IntrinsicWeather: Controllable Weather Editing in Intrinsic Space

Yixin Zhu[1] Zuo-Liang Zhu[2] Jian Yang[1] Miloš Hašan[3] Jin Xie[1*] Beibei Wang[1*]
[1]Nanjing University [2]Nankai University [3]NVIDIA

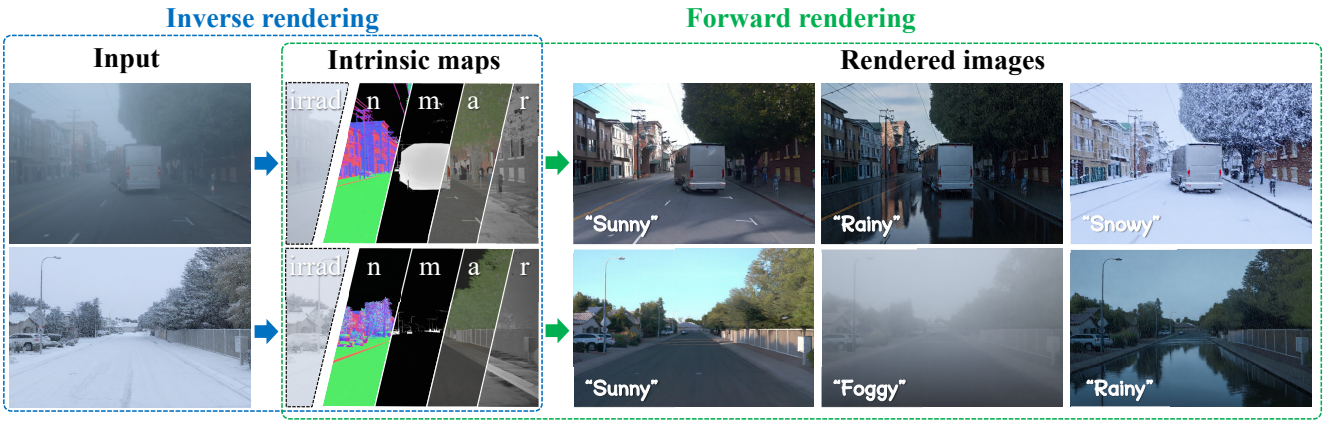

Figure 1. We introduce IntrinsicWeather, a framework for controllable weather editing in intrinsic space. Our framework includes two components, an inverse renderer and a forward renderer. The inverse renderer decomposes an input image into intrinsic maps, including weather-invariant material maps (albedo, roughness, metallicity), a normal map, and an irradiance map that captures illumination and weather effects. The forward renderer then combines these maps with a prompt specifying the target weather to synthesize a new image. By disentangling materials, geometry, and illumination, IntrinsicWeather enables realistic and controllable weather manipulation.

Abstract

We present IntrinsicWeather, a diffusion-based framework for controllable weather editing in intrinsic space. Our framework includes two components based on diffusion priors: an inverse renderer that estimates material properties, scene geometry, and lighting as intrinsic maps from an input image, and a forward renderer that utilizes these geometry and material maps along with a text prompt that describes specific weather conditions to generate a final image. The intrinsic maps enhance controllability compared to traditional pixel-space editing approaches. We propose an intrinsic map-aware attention mechanism that improves spatial correspondence and decomposition quality in large outdoor scenes. For forward rendering, we leverage CLIP-space interpolation of weather prompts to achieve fine-grained weather control. We also introduce a synthetic and a real-world dataset, containing 38k and 18k images under various weather conditions, each with intrinsic map annotations. IntrinsicWeather outperforms state-of-the-art pixel-space editing approaches, weather restoration methods, and rendering-based methods, showing promise for downstream tasks such as autonomous driving, enhancing the robustness of detection and segmentation in challenging weather scenarios.

1. Introduction

For autonomous vehicles, robust scene understanding requires the ability to handle adverse weather conditions. Techniques that handle weather effects, can substantially improve the robustness of perception models across different weather conditions. While diffusion-based image editing techniques [2, 11, 16, 44] have created significant opportunities here, a key limitation remains: the lack of fine-grained controllability in the generated scenarios.

Following existing image editing approaches, weather editing methods [6, 17, 41] typically leverage a unified generative model to perform weather transformations in the pixel space. Particularly, WeatherWeaver [23] decomposes weather editing into a two-stage strategy: weather removal

* Corresponding author.
[1] School of Intelligence Science and Technology, Nanjing University, Suzhou, China
[2] College of Computer Science, Nankai University, Tianjin, China

and weather synthesis. While these methods have achieved impressive results in weather manipulation, they still struggle to preserve the underlying material and geometry of the scene while generating natural illumination.

In this paper, we present *IntrinsicWeather*, a novel framework for controllable weather editing that operates in intrinsic space. Our key insight is that by decomposing a scene into its fundamental components—material properties, geometry, and lighting, we can achieve higher control over weather effects than in the image space. IntrinsicWeather consists of two key components: a forward renderer that uses CLIP-space interpolation and diffusion priors for fine-grained weather synthesis, and a novel inverse renderer tailored for outdoor and autonomous driving scenes. We are inspired by recent diffusion-based intrinsic decomposition and recomposition approaches, RGB↔X [46] and DiffusionRenderer [22], which have achieved impressive results at the indoor and object levels; however, they do not focus on weather and do not generalize to large-scale autonomous driving scenarios. We bridge this gap with a novel intrinsic map-aware attention mechanism that ensures spatial correspondence, thereby significantly enhancing decomposition fidelity in complex, unconstrained outdoor environments. By combining the inverse and forward renderers, our framework allows for controlled editing of weather and lighting.

Our IntrinsicWeather outperforms existing state-of-the-art inverse and forward rendering methods, achieving over 10 dB PSNR improvement in inverse rendering and higher PickScore [14] in forward rendering. Compared with weather restoration and pixel-space editing methods, our approach achieves higher text-image consistency and better DINO-based structural alignment, enabling cleaner and more controllable weather editing. By proactively correcting environmental distortions at the visual input level, we significantly boost the performance of downstream tasks such as object detection and semantic segmentation. We observe that after applying IntrinsicWeather, the detection and segmentation performance on the ACDC [37] benchmark increases by 87.1% (AP_{75} from 13.15% to 24.60%) and 24.5% (mIOU from 24.13% to 30.05%), respectively. We also introduce a synthetic dataset and a real-world dataset containing 38k and 18k images with intrinsic map annotations, covering diverse weather conditions and a wide range of driving scenes to train these two components. To summarize, our contributions are as follows:

- We propose IntrinsicWeather, a method to decompose images into intrinsic maps under various weather conditions, and synthesize them into another lighting or weather condition guided by text prompts.
- We introduce intrinsic map-aware attention that provides customized visual detail guidance for generative models, helping our decomposition.
- We construct two new datasets called WeatherSynthetic and WeatherReal, containing synthetic and real-world images covering various weather conditions on autonomous driving scenarios, along with their corresponding maps. The datasets will be released upon acceptance.

