mathematics

MDPI

Article

SGNet: Efficient Snow Removal Deep Network with a Global Windowing Transformer

Lie Shan [1], Haoxiang Zhang [2] and Bodong Cheng [3,*]

[1] Anhui Communications Vocational & Technical College, Hefei 230051, China; shan.lie@foxmail.com
[2] Shanghai University, Shanghai 200444, China; isaacpfino@gmail.com
[3] East China Normal University, Shanghai 200062, China
* Correspondence: 52275901029@stu.ecnu.edu.cn

Abstract: Image restoration under adverse weather conditions poses a challenging task. Previous research efforts have predominantly focused on eliminating rain and fog phenomena from images. However, snow, being another common atmospheric occurrence, also significantly impacts advanced computer vision tasks such as object detection and semantic segmentation. Recently, there has been a surge of methods specifically targeting snow removal, with the majority employing visual Transformers as the backbone network to enhance restoration effectiveness. Nevertheless, due to the quadratic computations required by Transformers to model long-range dependencies, this significantly escalates the time and space consumption of deep learning models. To address this issue, this paper proposes an efficient snow removal Transformer with a global windowing network (SGNet). This method forgoes the localized windowing strategy of previous visual Transformers, opting instead to partition the image into multiple low-resolution subimages containing global information using wavelet sampling, thereby ensuring higher performance while reducing computational overhead. Extensive experimentation demonstrates that our approach achieves outstanding performance across a wide range of benchmark datasets and can rival methods employing CNNs in terms of computational cost.

Keywords: image snow removal; transformer; efficient network; deep learning

MSC: 68U10

Citation: Shan, L.; Zhang, H.; Cheng, B. SGNet: Efficient Snow Removal Deep Network with a Global Windowing Transformer. *Mathematics* **2024**, *12*, 1424. https://doi.org/10.3390/math12101424

Academic Editor: Jakub Nalepa

Received: 8 April 2024
Revised: 30 April 2024
Accepted: 6 May 2024
Published: 7 May 2024

1. Introduction

With the widespread application of digital images in various fields, improving image quality has become one of the significant topics in both research and engineering domains. However, images captured in natural environments are often influenced by various weather conditions, including the presence of snow. Snow coverage not only affects the visibility and quality of images but may also have adverse effects on image processing and analysis. Therefore, snow removal from images has become a challenging yet important task aimed at recovering clear information from images affected by snow coverage.

Currently, snow removal from images has emerged as a research hotspot in the fields of computer vision and image processing. Numerous algorithms and techniques have been proposed to address this issue. Earlier, researchers proposed many methods involving manual feature extraction [1–5]. However, these methods mostly rely on human intuition and have a limited capacity to learn deep image features, resulting in suboptimal snow removal performance and weak generalization ability. In recent years, with the powerful feature extraction capability of Convolutional Neural Networks (CNNs), an increasing number of CNN-based snow removal methods have been proposed [6–8]. Among them, Liu et al. [6] introduced the first CNN-based snow removal method, named DesnowNet, which generates snow-free images by sequentially processing complex translucent and opaque snow particles. Chen et al. [7] treated the elimination of occlusion effects as part of the snow removal process and restored the images. To better guide the model

in learning snow particle information, they paid more attention to heavy snowfall. Jaw et al. [9] proposed JSTASR, a modular snow removal network with Generative Adversarial Networks (GANs) to enhance performance, but it has a large number of parameters. Chen et al. [8] proposed HDCWNet based on dual-tree complex wavelet transform (DTCWT) and discrepancy channel loss.

Despite the promising results achieved by recent Transformer-based snow removal methods, the quadratic computational complexity of self-attention modules in modeling long-range dependencies leads to a significant increase in inference time and memory consumption. This poses a major challenge for practical applications of these models, especially on resource-constrained devices. To address this limitation, we propose a novel efficient snow removal Transformer architecture that incorporates global windowing and wavelet sampling. By partitioning the image into multiple lower-resolution subimages while preserving global contextual information, our approach strikes a balance between performance and computational efficiency. The main contributions of this work can be summarized as follows:

(1) We introduce an efficient visual Transformer method with global windowing for reconstructing snow-affected images. This method ingeniously decomposes various information in the image into multiple low-resolution subimages containing global information using wavelet sampling.

(2) We propose a global information-sharing Transformer unit capable of interacting information among all subimages through parameter sharing. This approach breaks the information isolation issue present in previous localized windowing Transformers.

(3) We demonstrate the effectiveness of global windowing in image restoration, significantly alleviating the drawbacks of slow inference speed and high GPU memory consumption associated with Transformers in image restoration tasks. Extensive experimentation validates the superior performance and robustness of our method across various scenarios, indicating its high potential for practical applications.

2. Related Works

2.1. Single-Image Snow Removal

As another representation of atmospheric phenomena, the changes of snow are more complex, and the spatial states are more abundant. According to previous work [8], images affected by snow can be modeled as:

$$I(x) = K(x)T(x) + A(x)(1 - T(x)), \tag{1}$$

where $I(x)$ denotes the snowy image, $T(x)$ is the media transmission, and $A(x)$ is the atmospheric light. Meanwhile, $K(x)$ represents a snow scene image without veiling effect, which can be obtained by the following formula:

$$K(x) = J(x)(1 - Z(x)R(x)) + C(x)Z(x)R(x), \tag{2}$$

where $J(x)$ is the scene radiance, $R(x)$ is a binary mask that presents the snow location information, and $C(x)$ and $Z(x)$ are the chromatic aberration map for the snow image and the snow mask, respectively.

