Snow Removal in Video: A New Dataset and A Novel Method

Haoyu Chen[1], Jingjing Ren[1], Jinjin Gu[2], Hongtao Wu[1],
Xuequan Lu[3], Haoming Cai[4], Lei Zhu[1,5]

[1]The Hong Kong University of Science and Technology (Guangzhou) [2]The University of Sydney
[3]La Trobe University [4]The University of Maryland
[5]The Hong Kong University of Science and Technology

Project page: https://haoyuchen.com/VideoDesnowing

Abstract

Snowfall is a common weather phenomenon that can severely affect computer vision tasks by obscuring objects and scenes. However, existing deep learning-based snow removal methods are designed for single images only. In this paper, we target a more complex task – video snow removal, which aims to restore the clear video from the snowy video. To facilitate this task, we propose the first high-quality video dataset, which simulates realistic physical characteristics of snow and haze using a rendering engine and augmentation techniques. We also develop a deep learning framework for video snow removal. Specifically, we propose a snow-query temporal aggregation module and a snow-aware contrastive learning loss function. The module aggregates features between video frames and removes snow effectively, while the loss function helps identify and eliminate snow features. We conduct extensive experiments and demonstrate that our proposed dataset is more realistic than previous datasets, and the models trained on it achieve better performance in real-world snowing images. Our proposed method outperforms state-of-the-art video and image-based methods on both synthetic and real snowy videos.

1. Introduction

Snowfall is a common weather phenomenon that can significantly affect visibility in photographs, interfering with and reducing the accuracy of computer vision tasks such as target detection [34], tracking [14], and autonomous driving [33]. Image and video restoration [39, 23, 48, 37, 46, 7, 5, 26, 40, 43, 42, 41, 45, 31] is a widely studied direction, but research on image and video snow removal is still limited. Removing snow is crucial for improving accuracy and robustness of computer vision tasks. Additionally, snow can obstruct objects, decreasing image and video quality. Snow removal is a challenging task that requires distinguishing snowflake regions from the background and restoring obscured scenes, complicated by the complex geometric shape and texture of snowflakes. Moreover, snowfall often comes with haze, which further obscures the images.

Contrast with single image, the video provides richer information about the dynamic features of the scene, i.e., a consecutive frame sequence on the temporal dimension. Although recent image snow removal algorithms [25, 11, 12, 50, 8, 10] have achieved good results, applying these algorithms directly to video snow removal is inappropriate because image algorithms ignore temporal information in video frames and this information could be very useful to improve snow removal outcomes. As such, it is significant to conduct in-depth research and exploration for snow video data. Previous studies have explored some aspects of video desnowing, such as methods based on non-deep learning [32, 18] or online learning [22]. However, deep learning-based video desnowing remains an under-researched area. A major challenge for applying deep learning methods is the lack of high-quality video desnowing datasets. In the era of powerful deep learning, this becomes an urgent problem that needs to be addressed. In fact, the lack of appropriate datasets has significantly slowed down the progress of deep learning-based video snow removal research, which has been a bottleneck in this field.

To this end, we present the first high-quality annotated video dataset for video snow removal: **R**ealistic **V**ideo De**S**nowing **D**ataset (RVSD). RVSD contains a total of 110 pairs of videos. Each pair contains snowy and hazy videos and corresponding snow-free and haze-free ground truth videos. Unlike previous image datasets that simulate snow using Photoshop, we use a rendering engine and various augmentation techniques to generate snow with diverse and realistic physical properties. This results in more realistic and varied synthesized videos, which improve the model's performance on real-world data. To further enhance the model's performance, we also consider the effect of haze

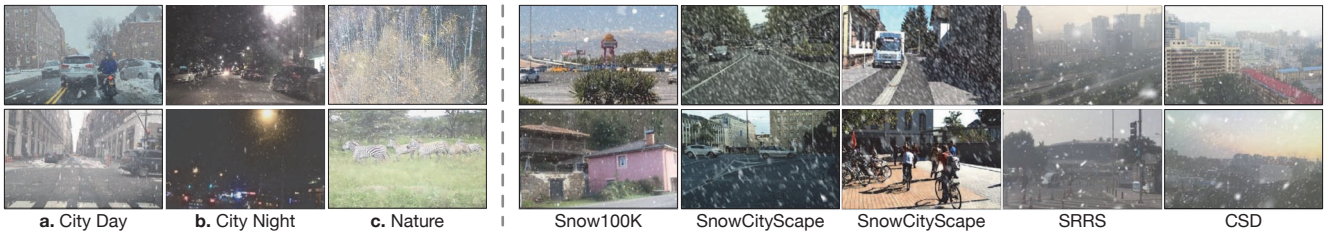


Figure 1: Samples of the proposed dataset (left) and previous datasets (right). Refer to the supplementary to see the videos.

in snowy conditions. Previous methods for fog synthesis relied on mathematical models with limitations, such as requiring accurate depth information and assuming fixed and uniform haze characteristics. However, real-world snowy conditions involve dynamic and heterogeneous haze. To overcome these limitations, we propose a new haze model that uses Unreal Engine 5 to render storm haze and obtain a haze layer. We then fuse the haze layer with the input video after data augmentation to produce more diverse and realistic haze, better simulating real snowy conditions. We conduct experiments to show that our proposed dataset has more-realistic snow scenes than existing datasets, and models trained on our dataset achieve better snow removal performance on real snowy images. Therefore, our dataset can facilitate the training and evaluation of video snow removal algorithms, and advance video snow removal research.

Trivially applying the video processing methods on our dataset may result in suboptimal since they do not consider snow and haze properties. Moreover, there are no existing deep learning-based methods for video snow removal. Therefore, we propose the first deep learning-based framework in this work. Our framework consists of three main steps. First, we perform preliminary snow removal on single frames to reduce the effect of snow occlusion on optical flow estimation and video feature alignment. Second, we fuse the video frames and perceive and remove the snow. We introduce a novel "snow-query temporal aggregation module", which aggregates features across video frames while perceiving and removing snow. Third, we design a new "snow-aware contrastive loss", which leverages the prior knowledge that snowflakes have different shapes, motion directions, and other characteristics in different videos. This loss function helps identify and remove snow features more accurately. We conduct extensive experiments to show that our proposed method outperforms the state-of-the-art methods. Our contributions are summarized as follows:

- We propose the first high-quality video dataset (RVSD) for video snow removal. We use the Unreal Engine and various augmentation techniques to produce snow and haze with diverse and realistic physical characteristics.
- Comprehensive experiments show that the visual effect of our proposed snow removal dataset is more realistic than previous datasets, and the models trained on our proposed dataset can achieve better performance in real-world snowy images.
- A deep learning-based video snow removal framework is proposed. The snow-aware temporal aggregation module and snow-aware contrastive learning loss are introduced, to effectively remove snow. Experiments prove that our method enables the best performance.

