

Generative and Spatial Modelling for Autonomous Driving under Adverse Weather

Chenghao Qian

Submitted in accordance with the requirements for the degree of
Doctor of Philosophy

The University of Leeds
Institute for Transport Studies

August 2026

Declaration

The candidate confirms that the work submitted is his own, except where work which has formed part of jointly authored publications has been included. The contribution of the candidate and the other authors to this work has been explicitly indicated below. The candidate confirms that appropriate credit has been given within the thesis where reference has been made to the work of others. The work contained in Chapters 4–7 of this thesis has appeared in the following peer-reviewed publications:

1. **Chapter 4** is based on **C. Qian**, M. Rezaei, S. Anwar, W. Li, T. Hussain, M. Azarmi, and W. Wang, “AllWeather-Net: Unified Image Enhancement for Autonomous Driving Under Adverse Weather and Low-Light Conditions,” in *International Conference on Pattern Recognition (ICPR)*, pp. 151–166, 2024. **Piero Zamperoni Overall Best Student Paper Award.**
2. **Chapter 5** is based on **C. Qian**, Y. Guo, W. Li, and G. Markkula, “WeatherGS: 3D Scene Reconstruction in Adverse Weather Conditions via Gaussian Splatting,” in *IEEE International Conference on Robotics and Automation (ICRA)*, 2025.
3. **Chapter 6** is based on **C. Qian**, Y. Guo, Y. Mo, and W. Li, “WeatherDG: LLM-assisted Procedural Weather Generation for Domain-Generalized Semantic Segmentation,” in *IEEE Robotics and Automation Letters (RA-L)*, vol. 10, no. 6, pp. 5919–5926, 2025.
4. **Chapter 7** is based on **C. Qian**, W. Li, Y. Guo, and G. Markkula, “WeatherEdit: Controllable Weather Editing with 4D Gaussian Field,” in *AAAI Conference on Artificial Intelligence (AAAI)*, 2026.

This copy has been supplied on the understanding that it is copyright material and that no quotation from the thesis may be published without proper acknowledgement.

Acknowledgements

This thesis marks the end of a long and meaningful journey. Along the way, I have been fortunate to be accompanied, supported, and encouraged by many people, to whom I owe my deepest gratitude.

I am grateful to Professor Gustav Markkula for the trust he has given me during my doctoral studies. His generous and supportive supervision allowed me to explore my academic interests, develop independent ideas, and grow with confidence as a researcher. I would also like to sincerely thank Dr Wenjing Li for her generous research support, insightful advice, and encouragement. Her guidance has helped me refine my ideas, strengthen my technical skills, and become a more rigorous researcher. I am also grateful to Professor Zia Wadud for his valuable feedback and broader perspective, which helped me think more critically about the wider implications of my work.

I would like to thank my parents for their unconditional love and support, both emotionally and financially. Their belief in me has given me the stability and confidence to continue this journey. Most importantly, I would like to thank my fiancée for her endless love, patience, and companionship over the past three years. She has stood by me through the most difficult and uncertain moments, without complaint, and has brought warmth and strength to my life. This thesis would not have been possible without her unwavering support.

They say that if you shoot for the moon, you may land among the stars. Having reached this meaningful chapter of my journey, I find myself standing among the stars. Yet the moon remains ahead, and the dream continues.

Contents

Declaration	i
Acknowledgements	ii
I Foundations and Research Questions	1
1 Introduction	2
1.1 Motivations	3
1.2 Background	4
1.2.1 Autonomous Driving Perception	4
1.2.2 The Adverse Weather Problem	6
1.3 Approaches to Weather Perception	10
1.4 Thesis Contributions	12
1.5 Thesis Organisation	12
2 Literature Review	13
2.1 Generative Models	13
2.1.1 Variational Autoencoders	14
2.1.2 Generative Adversarial Networks	15
2.1.3 Diffusion Models	17
2.1.4 Conditional and Controllable Generation	20
2.2 Spatial Models	22
2.2.1 3D Spatial Representation: Neural Radiance Fields	22
2.2.2 3D Spatial Representation: 3D Gaussian Splatting	25
2.3 Weather Restoration	28
2.3.1 Single-type Restoration	28
2.3.2 Unified, Multi-type Restoration	29
2.3.3 Low-light Enhancement	31
2.4 Weather Generation	32
2.4.1 Physics-Based Simulation	32
2.4.2 Neural Weather Synthesis	33
2.5 Summary	34

3	Research Gaps and Research Questions	36
3.1	Research Gaps	36
3.1.1	Gap 1: Unified Framework for Improving Clarity of Weather-Degraded Inputs	36
3.1.2	Gap 2: Simulation-Ready Scene Reconstruction from Weather-Degraded Captures	37
3.1.3	Gap 3: Scalable Adverse Weather Data Generation in Driving Context	38
3.1.4	Gap 4: Controllable and Dynamic Weather Data Generation for Driving Scene	39
3.2	The Data-Centric Framework	39
3.3	Research Questions	40
3.4	Chapter Roadmap	41

II Enhancement: Recovering Clean Representations from Degraded Observations 42

4	Improving Visual Clarity: AllWeatherNet	43
4.1	Introduction and Motivation	44
4.2	Related Work	45
4.3	Enhancement Pipeline	48
4.3.1	Residual Formulation	48
4.3.2	Theoretical Properties of the Residual Formulation	48
4.3.3	End-to-End Flow	50
4.4	Scaled Illumination-Aware Attention Mechanism	50
4.4.1	Motivation	50
4.4.2	Scaled Attention Formulation	51
4.4.3	Qualitative Behaviour of SIAM	52
4.5	Hierarchical Discrimination Framework	53
4.5.1	Scene-Level Discriminator	53
4.5.2	Object-Level Discriminator	54
4.5.3	Texture-Level Discriminator	55
4.5.4	Multi-Scale Synergy	55
4.6	Ranked Adaptive Window Pairing	55
4.6.1	The Misalignment Problem	55
4.6.2	Adaptive Window Search	56
4.6.3	Pairing Score and Best Patch Selection	56
4.7	Loss Function	56
4.8	Experimental Setup	57
4.8.1	Datasets	57

4.8.2	Evaluation Metrics	58
4.8.3	Comparison Methods	58
4.8.4	Implementation Details	59
4.9	Results and Analysis	59
4.9.1	Image Quality on ACDC	59
4.9.2	Image Quality on Dark Zurich	60
4.9.3	Semantic Segmentation Qualitative Results	60
4.9.4	Generalisation to Unseen Datasets	61
4.10	Ablation Studies	62
4.10.1	Component Ablation	62
4.10.2	Attention Mechanism Analysis	62
4.11	Summary	63
5	Reconstructing 3D Weather Scenes: WeatherGS	65
5.1	Introduction and Motivation	66
5.2	Related Work	68
5.3	Preliminaries: 3D Gaussian Splatting	69
5.4	Atmospheric Effect Filter	71
5.4.1	Motivation	71
5.4.2	Diffusion-Based Particle Removal	71
5.5	Lens Effect Detector	72
5.5.1	Motivation	72
5.5.2	AttGAN-Based Detection	73
5.5.3	Binary Mask Generation	73
5.6	Mask-Guided 3D Reconstruction	73
5.6.1	Masked Loss	73
5.6.2	Initialisation and Training Details	74
5.6.3	Dense-to-Sparse Integration	74
5.6.4	Multi-View Redundancy and the Limits of Mask-Guided Optimisation	75
5.7	Benchmark and Experimental Setup	75
5.7.1	A New Benchmark for Weather-Degraded 3D Reconstruction	75
5.7.2	Baselines	76
5.7.3	Metrics	76
5.8	Results	76
5.8.1	Quantitative Comparison	76
5.8.2	Qualitative Comparison	78
5.8.3	Training and Rendering Cost	78
5.9	Ablation Studies	80
5.9.1	Contribution of AEF and LED	80

5.9.2	Backbone Comparison: NeRF vs 3DGS	81
5.10	Discussion	81
5.11	Summary	83

III Generation: Synthesising Adverse Weather from Clean Inputs 84

6	Generalising to Adverse Weather: WeatherDG	85
6.1	Introduction	86
6.2	Related Work	89
6.3	Overview of the WeatherDG Pipeline	90
6.4	Stage 1: Driving Scene Fine-Tuning of Stable Diffusion	91
6.4.1	Why Fine-Tune	91
6.4.2	DreamBooth with the V^* Identifier	92
6.4.3	What V^* Encodes	94
6.5	Stage 2: Procedural Prompt Generation	94
6.5.1	Limitations of Single-Template Prompting	94
6.5.2	Three-Agent Decomposition	95
6.5.3	Instance Sampler $\mathcal{E}_{IS}$	95
6.5.4	Scene Composer $\mathcal{E}_{SC}$	97
6.5.5	Scene Descriptor $\mathcal{E}_{SD}$	97
6.6	Stage 3: Sample Generation and Segmentation-Model Training	98
6.6.1	Synthetic Image Generation	98
6.6.2	Pseudo-Target UDA Training	98
6.7	Experimental Setup	98
6.7.1	Datasets	98
6.7.2	Evaluation Metric and Backbones	99
6.7.3	Implementation Details	99
6.8	Results and Analysis	99
6.8.1	Comparison with State-of-the-Art Methods	99
6.8.2	Qualitative Segmentation Results	100
6.8.3	Comparison with Diffusion-Based Adaptation Methods	101
6.8.4	Per-Weather Breakdown on ACDC	102
6.9	Ablation Studies	102
6.9.1	Effect of Driving Scene Fine-Tuning	102
6.9.2	Contribution of Each LLM Agent	103
6.9.3	Effect of Synthetic-Dataset Size	103
6.10	Summary	104

7	Generating 3D Weather Scenes: WeatherEdit	105
7.1	Introduction	106
7.1.1	The Problem of 3D Weather Editing	106
7.1.2	Realism and Controllability Requirements	107
7.1.3	The WeatherEdit Approach	109
7.2	Related Work	109
7.3	Stage 1: Weather Background Editing	110
7.3.1	All-in-One LoRA Adapter	110
7.3.2	Semantic Conditioning	112
7.3.3	Temporal-View Attention	112
7.4	Stage 2: Weather Particle Construction	113
7.4.1	Gaussian Particle Field	113
7.4.2	Dynamic Simulation	115
7.4.3	Particle Lifecycle: Emission, Motion, and Recycling	115
7.4.4	Local Field Alignment	116
7.4.5	On the Constant-Velocity Approximation	116
7.5	Experimental Setup	117
7.5.1	Datasets	117
7.5.2	Baselines	118
7.5.3	Evaluation Metrics	118
7.5.4	Implementation	118
7.6	Results	119
7.6.1	Comparison with 2D-Based Editors	119
7.6.2	Comparison with 3D-Based Editors	120
7.6.3	Controllability	120
7.6.4	Downstream Utility for Perception	120
7.6.5	User Study	122
7.7	Ablation Studies	122
7.7.1	Effect of Semantic Conditioning	122
7.7.2	Effect of Temporal-View Attention	123
7.7.3	Effect of the Particle-Field Modules	124
7.8	Summary	124
8	Discussion and Conclusion	126
8.1	Summary of Contributions	127
8.1.1	AllWeatherNet: Unified 2D Image Enhancement	127
8.1.2	WeatherGS: 3D Reconstruction from Weather-Degraded Observa- tions	127
8.1.3	WeatherDG: LLM-Assisted 2D Data Generation	127
8.1.4	WeatherEdit: Controllable 3D Weather Generation	128

8.2	Cross-Chapter Analysis	128
8.2.1	Theme 1: From Enhancement to Generation	128
8.2.2	Theme 2: Generative and Spatial Modelling Are Complementary .	130
8.2.3	Theme 3: Composability Across the Framework	130
8.2.4	Theme 4: The Increasing Role of Foundation Models	131
8.2.5	Theme 5: Relation to General-Purpose World Models	132
8.3	Limitations	132
8.3.1	Method-Specific Limitations	132
8.3.2	Cross-Cutting Limitations	134
8.4	Broader Implications	135
8.4.1	Data-Centric AI Safety	135
8.4.2	Ethical Considerations	136
8.5	Future Directions	137
8.5.1	Multi-Sensor Extension	137
8.5.2	Weather-Adaptive Perception Models	137
8.5.3	Real-Time and Resource-Constrained Deployment	137
8.5.4	Extension to Additional Adverse Conditions	138
8.5.5	Foundation Models for Zero-Shot Weather Adaptation	138
8.6	Closing Remarks	139
A	Extended Experimental Results	141
A.1	AllWeatherNet: Per-Class and Per-Condition Analysis	141
A.1.1	Per-Condition Breakdown	142
A.2	WeatherDG: Detailed Domain-Generalisation Results	143
A.2.1	Per-Condition Breakdown across Backbones	143
A.2.2	Class Balance under Instance Sampling	143
A.3	WeatherGS: Scene-Level and Backbone Analysis	144
A.3.1	Scene Construction Details	144
A.3.2	Additional Real-World Scenes	145
B	Implementation Details and Hyperparameters	146
B.1	WeatherGS (Chapter 5)	146
B.1.1	Atmospheric Effect Filter	146
B.1.2	Lens Effect Detector	146
B.1.3	3DGS Training	146
B.1.4	Benchmark Generation	146
B.2	WeatherDG (Chapter 6)	147
B.2.1	Diffusion Fine-Tuning	147
B.2.2	LLM Agents	148
B.2.3	Image Generation	148
B.2.4	UDA Segmentation Training	148

B.3	WeatherEdit (Chapter 7)	148
B.3.1	All-in-One Adapter	148
B.3.2	Semantic Conditioning	148
B.3.3	Training Hyperparameters	149
B.3.4	TV Attention Parameters	149
B.3.5	Gaussian Particle Field	149
B.4	Computational Environment	150
C	Dataset Details and Evaluation Protocols	151
C.1	Training Datasets	151
C.1.1	Cityscapes	151
C.1.2	ACDC: Adverse Conditions Dataset with Correspondences	151
C.1.3	Dark Zurich	152
C.1.4	BDD100K	152
C.1.5	Foggy Zurich and Nighttime Driving	152
C.1.6	MUSES	152
C.2	3D Reconstruction Datasets	153
C.2.1	Synthetic Scenes for WeatherGS	153
C.2.2	Real-World Scenes for WeatherGS	153
C.2.3	Driving Scenes for WeatherEdit	153
C.3	Evaluation Protocols	154
C.3.1	Semantic Segmentation	154
C.3.2	Image Quality Metrics	154
C.3.3	CLIP-Based Metrics	155
C.3.4	Consistency Metrics	155
C.3.5	User-Study Protocol	155
C.4	Dataset Licensing and Ethical Use	156
D	LLM Prompt Templates and Generation Examples	157
D.1	WeatherDG Three-Agent Pipeline	157
D.1.1	Instance Sampler ($\mathcal{E}_{IS}$)	157
D.1.2	Scene Composer ($\mathcal{E}_{SC}$)	158
D.1.3	Scene Descriptor ($\mathcal{E}_{SD}$)	158
D.1.4	Pipeline Composition	159
D.2	WeatherEdit Prompts for the All-in-One Adapter	160
D.2.1	Adapter Training Prompts	160
D.2.2	Inference Prompts	160
D.3	Failure Modes and Failure Prompts	161
D.3.1	WeatherDG Failure Modes	161
D.3.2	WeatherEdit Failure Modes	161
D.4	Sample Generated Images	161

E	Publication Timeline and Author Contributions	163
E.1	Publication Timeline	163
E.2	Candidate Contributions	163
E.2.1	AllWeather-Net (ICPR 2024)	163
E.2.2	WeatherDG (IEEE RA-L 2025)	164
E.2.3	WeatherGS (ICRA 2025)	165
E.2.4	WeatherEdit (AAAI 2026)	165
E.3	Research Outputs Beyond the Four Core Papers	166
E.3.1	Code Releases	166
E.3.2	Dataset Releases	166
E.3.3	Conference Presentations	166
E.3.4	Awards and Recognition	167
F	Notation and Glossary	168
F.1	Mathematical Notation	168
F.1.1	Scalars, Vectors, Matrices	168
F.1.2	Common Operators	168
F.2	Notation Index by Chapter	169
F.2.1	Chapter 1: Introduction	169
F.2.2	Chapter 2: Literature Review	169
F.2.3	Chapter 4: AllWeatherNet	169
F.2.4	Chapter 5: WeatherGS	170
F.2.5	Chapter 6: WeatherDG	170
F.2.6	Chapter 7: WeatherEdit	171
F.3	Glossary of Abbreviations	171
F.3.1	Methods and Frameworks	171
F.3.2	Component-Level	172
F.3.3	Datasets and Benchmarks	172
F.3.4	Metrics	172
F.3.5	Frameworks and Standards	172
F.3.6	General Computing	173

List of Figures

1.1	Representative perception outputs on a single urban driving frame. . . .	4
1.2	Sensing modalities.	5
1.3	Principal visual artefacts of rain in camera images. Illustration generated by the author.	6
1.4	Principal visual artefacts of snow in camera images. Illustration generated by the author.	7
1.5	A driving scene under three fog densities, parameterised by the scattering coefficient β in Eq. (1.1). Illustration generated by the author.	8
1.6	Representative weather-induced perception failures on ACDC. Images and ground-truth annotations are from ACDC (Sakaridis et al., 2021); the SegFormer predictions were produced by the author using Xie et al. (2021).	8
2.1	The VAE pipeline.	14
2.2	The GAN training pipeline.	15
2.3	The diffusion forward process applied to a driving frame. The driving frame and visualisation were generated by the author using the forward noising formulation of Ho et al. (2020).	18
2.4	The latent diffusion architecture (Rombach et al., 2021). Reproduced from the Open Access version of the paper at https://openaccess.thecvf.com/CVPR2022	21
2.5	ControlNet (Zhang et al., 2023b) outputs under two spatial conditions. Reproduced from the Open Access version of the paper at https://openaccess.thecvf.com/ICCV2023	22
2.6	The NeRF training pipeline (Mildenhall et al., 2020). Reproduced from the Open Access version of the paper at https://www.ecva.net/papers/eccv_2020/	23
2.7	The 3D Gaussian Splatting pipeline.	25
2.8	Representative single-type weather restoration; each pair shows the degraded input (left) and restored output (right). Examples reproduced from the official WeatherDiffusion repository accompanying Özdenizci and Legenstein (2023).	28

2.9	Three architectural families for handling rain, snow, and haze within one restoration system, adapted from (Jose Valanarasu et al., 2022) at https://openaccess.thecvf.com/CVPR2022	30
2.10	Representative low-light driving image (left) and enhanced result (right). Example from EnlightenGAN (Jiang et al., 2021).	31
2.11	Physics-based weather synthesis in a graphics simulator.	33
2.12	A representative latent-space neural weather-synthesis pipeline (Lin et al., 2025). Reproduced from the Open Access version of the paper at https://openaccess.thecvf.com/ICCV2025	34
3.1	The data-centric framework of this thesis.	40
4.1	Overview of the AllWeatherNet architecture. A source scene patch P_s^s is processed by the Scaled Illumination-Aware Attention Mechanism (SIAM) and the generator to produce an enhancement mask M^s . The mask is added to the source patch to form the enhanced result P_f^s , which is evaluated by three hierarchical discriminators operating at scene, object, and texture scales. Ranked Adaptive Window Pairing (RAWP, not shown in this overview but detailed in Section 4.6) selects the target patches for the object and texture discriminators.	46
4.2	Qualitative comparison of representative image processing approaches on an adverse weather driving scene. (a) Input image. (b) A weather effect removal method (Chen et al., 2022b) preserves the overall scene structure but introduces noticeable colour imbalance. (c) A pixel-level translation method (Zhu et al., 2017) produces visually plausible appearance changes but hallucinates scene content. (d) A low-light enhancement method (Ma et al., 2022b) over-amplifies already bright regions, leading to saturation in snowy areas. (e) AllWeatherNet restores colour, contrast, and texture while preserving scene content and avoiding visible artefacts.	46
4.3	Comparison between pixel-level translation and residual image enhancement. (a) Pixel-level translation directly generates the enhanced image and therefore reconstructs all image content. (b) Residual image enhancement predicts an enhancement mask that is added to the input image, allowing unchanged regions to be preserved explicitly.	49
4.4	Integration of SIAM into the generator. The illumination-aware attention map is computed from the input image and injected at each scale of the U-Net generator to modulate intermediate feature activations.	52
4.5	Comparison between naive illumination attention and the scaled attention used in SIAM. The scaled formulation preserves higher attention over a broader illumination range, reducing under-attention to bright weather-affected regions.	52

4.6	Qualitative comparison between naive illumination attention and SIAM. In foggy and snowy scenes, SIAM preserves attention in bright weather-affected regions that receive limited attention under the naive formulation. In nighttime scenes, both formulations assign high attention to dark regions, consistent with the low-light enhancement setting.	53
4.7	Hierarchical discrimination framework. Scene-level patches supervise global coherence; object-level patches, with one quarter of the scene-patch side length, supervise the structural fidelity of individual scene elements; texture-level patches, obtained by centre-cropping object-level patches, supervise fine-grained local surface detail.	54
4.8	Qualitative comparison of AllWeatherNet against weather effect removal, pixel-level translation, and low-light enhancement baselines, under fog, snow, night, and rain. Zoom-in red regions highlight visual distinctions. .	59
4.9	Semantic segmentation predictions with and without AllWeatherNet enhancement. Zoom-in white regions highlight improved separation of semantic classes.	61
4.10	Qualitative generalisation results on Foggy Zurich and Nighttime Driving, both held out from training. AllWeatherNet improves clarity of vehicles, signs, and buildings in fog, and corrects colour casts from artificial lighting at night.	61
4.11	Naive attention versus SIAM attention maps across the four adverse conditions. Darker colours indicate higher attention scores. SIAM attends to weather-affected regions even when they are bright (fog, snow); naive attention does not.	63
4.12	Enhanced outputs from models trained with naive attention and with SIAM. The SIAM-trained model restores distant foggy objects that the naive-attention model leaves veiled.	63
5.1	Rendering failures of 3D reconstruction methods under adverse weather. NeRF tends to average view-inconsistent weather artefacts, producing blurred regions and residual occlusions. 3DGS can instead incorporate weather particles and lens droplets into the explicit scene representation. WeatherGS mitigates these failure modes through dense-to-sparse preprocessing and produces cleaner renderings that better preserve the underlying scene.	68
5.2	Atmospheric Effect Filter. A weather-degraded input image and a text prompt, such as “Remove the snowy effect in the image”, are encoded using CLIP text and vision encoders. A cosine-similarity criterion selects the appropriate restoration plugin, such as “Desnow”, which conditions a pre-trained Stable Diffusion model to remove dense atmospheric particles while preserving the underlying scene content.	71

5.3	Overview of WeatherGS. The Atmospheric Effect Filter first removes dense atmospheric particles. The Lens Effect Detector then operates on the cleaned images to identify sparse lens occlusions. The resulting images and masks are used in a mask-guided 3D Gaussian Splatting optimisation procedure to reconstruct a clean 3D scene representation.	74
5.4	Qualitative results on synthetic (a) and real-world (b) scenes. NeRF-based methods show the characteristic blurring pattern; 3DGS methods reconstruct weather particles and droplets; WeatherGS produces clean renderings.	79
5.5	Ablation of the AEF and LED modules. Without AEF: visible weather particles. Without LED: visible lens droplets. Full system: clean reconstruction.	80
5.6	Backbone comparison: the same preprocessing pipeline with a NeRF backbone versus a 3DGS backbone. The 3DGS variant produces sharper reconstructions at a fraction of the training and rendering cost.	81
6.1	Representative failure modes of alternative sources for adverse weather training images. (a) A driving simulator (Kerim et al., 2022) provides geometric control but produces visually synthetic scenes with coarse textures and simulator-like colour statistics. (b) Vanilla Stable Diffusion (Rombach et al., 2021) generates visually appealing images, but often violates dashcam geometry through artistic composition and incorrect camera perspective. (c) WeatherDG generates photorealistic adverse weather driving images with dashcam-compatible viewpoint and scene layout, closely resembling (d) real adverse weather driving images.	88
6.2	Overview of the WeatherDG pipeline. (a) Stable Diffusion is adapted to dashcam-perspective driving scenes through DreamBooth fine-tuning on a single source-domain image, binding the identifier V^* to driving scene geometry and layout. (b) A three-agent LLM pipeline, consisting of an Instance Sampler, Scene Composer, and Scene Descriptor, generates detailed and class-balanced weather prompts. (c) The fine-tuned diffusion model synthesises adverse weather driving images from these prompts. The generated images are then combined with the labelled source dataset to train a segmentation model using UDA-style self-training.	90
6.3	Representative outputs from vanilla Stable Diffusion and the DreamBooth-adapted WeatherDG model under the prompt template “A photo of [CLS].” The fine-tuned model produces images with dashcam-compatible viewpoints and driving scene composition, whereas the vanilla model often generates artistic, stylised, or geometrically unsuitable outputs.	92

6.4	DreamBooth fine-tuning of Stable Diffusion with the V^* identifier. After fine-tuning on a single source-domain image, the model can generate dashcam-perspective driving images across diverse class and weather prompts.	93
6.5	Incremental effect of the three LLM agents on the generated image. Rows correspond to prompts produced by $\mathcal{E}_{IS}$ alone (top), $\mathcal{E}_{IS} + \mathcal{E}_{SC}$ (middle), and the full $\mathcal{E}_{IS} + \mathcal{E}_{SC} + \mathcal{E}_{SD}$ pipeline (bottom). The Scene Descriptor enriches the prompt with detailed weather cues, transforming a generic weather-labelled image into a more photorealistic adverse weather scene.	96
6.6	Segmentation predictions of MIC (red box) versus MIC+WeatherDG (green box) on ACDC. WeatherDG augmentation resolves weather-induced failure modes including misclassification of wet-road reflections as vehicles, loss of sidewalk boundaries in fog, and failure to detect pedestrians on snowy roads.	101
6.7	Effect of synthetic-dataset size on downstream segmentation performance. The gains increase with the number of WeatherDG-generated images and saturate at approximately 1,000 samples.	104
7.1	Given clear weather multi-view and multi-frame driving images rendered from an input 3D scene, WeatherEdit generates the same scene under user-specified weather conditions, including fog, snow, and rain, with controllable severity levels. The generated results preserve multi-view and multi-frame consistency while supporting both background appearance changes and dynamic weather effects.	108
7.2	Single-style adapters versus the All-in-One adapter. The unified design supports multiple weather styles within one model, reducing training and storage cost while enabling knowledge sharing across related weather transformations.	111
7.3	Temporal-View attention. Side cameras attend to the centre camera to improve view consistency, while each frame attends to adjacent frames to improve temporal consistency. The mechanism is parameter-free and augments the standard self-attention of the pre-trained diffusion model.	114
7.4	Weather particle construction pipeline. Attribute modelling defines weather-specific particle distributions; dynamic simulation updates particles over time; local field alignment keeps the particle field centred around the rendering camera. The resulting Gaussian particle field is rendered together with the reconstructed 3D scene to produce the final weather-edited view.	117

7.5	Qualitative comparison with 2D-based editors. ControlNet over-stylises and modifies scene content; InstructPix2Pix shows inconsistency across adjacent frames and views; TurboEdit stylises correctly but distorts scene content (pedestrians become reflections, trees disappear). WeatherEdit preserves scene integrity and achieves multi-view and multi-frame consistency.	119
7.6	Qualitative comparison with 3D-based weather editors on snowy editing. 3D Stylisation produces a faint effect; ClimateNeRF produces unnatural snow cover without matching the atmospheric tone; WeatherEdit produces realistic snow with appropriate atmospheric context.	120
7.7	Controllable weather effect and severity across three driving datasets. WeatherEdit generates (a) the original scene, (b) the 2D-edited background, and (c) the composited particle fields at light, moderate, and heavy severity. .	121
7.8	User preference for the most realistic weather editing results across three weather conditions. WeatherEdit receives the majority of votes in all settings: 84.5% for fog, 62.1% for rain, and 79.3% for snow.	122
7.9	Effect of semantic conditioning. (a) Input image and semantic map. (b) With semantic conditioning, snow is placed in semantically plausible regions such as sidewalks, trees, and non-active surfaces. (c) Without semantic conditioning, snow is distributed more uniformly, including over road regions where it should be reduced or cleared.	123
7.10	Effect of Temporal-View attention. (a) Without TV attention, the rightmost view has a noticeably darker tone than the adjacent view, leading to cross-view inconsistency. (b) With TV attention, the edited appearance remains more consistent across views.	123
7.11	Ablation of the particle-field modules. (a) Original scene. (b) Without 2D background editing, dynamic particles appear over an otherwise clear scene. (c) Without attribute modelling, particles appear as implausible coloured blobs. (d) Without dynamic simulation, particles are visible but static. (e) Without local field alignment, the camera moves out of the particle field and the weather effect disappears. (f) The full system produces natural, camera-aligned, and dynamic weather.	124

List of Tables

1.1	Scale of publicly available autonomous driving datasets with camera imagery, grouped by collection environment.	10
4.1	Quantitative evaluation of image quality and downstream semantic segmentation on the ACDC validation set. Best results are in bold	60
4.2	Quantitative evaluation on the Dark Zurich nighttime validation set. Best results are in bold	60
4.3	mIoU of PSPNet on original and AllWeatherNet-enhanced versions of the Foggy Zurich and Nighttime Driving test sets. Both datasets are held out from training.	61
4.4	Ablation study on AllWeatherNet components. SSIM is reported on the ACDC validation set. Each row adds one component to the row above. .	62
5.1	PSNR, SSIM, and LPIPS on snowy scenes.	77
5.2	PSNR, SSIM, and LPIPS on rainy scenes.	77
5.3	Training and rendering cost comparison. Metrics are averaged across synthetic scenes. The 3DGS backend is approximately $14\times$ faster to train and $1,600\times$ faster to render than the NeRF variant, with better reconstruction quality.	78
5.4	Ablation of AEF and LED on the full benchmark. Metrics averaged across all synthetic scenes.	80
6.1	Comparison with state-of-the-art domain-generalisation methods (mIoU, %). All methods are trained on Cityscapes and evaluated on ACDC, BDD100K, and Dark Zurich.	100
6.2	Comparison with other diffusion-based adaptation methods. Baseline is HRDA+MiT-B5, trained on Cityscapes; all augmentation methods use their own diffusion-generated images as pseudo-target.	101
6.3	Per-weather mIoU on ACDC using WeatherDG augmentation. All methods use the MiT-B5 encoder.	102
6.4	Effect of DreamBooth fine-tuning on segmentation mIoU. The segmentation backbone is DAFormer with MiT-B5.	102

6.5	Contribution of each LLM agent. Results are reported as mIoU on ACDC using DAFormer with MiT-B5.	103
7.1	Weather-specific particle parameterisation.	115
7.2	Comparison with 2D-based weather editors. Higher is better.	119
7.3	Comparison with 3D-based weather editors. The high CLIP-S score of 3D Stylisation reflects minimal scene modification (it fails to edit) rather than editing quality.	120
7.4	Semantic segmentation mIoU (%) with and without WeatherEdit-generated data augmentation.	121
7.5	Effect of TV attention components. Temporal attention reduces warp error between adjacent frames, while spatial view attention reduces Bhattacharyya distance across views. The full mechanism provides the best overall consistency.	124
8.1	The four methods of the thesis, grouped by Part (capability) and then by dimensionality, characterised along five axes.	128
8.2	Comparison between inference-time enhancement and training-time generation for adverse-weather semantic segmentation on ACDC. Values report mIoU, with improvements over the source-only baseline shown in parentheses.	129
A.1	Per-class mIoU (%) on the ACDC validation set, comparing no enhancement, two representative baselines, and AllWeatherNet. Best per-class result in bold	142
A.2	Per-condition mIoU (%) of AllWeatherNet on the ACDC validation set.	142
A.3	Per-condition mIoU (%) on the ACDC validation set for WeatherDG with three backbones and two encoder choices.	143
A.4	Fractional pixel occupancy (%) of “thing” classes in the Cityscapes source, the ACDC target, and the WeatherDG generated set.	144
B.1	WeatherGS 3DGS training hyperparameters.	147
B.2	WeatherDG Stable Diffusion fine-tuning hyperparameters.	147
B.3	WeatherEdit training hyperparameters for the All-in-One adapter.	149
B.4	WeatherEdit Gaussian particle field parameters.	149
D.1	WeatherDG generated training-set composition.	162
E.1	Publication timeline of the four core papers.	163

Abstract

Autonomous driving promises to transform transportation by enabling automated perception and navigation in complex real-world environments. Vision perception has emerged as a compelling paradigm for autonomous driving, using visual inputs to provide dense cues for semantic understanding and spatial reasoning. However, its reliance on optical inputs renders such systems vulnerable to adverse weather, such as heavy rain and dense fog. These conditions corrupt the sensor observations through visual degradations, disrupting downstream perception and planning models and compromising driving safety.

To address these challenges, this thesis advocates a data-centric framework structured around two primary directions: *data enhancement* for recovering clean inputs from corrupted observations, and *data generation* for creating synthetic driving scenarios for training augmentation. To realize this framework, this thesis delivers a suite of generative and spatial modelling methodologies for both data enhancement and data generation, progressing from 2D image-level processing to 3D scene-level representations. For data enhancement, I first introduce a unified image enhancement framework capable of mitigating diverse visual degradations within a single network architecture. I then extend enhancement from image space to 3D scene reconstruction by explicitly decoupling volumetric atmospheric scattering and camera lens occlusions, enabling the recovery of cleaner scene geometry and appearance under adverse weather. Along the data generation direction, I first develop an LLM-assisted diffusion framework that incorporates autonomous driving domain knowledge into generative modelling and synthesizes corner-case scenarios to improve domain-generalized semantic segmentation. I further advance generation into 3D-aware dynamic scene editing through a controllable weather editing framework operating on a 4D Gaussian field. This framework enables the synthesis of temporally coherent and physically plausible weather scenarios while preserving the underlying 3D scene structure.

Across these settings, the proposed methods achieve competitive or superior performance against relevant task-specific baselines, while the enhanced and generated data improve perception tasks such as semantic segmentation under adverse weather conditions. In conclusion, this thesis advances weather-robust autonomous driving by combining generative priors with spatial scene representations for data enhancement and generation, establishing a scalable data-centric framework for robust visual perception.

Part I

Foundations and Research Questions

Chapter 1

Introduction

Autonomous vehicles have the potential to transform road transportation by reducing collision risk, improving traffic efficiency, and enabling safer mobility at scale. Achieving this promise requires reliable operation in complex and dynamic urban environments. At the core of this requirement lies visual perception: an autonomous vehicle must continuously observe its surroundings, infer the semantic and geometric structure of the scene, and use this understanding to support safe decision-making.

Among various sensors, cameras serve as a primary sensing modality for visual perception, underpinning a wide range of downstream tasks such as 3D object detection, depth estimation, scene reconstruction, and semantic segmentation. Despite their importance, cameras remain inherently vulnerable to adverse weather. Such conditions can degrade visual observations, compromise perception performance, and ultimately undermine the safety-critical decisions made by autonomous driving systems. To address these challenges, this thesis examines visual perception under adverse weather from a data-centric perspective and investigates generative and spatial modelling for weather-robust autonomous driving. Generative models provide powerful image-level priors for restoring weather-degraded observations and synthesising weather-affected data when real adverse-weather samples or paired references are limited. Spatial models lift visual evidence from 2D images into explicit 3D representations, providing the geometric structure required for multi-view consistency, controllable editing, and simulation-grade rendering. By combining these two modelling paradigms, this thesis develops a pipeline that enhances available weather-degraded observations and synthesises long-tailed adverse weather data.

Overview of the main contributions. This thesis develops four complementary methods. **AllWeatherNet** provides unified 2D image enhancement across rain, snow, fog, and low light. **WeatherGS** combines atmospheric-effect filtering and lens-occlusion detection with Gaussian Splatting to reconstruct clean 3D scenes from weather-degraded views. **WeatherDG** combines an adapted diffusion model with LLM-guided prompt generation to synthesise adverse weather training images without real adverse weather reference images. **WeatherEdit** combines background editing with a dynamic Gaussian particle field to generate controllable weather effects in reconstructed 3D scenes. To-

gether, these contributions address data enhancement and data generation at both image and scene levels. Section 1.4 summarises the contributions, with the methods and their experimental evaluation presented in Chapters 4–7.

The remainder of this chapter is organised as follows. Section 1.1 presents the safety motivation for adverse weather perception. Section 1.2 reviews vision-based autonomous driving perception and the visual phenomena induced by adverse weather. Section 1.3 surveys existing approaches and identifies four persistent research gaps that motivate the four technical chapters of this thesis. The final sections summarise the thesis contributions and outline the structure of the remaining chapters.

1.1 Motivations

Weather degrades both visual perception and road safety. An analysis of crash studies reports accident rates 71% above baseline in rain and 84% above baseline in snowfall (Becker et al., 2022), indicating that adverse weather directly increases the risk and complexity of real-world driving. This challenge also extends to autonomous perception: semantic segmentation models can suffer substantial performance degradation, with drops of more than 20 mIoU points reported in fog, rain, and snow (Sakaridis et al., 2021; Hahner et al., 2019; Zang et al., 2019). These two patterns point to the core difficulty of driving in adverse weather: such conditions not only make the driving environment more dangerous, but also reduce the reliability of visual perception in autonomous systems. Since rain, snow, and fog account for a substantial fraction of annual driving time across populated regions, adverse weather should not be treated as a rare corner case. It is a persistent, safety-critical, and operationally unavoidable source of perception failure.

