters, data sets, and experimental metrics. We also conducted qualitative and quantitative experiments.

A. Implementation Details

We conducted experiments using the Pytorch framework and trained the network using an Nvidia RTX3090 GPU. We used the AdamW optimiser [21] for optimal learning with a learning rate of 0.0002. We employed the learning rate scheduler [22], which restored the learning rate to 2 after 100 to 150 calendar elements. During training, the batch size was 32, with a total of 200 calendar elements.The size of the digital processor used in the Dataloader was 12.

B. Datasets and Evaluation Metrics

We use various types of degraded images in adverse weather conditions to train our network. For fair comparison, we followed the same dataset All-Weather as our baseline TransWeather. This dataset includes a total of 19,069 images from Snow100k [23], Raindrop [24], and Outdoor Rain [25]. These include a composite pattern of snow degradation, a true pattern of raindrops and a composite image of fog and rain. In this experiment, we used two test ,metrics, PSNR and SSIM. Structural Similarity Index (SSIM) is an index used to measure the similarity of two images. The calculation of SSIM consists of three main components: brightness (comparing the average brightness of the image), contrast (comparing the difference in pixel intensity), and structure (analyzing the similarity of the local structure of the image). The value of the SSIM is between 0 and 1, with 1 indicating that the two images are identical. Peak Signal-to-Noise Ratio (PSNR) is a commonly used image quality evaluation index, and we use it to measure the difference between the original image and the compressed or processed image [26]. PSNR is based on the mean square error (MSE) calculation of pixels, converting the difference measure between two images into a ratio on a logarithmic scale. Generally speaking, the higher the PSNR value, the higher the similarity between the two images, and the better the restoration of the image.

C. Qualitative Evaluation

We compare our method to the TransWeather model, with the results displayed in Figure 2. When processing photos, the TransWeather model fails to adequately address issues, such as yellowing and lightening caused by synthetic weather degradation. In contrast, our approach effectively reduces yellowing

Table 1: Quantitative Comparison on the Desnow, Derain&Dehazing and Raindrop Removal dataset based on PSNR↑ and SSIM↑.

	Derain&Dehaze			Desnow					Raindrop Removal		
	Method	Outdoor-Rain PSNR	Outdoor-Rain SSIM	Method	Snow100K-S PSNR	Snow100K-S SSIM	Snow100K-L PSNR	Snow100K-L SSIM	Method	RainDrop PSNR	RainDrop SSIM
Single Task Methods	pix2pix	18.97	0.711	DesnowGAN	30.05	0.823	23.55	0.79	pix2pix	27.89	0.853
	DRRN	20.48	0.751	DesnowNet	32.24	0.93	27.2	0.893	DuRN	31.22	0.924
	DID-MDN	26.3	0.901	GCA-Net	32.35	0.934	25.4	0.803	AttentiveGAN	31.5	0.926
	MPRNet	27.9	0.916	DDMSNet	34.4	0.945	28.9	0.876	MAXIM	31.75	0.927
	NAFNet	29.45	0.903	NAFNet	34.82	0.95	30.1	0.903	IDT	31.8	0.929
Multi Task Methods	Restormer	29.9	0.92	Restormer	35.95	0.958	30.3	0.908	Restormer	32.1	0.941
	All-in-One	24.7	0.898	All-in-One	34.15	0.945	28.4	0.883	All-in-One	31.1	0.926
	TransWeather	28.8	0.9	TransWeather	32.5	0.934	29.4	0.887	TransWeather	30.05	0.912
	ours	**30.39**	**0.922**	**ours**	**34.92**	**0.9503**	**31.33**	**0.9112**	**ours**	**32.72**	**0.938**