2. Related Work

2.1. Desnowing Datasets

There are several single-image snow removal datasets available. Snow100K [25] first used PhotoShop to synthesize snow. In order to model different types of snow, SnowKITTI2012 [50] and SnowCityScapes [50] contains three levels of snow, including light, medium, and heavy snow. However, these datasets do not take into account the veiling effect of snowy weather, the models trained on these datasets cannot remove haze and have limited practical performance in real snowy scenarios. SRRS [11] firstly simulated the veiling effect inspired by the Koschmieder model, and then used PhotoShop to synthesize the snow streaks. CSD [12] added Gaussian blur to the above two steps to better simulate the real snow image. Nearly all methods use PhotoShop to synthesize snow. As a result, the shape and style of snow are relatively homogeneous, which substantially limits the model's performance on realistic and complex snowing scenes. A quick comparison of these datasets can be found in Table 1. *To the best of our knowledge, there is no dataset for video desnowing, which significantly hinders the research of video desnowing.*

2.2. Image Desnowing

Early image snow removal methods tend to perform snow removal based on physical priors. [1] performed snow detection and removal by calculating the histogram of orientations of snow streaks. [29] used features on saturation and visibility to remove rain and snow from a single image. In recent years, deep learning has refreshed the snow removal

Dataset	Venue	Type	# images	Resolution	Snow synthesis	Static Haze	Dynamic Haze	Illumination	User Study (1 - 10)
Snow100K [25]	TIP 2018	image	100K	$\leq 640 \times 640$	Photoshop	✗	✗	✗	3.98
SnowCityScapes[50]	TIP 2021	image	2K+2K	512×256	Photoshop	✗	✗	✗	4.18
SnowKITTI2012 [50]	TIP 2021	image	1.5K+1K	884×256	Photoshop	✗	✗	✗	3.72
SRRS [11]	ECCV 2020	image	15K	640×480	Photoshop	✓	✗	✗	4.20
CSD [12]	ICCV 2021	image	10K	640×480	Photoshop	✓	✗	✗	4.67
Ours	–	video	–	480p - 4K	UE Rendering	✓	✓	✓	**6.94**

Table 1: Comparison with previous desnowing datasets.

Scenes	# videos	# frames	# frames (512×512)
city day	40	5,121	31,020
city night	20	2,378	23,878
nature	43	3,526	32,526
other	7	407	3,012
In total	110	11,432	90,436

Table 2: Statistics of our dataset.

task. The first deep learning based snow removal method DesnowNet [25] proposed a multi-stage, multi-scale design to remove translucent and opaque snow streaks. JSTASR [11] proposed a joint size and transparency-aware snow removal model that can classify snow particles based on their size and remove snow. HDCW-Net [12] used a hierarchical decomposition paradigm in which a dual-tree wavelet transform and a wavelet loss are used. It also proposed a discriminative feature for snow removal called contradict channel. DDMSNet [50] introduced semantic and depth information to learn semantic-aware and geometry-aware representations for snow removal. SnowFormer [9] used a vision transformer architecture that fully combines local and global information, and obtains superior results on multiple datasets.

2.3. Video Desnowing

Before the success of deep learning-based methods, traditional computer vision techniques were used for video-based snow removal. Ren et al. [32] utilized the low-rank assumption of the background to separate sparse and dense snow, and addressed heavy snow in dynamic scenes. Kim et al. [18] considered global and local motion, as well as snowflakes of various sizes, in their snowflake removal algorithm with low-rank matrix completion. Due to the rapid improvement of deep learning models and their ability to study complex patterns, some researchers have proposed the use of deep learning for video-based snow removal. [44] utilized self-adaptive snow detection and a patch-based Gaussian mixture model, adept at removing both sparse and dense snowflakes from videos. Li et al. [21, 22] proposed a method for dynamic backgrounds where snow was encoded and removed using an online multi-scale convolutional sparse coding model.

3. Realistic Video Desnowing Dataset

Previous snow and haze synthesis methods suffer from a lack of variation and poor physical accuracy. In contrast, our approach employs a 3D rendering engine and a range of augmentations, together with a more realistic fusion method to create snow and haze with more diverse, complex, and accurate physical properties. As a result, the synthesized video is more realistic and diverse, enabling our model to enjoy better generalization ability to real-world data. We demonstrate this in Section 5.1.

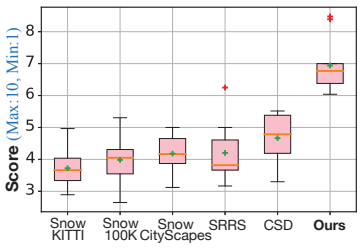


Figure 2: Box plots of the user study on the visual realism perception scores of different datasets. Higher scores represent better perceptual realism.

3.1. Collection of Clean Videos

Unlike previous work that only considered daytime video, we collect both daytime and nighttime videos in our dataset, which is unlike previous works. We apply different add-snow methods for daytime or nighttime videos to enhance the realism of the generated results. We will discuss these methods in the next section. Our dataset covers a variety of scenes such as buildings, streets, nature scenes, close shots, distant views, overhead and elevation angles to simulate the rich scenes of real snow videos. As shown in Table 2, we have collected 110 videos in total, among which 80 videos are used for training and 30 videos for testing. To accommodate the diverse resolutions of online videos, our video resolutions range from 480p to 4k. This enables the model to produce better results on videos with more sources and resolutions.

3.2. Haze Rendering

Previous studies have employed a fog synthesis method based on the atmospheric scattering model [20]: $I_{haze}(x) = J(x)t(x) + A(1 - t(x))$, where $I_{haze}(x)$ is hazy image, $J(x)$ is the haze-free image, A denotes the global atmospheric light, and $t(x)$ is the transmission matrix. This model has both strengths and weaknesses. One of the strengths is that the formula is intuitive and concise, making it easy to embed into network design. For limitations, it relies on depth information, and the generated haze exhibits a relatively uniform pattern. More importantly, there is a fundamental difference between snow-generated haze on a snowy day and conventional haze. Snow haze is often dynamic and follows the movement of snow, as illustrated in Figure 4. Therefore, using traditional haze synthesis methods that assume a homogeneous haze pattern may

