

WeatherStream: Light Transport Automation of Single Image Deweathering

Howard Zhang^{1*} Yunhao Ba^{1*} Ethan Yang¹ Varan Mehra¹ Blake Gella¹
 Akira Suzuki¹ Arnold Pfahnl¹ Chethan Chinder Chandrappa¹
 Alex Wong² Achuta Kadambi¹
¹University of California, Los Angeles ²Yale University
 {hwdz15508, yhba}@ucla.edu alex.wong@yale.edu achuta@ee.ucla.edu



(a) Snowy Input Image

(b) Model [57] trained on Previous Dataset

(c) Model [57] trained on Our Dataset

Figure 1. **We propose a new dataset, publicly available, that makes all image-based deweathering models we tested perform better.** In contrast to previous datasets, this dataset is not manually collected, but uses algorithms based on “light transport” [29] to automatically curate the dataset. Doing this algorithmically not only improves quality of the dataset, but also the scale. Download the dataset or test your model on the challenge website: <http://visual.ee.ucla.edu/wstream.htm/>.

Abstract

Today single image deweathering is arguably more sensitive to the dataset type, rather than the model. We introduce WeatherStream, an automatic pipeline capturing all real-world weather effects (rain, snow, and rain fog degradations), along with their clean image pairs. Previous state-of-the-art methods that have attempted the all-weather removal task train on synthetic pairs, and are thus limited by the Sim2Real domain gap. Recent work has attempted to manually collect time multiplexed pairs, but the use of human labor limits the scale of such a dataset. We introduce a pipeline that uses the power of light-transport physics and a model trained on a small, initial seed dataset to reject approximately 99.6% of unwanted scenes. The pipeline is able to generalize to new scenes and degradations that can, in turn, be used to train existing models just like fully human-labeled data. Training on a dataset collected through this procedure leads to significant improvements on multiple existing weather removal methods on a carefully human-collected test set of real-world weather effects. The dataset and code can be found in the following website: <http://visual.ee.ucla.edu/wstream.htm/>.

1. Introduction

Single-image deweathering aims to remove image degradations caused by rain, fog, or snow. Single-image

deweathering is a mainstay of modern computer vision, valued for the aesthetic appeal of removing weather degradations, as well as the ability to reuse pre-trained computer vision models, which work on clear weather conditions. Unfortunately, the field is dataset bottlenecked. State-of-the-art techniques use deep networks, but suffer from a common issue: the same scene cannot be observed at the same time, with and without weather artifacts. Therefore, it is not possible to train deep networks on **ideal pairs**, a pair of clean and degraded images of the same scene at the same time.

Previous work has attempted to solve the dataset bottleneck by using **simulated pairs**. A simulated pair is formed by starting with a clean image of a scene and artificially adding weather degradations. For example, one could carefully simulate the effect of raindrop streaks on a clean image. Unfortunately, simulating the diverse weather conditions that one can encounter is a very difficult path. Existing simulators for rain are difficult to generalize, and scaling simulators for rain, fog, and snow poses a further challenge. Nonetheless, simulated pairs have been the most common approach, and researchers have accepted the generalization errors that are encountered. Another emerging way to obtain pairs is to use **pseudo-real pairs**. A pseudo-real pair is a pair of clean and degraded images that is formed without the use of simulators. One way of doing so is to use a video-based deraining method to remove rain (which is dynamic) from a scene [60]. This form of ground-truthing

*Equal contribution.

assumes video-based deraining is itself a solved problem, which leads to limited performance, particularly in rain effects that are less dynamic (such as far-field veiling).

Perhaps the highest-quality pairs were obtained in the most recent work known as GT-RAIN [2]. This paper took a different tack, introducing **time multiplexed pairs**. A time multiplexed pair is obtained by taking an input video sequence and grabbing closely spaced frames in the video, with and without rain. This approach only works if nearby frames are grabbed in a **magic moment** when the scene conditions are just right, e.g., the rain is on the cusp of stopping, illumination constancy is observed, limited dynamic agents, and so on. In the less than one percent of videos that have suitable conditions, a time-multiplexed pair performs almost like real, ground truth.

Unfortunately, approaching time multiplexed pairing using human annotation (as has been done in previous work) is hard to scale to 100K+ pairs. As a generous lower bound, it would take 1 human labeler, 1 minute to carefully parse through 5 video sequences. Scaling this up to 2 million videos would take a whole year. Moreover, 99.6% of the video sequences do not meet the criteria for a magic moment. A further limit to scalability is that human observers must be highly trained to control for factors such as illumination shifts, weather API errors, and dynamic objects that are prevalent in over 99% of the videos.

In this paper, we formulate **light transport** techniques, which model the flow of light in a scene [29] to help us decide if frames should be included in training data. A key contribution is to formulate four principles of light transport to decide if a time-multiplexed pair is valid: (1) Background Conformity; (2) Particle Chromatic Variation; (3) Scatter-dependent Blur; and (4) Illumination Consistency.

Contributions: Our work is an initial attempt to use light transport to formulate how time multiplexed pairs should be selected, while the only previous approach is human annotated and limited to rain [2]. Automation scales the dataset, enabling us to obtain a dataset of 188K image pairs. This is the largest all-weather removal dataset to date, and includes diverse rain and snow of different shapes, sizes, and strengths, in various locations around the globe, with a plethora of backgrounds, camera settings, and illuminations. For this reason, we observe a 1.5 dB improvement in performance across various state of the art baselines. The dataset will be released conditional on acceptance.

2. Related work

Image deweathering aims to remove degradations caused by rain, fog, or snow for aesthetic appeal and the potential reuse of existing downstream task models, i.e. detection [27, 28], segmentation [6, 7], stereo [4, 12, 67], depth prediction [15, 19, 62], and completion [38, 42, 47, 64], trained on clear weather conditions.

Removing artifacts from a single weather type: Many

Dataset	Multi-weather?	# Pairs	Time Multiplexed?	Scalable?
All-in-One [32]	✓	111.6K	✗	✓
Snow100k [40]	✗	100K	✗	✓
SRRS [8]	✗	15K	✗	✓
CSD [9]	✗	10K	✗	✓
Rain100L [69]	✗	300	✗	✓
Rain100H [69]	✗	1.9K	✗	✓
Outdoor-Rain [31]	✗	10.5K	✗	✓
RainCityscapes [24]	✗	10.62K	✗	✓
Rain12000 [73]	✗	13.2K	✗	✓
Rain14000 [17]	✗	14K	✗	✓
GT-RAIN [2]	✗	31.5K	✓	✗
WeatherStream (Ours)	✓	202K	✓	✓

Table 1. **WeatherStream Dataset is the first automated time multiplexed dataset.** Compare with GT-RAIN [2] which is manually collected and for rain only; or the All-in-One weather dataset [32] which combines various simulated pairs [31, 40, 48].

previous works have focused on the removal of artifacts from a specific weather condition like rain [16, 20, 25, 26, 31, 46, 51, 58, 60, 63, 69, 72, 73] or snow [8, 9, 32, 35, 40, 68]. Most of these methods are trained on synthetic data, where rain streaks or snowflakes are added to images taken in clear weather – leaving a sim2real gap when transferring these models to real weather. Some works rely on unpaired real data through GANs [13, 54, 63] (see Sec. H in the Supp Mat). Recent work, known as GT-RAIN [2] collects a dataset of real rainy and clean image pairs separated by a time interval, but the process involves manually determining when the rain stopped and filtering thousands of unsuitable internet videos that violate their ground truth criterion. Hence, even curating a set of 30k images becomes unscalable. Our method, on the other hand, allows us to collect these image pairs at scale yielding a dataset that is larger than GT-RAIN; better in its quality than GT-RAIN; and inclusive of diverse weather conditions.