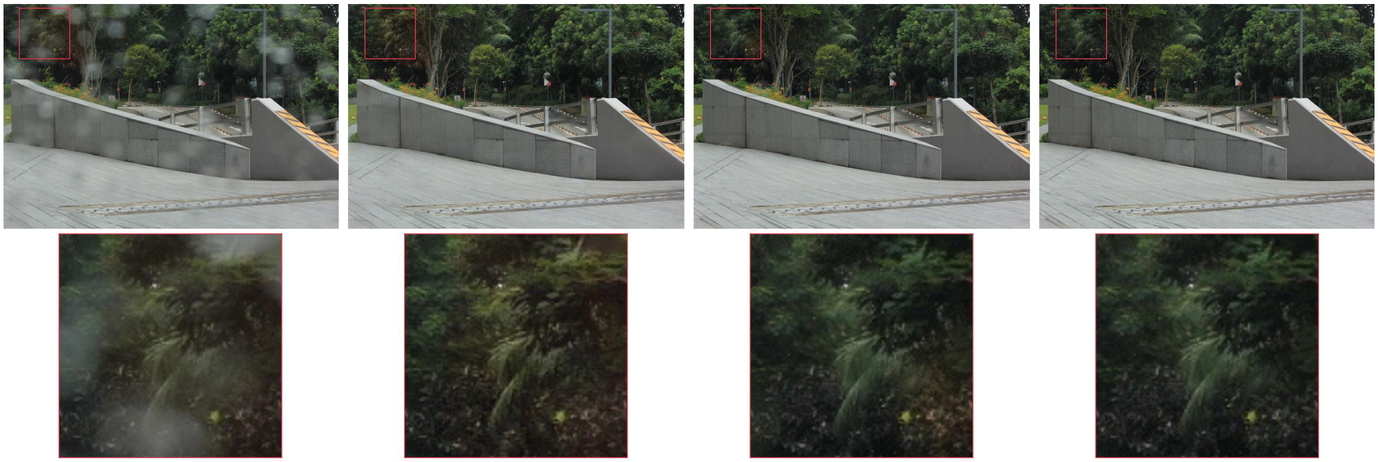
Input
TransWeather
Ours
Ground Truth

Figure 2: Results of qualitative experiments: comparison with baseline TransWeather

and lightening, showing that it significantly enhances image quality.

D. Quantitative Evaluation

We present a detailed and specific quantitative evaluation. In our research, we applied a series of methods to synthetic datasets, as shown in Table 1. We utilised some single degradation removal methods, including DRRN [27], DID-MDN [28] and MPRNet [29], NAFNet [30]. Additionally, we tested recent multiple degradation removal methods, such as Restormer [5], All-in-one [9] and our baseline TransWeather [4]. The comparison on these methods indicates that our method has an advantage on three different types of adverse weather conditions.

V. Conclusion

In this work, we propose ClimaWeather, a network designed to optimize Transformer-based multi-scene image restoration. Based on a single encoder-decoder network for recovery, we use the SK attention mechanism and LEFF block to optimize the feature extraction process. This approach allows the Transformer to more effectively recover weather degradations. We conduct extensive experiments on multiple synthetic datasets.

Acknowledgement

This work was supported in part by National Natural Science Foundation of China (No. 62331003, 62120106009), Beijing Natural Science Foundation (L223022), Natural Science Foundation of Hebei Province (F2024105029).

References

[1] Vineeth Murali and PV Sudeep. Image denoising using dncnn: An exploration study. In *Advances in Communication Systems and Networks: Select Proceedings of ComNet 2019*, pages 847–859. Springer, 2020.

[2] Kai Zhang, Wangmeng Zuo, and Lei Zhang. Ffdnet: Toward a fast and flexible solution for cnn-based image denoising. *IEEE Transactions on Image Processing*, 27(9):4608–4622, 2018.

[3] Shi Guo, Zifei Yan, Kai Zhang, Wangmeng Zuo, and Lei Zhang. Toward convolutional blind denoising of real photographs. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1712–1722, 2019.

[4] Jeya Maria Jose Valanarasu, Rajeev Yasarla, and Vishal M Patel. Transweather: Transformer-based

restoration of images degraded by adverse weather conditions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2353–2363, 2022.

[5] Syed Waqas Zamir, Aditya Arora, Salman Khan, Munawar Hayat, Fahad Shahbaz Khan, and Ming-Hsuan Yang. Restormer: Efficient transformer for high-resolution image restoration. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5728–5739, 2022.

[6] Hao Yang, Dongming Zhou, Jinde Cao, Qian Zhao, and Miao Li. Rainformer: a pyramid transformer for single image deraining. *The Journal of Supercomputing*, 79(6):6115–6140, 2023.

[7] Shengli Zhang, Zhiyong Tao, and Sen Lin. Waveletformernet: A transformer-based wavelet network for real-world non-homogeneous and dense fog removal. *Image and Vision Computing*, 146:105014, 2024.

[8] Xiang Li, Wenhai Wang, Xiaolin Hu, and Jian Yang. Selective kernel networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 510–519, 2019.