Despite its importance, adverse weather remains a substantially open problem for autonomous driving. The difficulty stems from two coupled bottlenecks: visual degradation and data scarcity. First, adverse weather introduces complex visual corruption that can blur object boundaries, suppress texture, attenuate distant regions, and etc. Consequently, it distorts the semantic and geometric evidence on which perception models depend. Second, the adverse weather training distribution is severely under-sampled. Large-scale autonomous driving benchmarks are predominantly collected under normal or common driving conditions. For example, nuScenes (Caesar et al., 2020) provides 1.4 million camera images, but its visual data are largely dominated by clear or non-extreme weather and it is not specifically designed as an adverse-weather benchmark. By contrast, datasets explicitly targeting adverse conditions remain much smaller: ACDC focuses on fog, nighttime, rain, and snow, but provides only 4,006 annotated adverse-condition images (Sakaridis et al., 2021). This imbalance is especially problematic because modern perception models rely heavily on large and diverse training distributions.

Consequently, the dual challenges of visual degradation and severe data scarcity create a formidable barrier to safe, year-round autonomous vehicles deployment. Robust

adverse weather perception requires more than improving a single model or collecting a larger dataset: it requires methods that can recover visual evidence from degraded inputs and expand the under-sampled adverse weather training distribution. This thesis addresses these challenges through generative and spatial modelling, developing methods for weather restoration and weather generation.

1.2 Background

1.2.1 Autonomous Driving Perception

Perception tasks. Autonomous driving perception systems transform raw sensor observations into structured scene representations that can support prediction, planning, and control. It comprises a set of related tasks, each capturing a different abstraction of the driving environment. 3D object detection localises and classifies discrete agents, such as vehicles, pedestrians, and cyclists, using oriented 3D bounding boxes that encode their position, extent, and orientation (Wang et al., 2021; Li et al., 2022). Depth estimation recovers per-pixel or sparse range information from stereo correspondences, monocular cues, or multi-view geometry, providing the spatial structure required for reasoning about distance and free space (Ranftl et al., 2021). Semantic segmentation assigns a class label to every image pixel, producing a dense semantic description of the scene (Cordts et al., 2016). Fig. 1.1 illustrates these three perception outputs on the same driving frame.

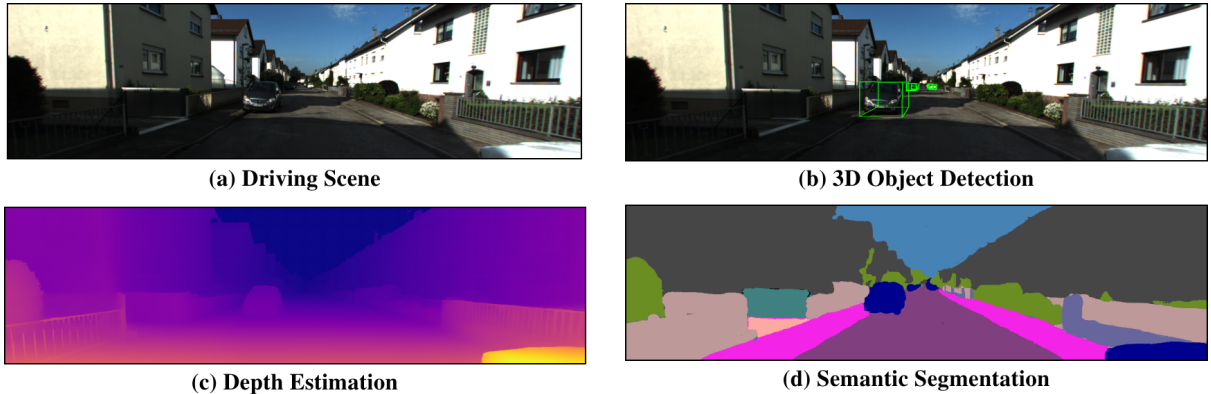


Figure 1.1: Representative perception outputs on a single urban driving frame.

Among these tasks, semantic segmentation is particularly important. It provides a dense, class-aware interpretation of the image by assigning each pixel to a semantic category (Long et al., 2015; Zhao et al., 2017; Xie et al., 2021). This output encodes both scene layout, such as road, sidewalk, sky, and buildings, and scene content, such as pedestrians, vehicles, traffic signs, and other safety-critical objects. Its density makes it a strong diagnostic signal for autonomous driving perception: a planner can reason about drivable space, object boundaries, occlusion, and safety margins only insofar as these semantic cues are correctly recovered. Modern segmentation architectures span fully convolutional encoder-decoders (Long et al., 2015; Ronneberger et al., 2015), atrous-pyramid

networks (Chen et al., 2018a,b; Zhao et al., 2017), and transformer-based hierarchies (Xie et al., 2021). Despite these architectural differences, they share a common output structure: a dense class map at or near the input resolution. They are also commonly evaluated using mean Intersection-over-Union (mIoU) over a fixed class taxonomy (Cordts et al., 2016).

Sensing modalities. Automated vehicle perception systems can be built upon multiple sensing modalities including camera, LiDAR and Radar. LiDAR measures the time of flight of laser pulses and produces accurate 3D point clouds with metric depth, but its semantic information is limited. Radar estimates range and Doppler velocity from millimetre-wave signals but it provides sparse spatial measurements and limited class evidence. Cameras project the visible spectrum onto a dense pixel grid, capturing colour, texture, and fine angular detail over a wide field of view. They therefore remain one of the primary sensing modalities for extracting dense environmental and semantic information in autonomous-driving perception (Huang et al., 2022). Fig. 1.2 contrasts the three main modalities: a colour camera, a spinning LiDAR, and an automotive radar, together with their corresponding outputs.

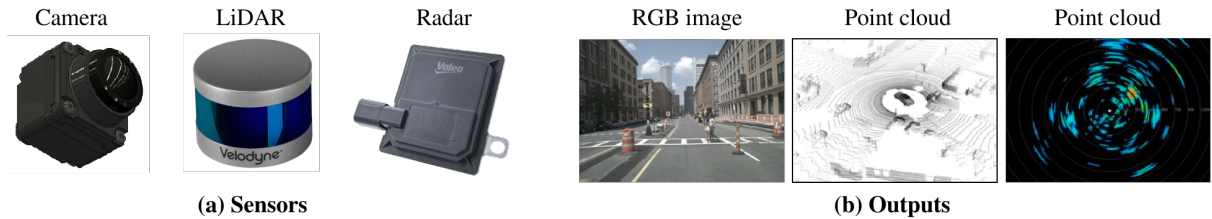


Figure 1.2: Sensing modalities.

Two properties make cameras especially important in autonomous driving. First, they provide dense colour and texture cues that are essential for semantic interpretation. Lane markings, traffic-sign symbols, pedestrian appearance, road boundaries, and object contours are all expressed in the visible spectrum and require pixel-level resolution. Second, cameras are inexpensive relative to LiDAR or high-resolution radar, making them scalable to the six- or eight-camera surround-view configurations now common in production vehicles. This combination of dense visual information and low unit cost has made camera-based perception a key interface between the external world and the autonomy stack. It also makes cameras a major entry point through which adverse weather degradation propagates downstream into prediction, planning, and control.

Dense prediction tasks, such as semantic segmentation and depth estimation, require reliable pixel-level cues and are particularly sensitive to image degradation caused by adverse weather. An object detector may tolerate moderate localisation noise if the predicted bounding box still encloses the target. A segmentation model, by contrast, is evaluated at the pixel level, so small radiometric or geometric corruptions can directly appear as visible errors in the predicted label map. Reduced contrast may erase lane boundaries; a lens droplet may occlude a pedestrian; fog may merge distant vehicles with

the road surface; and accumulated snow may overwrite the road-marking texture needed for lane interpretation. These failures directly affect class-level mIoU and, more importantly, compromise the semantic evidence required for safe decision-making. Semantic segmentation is therefore both the principal perception target of this thesis and a natural diagnostic for measuring the severity of weather degradation and the effectiveness of the proposed remedies.

1.2.2 The Adverse Weather Problem

1.2.2.1 Weather Phenomena

Adverse weather degrades camera imagery primarily through three main types of image-formation mechanisms. *Particle effects* occur between the camera and the scene, including fog scattering and rain or snow occlusions. *Lens effects* corrupt the imaging system itself, including cover-glass droplets and flare. *Background effects* alter the observed scene, such as snow-covered road markings, wet asphalt, and puddle reflections. This thesis focuses on three common adverse weather conditions in road environments: rain, fog, and snow. I next describe the visual degradations associated with each condition.

Rain. Rain introduces several visual artefacts. Rain streaks are particle effects: elongated traces produced by falling droplets during the camera exposure window. Their appearance depends on droplet size, fall velocity, exposure time, wind, and camera motion (Kang et al., 2012; Yang et al., 2020a). Droplet occlusions are lens effects caused by water adhering to the camera lens. They produce local blur, distortion, and occlusion that persist across consecutive frames from the same camera but are absent from other views of the scene, making them particularly harmful for multi-view reconstruction (Qian et al., 2018). Wet-surface reflections are background effects. They arise when roads, vehicles, or other surfaces become reflective under rain, producing scene-level appearance changes governed by the optical properties of wet materials. Such reflections can be misinterpreted by perception models as physical structures or object boundaries. Fig. 1.3 illustrates these three rain-induced artefacts.

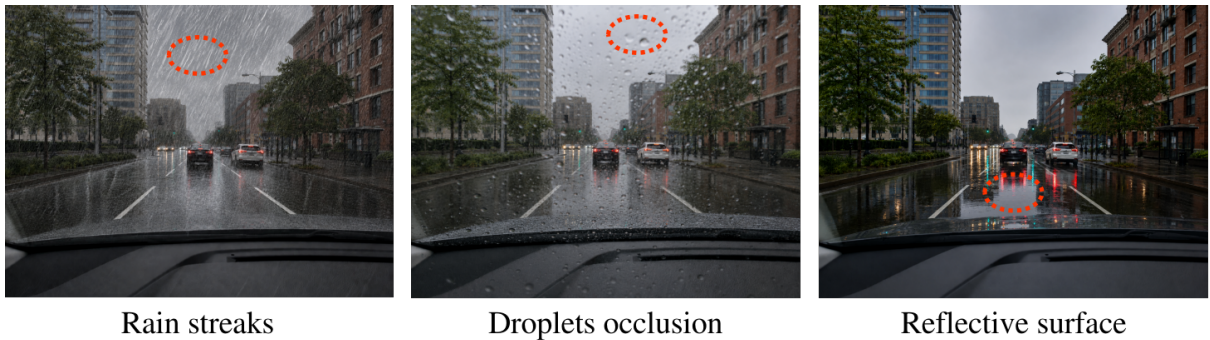


Figure 1.3: Principal visual artefacts of rain in camera images. Illustration generated by the author.

Snow. Snow also spans the three effect categories. Airborne snowflakes produce particle effects. They appear as local occluders whose image footprint depends on particle size, depth, velocity, wind, and exposure time (Liu et al., 2018; Zhang et al., 2021a). Adhered snowflakes on the cover glass are lens effects, analogous to lens raindrops. They remain fixed or slowly varying in the camera frame and are typically inconsistent across different cameras. Accumulated snow is a background effect. Once deposited on the road surface, pavements, vehicles, or roadside structures, snow becomes part of the persistent scene appearance and geometry. It can obscure road markings, kerbs, lane boundaries, and surface texture, thereby removing the visual cues on which dense prediction models depend. Unlike airborne particles, accumulated snow should not be treated as a transient occluder; it is a scene-level change that must be represented or synthesised as part of the environment. Fig. 1.4 illustrates the three snow-induced artefacts.



Figure 1.4: Principal visual artefacts of snow in camera images. Illustration generated by the author.

Fog. Fog is commonly described by the atmospheric scattering model (Berman et al., 2016; Sakaridis et al., 2018):

$$I(\mathbf{x}) = J(\mathbf{x}) \cdot t(\mathbf{x}) + A \cdot (1 - t(\mathbf{x})), \quad (1.1)$$

where $J(\mathbf{x})$ denotes the scene radiance at pixel $\mathbf{x}$, A is the global airlight, and $t(\mathbf{x}) = e^{-\beta d(\mathbf{x})}$ is the transmission map determined by the scattering coefficient β and scene depth $d(\mathbf{x})$. This model captures two coupled effects: fog attenuates direct scene radiance by an amount that increases with depth, and it adds a distance-dependent airlight component that shifts distant regions towards the atmospheric colour. The depth dependence of $t(\mathbf{x})$ makes fog a spatially structured degradation rather than a purely photometric one. Consequently, a global contrast or colour correction cannot fully recover fog-corrupted content, because the corruption varies with 3D scene geometry. Fig. 1.5 shows the same scene under three scattering coefficients, making the depth-dependent corruption visible: distant content fades into the airlight first, and dense fog (large β) collapses the entire depth range into a low-contrast veil.



Figure 1.5: A driving scene under three fog densities, parameterised by the scattering coefficient β in Eq. (1.1). Illustration generated by the author.

1.2.2.2 Weather Challenges

At the perception level, adverse weather induces a pronounced robustness gap. Models trained on standard clear weather driving datasets (Cordts et al., 2016; Geiger et al., 2012; Sun et al., 2020) often suffer performance drop when evaluated on adverse weather benchmarks such as ACDC (Sakaridis et al., 2021). The degradation is most severe for small and safety-critical classes, including pedestrians, cyclists, lane markings, and traffic signs, whose evidence is easily erased by contrast loss, occlusion, scattering, and boundary corruption (Hahner et al., 2019; Zang et al., 2019). Fig. 1.6 illustrates example failure modes on representative rainy and snowy ACDC frames, where a Cityscapes-trained SegFormer (Xie et al., 2021) mislabels a wall as vegetation under rain and merges snow-covered sidewalks into the road class. Thus, weather conditions make visual perception becomes less reliable and driving dangerous. I identify two core bottlenecks for addressing this problem.

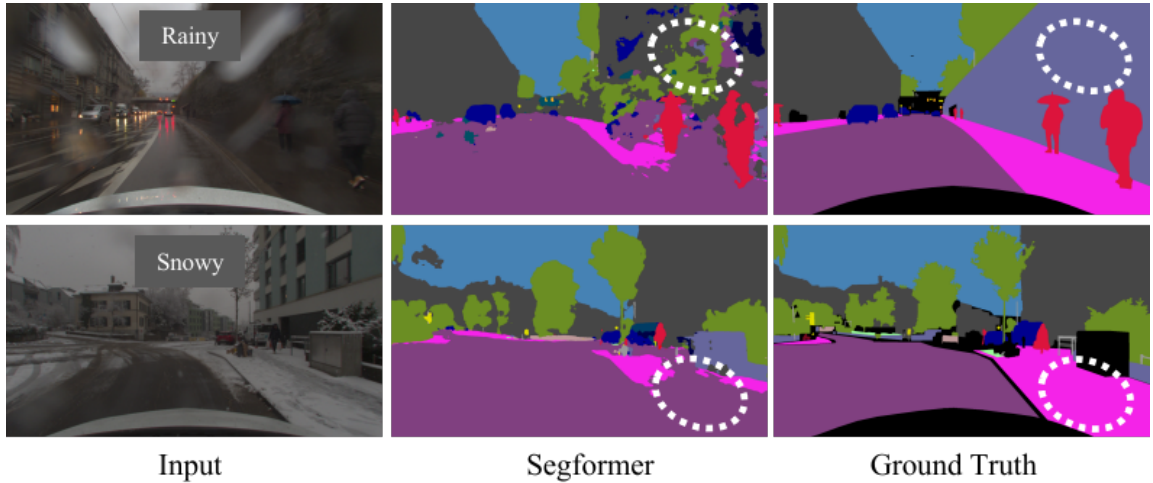


Figure 1.6: Representative weather-induced perception failures on ACDC. Images and ground-truth annotations are from ACDC (Sakaridis et al., 2021); the SegFormer predictions were produced by the author using Xie et al. (2021).

Visual degradation. Adverse weather is not a single image corruption but a heterogeneous family of physical effects. As discussed before, rain, fog, and snow affect images

through different combinations of lens, particle, and background phenomena. This heterogeneity limits the transferability of condition-specific restoration methods. A deraining model designed for streak removal does not address wet-road reflections; a dehazing model based on atmospheric transmission does not recover snow-covered road texture; and a desnowing model trained on airborne flakes does not resolve adhered snow or accumulated snow. This heterogeneity suggests that adverse weather robustness cannot be reduced to a single restoration problem. Different artefacts impose different constraints on what can be inferred from the image: some corrupt local appearance, some attenuate visibility in a depth-dependent manner, and others alter the persistent scene itself. A method that treats all of them as the same low-level corruption is therefore unlikely to generalise across weather types. This motivates modelling strategies that combine image-level generative priors with spatially grounded representations, so that weather effects can be enhanced, synthesised, or represented according to their physical role in the scene.

Data scarcity. The data-driven learning of perception models is constrained by an equally severe data bottleneck. Existing adverse weather datasets are much smaller than the clear weather corpora on which modern perception systems are typically trained. As shown in Table 1.1, adverse-weather driving datasets remain much smaller than large-scale benchmarks collected mainly under normal or common driving conditions. For example, ACDC provides 4,006 annotated adverse-condition images (Sakaridis et al., 2021), and DAWN contains 1,000 images (Kenk and Hassaballah, 2020). In contrast, large-scale general-purpose driving datasets are substantially larger: nuScenes provides 1.4 million camera images (Caesar et al., 2020), while the Waymo Open Dataset provides 1,150 annotated 20-second driving scenes (Sun et al., 2020). The imbalance is not merely a matter of scale. Dense annotation under adverse weather is expensive and error-prone, while the most safety-critical cases, such as pedestrians in heavy rain, occluded lane markings under snow, or dense fog at intersections, lie in the long tail of real-world data collection.

This creates a fundamental mismatch between training and deployment. The deployment domain contains rare, diverse, and physically complex weather conditions, but the labelled training distribution covers only a sparse subset of them. Collecting more real-world data is a straightforward remedy, but it is difficult to scale in practice. Acquiring data under severe adverse weather is itself risky, since the same conditions that degrade perception also increase the danger of on-road data collection. Moreover, extreme weather events are intermittent and geographically uneven: heavy snow, dense fog, or torrential rain may occur only rarely within a given collection campaign. As a result, the most safety-critical conditions are also the least controllable and least frequently observed in real-world datasets.

These two bottlenecks are coupled. Weather produces structured, non-uniform, and sometimes irreversible visual degradation, while the data needed to learn robust responses

Table 1.1: Scale of publicly available autonomous driving datasets with camera imagery, grouped by collection environment.

Dataset	Environment	Scale
Waymo Open (Sun et al., 2020)	Predominantly clear	1,150 20-second driving scenes
KITTI (Geiger et al., 2012)	Predominantly clear	$\sim$ 15,000 stereo frames with LiDAR and GPS
nuScenes (Caesar et al., 2020)	Predominantly clear	1,000 scenes, 1.4M images
BDD100K (Yu et al., 2020)	Predominantly clear	>100k videos and images
Cityscapes (Cordts et al., 2016)	Predominantly clear	5,000 fine + 20,000 coarse annotations
ACDC (Sakaridis et al., 2021)	Rain, fog, snow, night	4,006 images
DAWN (Kenk and Hassaballah, 2020)	Rain, fog, snow, sandstorm	1,000 images

to these effects are scarce precisely in the regimes that matter most for safety. Each existing strategy addresses only part of the problem. Enhancing images at inference time can make degraded inputs look cleaner, but it does not expose the model during training to the full diversity of adverse weather it may encounter at deployment. Simply collecting more data is also insufficient, because real-world weather varies continuously in intensity, appearance, location, illumination, and temporal dynamics, making it difficult to cover all rare but safety-critical cases. This thesis therefore argues for a combined generative and spatial modelling strategy: generative models expand and restore visual evidence, while spatial models provide the geometric structure required to represent, control, and render weather effects consistently.

1.3 Approaches to Weather Perception

Adverse weather perception has been studied along three main directions. Visual restoration improves the degraded image before it reaches the perception model. Domain adaptation and generalisation improve the robustness of the perception model itself. Data augmentation expands the training distribution by synthesising adverse weather observations. I briefly review these directions below and then identify the remaining gaps.

Visual restoration. Visual restoration aims to recover a clean or more perceptually useful image from a weather-degraded observation. Single-condition methods remain strong within their target setting, including deraining (Yang et al., 2020a; Wang et al., 2022), dehazing (Berman et al., 2016; Zhang et al., 2019b), desnowing (Liu et al., 2018;

Zhang et al., 2021a; Wang et al., 2023b), and low-light enhancement (Jiang et al., 2021; Guo et al., 2020). Their limitation is transferability: a model optimised for one degradation often fails when the weather effect changes or when multiple effects co-occur.

All-in-one restoration addresses this limitation by using a shared backbone together with degradation-aware routing, prompting, or conditional modulation, trained jointly across multiple weather conditions (Jose Valanarasu et al., 2022; Zhu et al., 2023; Chen et al., 2022b; Sun et al., 2024; Guo et al., 2024a; Cui et al., 2024a; Zamfir et al., 2025). More recently, diffusion models have become increasingly prominent in restoration pipelines (Yang et al., 2024a; Chen et al., 2024a; Liu et al., 2024b), improving perceptual quality and per-task reconstruction metrics on standard benchmarks. However, these methods remain primarily image-level operators. They are effective at suppressing local artefacts such as streaks, haze, or flakes, but are less reliable for the broader appearance shifts that matter for downstream perception, including contrast collapse, colour bias, view-dependent occlusion, and large-region texture loss (Jiang et al., 2025). Moreover, because they operate independently on 2D images, they do not explicitly preserve multi-view geometry or recover weather-degraded 3D scene structure.

Domain adaptation and generalisation. A second line of work targets the perception model rather than the input image. Instead of restoring weather-degraded images, these methods try to make the model itself more robust to the gap between clear and adverse conditions. Unsupervised domain adaptation assumes that labelled data are available in a clear-weather source domain, such as Cityscapes (Cordts et al., 2016), while images from the target adverse-weather domain are available without labels. The model is then trained to improve its predictions on these unlabelled target images, commonly by using its own confident predictions as pseudo-labels and refining them over time (Hoyer et al., 2022, 2024, 2023). Domain generalisation considers an even more restrictive setting: the model does not see target-weather images during training, and must learn from the source data alone in a way that remains effective under unseen weather conditions (Pan et al., 2018; Choi et al., 2021; Zhao et al., 2022).

Data augmentation. A third direction addresses the data bottleneck by synthesising adverse weather imagery. Graphics simulators, such as CARLA (Dosovitskiy et al., 2017), can render controllable weather conditions at scale, but they often suffer from a persistent sim-to-real gap in appearance, sensor characteristics, and scene diversity. Diffusion-based augmentation mitigates this limitation by leveraging large-scale natural-image priors (Niemeijer et al., 2024; Benigmim et al., 2024; Azuma et al., 2024; Jia et al., 2024). By producing more realistic adverse weather variants of driving scenes, these methods can improve the domain generalisation of segmentation models.

1.4 Thesis Contributions

This thesis makes four methodological contributions, each previously published in a peer-reviewed venue.

1. **Unified image enhancement for adverse weather** (Chapter 4, AllWeatherNet (Qian et al., 2024)). A unified framework for enhancing image quality under various adverse conditions such as snow, rain and fog.
2. **Clean 3D reconstruction from weather-degraded multi-view sequences** (Chapter 5, WeatherGS (Qian et al., 2025a)). A method for removing atmospheric particles and lens occlusions, enabling clean 3D scene reconstruction from degraded multi-view weather images.
3. **Synthetic adverse weather training data from domain-adapted generative models** (Chapter 6, WeatherDG (Qian et al., 2025b)). A weather generator that generates photorealistic, dashcam-consistent, class-balanced weather imagery without any real adverse weather image as references.
4. **Controllable weather synthesis in 3D reconstructed scenes** (Chapter 7, WeatherEdit (Qian et al., 2026)). A weather editing pipeline for generating both static and dynamic weather effects with controllable types and severity in 3D scenes, enabling user-specified weather types and severity levels while preserving multi-view consistency.

1.5 Thesis Organisation

Part I comprises Chapters 1–3, which introduce the motivation, review the literature, and establish the research gaps and research questions. The thesis proceeds as follows.

Chapter 2 reviews generative models (variational autoencoders, generative adversarial networks, diffusion models, conditional and controllable generation) and spatial models (Neural Radiance Fields, 3D Gaussian Splatting), closing with the applied literature on weather restoration and weather generation.

The four technical chapters form two parts grouped by capability. **Part II (Enhancement)**, covered in Chapters 4 and 5, develops AllWeatherNet at the 2D image level and WeatherGS at the 3D scene level. **Part III (Generation)**, covered in Chapters 6 and 7, develops WeatherDG at the 2D level and WeatherEdit at the 3D level.

Chapter 8 revisits the four methods as a single body of work, distils the cross-cutting themes that emerge only at the system level, assesses the framework’s limitations, and outlines directions for future research.

Chapter 2

Literature Review

This thesis builds on two closely related bodies of work: generative models, which learn the underlying distribution of natural images and can be conditioned to sample from targeted visual distributions; and spatial models, which represent scenes through explicit or implicit 3D geometry. Together, these modelling paradigms provide the foundations for addressing adverse weather perception in autonomous driving. Generative models offer the ability to restore degraded observations and synthesise missing weather-affected data, while spatial models provide the geometric structure required for multi-view consistency and controllable rendering.

In this chapter, I review these foundations and their applications to adverse weather autonomous driving. Section 2.1 introduces generative models, including variational autoencoders, generative adversarial networks, diffusion models, and the conditional or controllable mechanisms that turn learned priors into steerable generators. Section 2.2 reviews spatial models, focusing on Neural Radiance Fields and 3D Gaussian Splatting as two dominant representations for 3D scene modelling. Building on these foundations, Section 2.3 surveys the applied literature on weather restoration, where the goal is to recover usable visual evidence from degraded observations. Section 2.4 then reviews the literature on weather generation, where the goal is to synthesise adverse weather observations for data augmentation.

2.1 Generative Models

A generative model learns a probabilistic mechanism, or sampling procedure, that can produce new samples consistent with a data distribution $p_{\text{data}}(\mathbf{x})$. I introduce three foundations relevant to this thesis: variational autoencoders, generative adversarial networks, and diffusion models.

2.1.1 Variational Autoencoders

The variational autoencoder (VAE) (Kingma, 2013) is a latent-variable generative model that learns both an encoder and a decoder. The encoder $q_\phi(\mathbf{z} \mid \mathbf{x})$ maps a data point $\mathbf{x}$ to an approximate posterior distribution over a latent variable $\mathbf{z}$, while the decoder $p_\theta(\mathbf{x} \mid \mathbf{z})$ maps a latent sample back to a distribution over the data space (Fig. 2.1). Training maximises the evidence lower bound (ELBO):

$$\mathcal{L}_{\text{VAE}} = \mathbb{E}_{q_\phi(\mathbf{z} \mid \mathbf{x})} [\log p_\theta(\mathbf{x} \mid \mathbf{z})] - D_{\text{KL}}(q_\phi(\mathbf{z} \mid \mathbf{x}) \parallel p(\mathbf{z})), \quad (2.1)$$

where the first term encourages accurate reconstruction and the second regularises the approximate posterior towards a simple prior, typically a standard Gaussian. Sampling is made differentiable through the reparameterisation trick, $\mathbf{z} = \boldsymbol{\mu} + \boldsymbol{\sigma} \odot \boldsymbol{\epsilon}$, where $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ and $\odot$ denotes element-wise multiplication. This formulation allows the stochastic encoder–decoder pipeline to be trained end-to-end.

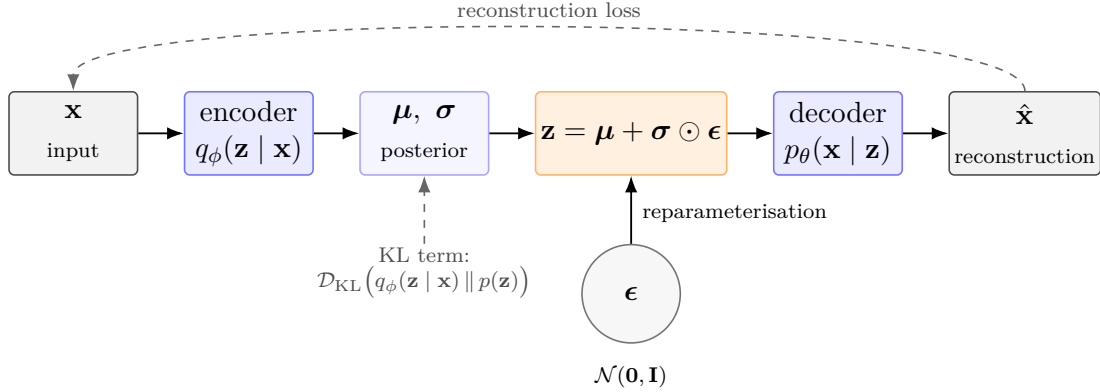


Figure 2.1: The VAE pipeline.

The VAE framework has been extended in several influential directions. β -VAE (Higgins et al., 2017) re-weights the KL term by a coefficient β , encouraging more factorised latent representations at the cost of reconstruction fidelity. Vector-Quantised VAE (VQ-VAE) (van den Oord et al., 2017) replaces the continuous Gaussian posterior with a discrete codebook, enabling generation over quantised latent tokens and often producing sharper reconstructions than standard VAEs. Hierarchical variants such as NVAE (Vahdat and Kautz, 2020) introduce multiple latent stages to capture coarse-to-fine structure and substantially improve image quality on standard benchmarks.

Two properties make VAEs relevant to this thesis. First, they provide a structured latent space that can be sampled from, interpolated within, and conditioned on. This property makes VAEs useful as compact representations for downstream generative modelling. Second, standalone VAEs trained with reconstruction-based objectives often produce smooth or blurry samples, particularly when multiple plausible outputs correspond to the same latent region. This limitation motivates the adversarial and diffusion-based generative models reviewed in the following subsections.

In this thesis, VAE functions as an enabling component of latent diffusion models (Rombach et al., 2021). Many contemporary text-to-image systems, including those used in Chapters 6 and 7, use a VAE encoder–decoder pair to map images into a compressed latent space where the diffusion process is performed. Operating in this latent space makes high-resolution diffusion substantially more efficient than pixel-space denoising, while the decoder maps the generated latent back to image space. The reconstruction fidelity of the VAE therefore constrains the visual detail that the diffusion model can ultimately express. Thus, VAEs are retained in this review not as a standalone solution to adverse weather perception, but as the compression mechanism that makes later diffusion-based methods computationally practical.

2.1.2 Generative Adversarial Networks

Generative adversarial networks (GANs) (Goodfellow et al., 2014) formulate generation as a two-player adversarial game between a generator G and a discriminator D (Fig. 2.2). The generator maps a latent vector $\mathbf{z} \sim p(\mathbf{z})$ to a synthetic sample $G(\mathbf{z})$, while the discriminator is trained to distinguish generated samples from real data. The standard objective is

$$\min_G \max_D \mathbb{E}_{\mathbf{x} \sim p_{\text{data}}} [\log D(\mathbf{x})] + \mathbb{E}_{\mathbf{z} \sim p(\mathbf{z})} [\log(1 - D(G(\mathbf{z})))] . \quad (2.2)$$

Under idealised assumptions, the generator distribution matches the data distribution at equilibrium, and the discriminator cannot distinguish real from generated samples better than chance. In practice, this equilibrium is difficult to reach. GANs are well known for training instability, sensitivity to hyperparameters, and mode collapse, where the generator produces a limited set of plausible samples at the cost of distributional coverage (Creswell et al., 2018).

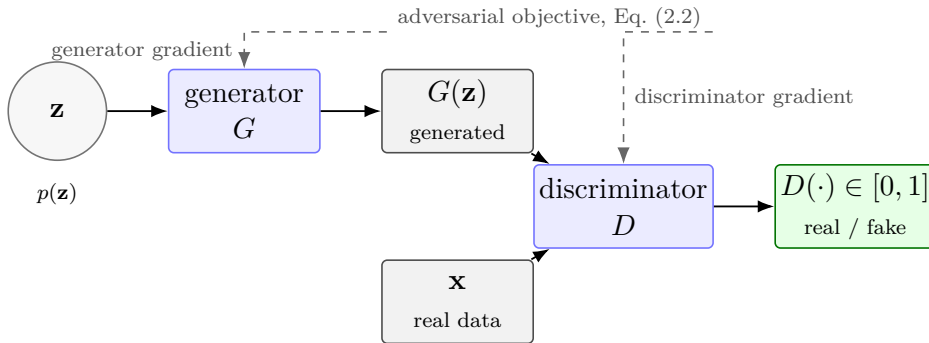


Figure 2.2: The GAN training pipeline.

Early progress in GANs was driven by architectural improvements that made adversarial generation more stable and visually expressive. DCGAN (Radford et al., 2016) established convolutional generator and discriminator architectures as the dominant design for image synthesis. Progressive GAN (Karras et al., 2018) introduced a curriculum that grows image resolution during training, improving stability for high-resolution synthesis. StyleGAN (Karras et al., 2019) and its successors StyleGAN2 (Karras et al., 2020)

and StyleGAN3 (Karras et al., 2021) introduced a style-based generator with per-layer modulation, enabling fine-grained control over visual attributes at multiple scales. BigGAN (Brock et al., 2019) further showed that large-scale class-conditional GANs can achieve high-fidelity synthesis when model capacity, batch size, and regularisation are carefully scaled.

A parallel line of work addressed the instability of adversarial optimisation. Wasserstein GAN (WGAN) (Arjovsky et al., 2017) replaces the Jensen–Shannon divergence implicit in the original GAN formulation with the Wasserstein distance, providing more informative gradients when the real and generated distributions have limited overlap. WGAN-GP (Gulrajani et al., 2017) adds a gradient penalty to enforce the Lipschitz constraint more smoothly, while spectral normalisation (Miyato et al., 2018) constrains the discriminator by normalising each weight matrix by its largest singular value. Least-squares GAN (LSGAN) (Mao et al., 2017) replaces the binary cross-entropy objective with a squared-error loss, yielding smoother gradients near the discriminator decision boundary. This stabilisation is particularly relevant for image enhancement, where the discriminator should encourage local realism without driving the generator towards unsupported hallucination.

For this thesis, the most relevant branch of GAN research is conditional image-to-image translation. Conditional GANs (Mirza and Osindero, 2014; Isola et al., 2017) inject additional information, such as class labels, source images, or semantic maps, into the generation process. In image-to-image translation, the source image acts as an anchor: the adversarial loss pushes the output towards the target domain, while reconstruction, perceptual, or task losses preserve correspondence with the input. Pix2Pix (Isola et al., 2017) established the paired translation setting with a U-Net generator and a PatchGAN discriminator, which classifies overlapping local patches and is therefore sensitive to texture and local realism. Pix2PixHD (Wang et al., 2018) extended this framework to high-resolution synthesis through a coarse-to-fine generator, while SPADE (Park et al., 2019) introduced spatially adaptive normalisation to preserve semantic-map structure during generation.

Unpaired image-to-image translation removes the need for aligned source–target pairs. CycleGAN (Zhu et al., 2017) introduced cycle consistency, enforcing that a source image translated to the target domain should be recoverable when translated back to the source domain. This formulation is attractive for weather translation, where paired clear and adverse weather images are rarely available. However, cycle consistency is a global constraint rather than a pixel-faithful guarantee; a model can satisfy the cycle objective while altering or hallucinating local structures. MUNIT (Huang et al., 2018) and DRIT (Lee et al., 2018) disentangle content and style to support diverse translation outputs. UNIT (Liu et al., 2017) assumes a shared latent space between domains, while CUT (Park et al., 2020) replaces cycle consistency with a contrastive objective that encourages correspondence between source and generated patches. Task-aware variants such

as ForkGAN (Zheng et al., 2020) further couple translation with downstream perception objectives, encouraging generated images to be not only visually plausible but also useful for segmentation or detection. Santa (Xie et al., 2023) formulates unpaired translation as a shortest-path transformation in a learned metric space, providing an alternative to editing based on cycle consistency.

Several GAN design principles are directly relevant to adverse weather restoration. Attention-based GANs (Qian et al., 2018; Zhang et al., 2019a) allow the generator to allocate capacity to spatially localised corruptions, such as rain streaks, lens droplets, or low-light noise. Multi-scale discriminators (Jiang et al., 2021) supervise realism at different spatial granularities: coarse discriminators encourage global tone and illumination consistency, while finer discriminators penalise texture loss and local artefacts. These ideas inform the AllWeatherNet architecture, which combines illumination-aware attention with hierarchical adversarial supervision at scene, object, and texture scales.

Despite continued progress, GANs have been partly displaced by diffusion models as the default backbone for high-capacity image generation. Diffusion models offer more stable training, stronger distributional coverage, and more natural integration with text and spatial conditioning. GANs nevertheless remain useful in constrained image-to-image settings. In AllWeatherNet, the objective is not open-ended image synthesis but residual enhancement of an existing weather-degraded input. This is a regime in which adversarial supervision is efficient: it can sharpen local detail and improve perceptual realism without requiring a large text-conditioned generative prior.

By contrast, the generation-oriented chapters of this thesis require diversity, semantic controllability, and prompt-level steering over weather type, scene composition, and severity. These requirements are less naturally served by conventional GANs and motivate the move towards diffusion and language-guided pipelines in WeatherDG and WeatherEdit. Recent GAN–diffusion hybrids further suggest that the two paradigms are complementary rather than mutually exclusive. CycleGAN-Turbo (Parmar et al., 2024), for example, adapts a single-step diffusion model to unpaired image translation through adversarial learning and serves as the image-translation backbone used in WeatherEdit.

