# Controllable Weather Synthesis and Removal with Video Diffusion Models

Chih-Hao Lin[1,2], Zian Wang[1,3,4], Ruofan Liang[1,3,4], Yuxuan Zhang[1],
Sanja Fidler[1,2,3], Shenlong Wang[2], Zan Gojcic[1]
[1]NVIDIA [2]University of Illinois Urbana-Champaign [3]University of Toronto [4]Vector Institute


Figure 1. We introduce WEATHERWEAVER, a generative editing method for synthesizing and removing weather effects. Given an input video, it creates corresponding videos with diverse weather condition (rain, snow, fog, clouds) and precise control over the intensity (left), removes weather from real footage (right). The results are photorealistic, temporally consistent, and faithfully preserve the original scene.

## Abstract

*Generating realistic and controllable weather effects in videos is valuable for many applications. Physics-based weather simulation requires precise reconstructions that are hard to scale to in-the-wild videos, while current video editing often lacks realism and control. In this work, we introduce* WEATHERWEAVER, *a video diffusion model that synthesizes diverse weather effects—including rain, snow, fog, and clouds—directly into any input video without the need for 3D modeling. Our model provides precise control over weather effect intensity and supports blending various weather types, ensuring both realism and adaptability. To overcome the scarcity of paired training data, we propose a novel data strategy combining synthetic videos, generative image editing, and auto-labeled real-world videos. Extensive evaluations show that our method outperforms state-of-the-art methods in weather simulation and removal, providing high-quality, physically plausible, and scene-identity-preserving results over various real-world videos.*

## 1. Introduction

Simulating photorealistic weather effects in videos, such as rain, snow, fog, or clouds, is a challenging yet essential task in computer vision and graphics. High-quality weather simulations enable a range of creative applications in film production, AR/VR, and video games. Moreover, controllable weather simulation is invaluable for training and evaluating perception systems in safety-critical domains such as autonomous driving and robotics, where robust performance under diverse weather conditions is crucial.

Comprehensive weather simulation must capture both transient effects—such as falling rain, swirling snow, or drifting fog—and persistent or accumulative changes, such as snow buildup on the ground or water puddles after rain. In modern graphics engines, transient effects are often handled using particle-based simulations [21, 25, 69], while persistent changes are approximated by modifying scene asset materials [19]. However, these methods rely on detailed, simulation-ready 3D models, limiting their applicability to synthetic environments. Recent work has attempted to adapt such pipelines to real-world videos by reconstructing scenes through methods like NeRF [51] or 3DGS [37], but imperfect reconstructions frequently introduce blending artifacts and unnatural shading [43].

Instead of employing a two-stage *reconstruct-then-simulate* approach, we formulate weather simulation in real-world videos as a video-to-video translation task, leveraging the recent success of large video generative models in video editing. Nevertheless, straightforward adaptations of general video editing methods fail to deliver the necessary realism—particularly for transient phenomena—and lack precise control over the weather type and intensity (Fig. 5). Two main challenges contribute to this: **(i)** acquiring high-quality paired data (videos of the same scene under different weather conditions) is difficult to scale in real-world settings, and **(ii)** directly translating from one weather condition to another (e.g., rainy to snowy) is inherently complex, as it requires removing one weather effect while adding another.

To overcome these challenges, we draw inspiration from modern graphics engines, which treat weather simulation as an added effect applied to an existing scene consisting of geometry, materials, and lighting. Concretely, we split our pipeline into two video diffusion models: a WEATHER REMOVAL MODEL that translates a real-world video into a "canonical," weather-free video[1], and a WEATHER SYNTHESIS MODEL that adds weather effects to a "canonical" video with precise control over both intensity and type of weather. This split offers two main advantages. First, the WEATHER REMOVAL MODEL can serve as a pseudo-labeling engine, producing paired data with realistically looking weather effects. Second, confining the WEATHER SYNTHESIS MODEL to solely adding the weather effects simplifies its task.

High-quality paired video training data is crucial to ensure both realism and scene preservation for the proposed models. However, acquiring real-world paired videos of the same dynamic scene is challenging. To address this, we introduce a new data strategy and train our models on a carefully curated combination of three data sources (see Table 1). First, we render a synthetic video dataset using standard graphics engines and fully modeled 3D environments, allowing precise control over weather attributes but yielding a synthetic appearance. Second, we generate paired image data via large image generative models (e.g., SDXL [57]) by leveraging Prompt-to-Prompt [30] method. This strategy yields more realistic outputs, albeit with lack of precise control and limitation to image data. Finally, we use these datasets to train the WEATHER REMOVAL MODEL and apply it to automatically convert real-world videos with weather effects to their "canonical" clear-day video, thus creating a large dataset of highly realistic video pairs. For training the WEATHER SYNTHESIS MODEL, we use all three sources of data.

Our resulting framework, WEATHERWEAVER, outperforms state-of-the-art methods by producing high-quality, controllable weather effects in real-world videos with precise control of intensity and type of weather. In summary, our contributions are:

- A controllable weather synthesis model that adds diverse weather effects to real-world videos, offering precise control over both intensity and type.
- A weather removal model that effectively handles both transient (*e.g*. rain, snow) and persistent (*e.g*. clouds, rain puddle, snow coverage) weather effects.
- A data curation strategy that combines synthetic data, generative models outputs, and auto-labeled real-world videos, thus improving realism and diversity of the paired data.

## 2. Related Work

**Video Editing** Image editing with generative priors has been extensively studied [3, 30, 50, 75]. However, directly applying image diffusion models in a frame-wise manner to video often leads to temporal inconsistencies. To mitigate flicker and jitter artifacts, recent methods [11, 38, 89] inverts the initial latent code and employs cross-attention control to enforce frame consistency. Similarly, [26, 58] fuse attention maps or diffusion features from the source video with those from the generated video, thereby preserving fine details and ensuring content consistency with source frames. Other approaches [20, 22, 45, 46] incorporate structural constraints or auxiliary information—such as depth maps, optical flow or G-buffers—to align generated frames with the original geometry and motion. Alternatively, some methods [29, 33] build 3D representations from source videos and apply a diffusion prior for 3D editing to ensure consistency.

Given sufficient computational budget, an alternative line of work explored one-shot fine-tuning to personalize the model to target video [52, 67, 81]. Our work builds on a pretrained video diffusion model, but eliminates the need for per-video fine-tuning and provides more precise control.

**Weather Synthesis** serves as a valuable augmentation to existing data and benefits perception tasks under adversarial weather conditions [63, 73, 77, 78]. ClimateGAN [16, 65] generates flood images from depth information; [28] synthesize controllable fog based on depth and semantics. These methods focus on specific weather effects for static images. Similarly, [64] uses CycleGAN [90] for image editing on a climate dataset. In contrast, WEATHERWEAVER is a general framework that synthesizes and controls various weather effects, including transient effects (*e.g*. rain, snow) in videos.

An alternative line of works synthesizes weather effects in 3D representations with graphics techniques [70]. [21, 27, 69] simulate snow particles and their interaction with objects and wind. These methods are typically limited to synthetic environments. ClimateNeRF [43] and subsequent works [17, 23] extend classic weather simulation by inserting physical entities into neural 3D reconstructions [37, 53],

[1]Note that *canonical weather* representation is not strictly defined. In this work, we use the term to refer to a clear sunny or overcast sky.

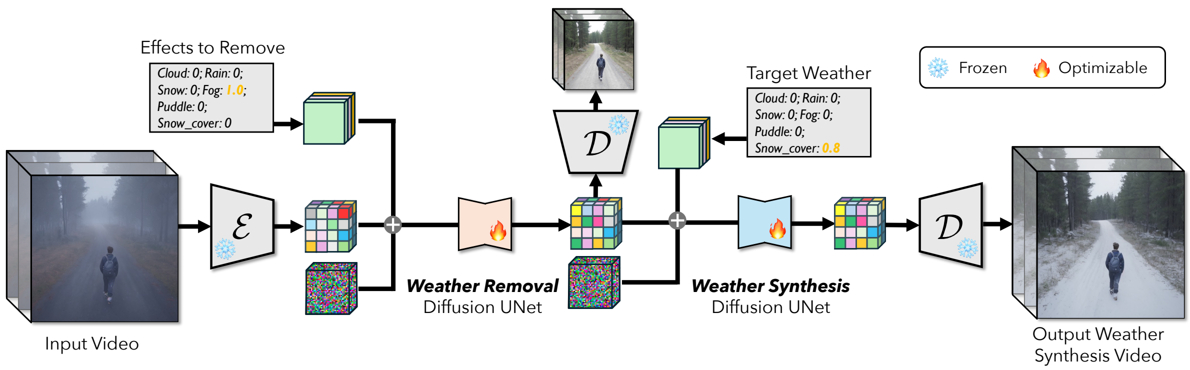



Figure 2. **Model Overview.** Our controllable weather simulation framework includes two complementary models for both weather removal and weather synthesis. These models can be used both independently and combined for weather editing tasks.

but they require accurate geometry that is challenging to acquire from sparse capture. WEATHERWEAVER leverages a data-driven video diffusion model, bypassing the need for geometry reconstruction and enabling realistic effects on diverse and dynamic videos.

**Weather Removal** is a long-standing problem for robust computer vision systems. Early methods targeted specific weather effects, such as deraining [59, 60, 83, 85], dehazing [10, 41, 48, 80], and desnowing [12, 14, 49], using specialized architectures tailored to each weather type. Recent approaches unify weather removal under a single model. All-in-One [42] handles fog, rain, and snow with a unified CNN model. [72, 76, 91] used transformer architectures with dedicated attention mechanisms to further improve restoration quality across diverse weather effects. ViWS-Net [86] introduced a video weather removal framework that incorporates temporal information for enhanced video restoration. Recent works explored using generative modelsfor weather removal [13, 55, 88]. WeatherDiffusion [55] uses patch-based diffusion denoising to effectively remove weather artifacts while preserving image details. Prior works and benchmarks in weather removal primarily focus on transient effects like fog, rain, and snow, neglecting persistent weather effects such as cloud, puddle, and snow coverage.