Traditional image snow removal methods still use feature priors to model snow particle information, such as histogram of gradients (HOG) [3,10], frequency separation [11], etc. Disappointingly, these methods cannot guarantee the generalization of snow removal and usually have poor performance. In order to further improve the ability of snow removal, ref. [6] proposed the first deep neural network-based snow removal method, called DesnowNet. DesnowNet adopts the multi-scale pyramid model of Inception-v4 as the backbone and performs well in its proposed Snow100K dataset. In [7], JSTASR is proposed to take into account the veiling effect and opaque snow particles. It removes the effects caused by snow phenomena using convolution and slightly darker channel priors. SMGARN [12] is a Snow Mask Guided Adaptive Residual Network, effectively removing

snow by accurately detecting snowflakes and guiding the snow removal process based on predicted snow masks.

The CNN-based methods discussed above have achieved significant progress in image snow removal. However, their performance is still limited by the inherent inductive biases of convolution operations, such as the local receptive field and spatial invariance. These properties make it difficult for CNNs to capture long-range dependencies and adapt to the diverse snow patterns in complex scenes.

2.2. Transformer-Based Methods

To model the dependency of pixel-level features, researchers began to pay attention to Transformer, which has made a difference in the Nature Language Process (NLP). The self-attention unit in Transformer can model the long-distance dependencies in the sequence. However, due to the particularity of the image, directly expanding it into a sequence as the input of the Transformer will cause excessive computational overhead. Expecting to solve this problem, ViT [13] uses the idea of dividing an image into multiple subimages of the same size. Later, to better promote the flow of information between subimages, Swin Transformer [14] introduced the idea of window displacement to indirectly model the entire image and demonstrated its excellent performance in high-level vision tasks such as image classification and target detection. Recently, some works refined the Transformers architecture to better-fit image restoration tasks, such as IPT [15], SwinIR [16], and Restormer [17]. Among them, IPT draws on the network structure of DERT [18], which uses 3×3 convolution with a step size of 3 to reduce the dimensionality of the image. This method can alleviate the dimensionality problem to a certain extent. Notably, the demanding requirements for GPU memory, training datasets, and reasoning time are unacceptable. SwinIR directly migrated the Swin Transformer to the IR task and achieved outstanding results. Despite its capabilities, SwinIR's stacking of numerous Transformer models also leads to prolonged execution times and remarkably high GPU memory usage. Restormer is an efficient Transformer model tailored for image restoration, mitigating the computational complexity through optimizing key design elements in multi-head attention and feed-forward network to capture long-range pixel interactions while remaining applicable to large images.

While Transformer enhances model performance, it incurs substantial GPU memory and time overhead, lacking inherent encoding of 2D image positional information which CNNs natively provide. Therefore, our goal is to explore and incorporate a more elegant and efficient Transformer for image restoration.

2.3. Wavelet Transform in Image Restoration

Wavelet is widely used in various computer vision tasks due to its effectiveness in capturing both spatial and frequency information of images. For example, ref. [19] proposed a novel Transformer for hyperspectral image classification. It unifies downsampling with wavelet transform to losslessly decompress feature maps, providing an efficient trade-off between performance and computation. Ref. [20] introduced a method for learning 3D shape representations using multiscale wavelet decomposition. Ref. [21] proposed a multiscale wavelet transformer framework for face forgery detection, which gradually aggregates multiscale wavelet representations from different stages of backbone networks and designs frequency-based spatial attention and cross-modal attention to better fuse frequency and spatial features through unified Transformer blocks for improved efficiency.

For image restoration, in HDCWNet [8], DTCWT is employed to locate high-frequency snow information in the image for better snow removal performance and mitigate color distortion, whereas wavelet transforms cannot accurately distinguish snow from other high-frequency components, thus potentially removing some useful details. Moreover, the changes in the subband information during wavelet upsampling inevitably affect the reconstructed image quality. Recently, Li et al. [22] proposed an Efficient Wavelet-Transformer (EWT) for single image denoising. Different from previous works, EWT integrates wavelet

transform into the Transformer architecture. By performing wavelet downsampling and upsampling, it can simultaneously exploit multi-scale and multi-frequence contextual information while reducing computational cost and GPU memory usage.

Inspired by the success of wavelet transforms in image restoration, we propose a global sampling module based on wavelet sampling in our architecture. This allows us to extract informative global features at reduced resolutions to lower the computational burden of self-attention modules.

3. Method

The architecture overview of our method is illustrated in Figure 1. It begins by decomposing the input RGB image (X_{input}) into multiple degraded subimages through a series of wavelet samplings, which differs from previous patch-based approaches:

$$\chi = DWT(X_{input}), \tag{3}$$

In the SGNet architecture, $DWT(\cdot)$ denotes the Discrete Wavelet Transform, and χ represents the subimage set obtained from the partition: $\chi = X_{LL}, X_{LH}, X_{HL}, X_{HH}$. Each subimage is treated as a "token", contrasting with prior methods where features of patches were set as concatenations of original pixel RGB values. In our implementation, we employ three stages of wavelet sampling, resulting in each image being downsampled to half its original size. To enhance representation, we apply several convolutional embedding layers to project the features sampled at each stage to an arbitrary dimension (denoted as C). Finally, the features are restored back to the pixel space through an inverse wavelet transform (IWT):

$$X_{output} = IWT(\chi'). \tag{4}$$

To generate hierarchical representations, as the network deepens, feature resolution is reduced through wavelet sampling layers. The first sampling layer downsamples the original image resolution using the discrete sampling method of the Haar wavelet, globally sampling the image with a stride of 1 (pixel intervals), and applying convolutional layers in the channel dimension.