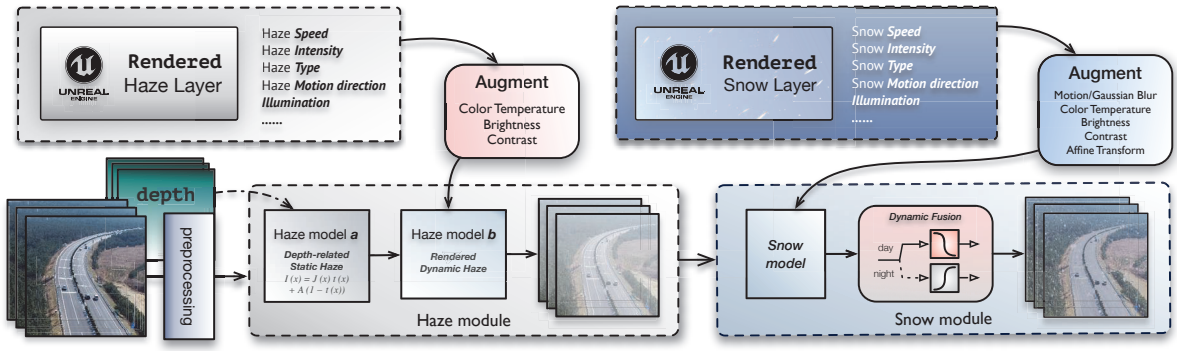


Figure 3: Snow-free videos with rich scenes are selected first. Then, we use two haze models to generate depth-related static haze and Unreal Engine (UE) rendered dynamic haze that is more realistic. After that, we use the UE to render snow layers with realistic physical characteristics. Further the snow layers are dynamically blended according to the background time after complex augmentation. These complex operations make the results more realistic with diverse features.

Figure 4: Dynamic haze in a snowstorm.

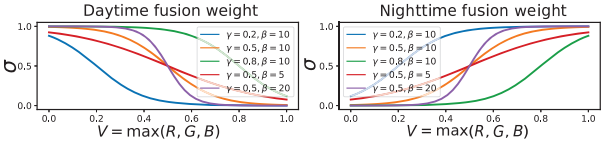


Figure 5: Fusion weight for daytime and nighttime.

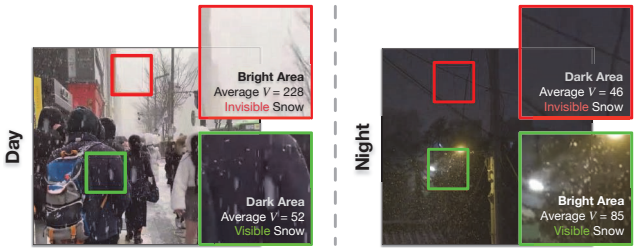


Figure 6: The visibility of daytime snow and nighttime snow varies depending on the background.

not be accurate in simulating the dynamic and varying snow haze that occurs during a snowstorm. We propose a novel haze model that leverages the capabilities of the Unreal Engine to render the haze of snowstorms and obtain different haze layers H. The dynamic haze of the storms is then integrated with the input video following data augmentation. As a result, our approach produces a more diverse and realistic haze that more closely resembles real snow days and nights. This enables the models trained on our data to handle more diverse and dynamic snow haze. Our composite haze model is defined as follows:

$$I_{haze}(x) = J(x)t(x) + A(1 - t(x)) + \mathrm{Aug}(H), \quad (1)$$

where Aug denotes data augmentation.

3.3. Snow Rendering

In previous snow production work [11, 25, 50, 12], snowflakes and snow streaks were synthesized using Photoshop, resulting in relatively homogeneous patterns. In contrast, our work employs Unreal Engine to synthesize snow. As a 3D computer graphics engine, Unreal Engine enables the creation of snow with more realistic and complex physical features, including snowflake flip, spatial changes from far to near, and perspective changes, among others. We use various 3D snow models to manipulate the physical properties of the snow, resulting in diverse and complex shapes that more closely resemble real-life snow. More details of the features are shown in Table 3. Once the snow layer S is generated, it needs to be blended with the clean video. We augment the snow layer by adjusting the color temperature, brightness, contrast, and applying random affine transformations. Then we add different levels of blur, such as Gaussian blur and motion blur. Finally, we consider the temporal properties of the video background based on time of day. During the day, especially when the sky is in the background, snowflakes tend to blend with the bright sky, resulting in snowflakes that are not easily distinguishable. At night, snowflakes are more visible in bright places, such as those illuminated by headlights or streetlights, while they tend to be less visible in places without light. We show some real samples in Figure 6. Based on this property, we utilize Eq. (2) to perform the integration of snow layer and video background.

$$\alpha(x) = \begin{cases} \sigma\left(-\left(V(x) - \gamma\right) \times \beta\right), \text{ if daytime} \\ \sigma\left(\left(V(x) - \gamma\right) \times \beta\right), \text{ if nighttime} \end{cases} \quad (2)$$

Feature	Details
Snow Speed	by setting gravity intensity, wind speed, air friction coefficient and other parameters, snow can move at different speeds
Snow Intensity	by setting the number of particles in the snow model, our dataset contains snow of various degrees.
Snow Type	by employing various snow models and adjusting various parameters, combined with the Unreal Engine to impart realistic physical features to the snowflakes (such as turbulence, rotation, etc.), the resulting snow has a highly diverse range of shapes, sizes, and characteristics.
Motion direction	head-on, back-on, left-facing, right-facing, and many other angles
Illumination	lights has different intensities and colors, such as different warm and cold tones
Veiling effect	depth-related static haze, as well as rendered dynamic complex haze.
Scenes	city, street, day & night, nature

Table 3: Features of our proposed dataset.

Specifically, we first convert the RGB image to the HSV color space, where $V = \max(R, G, B)$, represents the largest color component. We obtain a corresponding blending weight based on the pixel's brightness. v denotes the value of the V channel in the HSV space, which is normalized to the range of 0 to 1. γ and β are two parameters artificially adjusted based on the specific video, and σ represents the softmax function. Figure 5 shows some examples. Our dynamic video snow model is defined as follows:

$$Z(x) = I_{haze}(x) + \alpha(x)\,\mathrm{Aug}(S(x)). \tag{3}$$

where Aug denotes data augmentation. Z is the final output with both snow and haze.

4. Snow-Aware Video Desnowing Network

4.1. Network Architecture

Our network architecture is composed of three main modules: the encoder, the snow-query temporal aggregation module, and the decoder, as shown in Figure 7. To mitigate the impact of snow occlusion on video frame alignment and fusion, the encoder takes single-frame images as input and generates coarse features F^C, which are then transformed into coarse desnowed images by the recovery module. However, residual snow may still remain in the video frames due to the lack of utilization of video information. Therefore, it is imperative to incorporate multi-frame video information to further remove residual snow. To achieve this, the Snow-query Temporal Aggregation Module takes in continuous video frames and the features obtained by the encoder. This module aligns and fuses video features, and can perceive the location of snow, which facilitates more effective snow removal. Then this module outputs continuous refined video features F^F, which are fed into the decoder. Finally, the recovery module is employed to obtain the final desnowed video.