## 3. Preliminary: Video Diffusion Model

Diffusion models generate samples from a data distribution $p_{\text{data}}(\mathbf{I})$ by iteratively refining noisy inputs through a denoising process [18, 31, 68]. In the context of videos, video diffusion models (VDMs) typically operate in a compressed latent space to reduce computational complexity [7]. An input video $\mathbf{I} \in \mathbb{R}^{L\times H\times W\times 3}$, with $L$ frames at resolution $H \times W$, is encoded into a latent representation $\mathbf{z} = \mathcal{E}(\mathbf{I}) \in \mathbb{R}^{l\times h\times w\times C}$ using a pre-trained VAE encoder $\mathcal{E}$. The diffusion process is then applied within this latent space.

During training, noisy versions of the latent representation $\mathbf{z}_\tau$ are generated by adding Gaussian noise $\epsilon$ to the original latent $\mathbf{z}_0$ using a predefined noise schedule [36] $\mathbf{z}_\tau = \alpha_\tau \mathbf{z}_0 + \sigma_\tau \epsilon$ at timestep $\tau$. The diffusion model is trained to reverse this process using a denoising score matching objective [36] $\|\mathbf{f}_\theta(\mathbf{z}_\tau; \mathbf{c}, \tau) - \mathbf{z}_0\|_2^2$ where $\mathbf{c}$ denotes optional conditioning information. Once trained, the model generates new video samples by iteratively denoising Gaussian noise. The final output video $\hat{\mathbf{I}}$ is reconstructed by decoding the denoised latent with the VAE decoder $\mathcal{D}$.

| Dataset | Size | Weather Controllability | Temporal Consistency | Realism | Scene Diversity | Trajectory Diversity |
|---|---|---|---|---|---|---|
| Simulation | 2080k | ✓ | ✓ | ✓ | × | ✓ |
| Generation | 1147k | ✓ | × | ✓ | ✓ | × |
| Real videos | 460k | × | ✓ | ✓ | ✓ | ✓ |

Table 1. **Dataset Statistics.** We collect the weather data from three heterogeneous data sources, and mark each properties as high (✓), moderate (✓), and low/none (×). The data size is the number of image pairs (with and without weather effects).

Our method is designed to be model-agnostic and can be applied to any video diffusion model. In this work, we build on Stable Video Diffusion [7], which compresses the spatial dimensions of the video by a factor of 8 while preserving the temporal resolution, using a latent dimension of $C = 4$.

## 4. Method

We formulate weather simulation in real-world videos as a video-to-video translation task using two complementary and controllable video diffusion models. The WEATHER REMOVAL MODEL removes existing weather effects to generate a clear day video, while the WEATHER SYNTHESIS MODEL adds weather effects to the clear day video with precise control over both type and intensity.

To train these models, we decompose weather into its fundamental components (Sec. 4.1), curate a diverse multi-source dataset (Sec. 4.2), and propose a staged training strategy (Sec. 4.3). The overall pipeline is shown in Fig. 2.

### 4.1. Model Design

Our method is designed to flexibly represent and control individual weather effects. Both the weather removal and synthesis are formulated as conditional video generation task and use the same network architecture.

**Representing Weather Effects** To enable precise control over weather type and intensity, we decompose weather into six distinct effects: 1) cloud, 2) fog, 3) rain, 4) snow, 5) puddle, and 6) snow coverage (i.e., persistent snow accumulation on the ground and objects). Each effect is parameterized by a continuous strength value $s \in \mathbb{R}^+$, where higher values

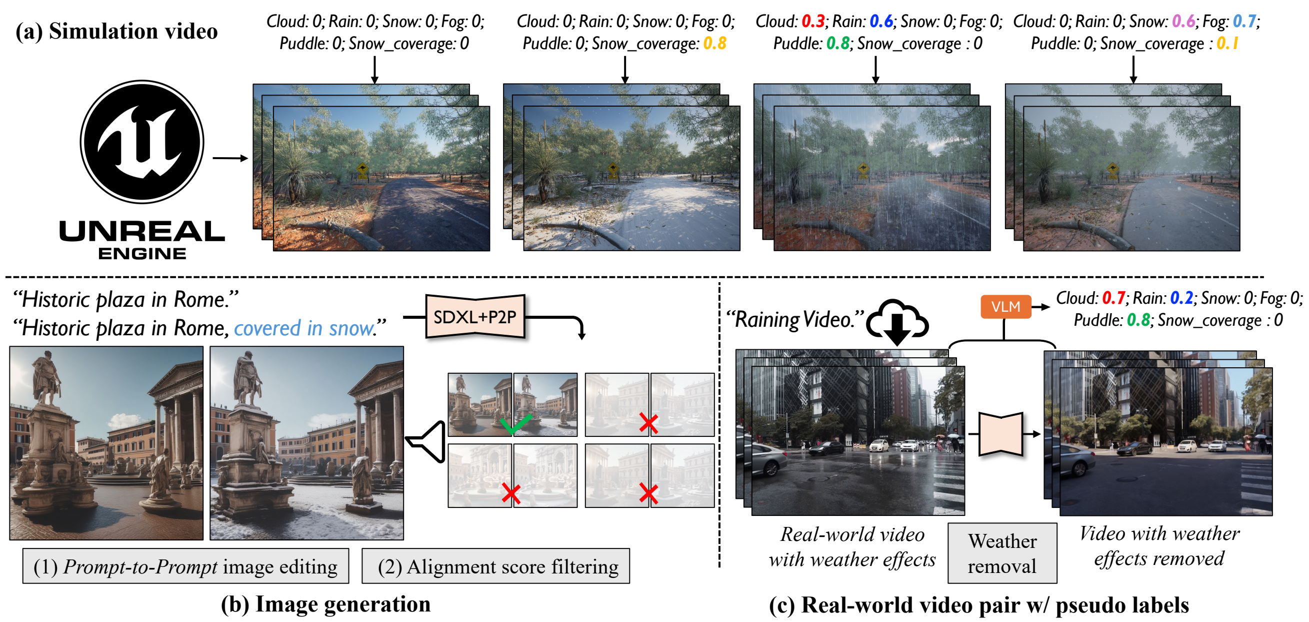



Figure 3. **Data Strategy.** We collect paired image and video data from (a) simulation engine, (b) text-to-image generative models with Prompt-to-prompt [30], and (c) auto-labeling real-world online videos.

indicate stronger manifestations (e.g., denser fog or heavier rain). The overall weather condition for a video is thus represented by the vector

$$\mathbf{s} = (s_{\text{cloud}}, s_{\text{fog}}, s_{\text{rain}}, s_{\text{snow}}, s_{\text{puddle}}, s_{\text{snow\_coverage}}) \in \mathbb{R}^6.$$

This parametric representation precisely captures weather variations and offers intuitive control over both the type and intensity of effects applied to the input video. By combining individual conditions, our model can synthesize a wide array of realistic weather conditions (Fig. 5, 7).

**Weather Synthesis** Given an input video $\mathbf{I}^c$ and a conditioning signal $\mathbf{s}$, our WEATHER SYNTHESIS MODEL outputs the synthesized video with desired weather effects $\hat{\mathbf{I}}^w$. We formulate weather synthesis as a conditional video generation task, and aim to approximate weather synthesis in a data-driven manner, allowing the model to operate on arbitrary input videos without relying on explicit 3D geometry.

Our WEATHER SYNTHESIS MODEL $\mathbf{f}_\theta^{c\to w}$ is initialized with the pre-trained weights of Stable Video Diffusion and operates in the VAE latent space. Specifically, for each data sample $(\mathbf{I}^c, \mathbf{I}^w, \mathbf{s})$, we encode both the input video $\mathbf{I}^c$ and the corresponding weather-affected video $\mathbf{I}^w$ into the latent space using the VAE encoder:

$$\mathbf{z}_0^c = \mathcal{E}(\mathbf{I}^c) \in \mathbb{R}^{l\times h\times w\times C}, \mathbf{z}_0^w = \mathcal{E}(\mathbf{I}^w) \in \mathbb{R}^{l\times h\times w\times C}$$

To represent the strength of the weather effect, we construct a condition map $\mathbf{S}$ by expanding the condition vectors across spatial and temporal dimensions $\mathbf{S} = \mathbb{1} \otimes \mathbf{s} \in \mathbb{R}^{l\times h\times w\times 6}$, where $\mathbb{1} \in \mathbb{R}^{l\times h\times w}$ denotes an all-one tensor.

During training, noisy video latents are obtained by adding Gaussian noise following the predefined noise schedule $\mathbf{z}_\tau^w = \alpha_\tau \mathbf{z}_0^w + \sigma_\tau \epsilon$. In each denoising step, the noisy latent $\mathbf{z}_\tau^w$, the video latent $\mathbf{z}_0^c$, and the weather strength map $\mathbf{S}$ are concatenated as input into the UNet denoising function $\mathbf{f}_\theta^{c\to w}$. To handle the concatenated input conditions, we add zero-initialized extra channels to the first convolution layer of the UNet. The model is optimized using the denoising score matching objective [36]:

$$\mathcal{L}^{c\to w} = \|\mathbf{f}_\theta^{c\to w}(\mathbf{z}_\tau^w; \mathbf{z}_0^c, \mathbf{S}, \tau) - \mathbf{z}_0^w\|_2^2 \tag{1}$$

**Weather Removal** is similarly formulated as a conditional video generation task, sharing the same architecture as the WEATHER SYNTHESIS MODEL. Given an input video with weather effects $\mathbf{I}^w$, and weather strengths $\mathbf{s}$ indicating the effects to remove, the WEATHER REMOVAL MODEL generates the corresponding clear-day video $\hat{\mathbf{I}}^c$.

During training, Gaussian noise is added to the clear-day video latent $\mathbf{z}_0^c$ to create noisy latent $\mathbf{z}_\tau^c$. The noisy latent is concatenated with the input video latent $\mathbf{z}_0^w$ and the weather strength map $\mathbf{S}$ to form the input for the UNet denoising function $\mathbf{f}_\theta^{w\to c}$. The training objective is defined as:

$$\mathcal{L}^{w\to c} = \|\mathbf{f}_\theta^{w\to c}(\mathbf{z}_\tau^c, \mathbf{z}_0^w, \mathbf{S}, \tau) - \mathbf{z}_0^c\|_2^2 \tag{2}$$

At inference time, both weather synthesis and removal models produce photorealistic edited videos by iteratively denoising Gaussian noise with learned denoising functions.