2.1.3 Diffusion Models

Denoising diffusion probabilistic models (DDPMs) (Ho et al., 2020) replace adversarial training with a stable supervised denoising objective. The forward process gradually corrupts a clean data sample $\mathbf{x}_0$ by adding Gaussian noise over T timesteps:

$$q(\mathbf{x}_t \mid \mathbf{x}_{t-1}) = \mathcal{N}\left(\mathbf{x}_t; \sqrt{1 - \beta_t} \mathbf{x}_{t-1}, \beta_t \mathbf{I}\right), \quad (2.3)$$

where $\{\beta_t\}_{t=1}^T$ is a fixed noise schedule. Defining $\alpha_t = 1 - \beta_t$ and $\bar{\alpha}_t = \prod_{s=1}^t \alpha_s$, the noisy sample at any timestep can be drawn directly from

$$q(\mathbf{x}_t \mid \mathbf{x}_0) = \mathcal{N}(\mathbf{x}_t; \sqrt{\bar{\alpha}_t} \mathbf{x}_0, (1 - \bar{\alpha}_t) \mathbf{I}). \quad (2.4)$$

Fig. 2.3 visualises this forward trajectory on a representative driving frame: local scene structure remains visible at early timesteps and progressively dissolves into isotropic Gaussian noise as $t \rightarrow T$. Generation then learns to reverse this process. A neural network $\epsilon_\theta(\mathbf{x}_t, t)$ is trained to predict the noise component added at each timestep, allowing samples to be generated by starting from Gaussian noise and iteratively denoising.

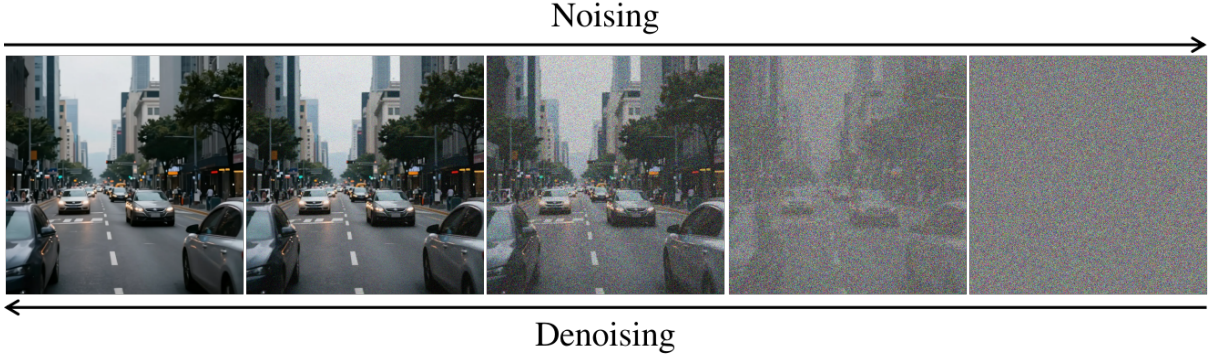


Figure 2.3: The diffusion forward process applied to a driving frame. The driving frame and visualisation were generated by the author using the forward noising formulation of Ho et al. (2020).

The commonly used simplified training objective is

$$\mathcal{L}_{\text{diff}} = \mathbb{E}_{t, \mathbf{x}_0, \epsilon} \left[\left\| \epsilon - \epsilon_\theta \left(\sqrt{\bar{\alpha}_t} \mathbf{x}_0 + \sqrt{1 - \bar{\alpha}_t} \epsilon, t \right) \right\|^2 \right], \quad (2.5)$$

where $\epsilon \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$. This objective trains the model to estimate the injected noise from a noisy sample and its timestep. At inference, a sample is drawn by initialising $\mathbf{x}_T$ from Gaussian noise and applying the learned reverse denoising process step by step.

Theoretical foundations. The denoising formulation has a principled connection to score matching and stochastic differential equations (SDEs). Song *et al.* (Song et al., 2021) showed that diffusion processes can be described by score-based SDEs, where the score function $\nabla_{\mathbf{x}} \log p_t(\mathbf{x})$ is estimated by a neural network. This view unifies DDPMs with noise-conditional score networks (Song and Ermon, 2019) and provides a continuous-time framework for analysing sampling, likelihood estimation, and the design of reverse-time solvers. (Karras et al., 2022) further distilled these theoretical insights into practical design choices, including noise schedules, preconditioning, and loss weighting, which have become standard components of high-quality diffusion training and sampling.

Diffusion models surpassed GANs on several image-generation benchmarks (Dhariwal and Nichol, 2021), and subsequent developments made them practical for large-scale

image synthesis. Denoising diffusion implicit models (DDIMs) (Song et al., 2020) introduced a deterministic non-Markovian sampling procedure that substantially reduces the number of denoising steps, making inference faster and more controllable. Latent diffusion models (LDMs) (Rombach et al., 2021) moved the diffusion process from pixel space to a VAE-compressed latent space, greatly reducing computational cost while enabling high-resolution synthesis. This latent formulation underpins many contemporary text-to-image systems, including Stable Diffusion, and is the basis of the diffusion models used in the later chapters of this thesis.

The discussion above has focused on how diffusion models generate high-quality images efficiently. However, the applications in this thesis require not only realistic generation, but also control over what is generated and how it aligns with the input driving scene. This motivates the next section, which reviews the conditioning mechanisms that make diffusion models useful for controllable weather generation and editing.

Video and 3D diffusion. Extending diffusion from images to temporal and 3D domains remains an active frontier. Video diffusion models (Ho et al., 2022; Blattmann et al., 2023b; Singer et al., 2023) extend image diffusion architectures with temporal attention or related temporal modules to improve frame-to-frame consistency. Large-scale systems such as SVD (Blattmann et al., 2023a) and CogVideoX (Yang et al., 2024c) demonstrate the ability of diffusion models to synthesise short videos with coherent motion and appearance.

In 3D-aware generation, diffusion priors are often used to guide optimisation or reconstruction rather than directly generating a complete 3D scene. DreamFusion (Poole et al., 2023) introduced score distillation sampling to lift 2D text-to-image diffusion priors into 3D NeRF representations. Subsequent work has adapted diffusion priors to 3D Gaussian representations, either through score-distillation optimisation or through feed-forward Gaussian generators such as LGM (Tang et al., 2024), which predicts high-resolution 3D Gaussian assets from text or single-view image inputs. ReconFusion (Wu et al., 2024b) conditions a diffusion model on sparse-view captures to synthesise novel viewpoints, bridging image generation and 3D reconstruction.

This intersection between diffusion and 3D representation is directly relevant to the thesis. WeatherEdit uses diffusion not as a full 3D generative model, but as an image-level prior for synthesising weather-affected appearance within reconstructed 3D scenes. The spatial consistency is supplied by the underlying 3D representation, while diffusion provides realistic weather appearance and controllable editing capability.

Diffusion for restoration. Diffusion priors can also be used for image restoration. Given a degraded observation, the reverse process can be conditioned on the available measurements, while the learned image prior infers plausible missing or corrupted content. Palette (Saharia et al., 2022) is an early example of this paradigm, applying diffusion to

general image-to-image restoration tasks. DDNM (Wang et al., 2023c) incorporates null-space projection to encourage consistency with observations under known degradation operators.

Weather-specific restoration methods extend this idea to adverse visual conditions. The diffusion test-time adaptation method of Yang *et al.* (Yang et al., 2024a) applies diffusion priors to video adverse-weather removal, while the plugin-based method of Liu *et al.* (Liu et al., 2024b) selects task-specific restoration adapters using CLIP-based similarity between the input image and textual descriptions of degradation types. The Atmospheric Effect Filter in WeatherGS builds on this plugin-based formulation by using diffusion-based restoration to suppress atmospheric degradation before geometry reconstruction.

Diffusion models therefore provide a strong learned prior over plausible street scenes and weather appearances. They do not, however, specify how that prior should be used for autonomous driving perception. They do not decide which weather conditions should be sampled, which semantic classes should be preserved, which camera geometry must remain fixed, or how generated effects should remain consistent across views. Much of the methodological contribution of the diffusion-based methods developed in this thesis lies in designing these task-specific protocols around existing diffusion priors, so that their outputs become useful for weather-robust perception rather than merely visually convincing images. The next subsection reviews the conditioning and control mechanisms that make such protocols possible.

2.1.4 Conditional and Controllable Generation

Unconditional generative models learn to sample from a data distribution, but most practical applications require more than unconstrained sampling. The generated output often needs to satisfy external requirements: a text description, a spatial layout, a reference image, a target domain, or an input image to be edited. Conditional and controllable generation addresses this problem by introducing mechanisms that steer the sampling process while preserving the realism learned by the generative prior.

Text conditioning allows a diffusion model to steer generation using a natural-language prompt. A common implementation is classifier-free guidance (Ho and Salimans, 2021), which compares the model’s prediction with and without the text prompt during sampling and strengthens the direction that better matches the prompt. Spatial conditioning (including ControlNet (Zhang et al., 2023b), T2I-Adapter (Mou et al., 2024), GLIGEN (Li et al., 2023b), Composer (Huang et al., 2023), and IP-Adapter (Ye et al., 2023)) anchors layout, depth, semantic maps, bounding boxes, or reference images, supplying the scene structure that text alone cannot enforce. Subject and style adaptation (Dream-Booth (Ruiz et al., 2023), Textual Inversion (Gal et al., 2023), LoRA (Hu et al., 2022), and related methods (Yeh et al., 2023)) specialises a frozen base model to a domain-specific

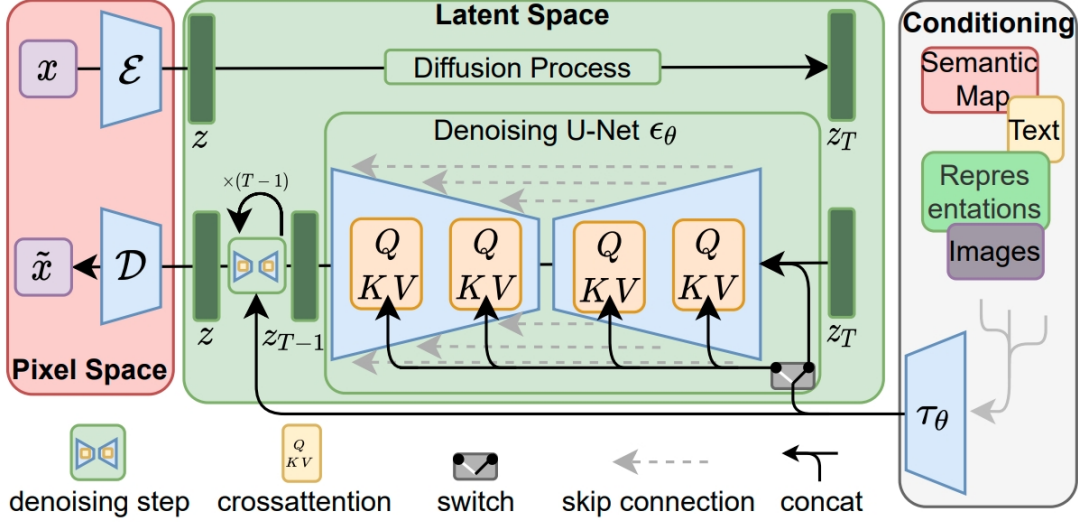


Figure 2.4: The latent diffusion architecture (Rombach et al., 2021). Reproduced from the Open Access version of the paper at <https://openaccess.thecvf.com/CVPR2022>.

visual identity through lightweight fine-tuning. Image-anchored editing (SDEdit (Meng et al., 2021), Prompt-to-Prompt (Hertz et al., 2023), Null-text Inversion (Mokady et al., 2023), InstructPix2Pix (Brooks et al., 2023), and TurboEdit (Deutch et al., 2024)) keeps the output tied to an input image while modifying specified attributes. Language-driven prompt control (Pratt et al., 2023; Benigimim et al., 2024) uses a language model to expand compact task specifications into structured prompts that downstream diffusion models can render.

Architecturally, many conditioning signals enter the denoising network through cross-attention, while spatial adapters such as ControlNet and T2I-Adapter additionally inject condition-dependent feature maps into the network. Fig. 2.4 reproduces the canonical latent diffusion architecture (Rombach et al., 2021): a VAE encoder–decoder pair maps images to and from a compressed latent space, while a conditioning encoder τ_θ projects heterogeneous signals (text, semantic maps, learned representations, or reference images) into tokens consumed by the denoising U-Net. This design highlights that conditioning is not an external post-processing step, but part of the generative process itself. By integrating control signals through attention and feature-injection pathways, a single diffusion backbone can support multiple control axes simultaneously.

Fig. 2.5 illustrates this principle with ControlNet (Zhang et al., 2023b). A Canny edge map constrains the object silhouette while text prompts alter appearance, lighting, and style; similarly, a human-pose skeleton fixes body configuration while the prompt changes identity and context. In both cases, spatial conditioning constrains layout, whereas text conditioning steers appearance. Taken together, these axes give the same generative prior multiple complementary control surfaces: text defines the semantic target, spatial controls preserve geometry, domain or style adaptation aligns the generator with driving imagery, and image anchors constrain edits to remain faithful to an input observation.



Figure 2.5: ControlNet (Zhang et al., 2023b) outputs under two spatial conditions. Reproduced from the Open Access version of the paper at <https://openaccess.thecvf.com/ICCV2023>.

2.2 Spatial Models

Generative models provide powerful priors for restoring and synthesising weather-affected images, yet they often lack an explicit representation of the spatial structure that governs weather phenomena in driving scenes. Fog varies with scene depth, particles move through the camera frustum, and accumulated effects such as snow or wet road surfaces remain consistent across viewpoints. Spatial modelling addresses this limitation by representing the 3D structure of the scene, thereby enabling geometry-aware reconstruction, multi-view consistency, and controllable weather rendering.

At the 3D level, two representations dominate the field: Neural Radiance Fields (Section 2.2.1) and 3D Gaussian Splatting (Section 2.2.2). NeRF represents a scene as a continuous implicit radiance field queried along camera rays, while 3D Gaussian Splatting represents it as a set of explicit anisotropic Gaussian primitives that can be rendered efficiently through differentiable rasterisation. I review both paradigms in turn and highlight the properties that make 3D Gaussian Splatting the foundation of the 3D contributions in this thesis.

2.2.1 3D Spatial Representation: Neural Radiance Fields

Neural Radiance Fields (NeRF) (Mildenhall et al., 2020) represent a scene as a continuous 5D radiance function

$$(\sigma, \mathbf{c}) = F_{\Theta}(\mathbf{x}, \mathbf{d}), \quad (2.6)$$

where $\mathbf{x} \in \mathbb{R}^3$ is a 3D spatial location, $\mathbf{d} \in \mathbb{R}^2$ a viewing direction, σ volumetric density, and $\mathbf{c}$ view-dependent emitted colour. The function is parameterised as a multi-layer

perceptron (MLP) and trained by differentiable volume rendering along rays cast from known camera poses. The key modelling choice is that geometry and appearance are implicit in the same function: high density along a ray indicates occupied space, while colour depends both on position and viewing direction. The pixel colour rendered through a ray $\mathbf{r}$ is

$$\hat{C}(\mathbf{r}) = \sum_{i=1}^N T_i \alpha_i \mathbf{c}_i, \quad T_i = \prod_{j=1}^{i-1} (1 - \alpha_j), \quad \alpha_i = 1 - e^{-\sigma_i \delta_i}, \quad (2.7)$$

with δ_i the spacing between adjacent samples along the ray. Training minimises the discrepancy between rendered and observed pixels over many posed images. Fig. 2.6 traces the four stages of this pipeline: ray casting and 5D sampling, MLP evaluation, volume rendering, and the photometric loss against ground truth. This makes NeRF powerful for clean multi-view reconstruction, but also explains its vulnerability to weather: any artefact present in the training images becomes part of the optimisation signal unless it is explicitly modelled, masked, or averaged away.

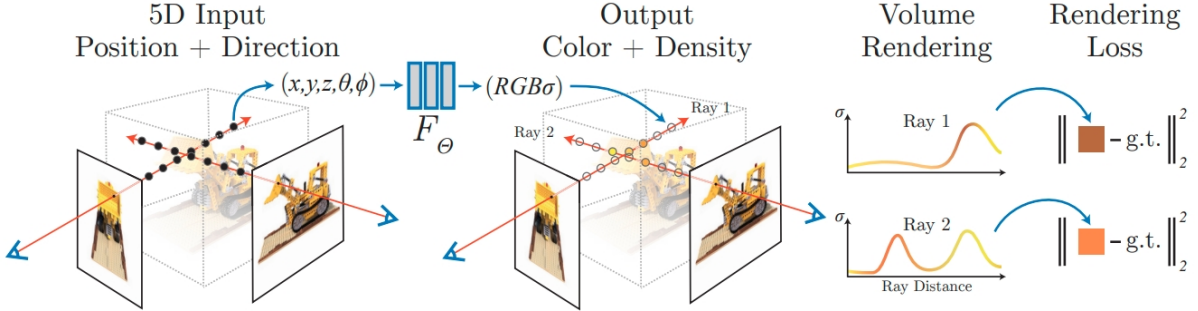


Figure 2.6: The NeRF training pipeline (Mildenhall et al., 2020). Reproduced from the Open Access version of the paper at https://www.ecva.net/papers/eccv_2020/.

Foundational extensions. NeRF established the quality standard for novel-view synthesis from multi-view captures and spawned a rich family of extensions. Mip-NeRF (Barron et al., 2021) replaced NeRF’s point-based sampling with conical frustum integration, reducing aliasing at multiple scales. Mip-NeRF 360 (Barron et al., 2022) handles unbounded outdoor scenes through integrated positional encoding and a scene contraction that maps distant points into a bounded region, which is relevant for driving scenes where content extends hundreds of metres down the road. Ref-NeRF (Verbin et al., 2022) models specular reflections explicitly, improving reconstruction of shiny surfaces, a situation that arises frequently in wet driving scenes. NeRF-W (Martin-Brualla et al., 2021) extends NeRF to unconstrained photo collections with per-image appearance embeddings and transient-content modelling.

Efficient training and representation. NeRF’s main practical limitation was computational cost: training a single scene took hours to days, and rendering a single frame took seconds to tens of seconds. A family of accelerated NeRF variants addresses this. Instant-NGP (Müller et al., 2022) uses a multi-resolution hash grid that encodes spatial

features efficiently, reducing training time to seconds for small scenes and minutes for larger ones. Plenoxels (Fridovich-Keil et al., 2022) replaces the MLP with a sparse voxel grid of spherical-harmonic coefficients, supporting fast gradient-based optimisation without neural networks. TensorRF (Chen et al., 2022a) decomposes the radiance field into low-rank tensors for compact representation and fast rendering. K-Planes (Fridovich-Keil et al., 2023) and HexPlane (Cao and Johnson, 2023) extend tensor factorisation to dynamic scenes.

Dynamic and driving-scale NeRFs. Extending NeRF to dynamic scenes introduces a time dimension. D-NeRF (Pumarola et al., 2021) and HyperNeRF (Park et al., 2021) model scene deformations through continuous warping fields. For driving scenes specifically, Block-NeRF (Tancik et al., 2022) decomposes city-scale reconstructions into tiled blocks; Mega-NeRF (Turki et al., 2022) supports aerial photogrammetry at building-scale. Neural Scene Graphs (Ost et al., 2021) model compositional driving scenes with separate NeRFs for the static background and each dynamic actor. SUDS (Turki et al., 2023) extends this decomposition to large-scale urban scenes.

Deblur-NeRF (Ma et al., 2022a; Chen and Liu, 2024) and the video-oriented approach of Seiskari *et al.* (Seiskari et al., 2024) handle motion blur and rolling-shutter artefacts. Night-NeRF and its derivatives (Cui et al., 2024b) address low-light scenes. These reconstruction-robustifying approaches are the closest prior art to the weather-robust NeRF reconstruction problem tackled by WeatherGSChapter 5, although none has the specific particle and occlusion structure that defines the weather case.

NeRF in adverse weather. For adverse weather, the NeRF literature is thinner. De-rainNeRF (Li et al., 2024) adapts NeRF to remove lens raindrops from multi-view captures, and Aleth-NeRF (Cui et al., 2024b) addresses low-light scenes. Both are specialised to a single degradation type and, more fundamentally, inherit NeRF’s computational profile: training takes hours per scene and rendering takes seconds per frame on standard hardware. This renders them impractical for driving-scale scene assets, which typically contain hundreds of metres of road captured at many viewpoints. ClimateNeRF (Li et al., 2023a) addresses the complementary problem of adding static weather (accumulated snow, floodwater) to clean NeRF reconstructions, a capability that the thesis’s 3D generation chapter7Chapter 7 extends with dynamic weather effects and the move to the 3DGS backbone.

The defining limitation of NeRF for the purposes of this thesis, therefore, is not quality but efficiency. This gap motivated the adoption of 3D Gaussian Splatting as the 3D backbone of both 3D chaptersChapter 7Chapter 5 of the thesis.

NeRF nevertheless remains important as the conceptual predecessor of the thesis’s 3D work. It established differentiable novel-view synthesis as a practical way to turn image sequences into scene assets, and it made clear why weather-aware reconstruction matters:

if corrupted observations are optimised into the scene representation, the resulting asset becomes unsuitable for simulation.

2.2.2 3D Spatial Representation: 3D Gaussian Splatting

3D Gaussian Splatting (3DGS) (Kerbl et al., 2023) replaces the implicit MLP representation of NeRF with an explicit set of 3D Gaussian primitives. Each Gaussian is parameterised by a centre $\boldsymbol{\mu} \in \mathbb{R}^3$, an anisotropic covariance matrix $\boldsymbol{\Sigma}$, an opacity $\alpha \in [0, 1]$, and a view-dependent colour represented by spherical harmonics. In practice, the covariance is factorised into rotation and scale parameters to ensure positive semi-definiteness. The density contribution of a Gaussian at a 3D location $\mathbf{x}$ is given by

$$G(\mathbf{x}) = \exp\left(-\frac{1}{2}(\mathbf{x} - \boldsymbol{\mu})^\top \boldsymbol{\Sigma}^{-1}(\mathbf{x} - \boldsymbol{\mu})\right). \quad (2.8)$$

At rendering time, each 3D Gaussian is projected onto the target image plane as a 2D Gaussian using the camera viewing transform and its local Jacobian. The projected Gaussians are then depth-sorted and composited front-to-back using α -blending (Fig. 2.7). This differs fundamentally from NeRF-style volume rendering: rather than evaluating an MLP at many samples along every ray, 3DGS rasterises a finite set of explicit primitives. Because this rasterisation can be implemented efficiently on the GPU using tile-based splatting, 3DGS achieves real-time or near-real-time rendering at high resolution while retaining competitive visual quality. Its explicit representation is also directly editable: opacity, colour, scale, position, and even individual primitives can be modified without reparameterising an implicit neural field.

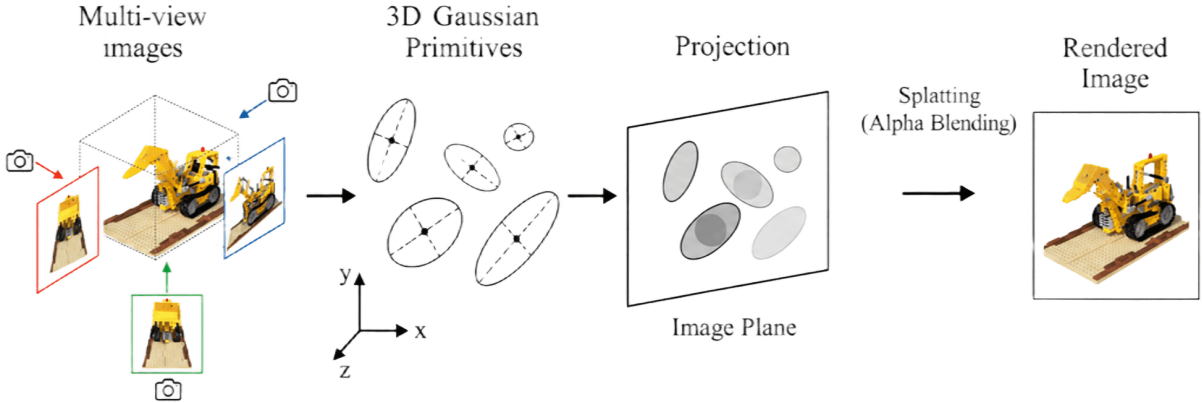


Figure 2.7: The 3D Gaussian Splatting pipeline.

Training dynamics. Training typically begins from a sparse 3D point cloud obtained by Structure-from-Motion (Schonberger and Frahm, 2016), which estimates camera poses and scene geometry by matching visual features across multiple images and triangulating their corresponding 3D locations. Each point is then converted into a Gaussian primitive, whose position, covariance, opacity, and colour coefficients are optimised through differ-

entiable rendering. A key component of 3DGS is adaptive density control: Gaussians are cloned or split in under-reconstructed regions and pruned when their opacity becomes negligible. This mechanism allows the representation to allocate greater capacity to regions that require finer geometric or appearance detail.

The standard photometric objective combines a reconstruction term with a structure term, balancing pixel-level fidelity with structural similarity. In practice, stable optimisation depends on careful initialisation, gradient-based densification criteria, opacity-reset schedules, and pruning thresholds. A substantial body of follow-up work has therefore analysed and improved these training dynamics (Lee et al., 2024; Lu et al., 2024; Yu et al., 2024). For adverse weather reconstruction, however, the same adaptive behaviour creates a specific failure mode: if rain streaks, fog, snowflakes, or lens droplets are repeatedly observed in the training views, the densification process may allocate Gaussian primitives to the weather artefacts themselves, treating them as scene content rather than corruption.

Quality and compactness improvements. Since its introduction, 3DGS has been extended along several axes. Mip-Splatting (Yu et al., 2024) introduces anti-aliasing filters to reduce scale-dependent artefacts when rendering views at distances different from those in the training set. Scaffold-GS (Lu et al., 2024) predicts Gaussian attributes from a learned neural scaffold, improving compactness and generalisation. 2D Gaussian Splatting (Huang et al., 2024) uses surface-aligned 2D Gaussians to obtain cleaner geometric reconstructions for surface-centric applications. SuGaR (Guédon and Lepetit, 2024) extracts an explicit mesh from a 3DGS reconstruction, bridging Gaussian radiance fields with traditional graphics pipelines. Compact3D (Niedermayr et al., 2024) applies codebook-based quantisation to reduce memory footprint while preserving rendering quality.

Dynamic 3DGS. A second line of work extends 3DGS from static scenes to dynamic ones. 4D Gaussian Splatting (Wu et al., 2024a) models spatio-temporal variation with explicit deformation fields. Dynamic 3D Gaussians (Luiten et al., 2024) track Gaussian primitives through time using dense tracking losses. Deformable 3D Gaussians (Yang et al., 2024b) introduce time-conditioned deformation networks, while Periodic Gaussians (Chen et al., 2023) target scenes with periodic motion. These methods provide the broader technical context for representing time-varying 3D structure.

Large-scale and driving-scale 3DGS. Autonomous driving scenes introduce additional challenges for 3DGS. They cover large spatial extents, contain rapid ego-motion, include dynamic agents such as vehicles and pedestrians, and often require consistent reconstruction across long trajectories. Recent work has therefore adapted Gaussian representations to outdoor and driving-scale environments. OmniRe (Chen et al., 2025)

reconstructs driving scenes by separately modelling static background, dynamic actors, and ego-motion; City-scale methods such as 3D Gaussian Street Mapping (Liu et al., 2024c) and Hierarchical 3DGS (Kerbl et al., 2024) improve scalability through partitioning and hierarchical scene organisation. DrivingGaussian (Zhou et al., 2024) focuses specifically on autonomous driving reconstruction, with mechanisms for handling ego-motion and dynamic agents. These developments make 3DGS increasingly suitable as a spatial backbone for driving scene simulation and editing.

Scene editing. The explicitness of 3DGS has also made it a natural representation for scene editing. Each Gaussian is a manipulable primitive rather than a hidden state inside an implicit neural network, allowing edits to be applied directly to individual Gaussians or semantically grouped regions. GaussianEditor (Chen et al., 2024b) combines 2D diffusion-based editing with 3D Gaussian manipulation for text-driven scene modification. StyleGaussian (Liu et al., 2024a) performs style transfer by modifying Gaussian colour attributes while preserving scene geometry. Gaussian Grouping (Ye et al., 2024) assigns semantic labels to Gaussians, enabling group-level editing. Instruct-GS2GS (Vachha and Haque, 2024) combines InstructPix2Pix with 3DGS for instruction-driven 3D scene editing.

WeatherEdit builds on this editing literature but targets a different problem: weather injection in reconstructed driving scenes. Instead of editing semantic objects or applying global style changes, WeatherEdit combines 2D diffusion editing for static background appearance with explicit Gaussian-particle manipulation for dynamic weather effects. This separation is important because adverse weather contains both persistent scene-level changes, such as snow accumulation or wet surfaces, and transient volumetric effects, such as rain or snow particles.

3DGS in adverse conditions. Despite rapid progress in 3DGS, reconstruction under adverse weather remains comparatively underexplored. At the time of the WeatherGS publication, no published method directly addressed 3DGS-based reconstruction under combined adverse weather degradations. BadGaussians (Zhao et al., 2024) considers motion blur, but not weather. GS-W (Zhang et al., 2024) handles in-the-wild appearance variation, but not the particle, scattering, and lens-occlusion structure of adverse weather. Concurrent Gaussian-based restoration methods for low light or haze address specific degradation types, but do not cover the combination of dense atmospheric effects and sparse lens occlusions.

This gap reflects a mismatch between the strengths of 3DGS and the demands of adverse weather reconstruction. 3DGS is efficient, explicit, and editable, but its photometric optimisation will faithfully reconstruct whatever is consistently observed in the training images, including weather artefacts, unless the optimisation is informed about which observations are corrupted. The resulting problem formulation motivates the 3D-

reconstruction component of this thesis.

2.3 Weather Restoration

A intuitive way to address weather-degraded observations is restoring them before they are passed to a downstream perception model. Weather restoration aims to recover visual evidence that has been corrupted by weather condition, or insufficient illumination. The literature can be broadly organised into three groups: specialised single-condition methods, unified multi-weather frameworks, and low-light enhancement methods. I review each group below, focusing on their relevance to autonomous driving perception and the limitations that motivate the restoration components of this thesis.

2.3.1 Single-type Restoration

Single-condition restoration assumes that the degradation type is known at deployment time and trains a dedicated model for each condition. Fig. 2.8 illustrates three representative targets: rain, snow, and fog. For each condition, a specialised model can recover a clearer version of the underlying scene, but its validity is largely restricted to the degradation type on which it was trained.



Figure 2.8: Representative single-type weather restoration; each pair shows the degraded input (left) and restored output (right). Examples reproduced from the official WeatherDiffusion repository accompanying Özdenizci and Legenstein (2023).

Deraining methods target rain streaks and, in some cases, lens raindrops. Rain streak removal is commonly formulated as a decomposition problem: the observed image is treated as a clean background plus a rain layer whose orientation, sparsity, and frequency statistics differ from scene texture. Early methods relied on signal-processing priors (Kang et al., 2012); later deep learning approaches introduced increasingly expressive restoration networks. Li *et al.* (Li et al., 2018a) proposed a recurrent network that decomposes rain into layers with different appearance characteristics. Qian *et al.* (Qian et al., 2018) introduced an attention-based raindrop-removal method that estimates a soft attention map over droplet-contaminated regions before restoring the image. This idea is inherited by the Lens Effect Detector in WeatherGS. PReNet (Ren et al., 2019) progressively refines the derained result over recurrent stages. DerainGAN (Yadav et al., 2021) casts single-image deraining as a Wasserstein-GAN problem, using the Wasserstein–Lipschitz objective to stabilise adversarial training. Uformer (Wang et al., 2022) applies U-shaped transformers to image restoration, including deraining. In the video domain, temporal

methods (Ren et al., 2017; Wen et al., 2023) exploit redundancy across frames to separate transient rain from persistent scene content.

Dehazing methods aim to invert the atmospheric scattering model in Eq. (1.1). They estimate or bypass the transmission process using either handcrafted priors (Berman et al., 2016) or learned mappings (Li et al., 2021; Wu et al., 2021; Zhang et al., 2022; Zheng et al., 2023). In the physical model, the observed image combines attenuated scene radiance with airlight, and the transmission term decreases with scene depth. Classical priors estimate this transmission directly from the image, while deep methods either predict the transmission map or learn an end-to-end mapping from hazy to clear images. Contrastive approaches (Wu et al., 2021) distinguish hazy and clear content at the feature level using positive and negative pairs. D3-Dehaze (Huo et al., 2025) adopts a divide-and-conquer strategy, using specialised sub-networks for near and far hazy regions to better handle the depth-dependent nature of fog. This depth dependence is precisely why fog is difficult to resolve from a single frame and motivates the geometry-aware treatment used later in WeatherGS and WeatherEdit.

Desnowing is the least developed of the specialised restoration tracks. DesnowNet (Liu et al., 2018) introduced a multi-stage CNN for snow removal. DDMSNet (Zhang et al., 2021a) incorporated semantic and depth priors to improve restoration under complex snow patterns. Özdenizci and Legenstein (Özdenizci and Legenstein, 2023) extended denoising diffusion to multi-weather restoration through patch-based sampling, demonstrating strong performance on snow removal alongside rain and haze. Snow is challenging because it combines several visual phenomena: bright airborne particles, partial occlusion, contrast reduction, and surface accumulation. Most desnowing methods focus on falling snowflakes, while accumulated ground snow, which changes road texture and can obscure lane markings or kerbs, is usually outside their scope. SmartAssign (Wang et al., 2023b) addresses the related desnowing–deraining setting through explicit knowledge assignment between the two tasks.

Overall, specialised restoration methods can perform well within their target domain, but they remain limited for autonomous driving deployment. A vehicle cannot assume that a single, known degradation will occur in isolation, and real driving conditions often involve mixtures such as rain at night, fog at dusk, or snow with low illumination. These limitations motivate unified restoration methods.

2.3.2 Unified, Multi-type Restoration

Unified restoration methods train a single network to handle multiple adverse conditions within one set of parameters. Fig. 2.9 contrasts three architectural families. Task-specific networks (Yadav et al., 2021; Liu et al., 2018) train a separate encoder–decoder pair for each weather type and select the appropriate model at inference time. This provides high per-condition capacity but prevents cross-condition sharing. Partial-sharing all-in-one

design(Li et al., 2020) uses weather-specific encoders with a shared decoder, assuming that low-level degradations require dedicated front-ends while high-level reconstruction can be shared. Fully shared all-in-one design(Chen et al., 2022b) use one encoder and one decoder across all conditions, yielding the smallest parameter footprint but requiring weather-agnostic representations throughout the network.

Most recent work has moved towards fully shared all-in-one restoration. The all-in-one network (Li et al., 2020) was among the first to use neural architecture search to discover a model capable of handling rain, fog, and snow. TransWeather (Jose Valanarasu et al., 2022) simplified the design with a single-encoder single-decoder transformer, showing that attention can learn features shared across weather types. Chen *et al.* (Chen et al., 2022b) distilled knowledge from specialised teacher networks into a unified student using contrastive regularisation. WGWS-Net (Zhu et al., 2023) introduced a similarity-aware mechanism that adapts representations according to the weather content of each image.

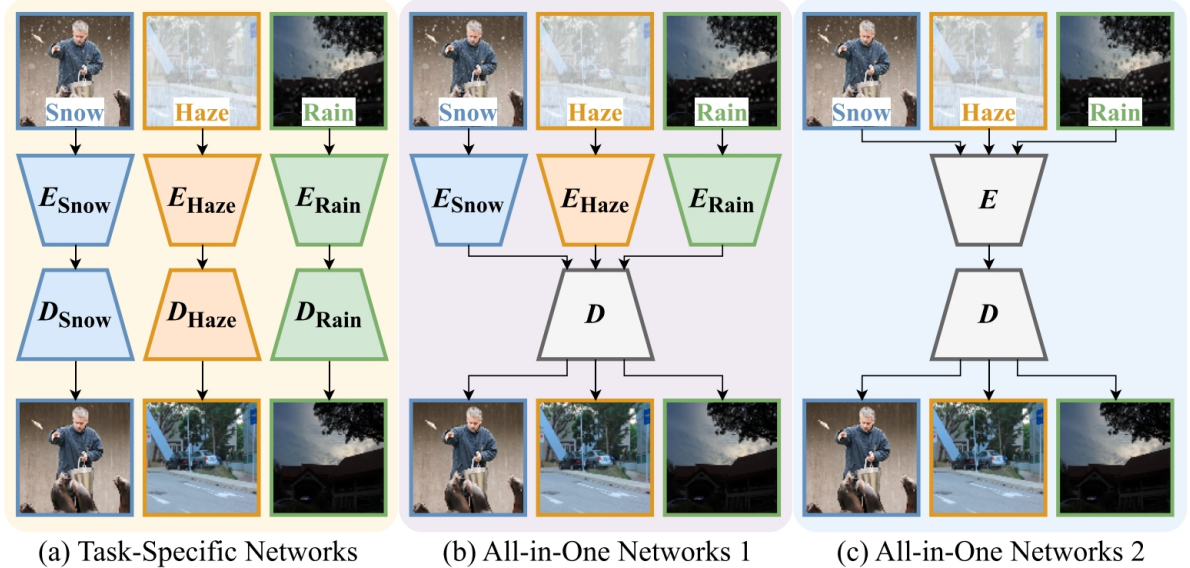


Figure 2.9: Three architectural families for handling rain, snow, and haze within one restoration system, adapted from (Jose Valanarasu et al., 2022) at <https://openaccess.thecvf.com/CVPR2022>.

Diffusion-based unified restoration has emerged more recently. The diffusion test-time adaptation method of Yang *et al.* (Yang et al., 2024a) applies diffusion priors to video adverse-weather removal, using the learned clean-image distribution as an implicit restoration prior. (Xie et al., 2026) extend this idea with a dual state-space diffusion backbone, where degradation-aware guidance routes weather-specific features through a shared denoising path. WeatherStream (Zhang et al., 2023a) contributes a large real-world paired dataset for benchmarking unified weather restoration. The plugin-based method of Liu *et al.* (Liu et al., 2024b) selects task-specific diffusion plugins using CLIP similarity between the input image and textual degradation descriptions, and forms the basis of the Atmospheric Effect Filter in WeatherGS.