Removing artifacts from multiple weather types:

While most papers focus on removal of a specific weather type, there are some works that can remove multiple weather conditions in a single model [10, 32, 57]. One example is the all-in-one network [32] that uses a multi-encoder single-decoder setup with neural architecture search and a discriminator to decide which encoder to backpropagate losses to. Meanwhile, TransWeather [57] uses a single encoder and decoder, with self-attention within patches to remove smaller weather degradations. A teacher-student scheme has also been explored to train a unified all-weather removal model by using multiple teachers for each weather condition [10]. While there has been tremendous progress in weather removal architectures, these prior methods for rain, snow, and fog again rely on the synthetic deraining and desnowing pairs. This paper introduces time multiplexed pairs for rain, fog, and snow. A comparison of representative datasets for weather removal can be found in Tab. 1.

3. Light Transport of Time Multiplexing

Time multiplexing pairs are a nascent form of training data, introduced only in one previous paper by Ba *et al.* [2]. This paper produced surprising visual improvements in image deraining results when compared to simulated pairs. Unfortunately, this previous work only works for rain; uses human annotation to grab time multiplexed pairs; and due to manual curation, consists of only a few scenes. Collecting the Ba *et al.* dataset is a laborious effort because less than one percent of time multiplexed pairs are suitable, due to changes in lighting or motion.

In this paper, we draw from a foundation of light transport principles to automatically determine what makes an informative time multiplexed pair that we can use for training data. We rely on four principles of light transport.

Principle 1 (Background Conformity): *Objects in a suitable paired scene exhibit no motion and color constancy.* A paired scene with dynamic illumination/objects will not work. This creates brightness variations across the pair that are not a function of weather effects. But how do we know if a brightness variation is due to changes in weather or the environment? To address the problem, we appeal to forward models of image-based changes that weather can create i.e. for rain streaks [2, 11, 17, 30–32, 34, 57, 58, 60, 70, 73, 75]:

$$\mathbf{K}_r(x) = \mathbf{J}(x) + \sum_i^n \mathbf{S}_i(x), \quad (1)$$

where $\mathbf{K}_r \in \mathbb{R}^{3 \times H \times W}$ represents the rainy image, $\mathbf{J} \in \mathbb{R}^{3 \times H \times W}$ represents the clean image, \mathbf{S}_i represent the different rain streak layers, and x represents the spatial location on the image. A similar equation holds for synthetic snow particles [8, 9, 40] of the form:

$$\mathbf{K}_s(x) = \mathbf{J}(x)(1 - \mathbf{Z}(x)) + \mathbf{C}(x)\mathbf{Z}(x), \quad (2)$$

where $\mathbf{K}_s \in \mathbb{R}^{3 \times H \times W}$ represents the snowy image, \mathbf{Z} represents a snow mask, and \mathbf{C} represents a chromatic aberration map. According to [18, 45, 60, 74], these rain streak layers $\mathbf{S}_i(x)$ and snow particle masks $\mathbf{Z}(x)$ are randomly distributed spatially throughout the scene. As a property of these equations, the only discrepancy between \mathbf{K} and \mathbf{J} is the addition of weather effects. In accordance with this principle, we design blocks in our pipeline to detect motion and static object variations between the clean and degraded images that are not related to brightness changes caused by Eq. (1) and Eq. (2).

Principle 2 (Particle Chromatic Variation): *Rain streaks and snowflakes exhibit isotropic derivatives in RGB pixel intensity [74].*

The photometry of falling particles is consistent with Eq. (3) for particle intensity I_p :

$$I_p = \int_0^\tau E_p dt + \int_\tau^T E_{bg} dt, \quad (3)$$

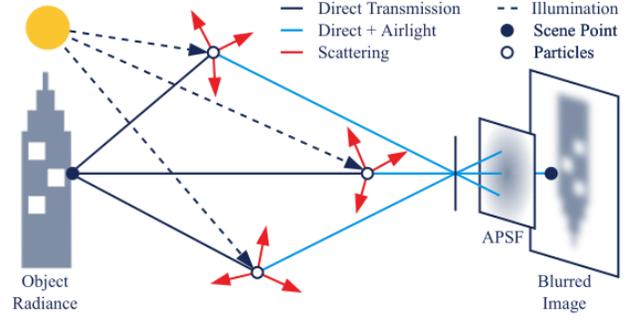


Figure 2. **Objects in adverse weather tend to be blurred due to the existence of scattering effects (Principle 3).** Figure adapted from multiscatter models in [22, 33, 44].

where T denotes the exposure time of the camera and τ the time in which the falling particle is in frame [18, 45]. Here, E_p and E_{bg} denote the irradiance of the particle and the background, respectively. Utilizing *Principle 1*, we know that E_{bg} should be constant throughout the exposure time. We can then obtain an expression for $I_p - I_{bg} = \Delta I$, where I_{bg} is the pixel intensity of the background:

$$I_p = \tau \bar{E}_p + (T - \tau) E_{bg}, \quad (4)$$

$$\bar{E}_p = \frac{1}{\tau} \int_0^\tau E_p dt, \quad (5)$$

$$\Delta I = \tau(\bar{E}_p - E_{bg}). \quad (6)$$

Empirical studies have shown that both rain and snow are bright regardless of background irradiance (as a result of compounding internal and specular reflections with a large refraction field) [18, 45, 60, 74]. Therefore, we can assume \bar{E}_p to be constant, yielding a linear relationship between ΔI and background intensity I_{bg} :

$$\Delta I = -\beta I_{bg} + \alpha, \beta = \frac{\tau}{T}, \alpha = \tau \bar{E}_p. \quad (7)$$

The implication of *Principle 2* is that one can write the intensity between clean and degraded image data $I_p - I_{bg}$ as being linearly related to the background intensity I_{bg} . This is true across color channels, such that $\Delta R, \Delta G, \Delta B$ of pixel intensities in the three color channels should be similar for weather artifacts [56, 74]. One can use this principle to infer that $[\Delta R, \Delta G, \Delta B]^T$ should be isotropic, and anisotropic deviations in this vector provide a clue about brightness variations in scenes that are not due to weather (again, such as objects or lighting).