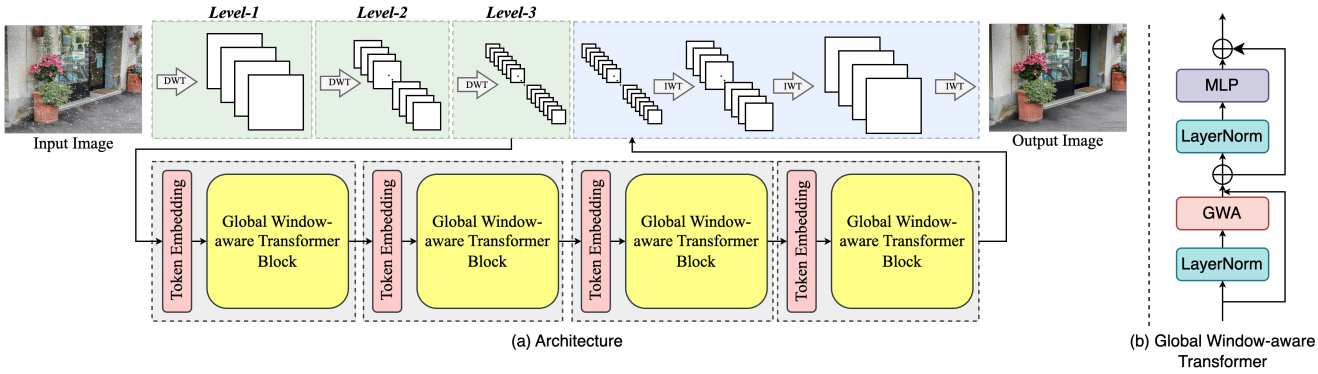

Figure 1. (**a**) Overview of the SGNet Structure. SGNet follows the architecture of a U-shaped network, with the distinction that in both the encoder and decoder, we introduce a global sampling approach instead of conventional pooling and convolution. The central portion of the network employs multi-stage Global Window-aware Transformer (GWT) blocks as the backbone for feature extraction. (**b**) Each Transformer block consists of multiple Transformer networks based on the channel dimension.

3.1. Global Window-Aware Transformer

The Global Window-aware Transformer (Figure 2b Lower) is constructed by replacing the standard Multi-head Self-Attention (MSA) module in the Transformer block with a module based on global windows while keeping other layers unchanged. As shown in Figure 1, this module consists of a multi-head self-attention module based on global

windows, followed by a two-layer MLP with GELU non-linear activation function. Layer Normalization (LN) layers are applied before each MSA module and each MLP, and residual connections are applied after each module. In GWT, the Attention mechanism (block GWA) operates individually on each subimage in χ. Assuming $\chi \in \mathbb{R}^{s^2 \times \frac{H}{s} \times \frac{W}{s} \times C}$, where s^2 is the number of subimages, H and W are the height and width of each subimage, and C is the channel dimension, then we have $\chi^i \in \mathbb{R}^{\frac{H}{s} \times \frac{W}{s} \times C}$ for each subimage χ^i. Following the operation rules of Attention, we obtain the computed features as follows:

$$\tilde{\chi}^i = Softmax(\chi^i_Q \cdot (\chi^i_K)^T)\chi^i_V, \tag{5}$$

where χ^i_Q, χ^i_K, and χ^i_V are the query, key, and value matrices computed from the input subimage χ^i, respectively. The subsequent GWT blocks follow the same computation process, enabling the model to capture long-range dependencies within each subimage.

In the past, CNNs were often relied upon to extract features from images, but their limited receptive fields cannot model the rich semantic information in images (Figure 2a). Although introducing Transformers to vision tasks has significantly addressed this issue, the enormous computational and storage requirements have led to current vision Transformers being constrained to use a window-based approach (Figure 2b Upper). The Local Window-aware Transformer originated from ViT [13]. ViT directly divides the image into fixed-size windows, then obtains patch embeddings through linear transformations, and feeds the patch embeddings into the Transformer for self-attention operations. Subsequently, Swin Transformer [14] adopts a rolling tensor approach to extract features across windows. However, the essence of these methods still involves dividing the entire image into several local regions for computation. The problem with this approach is the inability to break the information isolation between windows. Moreover, the standard Transformer architecture is designed to perform self-attention computation, where relationships between a token and all other tokens are computed. Such global computation results in quadratic complexity with respect to the number of tokens, making it unsuitable for visual problems involving high-resolution images, such as image restoration. To address this, our proposed global windowing approach can effectively solve the aforementioned problem by efficiently capturing global context while maintaining computational feasibility. To effectively model the global context, we consider constructing globally informative windows and computing self-attention within them. These windows are uniformly sampled from the image in a global manner. Assuming the original image size is $H \times W$, and the number of global samplings is s, then each window ultimately contains $\frac{H}{s} \times \frac{W}{s}$ patches. The computational complexity of the window-based module is:

$$\Omega(MSA) = 4HWC^2 + 2(HW)^2C \tag{6}$$

$$\Omega(G-MSA) = 4(\frac{HW}{s^2})C^2 + 2s^2C^3 \tag{7}$$

where C represents the channel dimension. The former is quadratic relative to the number of patches HW, while the latter, when s is fixed (typically set to 8), is linear. Global self-attention computation is typically impractical for large HW, whereas attention based on global windows is scalable, and computing attention across the channel dimension can further reduce computational costs.