Despite their wider coverage, unified methods still leave two limitations. First, they

mainly focus on removing visible weather particles, such as streaks, snowflakes, or haze, while broader appearance shifts (including contrast loss, colour bias, illumination imbalance, and texture suppression) also affect downstream perception. Second, they typically treat low-light enhancement as a separate problem, even though low illumination frequently co-occurs with rain, snow, and fog in real driving scenes.

2.3.3 Low-light Enhancement

Low-light enhancement has developed largely as a parallel restoration track. Classical methods include histogram equalisation (Abdullah-Al-Wadud et al., 2007) and Retinex-based approaches (Jobson et al., 1997b,a) decompose an observed image into reflectance and illumination components using handcrafted priors, thereby improving local contrast and correcting non-uniform illumination. Deep learning methods later introduced data-driven enhancement models. RetinexNet (Wei et al., 2018) learns Retinex decomposition end-to-end; EnlightenGAN (Jiang et al., 2021) removes the need for paired low-light and normal-light data through unsupervised adversarial training; and Zero-DCE (Guo et al., 2020) formulates enhancement as per-pixel curve estimation without paired or unpaired supervision. Subsequent SNR-aware, degradation-aware, and Retinex-inspired methods further improve robustness under diverse low-light conditions (Jiang et al., 2022; Dong et al., 2022; Xu et al., 2022; Yang et al., 2020b; Liu et al., 2021; Wu et al., 2022; Zhang et al., 2021b, 2019c; Ma et al., 2022b; Fu et al., 2023).

Fig. 2.10 shows a representative result: a dark driving scene becomes more legible after enhancement, revealing road users, vehicles, buildings, and surface detail. Such methods perform well when the dominant problem is insufficient illumination. However, they are not designed for fog, snow, or rain, where the main degradation may be scattering, occlusion, or surface appearance change rather than darkness. Applying a low-light enhancer to a snow-covered or foggy scene can therefore produce over-exposure, contrast exaggeration, or colour distortion. This motivates a unified treatment in which low-light enhancement is handled together with adverse weather restoration rather than as a separate preprocessing step.



Figure 2.10: Representative low-light driving image (left) and enhanced result (right). Example from EnlightenGAN (Jiang et al., 2021).

The restoration literature leaves two gaps. At the 2D level, existing methods do not jointly handle multiple weather types, low-light conditions, broad appearance correction, and imperfectly aligned driving data; this motivates AllWeatherNet. At the 3D level, restoration must preserve multi-view geometry rather than only improve per-image appearance; this motivates WeatherGS, which couples weather-aware restoration with 3D reconstruction.

2.4 Weather Generation

Robust autonomous driving perception requires exposure to weather conditions that are difficult, unsafe, or impractical to collect at scale. Weather generation addresses this need by synthesising adverse weather observations for training, testing, and validation. Existing methods span two broad families: physics-based simulation, which offers explicit control over weather parameters, and neural synthesis, which improves photorealism by learning weather appearance from data. I review both families below and highlight the trade-off between controllability, realism, and multi-view consistency that motivates WeatherDG and WeatherEdit.

2.4.1 Physics-Based Simulation

Traditional weather synthesis relies on physical models of the image-formation process. Rain rendering models droplet trajectories, fall velocities, exposure time, illumination, and optical scattering to generate streak patterns (Hahner et al., 2019). Streak length is determined by droplet velocity and camera exposure, while brightness depends on lighting and scattering. Fog synthesis typically applies the atmospheric scattering model in Eq. (1.1), using known or estimated depth maps to introduce distance-dependent attenuation and airlight. Sakaridis *et al.* (Sakaridis et al., 2018), for example, used this strategy to construct the Foggy Cityscapes benchmark. Snow synthesis combines particle systems for falling snowflakes with surface-accumulation models (Liu et al., 2018). Fig. 2.11 shows a representative example in which a Blender particle system emits snowflakes from a planar source under gravity and wind, and the renderer composites them into the scene as bright moving particles.

Graphics-based driving simulators such as CARLA (Dosovitskiy et al., 2017) and AirSim (Shah et al., 2018) extend this paradigm to full 3D environments, enabling controllable weather rendering with native temporal and multi-view consistency. Standalone particle-simulation libraries (Stomakhin et al., 2013; Zsolnai-Fehér, 2022; Feldman and O’Brien, 2002; Haas, 2014) can model dynamic behaviour in greater detail, but they still require a separate rendering pipeline to produce camera-like images. A physically plausible velocity field for snow or rain does not by itself generate the atmospheric tone, wet-road reflections, sensor noise, or material-level appearance changes that make driving

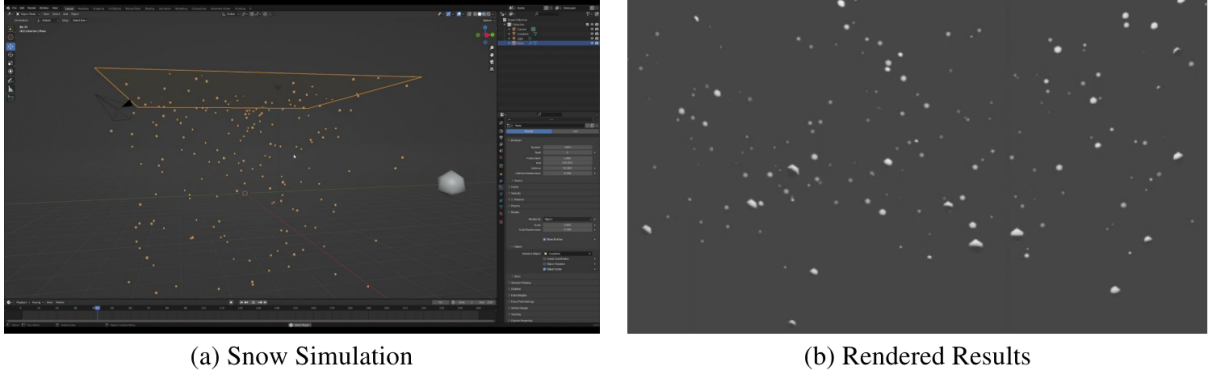


Figure 2.11: Physics-based weather synthesis in a graphics simulator.

footage look realistic.

Physics-based methods are therefore controllable by construction: density, fall speed, wind, visibility, and scattering coefficients can be exposed as explicit parameters. Their limitation is photorealism. In many driving simulators, rendered weather still carries a recognisable simulator aesthetic, with simplified materials, limited sensor realism, and insufficiently subtle interactions between weather, lighting, and scene appearance. This sim-to-real gap limits their usefulness as direct training data for perception models. It is the reason WeatherDG begins from diffusion-based image generation rather than relying solely on simulator output.

2.4.2 Neural Weather Synthesis

Neural weather synthesis improves photorealism by learning weather appearance from data. Early methods such as WeatherGAN (Hwang et al., 2022) and Schmidt *et al.* (Schmidt et al., 2019) use CycleGAN-style translation to edit weather conditions in driving images. ClimateGAN (Schmidt et al., 2022) uses GAN-based inpainting to visualise flooding and water reflections, targeting climate-change communication rather than autonomous driving perception directly. More recent diffusion-based editors, including ControlNet (Zhang et al., 2023b), InstructPix2Pix (Brooks et al., 2023), and TurboEdit (Deutch et al., 2024), produce higher-quality edits by leveraging large-scale generative priors. However, these general-purpose editors are not specialised for driving weather. They can produce stylised, viewpoint-inconsistent, or distorted outputs, and they typically operate on single images rather than temporally or multi-view consistent driving sequences.

Weather-aware neural editors attempt to introduce explicit control over the target weather condition. A video-based design is illustrated in Fig. 2.12: an input video is encoded into a latent space, a weather-removal network produces a clean intermediate, and a weather-synthesis network reintroduces a target condition according to a continuous attribute vector. This structure separates the input weather state from the desired output weather state, enabling controlled transitions between conditions such as fog, rain, snow, puddles, and snow cover.

Neural rendering approaches lift weather synthesis from 2D images to 3D scenes.

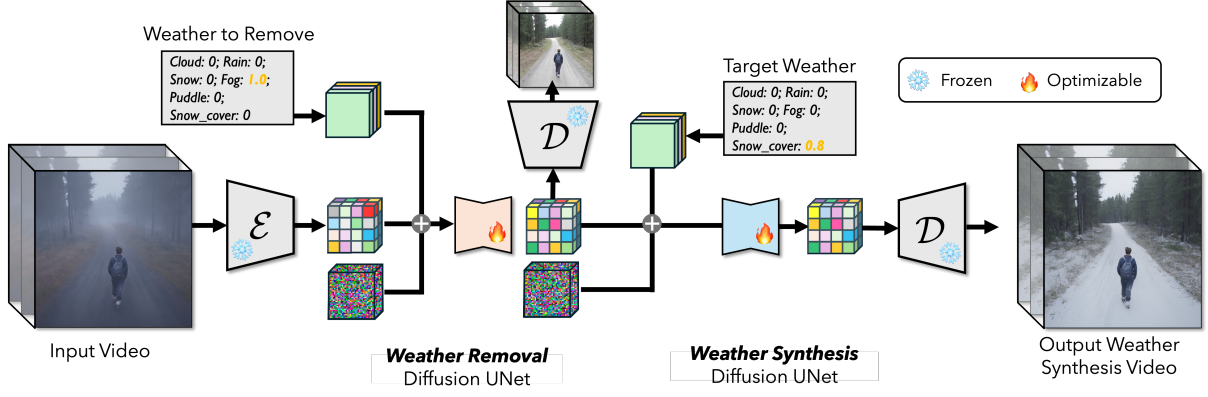


Figure 2.12: A representative latent-space neural weather-synthesis pipeline (Lin et al., 2025). Reproduced from the Open Access version of the paper at <https://openaccess.thecvf.com/ICCV2025>.

ClimateNeRF (Li et al., 2023a) uses NeRF to model static post-weather states, such as accumulated snow or flooded streets, with multi-view consistency. However, it does not generate dynamic weather events such as falling rain or snow and is not directly formulated for 3D Gaussian Splatting. General 3D stylisation methods (Li et al., 2018b) can alter appearance, but the resulting effects are usually too subtle or style-oriented for realistic adverse weather simulation. Emerging 3D-aware diffusion editors (Haque et al., 2023; Zhuang et al., 2023) use 2D diffusion priors to guide edits in 3D representations, but they are not designed around weather-specific requirements such as volumetric particles, accumulation, scattering, and controllable severity.

These methods extend weather synthesis beyond single-image editing, but they still leave the core requirements only partially addressed. Physics-based simulation provides explicit control but often lacks photorealism; 2D neural editing improves realism but lacks 3D consistency; and neural rendering provides spatial consistency but has so far focused mainly on static or weakly controllable effects. For autonomous driving training and validation, however, generated scenes must satisfy all three requirements simultaneously: they must be realistic enough to transfer to perception models, consistent enough to support multi-camera and temporal evaluation, and controllable enough to cover the desired operational design domain.

2.5 Summary

This chapter reviewed the foundations and applications that underlie the thesis. Generative models provide the priors needed to restore degraded images and synthesise missing weather-affected observations. Spatial models provide the 3D structure needed to reason about geometry, visibility, multi-view consistency, and controllable scene editing. The applied restoration and generation literature combines these capabilities only partially: restoration methods improve individual images but often ignore 3D consistency; physics-based simulation offers control but suffers from limited realism; neural generation

improves photorealism but often lacks precise control and multi-view structure.

The principal gap is therefore not the absence of generative models or spatial representations in isolation, but the lack of a framework that organises their interaction across the regimes required for adverse weather autonomous driving. This thesis addresses that gap through four complementary methods: 2D all-weather enhancement, 3D weather-robust reconstruction, 2D adverse weather data generation, and 3D controllable weather synthesis. The next chapter consolidates these unmet needs into four research gaps and formalises the framework and research questions that guide the remainder of the thesis.

Chapter 3

Research Gaps and Research Questions

Chapter 1 motivated weather-robust perception as a safety-critical problem and surveyed three families of existing approaches: visual restoration, domain adaptation and generalisation, and data augmentation. Chapter 2 then reviewed the underlying modelling foundations, including generative models that provide image-level priors and spatial models that capture 3D structure, as well as existing restoration and generation methods for addressing adverse weather problems. The present chapter introduces the gaps in existing literature. For each gap, we state the open problem, situate it against the most directly relevant prior art, and translate it into the explicit requirements that solution must satisfy. The chapter is organised as follows. Section 3.1 states the four gaps and their requirements. Section 3.2 develops the data-centric framework that situates them. Section 3.3 casts each gap as a research question. Section 3.4 maps gaps and questions to the technical chapters.

3.1 Research Gaps

The four gaps are presented in the order in which the thesis addresses them. The first two concern processing observations that already exist, at the single-frame (Gap 1) and multi-frame (Gap 2) levels of scope respectively. The second two concern synthesising observations that real data does not contain, at the same two levels of scope (Gap 3 for single-frame, Gap 4 for multi-frame).

3.1.1 Gap 1: Unified Framework for Improving Clarity of Weather-Degraded Inputs

Existing restoration methods do not yet provide a unified framework for addressing the global appearance shifts that adverse weather induces in driving imagery and that propagate into downstream perception. The two prevailing branches of the literature each fail

in a different way. Condition-specific methods (Wang et al., 2022; Zhang et al., 2019b; Jiang et al., 2021), (Berman et al., 2016) can recover strong results within their target degradation, but they require brittle deployment-time switching: the model that is selected at inference must match the weather actually present, an oracle that is not available on a moving vehicle. All-in-one restoration pipelines (Zhu et al., 2023; Sun et al., 2024; Guo et al., 2024a; Cui et al., 2024a; Zamfir et al., 2025) reduce this deployment-time switching, and recent diffusion-based variants further broaden adverse-weather restoration pipelines (Yang et al., 2024a; Chen et al., 2024a; Liu et al., 2024b). However, these methods remain primarily image-level operators tuned for the suppression of visible weather particles. They handle streaks, haze, and flakes effectively yet are less reliable for the broader appearance shifts that determine downstream perception accuracy, including contrast collapse, colour cast, illumination imbalance, and large-region texture loss (Jiang et al., 2025).

A unified enhancement framework that closes this gap must therefore satisfy three coupled requirements: (i) operate as a single inference-time model across the principal adverse conditions of driving (rain, snow, fog, and low light), so that no per-condition switching is required at deployment; (ii) address the global appearance shifts including illumination, contrast, colour, and texture jointly with particle-level artefacts, since these are the signals on which semantic segmentation actually depends; and (iii) demonstrate the improvement on a downstream perception metric, rather than on image-quality scores alone, since image-level cleanliness does not by itself guarantee that the segmentation model recovers the missing class evidence. No method in the literature meets all three of these requirements.

3.1.2 Gap 2: Simulation-Ready Scene Reconstruction from Weather-Degraded Captures

Current scene reconstruction approaches have limited applicability in recovering 3D geometry from weather-degraded multi-view observations. They typically address only a narrow subset of weather effects, such as fog or lens raindrops in isolation, while neglecting that real driving captures contain both rain streaks and lens droplets. Standard reconstruction pipelines, including Neural Radiance Fields (Mildenhall et al., 2020) and 3D Gaussian Splatting (Kerbl et al., 2023), are formulated under a clean-input assumption; when the training views are corrupted, view-inconsistent artefacts are either averaged into blurred geometry (NeRF) or absorbed into the explicit primitives as if they were genuine scene content (3DGS). The closest published precedent, DerainNeRF (Li et al., 2024), demonstrates that the NeRF backbone can be adapted to a single artefact type (lens droplets), but it does not address the dense atmospheric particles that often co-occur with droplets in the same capture.

The open problem is therefore reconstruction under the joint presence of two struc-

turally distinct artefact classes that co-occur in real driving footage: dense, small-scale atmospheric particles such as rain streaks and airborne snowflakes, and sparse, larger-scale lens-level occlusions such as water droplets and adhered snow on the cover glass. The two classes admit different remedies, and any 3D reconstruction framework that closes this gap must (i) treat the two artefact classes with distinct mechanisms appropriate to their spatial and statistical structure rather than relying on a single per-pixel restoration, and (ii) produce clean 3D geometry that can be rendered from arbitrary viewpoints without the weather artefacts reappearing as part of the reconstructed scene.

3.1.3 Gap 3: Scalable Adverse Weather Data Generation in Driving Context

Existing augmentation pipelines for augmenting weather driving datasets do not adequately account for driving scene structure or class-balanced adverse weather synthesis. The two prevailing alternatives each fail on one of these two aspects. Graphics-based simulators such as CARLA (Dosovitskiy et al., 2017) expose explicit controls for weather type, intensity, time of day, and viewpoint, and they naturally preserve dashcam perspective by construction; however, they retain a recognisable simulator aesthetic and a persistent sim-to-real gap in appearance, sensor characteristics, and material statistics. Pre-trained text-to-image generative models, conversely, deliver photorealistic appearance but are not specialised for autonomous driving: prompts intended to describe a driving scene routinely return artistic, cinematic, or otherwise non-dashcam compositions that fail to preserve dashcam perspective, semantic layout, and rare safety-critical classes such as riders, motorcycles, and pedestrians, which are intrinsically sparse in any image-level prior. Recent diffusion-based driving scene augmentation methods (Niemeijer et al., 2024; Benigmim et al., 2024; Azuma et al., 2024; Jia et al., 2024) mitigate one or the other failure mode but inherit the layout biases of either the underlying simulator or the underlying foundation prior.

A generation pipeline that closes this gap must therefore satisfy three jointly held requirements: (i) preserve the driving scene structure expected by downstream perception, that is, dashcam-level camera perspective and plausible semantic layout of road, vehicles, and surrounding agents; (ii) provide class-balanced coverage of the adverse weather distribution, with explicit over-sampling of the safety-critical object classes that are otherwise under-represented; and (iii) operate without any real adverse weather reference imagery, since the very scarcity of such data is what motivates the augmentation problem in the first place.

3.1.4 Gap 4: Controllable and Dynamic Weather Data Generation for Driving Scene

There remains no general mechanism for synthesising controllable, dynamic weather effects into a reconstructed 3D driving scene. The closest published precedent, ClimateNeRF (Li et al., 2023a), demonstrates that weather can be injected into a NeRF reconstruction, but only as static post-weather appearance (accumulated snow, floodwater); it does not synthesise active dynamic events such as falling rain or drifting snow, and it does not connect the synthesised conditions to perception robustness for autonomous driving tasks. Two-dimensional editors such as ControlNet (Zhang et al., 2023b), InstructPix2Pix (Brooks et al., 2023), and TurboEdit (Deutch et al., 2024), applied independently to rendered views, deliver photorealistic weather appearance but provide no mechanism for keeping the edited weather consistent across the multiple synchronised cameras and consecutive frames that any usable 3D scene asset for automated driving must support.

A 3D weather-synthesis framework that closes this gap must therefore deliver four jointly held properties: (i) controllability over weather type and severity through an explicit user-facing interface, matching the parametric coverage that simulation-based validation expects; (ii) dynamic weather effects such as falling rain and drifting snow, in addition to the static post-weather appearance considered by prior 3D methods; (iii) synthesis into a reconstructed 3D scene so that the same weather event is observable from arbitrary cameras and frames with multi-view and multi-frame consistency; and (iv) a demonstrable contribution to perception robustness for autonomous driving, since the operational purpose of a weather simulator in this setting is to expose perception models to adverse conditions that physical testing cannot reliably reproduce.

3.2 The Data-Centric Framework

The four gaps introduced above organise along two orthogonal axes that together yield the data-centric framework of this thesis (Fig. 3.1). The primary axis is the availability of weather-affected observations. When such observations are available, the corresponding task is *Enhancement*: recovering clean representations from inputs that already exist (Gaps 1–2). When they are not because the conditions of interest are rare, unsafe to collect, or geographically localised the task becomes *Generation*: synthesising the missing observations from a textual or visual specification (Gaps 3–4). Data availability therefore sets the direction of the pipeline: from a degraded observation to a clean output for Enhancement, or from a specification to a weather-affected output for Generation. The secondary axis is the spatial dimensionality of the representation on which the method operates: 2D image-plane pipelines (top row, Gaps 1 and 3) and 3D scene-level pipelines (bottom row, Gaps 2 and 4). The two axes are independent, and each of the four gaps of Section 3.1 sits at one combination of availability and dimensionality, populated by one

technical chapter of the thesis.

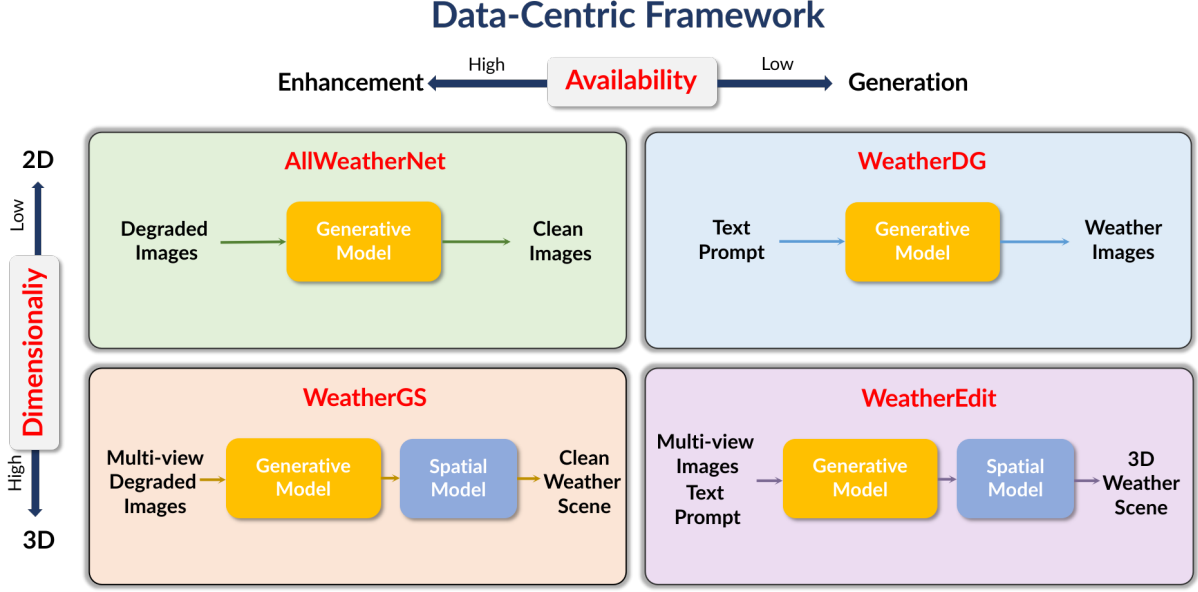


Figure 3.1: The data-centric framework of this thesis.

Two features of the framework merit emphasis. First, the title of the thesis *Generative and Spatial Modelling* names the structural asymmetry between the two rows. The 2D row uses a generative model alone, because a 2D image is the intrinsic object of perception and a single model can therefore map directly between input and output in pixel space. The 3D row demands both a generative model and an explicit spatial representation, because a 3D scene is a reconstructed abstraction whose cross-view consistency cannot be enforced by image-level generation alone. The spatial model (3D Gaussian Splatting in this thesis) supplies the geometric structure that ties multiple views to one world. Second, the four methods are composable rather than competing. WeatherGS (3D Enhancement) produces the clean 3D assets that WeatherEdit (3D Generation) edits, while WeatherDG (2D Generation) supplies the training distribution that complements AllWeatherNet’s (2D Enhancement) inference-time pipeline. Chapter 8 revisits this compositional view as the thesis’s organising contribution.

3.3 Research Questions

We translate the four gaps into four research questions, one per cell of the framework. The numbering follows the order of the technical chapters.

RQ1. *How can a single image enhancement model improve the visual clarity of driving images degraded by different adverse conditions, including rain, snow, fog, and low light? More specifically, how can the model be guided to learn condition-invariant restoration cues while remaining effective across heterogeneous weather degradations?*

Investigated in Chapter 4 (AllWeatherNet).

RQ2. *How can a clean 3D driving scene be reconstructed from multi-view observations corrupted by adverse weather? In particular, how can dense atmospheric particles and sparse camera-lens occlusions be removed or excluded so that they are not reconstructed as part of the scene geometry?*

Investigated in Chapter 5 (WeatherGS).

RQ3. *How can adverse weather training data be generated without collecting real-world weather-degraded images? More specifically, how can general-purpose generative models be adapted and controlled to produce realistic, diverse, and class-balanced synthetic driving images for robust semantic segmentation?*

Investigated in Chapter 6 (WeatherDG).

RQ4. *How can diverse adverse weather effects be synthesised from existing clean driving data? In particular, how can dynamic weather phenomena, such as falling rain, drifting snow, and fog, be simulated in 3D with controllable type, event state, and severity while preserving multi-view and temporal consistency?*

Investigated in Chapter 7 (WeatherEdit).

3.4 Chapter Roadmap

The chapters progress through the two-stage pipeline: first processing observations that already exist (Chapters 4 and 5), then synthesising observations that real data does not contain (Chapters 6 and 7). Within each stage, the scope expands from single-frame to multi-frame formulations. Chapter 8 synthesises the four contributions, develops the cross-chapter themes that emerge only at the system level, and revisits the framework’s limitations and implications.

With the gaps, research questions, and chapter ordering in place, the thesis turns to its first technical contribution: Part II, Enhancement.

Part II

Enhancement: Recovering Clean Representations from Degraded Observations

Chapter 4

Improving Visual Clarity: AllWeatherNet

A deployed perception system must operate on the visual observations delivered by its onboard cameras, regardless of the environmental conditions under which those observations are acquired. A direct way to improve robustness under adverse weather is therefore to enhance the degraded input images before they are passed to downstream perception models. In this setting, the enhancement module is expected to recover a visually clearer representation that approximates the corresponding clear weather appearance, while preserving the scene structure required for reliable recognition.

This chapter studies such a 2D image-enhancement strategy and forms the first chapter of Part II of the thesis. I ask whether a single inference-time model, without modifying the downstream perception pipeline, can consistently restore visual clarity across four common adverse conditions, namely rain, snow, fog, and low light. More importantly, we examine whether improved visual quality leads to measurable gains in downstream semantic segmentation performance.

To address this problem, i introduce **AllWeatherNet** (Qian et al., 2024), a unified image-enhancement framework designed for all-weather driving scenes. The method is built upon three key components. First, the Scaled Illumination-Aware Attention Mechanism (SIAM) exploits scene illumination as a condition-invariant guidance signal for weather-adaptive restoration. Second, a hierarchical discriminator supervises the enhancement process at scene, object, and texture levels, encouraging both global consistency and local detail fidelity. Third, a Ranked Adaptive Window Pairing (RAWP) strategy improves robustness to the imperfect image correspondences commonly found in real-world driving datasets.

Experiments reported in the AllWeatherNet paper (Qian et al., 2024) on the Adverse Conditions Dataset with Correspondences (ACDC) (Sakaridis et al., 2021) show that the method improves the mIoU of a downstream pre-trained semantic segmentation model by up to 5.3%. On the Dark Zurich nighttime dataset (Sakaridis et al., 2020), it further achieves a 6.6 percentage-point improvement over the next best competing method. The

work presented in this chapter was published at the International Conference on Pattern Recognition (ICPR) 2024 (Qian et al., 2024), where it received the Piero Zamperoni Overall Best Student Paper Award.

4.1 Introduction and Motivation

A direct and practically deployable strategy for mitigating weather-induced visual degradation is to enhance the input images before they are processed by downstream perception models. When the downstream semantic segmentation network is fixed, for example one trained on Cityscapes (Cordts et al., 2016) or an equivalent clear weather dataset, an image enhancement module can be introduced at inference time without modifying the segmentation architecture or retraining its parameters. This setting defines the scope of the present chapter. Rather than adapting the segmentation model itself, assuming access to pixel-aligned clean/adverse image pairs, or constructing separate enhancement models for individual weather conditions, i investigate a unified enhancement network that maps a degraded driving image to a representation equivalent to a clear daytime capture, which is subsequently consumed by an unchanged segmentation model.

Building on the broader restoration landscape reviewed in Section 2.3, three main difficulties shape the design of a unified all-weather enhancement model.

Condition-invariant restoration. A unified all-weather enhancement model must operate across rain, snow, fog, and low-light conditions, despite the substantially different image degradations induced by each. Rain introduces streak-like occlusions and local intensity perturbations; fog reduces contrast and visibility through atmospheric scattering; snow produces bright particle-like artefacts and may obscure surface texture; and low light reduces the overall signal-to-noise ratio while altering scene illumination. These heterogeneous degradations make it undesirable to rely solely on weather-specific cues. Instead, effective all-weather restoration requires a guidance signal that is consistently defined across different conditions and remains informative under both locally and globally degraded appearance.

Multi-scale degradation modelling. Weather-related degradations affect driving scenes at multiple spatial scales. Fog may appear as a scene-level veil whose strength varies with distance; rain streaks typically occupy intermediate spatial frequencies; snow can introduce fine local particles while also suppressing texture over larger surface regions. Consequently, supervision at a single spatial scale is insufficient. A purely global objective may encourage overall photometric plausibility while failing to recover local detail, whereas purely local objectives may improve texture sharpness without preserving object- or scene-level consistency. Robust all-weather enhancement therefore requires

multi-scale supervision that jointly constrains global appearance, object-level structure, and local texture fidelity.

Learning from imperfect correspondences. Supervised image-to-image restoration is most effective when degraded inputs and clean targets are spatially aligned at the pixel level. In real driving datasets, however, such alignment is rarely available. Clear weather and adverse weather traversals of the same route are typically captured at different times, under different traffic configurations, and from slightly different camera poses. ACDC (Sakaridis et al., 2021) provides approximate correspondences between adverse and clear images, but these pairs are not pixel-aligned. Treating them as exact supervision can therefore impose erroneous local constraints, while discarding the correspondences altogether weakens the training signal. The learning procedure must therefore exploit approximate image-level correspondences while remaining robust to local spatial mismatch.

These observations motivate the three principal components of the method presented in this chapter. The Scaled Illumination-Aware Attention Mechanism (SIAM) addresses the requirement for condition-invariant restoration by using illumination intensity as a shared per-pixel guidance signal and by rescaling this signal to avoid suppressing information in either dark or bright regions. The hierarchical discriminator addresses the need for multi-scale degradation modelling by imposing adversarial supervision at scene, object, and texture levels. Finally, Ranked Adaptive Window Pairing (RAWP) addresses the problem of imperfect correspondences by selecting, within a local search window, the source patch that is most compatible with each target patch. Fig. 4.1 situates these components within the overall enhancement framework.

The qualitative effects of these three components are illustrated in Fig. 4.2, where AllWeatherNet is compared with three representative categories of prior approaches on a challenging adverse weather image.

4.2 Related Work

Section 2.3 provides a broader review of image enhancement and weather restoration. This section focuses on the lines of work most directly related to AllWeatherNet: weather effect removal, pixel-level image translation, and low-light enhancement.

Weather effect removal. Existing deraining (Yang et al., 2020a; Wang et al., 2022), dehazing (Berman et al., 2016; Zhang et al., 2019b, 2022), and desnowing (Liu et al., 2018; Zhang et al., 2021a; Wang et al., 2023b) methods are typically designed for a single degradation type and can achieve strong performance within their respective problem settings. Unified multi-weather frameworks (Li et al., 2020; Jose Valanarasu et al., 2022; Chen et al., 2022b) extend this scope by addressing multiple weather effects within a single

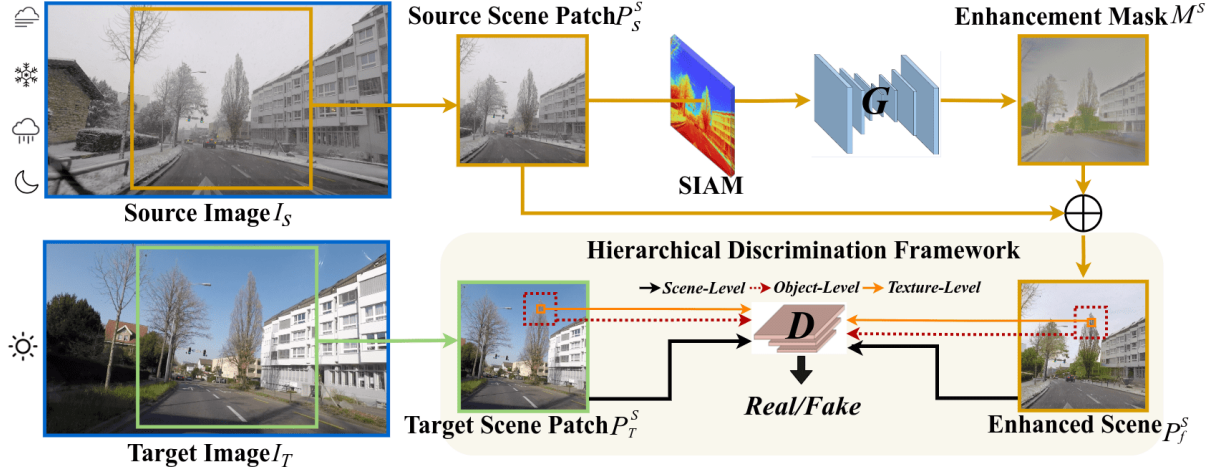


Figure 4.1: Overview of the AllWeatherNet architecture. A source scene patch P_s^S is processed by the Scaled Illumination-Aware Attention Mechanism (SIAM) and the generator to produce an enhancement mask M^S . The mask is added to the source patch to form the enhanced result P_f^S , which is evaluated by three hierarchical discriminators operating at scene, object, and texture scales. Ranked Adaptive Window Pairing (RAWP, not shown in this overview but detailed in Section 4.6) selects the target patches for the object and texture discriminators.

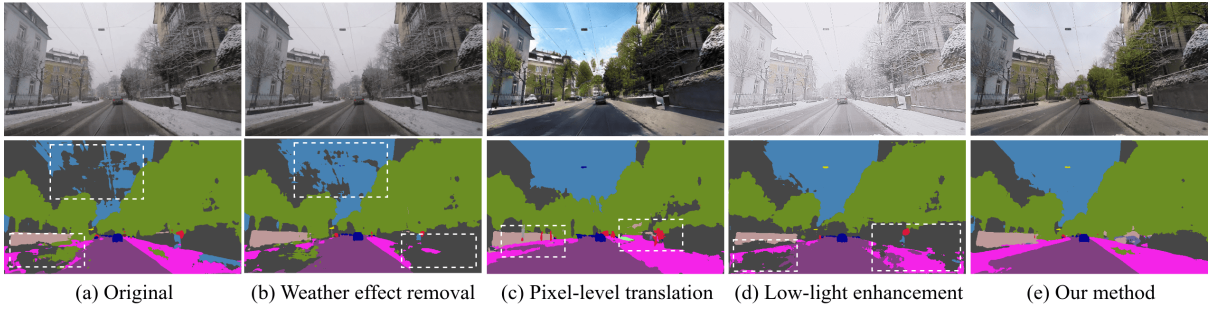


Figure 4.2: Qualitative comparison of representative image processing approaches on an adverse weather driving scene. (a) Input image. (b) A weather effect removal method (Chen et al., 2022b) preserves the overall scene structure but introduces noticeable colour imbalance. (c) A pixel-level translation method (Zhu et al., 2017) produces visually plausible appearance changes but hallucinates scene content. (d) A low-light enhancement method (Ma et al., 2022b) over-amplifies already bright regions, leading to saturation in snowy areas. (e) AllWeatherNet restores colour, contrast, and texture while preserving scene content and avoiding visible artefacts.

model. However, these methods primarily focus on the removal of visible weather particles or veil-like degradations, and are less effective at correcting broader appearance changes such as contrast attenuation, colour shift, and texture loss. Moreover, the mentioned existing unified methods do not explicitly address nighttime imagery, despite low-light conditions being a common and safety-critical adverse scenario for autonomous driving, often occurring together with rain.

Pixel-level image translation. Unpaired image-to-image translation methods, including CycleGAN (Zhu et al., 2017), CUT (Park et al., 2020), and Santa (Xie et al., 2023), can translate adverse weather images into visually plausible clear weather domains. Nevertheless, these methods operate by modifying the image appearance at the pixel level, which may alter scene content in addition to weather-related appearance. This introduces an important limitation for safety-critical perception: the translated image may hallucinate, remove, or distort objects that are relevant to downstream decision making. Task-aware variants such as ForkGAN (Zheng et al., 2020) mitigate this issue by incorporating downstream perception objectives, but the risk of content alteration remains when the translation process is not explicitly constrained to preserve scene structure.

Low-light enhancement. Classical low-light enhancement methods, such as histogram equalisation (Abdullah-Al-Wadud et al., 2007) and Retinex-based decomposition (Jobson et al., 1997b), provide early baselines for illumination correction. More recent deep learning approaches, including EnlightenGAN (Jiang et al., 2021), Zero-DCE (Guo et al., 2020), SCI (Ma et al., 2022b), and PairLIE (Fu et al., 2023), substantially improve visual quality and reduce the need for paired supervision. However, these methods are primarily formulated under the assumption that the dominant degradation is insufficient illumination. This assumption becomes problematic in adverse weather scenes where bright or moderately illuminated regions are already present, such as snow-covered surfaces or foggy regions with uniform grey intensity. Applying low-light enhancement indiscriminately to such scenes can therefore over-amplify bright areas and produce saturation artefacts, as illustrated in Fig. 4.2 (d).