Principle 3 (Scatter-dependent Blur): *A degraded image is blurrier than a clean image, due to scattering effects.*

We need additional principles beyond Eq. (1) and Eq. (2), as these two equations do not represent the gamut of complex rain and snow related phenomenon [2, 8, 9, 25, 31]. For example, dense rain, snow accumulation, haze or fog may

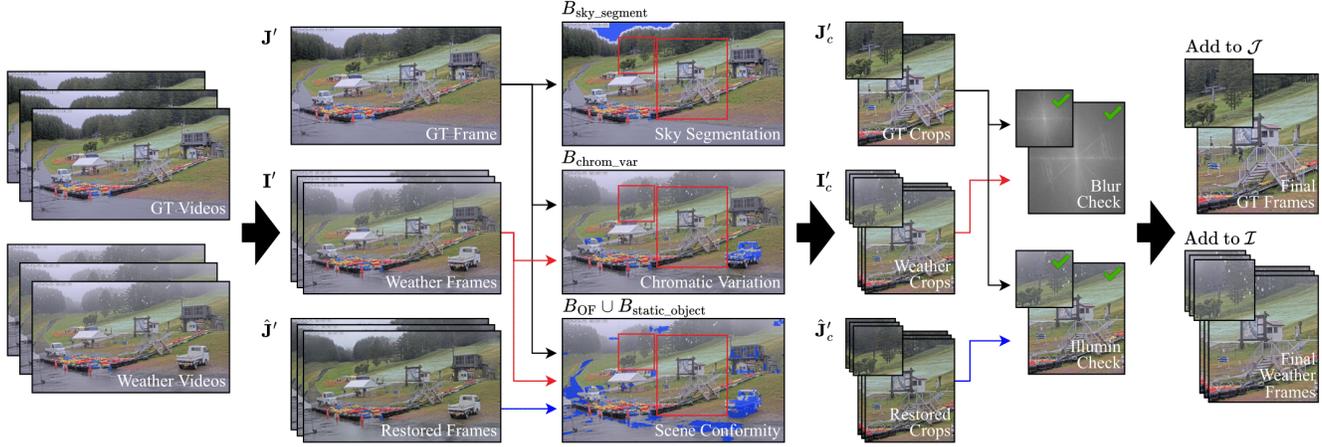


Figure 3. **The proposed WeatherStream pipeline utilizes light transport principles to identify magic moments for various weather effects.** It is capable of selecting high quality paired data from millions of candidate videos with diverse situations.

cause a “veiling” effect throughout the scene, due to the introduction of scattering effects [3, 8, 21, 24, 31, 55], which are not modeled by Eq. (1) and Eq. (2). To extend this to a single scattering model (SSM), one can define the radiance L_s of the scene at a distance d as:

$$\begin{aligned} L_s(d) &= L_o e^{-\int_0^d \beta dl} + \int_0^d L_\infty \beta e^{-\beta l} dl \\ &= L_o e^{-\beta d} + L_\infty (1 - e^{-\beta d}). \end{aligned} \quad (8)$$

Here, L_o represents the true underlying radiance and L_∞ the radiance of the airlight at the horizon. β is an atmospheric attenuation coefficient, which is constant throughout the scene. Though this approximation is in dehazing [21, 55], the model can be further refined to include multiple scattering effects (when scattered light is scattered again by different particles) [22, 33, 44, 59]:

$$L_m(d) = L_s(d) * \text{APSF}(d). \quad (9)$$

Here, the single scattering output radiance is convolved with the atmospheric point spread function (APSF), which accurately simulates the effects of multiple scattering, as illustrated in Fig. 2. Mapping this multiple scattering model to the RGB intensity space, and substituting rain and snow models for underlying radiance, allows us to obtain the following light transport model:

$$\mathbf{I}(x) = [\mathbf{K}(x)e^{-\beta d(x)} + \mathbf{L}_\infty(1 - e^{-\beta d(x)})] * h_{\text{APSF}}, \quad (10)$$

where distance d is a function of spatial location x , and \mathbf{K} represents the underlying scene with no fog effects (but with rain streaks and snowflakes). The h_{APSF} is a convolution kernel that represents the convolution operation with the APSF. From this equation, we arrive at *Principle 3*: Since the APSF can be approximated with a generalized gaussian distribution (GGD) [22, 43], we know that it has a low-pass

filter effect on our image, leading to a blur effect and lowered contrast. The exact parameters of this GGD approximation are dependent on particle size and particle atmospheric density, and are used in our pipeline to provide better scraping of time multiplexed pairs from webcam videos.

Principle 4 (Illumination Consistency): *The ambient illumination should remain consistent despite weather effects. This is difficult to analyze in closed-form since illumination stimuli to uncontrolled scenes are not known a priori.*

From *Principle 1* and Eq. (10), we know that the ambient illumination L_∞ should be consistent between clean and degraded images. In reality, this property is difficult to quantify, since veiling effects from falling particles contribute to a global intensity change which can be confounded with illumination shifts in an uncontrolled scene. The fourth principle is that we cannot predict this illumination shift, and use a seed model to learn whether the illumination has shifted with a sufficiently high safety margin to avoid false positive selection. This is further described in Sec. 4.

4. Pipeline to Obtain Time Multiplexed Pairs

We now describe our automated pipeline to build WeatherStream. We first begin with a seed dataset from previous work [2]. This seed dataset enables a model to be trained (“seed model”). The seed model is not used for the challenging task of deweathering, but to predict coarser light transport parameters. We detail the pipeline in Fig. 3 and explain how the light transport principles are used (Sec. 3).

Data Filtering: Without loss of generality, let us attempt to find a single pair of images (c, d) that represent a clean and degraded image, respectively.

The first goal is to download video streams within which, are candidates for the clean and degraded pair. This is done by analyzing attached weather metadata, discussed in Sec. 5. This yields a scraped data structure: $\mathcal{D}_c =$

$\{\tilde{\mathbf{c}}^n\}_{n=1}^N, \mathcal{D}_d = \{\tilde{\mathbf{d}}^m\}_{m=1}^M$, where the notation tilde in $\tilde{\mathbf{c}}^n, \tilde{\mathbf{d}}^m$ is used to emphasize that these are candidate clean and degraded images, respectively. Within these sets $\mathcal{D} := \mathcal{D}_c \cup \mathcal{D}_d$, we seek to find a suitable pair (\mathbf{c}, \mathbf{d}) that meets light transport constraints. Therefore, we seek an operator:

$$f_{\text{WeatherStream}} : \mathcal{D} \rightarrow (\mathbf{c}, \mathbf{d}). \quad (11)$$

Since the size of \mathcal{D} is larger than a possible 2-image pair (\mathbf{c}, \mathbf{d}) , the operator $f_{\text{WeatherStream}}$ acts as a filter. We filter \mathcal{D}_c and \mathcal{D}_d through four blocks, that each map to one of four light transport principles from the previous section.

Filtering Block One: Scene-level Verification: The first verification block conforms our data to *Principle 1* (Sec. 3). Images that exhibit object motion are filtered by computing optical flow between two temporally averaged and adjacent time frames. In particular, we compute the Gunnar-Farneback algorithm [14]. To avoid spurious motion in the flow field, we take the mean of consecutive frames that serve to average out rain streaks or snowflakes. As noted in Sec. 3, rain streaks and snowflakes are typically randomly distributed throughout the scene, and move at a much higher velocity compared to movements from scene objects. Prior works have shown that falling particles reach a terminal velocity of $200\sqrt{r}$, where the radius r of the falling particle is typically between 0.5-2.5 mm for rain drops and snowflakes [18, 45]. Due to the speed of these rain drops, and the framerate of our videos, the same streak will not appear in the same location in consecutive frames [74]. Thus, temporal averaging will remove weather particles. The magnitude of the resulting optical flow map is then thresholded through an indicator function to provide a binary map \mathbf{B}_{OF} representing motion regions in the image. Then, we check for the presence of static objects that are different between $\tilde{\mathbf{c}} \in \mathcal{D}_c$ and $\tilde{\mathbf{d}} \in \mathcal{D}_d$ using the following:

$$\mathbf{1}_{\text{static}}^{\tilde{\mathbf{c}}, \tilde{\mathbf{d}}}(x) = \begin{cases} 1 & \text{if } |\tilde{\mathbf{c}}(x) - \tilde{\mathbf{d}}(x)| > \gamma_{\text{static}}, \\ 0 & \text{otherwise.} \end{cases} \quad (12)$$

$$\hat{\mathbf{d}} = f_{\text{weather_removal}}(\tilde{\mathbf{d}}), \quad (13)$$

$$\mathbf{B}_{\text{static}}(x) = \mathbf{1}_{\text{static}}^{\tilde{\mathbf{c}}, \hat{\mathbf{d}}}(x) \text{ for } \tilde{\mathbf{c}} \in \mathcal{D}_c, \tilde{\mathbf{d}} \in \mathcal{D}_d. \quad (14)$$