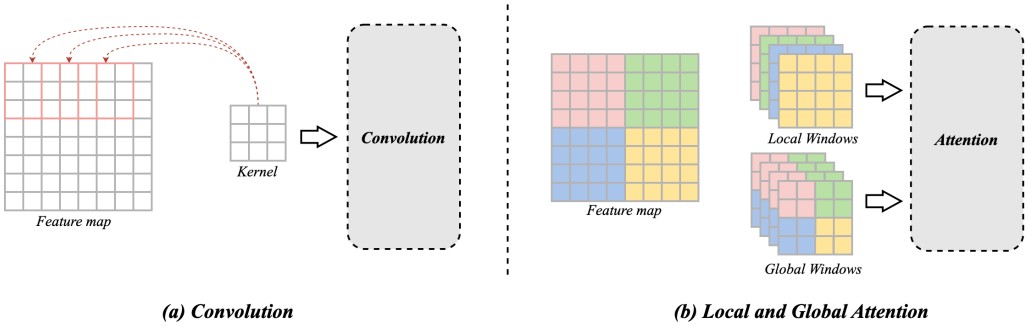

Figure 2. Schematic diagrams of CNN, traditional local window-aware visual Transformer, and global window-aware Transformer. In (**b**), "Local Windows" represents a schematic of the windowing concept in ViT [13], while "Global Windows" represents the strategy based on global sampling proposed in this paper. Global Windows provide the best summary of international information.

3.2. Global Sampling

The self-attention module based on local windows lacks connections across windows, limiting its modeling capability. To introduce cross-window connections while maintaining efficient computation with non-overlapping windows, Swin Transformer adopts a shifting window partitioning method, alternating between two partitioning configurations in consecutive Transformer blocks. While effective for visual tasks, such an approach inevitably disrupts information sharing across the global context. As shown in Figure 3, the global window utilizes a cross-pixel sampling strategy, such as uniformly sampling the 8×8 feature map into windows of size 4×4 starting from the top-left pixel.

As shown in Figure 3, each downsampling module utilizes a conventional window partitioning strategy, starting from the top-left pixel, dividing the 8×8 feature map into 4 windows of size $4 \times 4 (M = 4)$ each. Then, the next module employs the same sampling strategy to configure the windows, obtaining new window combinations by uniformly sampling the original features.

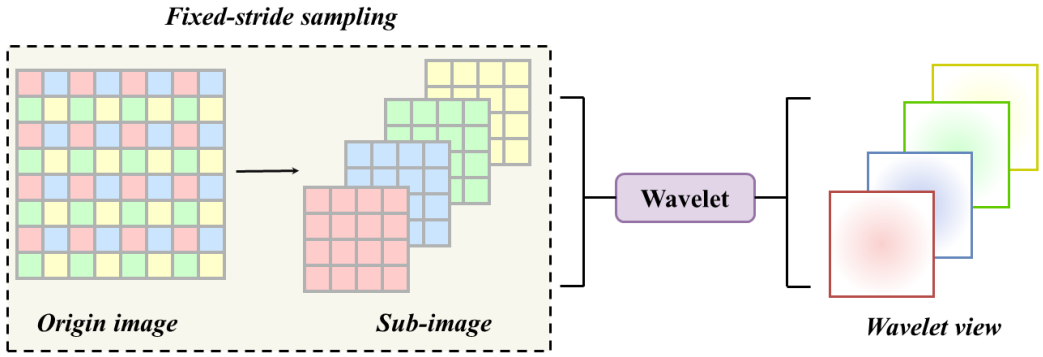

Figure 3. Schematic diagram of the efficient batch computation method of self-attention in the global window strategy.

Then, utilizing the computational method in the Haar wavelet transform, the operation to combine the four subimages I_1, I_2, I_3, I_4 is executed as follows:

$$\begin{bmatrix} I^1 \\ I^2 \\ I^3 \\ I^4 \end{bmatrix} = \frac{1}{2} \begin{bmatrix} 1 & 1 & 1 & 1 \\ -1 & -1 & 1 & 1 \\ -1 & 1 & -1 & 1 \\ 1 & -1 & -1 & 1 \end{bmatrix} \begin{bmatrix} I_1 \\ I_2 \\ I_3 \\ I_4 \end{bmatrix} \tag{8}$$

3.3. Implicit Positional Encoding

When computing self-attention, we typically follow [23] by adding a relative positional bias $B \in R^{M^2 \times M^2}$ for each head during similarity calculation, ensuring that positional information of the image is not lost during the process of computing local windows:

$$Attention(Q, K, V) = Softmax(\frac{QK^T}{\sqrt{d}} + B)V, \quad (9)$$

where $Q, K, V \in R^{M^2 \times d}$ are the query, key, and value matrices, d is the query/key dimension, and M^2 is the number of patches in a window.

The method proposed in this paper no longer requires such encoding forms. Instead, this process is implicitly transferred to the global window, referred to as implicit positional encoding. Since global sampling disperses global information into the interiors of each window, it alleviates positional ambiguity caused by non-overlapping local sampling.

4. Experiment

In this section, we provide relevant experimental details, descriptions, and results to validate the effectiveness and superiority of the proposed method.

4.1. Datasets

In this work, we trained our SGNet with the CSD [8] train set (8000 images) for synthetic images and used the Snow100K [6] train set (50,000 images) to train the model for real images. For evaluation, we use three benchmark test sets, including SRRS [7], CSD [8], and Snow100K [6]. Specifically, we adopted the last 2000 images in the SRRS test set for evaluation. We followed the approach of a recent work, which chose the last two thousand images from the SRRS dataset as test cases. This choice was based on their representativeness of typical snow conditions, as the SRRS dataset contains many images with repetitive backgrounds. In contrast, the last 2000 images cover a variety of snow densities and an ample number of scenes. For CSD, we use its 2000 test images for verification. Then, we use the three test sets Snow100K-S, Snow100K-M, and Snow100K-L provided by Snow100K for evaluation. Furthermore, we also conduct validation on two additional image snow removal datasets (Cityscape [24] and Kitti [24]) to fully demonstrate the effectiveness of SGNet.