Limitations motivating AllWeatherNet. Viewed as a whole, these three families of methods reveal a gap in existing image enhancement approaches for adverse weather driving scenes. Weather effect removal methods can reduce visible weather artefacts but often leave broader appearance degradation unresolved. Pixel-level translation methods can modify global appearance but may compromise scene fidelity through hallucination or content removal. Low-light enhancement methods improve dark regions but can fail when adverse weather introduces both under- and over-illuminated areas within the same image. Prior to AllWeatherNet, no unified framework simultaneously addressed rain, snow, fog, and nighttime imagery while restoring appearance beyond particle removal

and preserving scene content without wholesale image translation. The remainder of this chapter presents AllWeatherNet as a unified enhancement framework designed to address this gap.

4.3 Enhancement Pipeline

AllWeatherNet is motivated by a fundamental limitation of pixel-level image translation: the output space is weakly constrained. In such methods, the generator is required to synthesise the entire enhanced image, and the objective function may not sufficiently penalise content changes that remain visually plausible. This is undesirable for safety-critical perception, where preserving scene structure is as important as improving visual quality. AllWeatherNet therefore constrains the enhancement process by adopting a residual formulation. Rather than generating the final image directly, the network predicts a residual enhancement mask that is added to the degraded input. The generator is thus encouraged to modify only the weather-degraded components of the image, while leaving unaffected regions unchanged by construction.

4.3.1 Residual Formulation

Let I_S denote the source adverse weather image and let G denote the generator network. AllWeatherNet produces the enhanced image I' as

$$I' = G(I_S) + I_S. \quad (4.1)$$

Here, $G(I_S)$ represents the predicted residual enhancement mask. This formulation contrasts with pixel-level translation methods, as illustrated in Fig. 4.3(a), where the generator directly produces the final image and is typically supervised by adversarial, cycle-consistency, or contrastive objectives.

The residual formulation in Fig. 4.3(b) has two practical advantages. First, it reduces the effective search space of the generator. Image regions that already contain reliable visual information can be preserved through a near-zero residual, whereas a translation-based generator must reconstruct every pixel. Second, it weakens the tendency toward unconstrained content generation. Since the model only needs to predict the changes required for restoration, the enhancement process is biased toward local and appearance-level correction rather than wholesale image synthesis.

4.3.2 Theoretical Properties of the Residual Formulation

The residual formulation has several properties that are particularly relevant to image enhancement in safety-critical perception systems. These properties clarify why the method

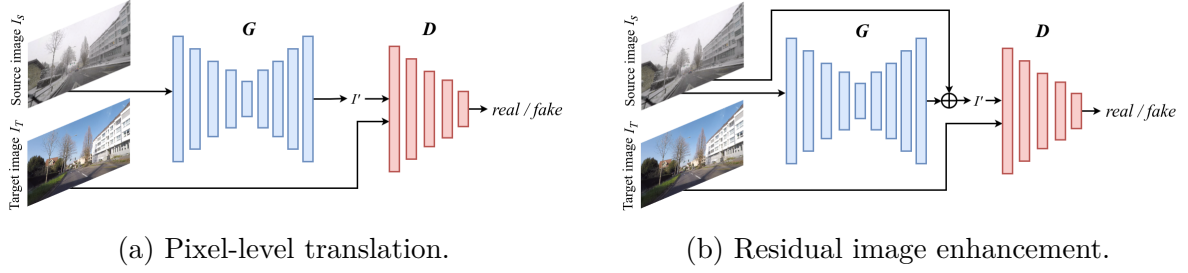


Figure 4.3: Comparison between pixel-level translation and residual image enhancement. (a) Pixel-level translation directly generates the enhanced image and therefore reconstructs all image content. (b) Residual image enhancement predicts an enhancement mask that is added to the input image, allowing unchanged regions to be preserved explicitly.

is less prone to the content-alteration failure modes associated with unconstrained image translation.

Constrained output space. A pixel-level translator $G_{\text{trans}}: I_S \mapsto I'$ optimises over the full image space of dimension $H \times W \times 3$. Even with perceptual or adversarial regularisation, this space may contain many visually plausible outputs that do not faithfully preserve the original scene content. The residual formulation restricts the effective output space by requiring the enhanced image to take the form $I_S + M$, where $M = G(I_S)$ is the predicted residual mask. In this view, enhancement is performed within a neighbourhood of the input image rather than through unconstrained resynthesis. Large content changes require correspondingly large residuals and are therefore less consistent with the inductive bias of the formulation. Conversely, leaving a pixel unchanged, i.e. $M(x, y) = 0$, is a natural solution whenever there is insufficient evidence that a modification is required. This behaviour is desirable for a perception preprocessor, for which preserving reliable input structure should be the default.

Gradient flow. The residual pathway also facilitates stable optimisation. For an adversarial loss L , the gradient with respect to a generator parameter θ can be written as

$$\frac{\partial L}{\partial \theta} = \frac{\partial L}{\partial I'} \frac{\partial G(I_S)}{\partial \theta}.$$

Since the enhanced image is obtained by adding the residual to the input, the input itself forms an identity pathway. When the predicted residual is close to zero, the model reproduces the source image, and the generator is primarily updated in regions where the discriminator provides evidence that enhancement is necessary. This resembles the optimisation benefit of identity mappings in residual networks (He et al., 2016). In the present setting, it allows large regions that are already visually reliable to remain stable, while concentrating learning on regions affected by weather degradation.

Failure mode under excessive residual growth. The residual constraint is imposed by the architecture and training objectives rather than by an explicit norm bound on M . Consequently, the formulation does not eliminate the possibility of content alteration entirely. In practice, excessively long training, an over-strong discriminator, or insufficient texture-level supervision can increase the magnitude of the residual and cause the generator to overwrite scene content. The implementation choices described in Section 4.8.4 define the operating regime in which this behaviour is avoided. In particular, reducing the texture-discriminator weight λ_3 weakens local fidelity constraints and can reproduce the hallucination artefacts that the residual formulation is designed to mitigate. This observation also supports the broader discussion in Chapter 8: the practical value of an enhancement method depends not only on its peak performance, but also on the stability and breadth of its well-conditioned operating regime.

4.3.3 End-to-End Flow

The complete enhancement pipeline, including the training-time auxiliary components, proceeds as follows. Given a source adverse weather image I_S and a loosely corresponding clear weather image I_T , a source scene patch P_S^s is cropped from I_S , and a target scene patch P_T^s is cropped from I_T . SIAM, described in Section 4.4, computes an illumination-aware attention map from P_S^s and injects this guidance into the U-Net-style generator shown in Fig. 4.1. The generator predicts a scene-scale residual mask M^s , producing the enhanced scene patch

$$P_f^s = P_S^s + M^s.$$

The scene-level discriminator D_s compares P_f^s with the corresponding target patch P_T^s . The object-level and texture-level discriminators, denoted by D_o and D_t , operate on progressively smaller centre crops. At these finer scales, RAWP, described in Section 4.6, is used to select candidate target patches that are better aligned with the source content. The final training objective combines the scene-, object-, and texture-level adversarial losses, as detailed in Section 4.7.

4.4 Scaled Illumination-Aware Attention Mechanism

4.4.1 Motivation

Training a single generator across multiple adverse conditions raises a fundamental question: what form of guidance remains consistently available and informative regardless of the specific degradation present in the input? Raw RGB statistics are insufficient for this purpose, since different adverse conditions may induce substantially different photometric distributions. For example, fog and snow can both produce bright regions, yet they differ in contrast, texture visibility, and spatial structure. Weather-condition labels

are also undesirable, as they may be unavailable at inference time and would require the model to rely on discrete condition supervision. Semantic masks are similarly unsuitable as inputs, since semantic segmentation is the downstream task that the enhancement module is intended to support.

Illumination intensity provides a more suitable condition-invariant signal. It can be computed directly from the input image without requiring an auxiliary model, varies continuously across the scene, and reflects an important aspect of the restoration problem. Dark regions often require enhancement to recover visibility, whereas sufficiently bright regions may require more conservative modification. However, in adverse weather driving scenes, brightness alone does not imply that a region is visually reliable. Fog and snow, for instance, often produce bright but degraded regions in which contrast, colour, and texture remain severely compromised. This motivates an attention mechanism that uses illumination as a shared guidance signal while avoiding the tendency to ignore bright weather-affected areas.

Prior work in low-light enhancement (Jiang et al., 2021) adopts the illumination-based attention formulation

$$\text{Att}_{ij} = 1 - I_{ij}, \quad (4.2)$$

where $I_{ij} \in [0, 1]$ denotes the normalised illumination intensity at pixel (i, j) . This formulation assigns high attention to dark pixels and low attention to bright pixels. While appropriate for conventional low-light enhancement, it is less suitable for all-weather restoration. In foggy and snowy scenes, bright low-contrast regions may require substantial enhancement, but they receive little attention under the linear formulation. This can lead to insufficient restoration in regions where weather-induced degradation is particularly pronounced.

4.4.2 Scaled Attention Formulation

The Scaled Illumination-Aware Attention Mechanism (SIAM) preserves the condition-invariant nature of illumination-based guidance while reshaping the attention distribution to better accommodate bright weather-affected regions. Given the naive attention map Att_{ij} in Eq. (4.2), SIAM defines the scaled attention as

$$\text{S_Att}_{ij} = -\text{Att}_{ij} \cdot (\text{Att}_{ij} - 2). \quad (4.3)$$

Equivalently, this corresponds to the quadratic mapping $f(x) = 2x - x^2$, where $x = \text{Att}_{ij} = 1 - I_{ij}$. The mapping preserves high attention for low-illumination regions while reducing the rate at which attention decreases as illumination increases. As a result, moderately bright regions continue to receive meaningful attention, rather than being prematurely suppressed as in the naive linear formulation.

The effect of this transformation is illustrated in Fig. 4.5. The naive attention curve

decreases linearly as illumination increases, assigning near-zero attention to bright regions. By contrast, the scaled attention curve maintains relatively high attention over a broader illumination range and decreases sharply only when the illumination is very high. This behaviour reflects the desired restoration prior: dark and moderately bright regions may both require enhancement, whereas only strongly illuminated regions, such as direct light sources or saturated highlights, should be modified conservatively.

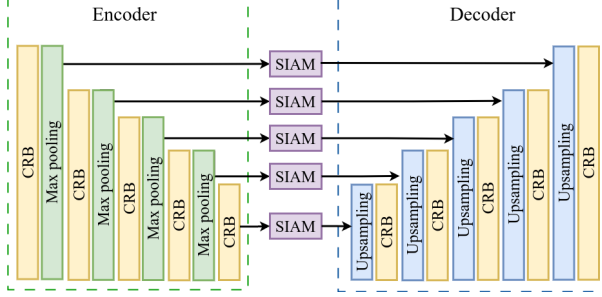


Figure 4.4: Integration of SIAM into the generator. The illumination-aware attention map is computed from the input image and injected at each scale of the U-Net generator to modulate intermediate feature activations.

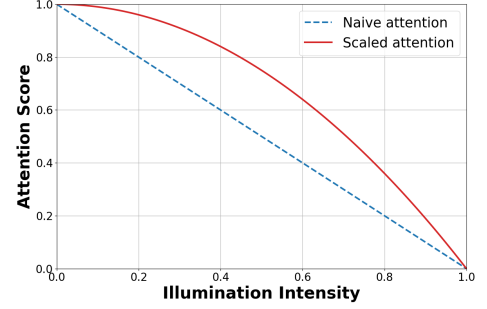


Figure 4.5: Comparison between naive illumination attention and the scaled attention used in SIAM. The scaled formulation preserves higher attention over a broader illumination range, reducing under-attention to bright weather-affected regions.

The attention map is computed once from the luminance channel of the input image and injected into the generator at each U-Net resolution, as shown in Fig. 4.4. At each resolution, the attention map is resized to match the spatial dimensions of the intermediate feature tensor and applied through element-wise modulation. In this way, SIAM guides the generator to allocate greater capacity to regions likely to require restoration while suppressing unnecessary modification in regions that are already visually reliable.

4.4.3 Qualitative Behaviour of SIAM

Fig. 4.6 compares the attention maps produced by the naive formulation and SIAM on representative adverse weather images. In foggy scenes, the naive formulation tends to concentrate attention on darker foreground regions while assigning limited attention to brighter distant regions affected by atmospheric scattering. SIAM retains meaningful attention in these distant regions, allowing the generator to address contrast attenuation and visibility loss. In snowy scenes, the naive formulation assigns low attention to bright snow-covered surfaces, which can limit the recovery of road texture and surface detail. SIAM instead assigns moderate attention across such regions, enabling more effective restoration. In nighttime scenes, where the dominant degradation is insufficient illumination, the scaled formulation behaves similarly to the naive formulation in low-illumination regions. This is desirable, since the original illumination-based attention is well matched

to conventional low-light enhancement.

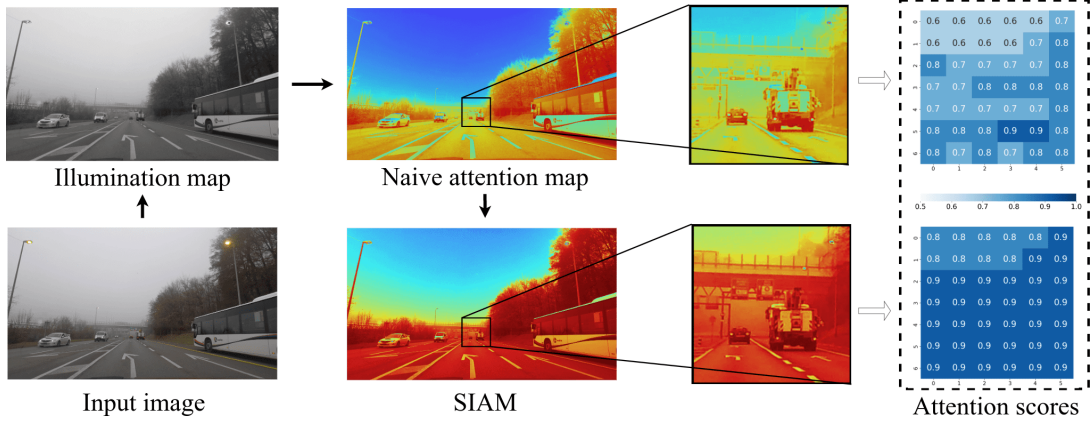


Figure 4.6: Qualitative comparison between naive illumination attention and SIAM. In foggy and snowy scenes, SIAM preserves attention in bright weather-affected regions that receive limited attention under the naive formulation. In nighttime scenes, both formulations assign high attention to dark regions, consistent with the low-light enhancement setting.

4.5 Hierarchical Discrimination Framework

A discriminator operating only on full-image or large scene-level patches primarily supervises global appearance. It can encourage plausible colour balance, illumination distribution, and coarse structure, but may provide insufficient constraints on fine-grained local detail. This limitation is particularly relevant in driving scenes, where road markings, traffic signs, vegetation boundaries, building textures, and object contours contain dense high-frequency information that is important for both perceptual quality and downstream recognition. Multi-scale discrimination addresses this limitation by supervising the generator at several spatial granularities. While prior work has explored two-level global-local discrimination (Jiang et al., 2021), our experiments in Section 4.10 indicate that two scales are insufficient to fully capture the structural and textural complexity of urban driving scenes.

We therefore introduces a three-level discrimination hierarchy consisting of scene-, object-, and texture-level supervision. Each level is associated with a distinct patch scale, discriminator, and target-patch selection strategy. The overall hierarchy is illustrated in Fig. 4.7.

4.5.1 Scene-Level Discriminator

The scene-level discriminator D_s evaluates a randomly cropped scene patch P^s from the enhanced image against the corresponding clear weather target patch P_T^s . Its role is to constrain global properties of the restored image, including colour distribution, illumination statistics, and coarse spatial composition. At this scale, the discriminator encourages

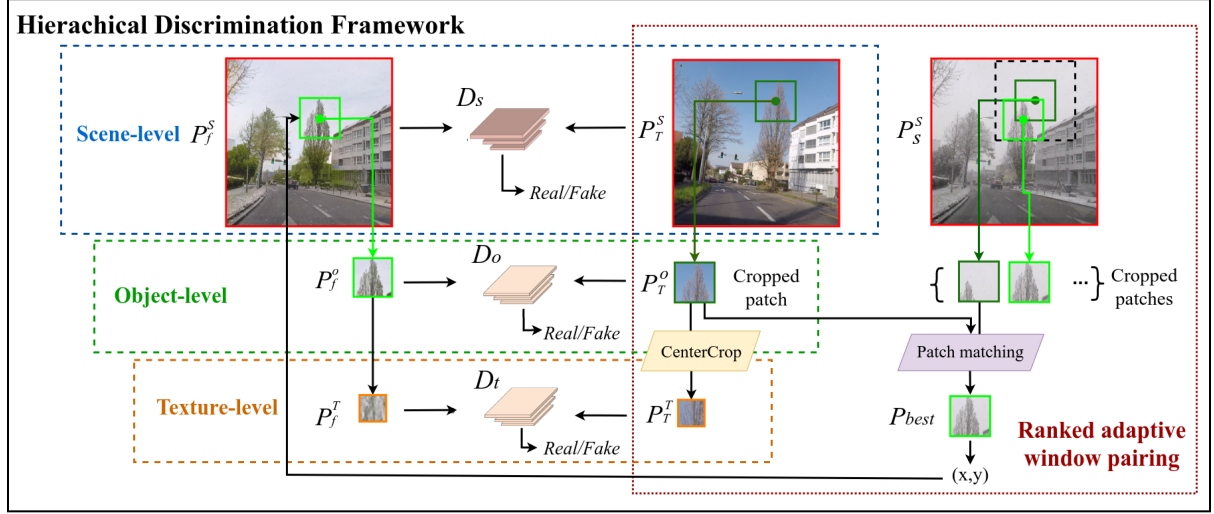


Figure 4.7: Hierarchical discrimination framework. Scene-level patches supervise global coherence; object-level patches, with one quarter of the scene-patch side length, supervise the structural fidelity of individual scene elements; texture-level patches, obtained by centre-cropping object-level patches, supervise fine-grained local surface detail.

the enhanced output to resemble a clear weather scene in its overall appearance. However, because its supervision is applied at a relatively coarse granularity, it may be less sensitive to local texture errors or small structural artefacts.

Scene-level discrimination is therefore effective for correcting global degradations such as colour cast, fog-induced grey veiling, and overall under-exposure in nighttime images. Nevertheless, it is insufficient on its own for enforcing local fidelity. An image may satisfy scene-level realism while still containing over-smoothed surfaces, distorted object details, or missing fine textures.

4.5.2 Object-Level Discriminator

The object-level discriminator D_o operates on patches P^o whose side length is one quarter of the scene-level patch size. At this spatial scale, patches often contain individual scene elements such as vehicles, pedestrians, traffic signs, lane markings, or local road regions. Supervision at this level targets errors that may be less visible in full-scene discrimination but are important for perception, including object-boundary distortions, local structural inconsistencies, and degraded semantic cues.

Patch selection at the object level requires additional care. A natural strategy is to crop source and target patches at the same spatial location. However, the source adverse weather image and the corresponding clear weather image are only loosely aligned. At the object scale, even small camera-pose or trajectory differences can result in mismatched content between the two patches, which may destabilise adversarial training. To address this issue, AllWeatherNet uses Ranked Adaptive Window Pairing, described in Section 4.6, to select better-matched target patches within a local search window.

4.5.3 Texture-Level Discriminator

The texture-level discriminator D_t operates on patches P^t obtained by centre-cropping object-level patches to a smaller spatial extent. At this scale, the patch no longer captures complete objects, but instead represents local surface appearance, such as road texture, vegetation detail, building facades, or other fine-grained material patterns. This supervision is designed to address a common limitation of coarse-scale enhancement: outputs may appear globally plausible while remaining locally over-smoothed or lacking natural high-frequency detail.

By enforcing realism at the texture level, D_t encourages the generator to recover fine local structures that are important for both perceptual fidelity and downstream dense prediction. This is particularly relevant in adverse weather scenes, where texture degradation often occurs together with contrast attenuation, motion blur, or particle occlusion.

4.5.4 Multi-Scale Synergy

The three discriminators provide complementary forms of supervision rather than redundant constraints. Scene-level discrimination encourages global photometric and compositional consistency; object-level discrimination constrains local structure and semantic detail; and texture-level discrimination promotes fine-grained surface realism. The ablation study in Section 4.10 confirms the cumulative contribution of these levels. Adding the object-level discriminator to the scene-level baseline increases SSIM; adding the texture-level discriminator further increases SSIM; and the complete system reaches the best. Although the individual gains are modest, they accumulate consistently, and the qualitative results in Fig. 4.8 show that the three levels address distinct failure modes.

4.6 Ranked Adaptive Window Pairing

4.6.1 The Misalignment Problem

Scene-level discrimination is relatively tolerant to spatial misalignment. As long as the source and target scene patches depict broadly similar content, such as an urban road segment, small viewpoint changes or local translations are unlikely to substantially alter the overall content distribution. At the object and texture levels, however, the same degree of misalignment can introduce significant training noise. For example, an object-level crop centred on a vehicle in the source image may correspond to a fence, road boundary, or background region in the target image if the same spatial coordinates are used directly. In this case, the discriminator is asked to compare patches that do not represent the same underlying content. Such mismatched supervision weakens adversarial training and may encourage the generator to alter scene content rather than restore

weather-degraded appearance.

This issue is not incidental in real driving datasets. In ACDC (Sakaridis et al., 2021), clear weather and adverse weather images of the same scene are captured during separate traversals and are aligned only approximately using GPS information. Consequently, even well-corresponding image pairs can exhibit local offsets at the pixel level. Strictly using co-located patches at the object scale therefore produces a training signal that is strongly affected by misalignment noise.

4.6.2 Adaptive Window Search

RAWP addresses this problem by searching within a local spatial window around the nominally co-located source patch and selecting the candidate patch that best matches the target. Let the nominal source patch location within the scene patch P^s be (x_0, y_0) , and let the patch size be $z \times z$. We define a search window of width w and height h centred at (x_0, y_0) . A $z \times z$ window is then slid across this search region with stride s , producing candidate patches P_{Sc}^o for $c = 1, \dots, C$, where $C = \left\lceil \frac{w-z}{s} \right\rceil \cdot \left\lceil \frac{h-z}{s} \right\rceil$.

4.6.3 Pairing Score and Best Patch Selection

For each candidate patch, we compute a pairing score with respect to the target patch. The score is defined as a per-pixel intensity difference:

$$F(P_{Sc}^o, P_T^o) = \sum_{i=0}^N \sum_{j=0}^M \left| I_{P_T^o}(x, y) - I_{P_{Sc}^o}(x + i \cdot s, y + j \cdot s) \right|, \quad (4.4)$$

where $N = \lceil (x + w - z)/s \rceil$ and $M = \lceil (y + h - z)/s \rceil$ are the horizontal and vertical step counts within the search window, and $I(\cdot, \cdot)$ denotes the RGB intensity at the indicated location. The best-matched source patch is selected as

$$P_{\text{best}} = \arg \min_{P \in W} F(P_{Sc}^o, P_T^o). \quad (4.5)$$

Once P_{best} is identified, the enhanced patch at the matched location is passed to the object-level discriminator. A centre crop of P_{best} and the target patch P_T^o is then used to provide texture-level supervision.

4.7 Loss Function

Training AllWeatherNet uses a relativistic adversarial objective (Jolicœur-Martineau, 2018) with least-squares GAN losses (Mao et al., 2017) at each of the three discrimination scales.

Scene-level losses. At the scene level, the relativistic discriminator compares the realism of the real and generated patches rather than evaluating either in isolation:

$$L_D^s = \mathbb{E}_{P_r^s \sim P_{\text{real}}} \left[(D_s(P_r^s, P_f^s) - 1)^2 \right] + \mathbb{E}_{P_f^s \sim P_{\text{fake}}} \left[D_s(P_f^s, P_r^s)^2 \right], \quad (4.6)$$

$$L_G^s = \mathbb{E}_{P_f^s \sim P_{\text{fake}}} \left[(D_s(P_f^s, P_r^s) - 1)^2 \right] + \mathbb{E}_{P_r^s \sim P_{\text{real}}} \left[D_s(P_r^s, P_f^s)^2 \right], \quad (4.7)$$

where P_r^s and P_f^s are real and fake scene patches, and $\mathbb{E}_{P \sim P_{\text{real}}}$ denotes expectation over the corresponding data distribution. The relativistic formulation penalises the generator for producing patches that are less realistic than real patches, a stronger signal than the absolute-realism objective of the standard LSGAN.

Object- and texture-level losses. At the object and texture levels, we use a standard LSGAN formulation without the relativistic comparison, because the object/texture patches are smaller and the mass-of-pixels interpretation of relativistic losses is less informative:

$$L_D^x = \mathbb{E}_{P_r^x \sim P_{\text{real}}^x} \left[(D_x(P_r^x) - 1)^2 \right] + \mathbb{E}_{P_f^x \sim P_{\text{fake}}^x} \left[(D_x(P_f^x) - 0)^2 \right], \quad (4.8)$$

$$L_G^x = \mathbb{E}_{P_f^x \sim P_{\text{fake}}^x} \left[(D_x(P_f^x) - 1)^2 \right], \quad (4.9)$$

with $x \in \{o, t\}$ indexing the level.

Total objective. Let $\mathcal{L} = \{s, o, t\}$. The total training loss is a weighted sum across the three scales,

$$\text{Total Loss} = \sum_{\ell \in \mathcal{L}} \Lambda_\ell \cdot (L_G^\ell + L_D^\ell), \quad (4.10)$$

with weights $\Lambda_s = \lambda_1$, $\Lambda_o = \lambda_2$, $\Lambda_t = \lambda_3$. The weights are set empirically such that the three scales contribute comparable loss magnitudes during the middle of training; in our experiments $\lambda_1 = \lambda_2 = \lambda_3 = 1$ gave robust results.

4.8 Experimental Setup

4.8.1 Datasets

Training uses 1,600 images from ACDC (Sakaridis et al., 2021), evenly distributed across the four adverse conditions (400 per condition: rain, snow, fog, night). For the night condition specifically, we additionally use 2,416 nighttime images from the Dark Zurich dataset (Sakaridis et al., 2020), which provides coarse GPS correspondences between nighttime, twilight, and daytime captures.

Evaluation of semantic segmentation is performed on the ACDC validation set (406 images) and on the Dark Zurich nighttime validation subset (50 images). Generalisation

is assessed on two datasets held out entirely from training: Foggy Zurich (Sakaridis et al., 2018) and Nighttime Driving (Dai and Van Gool, 2018).

4.8.2 Evaluation Metrics

We report four image-quality metrics and one downstream-task metric.

- **Peak Signal-to-Noise Ratio (PSNR)**(Fardo et al., 2016) measures pixel-level fidelity. Under imperfect pairs, PSNR is only weakly informative, because pixel-aligned comparison is not possible.
- **Structural Similarity Index Measure (SSIM)**(Hore and Ziou, 2010) measures structural similarity in luminance, contrast, and structure. Under loose pairing, SSIM is more informative than PSNR because it is less sensitive to small spatial displacements.
- **Natural Image Quality Evaluator (NIQE)**(Mittal et al., 2013) is a no-reference quality metric derived from natural-scene statistics. Lower NIQE indicates more natural appearance.
- **Learned Perceptual Image Patch Similarity (LPIPS)**(Zhang et al., 2018) measures perceptual distance in deep-feature space. Lower LPIPS indicates stronger perceptual similarity to the reference.
- **Mean Intersection over Union (mIoU)** measures the quality of semantic segmentation produced by a pre-trained PSPNet (Zhao et al., 2017) applied to the enhanced images. This is the primary metric for our purposes: it directly measures the downstream utility of enhancement.

4.8.3 Comparison Methods

We compare against three distinct categories of prior work:

1. **Weather effect removal:** WGWS-Net (Zhu et al., 2023), MultiAdverse (Chen et al., 2022b).
2. **Pixel-level translation:** CycleGAN (Zhu et al., 2017), CUT (Park et al., 2020), Santa (Xie et al., 2023).
3. **Low-light enhancement:** EnlightenGAN (Jiang et al., 2021), Zero-DCE (Guo et al., 2020), SCI (Ma et al., 2022b), PairLIE (Fu et al., 2023).

4.8.4 Implementation Details

The generator is a U-Net with skip connections, with SIAM injected at every resolution. The three discriminators share the PatchGAN architecture with different input patch sizes. Training uses Adam with learning rate 2×10^{-4} and the $(\beta_1, \beta_2) = (0.5, 0.999)$ hyperparameters standard for GAN training. Scene patches are 256×256 ; object patches are 64×64 ; texture patches are 16×16 . The RAWP search window has width and height $w = h = 32$, with stride $s = 4$. Training runs for 200 epochs on a single NVIDIA RTX 3090.

4.9 Results and Analysis

4.9.1 Image Quality on ACDC

Fig. 4.8 presents qualitative comparisons across all four adverse conditions. Weather effect removal methods (WGWS-Net, MultiAdverse) preserve the gross structure of the scene but retain a weather-tinted appearance; pixel-level translation methods (CycleGAN, CUT, Santa) produce bright daytime-like images but exhibit content hallucinations, especially in nighttime; low-light enhancement methods (SCI, PairLIE) over-expose bright regions in fog and snow. AllWeatherNet produces outputs that are uniformly brightened to daytime-like appearance, with colour and contrast corrected and without visible hallucinations.

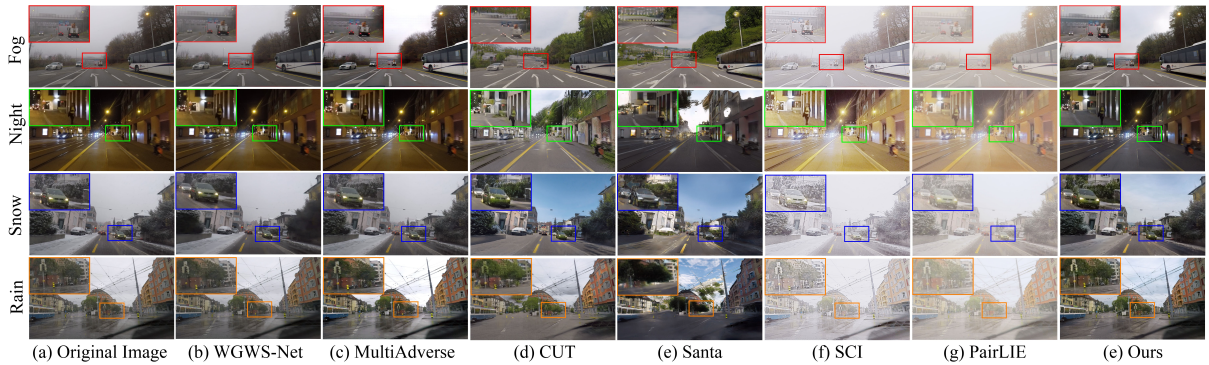


Figure 4.8: Qualitative comparison of AllWeatherNet against weather effect removal, pixel-level translation, and low-light enhancement baselines, under fog, snow, night, and rain. Zoom-in red regions highlight visual distinctions.

Quantitative results on ACDC are reported in Table 4.1. AllWeatherNet achieves the lowest NIQE (0.1257), indicating the most natural-looking output, and the highest SSIM (0.3983), indicating the best structural preservation. Although CycleGAN achieves higher PSNR, this reflects the limited informativeness of PSNR under loose pairing rather than genuinely better enhancement; CycleGAN’s NIQE (0.1578) and mIoU (33.6%) are substantially worse than AllWeatherNet’s, and its outputs contain the content hallucinations noted in Fig. 4.8.

Table 4.1: Quantitative evaluation of image quality and downstream semantic segmentation on the ACDC validation set. Best results are in **bold**.

Method Type	Models	Metrics				
		SSIM $\uparrow$	PSNR $\uparrow$	NIQE $\downarrow$	LPIPS $\downarrow$	mIoU $\uparrow$
Weather removal	WGWS-Net (Zhu et al., 2023)	0.3916	11.4812	0.1480	0.4740	36.4
	MultiAdverse (Chen et al., 2022b)	0.3822	10.9135	0.1782	0.4752	37.0
Pixel translation	CycleGAN (Zhu et al., 2017)	0.3981	12.2493	0.1578	0.4655	33.6
	CUT (Park et al., 2020)	0.3776	12.1043	0.1668	0.4833	29.3
	Santa (Xie et al., 2023)	0.3920	12.0863	0.1374	0.4770	25.7
Low-light	EnlightenGAN (Jiang et al., 2021)	0.3905	11.8725	0.1651	0.4649	37.2
	Zero-DCE (Guo et al., 2020)	0.3160	10.6015	0.2726	0.4428	32.4
	SCI (Ma et al., 2022b)	0.3239	8.5693	0.2600	0.5149	24.4
	PairLIE (Fu et al., 2023)	0.3577	8.9133	0.1564	0.4598	28.3
All-weather	Ours	0.3983	11.6618	0.1257	0.4619	38.2

As shown above, AllWeatherNet achieves 38.2%, ahead of every baseline. The next-best method (EnlightenGAN at 37.2%) is 1.0 percentage points behind; the gap from the uncorrected baseline (not shown in the table, corresponding to running PSPNet directly on the adverse weather images, which yields an mIoU of approximately 32.9%) is 5.3 percentage points. This improvement supports the primary claim of the chapter: a unified enhancement framework trained with condition-invariant supervision and hierarchical discrimination improves downstream perception.

4.9.2 Image Quality on Dark Zurich

On Dark Zurich, the nighttime-only benchmark, AllWeatherNet achieves the best SSIM (0.3849), the lowest NIQE (0.1850), and the lowest LPIPS (0.4589), while PSNR is narrowly below EnlightenGAN’s; mIoU is 17.6%, 6.6 points above the next-best method (Table 4.2). The large mIoU margin in this harder setting (the nighttime benchmark is more difficult than ACDC’s composite because the training distribution is heavily skewed towards daytime data) underscores the value of unified enhancement: the low-light-specific methods do not transfer well to the nighttime driving setting, while AllWeatherNet’s illumination-aware attention and hierarchical discrimination handle the setting natively.

Table 4.2: Quantitative evaluation on the Dark Zurich nighttime validation set. Best results are in **bold**.

Models	Metrics				
	SSIM $\uparrow$	PSNR $\uparrow$	NIQE $\downarrow$	LPIPS $\downarrow$	mIoU $\uparrow$
EnlightenGAN (Jiang et al., 2021)	0.3791	10.2460	0.3186	0.4766	10.8
Zero-DCE (Guo et al., 2020)	0.3324	8.6448	0.3425	0.5109	10.6
SCI (Ma et al., 2022b)	0.3519	8.5735	0.2554	0.5115	11.0
PairLIE (Fu et al., 2023)	0.3827	9.1442	0.2083	0.4844	7.1
Ours	0.3849	9.6121	0.1850	0.4589	17.6

4.9.3 Semantic Segmentation Qualitative Results

Fig. 4.9 presents the semantic segmentation predictions produced by PSPNet on the enhanced images. Compared with the other enhancement methods, our approach yields the best results with the highest miou. These qualitative results suggest that the proposed

method not only improves visual quality and scene visibility but also better preserves task-relevant semantic information, thereby benefiting downstream semantic segmentation.

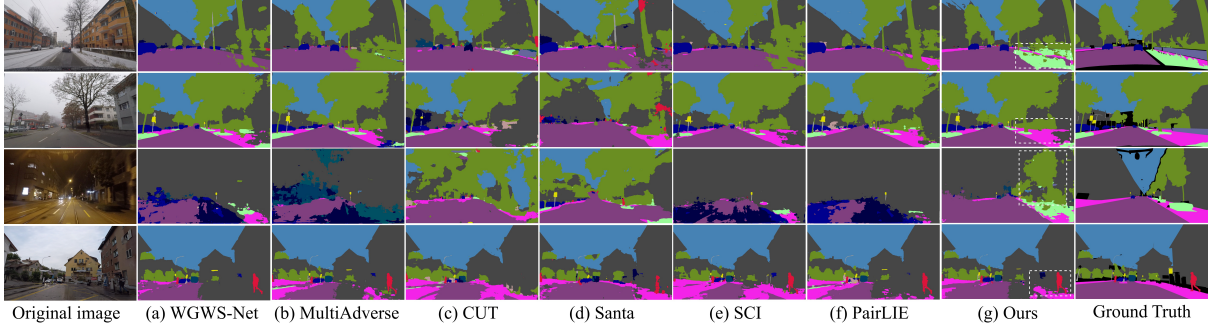


Figure 4.9: Semantic segmentation predictions with and without AllWeatherNet enhancement. Zoom-in white regions highlight improved separation of semantic classes.

4.9.4 Generalisation to Unseen Datasets

To assess whether AllWeatherNet’s improvements generalise beyond the ACDC and Dark Zurich training distribution, we apply the trained model to Foggy Zurich (Sakaridis et al., 2018) and Nighttime Driving (Dai and Van Gool, 2018), both of which are held out entirely from training. Qualitative results (Fig. 4.10) show that AllWeatherNet correctly corrects the yellow sodium-vapour cast of nighttime streetlights and reduces fog’s greying of distant objects in Foggy Zurich. Quantitatively (Table 4.3), mIoU improves by 1.8 percentage points on Foggy Zurich and 3.9 percentage points on Nighttime Driving. This demonstrates our method’s capability of generalising to unseen datasets.



Figure 4.10: Qualitative generalisation results on Foggy Zurich and Nighttime Driving, both held out from training. AllWeatherNet improves clarity of vehicles, signs, and buildings in fog, and corrects colour casts from artificial lighting at night.

Table 4.3: mIoU of PSPNet on original and AllWeatherNet-enhanced versions of the Foggy Zurich and Nighttime Driving test sets. Both datasets are held out from training.