The model, denoted by $f_{\text{weather_removal}}$, removes weather effects from $\tilde{\mathbf{d}}$ leaving object discrepancies as the only remaining variability between clean and degraded images, which is reflected in the output binary map $\mathbf{B}_{\text{static}}$. $\gamma_{\text{static}} = 0.1$, and is tuned such that the presence of rain streaks or snowflakes will not filter the scene. In addition to $\mathbf{B}_{\text{static}}$, we also obtain a \mathbf{B}_{sky} by running a sky segmentation algorithm [53] to avoid picking sky regions, as these are not typically regions that contain rain, snow, or fog weather effects. We then take their union i.e. $\mathbf{B} = \mathbf{B}_{\text{static}} \cup \mathbf{B}_{\text{OF}} \cup \mathbf{B}_{\text{sky}}$ and find all crops bigger than 256×256 that satisfies the complement of \mathbf{B} in $\tilde{\mathbf{c}} \in \mathcal{D}_c$ and $\tilde{\mathbf{d}} \in \mathcal{D}_d$.

Filtering Block Two: Color Verification: We now pass what remains of \mathcal{D} into another filtering block to remove false positive pairs. This second block leverages *Principle 2* from Sec. 3 to avoid false detection of rain streaks and snowflakes by ensuring isotropic intensity variations across RGB channels. The filtering function we use takes the form of:

$$\mathbf{C}_{\text{max}}(x) = \max_{c \in \{r, g, b\}} \tilde{\mathbf{d}}_c(x) - \tilde{\mathbf{c}}_c(x), \quad (15)$$

$$\mathbf{C}_{\text{min}}(x) = \min_{c \in \{r, g, b\}} \tilde{\mathbf{d}}_c(x) - \tilde{\mathbf{c}}_c(x), \quad (16)$$

$$\mathbf{B}_{\text{chrom_var}} = (\mathbf{C}_{\text{max}} - \mathbf{C}_{\text{min}}) > \gamma_{\text{chrom_var}}. \quad (17)$$

We threshold with $\gamma_{\text{chrom_var}}$ the gap between the maximum and minimum color channel intensities \mathbf{C}_{max} and \mathbf{C}_{min} . This $\gamma_{\text{chrom_var}}$ is set to .2 (for images ranging from 0 to 1). Removing pixels which have a high gap between $\Delta R, \Delta G$, or ΔB catches static object differences present in the image pairs while preserving rain streaks and snowflakes.

Filtering Block Three: Multi-scatter Verification: The third filtering block is designed to verify that multi-scatter exists in a possible image that could be a candidate for the degraded part of the pair, \mathbf{d} . According to *Principle 3* (Sec. 3), degraded images in all weather conditions contain a veiling effect which is represented through a multi-scattering model. An approximation of the accompanying APSF as a generalized Gaussian distribution leads us to consider two metrics: lower contrast and blur. In order to measure contrast and blur of a scene, we employ two commonly used metrics [37, 55]. The first metric f_{grad} is a normalized measurement of the difference between the magnitudes of the gradients of $\tilde{\mathbf{c}}$ and $\tilde{\mathbf{d}}$. We approximate the gradients using Sobel filters as:

$$\bar{\mathbf{d}} = \frac{1}{m} \sum_{i=0}^m \tilde{\mathbf{d}}, \quad (18)$$

$$f_{\text{grad}} = \frac{|\nabla \bar{\mathbf{d}}| - |\nabla \tilde{\mathbf{c}}|}{|\nabla \tilde{\mathbf{c}}|}. \quad (19)$$

The second metric f_{fit} is a normalized measurement of the difference between the magnitudes of the low-pass filtered reconstructions of $\tilde{\mathbf{c}}$ and $\bar{\mathbf{d}}$.

$$f_{\text{fit}} = \frac{|f_{\text{lp}}(\bar{\mathbf{d}}, \gamma_{\text{lp}})| - |f_{\text{lp}}(\tilde{\mathbf{c}}, \gamma_{\text{lp}})|}{|f_{\text{lp}}(\tilde{\mathbf{c}}, \gamma_{\text{lp}})|}. \quad (20)$$

The function f_{lp} is the log scale magnitude spectrum of a low-pass filtered reconstruction of an image with cutoff frequency γ_{lp} . Low-frequency components are filtered as per [37]. The chosen cutoff frequency depends on characterization of the APSF, which is dependent on two factors: optical thickness T , which is a function of particle density in the atmosphere, and forward scattering parameter q , which is a function of the particle size [59]. The

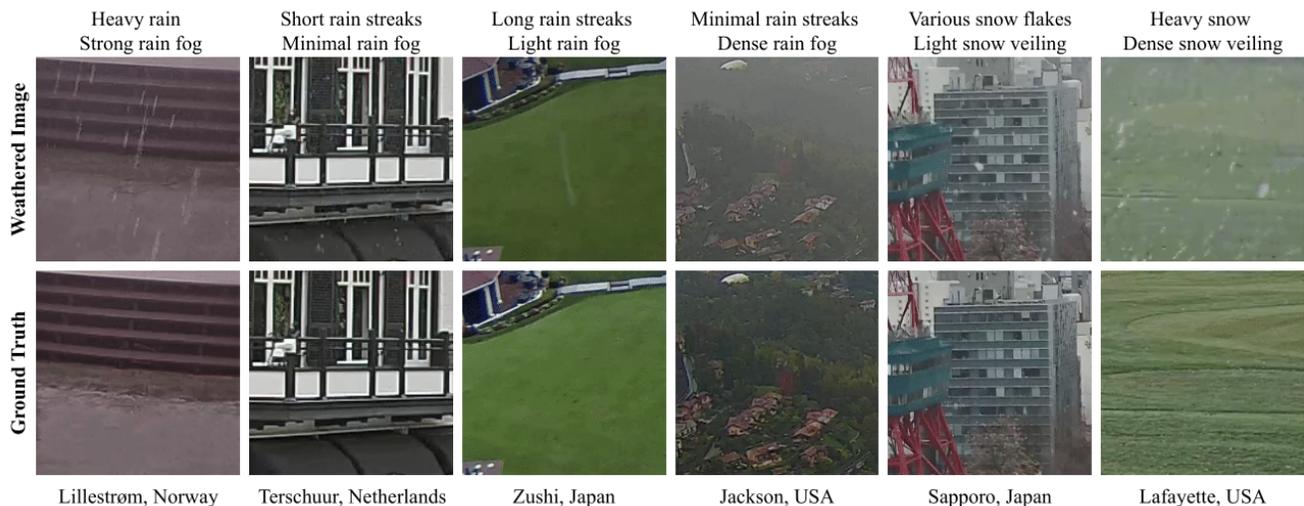


Figure 4. **WeatherStream Dataset contains diverse weather effects under rain, fog, and snow conditions.** Visualization highlights some representative image pairs with various weather effects across the world.

exact location of the cutoff we use is found through empirical analysis of the range of these factors in our dataset. For both of these metrics, we subtract degraded sharpness from clean sharpness, normalize, and reject scenes in which the degraded image is sharper than the clean image. To leave a gap for noise, we do not filter a scene if the degraded image is below 5% sharper than the clean image.