2. Related work

Diffusion models. Diffusion models have achieved remarkable progress in high-fidelity and text-conditioned image generation [7, 12, 33, 35]. Modern diffusion frameworks typically adopt a denoising process [12, 24, 25, 38], parameterized by UNet- or DiT-based backbones [30, 36]. In our research, we repurpose diffusion models to jointly estimate material, geometry, and lighting from an image while synthesizing new images under specified weather conditions. This demonstrates that the strong priors embedded in pretrained diffusion models can be effectively leveraged for faithful and physically grounded estimations.

Forward and inverse rendering using diffusion. The emergence of the diffusion model has catalyzed a novel methodology that leverages generative models to learn the joint probability distribution between images and their corresponding intrinsic maps [5, 8, 10, 15, 21, 22, 26, 46]. IID [15] focuses on material estimation in indoor scenes, while RGB↔X [46] extends diffusion to bidirectional mapping between RGB images and maps. Several works further incorporate geometric priors [8] or multi-view cues [21]. DiffusionRenderer [22] adapts a video diffusion model to achieve temporally consistent inverse and forward rendering, and UniRelight [10] jointly estimates albedo and relighted video frames. These methods are primarily designed for indoor scenes, small objects, or video relighting, and they struggle to generalize to large-scale outdoor driving scenes with diverse weather conditions. Moreover, none of these works address controllable weather editing or decomposition in intrinsic space under multiple weather conditions, which is the goal of our framework.

Image and weather editing. Most image editing techniques operate purely in the pixel space, using diffusion models to modify color, texture, or local appearance without modeling underlying scene factors [2, 3, 11, 13, 16, 29, 42, 44]. Following this paradigm, existing weather-editing models [6, 17, 41] directly translate one weather type to another, and WeatherWeaver [23] finetunes a video diffusion model for weather removal and synthesis. However, pixel-space editing lacks physical interpretability and cannot guarantee consistent material, geometry, or illumination. Intrinsic-space manipulation has been explored by IntrinsicEdit [27], but it focuses on object-level editing rather than large-scale outdoor scenes. Other approaches edit weather in 3D space [19, 31], but they require accurate geometry, which is unavailable for real-world driving scenes. In contrast, IntrinsicWeather performs controllable weather

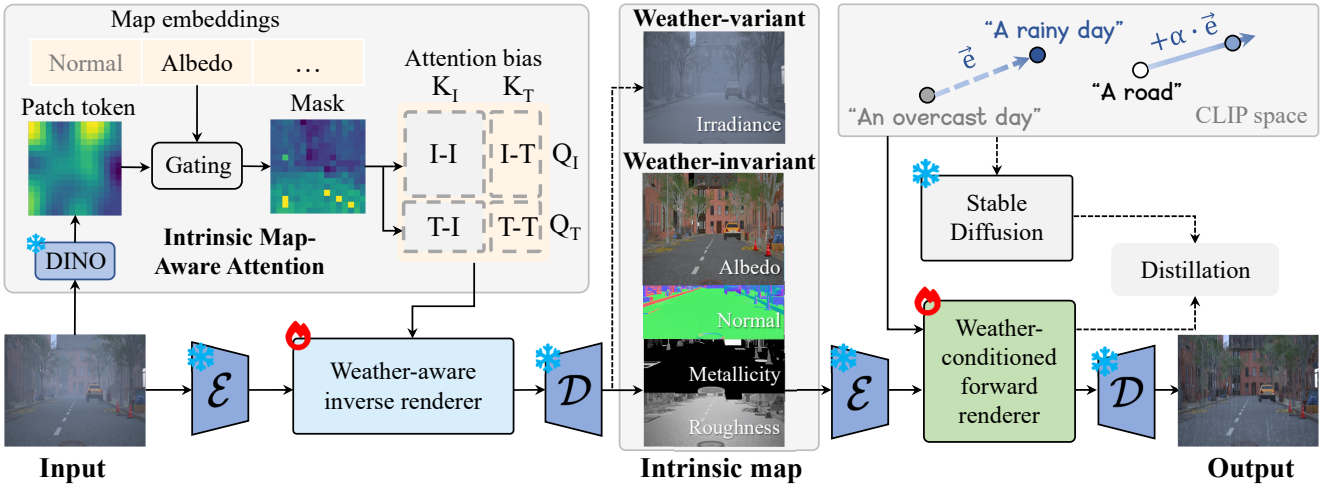

Figure 2. Overview of IntrinsicWeather. We propose a diffusion-based framework for controllable weather editing for autonomous driving in intrinsic space. The weather-aware inverse renderer decomposes images into weather-invariant and weather-variant maps, while the weather-conditioned forward renderer re-renders images based on given decomposed maps and text prompts that specify the target condition. For the inverse renderer, we design intrinsic map-aware attention to help the inverse renderer focus on important regions corresponding to target intrinsic maps, where the learned map embeddings filter patch tokens via a gating mechanism. For the forward renderer, we design an alpha interpolation in the CLIP semantic space to achieve fine-grained weather control, leveraging the prior in the original Stable Diffusion. By sampling different alpha values, the forward renderer can render natural transitional weather conditions.

editing in the intrinsic space. This formulation enables fine-grained and geometry-preserving control that is difficult to achieve in the pixel space.

3. Method

3.1. IntrinsicWeather

We aim to modify weather-related factors (*e.g.*, weather particles and accumulations) of a scene while preserving its underlying geometry and material properties. Pixel-space editing methods inherently entangle weather effects with appearance, making it difficult to maintain structural consistency and natural illumination across different weather conditions. Moreover, weather phenomena are primarily tied to a scene's lighting, rather than intrinsic material attributes. These observations motivate us to move beyond direct pixel manipulation and instead operate in intrinsic space.