4.2. Implementation Detail

Training Details

Following previous works, we randomly extract 16 snowy patches with a size of 128×128 as inputs. Meanwhile, we augment the training data with flips and rotates operations to further improve the generalization ability of the model. The learning rate is initialized to 10^{-4} and halved every 100 epochs. We implement our model with the PyTorch framework and update it with the Adam optimizer.

4.3. Comparisons with State-of-the-Art Methods

We first evaluate our approach on five synthetic snow removal image datasets and compare it with state-of-the-art methods. All methods are trained using the same configuration.

4.3.1. Quantitative Comparison

Table 1 presents the quantitative results of our proposed method (SGNet) and other methods on three commonly used benchmark datasets. The upper portion of the table includes results for image restoration methods specific to particular weather conditions. SGNet demonstrates reliable performance across all benchmark datasets (i.e., 33.90 on Snow100K-S and 29.56 on Snow100K-L). Although Restormer achieves results close to or even higher than SGNet on Snow100K-M, it is noteworthy that our method achieves a PSNR improvement of 0.11 dB over the previous best method on the most challenging

Snow100K-L dataset. Furthermore, while DS-GAN's performance on the CSD dataset is slightly higher than SGNet, from the perspective of average performance, SGNet still holds an absolute advantage.

Table 1. Quantitative comparison of image desnowing on Snow100K (-S, -M, and -L), SRRS, and CSD datasets (**PSNR (dB)**). Snow100K (-S, -M, and -L) are abbreviated S100K (-S, -M, and -L), respectively. The best results are **bold**, and the second best results are underlined.

Method	S100K-S	S100K-M	S100K-L	CSD	SRRS	Average
DesnowNet [6]	32.23	30.68	28.79	28.39	24.41	28.90
All-in-One [25]	32.73	31.82	28.99	28.88	24.63	29.41
TransWeather [26]	32.60	31.22	29.14	29.36	25.01	29.46
Restormer [17]	33.89	**32.42**	29.45	29.21	24.79	29.95
DS-GAN [9]	33.43	31.87	28.06	**29.38**	25.10	29.56
DDMSNET [24]	31.46	31.35	28.85	29.25	24.77	29.13
SGNet (Ours)	**33.90**	32.38	**29.56**	29.27	**25.31**	**30.08**

Table 2 presents the quantitative results of our proposed method (SGNet) and other methods on three commonly used benchmark datasets using no-reference metrics, including NIQE, NRQM, and PIQE. The three values in each column of Table 2 correspond to NIQE, NRQM, and PIQE. NIQE aims to assess the naturalness of images, indicating how much they resemble natural scenes, with lower scores being better; NRQM evaluates local frequency domain features, global frequency domain features, and spatial features comprehensively, with higher values being better; PIQE divides test images into non-overlapping blocks and analyzes the distortion of blocks, with smaller values being better. Our SGNet consistently achieves the best performance across all the datasets regarding the three metrics. On the most challenging Snow100K-L dataset, SGNet outperforms other methods by a large margin, demonstrating its superior capability in handling heavy snow images. These no-reference evaluation results further validate the effectiveness of our proposed method in generating high-quality snow-free images.

Additionally, Table 3 presents the quantitative results on the Cityscape and Kitti datasets. On the Cityscape dataset, SGNet achieves a PSNR gain of 0.14 dB compared to the second-best method. Similarly, on the Kitti dataset, SGNet also achieves the highest performance scores. In summary, our proposed method demonstrates competitive performance across all datasets and notably outperforms state-of-the-art methods in terms of average scores, showcasing a significant advantage. It is worth emphasizing that both TransWeather and Restormer employ a local window approach. The results in Tables 1–3 demonstrate the significant advantages of SGNet over these two methods in terms of performance scores. This highlights the superiority and feasibility of the global window strategy.

Table 2. Quantitative comparison of image desnowing on Snow100K (-S, -M, and -L), SRRS, and CSD datasets (No-Reference Metric: **NIQE↓, NRQM↑, and PIQE↓**). Snow100K (-S, -M, and -L) are abbreviated S100K (-S, -M, and -L), respectively. The best results are **bold**, and the second best results are underlined.

Method	S100K-S	S100K-M	S100K-L	CSD	SRRS
DesnowNet [6]	4.31/3.57/14.30	6.78/2.86/18.32	9.43/2.57/20.24	8.25/3.86/11.20	6.92/5.81/14.79
All-in-One [25]	2.96/6.02/10.88	3.45/4.15/14.65	5.56/3.41/15.66	6.74/5.12/10.65	4.74/9.02/12.61
TransWeather [26]	2.43/6.78/10.94	3.64/4.22/13.90	5.82/4.89/15.12	6.30/5.37/9.13	4.36/8.76/12.38
Restormer [17]	3.12/**7.04**/9.81	4.58/4.52/13.98	5.07/4.70/14.56	5.82/5.69/10.42	5.28/9.31/11.07
DS-GAN [9]	2.58/5.23/9.65	2.90/3.93/13.74	4.77/4.38/13.71	5.71/**7.01**/8.56	4.07/9.77/12.56
DDMSNET [24]	2.65/6.14/9.04	3.35/4.06/12.07	4.81/4.72/12.13	6.08/5.83/8.92	4.16/10.34/10.89
SGNet (Ours)	**2.17**/6.90/**8.63**	**2.77**/**4.87**/**11.72**	**4.53**/**5.01**/**11.09**	**5.25**/6.58/**8.03**	**3.70**/**10.61**/**10.72**

Table 3. Quantitative comparison of image desnowing on Cityscape and Kitti datasets (**PSNR(dB)** and **SSIM**). The best results are **bold** , and the second best results are underlined.