Filtering Block Four: Illumination Verification: As noted in *Principle 4* (Sec. 3), a particularly difficult part of the pipeline is determining whether a global change of intensity arose due to the veiling effect, or a stimulus change of illumination that may occur when time multiplexing. Here, we bring in the seed model comes into play [2]. The seed model can be adapted to distinguish between illumination shifts and veiling effects, indicating that training on a small seed dataset conveys enough scene understanding for the model to generate a coarse binary decision boundary to assess if the illumination is consistent between frames. The details of the network used are deferred to the supplement.

Final Output: Frames that remain in \mathcal{D}_c and \mathcal{D}_d are assigned as pairs based on temporal closeness within a hysteresis threshold.

5. WeatherStream Dataset

Scraping Weather Videos with Metadata: To download videos, we built a webscraper to automatically collect and maintain a database of over 5000 YouTube live streams and corresponding geographic coordinates. These streams cover a diverse range of scenes, camera parameters, and unique locations spanning 94 countries. Streams must adhere to a minimum resolution of 1280×720 pixels and a minimum frame rate of 30fps to be considered for inclusion. A quantitative evaluation of the streams used in our pipeline shows an NIQE score of 2.938 (lower is better), comparable to other weather removal datasets such as

Snow100K [40] (2.904) or notable detection datasets like COCO (3.648). We continuously determine the level of daylight and weather condition of the scene using the OpenWeatherMap API [41]. If a scene is between sunrise and sunset, and contains an adverse weather condition, a ten second video is downloaded with detailed weather metadata. Candidate videos are scraped for an additional hour when the weather condition ends. We expect one to process over 2 million candidate videos within a ten month time frame, of which a few hundred can be aggregated as pairs (because $> 99\%$ of videos will not meet light transport constraints).

Dataset Statistics: As of the time of writing, 136.9K pairs have been automatically collected by the WeatherStream pipeline. The manual seed dataset consists of 38.2K pairs from [2]. See Fig. 4 for representative samples.

Where is Human Intervention Used?: With 100K+ pairs automated, we recommend that one human spends a few hours a month managing the dataset, e.g., to make sure the scraper is working, or removing erroneous pairs, especially since weather metadata has errors and webcam streams terminate from time-to-time.

6. Experimental Configuration

Baseline Dataset: The baseline dataset is simulated pairs of all-weather data [32, 57]. We use configurations provided by the authors.

Models Evaluated: We retrain several state of the art models [2, 57, 61, 71] on both the synthetic baseline dataset and our time multiplexed dataset.

Quantitative Evaluation: We follow the protocol of Ba *et al.* [2]. However, Ba *et al.* is only for rain; we replicate the procedure for multiple weather conditions by manually collecting more test scenes with a strict set of criterion similar to the GT-RAIN collection process. The result is a test set merged with GT-RAIN of size 13.5K covering 5.1K rain,

Method		Rain		Fog		Snow		Overall	
		PSNR \uparrow	SSIM \uparrow						
Syn.	TransWeather [57] (CVPR'22)	21.36	0.7434	17.49	0.7118	21.04	0.7736	19.98	0.7416
	Restormer [71] (CVPR'22)	21.29	0.7664	18.22	0.7532	20.84	0.7890	20.14	0.7685
	Uformer [61] (CVPR'22)	22.15	0.7723	18.20	0.7519	20.66	0.7920	20.40	0.7712
	Rain-robust [2] (ECCV'22)	17.96	0.7075	17.13	0.6931	17.52	0.7318	17.56	0.7097
Manual	TransWeather [57] (CVPR'22)	21.73	0.7622	21.59	0.7612	21.42	0.7932	21.59	0.7708
	Restormer [71] (CVPR'22)	22.63	0.7940	20.14	0.7700	21.62	0.8123	21.51	0.7913
	Uformer [61] (CVPR'22)	22.13	0.7759	18.65	0.7445	20.91	0.7821	20.62	0.7672
	Rain-robust [2] (ECCV'22)	22.83	0.7887	20.95	0.7691	22.17	0.8058	22.01	0.7871
Ours	TransWeather [57] (CVPR'22)	22.21	0.7716	22.55	0.7735	21.79	0.7919	22.20	0.7781
	Restormer [71] (CVPR'22)	23.67	0.8027	22.90	0.8029	22.51	0.8279	23.08	0.8100
	Uformer [61] (CVPR'22)	22.25	0.7911	18.81	0.7628	20.94	0.8009	20.72	0.7845
	Rain-robust [2] (ECCV'22)	23.43	0.7961	22.84	0.7901	22.29	0.8128	22.90	0.7989

Table 2. **Improvements in PSNR and SSIM across many models demonstrate the significance of the data in WeatherStream.** “Syn.” denotes the models trained on the synthetic pairs used by [57], “Manual” denotes the models trained on the initial manually collected dataset (similar to [2]), and “Ours” denotes the models trained on the final WeatherStream Dataset.

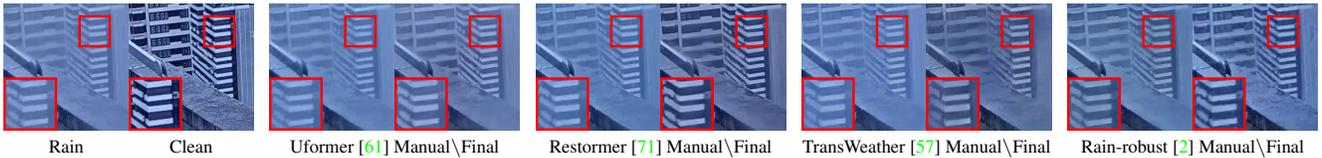


Figure 5. **Automation is not only more scalable, but also provides better color quality than manual collection.** Results on WeatherStream test set contrasted with manual curation from Ba *et al.* [2].

Metric	Input	No BG	No Illum./Blur	No Illum.	No Blur	Final
PSNR \uparrow	20.18	20.37	21.57	22.27	22.45	23.08
SSIM \uparrow	0.7699	0.7575	0.7863	0.7976	0.7863	0.8100

Table 3. **Improvements in PSNR and SSIM with the removal of each filtering block demonstrate the importance of each successive block.** “BG” refers to the first and second filtering blocks; “Blur” refers to the third; “Illum.” refers to the fourth.

3.9K snow and 4.5K fog weather conditions. PSNR and SSIM metrics are used as quantitative metrics of quality.

Qualitative Evaluation: Qualitative evaluation is performed at inference time on internet images.

Ablation Studies: We store intermediate outputs from the pipeline before certain filter blocks. The Restormer model [71] is retrained for results in Tab. 3.

7. Results

The results show that performance is impacted more from the introduction of large-scale time multiplexed pairs, rather than the specific model architecture.

Quantitative Results: Tab. 2 shows that our dataset improves almost all PSNR and SSIM metrics for all weather modalities, with the Restormer model notably improving by 1.49dB. This dataset provides improvements for both transformer and CNN architectures. As we can see from Fig. 5, the PSNR/SSIM increase is most likely due to over-smoothing in the manual model from lack of consideration

for the blurring effect from the multiple scattering model, as described in Sec. 3, which WeatherStream strongly favors. Additionally, we notice some color shifting in some of the outputs from the manual model, likely a result of an inability to generalize to certain scenes and backgrounds.