Recovery Module. To simultaneously remove snow and snow-induced haze, we adopt the atmospheric scattering model to decompose images containing both snow and haze into three parts: snow feature S, the global atmospheric light A, and transmission matrix T. We perform this operation in the feature space of the image. Here, $S \in \mathbb{R}^{N\times C\times H\times W}$, while $A \in \mathbb{R}^{N\times 1\times H\times W}$ and $T \in \mathbb{R}^{N\times C\times H\times W}$. N is the number of frames, C is the number of feature channel, H and W are the height and width. The recovery process can be formulated as:

$$\begin{aligned} F(x) &= \frac{E(I(x)) - S(x) - (1 - T(x))A(x)}{T(x)}, \\ J(x) &= M(F(x)), \end{aligned} \tag{4}$$

where $I(x)$ is the input snowy image, $F(x)$ is the output desnowed feature, $E(\cdot)$ is the feature extractor, $M(\cdot)$ is a convolution layer that maps the feature into image space, and $J(x)$ is the snow-free image.

Coarse Desnowing of Single Video Frames Prior to Alignment. Video restoration networks frequently employ optical flow or deformable convolution to integrate information from multiple frames for feature extraction. However, this is not entirely suitable for video snow removal owing to the significant masking of the video background by snowflakes and snow lines, which disturb temporal feature aggregation. The position discrepancies between snowflakes and snow lines in adjacent frames, especially in heavily snowing videos, pose a challenge to alignment methods in estimating accurate motion parameters. Thus, coarse desnowing of single frames before temporal feature attending is crucial for effectively integrating information between frames in video snow removal later on. This step does not cause any information loss for the further step. The reason is that the desnowing inputs of the further step are the features of original snowy video frames, instead of the desnowed images of the first step.

Figure 9 shows the results of optical flow computation on video frames with and without snow. Optical flow computation on snow-covered video frames shows significant errors, which may mislead the alignment of video frame features. Our idea is to obtain coarse desnowed frames before computing optical flows. Specifically, the encoder takes a single snowy frame as input and outputs coarse features S^C, A^C, and T^C. The recovery module then restores the coarse desnowed frame feature F^C and its corresponding desnowed frame x^C.

4.2. Snow-Query Temporal Aggregation

Temporal Aggregation Module. To fully utilize the features between video frames, we use bidirectional propagation to independently propagate the frames' features forward and backward in time dimension. This technique enables the information to flow in both directions, which can help capture more spatial-temporal context and improve the model's accuracy. Given coarse desnowed frame x_i^C and its neighboring frame x_{i-1}^C and x_{i+1}^C, we use SpyNet [30] to estimate the optical flows $O_{t+1\to t}$ and $O_{t-1\to t}$. To obtain the corresponding features propagated from the neighboring frames, which are denoted as h_{t-1}^f and h_{t+1}^b, we have

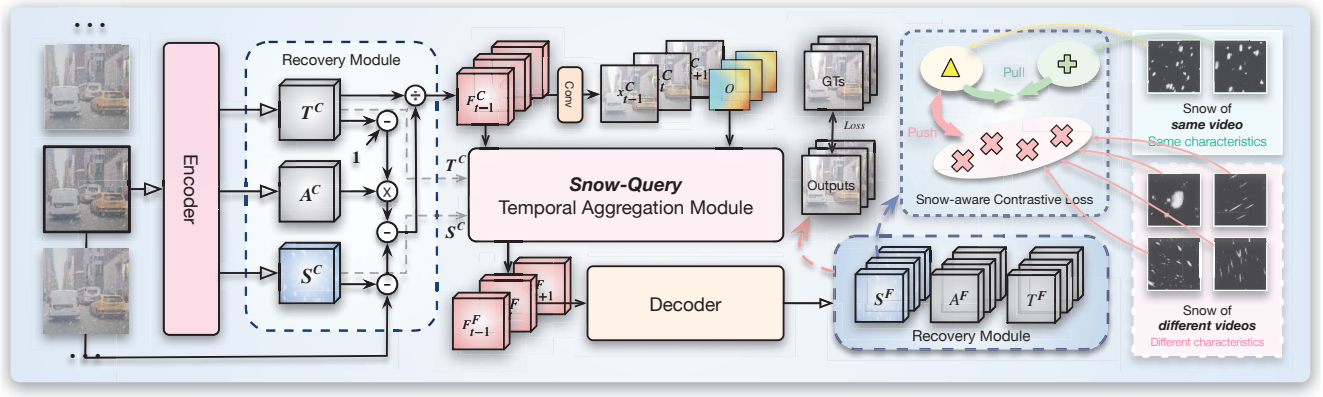
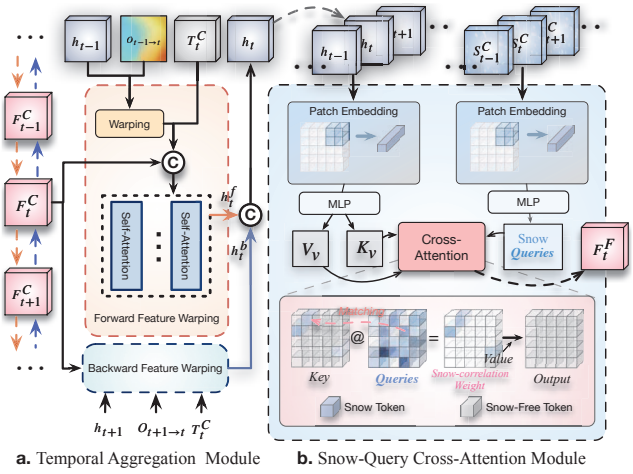


Figure 7: **Snow-Aware Video Desnowing Network (SVDNet).** The architecture is composed of three modules: the encoder, the snow-query temporal aggregation module, and the decoder. To avoid the effect of snow occlusion on video alignment, the encoder first uses a single frame to generate coarse feature F^C and desnowed image x^C. Then the Snow-query Temporal Aggregation Module aligns and fuses video features, detects snow, and outputs continuous fine video features F^F. These features are then fed into the decoder, and the recovery module is used to obtain the final desnowed video.



a. Temporal Aggregation Module **b.** Snow-Query Cross-Attention Module

Figure 8: Snow-Query Temporal Aggregation Module.

$$h_t^b = H_b\left(F_t^C, O_{t+1\to t}, h_{t+1}^b, T_t^C\right),$$
$$h_t^f = H_f\left(F_t^C, O_{t-1\to t}, h_{t-1}^f, T_t^C\right), \tag{5}$$

where H_b and H_f denote the backward and forward propagation branches.

$$w_t^{\{b,f\}} = W\left(h_{t\pm1}, O_{t\pm1\to t}\right),$$
$$h_t^{\{b,f\}} = \text{MSA}\left(\text{cat}\left(w_t^{\{b,f\}}, F_t^C, T_t^C\right)\right), \tag{6}$$
$$h_t = \text{cat}(h_t^f, h_t^b),$$

where W denotes feature warping, and MSA denotes multi-head self-attention.