Method	Cityscape	Kitti
DesnowNet [6]	31.74	31.51
All-in-one [25]	32.88	32.08
TransWeather [26]	32.61	31.97
Restormer [17]	33.36	33.09
DS-GAN [9]	33.15	32.61
SMGARN [12]	33.42	32.76
DDMSNET [24]	33.21	32.49
SGNet (Ours)	**33.56**	**33.12**

4.3.2. Visual Comparison

In Figure 4, we provide visual comparisons with other state-of-the-art models on Snow100K [6], CSD [8], and SRRS [7]. According to Figure 4, we can clearly observe that SGNet can reconstruct high-quality snow-free images very close to the ground truth (GT). We also visually compare with current state-of-the-art methods, and from the figures, it can be observed that both methods are visually very close, with our method even showing fewer residual snow remnants in certain areas. In Figure 4, it can be seen that the snow-free images reconstructed by SGNet are clearer and have fewer snow residues. These results indicate that SGNet effectively eliminates snow streaks and fine snowflakes' interference in the image, and it is more effective in removing large snowflakes without generating large areas of shadow. This suggests that our method achieves visually more appealing results compared to other baseline methods.

Additionally, in Figure 5, we showcase the snow removal capabilities of our method in real-world scenarios to validate the generalization ability of the proposed approach (please zoom in to view details). It can be clearly seen from the figure that SGNet can effectively remove large snow particles and dense snowflakes. This is attributed to the proposed global windowing mechanism, which accurately predicts the position and shape of the snow. With the predicted snow mask, the images reconstructed by SGNet have almost no snow and better preserve texture details.

Figure 4. Visual comparison with other advanced models. Obviously, our proposed method can reconstruct high-quality snow-free images.

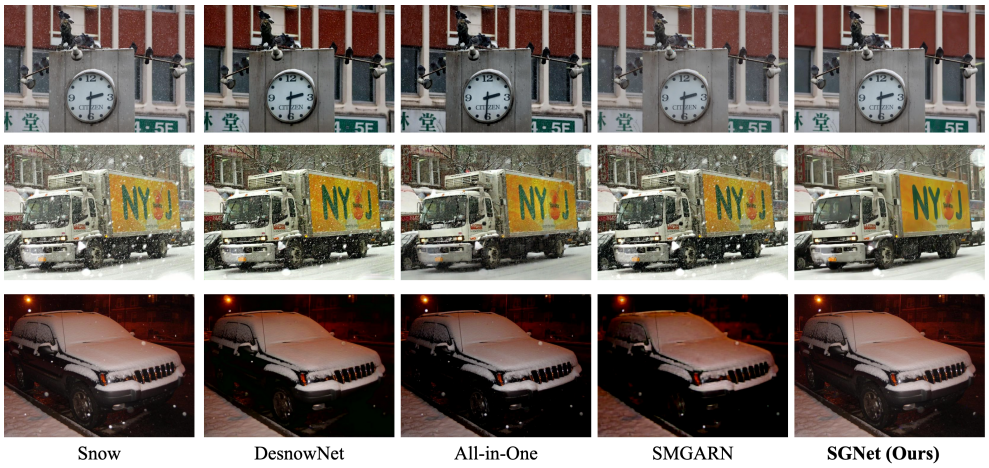

Figure 5. Visual comparison on real snow images from Snow100K. Obviously, our proposed method can reconstruct clear images with less snow residue.

4.4. Investigation

4.4.1. Model Size and Performance Comparison

Table 4 presents a comparison of model size, memory consumption, processing time, and PSNR performance among various snow removal methods. SGNet stands out as it achieves a more balanced trade-off between performance and resource consumption. In terms of model size, SGNet (12.64 M parameters) is notably smaller compared to several other methods, such as Restormer (16.98 M parameters) and TransWeather (21.90 M parameters). This indicates that SGNet requires fewer parameters while still delivering competitive performance. Regarding memory consumption, SGNet (14,682 MiB) demonstrates efficiency comparable to other methods, such as DesnowNet and DDMSNET, despite its superior performance in terms of PSNR. This suggests that SGNet achieves a more optimal utilization of memory resources. In terms of processing time, SGNet (0.39 s) exhibits a moderate processing speed, which is slightly faster than Restormer (1.95 s) and DS-GAN (0.77 s), while still maintaining high-quality results. This indicates that SGNet is capable of achieving efficient computation without compromising performance.

Overall, SGNet showcases a remarkable balance between model parameters, memory consumption, processing time, and PSNR performance. Its ability to achieve competitive performance while consuming fewer resources makes it a promising and practical solution for snow removal tasks.

Table 4. Model size and performance comparison. The results in the figure show that SGNet achieves a more perfect balance between model parameters and performance.

Method	Param (M)	Memory (MiB)	Time (s)	PSNR (dB)
DesnowNet [6]	15.60	12,076	0.32	28.79
All-in-one [25]	9.32	14,754	0.53	28.99
TransWeather [26]	21.90	17,627	0.81	29.14
Restormer [17]	16.98	22,910	1.95	29.45
DS-GAN [9]	13.51	15,370	0.77	28.06
DDMSNET [24]	10.26	11,923	0.28	28.85
SGNet (Ours)	12.64	14,682	0.39	29.56

4.4.2. Investigation of SGNet

In this section, we conducted an ablation analysis on the crucial design elements of the proposed SGNet by evaluating it on the Snow100K dataset.