We propose IntrinsicWeather, a framework designed for controllable weather editing in intrinsic space. The framework includes two components: the weather-aware inverse renderer and the weather-conditioned forward renderer. The image is input into the inverse renderer, which performs intrinsic decomposition, disentangling images into weather-invariant material and geometry, as well as weather-variant illumination. Correspondingly, the forward renderer re-renders images based on given intrinsic maps and text prompts that specify the target weather or lighting.

We repurpose Stable Diffusion 3.5 [1] to enable the inverse and forward renderers. To leverage the diffusion prior to achieve fine-grained weather control, we first obtain a weather transitional direction $\boldsymbol{e}$ in the CLIP space:

$$\boldsymbol{e} = \mathrm{Embed}(\boldsymbol{w}_1) - \mathrm{Embed}(\boldsymbol{w}_2), \tag{1}$$

where $\boldsymbol{w}_1$ denotes the target weather (*e.g.*, rainy), $\boldsymbol{w}_2$ denotes the original weather (*e.g.*, overcast), and $\mathrm{Embed}(\cdot)$ is the text encoder. Then we shift a weather-neutral embedding $\boldsymbol{w}_{base}$ by α steps along this direction:

$$\boldsymbol{E} = \mathrm{Embed}(\boldsymbol{w}_{base}) + \alpha \cdot \boldsymbol{e}. \tag{2}$$

Replacing the original prompt with $\boldsymbol{E}$ can force the model to generate reasonable intermediate results. To preserve the rich priors of the pre-trained model, we align part of the forward renderer's intermediate features with those of the original Stable Diffusion. The detailed implementation of feature distillation is shown in the supplementary material.

3.2. Intrinsic map-aware attention

Outdoor and autonomous driving scenes exhibit larger variations in object scale. As shown in Fig. 3, the original Stable Diffusion lacks explicit attention guidance and often performs poorly on distant, small objects and geometrically complex regions. We observe that different intrinsic maps require attention to distinct regions of an image, as shown in Fig. 4. To leverage these observations, we extend DiT with intrinsic map-aware attention (IMAA) to apply attention guidance for DiT.

We identify important regions based on the intrinsic maps and use them to construct an attention bias. Specifi-

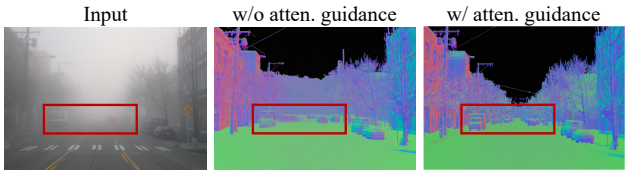

Figure 3. Attention guidance helps recover distant small objects and fine geometry details. We present the estimated normal map with and without attention guidance.

cally, we first employ DINOv2 [28] to extract a set of patch tokens $\boldsymbol{p}$. For each intrinsic map, we define a learnable embedding $\boldsymbol{d} \in \mathbb{R}^{D_{\text{model}}}$ that captures its inherent characteristics. A gating mechanism is applied to selectively filter patch tokens based on the current intrinsic map. Formally, we compute a map-aware mask:

$$\boldsymbol{m} = \text{gating}(\boldsymbol{p}, \boldsymbol{d}) = \text{MLP}\big([f_p(\boldsymbol{p}), f_{\boldsymbol{d}}(\boldsymbol{d})]\big), \quad (3)$$

where $f_p(\cdot)$ and $f_d(\cdot)$ are linear projections of patch tokens $\boldsymbol{p}$ and map embedding $\boldsymbol{d}$, $[\cdot]$ indicates the concatenated input to the MLP. This gating mechanism highlights image regions most relevant to the target map.

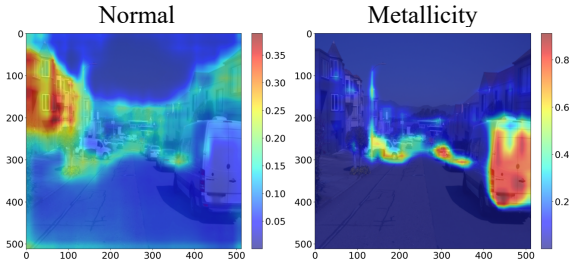

Figure 4. IMAA visualization. Normal estimation primarily concerns the geometry details, especially in regions with sharp variations in surface normals. Metallicity predictions need to selectively attend to metallic objects such as vehicles, poles, and railings. IMAA provides attention guidance for the diffusion model, ensuring spatial correspondence between input and maps.

We construct a joint attention bias $\boldsymbol{M}$ using $\boldsymbol{m}$ to guide the diffusion model, effectively enhancing the attention logits in DiT. We apply the map-aware mask to the text–image and image–image parts of the joint attention matrix: the former enforces textual guidance on important regions, while the latter strengthens feature aggregation within the image space. Formally, the bias $\boldsymbol{M}$ is defined as:

$$\boldsymbol{M}_{i,j} = \begin{cases} \boldsymbol{m}_i, & \text{if } i \text{ indexes an image token in } \boldsymbol{K}_I, \\ 0, & \text{otherwise.} \end{cases} \quad (4)$$

where i and j denote token indices, and $\boldsymbol{K}_I$ is the set of image tokens.

Then the bias is applied to the DiT:

$$\text{Attn}(\boldsymbol{Q}, \boldsymbol{K}, \boldsymbol{V}) = \text{Softmax}\big(\frac{\boldsymbol{Q}\boldsymbol{K}^T}{\sqrt{d_k}} + \boldsymbol{M}\big)\boldsymbol{V}, \quad (5)$$

where $\boldsymbol{Q}, \boldsymbol{K}, \boldsymbol{V}$ are combination of image and text token.

Moreover, we devise a heuristic-guided progressive training strategy for IMAA to stabilize learning and ensure that IMAA provides meaningful guidance in the early stages. For example, we use the gradient operator to extract illuminated regions and shadow boundaries for the irradiance map. The detailed description and ablation studies are shown in the supplementary material.

3.3. Dataset construction

Existing datasets of images with corresponding intrinsic maps suffer from the absence of outdoor environments (OpenRooms [20], Hypersim [34], InteriorVerse [47]) or insufficient weather diversity (MatrixCity [18]), and thus are inappropriate for large-scale outdoor and autonomous driving scenes. While MatrixCity provides continuous variations of fog density and illumination, it mainly targets relighting and reconstruction tasks. In contrast, our work requires diverse weather and lighting conditions rather than smooth transitions of the same type. Moreover, MatrixCity suffers from legal licensing issues. To fill the gap in large-scale weather-diverse autonomous driving datasets with paired images and intrinsic maps, we propose WeatherSynthetic and WeatherReal. Sample images from our datasets are shown in Fig. 5.