## 4.2. Data Collection

High-quality paired video data $(\mathbf{I}^c, \mathbf{I}^w, \mathbf{s})$ is essential for training our models, where $\mathbf{I}^c$ denotes clear-day videos without weather effects, $\mathbf{I}^w$ the corresponding videos with weather effects, and $\mathbf{s}$ represents the strength of these effects. Collecting such data in real-world scenarios is challenging, and existing public datasets [5, 7, 66] do not meet these specific requirements. To bridge this gap, we propose a data collection strategy that leverages three complementary sources: *Simulation*, *Generation*, and auto-labeled *Real-World Videos*. Table 1 summarizes the key properties of these sources, and Fig. 3 shows examples of the collected data.

**Simulation** To obtain paired video data with precise weather control, we use synthetic environments in Unreal Engine [19]. Specifically, we select four large-scale, artist-generated outdoor scenes consisting of city streets, wild forests, towns, and rural areas and simulate six weather effects at varying intensities. To mimic real-world conditions, we also randomly combine these individual effects.

We generate diverse camera trajectories by sampling an initial pose and then randomly selecting subsequent poses within defined spatial bounds, using collision detection to avoid asset intersections. Lighting was varied by randomly sampling environment maps covering different times of day.

By automating this workflow via Unreal Engine scripting, we produced 20.8k video pairs, each comprising 100 frames with labeled ground truth weather effects.

**Generation** High-quality synthetic assets are costly to obtain and often lack scene diversity. In contrast, generative models can synthesize a rich variety of data and scale with compute. To make use this resource, we follow Brooks et al. [8] and use Prompt-to-Prompt [30] in combination with SDXL [57] to generate paired images—with and without weather effects—while maintaining structural consistency.

Specifically, we use large language models [9, 54] to generate 61k scene descriptions (*e.g*. "A coastal road bordered by palm trees") and 10 pairs of weather-related captions for each of the six weather effects (*e.g*. "on a sunny day" versus "on a snowy day"). These paired captions enable us to generate image pairs through Prompt-to-Prompt. To synthesize varying weather intensities, we adjust the cross-attention weights for weather-related tokens (e.g., "snowy") and use these weights as strength labels (see [30] for further details).

We observed that the generative model often fails to adhere to the provided prompts. To address this, we filter the generated samples by measuring the consistency between image pairs and their corresponding caption pairs in the CLIP embedding space [61], following the approach in [8, 24]. We then select the top 4% of samples based on their consistency scores. For each selected sample, we generate 5 image pairs with varying effect strengths, resulting in a total of 1,147k high-quality paired images that capture diverse weather variations across numerous scenes.

Although this pipeline produces image pairs rather than video pairs, the diversity provided by these images significantly benefits our model. Extending attention-based techniques to text-to-video generation [7, 32, 87] is promising but demands considerably more resources and less scalable. Hence, we leave video-based data generation for future work.

**Real-world Videos** offer high diversity and realism, yet obtaining paired examples with and without weather effects remains challenging. To address this, we introduce an auto-labeling strategy that leverages the abundance of photorealistic weather videos available online to generate additional training data for our WEATHER SYNTHESIS MODEL.

Specifically, we collect online videos capturing significant weather events such as heavy rainstorms and snowfall. We then use our pre-trained WEATHER REMOVAL MODEL and generate corresponding weather-free versions, effectively transforming the input videos into clear-day sequences (see Fig.3). To label the weather effect strengths we use a vision-language model (VLM) [79] with in-context learning. By providing the VLM with simulation data examples and their corresponding strength labels, we instruct them to estimate weather effect strengths for the collected real-world videos.

In total, we collected and processed 4.6k video pairs (100 frames per video) that capture the realistic appearance and dynamic variations of diverse weather conditions.

### 4.3. Training Strategy

We use a multi-stage training strategy to combine the strengths of different data sources. We first train the WEATHER REMOVAL MODEL $\mathbf{f}_\theta^{w \to c}$ using a combination of simulation and generation data. Since the generation dataset contains only images, we perform image-video co-training by treating each image as a single-frame video. Once trained, we use the model and auto-label real-world videos by generating corresponding videos with weather effects removed.

For WEATHER SYNTHESIS MODEL $\mathbf{f}_\theta^{c \to w}$, we start by training on both simulation and generation data, enabling the model to learn precise control over weather effects. Finally, we jointly train $\mathbf{f}_\theta^{c \to w}$ on all three data sources of simulation, generation, and auto-labeled real-world video data.

## 5. Experiments

We extensively evaluate our method on real-world video sequences and compare with state-of-the-art. Both qualitative and quantitative results demonstrate the effectiveness of our approach for weather synthesis, removal, and downstream applications. Video results are included in the Supplement.

**Datasets** To evaluate generalization and ensure a fair comparison with baselines, we collect test video sequences from three distinct, non-overlapping sources: driving sequences from the Waymo Open Dataset [71], outdoor scenes from DL3DV [47], and casual in-the-wild videos from Pexels [1]. In total, we use 40 videos for weather synthesis and 55 videos (with fog, rain, or snow) for weather removal evaluation.

**Baselines** We compare our method with diffusion-based video editing approaches, including Text2Live [6], AnyV2V [40], TokenFlow [26], and FRESCO [84]. These works rely on text input for guidance. To enable scalable evaluation and reduce human bias, we use state-of-the-art VLM [79] to generate synthesis/removal prompts from the

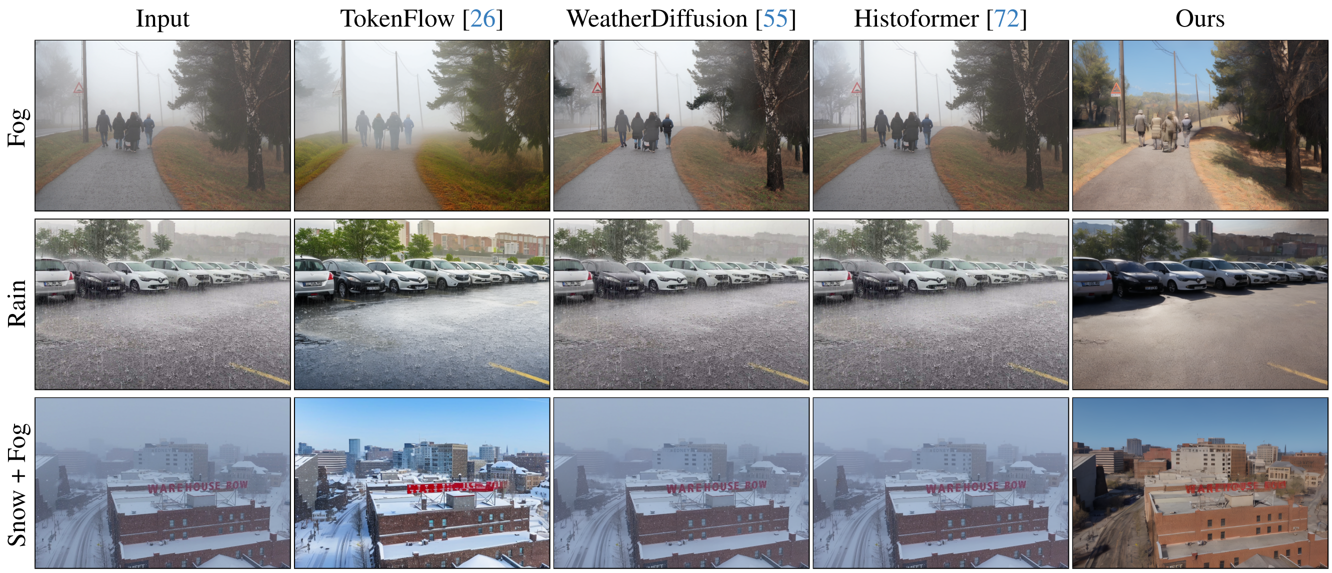



Figure 4. Qualitative comparison with state-of-the-art methods on weather removal.


Figure 5. Qualitative comparison with state-of-the-art methods on weather synthesis.

first frame of each input sequence. We also compare with specialized methods for weather removal, including WeatherDiffusion [55] and Histoformer [72]. Finally, we perform qualitative comparison with ClimateNeRF [43] on weather synthesis.

**Evaluation Metrics** For weather synthesis, all methods generate three effects (fog, rain, snow) for each input video. Our method uses a fixed effect strength of 1.0 to generate the results. To measure how well the output aligns with target effects, we use VLM [79] to estimate alignment scores (denoted as Align. VLM) based on weather descriptions, and measure the average cosine similarity of edited frames using CLIP [61] (denoted as Align. CLIP). Following prior works [15, 81], we also adopt PickScore [39], which estimates alignment with human preferences. Temporal consistency is evaluated using VBench++ [34, 35], which computes CLIP feature similarity across frames and evaluate motion smoothness using motion priors from video model [44]. Structure preservation is measured using the DINO Structure score (DINO Struct.), following [56, 74], with all scores multiplied by 100. Finally, we evaluate the perceptual quality of generated videos with a user study.