[9] Ruoteng Li, Robby T Tan, and Loong-Fah Cheong. All in one bad weather removal using architectural search. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 3175–3185, 2020.

[10] Hongxu Yin, Arash Vahdat, Jose M Alvarez, Arun Mallya, Jan Kautz, and Pavlo Molchanov. A-vit: Adaptive tokens for efficient vision transformer. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10809–10818, 2022.

[11] Dan Hendrycks and Kevin Gimpel. Gaussian error linear units (gelus). *arXiv preprint arXiv:1606.08415*, 2016.

[12] Yawei Li, Kai Zhang, Jiezhang Cao, Radu Timofte, and Luc Van Gool. Localvit: Bringing locality to vision transformers. *arXiv preprint arXiv:2104.05707*, 2021.

[13] Haiping Wu, Bin Xiao, Noel Codella, Mengchen Liu, Xiyang Dai, Lu Yuan, and Lei Zhang. Cvt: Introducing convolutions to vision transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 22–31, 2021.

[14] Enze Xie, Wenhai Wang, Zhiding Yu, Anima Anandkumar, Jose M Alvarez, and Ping Luo. Segformer: Simple and efficient design for semantic segmentation with transformers. *Advances in neural information processing systems*, 34:12077–12090, 2021.

[15] Zi-Rong Jin, Liang-Jian Deng, Tai-Xiang Jiang, and Tian-Jing Zhang. Laconv: Local adaptive convolution for image fusion. *arXiv preprint arXiv:2107.11617*, 2021.

[16] Zihang Jiang, Bingyi Kang, Kuangqi Zhou, and Jiashi Feng. Few-shot classification via adaptive attention. *arXiv preprint arXiv:2008.02465*, 2020.

[17] Yunhui Guo, Yandong Li, Liqiang Wang, and Tajana Rosing. Depthwise convolution is all you need for learning multiple visual domains. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 8368–8375, 2019.

[18] Marius-Constantin Popescu, Valentina E Balas, Liliana Perescu-Popescu, and Nikos Mastorakis. Multilayer perceptron and neural networks. *WSEAS Transactions on Circuits and Systems*, 8(7):579–588, 2009.

[19] Venkatesan Rajinikanth, Alex Noel Joseph Raj, Krishnan Palani Thanaraj, and Ganesh R Naik. A customized vgg19 network with concatenation of deep and handcrafted features for brain tumor detection. *Applied Sciences*, 10(10):3429, 2020.

[20] Nick Kanopoulos, Nagesh Vasanthavada, and Robert L Baker. Design of an image edge detection filter using the sobel operator. *IEEE Journal of solid-state circuits*, 23(2):358–367, 1988.

[21] Zhewei Yao, Amir Gholami, Sheng Shen, Mustafa Mustafa, Kurt Keutzer, and Michael Mahoney. Adahessian: An adaptive second order for machine learning. In *proceedings of the AAAI conference on artificial intelligence*, volume 35, pages 10665–10673, 2021.

[22] Diederik P Kingma. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[23] Yun-Fu Liu, Da-Wei Jaw, Shih-Chia Huang, and Jenq-Neng Hwang. Desnownet: Context-aware deep network for snow removal. *IEEE Transactions on Image Processing*, 27(6):3064–3073, 2018.

[24] Rui Qian, Robby T Tan, Wenhan Yang, Jiajun Su, and Jiaying Liu. Attentive generative adversarial network for raindrop removal from a single image. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2482–2491, 2018.

[25] Ruoteng Li, Loong-Fah Cheong, and Robby T Tan. Heavy rain image restoration: Integrating physics model and conditional adversarial learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1633–1642, 2019.

[26] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612, 2004.

[27] Ying Tai, Jian Yang, and Xiaoming Liu. Image super-resolution via deep recursive residual network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3147–3155, 2017.

[28] He Zhang and Vishal M Patel. Density-aware single image de-raining using a multi-stream dense network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 695–704, 2018.

[29] Armin Mehri, Parichehr B Ardakani, and Angel D Sappa. Mprnet: Multi-path residual network for lightweight image super resolution. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 2704–2713, 2021.

[30] Liangyu Chen, Xiaojie Chu, Xiangyu Zhang, and Jian Sun. Simple baselines for image restoration. In *European conference on computer vision*, pages 17–33. Springer, 2022.