Snow-Query Cross-Attention Module. The snow features for the same video are always similar. Thus, we can further use the existing snow features to match and guide the network to find the remaining snow. We show this module in Figure 8. First, we perform patch embedding for h_t and corresponding snow features S_t^C. We consider each patch of size $2 \times 2 \times C$ as a token. Then, a linear embedding layer is applied to project the features of each token to an arbitrary dimension. The key design is using snow features as queries. The snow features can help find the residual snow regions in the video feature. Specifically, the queries obtained from the snow feature and the key obtained from the

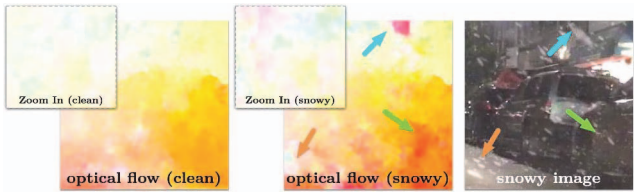


Figure 9: Incorrect estimation of optical flow due to snow.

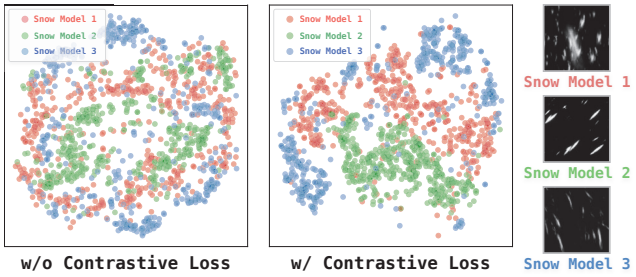


Figure 10: t-SNE visualization of features learned with/without the contrastive loss. Our contrastive loss function effectively distinguishes different types of snow. while without it, their feature distributions highly overlap.

Figure 11: **Visual results on the real-world images.** The same model is trained separately on different datasets for image snow removal. The model trained on our proposed dataset achieves the best results for snow and haze removal on real images.

video feature are multiplied to obtain a correlation matrix. The coordinates with high correlation represent the residual snow in the video, which enables the network to locate the residual snow and further remove it.

$$\text{CrossAttn}\left(\mathcal{Q}_s, \mathcal{K}_v, \mathcal{V}_v\right) = \text{Softmax}\left(\frac{\mathcal{Q}_s \mathcal{K}_v}{\sqrt{d}} + p\right)\mathcal{V}_v, \tag{7}$$

where Q_s, K_v, V_v are the snow query, video key, and value matrices. d is the dimension. p is the relative position embedding.

4.3. Snow-Aware Contrastive Loss

We introduce a prior for snow-covered images: the snow features vary across videos based on the snow's angle and properties (e.g., density, shape, direction, and speed), but are consistent within the same video, as shown in Figure 7. We design a snow-aware contrastive learning paradigm. It uses snow features from different frames of the same video as positive samples S^+ and snow features from different videos as negative samples S^-. By pulling positive samples closer together and pushing negative samples farther apart in feature space, the network can more accurately identify snow features and improve snow removal. The loss function for contrastive learning is formulated as:

$$\mathcal{L}_{cont}\left(S, S^+, S^-\right) = -\log\left[\frac{\exp\left(D(S^+, S^-)/\tau\right)}{\exp\left(D(S^+, S^-)/\tau\right) + \sum_{r=1}^{N}\exp\left(D(S^+, S^-)/\tau\right)}\right] \tag{8}$$

where $D(x, y) = \Psi(x) \cdot \Psi(y)$. $\Psi(\cdot)$ is the two-layer MLP function which implements feature projection. τ is the scale temperature which is set to 0.1. N denotes the total number of negative samples.

To illustrate the effect of the proposed snow-aware contrastive learning loss on snow feature learning, we visualize the learned features using t-SNE for three distinct types of snow with different motion directions and snow stripe sizes, as shown in Figure 10. Without the proposed contrastive learning, different snow types have highly similar and overlapping feature distributions. In contrast, with the contrastive learning loss, the features of different snow types are clearly distinguished, demonstrating the ability of the method to accurately identify diverse snow features. The overall loss function is formulated as: $\mathcal{L} = \mathcal{L}_1 + \lambda\mathcal{L}_{cont}$, where λ is set to 0.1.

5. Experiments

5.1. Proposed Dataset vs. Previous Datasets

Objective Evaluation. Previous work [4, 51] has demonstrated the significant impact of training set data on the generalization performance of the model. To demonstrate that the snow scenes in our created dataset are more realistic and closer to real-world scenarios than those in other synthetic datasets, we train the same model on different datasets and test it on real snowy images. The results show that the model trained on our proposed dataset performs better in handling real snow, thus validating the authenticity and usefulness of our dataset. Figure 11 displays two samples. In the first row, the model trained on our dataset successfully removes all snow residues, whereas models trained on other datasets fails to completely eliminate snow, leaving numerous snow residues in the output images. Since the Snow100K, SnowCityScapes, and SnowKITTI2012 datasets do not include the effect of snow haze, the models trained on these datasets cannot remove snow haze. In the second row, the models trained on the Snow100K, SnowCityScapes, and SnowKITTI2012 datasets remove slightly larger snowflakes but preserve many smaller ones. Furthermore, these models mistakenly remove white parallelogram backgrounds, mistaking them for snowflakes. Conversely, SRRS and CSD remove smaller snowflakes but fail to remove larger ones. In contrast, our proposed dataset facilitates the training of a model that effectively removes snowflakes at all sizes. It does not misidentify the white background as snow. Since there is no ground truth for real-world images, we conduct a user study to compare the

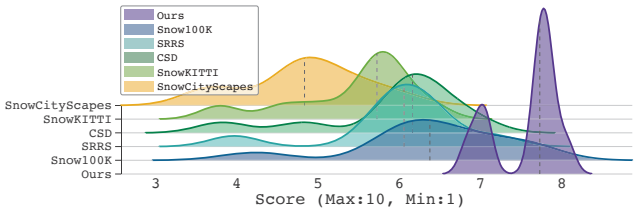


Figure 12: Distributions of mean opinion scores obtained by models trained on different datasets (**testing on real-world snow images**) .

desnowing results of models which are trained on different datasets, as shown in Figure 12. Model trained on our dataset has the best desnowing performance on real-world images.

Subjective Evaluation. To demonstrate that our dataset produces snow images that are closer to real snow in human visual perception than previous datasets, we conduct a user study to compare the perceived realism of snow generated from different datasets. The results are shown in Figure 2, dataset with higher scores indicating greater fidelity. Thanks to our proposed complex, specially designed dataset construction method, our dataset is scored much higher than the other datasets.