## 5.1. Quantitative Evaluation

Table 2 shows the quantitative comparison of weather synthesis and removal tasks compared with four baseline methods. Our method consistently outperforms all baselines in terms of Align. VLM, Align. CLIP, and PickScore, demonstrating its effectiveness in synthesizing diverse weather conditions and removing existing weather effects. For structure preservation (DINO Struct.), our method ranks second best in synthesis and third best in removal, suggesting that while videos are modified with weather change, the overall structure is preserved well. While WeatherDiffusion [55] and Histoformer [72] achieve higher structure preservation

Weather Synthesis

| Method | Align. VLM ↑ | Align. CLIP ↑ | PickScore ↑ | Temporal Consistency ↑ | Motion Smooth. ↑ | DINO Struct. ↓ |
|---|---|---|---|---|---|---|
| Text2Live [6] | 70.45 | **0.22** | 20.41 | **0.96** | **0.99** | 3.86 |
| AnyV2V [40] | 65.62 | 0.18 | 20.11 | 0.95 | 0.98 | 3.98 |
| TokenFlow [26] | 62.38 | 0.17 | 19.89 | **0.96** | 0.97 | **1.93** |
| FRESCO [84] | 70.23 | 0.18 | 19.81 | 0.95 | 0.98 | 2.42 |
| Ours | **77.29** | **0.22** | **20.75** | **0.96** | **0.99** | 2.30 |

Weather Removal

| Method | Align. VLM ↑ | Align. CLIP ↑ | PickScore ↑ | Temporal Consistency ↑ | Motion Smooth. ↑ | DINO Struct. ↓ |
|---|---|---|---|---|---|---|
| TokenFlow [26] | 66.39 | 0.15 | 19.07 | **0.98** | 0.98 | 2.20 |
| FRESCO [84] | 60.98 | 0.16 | 18.94 | 0.97 | 0.98 | 2.71 |
| WeatherDiffusion [55] | 22.79 | 0.15 | 18.82 | **0.98** | **0.99** | 0.26 |
| Histoformer [72] | 13.30 | 0.15 | 18.81 | **0.98** | **0.99** | **0.05** |
| Ours | **71.61** | **0.17** | **19.10** | **0.98** | **0.99** | 2.09 |

Table 2. Quantitative evaluation for weather synthesis and removal.

Weather Synthesis

| Baselines | Human Evaluator | | | VLM Evaluator | | |
|---|---|---|---|---|---|---|
| | Fog | Rain | Snow | Fog | Rain | Snow |
| AnyV2V [40] | $85\% \pm 24\%$ | $86\% \pm 18\%$ | $82\% \pm 19\%$ | $80\%$ | $70\%$ | $58\%$ |
| FRESCO [84] | $60\% \pm 17\%$ | $76\% \pm 4\%$ | $78\% \pm 23\%$ | $60\%$ | $50\%$ | $53\%$ |
| Text2Live [6] | $89\% \pm 4\%$ | $88\% \pm 10\%$ | $76\% \pm 19\%$ | $80\%$ | $80\%$ | $73\%$ |
| TokenFlow [26] | $59\% \pm 10\%$ | $66\% \pm 10\%$ | $67\% \pm 10\%$ | $58\%$ | $55\%$ | $50\%$ |

Weather Removal

| Baselines | Human Evaluator | | | VLM Evaluator | | |
|---|---|---|---|---|---|---|
| | Fog | Rain | Snow | Fog | Rain | Snow |
| AnyV2V [40] | $74\% \pm 6\%$ | $62\% \pm 21\%$ | $70\% \pm 7\%$ | $63\%$ | $75\%$ | $63\%$ |
| FRESCO [84] | $59\% \pm 6\%$ | $71\% \pm 15\%$ | $67\% \pm 22\%$ | $88\%$ | $65\%$ | $67\%$ |
| Text2Live [6] | $85\% \pm 17\%$ | $94\% \pm 11\%$ | $93\% \pm 12\%$ | $75\%$ | $90\%$ | $92\%$ |
| TokenFlow [26] | $52\% \pm 6\%$ | $65\% \pm 18\%$ | $75\% \pm 17\%$ | $50\%$ | $60\%$ | $58\%$ |
| Histoformer [72] | $82\% \pm 6\%$ | $80\% \pm 14\%$ | $82\% \pm 16\%$ | $75\%$ | $65\%$ | $75\%$ |
| WeatherDiffusion [55] | $89\% \pm 11\%$ | $87\% \pm 14\%$ | $87\% \pm 14\%$ | $100\%$ | $60\%$ | $75\%$ |

Table 3. **User study.** Evaluated by human and VLM evaluators, we report the percentage of samples where Ours is preferred over baselines. A preference $> 50\%$ indicates Ours outperforming baselines.

scores, their outputs often fail to remove weather effects, resulting in videos that are nearly identical to the inputs. This limitation is reflected in their lower alignment scores, PickScores, and the qualitative results shown in Fig. 4. The supplementary video shows that our method also demonstrates good temporal consistency and motion smoothness.

**User Study** We conducted a user study to assess the perceptual quality of our method's video outputs. Participants were shown the reference input video alongside two edited video results–one generated by our method and the other by a baseline model, with the order randomized. For each sample pair, we invited 11 users to perform binary selection from the video pairs, and used majority voting to determine the preferred video for each comparison. For the task of weather synthesis, users are instructed to select the video with more realistic weather effects. For weather removal, users select the videos with least visible weather effects. We repeat the full user study three times, and report the average percentage of samples where our method is preferred over baselines in Table 3. We also provide the standard deviation across the three experiments.

Additionally, following recent research on using VLMs as perceptual evaluators [82], we randomly extract a single frame of each video and conduct the same evaluation on *image* pairs using Qwen2.5-VL-72B [79] as the perceptual evaluator. Our method is consistently preferred by both human and VLM evaluators on both weather synthesis and removal tasks.

## 5.2. Qualitative Evaluation

Fig. 5 compares our weather synthesis results with state-of-the-art video editing models [26, 40, 84]. Our method effectively adapts lighting conditions for different weather, such as removing shadows and dimming lake reflections to simulate cloudy shading. Compared to baselines, our method introduces realistic weather elements that prior methods cannot handle, including reflective puddles, snow-covered roofs, falling snow and rain. Our approach preserves the overall structure by only modifying weather-related regions, while previous methods often change shapes, colors, and hallucinate new contents.


Figure 6. Controlling the strength of weather effects.

We compare weather removal methods in Fig. 4. TokenFlow [26] slightly changes the shading and synthesizes some background details, but struggles with strong fog, rain, puddle, and snow. WeatherDiffusion [55] and Histoformer [72] are designed to remove transient snow and rain, but since they are trained only on images with synthetic patterns [76], they do not generalize well to diverse real-world videos and cannot handle other weather effects such as fog, puddles, and snow coverage. In contrast, our method is trained on diverse data sources, and effectively generalize to various weather conditions. It not only removes weather effects but also generates realistic scene content and simulates natural shading, consistently transforming videos into a clear-day appearance. In Fig. 6, we control the fog density and puddle reflection by changing the corresponding effect strength, demonstrating the high controllability of our method. Please refer to the supplementary for the results of all six effects.

## 5.3. Ablation Study

We qualitatively ablate our method in Fig. 8. Compared to our full method, the image-model variant (*i.e.*, without temporal modules) often fails to generate transient effects such as falling raindrops and snowflakes.

We also ablate the benefit of each data source described in Sec. 4.2. When *simulation* data are excluded, the model struggles to control effects and shading precisely. Excluding

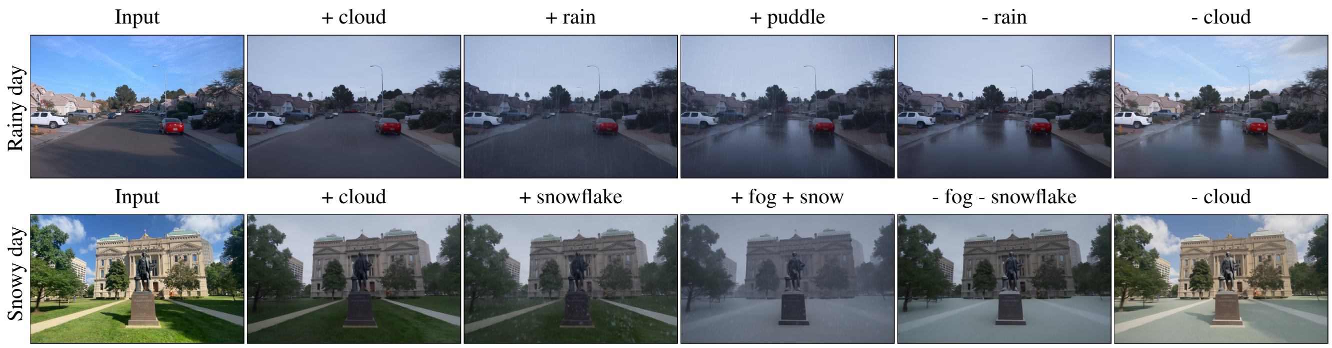



Figure 7. **Weather Editing with Multiple Effects.** Our method allows sequential application and combination of multiple effects. From left to right, we control the weather effect strengths and simulate how weather changes during rainy/snowy days.

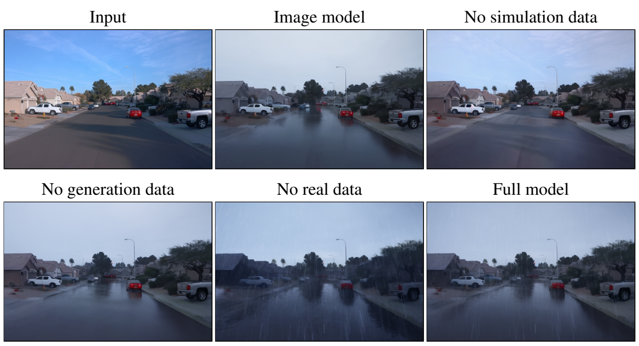



Figure 8. **Ablation Study.** Our video model formulation improves the quality of transient effects and temporal consistency. Joint training with all data sources produces the best results.


Figure 9. **Weather Editing.** Combined weather removal and synthesis models allow users to edit existing weather to different states.

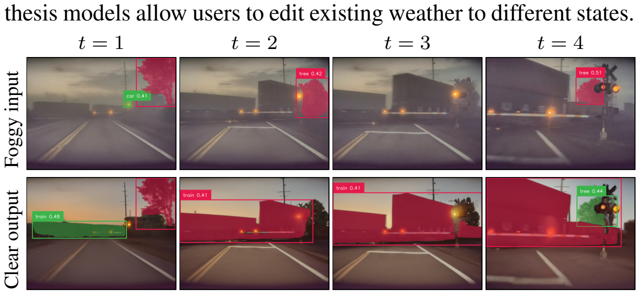



Figure 10. **Improved perception with weather removal.** After removing dense fog with our weather removal model, Grounded SAM [62] detects objects (e.g. train, tree) more accurately.