Dataset	Original	Enhanced
Foggy Zurich (Sakaridis et al., 2018)	26.3	28.1
Nighttime Driving (Dai and Van Gool, 2018)	23.0	26.9

4.10 Ablation Studies

To isolate the contribution of each component of AllWeatherNet, we run a sequence of ablation experiments. We report SSIM on the ACDC validation set, because (as noted in Section 4.8.2) SSIM is the most informative image-quality metric under our loose-pair training regime.

4.10.1 Component Ablation

Table 4.4 shows the effect of progressively adding each component of AllWeatherNet to a baseline that uses only the scene-level discriminator. Three observations follow. First, each level of hierarchical discrimination contributes a positive increment: adding the object discriminator (+0.0015), adding the texture discriminator (+0.0031), and the two together (+0.0046) all improve quality. The texture-level contribution is the larger of the two, confirming that the finest-scale supervision catches failure modes that scene and object discriminators miss. Second, RAWP contributes an increment of similar magnitude to a discrimination level (+0.0012), confirming that well-aligned training pairs matter even when the alignment search is cheap and intensity-based. Third, SIAM contributes the largest single increment (+0.0061), more than any individual discrimination level. This reflects SIAM’s role as the guiding signal for the entire generator: it affects every pixel of every patch, whereas a given discrimination level affects only patches of its own scale.

Table 4.4: Ablation study on AllWeatherNet components. SSIM is reported on the ACDC validation set. Each row adds one component to the row above.

Model	Components					SSIM↑
	D_s	D_o	D_t	RAWP	SIAM	
M_1	✓					0.3864
M_2	✓	✓				0.3879
M_3	✓	✓	✓			0.3910
M_4	✓	✓	✓	✓		0.3922
M_5 (full)	✓	✓	✓	✓	✓	0.3983

4.10.2 Attention Mechanism Analysis

Fig. 4.11 compares naive illumination attention with SIAM across the four adverse conditions.

Under fog, the naive mechanism concentrates attention on the relatively dark road surface close to the vehicle, neglecting the veiled distant regions that most need enhancement. Under snow, the naive mechanism assigns near-zero attention to the bright snow surface, preventing recovery of road-marking cues. Under night and rain, where the dominant degradation is low illumination, the two mechanisms behave similarly. The outputs produced by the two models (Fig. 4.12) reflect this: the SIAM-trained model restores dis-

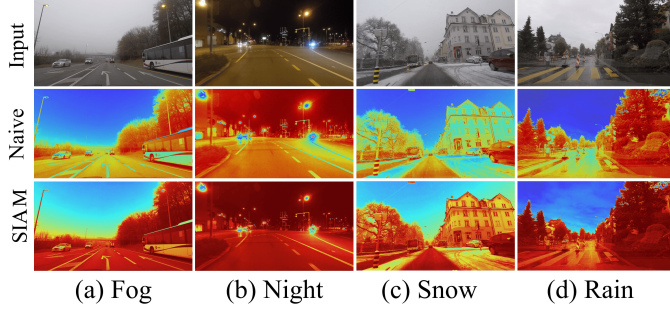


Figure 4.11: Naive attention versus SIAM attention maps across the four adverse conditions. Darker colours indicate higher attention scores. SIAM attends to weather-affected regions even when they are bright (fog, snow); naive attention does not.

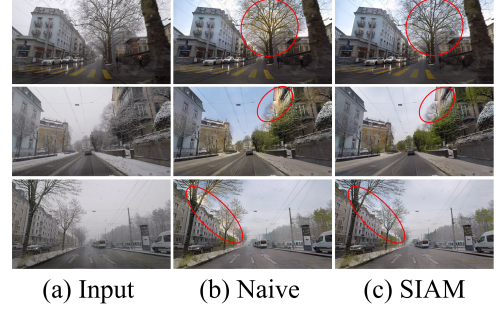


Figure 4.12: Enhanced outputs from models trained with naive attention and with SIAM. The SIAM-trained model restores distant foggy objects that the naive-attention model leaves veiled.

tant content in fog and sidewalk texture in snow, while the naive-attention model leaves these regions under-enhanced.

4.11 Summary

This chapter has presented AllWeatherNet, a unified image enhancement framework for adverse weather driving scenes. The method addresses four principal degradation conditions, namely rain, snow, fog, and nighttime imagery, within a single GAN-based model. Rather than synthesising the enhanced image directly, AllWeatherNet predicts residual enhancement masks and trains the generator using adversarial supervision at three spatial scales. The Scaled Illumination-Aware Attention Mechanism provides a condition-invariant guidance signal, the hierarchical discriminator constrains both global coherence and fine-grained detail, and the Ranked Adaptive Window Pairing strategy improves robustness to the imperfect correspondences available in real driving datasets. On ACDC, the resulting model achieves the best NIQE score of 0.1257 and SSIM of 0.3983, improves downstream semantic segmentation by 5.3 mIoU points over the uncorrected baseline, and further yields gains of 1.8 and 3.9 mIoU points on the held-out Foggy Zurich and Nighttime Driving datasets, respectively.

These results demonstrate that inference-time enhancement can improve weather-robust perception without modifying the downstream segmentation model. At the same time, this strategy remains subject to two inherent limitations. The first is a representation ceiling: an image-level enhancement method operates on a single 2D projection of a 3D scene and therefore cannot explicitly model the spatial structure of weather effects. The second is a data-distribution ceiling: improving degraded test-time inputs does not, by itself, expose the downstream segmentation model to the adverse weather distributions that were absent during training. These two limitations motivate the subsequent extensions developed in this thesis. The next chapter addresses the representation ceiling

while remaining within Part II on enhancement, whereas Part III later addresses the data-distribution ceiling through weather-aware generation.

Chapter 5

Reconstructing 3D Weather Scenes: WeatherGS

The preceding chapter examined the 2D formulation of enhancement through AllWeatherNet. Although it produces measurable improvements in downstream semantic segmentation, its operation remains confined to the image plane. An image is only a two-dimensional projection of an underlying three-dimensional scene and, by itself, does not provide an explicit geometric relationship to alternative viewpoints. Lens-level raindrops observed by one camera do not admit a principled correspondence to neighbouring views; falling snowflakes move through three-dimensional space rather than along a fixed image plane; and the depth-dependent transmission that characterises fog cannot be fully resolved from a single viewpoint.

This limitation is shared by image-level weather-restoration methods more broadly. In applications that require an explicit geometric scene representation, particularly those involving novel-view rendering or controlled scene modification, a purely 2D formulation is structurally insufficient. This chapter extends enhancement from 2D images to 3D scene reconstruction. The motivation is both methodological and practical. Closed-loop simulation has become an important tool for autonomous driving safety validation, and such simulation requires scene representations that can be queried from arbitrary viewpoints rather than replayed only as fixed image sequences. Since the fidelity of simulation is constrained by the fidelity of the underlying 3D reconstruction, robust reconstruction from weather-degraded observations is a necessary step towards simulation-ready scene assets.

Standard 3D reconstruction methods, including Neural Radiance Fields (NeRF) (Mildenhall et al., 2020) and 3D Gaussian Splatting (3DGS) (Kerbl et al., 2023), are typically formulated under the assumption that the input images are visually clean and geometrically consistent. When multi-view captures are degraded by adverse weather, this assumption is violated. NeRF’s continuous implicit representation tends to average view-inconsistent weather artefacts across observations, leading to blurred reconstructions. By contrast, the explicit representation of 3DGS can preserve high-frequency content, but may also

reconstruct dense weather particles, such as rain streaks and snowflakes, or persistent lens occlusions, such as water droplets on the cover glass, as if they were part of the physical scene geometry. Neither behaviour is desirable when the objective is to obtain a clean 3D scene asset.

This chapter therefore investigates whether a clean 3D scene representation can be reconstructed from multi-view images corrupted by a mixture of atmospheric and lens-level weather artefacts, and whether such reconstruction can be achieved at training and rendering costs suitable for driving-scale applications. I address this problem through **WeatherGS** (Qian et al., 2025a), a 3DGS-based framework designed specifically for weather-degraded multi-view inputs. WeatherGS adopts a dense-to-sparse preprocessing strategy that decomposes weather artefacts into two categories: dense, small-scale atmospheric particles, which are removed by a diffusion-based Atmospheric Effect Filter (AEF), and sparse, large-scale lens occlusions, which are detected by an attention-based Lens Effect Detector (LED). The preprocessed images and occlusion masks are then used in a mask-guided 3DGS optimisation procedure, enabling clean scene geometry to be reconstructed while corrupted regions are excluded from supervision.

Experiments are conducted on a benchmark comprising six synthetic scenes, formed by combining 3 Scenes with snow and rain, together with two real-world scenes. WeatherGS achieves the highest average PSNR and the lowest average LPIPS among five baselines. It also substantially improves computational efficiency, with training approximately $14\times$ faster and rendering approximately $1,600\times$ faster than a NeRF-based variant of the same pipeline. The material presented in this chapter was published at the IEEE International Conference on Robotics and Automation (ICRA) 2025 (Qian et al., 2025a).

The remainder of this chapter is organised as follows. Section 5.1 develops the motivation for 3D weather-aware reconstruction. Section 5.2 reviews the directly relevant prior work, and Section 5.3 introduces the 3D Gaussian Splatting preliminaries used throughout the chapter. Sections 5.4 to 5.6 present the three main components of WeatherGS. Section 5.7 describes the benchmark and training setup, while Sections 5.8 and 5.9 report the experimental results and ablation studies. Finally, Section 5.11 concludes the chapter, closes Part II on enhancement, and motivates the transition to Part III on generation through the 2D generation method introduced in Chapter 6.

5.1 Introduction and Motivation

3D scene reconstruction from multi-view imagery is a core capability in computer vision. Given a set of images captured from known or estimable camera poses, the objective is to recover a 3D scene representation from which novel viewpoints can be rendered. Such representations support a wide range of applications, including virtual reality content creation, cultural heritage preservation, robotic navigation, and, most relevant to this thesis, simulation environments for autonomous driving. In these settings, the recon-

structured scene is not merely a visual output, but a reusable asset that can be queried, modified, and rendered from viewpoints beyond those observed during capture.

Existing neural reconstruction is largely represented by two paradigms. Neural Radiance Fields (NeRF) (Mildenhall et al., 2020) model a scene as a continuous volumetric function optimised through differentiable volume rendering. 3D Gaussian Splatting (3DGS) (Kerbl et al., 2023) instead represents the scene explicitly as a set of anisotropic 3D Gaussian primitives rendered through differentiable tile-based rasterisation. Both approaches can produce photorealistic novel views, while 3DGS offers substantially improved efficiency in both training and rendering.

However, standard reconstruction pipelines typically assume that the multi-view training images are clean observations of the scene radiance. This assumption is violated when the input views are corrupted by adverse weather, such as rain, snow, fog, or lens droplets. Under such conditions, the reconstruction does not simply become less accurate; rather, the corruption is absorbed into the learned representation in representation-specific ways.

NeRF fails through averaging. NeRF’s continuous implicit representation tends to average observations that are inconsistent across views. A snowflake that appears in one image but not in another, or appears at different image locations across views, creates an ambiguous training signal for the implicit network. Since the model cannot simultaneously explain both the transient particle and its absence, it tends to distribute the inconsistent evidence over the reconstructed radiance field. This produces blurred artefacts that neither faithfully reconstruct the weather particle nor cleanly remove it from the scene. Similar behaviour occurs for rain streaks and other transient atmospheric effects. Lens-level occlusions, which are tied to a camera or lens surface rather than to the world coordinate frame, are likewise treated as view-inconsistent observations and may be averaged into faint haze or local blur.

3DGS fails through artefact baking. 3DGS exhibits a different failure mode. Its explicit primitive-based representation is highly effective at fitting high-frequency visual evidence, but this strength can become problematic when the input images contain weather artefacts. Dense atmospheric particles that appear with repeated spatial or statistical structure across views may be assigned explicit Gaussian primitives with their own positions, scales, colours, and opacities. Persistent lens occlusions can similarly be absorbed into the representation when they are repeatedly observed in the training images. As a result, the reconstructed model may contain rain streaks, snowflakes, or lens droplets as part of the scene representation, causing these artefacts to appear in rendered novel views.

Neither failure mode is acceptable when the goal is to obtain a clean 3D scene asset. Progress from NeRF to 3DGS therefore does not by itself solve the problem of weather-degraded reconstruction; rather, it changes the dominant failure mode from implicit

averaging to explicit artefact incorporation. Fig. 5.1 illustrates these behaviours and compares them with the clean reconstruction produced by WeatherGS.

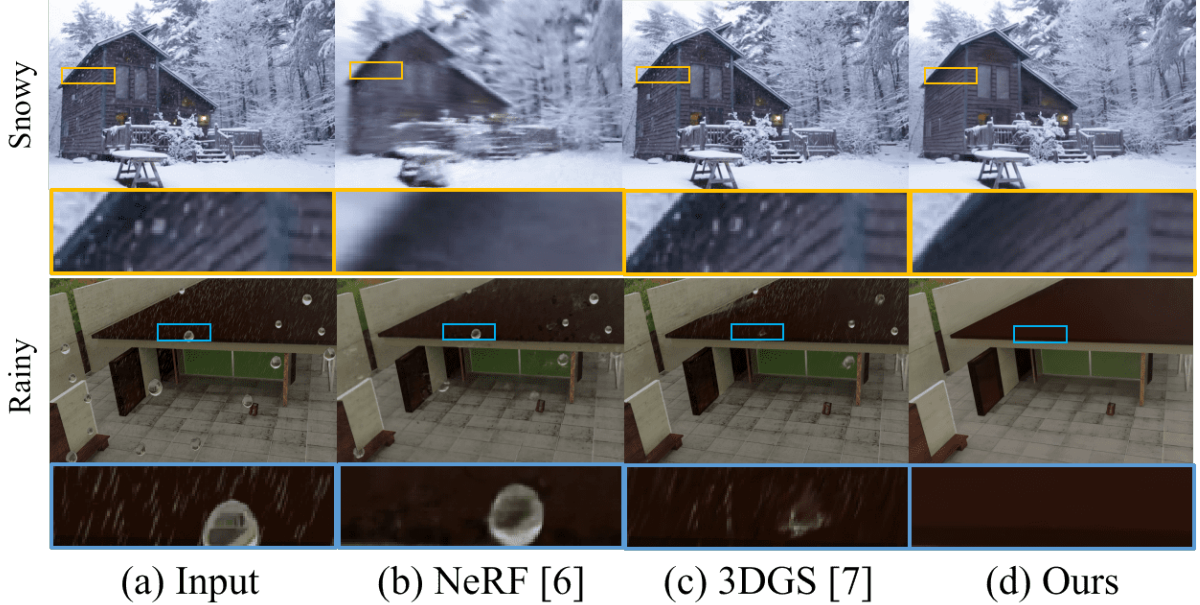


Figure 5.1: Rendering failures of 3D reconstruction methods under adverse weather. NeRF tends to average view-inconsistent weather artefacts, producing blurred regions and residual occlusions. 3DGS can instead incorporate weather particles and lens droplets into the explicit scene representation. WeatherGS mitigates these failure modes through dense-to-sparse preprocessing and produces cleaner renderings that better preserve the underlying scene.

5.2 Related Work

3D scene reconstruction. Recent advances in 3D scene reconstruction have enabled high-fidelity novel-view synthesis from posed multi-view images. NeRF (Mildenhall et al., 2020) represents a scene as a continuous radiance field learned through differentiable volume rendering, whereas 3DGS (Kerbl et al., 2023) uses an explicit set of anisotropic Gaussian primitives that can be optimised and rendered more efficiently. Building on these representations, subsequent work has explored editable 3D scenes (Chen et al., 2024b; Liu et al., 2024a; Ye et al., 2024), dynamic reconstruction (Guo et al., 2024b; Luiten et al., 2024; Yang et al., 2024d; Chen et al., 2023; Zhang et al., 2024), and generative 3D content creation (Ren et al., 2023; Voleti et al., 2024; Tang et al., 2024). For autonomous driving, OmniRe (Chen et al., 2025) demonstrates the use of 3DGS for reconstructing driving scenes, which also serves as the geometric foundation for WeatherEdit in the next chapter. Despite this progress, existing reconstruction pipelines are typically developed under the clean-input assumption. WeatherGS instead considers the adverse weather setting, where rain, snow, fog, and lens-level occlusions violate multi-view consistency and can be incorrectly absorbed into the reconstructed representation.

Weather effect removal in 2D. The 2D weather-removal literature includes methods for deraining (Yang et al., 2020a; Wang et al., 2022; Ren et al., 2019; Qian et al., 2018; Li et al., 2018a; Wen et al., 2023), dehazing (Berman et al., 2016; Zhang et al., 2019b, 2022; Zheng et al., 2023), desnowing (Liu et al., 2018; Zhang et al., 2021a; Wang et al., 2023b; Chen et al., 2020), and unified multi-weather restoration (Qian et al., 2024; Valanarasu et al., 2021; Li et al., 2020). This body of work is relevant to WeatherGS in two ways. First, the Atmospheric Effect Filter (AEF) adopts a diffusion-based restoration prior following Liu *et al.* (Liu et al., 2024b), thereby inheriting strengths of recent 2D restoration models for removing dense atmospheric artefacts. Second, the Lens Effect Detector (LED) builds on the attention mechanism of AttGAN (Qian et al., 2018), which was originally developed for 2D raindrop removal. In WeatherGS, however, these 2D modules are not presented as standalone restoration contributions. Their role is to provide weather-aware preprocessing and occlusion detection for a 3DGS reconstruction pipeline. The main contribution lies in how these components are composed with mask-guided 3D optimisation to recover clean scene geometry from degraded multi-view inputs.

Weather removal in 3D reconstruction. As discussed in Section 5.1, DerainNeRF (Li et al., 2024) is the closest prior work that explicitly addresses weather degradation in 3D reconstruction. It focuses on lens-droplet removal within a NeRF-based framework. WeatherGS differs in both scope and representation: it handles both dense atmospheric particles and sparse lens occlusions, and it is built on 3DGS rather than NeRF. This enables WeatherGS to achieve higher reconstruction quality while retaining the training and rendering efficiency associated with explicit Gaussian representations.

Related robust reconstruction methods. Several related works improve neural reconstruction under non-ideal imaging conditions. Low-light reconstruction methods (Wang et al., 2023a; Cui et al., 2024b; Zheng et al., 2024) address insufficient illumination; motion-blur reconstruction methods (Ma et al., 2022a; Zhao et al., 2024; Lee et al., 2023; Chen and Liu, 2024; Seiskari et al., 2024) compensate for camera or scene motion during exposure; and in-the-wild appearance modelling (Zhang et al., 2024) accounts for appearance variations across unconstrained image collections. These methods are relevant in that they relax the clean-input assumption of standard reconstruction pipelines. However, they do not address the combined presence of dense atmospheric particles and sparse lens-level occlusions that characterises adverse weather driving captures.

5.3 Preliminaries: 3D Gaussian Splatting

This section summarises the 3DGS formulation used throughout the chapter. A more detailed review is provided in Section 2.2.2; here, I focus on the components that are directly relevant to the WeatherGS reconstruction pipeline.

A 3DGS scene representation consists of a set of N 3D Gaussian primitives. Each primitive is parameterised by a centre $\boldsymbol{\mu} \in \mathbb{R}^3$, a covariance matrix $\boldsymbol{\Sigma} \in \mathbb{R}^{3 \times 3}$, an opacity $\alpha \in [0, 1]$, and a view-dependent colour $\mathbf{c}$ represented using spherical harmonics. The covariance matrix is typically factorised into rotation and scale components. The density contribution of a Gaussian at a 3D point $\mathbf{x}$ is defined as

$$G(\mathbf{x}) = \exp\left(-\frac{1}{2}(\mathbf{x} - \boldsymbol{\mu})^\top \boldsymbol{\Sigma}^{-1}(\mathbf{x} - \boldsymbol{\mu})\right). \quad (5.1)$$

To render an image, each 3D Gaussian is projected into the image plane using the camera viewing transformation W . Under an affine approximation of the local projective transformation, the projected 2D covariance is given by

$$\boldsymbol{\Sigma}' = JW\boldsymbol{\Sigma}W^\top J^\top, \quad (5.2)$$

where J denotes the Jacobian of the projection. The projected Gaussians are then sorted by depth and composited in front-to-back order through α -blending:

$$C = \sum_{n=1}^N c_n \alpha'_n \prod_{j=1}^{n-1} (1 - \alpha'_j). \quad (5.3)$$

The effective opacity of the n -th projected Gaussian at a 2D point $\mathbf{x}'$ is

$$\alpha'_n(\mathbf{x}') = \alpha_n \exp\left(-\frac{1}{2}(\mathbf{x}' - \boldsymbol{\mu}'_n)^\top \boldsymbol{\Sigma}'_n^{-1}(\mathbf{x}' - \boldsymbol{\mu}'_n)\right), \quad (5.4)$$

where $\boldsymbol{\mu}'_n$ is the projected centre.

During optimisation, the scene is rendered from each training viewpoint and compared against the corresponding ground-truth image using a photometric reconstruction loss. Gradients are back-propagated to update the Gaussian parameters. The representation is further adapted through densification, which clones or splits Gaussians in under-reconstructed regions, and pruning, which removes primitives with negligible opacity.

In this work, I am not aiming at modifying the Gaussian definition, projection model, compositing rule, or effective-opacity formulation in Eqs. (5.1) to (5.4). Instead, I modify the inputs and supervision used during optimisation. First, the training images are processed by the Atmospheric Effect Filter to remove dense atmospheric particles. Second, the photometric loss is masked to exclude regions affected by lens-level occlusions, as detected by the Lens Effect Detector. The following sections describe these two preprocessing modules and the resulting mask-guided 3DGS reconstruction procedure.

5.4 Atmospheric Effect Filter

5.4.1 Motivation

Dense atmospheric artefacts, such as rain streaks and airborne snowflakes, are characterised by high spatial density and small per-particle spatial support. They can appear across large portions of an outdoor image, while each individual streak or flake occupies only a small number of pixels. In multi-view captures, their structure is neither fully view-consistent nor fully random. At the scene level, images captured under the same weather event may share similar global particle statistics. At the particle level, however, individual droplets or snowflakes are transient and may occupy different positions across frames and viewpoints.

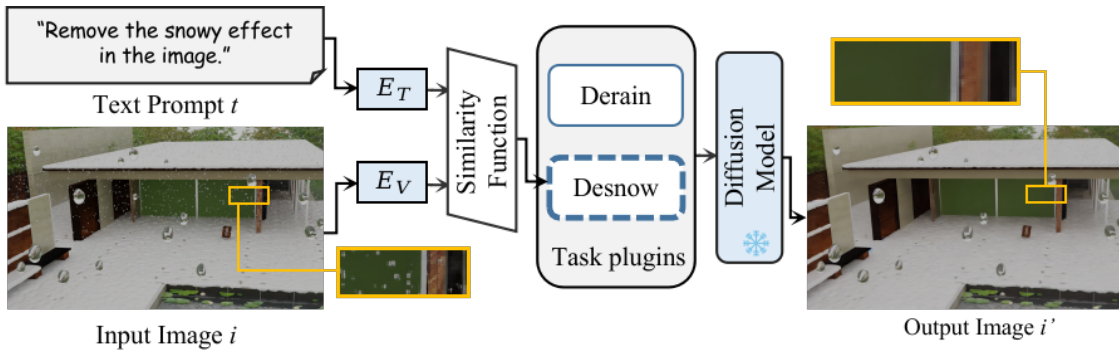


Figure 5.2: Atmospheric Effect Filter. A weather-degraded input image and a text prompt, such as “Remove the snowy effect in the image”, are encoded using CLIP text and vision encoders. A cosine-similarity criterion selects the appropriate restoration plugin, such as “Desnow”, which conditions a pre-trained Stable Diffusion model to remove dense atmospheric particles while preserving the underlying scene content.

For 3D reconstruction, these dense atmospheric artefacts introduce substantial noise into the photometric supervision. They affect a large fraction of pixels in each training view and, due to their partially consistent statistical structure, may not be eliminated by the densification and pruning mechanisms of 3DGS. Instead, they can be absorbed into the reconstructed representation as spurious geometry or opacity. The Atmospheric Effect Filter (AEF) is therefore introduced as a 2D preprocessing module that removes dense atmospheric particles before 3DGS optimisation.

5.4.2 Diffusion-Based Particle Removal

Diffusion models provide strong image priors and have shown substantial capability in restoring images degraded by noise, corruption, or adverse visual effects. However, as discussed in Section 2.1.3, applying generic diffusion restoration to weather-degraded images is non-trivial. The stochastic nature of diffusion sampling can modify scene content beyond the intended weather artefact, which is undesirable when the restored images are used as supervision for 3D reconstruction. To mitigate this issue, Liu *et al.* (Liu et al.,

2024b) introduce task-specific restoration plugins that condition the diffusion process according to the desired degradation-removal task, such as deraining or desnowing. The task is selected using CLIP-based similarity between the input image and a text prompt.

I adopt this plugin-based restoration strategy as the Atmospheric Effect Filter. Given a weather-degraded input image i and a text prompt t , such as “Remove the snowy effect in the image”, AEF encodes the image and text using the CLIP vision encoder E_V and text encoder E_T (Radford et al., 2021). The cosine similarity between the encoded features is then used to select the appropriate restoration plugin:

$$\mathcal{D}^* = \begin{cases} \mathcal{D}_{\text{derain}}, & \text{if } \cos(E_T(t), E_V(i)) > \theta, \\ \mathcal{D}_{\text{desnow}}, & \text{otherwise,} \end{cases} \quad (5.5)$$

where θ is a threshold tuned on held-out validation data. The selected plugin consists of ResNet (He et al., 2016) and cross-attention layers trained on weather-restoration pairs. It conditions a pre-trained Stable Diffusion model (Rombach et al., 2021) operating in the VAE-encoded latent space of the input image (Kingma, 2013). During inference, the diffusion model iteratively denoises the latent representation under the selected plugin conditioning, producing a restored image i' in which dense atmospheric artefacts are suppressed while the underlying scene structure is preserved, as illustrated in Fig. 5.2.

5.5 Lens Effect Detector

5.5.1 Motivation

Sparse lens occlusions, such as water droplets on the camera cover glass or windshield and patches of adhered snow, exhibit a different structure from dense atmospheric particles. They typically occupy a larger spatial extent, persist across consecutive frames from the same camera, and are specific to a particular optical path. Unlike airborne rain streaks or snowflakes, they are not part of the 3D scene and do not appear consistently across neighbouring camera views. The Atmospheric Effect Filter is not designed to handle such occlusions reliably. Since lens occlusions can cover substantial image regions, removing them would require inpainting the underlying scene content, which may introduce hallucinated structures and thereby contaminate the subsequent 3D reconstruction.

For this reason, the Lens Effect Detector (LED) does not attempt to remove lens occlusions in the image plane. Instead, it detects occluded regions and generates binary masks that exclude them from the 3DGS photometric loss. From the perspective of multi-view reconstruction, these pixels can be treated as unobserved in the affected training views. The underlying scene content can still be reconstructed from other views in which the same region is not occluded, or in which the occlusion appears at a different image location.

5.5.2 AttGAN-Based Detection

WeatherGS adopts the detection module from AttGAN (Qian et al., 2018), which was originally developed for 2D raindrop removal. The attention module produces a per-pixel confidence map $C(x, y) \in [0, 1]$, where higher values indicate a higher likelihood that pixel (x, y) is affected by a lens-level occlusion. The module is pre-trained on paired raindrop-removal data and can be applied directly to input images without requiring text-prompt conditioning.

The LED is applied after the Atmospheric Effect Filter rather than to the raw weather-degraded images. This ordering is important. Once dense atmospheric particles have been suppressed by the AEF, the remaining salient weather artefacts are primarily sparse lens occlusions, which are the target of the LED. Applying the detector directly to raw inputs would increase the risk of false positives on rain streaks or airborne snowflakes. The dense-to-sparse ordering of AEF followed by LED is therefore a deliberate design choice that improves the reliability of occlusion detection.

5.5.3 Binary Mask Generation

Given the confidence map, I derive a binary occlusion mask as

$$M(x, y) = \mathbb{I}(C(x, y) \geq t), \quad (5.6)$$

where t is an empirically selected threshold. Pixels with $M(x, y) = 1$ are marked as occluded and excluded from the photometric reconstruction loss. A morphological dilation operation is further applied to the binary mask to cover the feathered boundaries of droplets or adhered snow patches. The resulting mask is passed to the mask-guided reconstruction stage.

5.6 Mask-Guided 3D Reconstruction

5.6.1 Masked Loss

After AEF preprocessing and LED mask generation, WeatherGS follows the standard 3DGS optimisation pipeline with one modification: the photometric loss is evaluated only on unmasked pixels. Let I denote the AEF-processed training image, $\hat{I}$ the image rendered by 3DGS from the same camera viewpoint, and $\mathbf{M}$ the LED-generated occlusion mask. The masked L_1 loss is defined as

$$\mathcal{L}_1 = \sum_{\mathbf{t}} \left\| \left(\hat{I}(\mathbf{t}) - I(\mathbf{t}) \right) \circ (\mathbf{1} - \mathbf{M}) \right\|^2, \quad (5.7)$$

where $\circ$ denotes element-wise multiplication. The masked D-SSIM loss is

$$\mathcal{L}_{\text{D-SSIM}} = 1 - \text{SSIM}(\hat{I} \circ (\mathbf{1} - \mathbf{M}), I \circ (\mathbf{1} - \mathbf{M})) . \quad (5.8)$$

The total reconstruction objective follows the standard 3DGS weighted combination:

$$\mathcal{L} = (1 - \lambda) \mathcal{L}_1 + \lambda \mathcal{L}_{\text{D-SSIM}}, \quad (5.9)$$

with $\lambda = 0.2$, as in the original 3DGS formulation.

5.6.2 Initialisation and Training Details

Following the standard 3DGS protocol, Gaussian primitives are initialised from a Structure-from-Motion (SfM) reconstruction (Schonberger and Frahm, 2016) of the raw input views. I use the raw views for SfM because they preserve the original camera-pose information and geometric cues required for feature matching. The AEF-processed images are instead used for the photometric reconstruction loss, which reduces the influence of dense atmospheric particles during optimisation. This design avoids introducing diffusion-generated artefacts into the SfM initialisation while still benefiting from cleaned images during photometric training. Densification, pruning, and the remaining 3DGS optimisation procedures are inherited unchanged. Training is performed for 30,000 iterations on a single NVIDIA RTX 3090 GPU.

5.6.3 Dense-to-Sparse Integration

WeatherGS integrates three components into a single dense-to-sparse reconstruction pipeline: the Atmospheric Effect Filter removes dense atmospheric particles, the Lens Effect Detector identifies sparse lens occlusions, and mask-guided 3DGS reconstructs the final clean scene representation. The full pipeline is illustrated in Fig. 5.3.

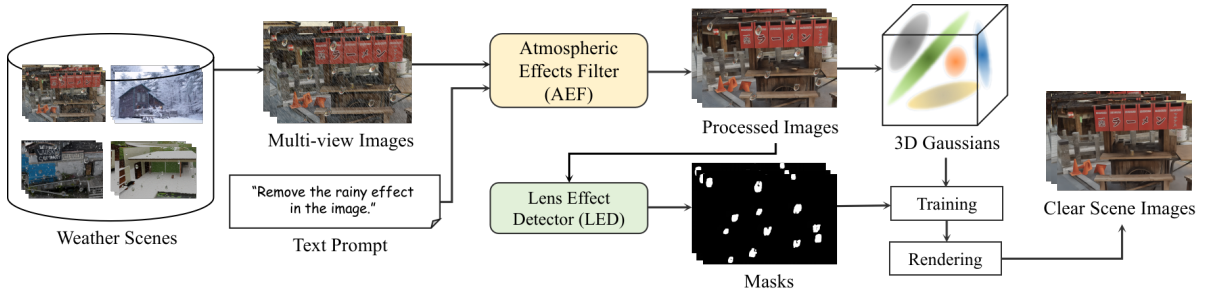


Figure 5.3: Overview of WeatherGS. The Atmospheric Effect Filter first removes dense atmospheric particles. The Lens Effect Detector then operates on the cleaned images to identify sparse lens occlusions. The resulting images and masks are used in a mask-guided 3D Gaussian Splatting optimisation procedure to reconstruct a clean 3D scene representation.

5.6.4 Multi-View Redundancy and the Limits of Mask-Guided Optimisation

The masked photometric losses in Eqs. (5.7) and (5.8) rely on a multi-view redundancy assumption: a scene region that is occluded in one training view is expected to be visible in a sufficient number of other views. When this assumption holds, masking has limited adverse effect on reconstruction, since the optimiser can recover the underlying scene structure from unmasked observations. When the assumption fails, for example if a region is occluded by lens droplets in all views that observe it, the reconstruction has no direct photometric signal for that region. In this case, the optimiser must rely on neighbouring observations and the inductive biases of the 3DGS densification process, which may lead to over-smoothed or incomplete reconstruction.

This limitation can be described through a simple probabilistic argument. Let ρ denote the average masked fraction across views, and let a scene region be observed in K training views. Under an independence assumption, the probability that the region is masked in all K views is approximately ρ^K . For example, if $\rho = 0.1$ and $K = 10$, this probability is 10^{-10} , indicating that complete loss of supervision is unlikely. In a more severe case with $\rho = 0.5$ and $K = 4$, the probability increases to 0.0625, which is large enough to produce visible reconstruction gaps. The synthetic and real benchmarks considered in Section 5.7 remain well below this severe-occlusion regime, but the analysis clarifies the conditions under which mask-guided optimisation may become unreliable.

The mask-guided formulation also interacts with SfM initialisation. SfM relies on feature correspondences across views. A feature that is occluded in some views but visible in others can still contribute to pose estimation, whereas a feature consistently occluded across views provides no reliable constraint. WeatherGS performs SfM on the raw unmasked inputs, accepting that some feature matches may be influenced by lens artefacts, and then uses the AEF-processed images together with LED masks for photometric optimisation. Running SfM on AEF-processed images would avoid some weather artefacts but could introduce diffusion-generated content that is not geometrically consistent across views. The adopted design, raw inputs for SfM and cleaned masked inputs for photometric training, therefore reduces the propagation of preprocessing artefacts into the reconstructed geometry.

5.7 Benchmark and Experimental Setup

5.7.1 A New Benchmark for Weather-Degraded 3D Reconstruction

Prior work on weather-aware 3D reconstruction has not established a common benchmark. I construct one by combining synthetic and real-world scenes.

Synthetic scenes. I adopt three scenes from Deblur-NeRF (Ma et al., 2022a) (*Tanabata*, *Factory*, and *Pool*) and modify them in Blender (Hess, 2010) to add weather. Snow is added through Blender’s snow material add-on for surface accumulation and a physics-simulated particle system for airborne flakes; rain is modelled through a streak-emitting particle system. Motion blur during rendering adds camera-side realism, and OpenCV post-processing densifies the weather effects. The result is six synthetic scene variants: Tanabata-snow, Tanabata-rain, Factory-snow, Factory-rain, Pool-snow, Pool-rain. Each scene provides both weather-degraded training views and weather-free ground-truth reference views for quantitative evaluation.

Real-world scenes. I construct two real-world scenes by extracting keyframes from publicly available videos captured in snowy and rainy weather. For these, no weather-free ground truth is available, and evaluation is qualitative only.

5.7.2 Baselines

I compare against four state-of-the-art 3D reconstruction baselines:

- **NeRF** (Mildenhall et al., 2020): the original implicit volumetric representation.
- **DerainNeRF** (Li et al., 2024): the closest prior work on weather-aware NeRF, targeting lens droplets.
- **3DGS** (Kerbl et al., 2023): vanilla 3D Gaussian Splatting without weather-specific preprocessing.
- **GS-W** (Zhang et al., 2024): a 3DGS variant designed for in-the-wild appearance variation, trained with appearance embedding.

5.7.3 Metrics

I report three standard reconstruction metrics: Peak Signal-to-Noise Ratio (PSNR), Structural Similarity Index Measure (SSIM), and Learned Perceptual Image Patch Similarity (LPIPS) (Zhang et al., 2018). Higher PSNR and SSIM, and lower LPIPS, indicate better reconstruction quality. For real-world scenes without ground truth, I report only qualitative comparisons.

5.8 Results

5.8.1 Quantitative Comparison

Tables 5.1 and 5.2 report per-scene and average PSNR, SSIM, and LPIPS on the snowy and rainy benchmark scenes.

Table 5.1: PSNR, SSIM, and LPIPS on snowy scenes.

Model	Tanabata-snow			Pool-snow			Factory-snow			Average		
	PSNR $\uparrow$	SSIM $\uparrow$	LPIPS $\downarrow$	PSNR $\uparrow$	SSIM $\uparrow$	LPIPS $\downarrow$	PSNR $\uparrow$	SSIM $\uparrow$	LPIPS $\downarrow$	PSNR $\uparrow$	SSIM $\uparrow$	LPIPS $\downarrow$
NeRF (Mildenhall et al., 2020)	21.377	0.729	0.215	21.820	0.731	0.376	21.666	0.612	0.383	21.621	0.691	0.325
DerainNeRF (Li et al., 2024)	23.428	0.763	0.175	23.382	0.752	0.363	23.381	0.652	0.335	23.397	0.722	0.291
3DGS (Kerbl et al., 2023)	22.589	0.836	0.144	26.431	0.860	0.145	20.860	0.768	0.234	23.293	0.821	0.174
GS-W (Zhang et al., 2024)	20.856	0.793	0.200	25.848	0.863	0.159	21.897	0.761	0.248	22.867	0.806	0.202
WeatherGS (Ours)	23.625	0.796	0.143	27.596	0.864	0.137	23.977	0.700	0.221	25.066	0.787	0.167

Table 5.2: PSNR, SSIM, and LPIPS on rainy scenes.