Global Windows. The ablation results for the global window approach on Snow100K are presented in Table 5. SGNet employing global window partitioning outperforms models built on single window partitioning at each stage. Accuracy improves by +0.25 on the

heavily disturbed Snow100K-L dataset, with scores also increasing by +0.39 and +0.59 on Snow100K-S and -M, respectively. These findings indicate the effectiveness of leveraging global image information through global window construction.

Table 5. Ablation studies were conducted on SGNet's three benchmark tests to investigate the effects of the global window approach and different position embedding methods. Specifically, "w/o global" denotes that all self-attention modules utilize conventional window partitioning without global sampling. "abs. pos." refers to the absolute position embedding of the Vision Transformer (ViT), while "rel. pos." indicates the default setting using additional relative position bias terms.

Method	Snow100K-S	Snow100K-M	Snow100K-L
w/o global	33.51	31.79	29.31
global	**33.90**	**32.38**	**29.56**
no pos.	**33.90**	**32.38**	**29.56**
abs. pos.	33.77	32.26	29.52
abs. + rel. pos.	33.68	32.28	29.47
rel. pos.	33.70	32.31	29.48

Sampling methods. Table 6 presents the performance comparison of different sampling methods on the Snow100K dataset. Evidently, the approach utilizing global sampling achieves the highest PSNR scores across all three datasets: 33.90 on Snow100K-S, 32.38 on Snow100K-M, and 29.56 on Snow100K-L. This indicates that global sampling effectively captures comprehensive image information and contributes to enhanced restoration quality. In comparison to global sampling, methods employing max-pooling for sampling yield slightly lower PSNR scores, with scores of 33.47, 32.06, and 29.36 on Snow100K-S, Snow100K-M, and Snow100K-L, respectively. This suggests that max-pooling may not fully capture global context, leading to slightly inferior results. Similarly, methods employing average pooling for sampling also yield slightly lower PSNR scores compared to global sampling, with scores of 33.52, 32.14, and 29.45 on Snow100K-S, Snow100K-M, and Snow100K-L, respectively. This further emphasizes the superiority of global sampling in capturing comprehensive image information. Therefore, global sampling outperforms other sampling methods in enhancing restoration performance across all evaluated datasets.

Table 6. Comparison of different sampling methods on the Snow100K dataset.

Method	Snow100K-S	Snow100K-M	Snow100K-L
Global	**33.90**	**32.38**	**29.56**
Max-pooling	33.47	32.06	29.36
Avg-pooling	33.52	32.14	29.45

5. Discussion

In terms of limitations, we employed wavelet transform as a sampling method, which requires the input image dimensions to be powers of 2. To align images, we had to pad the input images, which to some extent can affect the distribution of image information. In future work, we plan to downsample images using an overlapping approach to address this. Additionally, SGNet adopts an end-to-end processing flow, delegating the extraction of image texture and color to the wavelet transform. This may result in some snow-related information not being fully covered. In our future work, we will focus on studying the extraction of weather degradation features.

6. Conclusions

This paper proposes an efficient snow removal Transformer method utilizing global windowing to address the challenge of snow removal from images. Distinguished from previous visual Transformers, our approach partitions the image into multiple low-resolution

subimages containing global information using wavelet sampling, aiming to reduce computational overhead while maintaining high performance. The key contributions include introducing an efficient visual Transformer method with global windowing, proposing a global information-sharing Transformer unit to address information isolation issues, and demonstrating the effectiveness of global windowing in image restoration through extensive experimentation. Extensive experiments demonstrate that SGNet achieves a processing time difference of only 0.1 s compared to CNN-based methods, while also realizing a 0.7 dB improvement in PSNR performance. Compared to Transformer-based methods, our approach achieves the best results across all performance metrics. In summary, our method offers a promising solution with improved efficiency and performance, paving the way for further advancements in image restoration and computer vision tasks.