*generation* data impacts the generalization of specific effects, such as rain, leading to their absence in the output. Without *real-world* data, the generated videos often appears less realistic. In general, our full model combines a video-based approach with three diverse data sources, achieving the best quality and controllability.

### 5.4. Applications

Realistic weather editing in videos enables real-world applications. Combining both weather removal and synthesis models, our method enables weather editing by first applying the weather removal model, and re-generate weather effects with weather synthesis model in Fig. 9. Furthermore, in Fig.7, we show that our method can be sequentially applied to the same scene to simulate "time-lapse" sequences with diverse weather changes.

Effective weather removal also enhances the accuracy of perception models. In Fig. 10, Grounded-SAM [62] fails to detect trains in dense fog, but succeeds after applying our weather removal model, demonstrating potential applications for self-driving and robotics.

## 6. Conclusion

We propose a scalable, data-driven framework for controllable weather simulation in real-world videos. Drawing inspiration from modern graphics engines, we decompose the task into WEATHER REMOVAL and WEATHER SYNTHESIS and train two complementary conditional video diffusion models that can be applied independently or combined. By leveraging synthetic, generated, and automatically labeled real-world data in a unified training scheme, WEATHERWEAVER consistently outperforms state-of-the-art methods.

**Limitations** While WEATHERWEAVER demonstrates realistic, controllable, and temporally consistent weather synthesis and removal, its performance is bounded by the quality of the underlying Stable Video Diffusion model. Consequently, fine details such as text and facial features are not always preserved. Our model also struggles with nighttime videos, in part due to the scarcity of such footage in our current data-curation pipeline. Finally, Stable Video Diffusion is an offline model that can only process relatively short videos. With rapid progress in video diffusion quality and efficiency, we anticipate that integrating a more robust and efficient base model will lead to even stronger performance.

## References


[1] Pexels.com. https://www.pexels.com/. 5

[2] Niket Agarwal, Arslan Ali, Maciej Bala, Yogesh Balaji, Erik Barker, Tiffany Cai, Prithvijit Chattopadhyay, Yongxin Chen, Yin Cui, Yifan Ding, et al. Cosmos world foundation model platform for physical ai. *arXiv preprint arXiv:2501.03575*, 2025. 13

[3] Omri Avrahami, Dani Lischinski, and Ohad Fried. Blended diffusion for text-driven editing of natural images. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 18208–18218, 2022. 2

[4] Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, Humen Zhong, Yuanzhi Zhu, Mingkun Yang, Zhaohai Li, Jianqiang Wan, Pengfei Wang, Wei Ding, Zheren Fu, Yiheng Xu, Jiabo Ye, Xi Zhang, Tianbao Xie, Zesen Cheng, Hang Zhang, Zhibo Yang, Haiyang Xu, and Junyang Lin. Qwen2.5-vl technical report. *arXiv preprint arXiv:2502.13923*, 2025. 13

[5] Max Bain, Arsha Nagrani, Gül Varol, and Andrew Zisserman. Frozen in time: A joint video and image encoder for end-to-end retrieval. In *ICCV*, 2021. 5

[6] Omer Bar-Tal, Dolev Ofri-Amar, Rafail Fridman, Yoni Kasten, and Tali Dekel. Text2live: Text-driven layered image and video editing. In *ECCV*. Springer, 2022. 5, 7

[7] Andreas Blattmann, Tim Dockhorn, Sumith Kulal, Daniel Mendelevitch, Maciej Kilian, Dominik Lorenz, Yam Levi, Zion English, Vikram Voleti, Adam Letts, et al. Stable video diffusion: Scaling latent video diffusion models to large datasets. *arXiv preprint arXiv:2311.15127*, 2023. 3, 5, 13

[8] Tim Brooks, Aleksander Holynski, and Alexei A Efros. Instructpix2pix: Learning to follow image editing instructions. In *CVPR*, 2023. 5

[9] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *NeurIPS*, 2020. 5

[10] Bolun Cai, Xiangmin Xu, Kui Jia, Chunmei Qing, and Dacheng Tao. Dehazenet: An end-to-end system for single image haze removal. *IEEE transactions on image processing*, 25(11):5187–5198, 2016. 3

[11] Duygu Ceylan, Chun-Hao P Huang, and Niloy J Mitra. Pix2video: Video editing using image diffusion. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 23206–23217, 2023. 2

[12] 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)*. IEEE, 2023. 3

[13] Sixiang Chen, Tian Ye, Kai Zhang, Zhaohu Xing, Yunlong Lin, and Lei Zhu. Teaching tailored to talent: Adverse weather restoration via prompt pool and depth-anything constraint. In *European Conference on Computer Vision*, pages 95–115. Springer, 2024. 3

[14] 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 *ECCV*. Springer, 2020. 3

[15] Yuren Cong, Mengmeng Xu, Christian Simon, Shoufa Chen, Jiawei Ren, Yanping Xie, Juan-Manuel Perez-Rua, Bodo Rosenhahn, Tao Xiang, and Sen He. Flatten: optical flow-guided attention for consistent text-to-video editing. *ICLR*, 2024. 6

[16] Gautier Cosne, Adrien Juraver, Mélisande Teng, Victor Schmidt, Vahe Vardanyan, Alexandra Luccioni, and Yoshua Bengio. Using simulated data to generate images of climate change. *ICLR Workshop*, 2020. 2

[17] Qiyu Dai, Xingyu Ni, Qianfan Shen, Wenzheng Chen, Baoquan Chen, and Mengyu Chu. Rainygs: Efficient rain synthesis with physically-based gaussian splatting. *arXiv preprint arXiv:2503.21442*, 2025. 2

[18] Prafulla Dhariwal and Alexander Quinn Nichol. Diffusion models beat GANs on image synthesis. In *NeurIPS*, 2021. 3

[19] Epic Games. Unreal engine, 2019. 1, 5

[20] Patrick Esser, Johnathan Chiu, Parmida Atighehchian, Jonathan Granskog, and Anastasis Germanidis. Structure and content-guided video synthesis with diffusion models. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 7346–7356, 2023. 2

[21] Bryan E Feldman and James F O'Brien. Modeling the accumulation of wind-driven snow. In *ACM SIGGRAPH 2002 conference abstracts and applications*, 2002. 1, 2

[22] Ruoyu Feng, Wenming Weng, Yanhui Wang, Yuhui Yuan, Jianmin Bao, Chong Luo, Zhibo Chen, and Baining Guo. Ccedit: Creative and controllable video editing via diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6712–6722, 2024. 2

[23] Gal Fiebelman, Hadar Averbuch-Elor, and Sagie Benaim. Let it snow! animating static gaussian scenes with dynamic weather effects. *arXiv preprint arXiv:2504.05296*, 2025. 2

[24] Rinon Gal, Or Patashnik, Haggai Maron, Amit H Bermano, Gal Chechik, and Daniel Cohen-Or.

Stylegan-nada: Clip-guided domain adaptation of image generators. *ACM Transactions on Graphics (TOG)*, 2022. 5

[25] Kshitiz Garg and Shree K Nayar. Photorealistic rendering of rain streaks. *ACM Transactions on Graphics (TOG)*, 2006. 1

[26] Michal Geyer, Omer Bar-Tal, Shai Bagon, and Tali Dekel. Tokenflow: Consistent diffusion features for consistent video editing. *ICLR*, 2024. 2, 5, 6, 7, 15, 16

[27] Christoph Gissler, Andreas Henne, Stefan Band, Andreas Peer, and Matthias Teschner. An implicit compressible sph solver for snow simulation. *ACM Transactions on Graphics (TOG)*, 2020. 2

[28] Martin Hahner, Dengxin Dai, Christos Sakaridis, Jan-Nico Zaech, and Luc Van Gool. Semantic understanding of foggy scenes with purely synthetic data. In *IEEE Intelligent Transportation Systems Conference (ITSC)*, 2019. 2

[29] Ayaan Haque, Matthew Tancik, Alexei A Efros, Aleksander Holynski, and Angjoo Kanazawa. Instruct-nerf2nerf: Editing 3d scenes with instructions. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 19740–19750, 2023. 2

[30] Amir Hertz, Ron Mokady, Jay Tenenbaum, Kfir Aberman, Yael Pritch, and Daniel Cohen-Or. Prompt-to-prompt image editing with cross attention control. *arXiv preprint arXiv:2208.01626*, 2022. 2, 4, 5

[31] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *NeurIPS*, 2020. 3

[32] Wenyi Hong, Ming Ding, Wendi Zheng, Xinghan Liu, and Jie Tang. Cogvideo: Large-scale pretraining for text-to-video generation via transformers. *arXiv preprint arXiv:2205.15868*, 2022. 5

[33] Hao-Yu Hsu, Zhi-Hao Lin, Albert Zhai, Hongchi Xia, and Shenlong Wang. Autovfx: Physically realistic video editing from natural language instructions. *arXiv preprint arXiv:2411.02394*, 2024. 2

[34] Ziqi Huang, Yinan He, Jiashuo Yu, Fan Zhang, Chenyang Si, Yuming Jiang, Yuanhan Zhang, Tianxing Wu, Qingyang Jin, Nattapol Chanpaisit, Yaohui Wang, Xinyuan Chen, Limin Wang, Dahua Lin, Yu Qiao, and Ziwei Liu. VBench: Comprehensive benchmark suite for video generative models. In *CVPR*, 2024. 6

[35] Ziqi Huang, Fan Zhang, Xiaojie Xu, Yinan He, Jiashuo Yu, Ziyue Dong, Qianli Ma, Nattapol Chanpaisit, Chenyang Si, Yuming Jiang, Yaohui Wang, Xinyuan Chen, Ying-Cong Chen, Limin Wang, Dahua Lin, Yu Qiao, and Ziwei Liu. Vbench++: Comprehensive and versatile benchmark suite for video generative models. *arXiv preprint arXiv:2411.13503*, 2024. 6

[36] Tero Karras, Miika Aittala, Timo Aila, and Samuli Laine. Elucidating the design space of diffusion-based generative models. In *NeurIPS*, 2022. 3, 4