Model	Tanabata-rain			Pool-rain			Factory-rain			Average		
	PSNR $\uparrow$	SSIM $\uparrow$	LPIPS $\downarrow$	PSNR $\uparrow$	SSIM $\uparrow$	LPIPS $\downarrow$	PSNR $\uparrow$	SSIM $\uparrow$	LPIPS $\downarrow$	PSNR $\uparrow$	SSIM $\uparrow$	LPIPS $\downarrow$
NeRF (Mildenhall et al., 2020)	20.860	0.564	0.370	25.324	0.771	0.183	14.305	0.276	0.543	20.163	0.537	0.365
DerainNeRF (Li et al., 2024)	22.342	0.602	0.326	18.934	0.581	0.401	14.285	0.273	0.540	18.520	0.485	0.422
3DGS (Kerbl et al., 2023)	21.333	0.609	0.338	26.994	0.817	0.183	21.406	0.601	0.329	23.244	0.676	0.283
GS-W (Zhang et al., 2024)	21.185	0.537	0.402	23.255	0.786	0.191	21.218	0.607	0.366	21.886	0.643	0.319
WeatherGS (Ours)	23.623	0.628	0.232	24.370	0.805	0.140	23.016	0.620	0.218	23.670	0.684	0.197

Several observations can be drawn from the quantitative results. First, WeatherGS achieves the highest average PSNR under both snowy and rainy conditions, reaching 25.066 on snow and 23.670 on rain. These results indicate that the proposed dense-to-sparse preprocessing and mask-guided reconstruction strategy improves pixel-level reconstruction fidelity relative to both NeRF- and 3DGS-based baselines. Second, WeatherGS obtains the lowest average LPIPS in both settings, suggesting that the reconstructed views are perceptually closer to the clean reference images. This is particularly important for adverse weather reconstruction, where residual particles and lens artefacts may occupy only a small fraction of pixels but can strongly affect perceived rendering quality.

The SSIM results show a more nuanced trend. Vanilla 3DGS achieves the highest average SSIM on snowy scenes and outperforms WeatherGS on several individual cases. This behaviour is consistent with the sensitivity of SSIM to local structural correspondence. Since the AEF removes weather particles through diffusion-based restoration, it may introduce subtle texture variations in regions where particles previously occluded the scene. These variations can reduce SSIM even when the resulting reconstruction is visually cleaner and less contaminated by weather artefacts. LPIPS, which compares images in a learned feature space, better reflects this perceptual improvement and consistently favours WeatherGS across both snow and rain.

The main exception occurs in the Pool-rain scene, where vanilla 3DGS achieves higher PSNR and SSIM than WeatherGS. This scene contains relatively light rain and no substantial lens-level occlusion, so the benefits of preprocessing are limited. In this regime, the AEF acts closer to a mild restoration or near-identity operation, and any diffusion-induced texture modification can slightly reduce low-level similarity to the reference. This result identifies an expected boundary case of the method: WeatherGS provides the largest benefits when weather artefacts are sufficiently dense or occlusive to corrupt 3D reconstruction, while its advantage diminishes when the input degradation is already mild.

DerainNeRF performs poorly on rainy scenes, with an average PSNR of 18.520. This

reflects the mismatch between its design objective and the benchmark setting. Derain-NeRF is specialised for lens-droplet degradation, whereas several rainy benchmark scenes are dominated by atmospheric rain streaks rather than sparse lens occlusions. In such cases, its droplet-oriented modelling provides limited corrective signal, and the underlying NeRF representation remains vulnerable to the averaging artefacts discussed in Section 5.1.

5.8.2 Qualitative Comparison

Fig. 5.4 shows rendered views from each method on representative synthetic and real-world scenes.

The qualitative differences are stark. NeRF and DerainNeRF produce fragmented, over-smoothed reconstructions in both synthetic and real scenes; the snow-capped house in the real snowy scene is recognisable through NeRF but texture details are blurred to the point of unrecognisability. Vanilla 3DGS and GS-W produce sharper reconstructions but contain visible weather particles (snowflakes, rain streaks) and lens droplets, as if those artefacts were part of the scene. WeatherGS produces renderings that are both sharp (inherited from 3DGS) and clean (the artefacts are gone).

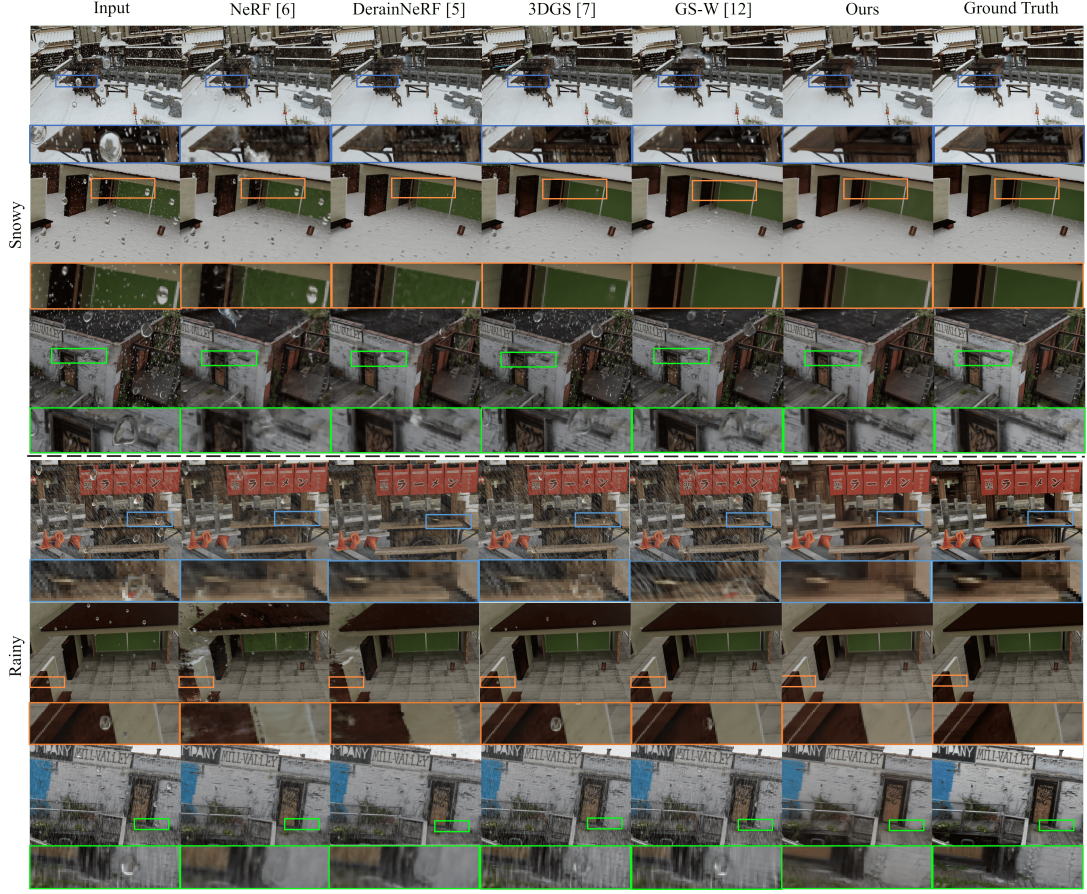
5.8.3 Training and Rendering Cost

A key practical claim of WeatherGS is that its efficiency matches vanilla 3DGS: the preprocessing is a one-time cost, and 3DGS optimisation itself is unchanged. To substantiate this claim, I compare WeatherGS (3DGS backend) against a NeRF-backed variant of the same preprocessing pipeline (Table 5.3).

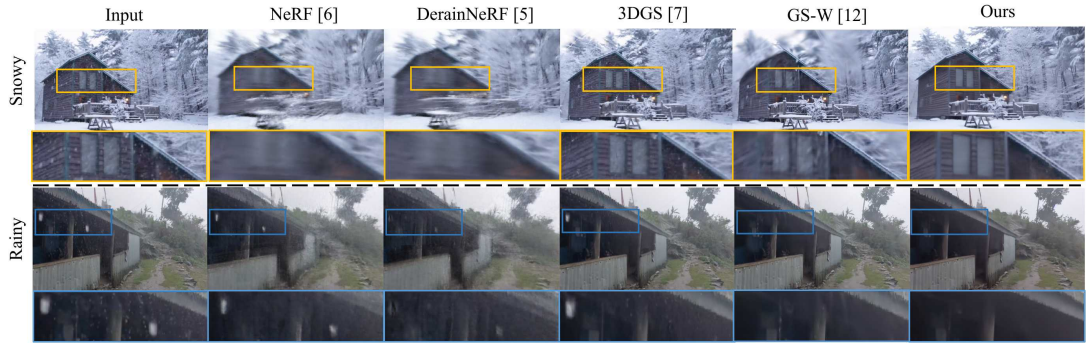
Table 5.3: Training and rendering cost comparison. Metrics are averaged across synthetic scenes. The 3DGS backend is approximately $14\times$ faster to train and $1,600\times$ faster to render than the NeRF variant, with better reconstruction quality.

Model	Image Quality			Computational Performance		
	PSNR	SSIM	LPIPS	GPU Mem.	Training	Render
Ours-NeRF	22.288	0.648	0.363	7.8 GB	~ 395 min	32.3 sec
Ours-3DGS	24.627	0.740	0.181	1.1 GB	~ 28 min	0.02 sec

The 3DGS backend is approximately $14\times$ faster to train (28 minutes versus 395 minutes), $1,600\times$ faster to render (0.02 seconds versus 32.3 seconds), and uses approximately $7\times$ less GPU memory (1.1 GB versus 7.8 GB). It also produces better reconstruction quality by every metric. The efficiency argument for moving from NeRF to 3DGS in the standard case is well known; what WeatherGS demonstrates is that the efficiency gap persists in the weather-robust setting once the AEF and LED are in place. A driving-scale scene with hundreds of training views can be reconstructed in under an hour with WeatherGS and rendered at 50 frames per second, which is not far from the operating envelope of real-time simulation.



(a) Synthetic scenes.



(b) Real-world scenes.

Figure 5.4: Qualitative results on synthetic (a) and real-world (b) scenes. NeRF-based methods show the characteristic blurring pattern; 3DGS methods reconstruct weather particles and droplets; WeatherGS produces clean renderings.

5.9 Ablation Studies

5.9.1 Contribution of AEF and LED

I evaluate the contribution of each preprocessing module by removing one at a time (Table 5.4, Fig. 5.5). Without the AEF, the pipeline still attempts to mask lens occlusions but has no mechanism for atmospheric particles; the reconstruction contains visible streaks and flakes. Without the LED, the pipeline has clean atmospheric content but reconstructs lens droplets as scene geometry. Only the full system delivers clean reconstructions on both fronts.

Table 5.4: Ablation of AEF and LED on the full benchmark. Metrics averaged across all synthetic scenes.

Model	PSNR $\uparrow$	SSIM $\uparrow$	LPIPS $\downarrow$
w/o AEF	24.366	0.7452	0.2049
w/o LED	23.946	0.7239	0.1986
Ours (Full)	24.627	0.7395	0.1808

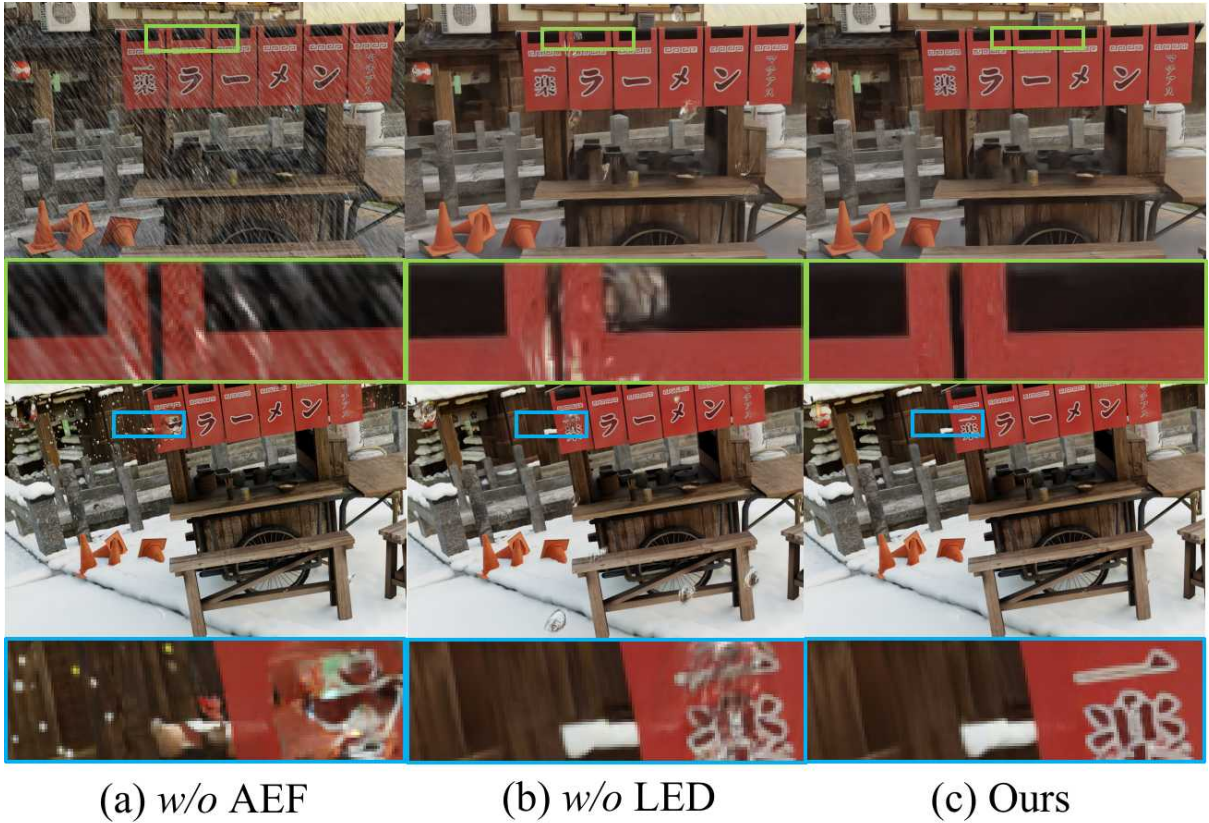


Figure 5.5: Ablation of the AEF and LED modules. Without AEF: visible weather particles. Without LED: visible lens droplets. Full system: clean reconstruction.

The dense-to-sparse ordering is also informative. Running the LED before the AEF produces substantially worse masks, because the LED’s attention module is distracted by the dense atmospheric particles. I report only the correct ordering in the main com-

parison; the reversed ordering produces results close to the baseline vanilla 3DGS.

5.9.2 Backbone Comparison: NeRF vs 3DGS

Table 5.3 and Fig. 5.6 compare the NeRF and 3DGS variants of my pipeline. The NeRF variant is noticeably blurrier, reflecting NeRF’s continuous-representation limitations in the presence of any residual inconsistency after AEF preprocessing. The 3DGS variant produces sharper textures with correctly reconstructed fine details.

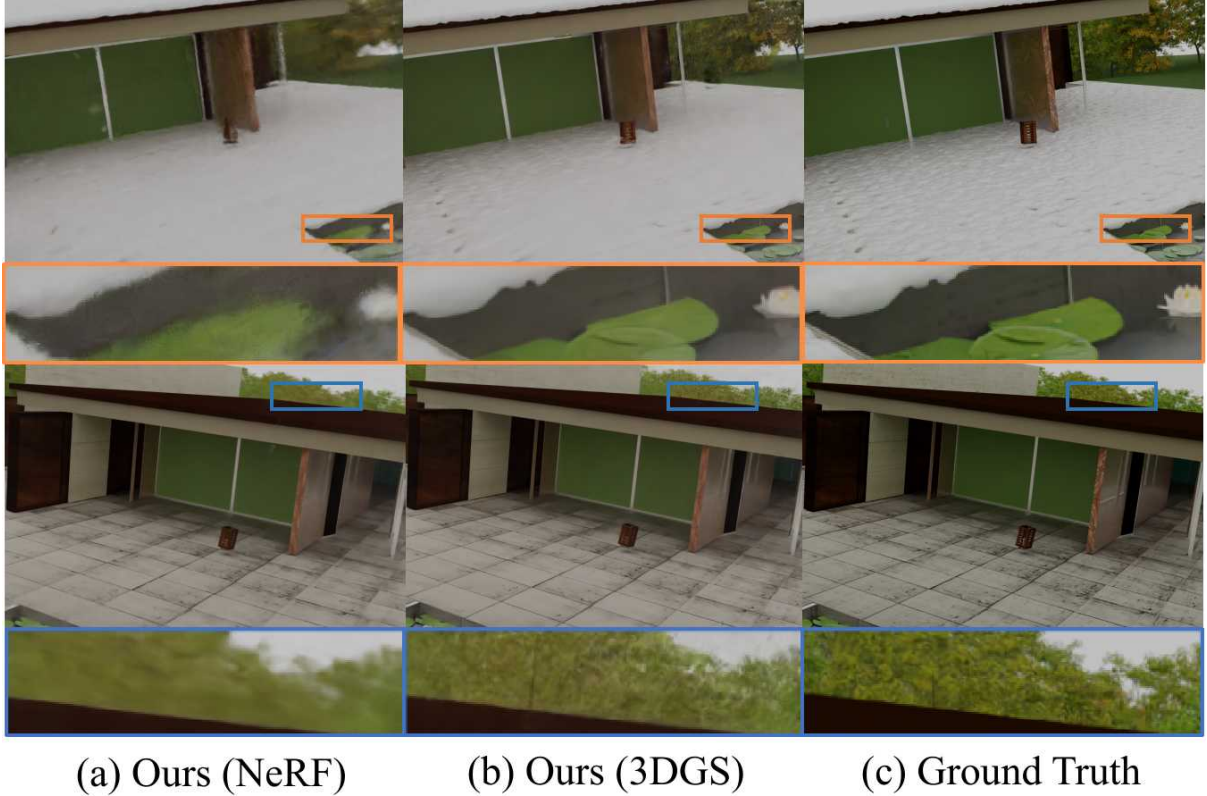


Figure 5.6: Backbone comparison: the same preprocessing pipeline with a NeRF backbone versus a 3DGS backbone. The 3DGS variant produces sharper reconstructions at a fraction of the training and rendering cost.

5.10 Discussion

Before closing the chapter, I draw out four observations that the experiments make visible only in retrospect.

Preprocessing versus end-to-end learning. A natural research instinct is to push preprocessing modules inside the optimisation loop: rather than running the AEF as a fixed preprocessor, jointly fine-tune the diffusion model alongside the 3DGS reconstruction so that the restoration learns to produce outputs that are maximally useful for the downstream reconstruction. I considered this design and rejected it on practical grounds.

Differentiating through the diffusion sampler is expensive and requires many denoising steps to be replayed at every reconstruction iteration; the resulting joint training is dominated by the 2D diffusion’s computational cost rather than the 3DGS reconstruction. The fixed-preprocessing alternative trades a small loss in restoration–reconstruction alignment for an order-of-magnitude reduction in compute, and may therefore be a more appropriate design point for a method whose downstream consumer is simulation rather than per-frame perception. The broader lesson is that end-to-end learning is not automatically better than modular composition when the modules are well engineered and the cost of joint optimisation is high.

The asymmetry between SSIM and LPIPS. A recurring observation across the experimental tables is that WeatherGS achieves the lowest LPIPS but is occasionally outperformed on SSIM by vanilla 3DGS. SSIM penalises any deviation from the ground-truth pixel structure, including the small content modifications introduced by the diffusion-based inpainting. LPIPS, evaluated in deep-feature space, is invariant to such cosmetic modifications and more aligned with perceptual judgement. The asymmetry between the two metrics is therefore informative: it indicates that WeatherGS has traded some pixel-level fidelity (which the diffusion inpainting cannot guarantee) for a substantial gain in perceptual cleanliness (which the LED-based masking does guarantee).

Generalisation to unseen weather types. The AEF currently supports two task plugins (derain, desnow), and the LED was trained on raindrop data. Generalising WeatherGS to additional conditions (dust storms, smoke, volcanic ash, freezing rain) is in principle a matter of training new plugins. In practice, the architectural separation between dense and sparse artefacts that the dense-to-sparse strategy exploits is condition-specific: it assumes that the new condition exhibits the same two-scale structure (small dense particles in the air; larger sparse occlusions on the lens). Conditions that violate this assumption, such as smoke, which is a continuous volume without a clear scale separation, would require a different decomposition. The chapter’s contribution is therefore best read as establishing the dense-to-sparse template for a class of weather artefacts rather than as a universal weather-handling system; the four conditions targeted by AllWeatherNet’s 2D enhancement (Chapter 4) and the rain/snow/fog of WeatherEdit (Chapter 7) do not all fit cleanly within the same dense-to-sparse decomposition.

The 3DGS-over-NeRF case, revisited. The $14\times/1,600\times$ speed-ups reported in Table 5.3 are not unique to WeatherGS; they are the speed-ups that 3DGS provides over NeRF in general. WeatherGS’s contribution at the backbone level is not the speed-up itself but the demonstration that the speed-up survives the addition of weather-aware preprocessing. This matters because a naive concern would be that the preprocessing adds enough computational overhead to negate the 3DGS efficiency advantage; the experimen-

tal evidence shows that it does not. The AEF is run once per training view (a one-time cost), the LED is similarly run once, and the 3DGS optimisation itself proceeds at its standard speed. The lesson is that the modular architecture of WeatherGS preserves the efficiency profile of its backbone, which is the property that enables driving-scale deployment.

5.11 Summary

This chapter presented WeatherGS, the first 3D Gaussian Splatting framework for clean scene reconstruction from weather-degraded multi-view inputs. Its dense-to-sparse pre-processing pipeline first removes atmospheric particles using a diffusion-based Atmospheric Effect Filter (AEF) and then detects lens occlusions using an AttGAN-based Lens Effect Detector (LED). Combined with mask-guided 3DGS optimisation, these components produce reconstructions that are clean, sharp, and computationally efficient. On a new benchmark comprising six synthetic and two real-world weather-degraded scenes, WeatherGS achieves the highest average PSNR, reaching 25.066 dB for snow and 23.670 dB for rain, as well as the lowest average LPIPS under both conditions. Compared with the NeRF-based alternative, it also provides approximately $14\times$ faster training and $1,600\times$ faster rendering.

Together with Chapter 4, this chapter completes Part II of the thesis on adverse-weather enhancement. The two chapters demonstrate that weather-degraded observations can be recovered into clean representations at both the 2D image level and the 3D scene level. In particular, the reconstructed 3D assets support downstream capabilities such as novel-view synthesis, simulation-asset creation, and controlled perception testing that cannot be provided by image enhancement alone.

Enhancement, however, assumes that observations of the target weather condition are already available. Autonomous-driving training and validation may instead require conditions that are rare, unsafe, or absent from existing real-world captures. Part III therefore turns to the complementary problem of generation: synthesising weather-degraded observations from clean scenes while preserving their semantic content and providing control over the generated weather conditions.

Part III

Generation: Synthesising Adverse Weather from Clean Inputs

Chapter 6

Generalising to Adverse Weather: WeatherDG

Part II showed that weather-degraded observations can be restored into clean 2D images and 3D scene representations when such observations are available. However, this enhancement setting only addresses weather conditions that have already been captured. In practice, autonomous driving systems must also generalise to adverse conditions that are rare, unsafe, seasonal, or geographically localised, such as heavy fog, nighttime snow, or region-specific rain patterns. Collecting sufficient real data for these conditions is costly and often impractical. As a result, perception models trained on clear weather or limited-weather datasets may suffer a substantial domain gap when deployed under unseen adverse weather.

This chapter studies the complementary problem of weather generation for domain generalisation. Instead of restoring a degraded image, I aim to synthesise adverse weather training images from clear weather driving data. The goal is not merely to produce visually plausible weather effects, but to generate images that are useful for downstream semantic segmentation. This requires three properties: photorealistic weather appearance, preservation of driving scene geometry, and sufficient semantic diversity to cover safety-critical classes. This can be challenging because real-world driving images captured under adverse weather conditions are not always available. The generation process must therefore rely on clear weather sources, generative priors, and structured prompt control, rather than direct supervision from real weather data.

To address this, I introduce **WeatherDG** (Qian et al., 2025b), a generative data-augmentation framework for adverse weather domain generalisation. WeatherDG combines a Stable Diffusion model fine-tuned with DreamBooth with a three-agent Large Language Model (LLM) pipeline consisting of an Instance Sampler, a Scene Composer, and a Scene Descriptor. The diffusion model provides a strong generative prior for realistic adverse weather synthesis, while the LLM agents explicitly control scene composition, weather diversity, and class balance. This design enables the generation of photorealistic and semantically diverse driving images without requiring real adverse weather training

samples.

Experiments reported in the WeatherDG paper (Qian et al., 2025b) show that WeatherDG-generated data substantially improves the robustness of downstream semantic segmentation models. When trained with a mixture of Cityscapes (Cordts et al., 2016) and WeatherDG-generated images, segmentation models achieve up to 13.9% mIoU improvement on ACDC (Sakaridis et al., 2021), up to 9.5% on Dark Zurich (Sakaridis et al., 2020), and up to 7.6% on BDD100K (Yu et al., 2020). These results demonstrate that structured generative augmentation can effectively bridge the gap between clear weather training data and adverse weather deployment domains. This work was published in *IEEE Robotics and Automation Letters* in 2025.

The remainder of this chapter is organised as follows. Section 6.1 introduces the motivation and problem setting. Section 6.2 reviews related work on domain generalisation, diffusion-based augmentation, and weather synthesis. Section 6.3 presents the WeatherDG pipeline. Sections 6.4 to 6.6 describe the DreamBooth adaptation, LLM-based prompt generation, and segmentation training procedure. Sections 6.7 to 6.9 report the experimental results and ablation studies. Finally, Section 6.10 concludes the chapter and motivates the 3D generation setting developed in Chapter 7.

6.1 Introduction

Perception models trained on standard clear weather datasets often suffer a substantial performance drop when deployed under adverse weather, such as fog, rain, snow (Qian et al., 2024). Yet, when training weather data are available, a large fraction of this degradation can be alleviated (Hoyer et al., 2024, 2022). This suggests that the bottleneck is not only model capacity, but also training-set coverage. In other words, adverse weather robustness is a problem of distributional coverage: the model may be expressive enough to recognise the relevant scene semantics, but the training distribution does not sufficiently cover the visual conditions encountered at deployment.

The key question is therefore how to expand the training distribution towards adverse weather. Direct real-world collection is the most natural solution, but it is difficult to scale. Capturing severe weather is costly, seasonal, geographically constrained, and potentially unsafe. Moreover, semantic segmentation requires dense pixel-level annotations, which remain expensive to obtain for each newly collected image. Existing datasets reflect this imbalance. Cityscapes (Cordts et al., 2016) provides 5,000 finely annotated urban images, while adverse weather datasets such as ACDC (Sakaridis et al., 2021) and DAWN (Kenk and Hassaballah, 2020) contain substantially fewer annotated samples. By contrast, large-scale driving datasets such as BDD100K (Yu et al., 2020) and nuScenes (Caesar et al., 2020) are much broader in scale, but still do not provide exhaustive coverage of rare or extreme weather regimes. As a result, extending adverse weather coverage through manual collection and annotation alone is unlikely to provide a scalable

solution.

Synthetic generation offers an alternative, but existing methods remain insufficient. Prior attempts can be broadly grouped into two families, each of which fails in a different way.

Graphics-based simulators. Driving simulators such as CARLA (Dosovitskiy et al., 2017) and the semantic driving simulator proposed by Kerim et al. (2022) can generate adverse weather scenes with explicit control over weather type, intensity, time of day, and camera viewpoint. This controllability is valuable for systematic data generation. However, simulator-rendered images often lack the fine-grained visual statistics of real dashcam imagery. Texture quality, material appearance, lighting, and colour distribution can differ substantially from real sensor data. Consequently, segmentation models trained on such images may overfit to simulator-specific visual cues and suffer from a sim-to-real gap when deployed on real adverse weather images. Fig. 6.1(a) illustrates this characteristic failure mode: the generated scene is geometrically coherent, but its coarse textures and synthetic colour rendering remain visibly different from real driving images.

Text-to-image diffusion. Large pre-trained text-to-image diffusion models, such as Stable Diffusion (Rombach et al., 2021), provide a second possible source of synthetic adverse weather data. Given a prompt such as “a foggy urban street at night”, they can produce visually compelling images without requiring a graphics engine. However, generic text-to-image models are trained on broad internet-scale image distributions rather than dashcam-specific driving data. As a result, they often generate images that are aesthetically plausible but geometrically unsuitable for autonomous driving perception. Camera viewpoints may deviate from dashcam geometry, object scales and layouts may be unrealistic, and the overall composition may resemble cinematic or artistic photography rather than real driving footage. When such images are used for segmentation training, especially under pseudo-labelling or self-training, they can shift the model away from the real driving distribution rather than towards it. Fig. 6.1(b) shows this failure mode: the image is visually impressive, but its perspective and scene composition are not consistent with real dashcam captures.

WeatherDG addresses these two limitations through two complementary design choices. First, I adapt Stable Diffusion to the driving domain via DreamBooth (Ruiz et al., 2023), fine-tuning it on a single Cityscapes image and associating a unique identifier V^* with the visual characteristics of a dashcam-perspective driving scene. This adaptation reduces the geometric gap: the resulting model generates images with more realistic camera perspective, road layout, and object context, while preserving Stable Diffusion’s strong generative priors for weather appearance, illumination, and object texture. Second, I introduce a three-agent LLM pipeline that procedurally constructs detailed prompts spanning diverse weather conditions, times of day, and semantic object categories. This addresses the di-



Figure 6.1: Representative failure modes of alternative sources for adverse weather training images. (a) A driving simulator (Kerim et al., 2022) provides geometric control but produces visually synthetic scenes with coarse textures and simulator-like colour statistics. (b) Vanilla Stable Diffusion (Rombach et al., 2021) generates visually appealing images, but often violates dashcam geometry through artistic composition and incorrect camera perspective. (c) WeatherDG generates photorealistic adverse weather driving images with dashcam-compatible viewpoint and scene layout, closely resembling (d) real adverse weather driving images.

versity gap: instead of relying on a small set of manually designed prompt templates, which typically cover only a narrow portion of the generation space, the LLM pipeline systematically explores the weather–scene–object combination space in a controlled and class-balanced manner.

WeatherDG is agnostic to the segmentation model. It is a data augmentation framework rather than a segmentation architecture, and can therefore be integrated with different self-training segmentation backbones. In my experiments, I evaluate WeatherDG with DAFormer (Hoyer et al., 2022), HRDA (Hoyer et al., 2024), and MIC (Hoyer et al., 2023), and show that it yields consistent, and often substantial, improvements across all three.

6.2 Related Work

The broader literature on diffusion-based generation has been reviewed in Section 2.1.3. This section focuses on three lines of work most directly related to WeatherDG: domain generalisation for semantic segmentation, unsupervised domain adaptation backbones, and text-guided image synthesis for driving scene augmentation.

Domain generalisation for semantic segmentation. Domain generalisation (DG) addresses the setting in which no target-domain data are available during training. Existing DG methods for semantic segmentation typically improve robustness by learning domain-invariant representations or by expanding the source distribution. Normalisation-based approaches, such as IBN-Net (Pan et al., 2018), RobustNet (Choi et al., 2021), and ISW (Peng et al., 2022), reduce sensitivity to appearance shifts by encouraging style-invariant features. Augmentation-based methods, such as SHADE (Zhao et al., 2022) and MoDify (Jiang et al., 2023), instead diversify the source data through synthetic style or appearance variations. WeatherDG follows the augmentation paradigm, but differs from prior work in two aspects. First, it generates photorealistic adverse weather images rather than applying low-level photometric perturbations to existing samples. Second, it explicitly controls weather type, time of day, and semantic class coverage through LLM-guided prompt construction, enabling more systematic coverage of the adverse weather generation space.

Unsupervised domain adaptation backbones. WeatherDG also builds on recent progress in unsupervised domain adaptation (UDA) for semantic segmentation. Target-data UDA backbones such as DAFormer (Hoyer et al., 2022), HRDA (Hoyer et al., 2024), and MIC (Hoyer et al., 2023) assume access to labelled source-domain images and unlabelled target-domain images during training. They typically align the two domains through self-training, pseudo-labelling, or adversarial objectives. Diffusion- and language-driven adaptation methods relax this assumption in different ways: DATUM (Benigmin et al., 2023) studies one-shot UDA with a single unlabelled target image, while PODA (Fahes et al., 2023) studies prompt-driven zero-shot domain adaptation. WeatherDG operates in a DG setting where real target-domain images are unavailable. I therefore use WeatherDG-generated adverse weather images as synthetic target-style data, allowing self-training machinery to be applied without requiring real target samples. This substitution trades exact target-domain fidelity for scalability: the generated images may not perfectly match the deployment distribution, but they provide controllable adverse weather coverage without target-domain data collection.

Text-guided image synthesis for driving. Text-to-image diffusion models have recently been explored for driving scene augmentation. DreamBooth (Ruiz et al., 2023) provides a subject-driven fine-tuning mechanism, which WeatherDG uses to adapt Stable

Diffusion to dashcam-perspective driving scenes. DDEX (Niemeijer et al., 2024) combines ControlNet (Zhang et al., 2023b) with simulator renderings to produce photorealistic driving images. While effective in improving visual realism, such methods still inherit the layout and object-distribution biases of the underlying simulator. CLOUDS (Benigim et al., 2024) uses a Llama-family language model (Touvron et al., 2023) to generate prompts for Stable Diffusion, but adopts a single-agent, single-round prompting strategy. This limits its ability to explicitly structure the generation space along task-relevant axes such as weather type, time of day, and object-class distribution.

LLM-assisted prompting has also been studied beyond driving. CuPL (Pratt et al., 2023) uses GPT-3 (Brown et al., 2020) to generate class descriptions for CLIP-based zero-shot classification, showing that LLM-generated prompts can outperform manually written templates. WeatherDG adopts the broader idea of language-guided prompt construction, but extends it to a multi-agent setting for controllable data generation. Specifically, it decomposes prompt generation into instance sampling, scene composition, and scene description, allowing each agent to specialise in a distinct aspect of the synthetic training distribution.

ZoDi (Azuma et al., 2024) is another diffusion-based zero-shot domain adaptation method for semantic segmentation and is included as a direct comparison in Section 6.8.

6.3 Overview of the WeatherDG Pipeline

WeatherDG consists of three sequential stages, as illustrated in Fig. 6.2.

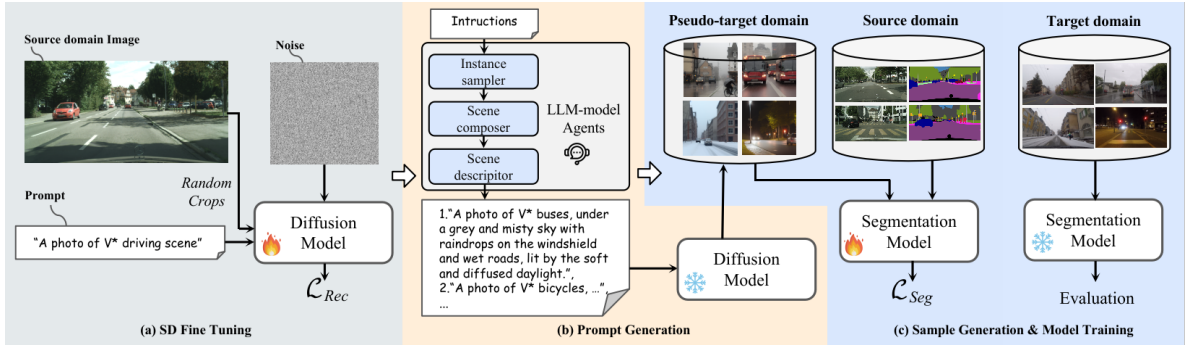


Figure 6.2: Overview of the WeatherDG pipeline. (a) Stable Diffusion is adapted to dashcam-perspective driving scenes through DreamBooth fine-tuning on a single source-domain image, binding the identifier V^* to driving scene geometry and layout. (b) A three-agent LLM pipeline, consisting of an Instance Sampler, Scene Composer, and Scene Descriptor, generates detailed and class-balanced weather prompts. (c) The fine-tuned diffusion model synthesises adverse weather driving images from these prompts. The generated images are then combined with the labelled source dataset to train a segmentation model using UDA-style self-training.

Stage 1: Driving scene adaptation. I first adapt Stable Diffusion to the driving domain using DreamBooth. A single image is randomly selected from Cityscapes and

used to bind the unique identifier V^* to the visual characteristics of dashcam-perspective driving scenes. The fine-tuning prompt, “A photo of V^* driving scene”, encourages the model to associate V^* with low-elevation forward-facing viewpoints, road-scene layout, and realistic object context. This stage takes approximately 10 minutes on a single NVIDIA A6000 GPU and yields a diffusion model that retains the general generative priors of Stable Diffusion while producing images with driving-compatible geometry and composition.

Stage 2: Procedural prompt generation. I then generate prompts through a three-agent LLM pipeline. The Instance Sampler $\mathcal{E}_{IS}$ selects object classes with probability inversely proportional to their frequency in a reference adverse weather dataset, encouraging coverage of long-tail categories. The Scene Composer $\mathcal{E}_{SC}$ places the selected class into a balanced combination of weather condition and time of day. The Scene Descriptor $\mathcal{E}_{SD}$ converts this structured specification into a detailed natural-language prompt. Together, these agents provide controlled coverage of the weather–time–class space, rather than relying on a small set of manually written prompt templates.