5.2. Performance Evaluation

Implementation Details. Each input image is randomly cropped to a spatial resolution of 256×256, and the number of the total training iterations is 500K. During the training phase, we adopt the Adam optimizer [19] with β_1=0.9 and β_2=0.99. The initial learning rate is set to 2×10^{-4} and reduced by half at the milestone of 100K iterations and 150K iterations. The batch size is set to 4. PyTorch [28] is used to implement our model with 4 RTX 3090 GPUs.

Compared Methods. We compare our network against 18 state-of-the-art methods, including three single-image desnowing models, two image adverse weather restoration models, six image restoration methods, two video deraining methods, and five video restoration methods. Please refer to Table 5 for more details. Moreover, we follow [16, 15] to employ PSNR, SSIM [38], and LPIPS [52] to quantitatively compared different methods.

Quantitative Comparison. Table 5 reports the quantitative results of our network and 17 compared methods. From these quantitative results, we can find that the performance of video-based models is generally inferior to that of single-image models. The major reason behind is that the severe occlusion of the video background by snowflakes hinders the alignment and fusion of the video frames. Among all compared methods, Restormer has the largest PSNR score of 24.34 and the smallest 0.1164, while MPRNet has the largest SSIM score of 0.8960. More importantly, our network further outperform these methods in terms of PSNR, SSIM, and LPIPS scores. It improves the PSNR score from 24.34 to 25.06, the SSIM score from 0.8960 to 0.9210, and the LPIPS score from 0.1164 to 0.0842. It demonstrates the effectiveness of our proposed framework, which first obtains a coarse snow removal result for a single frame image to reduce the challenging alignment issue on the snowy video frames, and then fully utilizes the temporal information between video frames for video desnowing.

Method	Restormer	TransWeather	S2VD	RVRT	BasicVSR++	**Ours**
$E_{warp}\downarrow(\times10^{-3})$	1.594	2.392	2.009	3.425	2.613	**1.203**
VFID ↓	0.0938	0.1511	0.1686	0.2207	0.1539	**0.0492**

Table 4: Results of temporal consistency preservation.

Type	Method	Venue	PSNR ↑	SSIM ↑	LPIPS ↓
Image Desnowing	JSTASR [11]	ECCV 2020	22.08	0.8280	0.2336
	HDCW-Net [12]	ICCV 2021	22.63	0.8592	0.2010
	SnowFormer [8]	arXiv 2022	24.01	0.8939	0.1219
Image Adverse Weather	TransWeather [35]	CVPR 2022	23.11	0.8543	0.2086
	TKL [13]	CVPR 2022	23.05	0.8589	0.2027
Image Restoration	SwinIR [23]	CVPR 2021	22.51	0.8562	0.2105
	MPRNet [49]	CVPR 2021	24.27	0.8960	0.1266
	Uformer [39]	CVPR 2022	23.61	0.8730	0.1706
	DGUNet [27]	CVPR 2022	24.18	0.8985	0.1238
	NAFNet [6]	ECCV 2022	24.01	0.8838	0.1472
	Restormer [48]	CVPR 2022	24.34	0.8929	0.1164
Video Deraining	S2VD [47]	CVPR 2021	22.95	0.8590	0.1856
	RDD [36]	ECCV 2022	22.97	0.8742	0.1631
Video Restoration	EDVR [37]	CVPRW 2019	17.93	0.5790	0.3771
	BasicVSR [2]	CVPR 2021	22.46	0.8473	0.2087
	IconVSR [2]	CVPR 2021	22.35	0.8482	0.2034
	BasicVSR++ [3]	CVPR 2022	22.64	0.8618	0.1868
	RVRT [24]	NeurIPS 2022	20.90	0.7974	0.2977
	SVDNet (Ours)	–	**25.06**	**0.9210**	**0.0842**

Table 5: Quantitative results on the proposed video dataset.

To further verify the temporal consistency preservation, we utilize a widely-used metric E_{warp} to compute the temporal consistency,and VFID [17] to compute video restoration performance. Table below shows our method has the smallest E_{warp} and VFID. Our method can achieve consistent desnowing between frames.

Visual comparisons on synthetic videos. Figure 13 visually compares video desnowing results produced by our network and state-of-the-art methods. We can find that existing methods tend to maintain snowflakes (e.g., TKL, TransWeather), or haze (e.g., Restomer), or both snowflakes and haze (i.e., RVRT) in their desnowing results. On the contrary, our method can effectively remove both snowflakes and haze, and better preserve background details.

Visual comparisons on real-world videos. Moreover, Figure 14 shows the visual comparisons between our network and state-of-the-art methods in terms of input frames from real-world snowstorm videos with both heavy snow and haze. We can find that our method can effectively remove

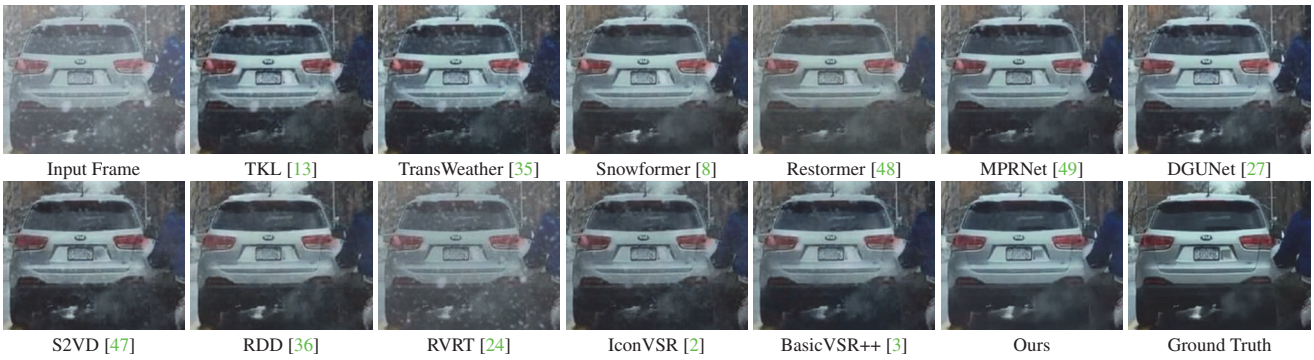


Figure 13: Visual video desnowing results produced by our network and state-of-the-art methods.