[37] Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3d gaussian splatting for real-time radiance field rendering. *ACM Transactions on Graphics*, 2023. 2

[38] Levon Khachatryan, Andranik Movsisyan, Vahram Tadevosyan, Roberto Henschel, Zhangyang Wang, Shant Navasardyan, and Humphrey Shi. Text2video-zero: Text-to-image diffusion models are zero-shot video generators. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15954–15964, 2023. 2

[39] Yuval Kirstain, Adam Polyak, Uriel Singer, Shahbuland Matiana, Joe Penna, and Omer Levy. Pick-a-pic: An open dataset of user preferences for text-to-image generation. *NeurIPS*, 2023. 6

[40] Max Ku, Cong Wei, Weiming Ren, Harry Yang, and Wenhu Chen. Anyv2v: A tuning-free framework for any video-to-video editing tasks. *arXiv preprint arXiv:2403.14468*, 2024. 5, 6, 7, 16

[41] Boyi Li, Xiulian Peng, Zhangyang Wang, Jizheng Xu, and Dan Feng. Aod-net: All-in-one dehazing network. In *Proceedings of the IEEE international conference on computer vision*, pages 4770–4778, 2017. 3

[42] Ruoteng Li, Robby T Tan, and Loong-Fah Cheong. All in one bad weather removal using architectural search. In *CVPR*, 2020. 3

[43] Yuan Li, Zhi-Hao Lin, David Forsyth, Jia-Bin Huang, and Shenlong Wang. Climatenerf: Extreme weather synthesis in neural radiance field. In *ICCV*, 2023. 2, 6, 13

[44] Zhen Li, Zuo-Liang Zhu, Ling-Hao Han, Qibin Hou, Chun-Le Guo, and Ming-Ming Cheng. Amt: All-pairs multi-field transforms for efficient frame interpolation. In *CVPR*, 2023. 6

[45] Feng Liang, Bichen Wu, Jialiang Wang, Licheng Yu, Kunpeng Li, Yinan Zhao, Ishan Misra, Jia-Bin Huang, Peizhao Zhang, Peter Vajda, et al. Flowvid: Taming imperfect optical flows for consistent video-to-video synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8207–8216, 2024. 2

[46] Ruofan Liang, Zan Gojcic, Huan Ling, Jacob Munkberg, Jon Hasselgren, Zhi-Hao Lin, Jun Gao, Alexander Keller, Nandita Vijaykumar, Sanja Fidler, and Zian Wang. Diffusionrenderer: Neural inverse and forward rendering with video diffusion models. *arXiv preprint arXiv: 2501.18590*, 2025. 2

[47] Lu Ling, Yichen Sheng, Zhi Tu, Wentian Zhao, Cheng Xin, Kun Wan, Lantao Yu, Qianyu Guo, Zixun Yu, Yawen Lu, et al. Dl3dv-10k: A large-scale scene dataset for deep learning-based 3d vision. In *CVPR*, 2024. 5

[48] Xiaohong Liu, Yongrui Ma, Zhihao Shi, and Jun Chen. Griddehazenet: Attention-based multi-scale network

for image dehazing. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 7314–7323, 2019. 3

[49] Yun-Fu Liu, Da-Wei Jaw, Shih-Chia Huang, and Jenq-Neng Hwang. Desnownet: Context-aware deep network for snow removal. *IEEE TIP*, 2018. 3

[50] Chenlin Meng, Yang Song, Jiaming Song, Jiajun Wu, Jun-Yan Zhu, and Stefano Ermon. Sdedit: Image synthesis and editing with stochastic differential equations. *arXiv preprint arXiv:2108.01073*, 2021. 2

[51] Ben Mildenhall, Pratul P. Srinivasan, Matthew Tancik, Jonathan T. Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. In *ECCV*, 2020. 2

[52] Eyal Molad, Eliahu Horwitz, Dani Valevski, Alex Rav Acha, Yossi Matias, Yael Pritch, Yaniv Leviathan, and Yedid Hoshen. Dreamix: Video diffusion models are general video editors. *arXiv preprint arXiv:2302.01329*, 2023. 2

[53] Thomas Müller, Alex Evans, Christoph Schied, and Alexander Keller. Instant neural graphics primitives with a multiresolution hash encoding. *ACM TOG*, 2022. 2

[54] OpenAI. Chatgpt: A conversational ai model, 2024. Accessed: 2025-03-04. 5

[55] Ozan Özdenizci and Robert Legenstein. Restoring vision in adverse weather conditions with patch-based denoising diffusion models. *IEEE TPAMI*, 2023. 3, 6, 7, 15

[56] Gaurav Parmar, Taesung Park, Srinivasa Narasimhan, and Jun-Yan Zhu. One-step image translation with text-to-image models. *arXiv preprint arXiv:2403.12036*, 2024. 6

[57] Dustin Podell, Zion English, Kyle Lacey, Andreas Blattmann, Tim Dockhorn, Jonas Müller, Joe Penna, and Robin Rombach. Sdxl: Improving latent diffusion models for high-resolution image synthesis. *arXiv preprint arXiv:2307.01952*, 2023. 2, 5

[58] Chenyang Qi, Xiaodong Cun, Yong Zhang, Chenyang Lei, Xintao Wang, Ying Shan, and Qifeng Chen. Fatezero: Fusing attentions for zero-shot text-based video editing. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15932–15942, 2023. 2

[59] Rui Qian, Robby T Tan, Wenhan Yang, Jiajun Su, and Jiaying Liu. Attentive generative adversarial network for raindrop removal from a single image. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2482–2491, 2018. 3

[60] Ruijie Quan, Xin Yu, Yuanzhi Liang, and Yi Yang. Removing raindrops and rain streaks in one go. In *CVPR*, 2021. 3

[61] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *ICML*, 2021. 5, 6

[62] Tianhe Ren, Shilong Liu, Ailing Zeng, Jing Lin, Kunchang Li, He Cao, Jiayu Chen, Xinyu Huang, Yukang Chen, Feng Yan, Zhaoyang Zeng, Hao Zhang, Feng Li, Jie Yang, Hongyang Li, Qing Jiang, and Lei Zhang. Grounded sam: Assembling open-world models for diverse visual tasks, 2024. 8

[63] Jenny Schmalfuss, Lukas Mehl, and Andrés Bruhn. Distracting downpour: Adversarial weather attacks for motion estimation. In *ICCV*, 2023. 2

[64] Victor Schmidt, Alexandra Luccioni, S Karthik Mukkavilli, Narmada Balasooriya, Kris Sankaran, Jennifer Chayes, and Yoshua Bengio. Visualizing the consequences of climate change using cycle-consistent adversarial networks. *ICLR*, 2019. 2

[65] Victor Schmidt, Alexandra Sasha Luccioni, Mélisande Teng, Tianyu Zhang, Alexia Reynaud, Sunand Raghupathi, Gautier Cosne, Adrien Juraver, Vahe Vardanyan, Alex Hernandez-Garcia, et al. Climategan: Raising climate change awareness by generating images of floods. *ICLR*, 2022. 2

[66] Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, et al. Laion-5b: An open large-scale dataset for training next generation image-text models. *NeurIPS*, 2022. 5

[67] Chaehun Shin, Heeseung Kim, Che Hyun Lee, Sang-gil Lee, and Sungroh Yoon. Edit-a-video: Single video editing with object-aware consistency. In *Asian Conference on Machine Learning*, pages 1215–1230. PMLR, 2024. 2

[68] Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *International Conference on Machine Learning*, 2015. 3

[69] Alexey Stomakhin, Craig Schroeder, Lawrence Chai, Joseph Teran, and Andrew Selle. A material point method for snow simulation. *ACM Transactions on Graphics (TOG)*, 2013. 1, 2

[70] Deborah Sulsky, Shi-Jian Zhou, and Howard L Schreyer. Application of a particle-in-cell method to solid mechanics. *Computer physics communications*, 1995. 2

[71] Pei Sun, Henrik Kretzschmar, Xerxes Dotiwalla, Aurelien Chouard, Vijaysai Patnaik, Paul Tsui, James Guo, Yin Zhou, Yuning Chai, Benjamin Caine, et al. Scal-

ability in perception for autonomous driving: Waymo open dataset. In *CVPR*, 2020. 5

[72] Shangquan Sun, Wenqi Ren, Xinwei Gao, Rui Wang, and Xiaochun Cao. Restoring images in adverse weather conditions via histogram transformer. *ECCV*, 2024. 3, 6, 7, 15

[73] Maxime Tremblay, Shirsendu Sukanta Halder, Raoul De Charette, and Jean-François Lalonde. Rain rendering for evaluating and improving robustness to bad weather. *IJCV*, 2021. 2

[74] Narek Tumanyan, Omer Bar-Tal, Shai Bagon, and Tali Dekel. Splicing vit features for semantic appearance transfer. In *CVPR*, 2022. 6

[75] Narek Tumanyan, Michal Geyer, Shai Bagon, and Tali Dekel. Plug-and-play diffusion features for text-driven image-to-image translation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1921–1930, 2023. 2

[76] Jeya Maria Jose Valanarasu, Rajeev Yasarla, and Vishal M Patel. Transweather: Transformer-based restoration of images degraded by adverse weather conditions. In *CVPR*, 2022. 3, 7

[77] Georg Volk, Stefan Müller, Alexander Von Bernuth, Dennis Hospach, and Oliver Bringmann. Towards robust cnn-based object detection through augmentation with synthetic rain variations. In *2019 IEEE intelligent transportation systems conference (ITSC)*. IEEE, 2019. 2

[78] Alexander Von Bernuth, Georg Volk, and Oliver Bringmann. Simulating photo-realistic snow and fog on existing images for enhanced cnn training and evaluation. In *2019 IEEE Intelligent Transportation Systems Conference (ITSC)*. IEEE, 2019. 2

[79] Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhihao Fan, Jinze Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Yang Fan, Kai Dang, Mengfei Du, Xuancheng Ren, Rui Men, Dayiheng Liu, Chang Zhou, Jingren Zhou, and Junyang Lin. Qwen2-vl: Enhancing vision-language model's perception of the world at any resolution. *arXiv preprint arXiv:2409.12191*, 2024. 5, 6, 7