WeatherSynthetic is a large-scale synthetic dataset encompassing a wide range of scene and weather types:

- **Weather:** sunny, overcast, rainy, thunderstorm, snowy, foggy, sandstorm.
- **Time of day:** early morning, morning, noon, afternoon.
- **Environment:** urban, suburban, highway, parking

We use Unreal Engine 5 to render all images and intrinsic maps. We purchased 3D assets that are cleared for generative model use in Fab. The rendering pipeline uses the movie render queue and multi-sample anti-aliasing, producing high-quality rendering results. The UltraDynamicSky and UltraDynamicWeather are applied to modify the weather and daytime. Note that all images are in linear space without tone mapping. In total, rendering the 38K images and maps took about 24 hours on our setup.

WeatherReal is a real-world dataset on autonomous driving scenes with various weather conditions. We use our inverse renderer to generate intrinsic maps of open-source datasets such as Waymo [39] and Kitti [9]. We use a multimodal model to remove challenging scenarios (e.g., rainy nights) to ensure the generation of high-quality pseudo-labels, and check the quality manually. The samples are shown in the supplementary materials. Moreover,

Figure 5. Example of our WeatherSynthetic (the first row) and WeatherReal (the second row).

we employ a pre-trained image editing model (*i.e.*, Flux-Kontext [16]) to alter the weather types. Our WeatherReal is motivated by the observation that after training our model merely on synthetic data, the inverse and forward renderers lack sufficient generalization capability on real-world samples. Note that WeatherReal is only used to finetune models and is not used to evaluate results.

4. Experimental results

In this section, we first compare our method with pixel-space editing and weather restoration methods. Then we evaluate the inverse and forward rendering. We then conduct ablation studies on IMAA and datasets, and conclude with a discussion of the limitations of our approach.

Following WeatherWeaver [23], we use PickScore [14], CLIP image-text consistency (denoted as CLIP-S), and DINO structure similarity (denoted as DINO-S) to evaluate editing results. Following previous works [21, 46], we report Peak Signal-to-Noise Ratio (PSNR), Structural Similarity Index Measure (SSIM), Mean Angular Error (MAE), and Learned Perceptual Image Patch Similarity (LPIPS) for inverse rendering. We compare our performance with pixel-space editing methods (Flux-Kontext [16], Qwen-Image-Edit [44], Instruct-Pix2Pix [2], WeatherWeaver [23]) and weather restoration methods (AWRaCL [32], Histoformer [40]). We also compare our inverse and forward rendering results with RGB↔X [46], IID [15], Geowizard [8], IDArb [21] and DiffusionRenderer [22]. We evaluate different methods on WeatherSynthetic, Waymo [39], TransWeather [43], ACDC [37], and additional Internet images covering diverse weather conditions.

4.1. Comparison with pixel-space editing methods

We show quantitative comparisons in Tab. 1. Our method achieves the highest CLIP-S, indicating that it produces the most text-aligned and plausible weather editing results. In terms of DINO-S, we rank second only to Flux-Kontext [16], which, however, fails to remove or synthesize weather effects effectively. Although Qwen-Image-Edit [44] achieves a slightly higher PickScore, it often introduces inconsistent textures and geometry. PickScore is suitable for measuring user preference, but it does not measure physical consistency and editing plausibility.

A qualitative comparison is shown in Fig. 6. In the first row, our method removes all the snowflakes and snow accumulation on the trees. Flux-Kontext fails to remove them, while Qwen-Image-Edit and WeatherWeaver mistakenly change the geometry of the scene and the car's color. For the second row, our method removes the dense mist, recovering the color and pose of pedestrians. Flux-Kontext generates noisy textures while Qwen-Image-Edit and WeatherWeaver change the count and pose of pedestrians. In the last row, we transform the weather into a rainy day, generating natural reflection and lighting. The other methods produce unnatural lighting while struggling to preserve geometry and material. Flux-Kontext adds some rain streaks on the original image, making the sunny-day shadows on the ground look noticeably out of place. Instruct-Pix2Pix struggles to manipulate weather effects and instead performs incorrect operations resembling style transfer.

We also present the albedo and normal map obtained from our inverse renderer. Our re-rendered images align well with these maps. Weather editing in intrinsic space allows our model to completely remove weather-related artifacts, including both airborne particles and surface accumulations, while preserving geometric and material consistency. Furthermore, the disentangled material and geometry obtained from inverse rendering facilitate realistic illumination and shadow generation during re-rendering. More comparisons are shown in the supplementary.

We further compare our method with WeatherWeaver on fine-grained weather control in Fig. 7. Our editing results show natural transitions: under light to heavy rain, the road surface gradually becomes wetter; under different levels of snowfall, snow first appears along the roadside and on branches, and eventually accumulates to cover the entire road. In contrast, WeatherWeaver lacks this sense of realism and instead looks more like applying filters of increasing intensity to the original image.

User study. We conducted a user study to evaluate the consistency and realism of different editing methods. A total of 61 participants were asked to vote on 8 cases covering weather removal and weather synthesis. The average preference for our results is 81.67%, showing that our method is preferred by users. The detailed setup and results are provided in the supplementary material.

4.2. Comparison with weather restoration methods

We present qualitative comparisons with AWRaCLe [32] and Histoformer [40] in Fig. 8. The advantage of our method is its ability to effectively remove various degradations in adverse weather images, such as airborne particles (like snowflakes), snow on the ground, and overall illumination. The weather restoration methods only remove par-