Qualitative Results: In addition to quantitative improvements, we see an increased ability to remove snowflakes and rain streaks (Fig. 6). This is also likely due to increased generalizability from more data, as well as considerations for the particle chromatic variation. Notice the removal of raindrops and snowflakes, as well as removal of the veiling effect when using WeatherStream. The inter-model variation appears less than inter-dataset variation going from synthetic to our proposed WeatherStream. This reinforces the importance of having the right data.

Ablation Studies: Tab. 3 shows the improvement in PSNR and SSIM with the introduction of each successive filter block, demonstrating the contribution of each light transport property towards the final result.

8. Discussion

In summary, this dataset is important because we observe larger inter-dataset change in performance, as compared to inter-model performance. Although simulated pairs are scalable and valuable, one can see visible weather effects in many of the models trained on simulated pairs in this paper.

The introduction of large-scale time-multiplexed pairs

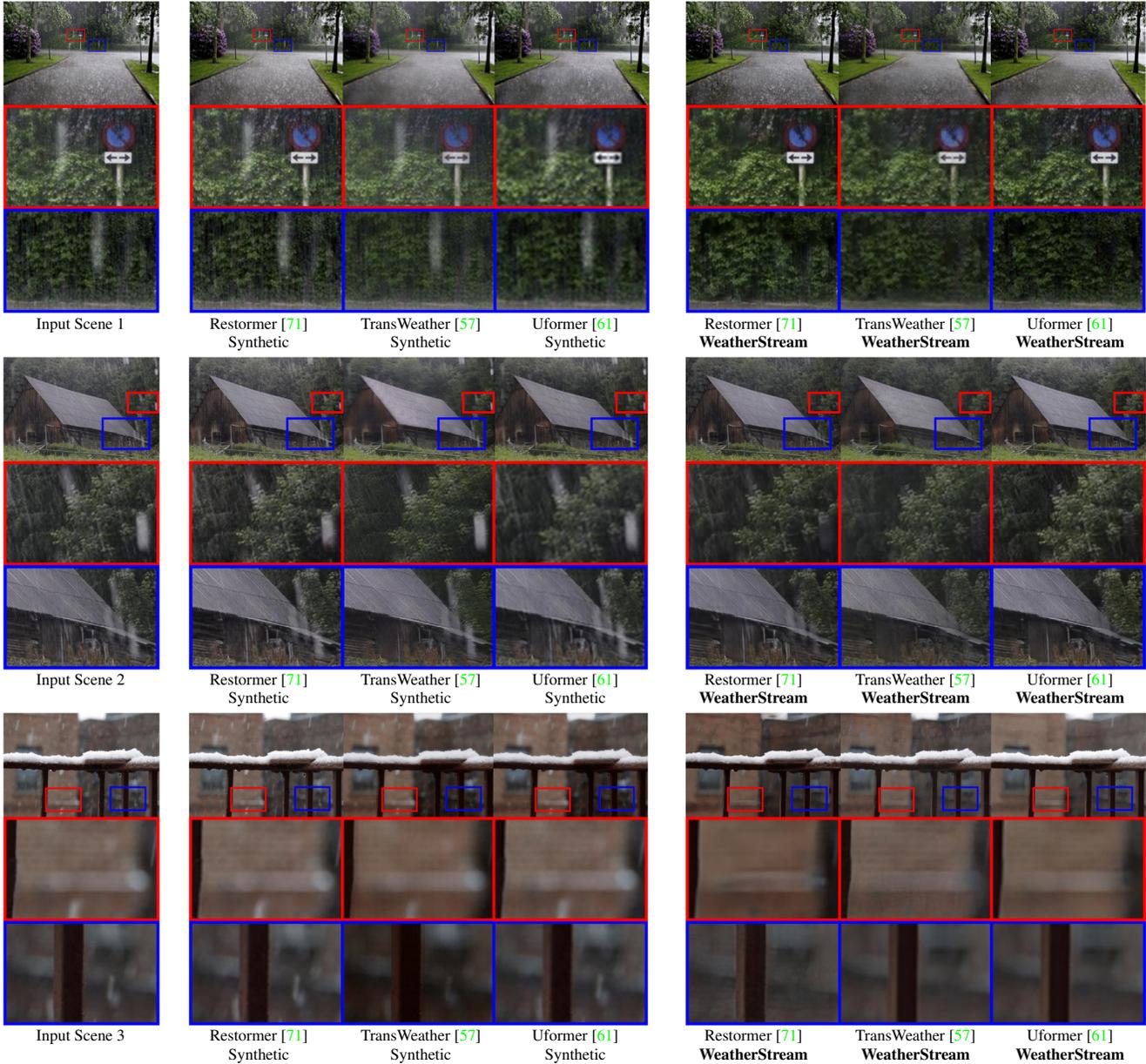


Figure 6. **Qualitative results showing generalization capability of synthetic data and WeatherStream** Trained on internet images. Models trained on WeatherStream are better at removing particles and veiling effects.

leads to larger changes in performance and visual quality for deweathering, as compared to changing the model architecture. WeatherStream employs light transport properties with an initial model trained on a small seed dataset. The end-result: all models trained on WeatherStream perform better. The improvements are not just about scalability; automating light transport leads to more consistent ground-truth pairs, laying a foundation for future challenges.

An area of future work is to combine our dataset with simulated datasets. We did not do so in this paper, as time multiplexed datasets are not widely used and need to be established on their own first. WeatherStream is also an on-

going effort, and we aim to later include paired segmentation labels to support recognition tasks in adverse conditions. Deweathering models trained on WeatherStream may also support the re-use of pretrained models for downstream tasks such as detection [39, 50], segmentation [5, 36, 52], depth prediction [1, 49, 65], and completion [23, 66].

Acknowledgements: We thank members of the Visual Machines Group (VMG) at UCLA for feedback and support. This research was partially supported by ARL W911NF20-2-0158 under the cooperative A2I2 program. A.K. was supported by an NSF CAREER Award IIS-2046737 and Army Young Investigator Program (YIP) Award.