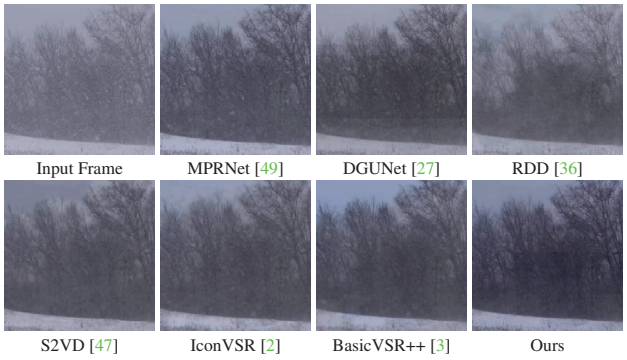


Figure 14: Visual comparison on **real snowstorm video**.

Index	Temporal Aggregation	Snow-Query	Contrastive Loss	PSNR ↑	SSIM ↑	LPIPS ↓
M1(basic)				24.16	0.8990	0.1125
M2	✓			24.58	0.9130	0.0934
M3	✓	✓		24.80	0.9108	0.9004
M4	✓		✓	24.84	0.9181	0.0882
Ours	✓	✓	✓	**25.06**	**0.9210**	**0.0842**

Table 6: Quantitative results of the ablation study.

both the haze and snowflakes and betterrecover the obscured background details, while compared methods tend to main parts of haze and snowflakes.

Ablation Study. We perform ablation study experiments to verify the temporal aggregation module, the snow-query cross-attention module, and the snow-aware contrastive loss of our network. To do so, we first built a basic model (see M1 of Table 6) by removing these three major components from our network, and then add the temporal aggregation module into M1 to construct M2. Then we add the snow-query cross-attention module into M2 to build M3, and add the snow-aware contrastive loss into M2 to build M4. Table 6 reports the PSNR, SSIM, and LPIPS results of our network and four baseline networks. Specifically, compared to M1, M2 improves the PSNR score from 24.16 to 24.58, the SSIM score from 0.8990 to 0.9130, and the LPIPS score from 0.1125 to 0.0934. It shows that the temporal aggregation module incurs a better video desnowing performance. Then, we find that M3 and M4 have a superior PSNR, SSIM, and LPIPS performance over M2. It demonstrates the effectiveness of the snow-aware cross-attention module or snow-aware contrastive loss for enhancing the video desnowing performance. Moreover, our method has larger PSNR and SSIM scores and a smaller LPIPS score than M3 and M4. It shows that combing three components together has the best performance of video desnowing.

6. Conclusion

The first contribution of our work is to synthesize the first high-quality video desnowing dataset, which is more realistic than previous image datasets. Then, we devise a deep learning-based framework that incorporates a snow-query temporal aggregation module and a snow-aware contrastive learning loss. Experimental results show that our network outperforms state-of-the-art methods in terms of video denosing on synthetic and real-world snowy videos.

Acknowledgment. This work was supported by the National Natural Science Foundation of China (Grant No. 61902275), and Guangzhou Municipal Science and Technology Project (Grant No. 2023A03J0671).

References

[1] Jérémie Bossu, Nicolas Hautiere, and Jean-Philippe Tarel. Rain or snow detection in image sequences through use of a histogram of orientation of streaks. *International journal of computer vision*, 93:348–367, 2011.

[2] Kelvin C.K. Chan, Xintao Wang, Ke Yu, Chao Dong, and Chen Change Loy. Basicvsr: The search for essential components in video super-resolution and beyond. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2021.

[3] Kelvin C.K. Chan, Shangchen Zhou, Xiangyu Xu, and Chen Change Loy. BasicVSR++: Improving video super-resolution with enhanced propagation and alignment. In

IEEE Conference on Computer Vision and Pattern Recognition, 2022.

[4] Haoyu Chen, Jinjin Gu, Yihao Liu, Salma Abdel Magid, Chao Dong, Qiong Wang, Hanspeter Pfister, and Lei Zhu. Masked image training for generalizable deep image denoising. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1692–1703, June 2023.

[5] Haoyu Chen, Jinjin Gu, and Zhi Zhang. Attention in attention network for image super-resolution, 2021.

[6] Liangyu Chen, Xiaojie Chu, Xiangyu Zhang, and Jian Sun. Simple baselines for image restoration. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part VII*, pages 17–33. Springer, 2022.

[7] Sixiang Chen, Tian Ye, Yun Liu, and Erkang Chen. Dualformer: Hybrid self-attention transformer for efficient image restoration. *arXiv preprint arXiv:2210.01069*, 2022.

[8] Sixiang Chen, Tian Ye, Yun Liu, Erkang Chen, Jun Shi, and Jingchun Zhou. Snowformer: Scale-aware transformer via context interaction for single image desnowing. *arXiv preprint arXiv:2208.09703*, 2022.

[9] Sixiang Chen, Tian Ye, Yun Liu, Erkang Chen, Jun Shi, and Jingchun Zhou. Snowformer: Scale-aware transformer via context interaction for single image desnowing. *arXiv preprint arXiv:2208.09703*, 2022.

[10] Sixiang Chen, Tian Ye, Yun Liu, Taodong Liao, Jingxia Jiang, Erkang Chen, and Peng Chen. Msp-former: Multi-scale projection transformer for single image desnowing. In *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5. IEEE, 2023.

[11] Wei-Ting Chen, Hao-Yu Fang, Jian-Jiun Ding, Cheng-Che Tsai, and Sy-Yen Kuo. Jstasr: Joint size and transparency-aware snow removal algorithm based on modified partial convolution and veiling effect removal. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXI 16*, pages 754–770. Springer, 2020.

[12] Wei-Ting Chen, Hao-Yu Fang, Cheng-Lin Hsieh, Cheng-Che Tsai, I Chen, Jian-Jiun Ding, Sy-Yen Kuo, et al. All snow removed: Single image desnowing algorithm using hierarchical dual-tree complex wavelet representation and contradict channel loss. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4196–4205, 2021.

[13] Wei-Ting Chen, Zhi-Kai Huang, Cheng-Che Tsai, Hao-Hsiang Yang, Jian-Jiun Ding, and Sy-Yen Kuo. Learning multiple adverse weather removal via two-stage knowledge learning and multi-contrastive regularization: Toward a unified model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17653–17662, 2022.

[14] Heng Fan, Liting Lin, Fan Yang, Peng Chu, Ge Deng, Sijia Yu, Hexin Bai, Yong Xu, Chunyuan Liao, and Haibin Ling. Lasot: A high-quality benchmark for large-scale single object tracking. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5374–5383, 2019.

[15] Jinjin Gu, Haoming Cai, Haoyu Chen, Xiaoxing Ye, Jimmy Ren, and Chao Dong. Image quality assessment for perceptual image restoration: A new dataset, benchmark and metric. *arXiv preprint arXiv:2011.15002*, 2020.

[16] Jinjin Gu, Haoming Cai, Haoyu Chen, Xiaoxing Ye, Jimmy Ren, and Chao Dong. Pipal: a large-scale image quality assessment dataset for perceptual image restoration. In *European Conference on Computer Vision*, pages 633–651. Springer, 2020.