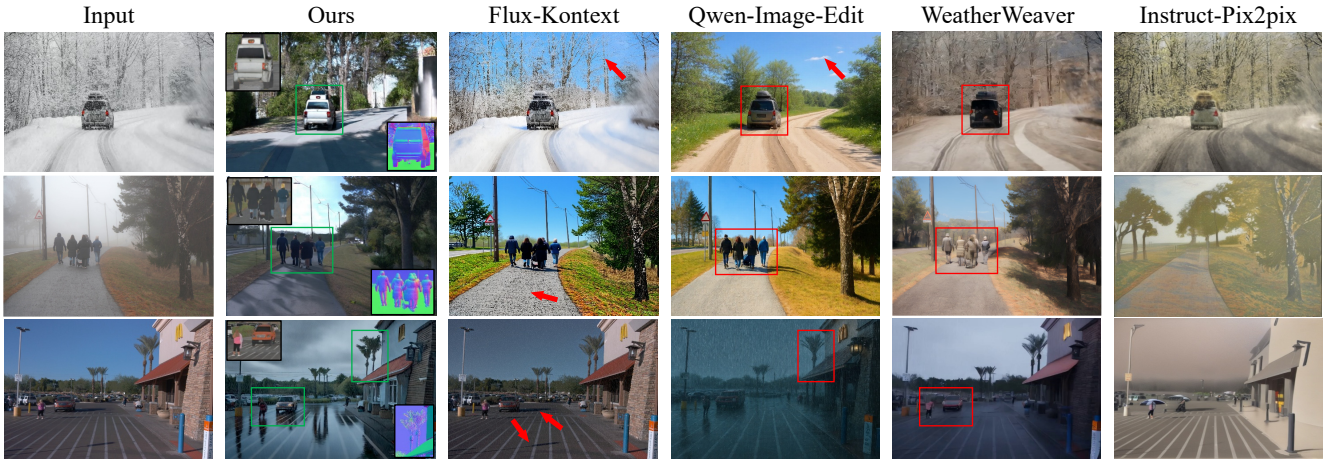

Figure 6. Comparison with pixel-space editing methods. We use the prompt "A sunny/rainy day" for our model. For pixel-space editing methods, we use their recommended instruction format, *i.e.*, "Turn weather into a sunny/rainy day". Pixel-space methods struggle to preserve scene geometry and materials, often introducing hallucinated objects, distorted structures, or unnatural lighting. In contrast, our intrinsic-space editing preserves geometry and appearance while modifying only weather-related components. For clarity, we highlight corresponding artifacts with red boxes and arrows. We also show intrinsic maps (albedo + normal) for our method, demonstrating explicit disentanglement of material, geometry, and illumination.

Table 1. Comparison with rendering-based methods and pixel-space editing methods.

Method	PickScore ↑ [14]				CLIP-S ↑	DINO-S ↑
	Sunny	Snowy	Foggy	Rainy		
Rendering-based methods						
Ours	20.59	22.32	21.34	20.76	27.66	73.63
RGB↔X	20.40	19.92	19.71	20.29	19.00	55.87
DiffusionRenderer	20.24	–	–	–	–	43.12
Pixel-space editing methods						
Flux-Kontext	20.72	22.25	20.99	19.46	24.46	85.50
Qwen-Image-Edit	20.77	22.43	21.82	21.56	27.14	53.70
WeatherWeaver	20.13	21.41	20.93	20.25	26.78	67.01
Instruct-Pix2Pix	20.27	21.39	20.98	19.83	23.79	22.41

ticles while failing to change surface material or lighting conditions.

To further verify the improvement brought by different models to downstream applications, we choose object detection and semantic segmentation as tasks, evaluating performance improvement before and after weather editing. Specifically, we chose DETR [4] and Segformer [45] as base models. When applying object detection and semantic segmentation models to our re-rendered images, both tasks achieve more accurate and consistent results, as shown in the lower rows of Fig. 8. Following this way, we apply weather editing and evaluate on the validation set of the ACDC benchmark [37], and the results are shown in Tab. 2.

Figure 7. Comparison of fine-grained weather control. Our editing results show natural transitions. WeatherWeaver shows lower realism and looks closer to a blending effect.

Table 2. Object detection and semantic segmentation results on the ACDC validation set before and after applying IntrinsicWeather.

	$AP_{0.5}$	$AP_{0.75}$	$mAP_{[0.5:0.95]}$	mIOU
DETR	56.56	13.15	47.00	–
DETR + ours	61.32	24.60	54.87	–
Segformer	–	–	–	24.13
Segformer + ours	–	–	–	30.05
Absolute Gain	+4.76	+11.45	+7.87	+5.92

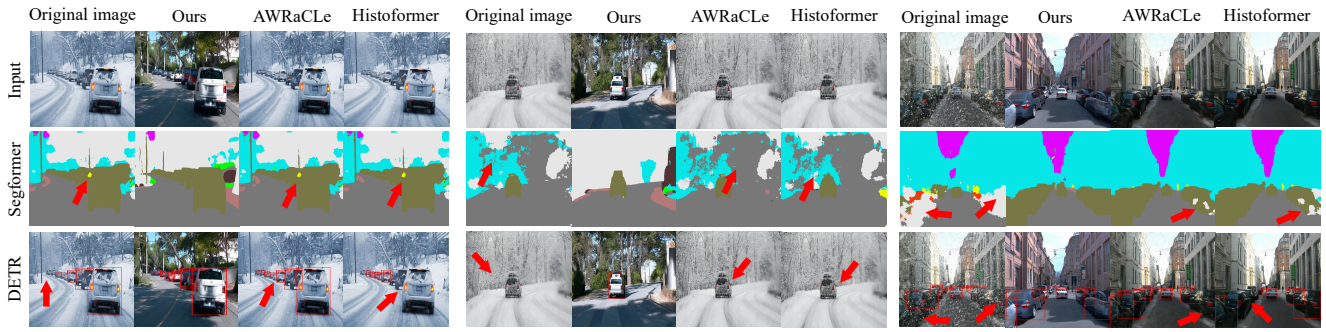

Figure 8. Validation of the enhancement of detection and segmentation. Our re-rendered images (prompt: "The image depicts a bright sunny day.") not only remove airborne particles (*e.g.*, snowflakes) but also restore material and lighting conditions (*e.g.*, removing surface snow), leading to more accurate segmentation and detection results. In contrast, AWRaCLe and Histoformer primarily remove particles in the air but fail to correct material or lighting degradations. Red arrows note the wrong estimations.

Table 3. Quantitative evaluations of our method against existing methods in terms of decomposition quality on the WeatherSynthetic test set. Considering that IID and RGB↔X were only trained on indoor datasets, we finetune them on our WeatherSynthetic and show the results before and after finetuning. We highlight the best results in red and the second-best ones in orange.