Author Contributions: Methodology, L.S.; Investigation, B.C.; Writing—original draft, H.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: We will open-source the code for this project on GitHub, which will be accessible.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Xu, J.; Zhao, W.; Liu, P.; Tang, X. An improved guidance image based method to remove rain and snow in a single image. *Comput. Inf. Sci.* **2012**, *5*, 49. [CrossRef]
2. Zheng, X.; Liao, Y.; Guo, W.; Fu, X.; Ding, X. Single-image-based rain and snow removal using multi-guided filter. In Proceedings of the ICONIP, Daegu, Republic of Korea, 3–7 November 2013; pp. 258–265.
3. Pei, S.C.; Tsai, Y.T.; Lee, C.Y. Removing rain and snow in a single image using saturation and visibility features. In Proceedings of the ICME Workshop, Chengdu, China, 14–18 July 2014; pp. 1–6.
4. Wang, Y.; Liu, S.; Chen, C.; Zeng, B. A hierarchical approach for rain or snow removing in a single color image. *IEEE Trans. Image Process.* **2017**, *26*, 3936–3950. [CrossRef] [PubMed]
5. Yu, S.; Zhao, Y.; Mou, Y.; Wu, J.; Han, L.; Yang, X.; Zhao, B. Content-adaptive rain and snow removal algorithms for single image. In Proceedings of the International Symposium on Neural Networks, Macao, China, 28 November–1 December 2014; pp. 439–448.
6. Liu, Y.F.; Jaw, D.W.; Huang, S.C.; Hwang, J.N. DesnowNet: Context-aware deep network for snow removal. *IEEE Trans. Image Process.* **2018**, *27*, 3064–3073. [CrossRef] [PubMed]
7. Chen, W.T.; Fang, H.Y.; Ding, J.J.; Tsai, C.C.; Kuo, S.Y. JSTASR: Joint size and transparency-aware snow removal algorithm based on modified partial convolution and veiling effect removal. In Proceedings of the ECCV, Glasgow, UK, 23–28 August 2020; pp. 754–770.
8. Chen, W.T.; Fang, H.Y.; Hsieh, C.L.; Tsai, C.C.; Chen, I.; Ding, J.J.; Kuo, S.Y. ALL Snow Removed: Single Image Desnowing Algorithm Using Hierarchical Dual-Tree Complex Wavelet Representation and Contradict Channel Loss. In Proceedings of the ICCV, Montreal, QC, Canada, 11–17 October 2021; pp. 4196–4205.
9. Jaw, D.W.; Huang, S.C.; Kuo, S.Y. DesnowGAN: An efficient single image snow removal framework using cross-resolution lateral connection and GANs. *IEEE Trans. Circuits Syst. Video Technol.* **2020**, *31*, 1342–1350. [CrossRef]
10. Bossu, J.; Hautiere, N.; Tarel, J.P. Rain or snow detection in image sequences through use of a histogram of orientation of streaks. *Int. J. Comput. Vis.* **2011**, *93*, 348–367. [CrossRef]
11. Rajderkar, D.; Mohod, P. Removing snow from an image via image decomposition. In Proceedings of the ICECCN, Tirunelveli, India, 25–26 March 2013; pp. 576–579.
12. Cheng, B.; Li, J.; Chen, Y.; Zeng, T. Snow mask guided adaptive residual network for image snow removal. *Comput. Vis. Image Underst.* **2023**, *236*, 103819. [CrossRef]
13. Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S.; et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv* **2020**, arXiv:2010.11929.
14. Liu, Z.; Lin, Y.; Cao, Y.; Hu, H.; Wei, Y.; Zhang, Z.; Lin, S.; Guo, B. Swin transformer: Hierarchical vision transformer using shifted windows. In Proceedings of the ICCV, Montreal, QC, Canada, 11–17 October 2021; pp. 10012–10022.
15. Chen, H.; Wang, Y.; Guo, T.; Xu, C.; Deng, Y.; Liu, Z.; Ma, S.; Xu, C.; Xu, C.; Gao, W. Pre-trained image processing transformer. In Proceedings of the CVPR, Nashville, TN, USA, 20–25 June 2021; pp. 12299–12310.
16. Liang, J.; Cao, J.; Sun, G.; Zhang, K.; Van Gool, L.; Timofte, R. Swinir: Image restoration using swin transformer. In Proceedings of the ICCV, Montreal, QC, Canada, 11–17 October 2021; pp. 1833–1844.
17. Zamir, S.W.; Arora, A.; Khan, S.; Hayat, M.; Khan, F.S.; Yang, M.H. Restormer: Efficient transformer for high-resolution image restoration. In Proceedings of the CVPR, New Orleans, LA, USA, 18–24 June 2022; pp. 5728–5739.

18. Carion, N.; Massa, F.; Synnaeve, G.; Usunier, N.; Kirillov, A.; Zagoruyko, S. End-to-end object detection with transformers. In Proceedings of the ECCV, Glasgow, UK, 23–28 August 2020; pp. 213–229.
19. Ahmad, M.; Ghous, U.; Usama, M.; Mazzara, M. WaveFormer: Spectral–spatial wavelet transformer for hyperspectral image classification. *IEEE Geosci. Remote Sens. Lett.* **2024**, *21*, 5502405. [CrossRef]
20. Huang, H.; Fang, Y. Adaptive wavelet transformer network for 3d shape representation learning. In Proceedings of the International Conference on Learning Representations, Vienna, Austria, 3–7 May 2021.
21. Liu, J.; Wang, J.; Zhang, P.; Wang, C.; Xie, D.; Pu, S. Multi-scale wavelet transformer for face forgery detection. In Proceedings of the Asian Conference on Computer Vision, Macao, China, 4–8 December 2022; pp. 1858–1874.
22. Li, J.; Cheng, B.; Chen, Y.; Gao, G.; Zeng, T. EWT: Efficient Wavelet-Transformer for Single Image Denoising. *arXiv* **2023**, arXiv:2304.06274.
23. Raffel, C.; Shazeer, N.; Roberts, A.; Lee, K.; Narang, S.; Matena, M.; Zhou, Y.; Li, W.; Liu, P.J. Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Mach. Learn. Res.* **2020**, *21*, 1–67.
24. Zhang, K.; Li, R.; Yu, Y.; Luo, W.; Li, C. Deep dense multi-scale network for snow removal using semantic and depth priors. *IEEE Trans. Image Process.* **2021**, *30*, 7419–7431. [CrossRef] [PubMed]
25. Li, R.; Tan, R.T.; Cheong, L.F. All in one bad weather removal using architectural search. In Proceedings of the CVPR, Seattle, WA, USA, 13–19 June 2020; pp. 3175–3185.
26. Valanarasu, J.M.J.; Yasarla, R.; Patel, V.M. Transweather: Transformer-based restoration of images degraded by adverse weather conditions. In Proceedings of the CVPR, New Orleans, LA, USA, 18–24 June 2022; pp. 2353–2363.