[17] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. *Advances in neural information processing systems*, 30, 2017.

[18] Jin-Hwan Kim, Jae-Young Sim, and Chang-Su Kim. Video deraining and desnowing using temporal correlation and low-rank matrix completion. *IEEE Transactions on Image Processing*, 24(9):2658–2670, 2015.

[19] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[20] Boyi Li, Wenqi Ren, Dengpan Fu, Dacheng Tao, Dan Feng, Wenjun Zeng, and Zhangyang Wang. Benchmarking single-image dehazing and beyond. *IEEE Transactions on Image Processing*, 28(1):492–505, 2018.

[21] Minghan Li, Xiangyong Cao, Qian Zhao, Lei Zhang, Chenqiang Gao, and Deyu Meng. Video rain/snow removal by transformed online multiscale convolutional sparse coding. *arXiv preprint arXiv:1909.06148*, 2019.

[22] Minghan Li, Xiangyong Cao, Qian Zhao, Lei Zhang, and Deyu Meng. Online rain/snow removal from surveillance videos. *IEEE Transactions on Image Processing*, 30:2029–2044, 2021.

[23] Jingyun Liang, Jiezhang Cao, Guolei Sun, Kai Zhang, Luc Van Gool, and Radu Timofte. Swinir: Image restoration using swin transformer. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 1833–1844, 2021.

[24] Jingyun Liang, Yuchen Fan, Xiaoyu Xiang, Rakesh Ranjan, Eddy Ilg, Simon Green, Jiezhang Cao, Kai Zhang, Radu Timofte, and Luc Van Gool. Recurrent video restoration transformer with guided deformable attention. *arXiv preprint arXiv:2206.02146*, 2022.

[25] 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.

[26] Xin Luo, Yunan Zhu, Shunxin Xu, and Dong Liu. On the effectiveness of spectral discriminators for perceptual quality improvement. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023.

[27] Chong Mou, Qian Wang, and Jian Zhang. Deep generalized unfolding networks for image restoration. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17399–17410, 2022.

[28] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming

Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32, 2019.

[29] Soo-Chang Pei, Yu-Tai Tsai, and Chen-Yu Lee. Removing rain and snow in a single image using saturation and visibility features. In *2014 IEEE International Conference on Multimedia and Expo Workshops (ICMEW)*, pages 1–6. IEEE, 2014.

[30] Anurag Ranjan and Michael J Black. Optical flow estimation using a spatial pyramid network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4161–4170, 2017.

[31] Jingjing Ren, Qingqing Zheng, Yuanyuan Zhao, Xuemiao Xu, and Chen Li. Dlformer: Discrete latent transformer for video inpainting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3511–3520, June 2022.

[32] Weihong Ren, Jiandong Tian, Zhi Han, Antoni Chan, and Yandong Tang. Video desnowing and deraining based on matrix decomposition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4210–4219, 2017.

[33] Pei Sun, Henrik Kretzschmar, Xerxes Dotiwalla, Aurelien Chouard, Vijaysai Patnaik, Paul Tsui, James Guo, Yin Zhou, Yuning Chai, Benjamin Caine, et al. Scalability in perception for autonomous driving: Waymo open dataset. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2446–2454, 2020.

[34] Mingxing Tan, Ruoming Pang, and Quoc V Le. Efficientdet: Scalable and efficient object detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10781–10790, 2020.

[35] 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.

[36] Shuai Wang, Lei Zhu, Huazhu Fu, Jing Qin, Carola-Bibiane Schönlieb, Wei Feng, and Song Wang. Rethinking video rain streak removal: A new synthesis model and a deraining network with video rain prior. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XIX*, pages 565–582. Springer, 2022.

[37] Xintao Wang, Kelvin CK Chan, Ke Yu, Chao Dong, and Chen Change Loy. Edvr: Video restoration with enhanced deformable convolutional networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 0–0, 2019.

[38] 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.

[39] Zhendong Wang, Xiaodong Cun, Jianmin Bao, Wengang Zhou, Jianzhuang Liu, and Houqiang Li. Uformer: A general u-shaped transformer for image restoration. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17683–17693, 2022.

[40] Zeyu Xiao, Xueyang Fu, Jie Huang, Zhen Cheng, and Zhiwei Xiong. Space-time distillation for video super-resolution. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2113–2122, June 2021.

[41] Zeyu Xiao, Yutong Liu, Ruisheng Gao, and Zhiwei Xiong. Cutmib: Boosting light field super-resolution via multi-view image blending. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1672–1682, June 2023.

[42] Zeyu Xiao, Wenming Weng, Yueyi Zhang, and Zhiwei Xiong. Eva^2: Event-assisted video frame interpolation via cross-modal alignment and aggregation. *IEEE Transactions on Computational Imaging*, 8:1145–1158, 2022.

[43] Zeyu Xiao, Zhiwei Xiong, Xueyang Fu, Dong Liu, and Zheng-Jun Zha. Space-time video super-resolution using temporal profiles. In *Proceedings of the 28th ACM International Conference on Multimedia*, pages 664–672, 2020.

[44] Bin Yang, Zhenhong Jia, Jie Yang, and Nikola K Kasabov. Video snow removal based on self-adaptation snow detection and patch-based gaussian mixture model. *IEEE Access*, 8:160188–160201, 2020.

[45] Yijun Yang, Angelica Aviles-Rivero, Huazhu Fu, Ye Liu, Weiming Wang, and Lei Zhu. Video adverse-weather-component suppression network via weather messenger and adversarial backpropagation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023.

[46] Tian Ye, Sixiang Chen, Yun Liu, Yi Ye, Erkang Chen, and Yuche Li. Underwater light field retention: Neural rendering for underwater imaging. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 488–497, 2022.

[47] Zongsheng Yue, Jianwen Xie, Qian Zhao, and Deyu Meng. Semi-supervised video deraining with dynamical rain generator. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 642–652, 2021.

[48] 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.

[49] Syed Waqas Zamir, Aditya Arora, Salman Khan, Munawar Hayat, Fahad Shahbaz Khan, Ming-Hsuan Yang, and Ling Shao. Multi-stage progressive image restoration. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 14821–14831, 2021.

[50] Kaihao Zhang, Rongqing Li, Yanjiang Yu, Wenhan Luo, and Changsheng Li. Deep dense multi-scale network for snow removal using semantic and depth priors. *IEEE Transactions on Image Processing*, 30:7419–7431, 2021.

[51] Ruofan Zhang, Jinjin Gu, Haoyu Chen, Chao Dong, Yulun Zhang, and Wenming Yang. Crafting training degradation distribution for the accuracy-generalization trade-off. *arXiv preprint arXiv:2305.18107*, 2023.

[52] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *CVPR*, 2018.