References

- [1] Yunhao Ba, Alex Gilbert, Franklin Wang, Jinfa Yang, Rui Chen, Yiqin Wang, Lei Yan, Boxin Shi, and Achuta Kadambi. Deep shape from polarization. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXIV 16*, pages 554–571. Springer, 2020. 8
- [2] Yunhao Ba, Howard Zhang, Ethan Yang, Akira Suzuki, Arnold Pfahnl, Chethan Chinder Chandrappa, Celso de Melo, Suya You, Stefano Soatto, Alex Wong, and Achuta Kadambi. Not just streaks: Towards ground truth for single image deraining. In *European Conference on Computer Vision*. Springer, 2022. 2, 3, 4, 6, 7
- [3] Ayush Bhandari, Achuta Kadambi, and Ramesh Raskar. *Computational Imaging*. MIT Press, 2022. 4
- [4] Jia-Ren Chang and Yong-Sheng Chen. Pyramid stereo matching network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5410–5418, 2018. 2
- [5] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE transactions on pattern analysis and machine intelligence*, 40(4):834–848, 2017. 8
- [6] Liang-Chieh Chen, George Papandreou, Florian Schroff, and Hartwig Adam. Rethinking atrous convolution for semantic image segmentation. *arXiv preprint arXiv:1706.05587*, 2017. 2
- [7] Liang-Chieh Chen, Yukun Zhu, George Papandreou, Florian Schroff, and Hartwig Adam. Encoder-decoder with atrous separable convolution for semantic image segmentation. In *Proceedings of the European conference on computer vision (ECCV)*, pages 801–818, 2018. 2
- [8] 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 *European Conference on Computer Vision*, pages 754–770. Springer, 2020. 2, 3, 4
- [9] 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. 2, 3
- [10] 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. 2
- [11] Liang-Jian Deng, Ting-Zhu Huang, Xi-Le Zhao, and Tai-Xiang Jiang. A directional global sparse model for single image rain removal. *Applied Mathematical Modelling*, 59:662–679, 2018. 3
- [12] Shivam Duggal, Shenlong Wang, Wei-Chiu Ma, Rui Hu, and Raquel Urtasun. Deeppruner: Learning efficient stereo matching via differentiable patchmatch. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 4384–4393, 2019. 2
- [13] Deniz Engin, Anil Genç, and Hazim Kemal Ekenel. Cycledehaze: Enhanced cyclegan for single image dehazing. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pages 825–833, 2018. 2
- [14] Gunnar Farneback. Two-frame motion estimation based on polynomial expansion. In *Scandinavian conference on Image analysis*, pages 363–370. Springer, 2003. 5
- [15] Xiaohan Fei, Alex Wong, and Stefano Soatto. Geosupervised visual depth prediction. *IEEE Robotics and Automation Letters*, 4(2):1661–1668, 2019. 2
- [16] Xueyang Fu, Jiabin Huang, Xinghao Ding, Yinghao Liao, and John Paisley. Clearing the skies: A deep network architecture for single-image rain removal. *IEEE Transactions on Image Processing*, 26(6):2944–2956, 2017. 2
- [17] Xueyang Fu, Jiabin Huang, Delu Zeng, Yue Huang, Xinghao Ding, and John Paisley. Removing rain from single images via a deep detail network. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3855–3863, 2017. 2, 3
- [18] Kshitiz Garg and Shree K Nayar. Detection and removal of rain from videos. In *Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2004. CVPR 2004.*, volume 1, pages I–I. IEEE, 2004. 3, 5
- [19] Clément Godard, Oisín Mac Aodha, Michael Firman, and Gabriel J Brostow. Digging into self-supervised monocular depth estimation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019. 2
- [20] Qing Guo, Jingyang Sun, Felix Juefei-Xu, Lei Ma, Xiaofei Xie, Wei Feng, Yang Liu, and Jianjun Zhao. Efficienterain: Learning pixel-wise dilation filtering for high-efficiency single-image deraining. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 1487–1495, 2021. 2
- [21] Kaiming He, Jian Sun, and Xiaoou Tang. Single image haze removal using dark channel prior. *IEEE transactions on pattern analysis and machine intelligence*, 33(12):2341–2353, 2010. 4
- [22] Renjie He, Zhiyong Wang, Yangyu Fan, and David Dagan Feng. Multiple scattering model based single image dehazing. In *2013 IEEE 8th Conference on Industrial Electronics and Applications (ICIEA)*, pages 733–737. IEEE, 2013. 3, 4
- [23] Mu Hu, Shuling Wang, Bin Li, Shiyu Ning, Li Fan, and Xiaojin Gong. Penet: Towards precise and efficient image guided depth completion. *arXiv preprint arXiv:2103.00783*, 2021. 8
- [24] Xiaowei Hu, Chi-Wing Fu, Lei Zhu, and Pheng-Ann Heng. Depth-attentional features for single-image rain removal. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8022–8031, 2019. 2, 4
- [25] Xiaowei Hu, Lei Zhu, Tianyu Wang, Chi-Wing Fu, and Pheng-Ann Heng. Single-image real-time rain removal

- based on depth-guided non-local features. *IEEE Transactions on Image Processing*, 30:1759–1770, 2021. 2, 3
- [26] Kui Jiang, Zhongyuan Wang, Peng Yi, Chen Chen, Baojin Huang, Yimin Luo, Jiayi Ma, and Junjun Jiang. Multi-scale progressive fusion network for single image deraining. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8346–8355, 2020. 2
- [27] Glenn Jocher, Ayush Chaurasia, Alex Stoken, Jirka Borovec, NanoCode012, Yonghye Kwon, TaoXie, Jiacong Fang, imyhxy, Kalen Michael, Lorna, Abhiram V, Diego Montes, Je-bastin Nadar, Laughing, tkianai, yxNONG, Piotr Skalski, Zhiqiang Wang, Adam Hogan, Cristi Fati, Lorenzo Mammana, AlexWang1900, Deep Patel, Ding Yiwei, Felix You, Jan Hajek, Laurentiu Diaconu, and Mai Thanh Minh. ultralytics/yolov5: v6.1 - TensorRT, TensorFlow Edge TPU and OpenVINO Export and Inference, Feb. 2022. 2
- [28] Agastya Kalra, Guy Stoppi, Bradley Brown, Rishav Agarwal, and Achuta Kadambi. Towards rotation invariance in object detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3530–3540, 2021. 2
- [29] Kyros Kutulakos. Technical perspective: The dawn of computational light transport. *Communications of the ACM*, 59(9):78–78, 2016. 1, 2
- [30] Guanbin Li, Xiang He, Wei Zhang, Huiyou Chang, Le Dong, and Liang Lin. Non-locally enhanced encoder-decoder network for single image de-raining. In *Proceedings of the 26th ACM international conference on Multimedia*, pages 1056–1064, 2018. 3
- [31] Ruoteng Li, Loong-Fah Cheong, and Robby T Tan. Heavy rain image restoration: Integrating physics model and conditional adversarial learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1633–1642, 2019. 2, 3, 4
- [32] Ruoteng Li, Robby T Tan, and Loong-Fah Cheong. All in one bad weather removal using architectural search. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 3175–3185, 2020. 2, 3, 6
- [33] Yu Li, Robby T Tan, and Michael S Brown. Nighttime haze removal with glow and multiple light colors. In *Proceedings of the IEEE international conference on computer vision*, pages 226–234, 2015. 3, 4
- [34] Yu Li, Robby T. Tan, Xiaojie Guo, Jiangbo Lu, and Michael S. Brown. Rain streak removal using layer priors. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2736–2744, 2016. 3
- [35] Zhi Li, Juan Zhang, Zhijun Fang, Bo Huang, Xiaoyan Jiang, Yongbin Gao, and Jenq-Neng Hwang. Single image snow removal via composition generative adversarial networks. *IEEE Access*, 7:25016–25025, 2019. 2
- [36] Chenxi Liu, Liang-Chieh Chen, Florian Schroff, Hartwig Adam, Wei Hua, Alan L Yuille, and Li Fei-Fei. Auto-deeplab: Hierarchical neural architecture search for semantic image segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 82–92, 2019. 8
- [37] Renting Liu, Zhaorong Li, and Jiaya Jia. Image partial blur detection and classification. In *2008 IEEE conference on computer vision and pattern recognition*, pages 1–8. IEEE, 2008. 5
- [38] Tian Yu Liu, Parth Agrawal, Allison Chen, Byung-Woo Hong, and Alex Wong. Monitored distillation for positive congruent depth completion. *arXiv preprint arXiv:2203.16034*, 2022. 2
- [39] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C Berg. Ssd: Single shot multibox detector. In *European conference on computer vision*, pages 21–37. Springer, 2016. 8
- [40] Yun-Fu Liu, Da-Wei Jaw, Shih-Chia Huang, and Jenq-Neng Hwang. Desnet: Context-aware deep network for snow removal. *IEEE Transactions on Image Processing*, 27(6):3064–3073, 2018. 2, 3, 6
- [41] OpenWeather Ltd. OpenWeatherMap API. <https://openweathermap.org/>. Accessed: 2021-11-05. 6
- [42] Nathaniel Merrill, Patrick Geneva, and Guoquan Huang. Robust monocular visual-inertial depth completion for embedded systems. In *International Conference on Robotics and Automation (ICRA)*. IEEE, 2021. 2
- [43] Samy Metari and François Deschenes. A new convolution kernel for atmospheric point spread function applied to computer vision. In *2007 IEEE 11th international conference on computer vision*, pages 1–8. IEEE, 2007. 4
- [44] Srinivasa G Narasimhan and Shree K Nayar. Shedding light on the weather. In *2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2003. Proceedings.*, volume 1, pages I–I. IEEE, 2003. 3, 4
- [45] Shree K Nayar and Srinivasa G Narasimhan. Vision in bad weather. In *Proceedings of the seventh IEEE international conference on computer vision*, volume 2, pages 820–827. IEEE, 1999. 3, 5
- [46] Jinshan Pan, Sifei Liu, Deqing Sun, Jiawei Zhang, Yang Liu, Jimmy Ren, Zechao Li, Jinhui Tang, Huchuan Lu, Yu-Wing Tai, et al. Learning dual convolutional neural networks for low-level vision. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3070–3079, 2018. 2
- [47] Jinsun Park, Kyungdon Joo, Zhe Hu, Chi-Kuei Liu, and In-So Kweon. Non-local spatial propagation network for depth completion. In *European Conference on Computer Vision, ECCV 2020*. European Conference on Computer Vision, 2020. 2
- [48] Rui Qian, Robby T Tan, Wenhan Yang, Jiajun Su, and Jiaying Liu. Attentive generative adversarial network for rain-drop removal from a single image. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2482–2491, 2018. 2
- [49] René Ranftl, Alexey Bochkovskiy, and Vladlen Koltun. Vision transformers for dense prediction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 12179–12188, 2021. 8
- [50] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 779–788, 2016. 8