[80] Haiyan Wu, Yanyun Qu, Shaohui Lin, Jian Zhou, Ruizhi Qiao, Zhizhong Zhang, Yuan Xie, and Lizhuang Ma. Contrastive learning for compact single image dehazing. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10551–10560, 2021. 3

[81] Jay Zhangjie Wu, Yixiao Ge, Xintao Wang, Stan Weixian Lei, Yuchao Gu, Yufei Shi, Wynne Hsu, Ying Shan, Xiaohu Qie, and Mike Zheng Shou. Tune-a-video: One-shot tuning of image diffusion models for text-to-video generation. In *ICCV*, 2023. 2, 6

[82] Tong Wu, Guandao Yang, Zhibing Li, Kai Zhang, Ziwei Liu, Leonidas Guibas, Dahua Lin, and Gordon Wetzstein. Gpt-4v(ision) is a human-aligned evaluator for text-to-3d generation. In *CVPR*, 2024. 7, 13

[83] Jie Xiao, Xueyang Fu, Aiping Liu, Feng Wu, and Zheng-Jun Zha. Image de-raining transformer. *IEEE transactions on pattern analysis and machine intelligence*, 45(11):12978–12995, 2022. 3

[84] Shuai Yang, Yifan Zhou, Ziwei Liu, and Chen Change Loy. Fresco: Spatial-temporal correspondence for zero-shot video translation. In *CVPR*, 2024. 5, 6, 7, 16

[85] Wenhan Yang, Robby T Tan, Jiashi Feng, Jiaying Liu, Zongming Guo, and Shuicheng Yan. Deep joint rain detection and removal from a single image. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1357–1366, 2017. 3

[86] Yijun Yang, Angelica I 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*, pages 13200–13210, 2023. 3

[87] Zhuoyi Yang, Jiayan Teng, Wendi Zheng, Ming Ding, Shiyu Huang, Jiazheng Xu, Yuanming Yang, Wenyi Hong, Xiaohan Zhang, Guanyu Feng, et al. Cogvideox: Text-to-video diffusion models with an expert transformer. *arXiv preprint arXiv:2408.06072*, 2024. 5, 13

[88] Tian Ye, Sixiang Chen, Jinbin Bai, Jun Shi, Chenghao Xue, Jingxia Jiang, Junjie Yin, Erkang Chen, and Yun Liu. Adverse weather removal with codebook priors. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 12653–12664, 2023. 3

[89] Yabo Zhang, Yuxiang Wei, Dongsheng Jiang, Xiaopeng Zhang, Wangmeng Zuo, and Qi Tian. Controlvideo: Training-free controllable text-to-video generation. *arXiv preprint arXiv:2305.13077*, 2023. 2

[90] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *ICCV*, 2017. 2

[91] Ruoxi Zhu, Zhengzhong Tu, Jiaming Liu, Alan C Bovik, and Yibo Fan. Mwformer: Multi-weather image restoration using degradation-aware transformers. *IEEE TIP*, 2024. 3

# Controllable Weather Synthesis and Removal with Video Diffusion Models

## Supplementary Material

In the supplementary material, we provide additional implementation details (Sec. A) and further results (Sec. B). Please refer to the project website for more qualitative results and comparisons.

## A. Implementation Details

Both weather removal and synthesis models are trained using AdamW optimizer with a learning rate of $3 \times 10^{-5}$ for 20k iterations. The models are trained on 32 A100 GPUs with fp16 mixed-precision for around 2 days. During training, the video resolution and number of frames are randomized at multiple scales, making the model robust to various input resolutions and frame lengths. The resolutions include $384 \times 576$, $512 \times 512$, $1280 \times 1920$, and the frame lengths range from 1 to 16. After the full training stages, the models can precisely control six effects (benefited from simulation data), generalize to diverse content (benefited from generation data), and simulate realistic weather (benefited from real-world data), supported by the evaluation in main Sec. 5.

## B. Additional Results

In Fig. S5, both our WEATHER SYNTHESIS MODEL and WEATHER REMOVAL MODEL effectively edit the weather, preserve details (e.g., "STOP" on the road), and also maintain temporal consistency. In addition, the different weather conditions can be controlled precisely by changing the strength values of each effect, shown in Fig. S4.

In addition to video editing methods, we also compare the weather synthesis with 3D simulation method in Fig. S1. ClimateNeRF [43] relies on the high-quality geometry to integrate weather effects with the scene successfully and cannot perform well for regions that are not captured densely (e.g., rooftop). On the other hand, our weather synthesis model leverages the video diffusion model and synthesizes snowflakes, snow coverage covering the whole scene. Furthermore, we provide additional qualitative results of weather removal and weather synthesis in Fig. S6 and Fig. S7, showing that our method generalize well to diverse video inputs.

**User Study** is a common approach for assessing perceptual realism. We conducted the user study mentioned in Sec. 5.1 on Amazon Mechanical Turk (MTurk) to compare our method with other baselines. Fig. S2 visualizes the example interface used for user study on the weather synthesis task. We asked users to make perceptual decisions on the pairwise comparison with the following criteria: 1) the integration of weather effects, 2) temporal consistency, and 3) content consistency. For weather removal, we used a similar user interface but asked users to choose videos with the least visible weather effects instead of the integration of weather effects.

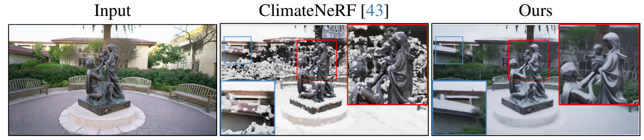



Figure S1. **Comparison with ClimateNeRF [43].** Our video model can coat delicate snow on the statue and rooftop surfaces, and also adjust the shading, which is hard for 3D simulation approaches [43].

During the user study, we invited 11 users for each sample pair to perform binary preference selection. We used 40 videos for weather synthesis (4 baselines, 3 effects) and 55 for weather removal (6 baselines) evaluation. This results in $3 \times 40 \times 4 \times 11 \times 3 = 15840$ and $55 \times 6 \times 11 \times 3 = 10,890$ user selections for each evaluated task. For each evaluated scene video, we did majority voting from 11 users to determine which method is more preferred in this scene. The majority voting can efficiently filter the effects of random users. The full experiments are repeated 3 times to calculate the mean and standard deviation on the preference percentage.

Inspired by [82], we also used large vision-language models (VLM) as perceptual evaluators to perform similar perceptual preference selections. For each pair of methods to be compared, we randomly selected a frame of the video and fed these frames into VLM, then asked VLM to give a binary preference selection with the same criteria as we used in the human user study. We used Qwen2.5-VL-72B [4] as our local VLM perceptual evaluator. For each sample pair, we run VLM 7 times with different random seeds. The final VLM preference of a scene video is determined by the same majority voting process. Fig. S3 demonstrates two example preference outputs from VLM.

**Failure Cases** We show failure cases of our models in Fig. S8. High-frequency details such as human faces are sometimes lost. This issue is primarily due to the limited capacity of our base model Stable Video Diffusion [7]. The VAE of Stable Video Diffusion has 8x spatial compression, leading to causes significant degradation and altering of image details. In contrast, recent tokenizers offer significantly improved fidelity [2, 87]. Our results appear to have reached Stable Video Diffusion's quality limit. Upgrading to a more powerful video model could significantly improve the overall quality.

Our data collection includes limited night-time videos, leading to potential imperfect simulation in these scenarios. Future work could improve visual quality by collecting additional specialized data.

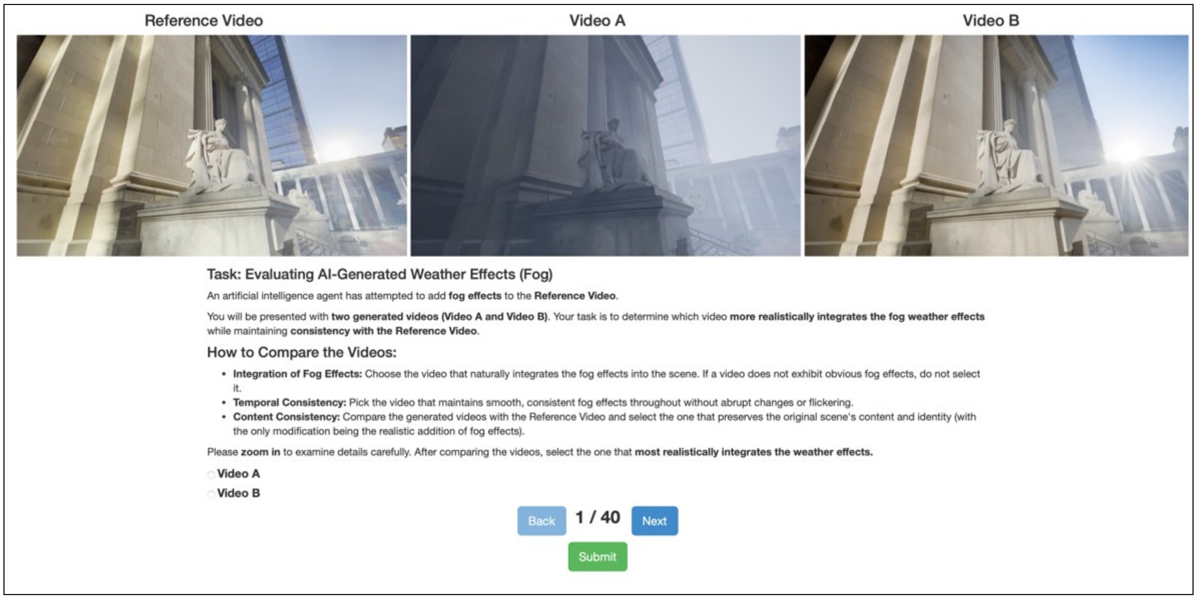
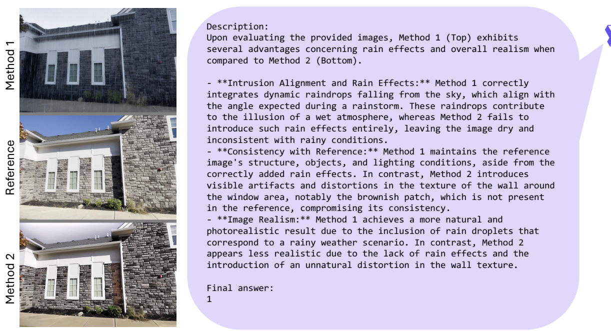
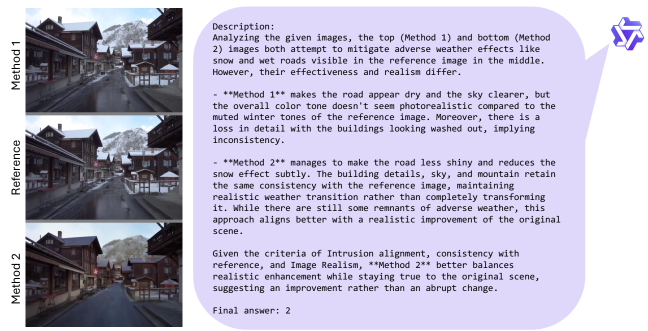



Figure S2. Example of user study interface for comparing two generated videos for weather synthesis.