Method	Albedo			Normal			Roughness		Metallicity		Irradiance	
	PSNR ↑	SSIM ↑	LPIPS ↓	PSNR ↑	SSIM ↑	MAE ↓	PSNR ↑	LPIPS ↓	PSNR ↑	LPIPS ↓	PSNR ↑	LPIPS ↓
IID	7.80	0.26	0.63	–	–	–	10.30	0.55	12.37	0.64	–	–
IID (w/ finetune)	11.55	0.53	0.40	–	–	–	12.34	0.43	12.22	0.55	–	–
RGB↔X	9.66	0.44	0.47	11.90	0.41	15.51	13.62	0.55	–	–	16.24	0.58
RGB↔X (w/ finetune)	11.35	0.59	0.37	16.14	0.49	7.05	13.65	0.57	11.96	0.66	16.38	0.69
GeoWizard	–	–	–	16.24	0.54	12.47	–	–	–	–	–	–
IDArb	6.40	0.48	0.65	10.77	0.43	22.42	10.70	0.62	14.66	0.62	–	–
DiffusionRenderer	11.91	0.64	0.34	16.43	0.70	28.68	11.31	0.42	10.05	0.43	–	–
Ours	27.99	0.86	0.35	25.06	0.84	4.24	25.81	0.23	29.29	0.04	29.66	0.22
Ours (w/o IMAA)	26.78	0.84	0.43	23.63	0.79	6.33	24.60	0.25	28.16	0.05	26.99	0.32

4.3. Evaluation for components

4.3.1. Inverse rendering

In this part, we first conduct quantitative evaluations on WeatherSynthetic, then we evaluate on real-world TransWeather [43] datasets. We show the comparison between our method and existing methods on the test set of our WeatherSynthetic in Tab. 3. Our method outperforms existing approaches across all evaluation metrics. We fine-tune IID [15] and RGB↔X [46] with the same training steps on our WeatherSynthetic. Their performance improves, but they fail to provide high-quality estimation. An overall qualitative comparison is shown in the supplementary.

We show a comparison of real images of heavy rain in Fig. 9. All other methods fail to give faithful estimations, while our IntrinsicWeather provides reasonable estimations. We further validate the consistency of our inverse renderer across weather conditions in the supplementary materials. For each scene, we run the inverse renderer on images captured under different weather types and compute the PSNR between the recovered intrinsic maps and those obtained under sunny conditions.

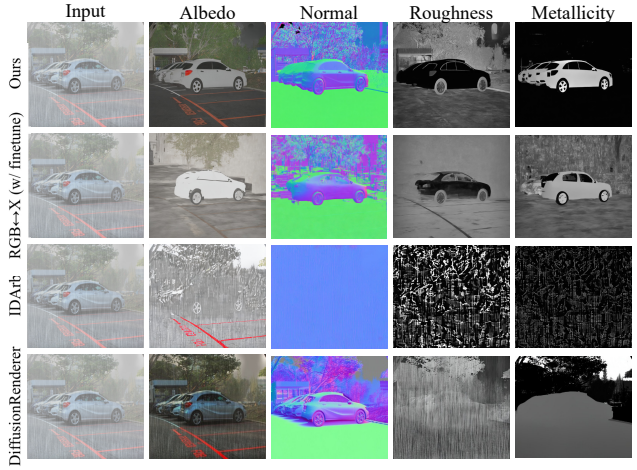

Figure 9. Qualitative comparison of inverse rendering on real-world data. All other methods are affected by rain, but ours removes the disturbance and generates a reasonable estimation. Other map comparisons are shown in the supplementary material.

Figure 10. Comparison of forward rendering results. We use the inverse renderer to obtain intrinsic maps of the original images, and then use the forward renderer to re-render images.

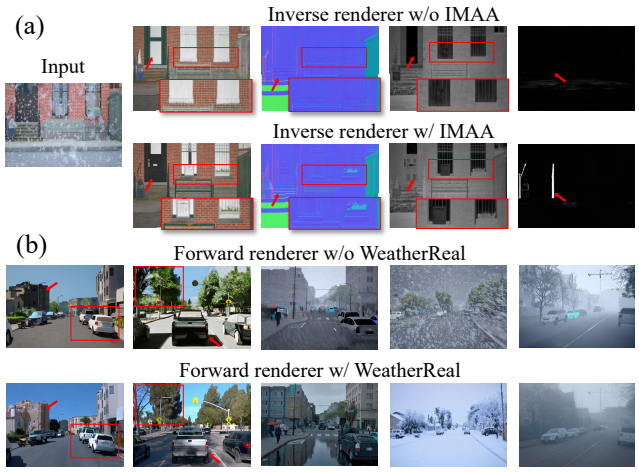

Figure 11. Ablation study. The inverse renderer with IMAA can focus on the details of the original image and recover detailed information. The forward renderer with WeatherReal can generate more realistic lighting, objects, and weather effects.

4.3.2. Forward rendering

A comparison of forward rendering between IntrinsicWeather, RGB↔X, and DiffusionRenderer [22] is shown in Fig. 10. We use the prompt "A sunny day in the city." for ours and RGB↔X, and provide an environment lighting for DiffusionRenderer. Our IntrinsicWeather produces images that better align with the text description than RGB↔X, and avoid abnormal textures and illuminations. DiffusionRenderer fails to recover all details, such as road signs and distant buildings. As shown in Tab. 1, our method achieves the highest metrics in rendering-based methods on all weather conditions. More results are shown in the supplementary.

4.4. Ablation study

Effect of IMAA. We train an inverse renderer without IMAA for the same steps. As shown in Tab. 3, the model without IMAA behaves poorly than our full model. We show a qualitative result in part (a) of Fig. 11. With the attention guidance related to the map provided by IMAA, the model produced more refined geometry and material predictions and successfully identified the metallic handrail, assigning it a reasonable level of metallicity.

Effect of WeatherReal. We explore the effect of WeatherReal. We train the forward renderer with only the synthetic dataset and with the WeatherReal dataset. The qualitative results are shown in part (b) of Fig. 11. We use the same intrinsic maps as input and evaluate each model. After training solely on the synthetic dataset, the forward renderer fails to reach high realism, resulting in unrealistic lighting and objects. After introducing WeatherReal, the model learn the distribution of the real-world data and then generates high-quality rendered images. We explore more ablation of datasets in the supplementary material.

Effect of intrinsic representation. We explore the necessity of intrinsic representation. We replace the intrinsic with a clean-weather image like WeatherWeaver [23], and we find that the model struggles to maintain scene details without physical inductive bias. The detailed analysis and results are available in the supplementary.