- [51] Dongwei Ren, Wei Shang, Pengfei Zhu, Qinghua Hu, Deyu Meng, and Wangmeng Zuo. Single image deraining using bilateral recurrent network. *IEEE Transactions on Image Processing*, 29:6852–6863, 2020. [2](#)
- [52] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, October 5-9, 2015, Proceedings, Part III 18*, pages 234–241. Springer, 2015. [8](#)
- [53] Yehu Shen and Qicong Wang. Sky region detection in a single image for autonomous ground robot navigation. *International Journal of Advanced Robotic Systems*, 10(10):362, 2013. [5](#)
- [54] Zheng Shi, Ethan Tseng, Mario Bijelic, Werner Ritter, and Felix Heide. Zeroscatter: Domain transfer for long distance imaging and vision through scattering media. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3476–3486, 2021. [2](#)
- [55] Robby T Tan. Visibility in bad weather from a single image. In *2008 IEEE conference on computer vision and pattern recognition*, pages 1–8. IEEE, 2008. [4](#), [5](#)
- [56] Jiandong Tian, Zhi Han, Weihong Ren, Xia Chen, and Yandong Tang. Snowflake removal for videos via global and local low-rank decomposition. *IEEE Transactions on Multimedia*, 20(10):2659–2669, 2018. [3](#)
- [57] 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. [1](#), [2](#), [3](#), [6](#), [7](#), [8](#)
- [58] Hong Wang, Qi Xie, Qian Zhao, and Deyu Meng. A model-driven deep neural network for single image rain removal. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, June 2020. [2](#), [3](#)
- [59] Rui Wang, Rui Li, and Haiyan Sun. Haze removal based on multiple scattering model with superpixel algorithm. *Signal Processing*, 127:24–36, 2016. [4](#), [5](#)
- [60] Tianyu Wang, Xin Yang, Ke Xu, Shaozhe Chen, Qiang Zhang, and Rynson WH Lau. Spatial attentive single-image deraining with a high quality real rain dataset. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12270–12279, 2019. [1](#), [2](#), [3](#)
- [61] 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. [6](#), [7](#), [8](#)
- [62] Jamie Watson, Michael Firman, Gabriel J Brostow, and Daniyar Turmukhambetov. Self-supervised monocular depth hints. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2162–2171, 2019. [2](#)
- [63] Yanyan Wei, Zhao Zhang, Yang Wang, Mingliang Xu, Yi Yang, Shuicheng Yan, and Meng Wang. Deraincyclegan: Rain attentive cyclegan for single image deraining and rain-making. *IEEE Transactions on Image Processing*, 30:4788–4801, 2021. [2](#)
- [64] Alex Wong, Xiaohan Fei, Stephanie Tsuei, and Stefano Soatto. Unsupervised depth completion from visual inertial odometry. *IEEE Robotics and Automation Letters*, 2020. [2](#)
- [65] Alex Wong and Stefano Soatto. Bilateral cyclic constraint and adaptive regularization for unsupervised monocular depth prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5644–5653, 2019. [8](#)
- [66] Alex Wong and Stefano Soatto. Unsupervised depth completion with calibrated backprojection layers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 12747–12756, 2021. [8](#)
- [67] Haofei Xu and Juyong Zhang. Aanet: Adaptive aggregation network for efficient stereo matching. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1959–1968, 2020. [2](#)
- [68] Tao Yan, Yuyang Ding, Fan Zhang, Ningyu Xie, Wenxi Liu, Zhengtian Wu, and Yuan Liu. Snow removal from light field images. *IEEE Access*, 7:164203–164215, 2019. [2](#)
- [69] Wenhao 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/CVF Conference on Computer Vision and Pattern Recognition*, pages 1357–1366, 2017. [2](#)
- [70] Rajeev Yasarla and Vishal M. Patel. Uncertainty guided multi-scale residual learning-using a cycle spinning cnn for single image de-raining. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8405–8414, 2019. [3](#)
- [71] Syed Waqas Zamir, Aditya Arora, Salman Khan, Munawar Hayat, Fahad Shahbaz Khan, and Ming-Hsuan Yang. Restormer: Efficient transformer for high-resolution image restoration. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5728–5739, 2022. [6](#), [7](#), [8](#)
- [72] Syed W. Zamir, Aditya Arora, Salman Khan, Munawar Hayat, Fahad S. 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. [2](#)
- [73] He Zhang and Vishal M Patel. Density-aware single image de-raining using a multi-stream dense network. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 695–704, 2018. [2](#), [3](#)
- [74] Xiaopeng Zhang, Hao Li, Yingyi Qi, Wee Kheng Leow, and Teck Khim Ng. Rain removal in video by combining temporal and chromatic properties. In *2006 IEEE international conference on multimedia and expo*, pages 461–464. IEEE, 2006. [3](#), [5](#)
- [75] Lei Zhu, Chi-Wing Fu, Dani Lischinski, and Pheng-Ann Heng. Joint bi-layer optimization for single-image rain streak removal. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2526–2534, 2017. [3](#)