(a) Weather Synthesis (Rain) Example: Ours vs. AnyV2V




(b) Weather Removal Example: HistoFormer vs. Ours

Figure S3. Examples on perceptual preference evaluation with VLM. We instructed VLM to first briefly describe the observation, then give the reason why it makes this decision.

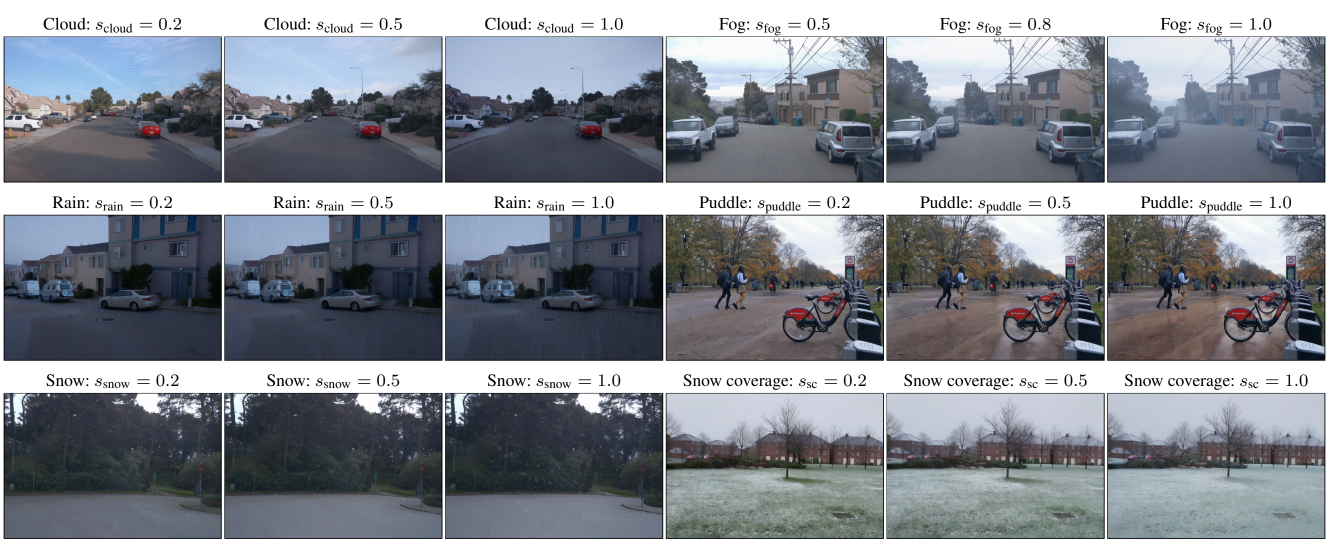



Figure S4. Controlling the strength of weather effects.


Figure S5. Temporally-Consistent Synthesis and Removal. Left: weather synthesis. Right: weather removal.

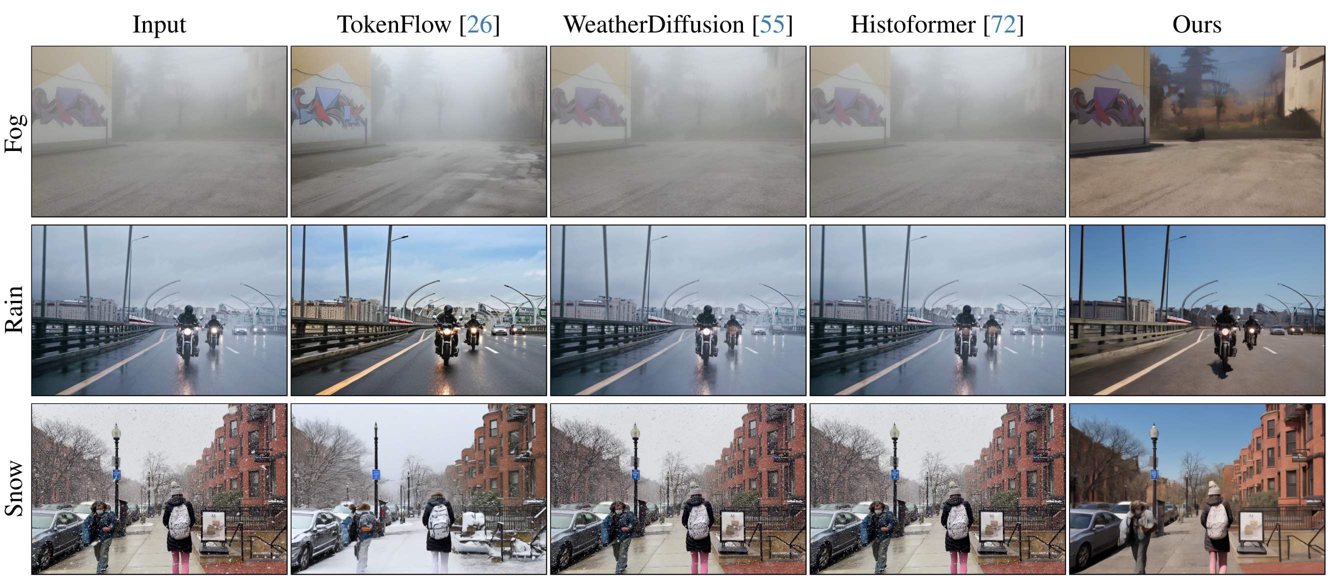



Figure S6. Additional qualitative results of weather removal.

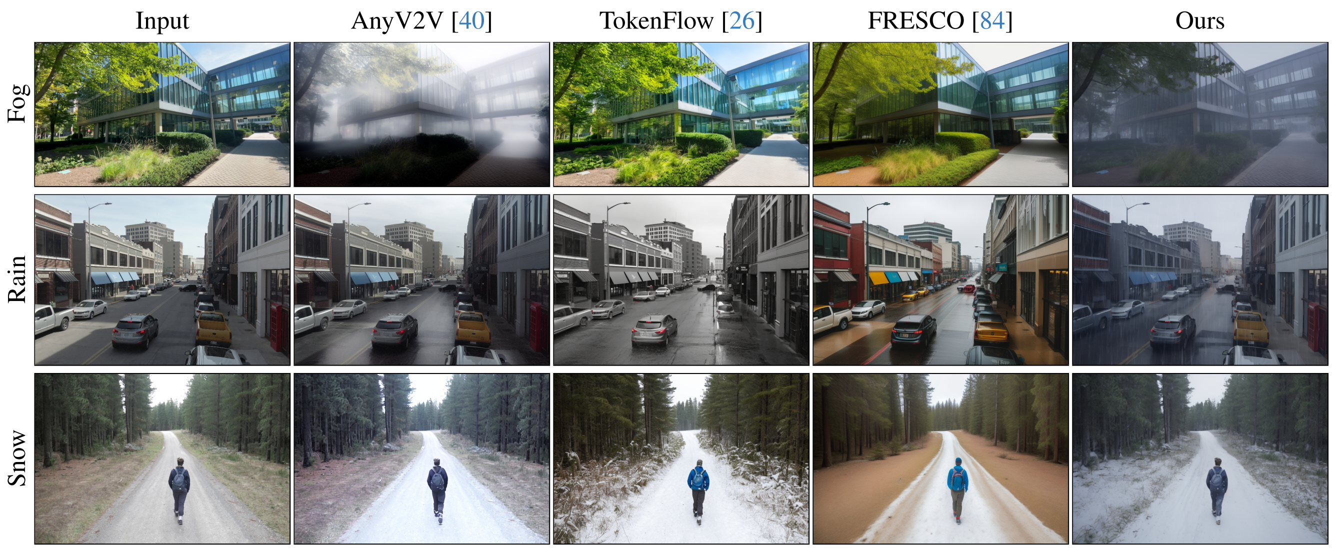



Figure S7. Additional qualitative results of weather synthesis.

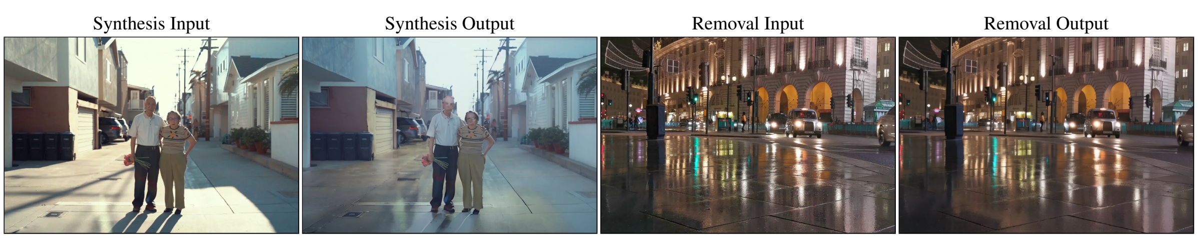



Figure S8. **Limitation.** Our method has a few failure cases, such as human facial details and night videos.