4.5. Limitations

Our framework is designed for single-image weather editing in the intrinsic space. Temporal modeling introduces additional factors, such as object motion and occlusion changes, that are orthogonal to our core contribution and require a video prior. Therefore, the current framework does not guarantee temporal consistency in video sequences. Extending IntrinsicWeather to videos would likely be possible by building on a video diffusion model, similar to DiffusionRenderer [22].

5. Conclusion

We propose IntrinsicWeather, a novel framework for controllable weather editing in intrinsic space. Our approach achieves robust intrinsic decomposition across diverse weather and illumination conditions while enabling controlled weather editing based on maps and prompts. For the inverse renderer, we propose IMAA to provide attention guidance to help the model focus on semantically important regions. For the forward renderer, we leverage CLIP interpolation and diffusion priors to achieve fine-grained weather control. Last, we construct two datasets, WeatherSynthetic and WeatherReal, containing intrinsic maps to address the lack of large-scale autonomous driving rendering datasets under varied weather conditions. Our IntrinsicWeather demonstrates performing weather editing in intrinsic space can achieve controllable and plausible editing while preserving geometry and material. Future work includes extending the framework to video-based weather editing and collecting more diverse and realistic data.

Acknowledgement

We thank the reviewers for the valuable comments. This work has been partially supported by the National Natural Science Foundation of China under grant No. 62572230.

References

[1] Stability AI. https://huggingface.co/stabilityai/stable-diffusion-3.5-medium, 2025. 3

[2] Tim Brooks, Aleksander Holynski, and Alexei A Efros. Instructpix2pix: Learning to follow image editing instructions. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 18392–18402, 2023. 1, 2, 5

[3] Mingdeng Cao, Xintao Wang, Zhongang Qi, Ying Shan, Xiaohu Qie, and Yinqiang Zheng. Masactrl: Tuning-free mutual self-attention control for consistent image synthesis and editing, 2023. 2

[4] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *European conference on computer vision*, pages 213–229. Springer, 2020. 6

[5] Zhifei Chen, Tianshuo Xu, Wenhang Ge, Leyi Wu, Dongyu Yan, Jing He, Luozhou Wang, Lu Zeng, Shunsi Zhang, and Yingcong Chen. Uni-renderer: Unifying rendering and inverse rendering via dual stream diffusion. *arXiv preprint arXiv:2412.15050*, 2024. 2

[6] Gautier Cosne, Adrien Juraver, Mélisande Teng, Victor Schmidt, Vahe Vardanyan, Alexandra Luccioni, and Yoshua Bengio. Using simulated data to generate images of climate change. *arXiv preprint arXiv:2001.09531*, 2020. 1, 2

[7] Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entezari, Jonas Müller, Harry Saini, Yam Levi, Dominik Lorenz, Axel Sauer, Frederic Boesel, et al. Scaling rectified flow transformers for high-resolution image synthesis. In *Forty-first international conference on machine learning*, 2024. 2

[8] Xiao Fu, Wei Yin, Mu Hu, Kaixuan Wang, Yuexin Ma, Ping Tan, Shaojie Shen, Dahua Lin, and Xiaoxiao Long. Geowizard: Unleashing the diffusion priors for 3d geometry estimation from a single image. In *European Conference on Computer Vision*, pages 241–258. Springer, 2024. 2, 5

[9] Andreas Geiger, Philip Lenz, Christoph Stiller, and Raquel Urtasun. Vision meets robotics: The kitti dataset. *The international journal of robotics research*, 32(11):1231–1237, 2013. 4

[10] Kai He, Ruofan Liang, Jacob Munkberg, Jon Hasselgren, Nandita Vijaykumar, Alexander Keller, Sanja Fidler, Igor Gilitschenski, Zan Gojcic, and Zian Wang. Unirelight: Learning joint decomposition and synthesis for video relighting, 2025. 2

[11] Amir Hertz, Ron Mokady, Jay Tenenbaum, Kfir Aberman, Yael Pritch, and Daniel Cohen-Or. Prompt-to-prompt image editing with cross attention control, 2022. 1, 2

[12] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020. 2

[13] Bahjat Kawar, Shiran Zada, Oran Lang, Omer Tov, Huiwen Chang, Tali Dekel, Inbar Mosseri, and Michal Irani. Imagic: Text-based real image editing with diffusion models, 2023. 2

[14] 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, 2023. 2, 5, 6

[15] Peter Kocsis, Vincent Sitzmann, and Matthias Nießner. Intrinsic image diffusion for indoor single-view material estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5198–5208, 2024. 2, 5, 7

[16] Black Forest Labs, Stephen Batifol, Andreas Blattmann, Frederic Boesel, Saksham Consul, Cyril Diagne, Tim Dockhorn, Jack English, Zion English, Patrick Esser, Sumith Kulal, Kyle Lacey, Yam Levi, Cheng Li, Dominik Lorenz, Jonas Müller, Dustin Podell, Robin Rombach, Harry Saini, Axel Sauer, and Luke Smith. Flux.1 kontext: Flow matching for in-context image generation and editing in latent space, 2025. 1, 2, 5

[17] Xuelong Li, Kai Kou, and Bin Zhao. Weather gan: Multi-domain weather translation using generative adversarial networks, 2021. 1, 2

[18] Yixuan Li, Lihan Jiang, Linning Xu, Yuanbo Xiangli, Zhenzhi Wang, Dahua Lin, and Bo Dai. Matrixcity: A large-scale city dataset for city-scale neural rendering and beyond. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3205–3215, 2023. 4

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

[20] Zhengqin Li, Mohammad Shafiei, Ravi Ramamoorthi, Kalyan Sunkavalli, and Manmohan Chandraker. Inverse rendering for complex indoor scenes: Shape, spatially-varying lighting and svbrdf from a single image. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2475–2484, 2020. 4

[21] Zhibing Li, Tong Wu, Jing Tan, Mengchen Zhang, Jiaqi Wang, and Dahua Lin. Idarb: Intrinsic decomposition for arbitrary number of input views and illuminations. *arXiv preprint arXiv:2412.12083*, 2024. 2, 5

[22] Ruofan Liang, Zan Gojcic, Huan Ling, Jacob Munkberg, Jon Hasselgren, Zhi-Hao Lin, Jun Gao, Alexander Keller, Nandita Vijaykumar, Sanja Fidler, et al. Diffusionrenderer: Neural inverse and forward rendering with video diffusion models. *arXiv preprint arXiv:2501.18590*, 2025. 2, 5, 8

[23] Chih-Hao Lin, Zian Wang, Ruofan Liang, Yuxuan Zhang, Sanja Fidler, Shenlong Wang, and Zan Gojcic. Controllable weather synthesis and removal with video diffusion models. *arXiv preprint arXiv:2505.00704*, 2025. 1, 2, 5, 8

[24] Yaron Lipman, Ricky TQ Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling. *arXiv preprint arXiv:2210.02747*, 2022. 2

[25] Luping Liu, Yi Ren, Zhijie Lin, and Zhou Zhao. Pseudo numerical methods for diffusion models on manifolds. *arXiv preprint arXiv:2202.09778*, 2022. 2

[26] Jundan Luo, Duygu Ceylan, Jae Shin Yoon, Nanxuan Zhao, Julien Philip, Anna Frühstück, Wenbin Li, Christian Richardt, and Tuanfeng Wang. Intrinsicdiffusion: Joint intrinsic layers from latent diffusion models. In *ACM SIGGRAPH 2024 Conference Papers*, New York, NY, USA, 2024. Association for Computing Machinery. 2

[27] Linjie Lyu, Valentin Deschaintre, Yannick Hold-Geoffroy, Miloš Hašan, Jae Shin Yoon, Thomas Leimkühler, Christian Theobalt, and Iliyan Georgiev. Intrinsicedit: Precise generative image manipulation in intrinsic space. *ACM Transactions on Graphics*, 44(4):1–13, 2025. 2

[28] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023. 4

[29] Gaurav Parmar, Krishna Kumar Singh, Richard Zhang, Yijun Li, Jingwan Lu, and Jun-Yan Zhu. Zero-shot image-to-image translation, 2023. 2

[30] William Peebles and Saining Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 4195–4205, 2023. 2

[31] Chenghao Qian, Wenjing Li, Yuhu Guo, and Gustav Markkula. Weatheredit: Controllable weather editing with 4d gaussian field, 2025. 2

[32] Sudarshan Rajagopalan and Vishal M Patel. Awracle: All-weather image restoration using visual in-context learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 6675–6683, 2025. 5

[33] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*, 1 (2):3, 2022. 2

[34] Mike Roberts, Jason Ramapuram, Anurag Ranjan, Atulit Kumar, Miguel Angel Bautista, Nathan Paczan, Russ Webb, and Joshua M Susskind. Hypersim: A photorealistic synthetic dataset for holistic indoor scene understanding. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 10912–10922, 2021. 4

[35] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10684–10695, 2022. 2

[36] 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. 2

[37] Christos Sakaridis, Haoran Wang, Ke Li, René Zurbrügg, Arpit Jadon, Wim Abbeloos, Daniel Olmeda Reino, Luc Van Gool, and Dengxin Dai. Acdc: The adverse conditions dataset with correspondences for robust semantic driving scene perception. *arXiv preprint arXiv:2104.13395*, 2021. 2, 5, 6

[38] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*, 2020. 2

[39] Pei Sun, Henrik Kretzschmar, Xerxes Dotiwalla, Aurelien Chouard, Vijaysai Patnaik, Paul Tsui, James Guo, Yin Zhou, Yuning Chai, Benjamin Caine, et al. Scalability in perception for autonomous driving: Waymo open dataset. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2446–2454, 2020. 4, 5

[40] Shangquan Sun, Wenqi Ren, Xinwei Gao, Rui Wang, and Xiaochun Cao. Restoring images in adverse weather conditions via histogram transformer, 2024. 5

[41] Sanjida Tasnim, Ashif Mahmud Mostafa, Azmain Morshed, Namreen Shaiyaz, Shakib Mahmud Dipto, Saad Aloteibi, Mohammad Ali Moni, Md Golam Rabiul Alam, and Md Ashraful Alam. Normalizing images in various weather and lighting conditions using colorpix2pix generative adversarial network. *Scientific Reports*, 15(1):33904, 2025. 1, 2

[42] Narek Tumanyan, Michal Geyer, Shai Bagon, and Tali Dekel. Plug-and-play diffusion features for text-driven image-to-image translation, 2022. 2

[43] Jeya Maria Jose Valanarasu, Rajeev Yasarla, and Vishal M. Patel. Transweather: Transformer-based restoration of images degraded by adverse weather conditions, 2021. 5, 7

[44] Chenfei Wu, Jiahao Li, Jingren Zhou, Junyang Lin, Kaiyuan Gao, Kun Yan, Sheng ming Yin, Shuai Bai, Xiao Xu, Yilei Chen, Yuxiang Chen, Zecheng Tang, Zekai Zhang, Zhengyi Wang, An Yang, Bowen Yu, Chen Cheng, Dayiheng Liu, Deqing Li, Hang Zhang, Hao Meng, Hu Wei, Jingyuan Ni, Kai Chen, Kuan Cao, Liang Peng, Lin Qu, Minggang Wu, Peng Wang, Shuting Yu, Tingkun Wen, Wensen Feng, Xiaoxiao Xu, Yi Wang, Yichang Zhang, Yongqiang Zhu, Yujia Wu, Yuxuan Cai, and Zenan Liu. Qwen-image technical report, 2025. 1, 2, 5

[45] Enze Xie, Wenhai Wang, Zhiding Yu, Anima Anandkumar, Jose M Alvarez, and Ping Luo. Segformer: Simple and efficient design for semantic segmentation with transformers. *Advances in neural information processing systems*, 34: 12077–12090, 2021. 6

[46] Zheng Zeng, Valentin Deschaintre, Iliyan Georgiev, Yannick Hold-Geoffroy, Yiwei Hu, Fujun Luan, Ling-Qi Yan, and Miloš Hašan. Rgb↔x: Image decomposition and synthesis using material- and lighting-aware diffusion models. In *ACM SIGGRAPH 2024 Conference Papers*, New York, NY, USA, 2024. Association for Computing Machinery. 2, 5, 7

[47] Jingsen Zhu, Fujun Luan, Yuchi Huo, Zihao Lin, Zhihua Zhong, Dianbing Xi, Rui Wang, Hujun Bao, Jiaxiang Zheng, and Rui Tang. Learning-based inverse rendering of complex indoor scenes with differentiable monte carlo raytracing. In *SIGGRAPH Asia 2022 Conference Papers*, pages 1–8, 2022. 4