Stage 3: Synthetic data generation and segmentation training. Finally, the fine-tuned Stable Diffusion model renders synthetic adverse weather images from the generated prompts. These synthetic images are treated as unlabelled target-style data and combined with the labelled Cityscapes source set. A UDA backbone, such as DAFormer, HRDA, or MIC, is then trained through pseudo-labelling and self-training. The resulting segmentation model is evaluated on real adverse weather benchmarks, including ACDC, BDD100K, and Dark Zurich, none of which is used during training.

The remainder of this chapter describes each stage in detail and evaluates the contribution of the proposed design choices through experiments and ablation studies.

6.4 Stage 1: Driving Scene Fine-Tuning of Stable Diffusion

6.4.1 Why Fine-Tune

Although Stable Diffusion has strong general-purpose image-generation capability, its pre-training distribution is not specialised for autonomous driving imagery. The LAION-scale data used to train Stable Diffusion (Rombach et al., 2021) contains diverse web images, including photographs, illustrations, cinematic scenes, and stylised compositions. As a result, the model does not have a strong prior for the low-elevation, forward-facing, dashcam perspective required for driving scene segmentation. When prompted with a phrase such as “a photo of a motorcycle at night”, the model may generate a studio

photograph, a cinematic close-up, or an image with an elevated viewpoint, rather than a road-level driving scene.

Fig. 6.3 illustrates this mismatch across several driving scene categories, including bus, truck, car, motorcycle, bicycle, pedestrian, traffic light, traffic sign, and rider. Vanilla Stable Diffusion often produces aesthetically plausible images, but they are unsuitable as training data for autonomous driving segmentation. The generated images may contain stylised compositions, incorrect camera viewpoints, unrealistic object scales, or non-driving semantic associations. For example, the prompt “rider” may activate surf-riding or sports imagery rather than a rider in a traffic scene. Such images can produce misleading pseudo-labels and may shift the segmentation model away from the real driving distribution.



Figure 6.3: Representative outputs from vanilla Stable Diffusion and the DreamBooth-adapted WeatherDG model under the prompt template “A photo of [CLS].” The fine-tuned model produces images with dashcam-compatible viewpoints and driving scene composition, whereas the vanilla model often generates artistic, stylised, or geometrically unsuitable outputs.

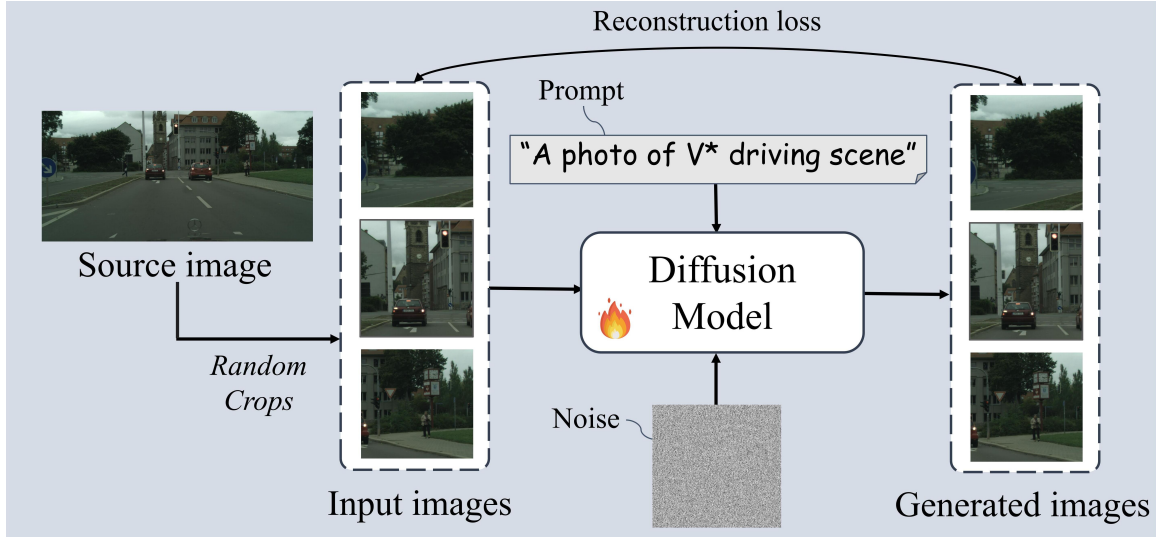
The fine-tuning stage addresses this limitation by anchoring the generation process to a source-domain driving image, while preserving the broad visual priors of the pre-trained diffusion model.

6.4.2 DreamBooth with the V^* Identifier

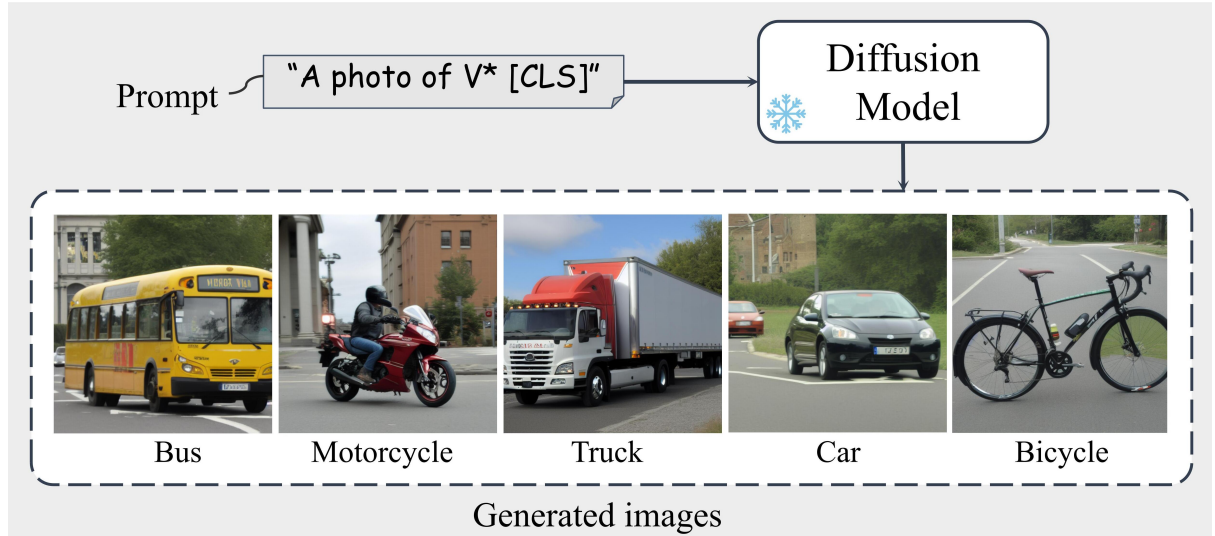
I adapt DreamBooth (Ruiz et al., 2023) to the driving scene generation setting, as shown in Fig. 6.4. A single image is randomly selected from the Cityscapes training set. During fine-tuning, I sample random 512×512 crops from this image and pair each crop with the prompt “A photo of V^* driving scene”. The model is optimised using the standard diffusion reconstruction loss:

$$\mathcal{L}_{\text{fine-tune}} = \mathbb{E}_{t, \mathbf{x}_0, \epsilon} \left[\left\| \epsilon - \epsilon_{\theta + \Delta\theta} \left(\sqrt{\bar{\alpha}_t} \mathbf{z}_0 + \sqrt{1 - \bar{\alpha}_t} \epsilon, t, p \right) \right\|^2 \right], \quad (6.1)$$

where $\mathbf{z}_0 = \mathcal{E}(\mathbf{x}_0)$ is the latent code of the input crop, $\epsilon \sim \mathcal{N}(0, \mathbf{I})$ is the added Gaussian noise, t is the diffusion timestep, θ denotes the frozen Stable Diffusion parameters, $\Delta\theta$ denotes the learnable fine-tuning parameters, and p is the fine-tuning prompt containing the identifier V^* .



(a) Training stage: the identifier V^* is bound to visual content and layout statistics extracted from crops of a single Cityscapes image.



(b) Inference stage: V^* primes the model to generate dashcam-perspective driving images when combined with class, weather, and time descriptors.

Figure 6.4: DreamBooth fine-tuning of Stable Diffusion with the V^* identifier. After fine-tuning on a single source-domain image, the model can generate dashcam-perspective driving images across diverse class and weather prompts.

The coarse descriptor “driving scene” is included after V^* to stabilise the language grounding during fine-tuning. Without this descriptor, the identifier may absorb too much semantic information from the single training image, causing language drift and weakening the model’s ability to associate natural-language class names with their intended visual concepts. The phrase “driving scene” preserves the link between class tokens and their general visual meanings, while allowing V^* to encode the dashcam-specific scene prior. After fine-tuning, prompts such as “A photo of V^* [CLS]” or “A photo of V^* [CLS], [WEATHER], [TIME]” generate images that combine the requested semantic class and weather condition with the desired driving scene geometry.

6.4.3 What V^* Encodes

The learned identifier V^* encodes a compact driving scene prior. This prior captures structural properties that are difficult to obtain from generic text prompting alone, including low camera elevation, forward-facing viewpoint, rectilinear road perspective, and typical object scales and placements in urban driving scenes. Other aspects of the image, such as weather condition, illumination, colour, and object identity, remain controlled by the surrounding prompt and the pre-trained diffusion prior. This separation is key to the data efficiency of the approach: WeatherDG does not attempt to teach Stable Diffusion the full driving distribution from one image, but instead uses the single image to anchor the geometric and compositional properties needed for dashcam-compatible generation.

6.5 Stage 2: Procedural Prompt Generation

6.5.1 Limitations of Single-Template Prompting

Given the DreamBooth-adapted Stable Diffusion model, a straightforward augmentation strategy is to use a fixed prompt template, such as “A photo of V^* [CLS], [WEATHER], [TIME].” Although simple, this strategy has two important limitations.

First, single-template prompting provides limited semantic coverage. Varying only the class, weather, and time tokens produces images that are syntactically different but often semantically repetitive. The generated dataset therefore covers only a narrow subset of the possible weather–scene–object combinations, reducing its value for robust segmentation training.

Second, single-template prompting does not address class imbalance at the pixel level. Uniformly sampling class names treats all classes as equally represented in terms of image count, but this does not translate into balanced pixel coverage. Stuff classes such as road, sidewalk, building, and sky dominate most generated images regardless of the requested object class. For example, an image prompted by “a photo of a motorcycle” may contain a motorcycle only in a small region, while the majority of pixels still correspond to road,

pavement, buildings, or sky. What is required for segmentation is therefore not image-count balance alone, but improved pixel-level coverage of under-represented classes.

WeatherDG addresses these two limitations through a structured three-agent prompt-generation pipeline.

6.5.2 Three-Agent Decomposition

I decompose prompt generation into three sequential LLM agents. The pipeline is summarised as

$$\mathcal{T} = \mathcal{E}_{SD}(\mathcal{E}_{SC}(\mathcal{E}_{IS}(\mathcal{P}))), \quad (6.2)$$

where $\mathcal{P}$ denotes the class-sampling probability distribution and $\mathcal{T}$ is the final prompt passed to Stable Diffusion. Each agent is assigned a specific role: the Instance Sampler selects the semantic class, the Scene Composer specifies the weather and time context, and the Scene Descriptor expands the prompt into a detailed visual description. This decomposition makes the generation process more controllable than a single handcrafted template, while preserving the flexibility of natural-language prompting.

Fig. 6.5 illustrates the progressive effect of the three agents. Using $\mathcal{E}_{IS}$ alone produces a class-conditioned driving image with limited environmental variation. Adding $\mathcal{E}_{SC}$ introduces explicit weather and time-of-day context, but the resulting visual effects may remain weak. The full pipeline, including $\mathcal{E}_{SD}$, generates richer and more photorealistic adverse weather scenes. The agents are complementary: the Scene Descriptor is more effective when the weather context has already been specified by the Scene Composer, and the Scene Composer operates on the class selected by the Instance Sampler. As shown in Section 6.9.2, removing any of these components leads to a measurable drop in segmentation performance.

6.5.3 Instance Sampler $\mathcal{E}_{IS}$

The Instance Sampler selects the object class to be emphasised in the generated image. A naive strategy would be to sample classes uniformly. However, as discussed in Section 6.5.1, uniform image-level sampling does not correct pixel-level class imbalance, since many safety-critical thing classes occupy only small regions even when explicitly mentioned in the prompt.

Let D_i denote the pixel count of the i -th thing class in a reference adverse weather dataset, for which I use ACDC (Sakaridis et al., 2021), and let $D_{\text{thing}} = \sum_i D_i$ be the total thing-class pixel count. The empirical class distribution is

$$E_i = \frac{D_i}{D_{\text{thing}}}. \quad (6.3)$$

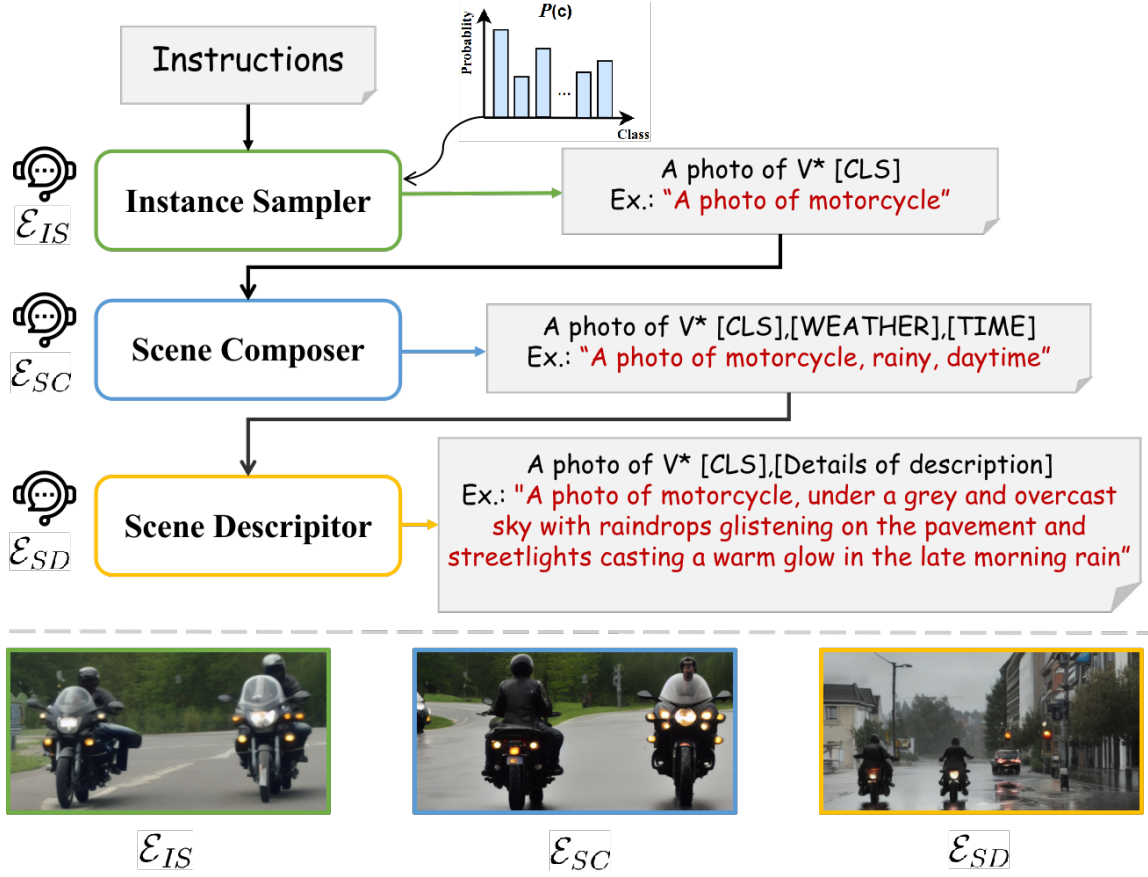


Figure 6.5: Incremental effect of the three LLM agents on the generated image. Rows correspond to prompts produced by $\mathcal{E}_{IS}$ alone (top), $\mathcal{E}_{IS} + \mathcal{E}_{SC}$ (middle), and the full $\mathcal{E}_{IS} + \mathcal{E}_{SC} + \mathcal{E}_{SD}$ pipeline (bottom). The Scene Descriptor enriches the prompt with detailed weather cues, transforming a generic weather-labelled image into a more photorealistic adverse weather scene.

The Instance Sampler then samples class i with probability inversely proportional to E_i :

$$P_i = \frac{1/E_i}{\sum_{j=1}^n 1/E_j}. \quad (6.4)$$

This inverse-frequency sampling increases the probability of selecting classes that occupy fewer pixels in the reference distribution, such as rider, motorcycle, or train. It therefore compensates for the small spatial footprint these classes typically have within a generated driving image.

The Instance Sampler is not a generative model in itself. It is an LLM-mediated sampling routine over a pre-computed class-probability table. This implementation provides a flexible interface: alternative reference datasets, class subsets, or weighting schemes can be specified through natural-language instructions without changing the surrounding pipeline.

6.5.4 Scene Composer $\mathcal{E}_{SC}$

The Scene Composer takes the sampled class and constructs a structured prompt of the form

“A photo of V^* [CLS], [WEATHER], [TIME].”

The WEATHER slot is sampled from a balanced set of adverse conditions, namely snowy, rainy, and foggy, while the TIME slot is sampled from daytime and nighttime. All weather–time combinations are generated with equal frequency, encouraging balanced coverage across the major adverse weather settings considered in this chapter.

The output of the Scene Composer specifies the desired class, weather condition, and time of day, but remains visually sparse. As illustrated in the middle row of Fig. 6.5, a prompt containing only a bare weather adjective such as “rainy” may produce a weak weather effect, for example a wet road surface without visible raindrops, reflections, or atmospheric haze. This occurs because short attribute prompts often under-specify the visual manifestations of weather. The Scene Descriptor addresses this by expanding the structured prompt into a richer natural-language description.

6.5.5 Scene Descriptor $\mathcal{E}_{SD}$

The Scene Descriptor converts the structured prompt from the Scene Composer into a detailed weather-aware description. For example, the prompt “A photo of V^* motorcycle, rainy, daytime” can be expanded into a description containing overcast sky, wet pavement, puddle reflections, raindrops, mist, and diffuse lighting. These details provide the diffusion model with stronger visual cues for rendering adverse weather effects.

This elaboration is important for scalable data generation. Manually writing detailed prompts for thousands of synthetic images would be time-consuming and difficult to keep diverse. The LLM-based Scene Descriptor automates this process and makes it possible to regenerate the synthetic training set under different stylistic or deployment assumptions, such as changing the urban environment or regional visual context.

The detailed prompts produced by $\mathcal{E}_{SD}$ are the main driver of rich weather appearance in the generated images. For snowy scenes, the descriptors encourage accumulated snow, airborne snowflakes, and snow-covered vegetation. For rainy scenes, they introduce reflections, raindrops, wet surfaces, and mist. For nighttime scenes, they promote street-light illumination, appropriate colour temperature, and stronger shadow structure. As shown in the bottom row of Fig. 6.5, these descriptions substantially improve the visual realism and weather specificity of the generated samples.

6.6 Stage 3: Sample Generation and Segmentation-Model Training

6.6.1 Synthetic Image Generation

For each final prompt $\mathcal{T}$ produced by the procedural pipeline, I sample one image from the fine-tuned Stable Diffusion model using the standard DDIM sampler with 50 denoising steps. At the recommended 1,000-image scale (see Section 6.9.3), generation takes approximately 5 hours on a single NVIDIA A6000 GPU. This is a one-time cost: once the synthetic images are generated, they serve as a reusable training asset across any segmentation backbone.

6.6.2 Pseudo-Target UDA Training

The generated images are unlabelled. Direct supervised training is therefore not possible on them alone. I adopt the standard UDA paradigm: the labelled Cityscapes dataset serves as the source domain, and the unlabelled WeatherDG-generated images serve as the pseudo-target domain. A UDA backbone (DAFormer (Hoyer et al., 2022), HRDA (Hoyer et al., 2024), or MIC (Hoyer et al., 2023)) is trained with its standard combination of source supervision, self-training on pseudo-labelled target-domain images, and any architecture-specific regularisation (rare-class sampling, masked consistency, multi-resolution processing, and so on). The key departure from standard UDA is that the target domain is synthetic rather than real; the key claim is that the synthetic target domain is a sufficiently good proxy for the real adverse weather distribution to drive useful training.

The model trained in this way is then evaluated on three held-out real-world benchmarks: ACDC (Sakaridis et al., 2021), BDD100K (Yu et al., 2020), and Dark Zurich (Sakaridis et al., 2020). Because none of these benchmarks are seen at any stage of WeatherDG training, the evaluation is a genuine test of domain generalisation.

6.7 Experimental Setup

6.7.1 Datasets

Source domain (labelled). Cityscapes (Cordts et al., 2016): 2,975 finely annotated urban driving images captured in clear weather and daylight across German cities.

Target domains (real, held out from WeatherDG training).

- ACDC (Sakaridis et al., 2021): 4,006 adverse-condition images, evenly distributed across fog, rain, snow, and night. 406 validation images are used for evaluation.

- BDD100K (Yu et al., 2020): large-scale driving video dataset covering diverse weather and geographic locations. 1,000 validation images are used.
- Dark Zurich (Sakaridis et al., 2020): 8,779 nighttime-focused driving images. 50 validation images are used.

WeatherDG training data. I fine-tune Stable Diffusion on a single randomly chosen image from the Cityscapes training set. I generate up to 1,273 synthetic images using the procedural prompting pipeline; the 1,000 mark is where returns diminish (see Section 6.9.3).

6.7.2 Evaluation Metric and Backbones

I report mean Intersection-over-Union (mIoU) on each target dataset, following the Cityscapes 19-class protocol. Performance is evaluated across three UDA backbones (DAFormer, HRDA, and MIC), each paired with two encoder choices: ResNet-50 (with DeepLabV2 decoder) and MiT-B5 (with SegFormer decoder). The matrix of 3 backbones $\times$ 2 encoders $\times$ 3 target datasets yields 18 experimental conditions per configuration.

6.7.3 Implementation Details

Stable Diffusion fine-tuning uses the DreamBooth implementation of (Ruiz et al., 2023) with a learning rate of 10^{-6} , batch size 1, and 800 training steps; convergence occurs within approximately 10 minutes on a single NVIDIA A6000 with 20 GB of memory. Images are generated at 512×512 with classifier-free guidance scale 7.5. The LLM pipeline uses Llama-2-7B-chat for all three agents. Segmentation-backbone training follows each backbone’s published recipe; I refer the reader to (Hoyer et al., 2022, 2024, 2023) for hyperparameter details.

6.8 Results and Analysis

6.8.1 Comparison with State-of-the-Art Methods

I compare WeatherDG with representative domain-generalisation methods using ResNet-50 and MiT-B5 encoders. All methods are trained on Cityscapes and evaluated, without access to target-domain training data, on ACDC, BDD100K, and Dark Zurich. These benchmarks jointly test generalisation across snow, rain, fog, low-light conditions, and changes in geographic appearance. The results are reported in Table 6.1.

With ResNet-50, WeatherDG achieves the highest average mIoU (38.2%) and the strongest results on ACDC and Dark Zurich. SHADE obtains a higher score on BDD100K, but WeatherDG provides the best overall performance across the three target domains.

Table 6.1: Comparison with state-of-the-art domain-generalisation methods (mIoU, %). All methods are trained on Cityscapes and evaluated on ACDC, BDD100K, and Dark Zurich.

Method	Encoder	ACDC	BDD100K	Dark Zurich	Avg.
Source-only		35.9	37.3	9.0	27.4
IBN-Net (Pan et al., 2018)		42.0	45.8	17.3	35.0
RobustNet (Choi et al., 2021)	ResNet-50	41.7	43.4	19.4	34.8
SHADE (Zhao et al., 2022)		42.1	49.1	22.6	37.9
DPCL (Yang et al., 2023)		43.8	44.9	23.4	37.3
WeatherDG (Ours)		45.2	45.8	23.5	38.2
Source-only		44.6	44.7	18.7	36.0
DAFormer (Hoyer et al., 2022)		47.8	49.2	24.7	40.6
HRDA (Hoyer et al., 2024)	MiT-B5	46.3	49.8	25.8	40.6
MIC (Hoyer et al., 2023)		50.0	53.1	21.5	41.5
WeatherDG (Ours)		60.2	57.4	35.3	51.0

The advantage is more pronounced with MiT-B5: WeatherDG reaches 60.2%, 57.4%, and 35.3% mIoU on ACDC, BDD100K, and Dark Zurich, respectively, exceeding MIC by 10.2 points on ACDC, 4.3 points on BDD100K, and 13.8 points on Dark Zurich. Its average mIoU of 51.0% is 9.5 points above MIC, the strongest competing MiT-B5 method in the comparison.

The gains are largest on ACDC and Dark Zurich, where adverse weather and low illumination create substantial appearance shifts from the clear daytime Cityscapes source domain. Together with the qualitative examples in Fig. 6.6, these results show that WeatherDG improves generalisation under fog, rain, snow, and nighttime conditions while remaining independent of a particular segmentation architecture.

6.8.2 Qualitative Segmentation Results

Fig. 6.6 compares segmentation predictions of MIC (red) and MIC+WeatherDG (green) on representative ACDC images. Four patterns emerge.

- In foggy scenes, the baseline merges distant sidewalks into the road class or hallucinates trees where buildings should be; WeatherDG correctly separates sidewalk, building, and sky.
- In nighttime scenes, the baseline assigns wide road regions to background or misses road boundaries; WeatherDG preserves road extent even under complex lighting.
- In rainy scenes, the baseline misclassifies road reflections as vehicles (the puddle-reflection-as-truck failure mode); WeatherDG correctly recognises these as road.
- In snowy scenes, the baseline fails to detect pedestrians whose figures blend with the snow-covered background; WeatherDG detects them.

These qualitative patterns match the structural argument of Section 6.8.1: WeatherDG is providing the segmentation model with examples of exactly the failure modes that baseline training would not have prepared it for.

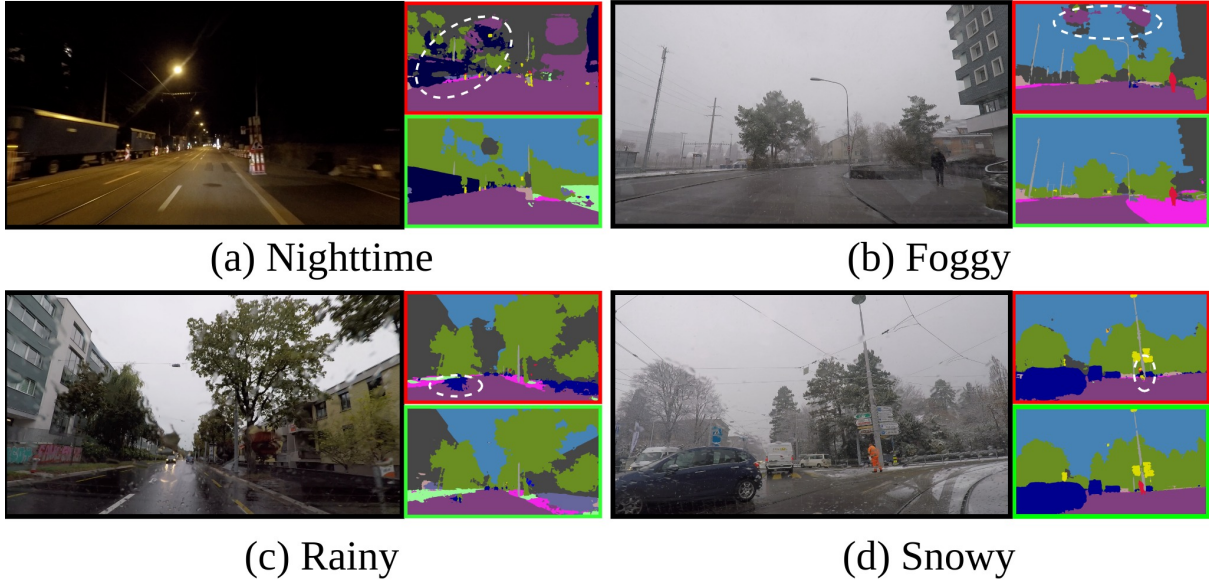


Figure 6.6: Segmentation predictions of MIC (red box) versus MIC+WeatherDG (green box) on ACDC. WeatherDG augmentation resolves weather-induced failure modes including misclassification of wet-road reflections as vehicles, loss of sidewalk boundaries in fog, and failure to detect pedestrians on snowy roads.

6.8.3 Comparison with Diffusion-Based Adaptation Methods

Table 6.2 compares WeatherDG with other diffusion-based adaptation methods (DATUM (Benigmim et al., 2023), ZoDi (Azuma et al., 2024)) on a common baseline (HRDA+MiT-B5).

Table 6.2: Comparison with other diffusion-based adaptation methods. Baseline is HRDA+MiT-B5, trained on Cityscapes; all augmentation methods use their own diffusion-generated images as pseudo-target.

Method	Test domains mIoU			Avg.
	ACDC	BDD100K	Dark Zurich	
Source-only HRDA (Hoyer et al., 2024)	46.3	49.8	25.8	40.6
+ DATUM (Benigmim et al., 2023)	56.2	55.9	29.7	47.2
+ ZoDi (Azuma et al., 2024)	53.0	51.5	21.5	42.0
+ WeatherDG (Ours)	60.2	57.4	35.3	51.0

WeatherDG outperforms DATUM by +3.8 mIoU on average and ZoDi by +9.0 mIoU on average. The gap is largest on Dark Zurich, where WeatherDG’s explicit control of the time-of-day axis through the Scene Composer shines, and on ACDC, where the range of weather conditions makes balanced prompt generation particularly valuable. The result indicates that the gains of WeatherDG over baselines are not simply the result of using Stable Diffusion for augmentation (ZoDi already does this); they come specifically from the combination of driving scene fine-tuning and multi-agent LLM prompting.

6.8.4 Per-Weather Breakdown on ACDC

Table 6.3 reports ACDC mIoU broken down by weather condition for WeatherDG combined with DAFormer, MIC, and HRDA. All methods use the MiT-B5 encoder.

Table 6.3: Per-weather mIoU on ACDC using WeatherDG augmentation. All methods use the MiT-B5 encoder.

Method	Test weathers			
	Snowy	Foggy	Rainy	Nighttime
DAFormer+Ours	54.2	66.8	54.1	27.3
MIC+Ours	59.4	74.2	62.0	36.9
HRDA+Ours	59.7	75.9	64.6	38.7
Average	57.8	72.3	60.2	34.3

All three backbones show the same ordering: foggy (easiest) > rainy > snowy > nighttime (hardest). Nighttime is the most challenging condition, with no backbone reaching 40 mIoU, despite WeatherDG’s inclusion of nighttime prompts and generation of convincing nighttime training images. Investigation traces the residual nighttime gap to the pseudo-labelling step of self-training: because the labelled source domain (Cityscapes) is entirely daytime, the pseudo-labels on WeatherDG’s generated nighttime images are themselves noisy, limiting the information the segmentation model can extract from them. This finding suggests a promising direction for future work: specialised pseudo-labelling protocols for nighttime data, potentially using auxiliary models that are themselves nighttime-robust.

6.9 Ablation Studies

6.9.1 Effect of Driving Scene Fine-Tuning

I first evaluate the contribution of DreamBooth-based driving scene fine-tuning. I compare two variants of WeatherDG: one using vanilla Stable Diffusion and the other using the DreamBooth-adapted model. Both variants use the same three-agent LLM prompting pipeline, so the only difference is whether the diffusion model is adapted to dashcam-perspective driving scenes.

Table 6.4: Effect of DreamBooth fine-tuning on segmentation mIoU. The segmentation backbone is DAFormer with MiT-B5.

Method	ACDC	BDD100K	Dark Zurich
w/o Fine-tuning	49.9	52.1	24.7
w/ Fine-tuning	50.8	53.2	25.6

As shown in Table 6.4, driving scene fine-tuning provides consistent gains across all three target datasets, improving mIoU by 0.9–1.1 points. This confirms that adapting

the diffusion prior to dashcam geometry and road-scene layout improves the usefulness of the generated images for segmentation training. The gain is moderate compared with the visual improvement observed in Fig. 6.3, suggesting that downstream performance is also constrained by the quality of pseudo-labels used during UDA-style training.

6.9.2 Contribution of Each LLM Agent

I next ablate the three-agent prompting pipeline by progressively adding the Instance Sampler, Scene Composer, and Scene Descriptor. The base model is trained without WeatherDG augmentation.

Table 6.5: Contribution of each LLM agent. Results are reported as mIoU on ACDC using DAFormer with MiT-B5.

Model	LLM agents			mIoU
	$\mathcal{E}_{IS}$	$\mathcal{E}_{SC}$	$\mathcal{E}_{SD}$	
M_{base}	–	–	–	50.8
M_1	✓	–	–	51.5
M_2	✓	✓	–	52.1
M_3	✓	✓	✓	53.3

Each agent improves performance over the previous configuration. The Instance Sampler brings a 0.7-point gain by increasing the sampling frequency of under-represented thing classes. Adding the Scene Composer raises the improvement to 1.3 points, indicating the benefit of explicitly balancing weather and time-of-day conditions. The full pipeline achieves a 2.5-point gain, with the largest additional improvement coming from the Scene Descriptor. This is consistent with the qualitative results in Fig. 6.5: detailed weather descriptions produce richer and more visually specific adverse weather images than short attribute prompts.

6.9.3 Effect of Synthetic-Dataset Size

Finally, I study how performance varies with the number of WeatherDG-generated images. Fig. 6.7 reports mIoU as a function of synthetic-dataset size using the DAFormer backbone.

The results show two trends. First, performance improves as more synthetic images are added, but the gain saturates around 1,000 images. I therefore use this setting for the main experiments in Section 6.8. Second, the saturation level is lower on Dark Zurich than on ACDC or BDD100K. This is consistent with the pseudo-labelling bottleneck discussed in Section 6.8.4: additional nighttime images are useful only if the teacher model can provide sufficiently reliable pseudo-labels. When pseudo-label quality is limited, increasing the number of generated samples yields diminishing returns.

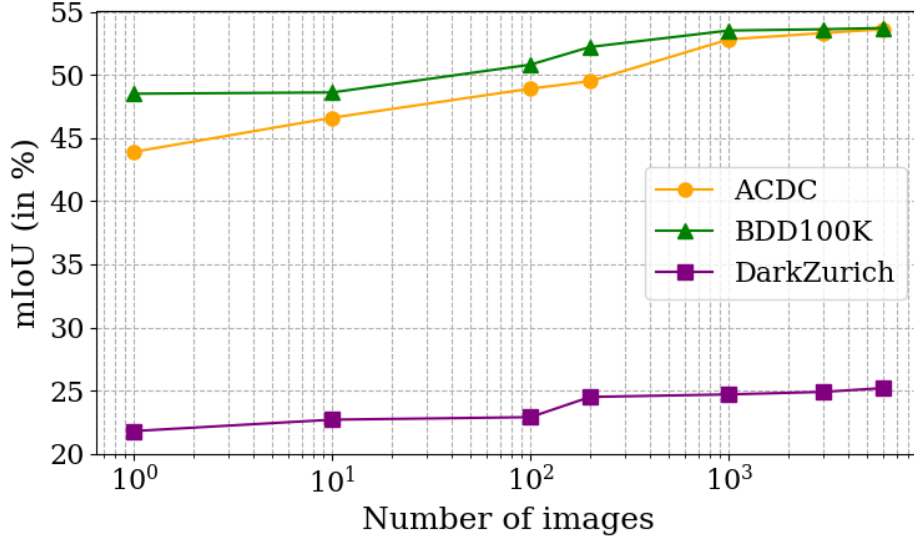


Figure 6.7: Effect of synthetic-dataset size on downstream segmentation performance. The gains increase with the number of WeatherDG-generated images and saturate at approximately 1,000 samples.

6.10 Summary

This chapter presented WeatherDG, a data-augmentation framework that generates photorealistic adverse weather training images through the combination of DreamBooth fine-tuning of Stable Diffusion and a three-agent LLM prompting pipeline. The fine-tuning ensures that generated images have the dashcam-perspective layout of real driving data; the LLM pipeline ensures that generated images cover a balanced, diverse, and richly described range of weather conditions, times of day, and object classes. On three held-out real-world benchmarks (ACDC, BDD100K, Dark Zurich), segmentation models trained with WeatherDG augmentation outperform their clean-trained baselines by up to 13.9, 7.6, and 9.5 mIoU points respectively, and outperform other diffusion-based augmentation methods by substantial margins.

WeatherDG is the 2D realisation of the Generation capability that defines Part III of the thesis. Its generated images are individual dashcam-perspective frames, independently sampled from the fine-tuned diffusion model: they are sufficient for augmenting single-camera perception training, but they are not geometrically linked to any underlying 3D scene. Two limitations follow from this 2D framing. First, the generated images are mutually inconsistent: two images generated for the same prompt do not correspond to two views of the same scene, making the data unsuitable for multi-camera training where cross-view consistency is structurally required. Second, the 2D images cannot drive simulation-grade evaluation: a closed-loop simulator that renders novel viewpoints of a scene under specified weather conditions needs a 3D representation that can be queried from arbitrary cameras, not a library of per-prompt 2D samples.

The next chapter, the second and final chapter of Part III, extends the Generation capability from 2D to 3D.

Chapter 7

Generating 3D Weather Scenes: WeatherEdit

The preceding chapter studied the 2D formulation of generation through WeatherDG, which synthesises independent weather-augmented images for single-camera segmentation training. This formulation is effective for diversifying the training distribution of per-frame perception models. However, it is not sufficient for simulation-grade evaluation. In driving simulation, weather effects must remain consistent across multiple cameras in a rig, persist coherently over time, and be renderable from novel viewpoints beyond those observed in the original sequence. These properties cannot be obtained by independently editing individual images and aggregating the results post hoc. They require weather to be represented in the same three-dimensional space as the scene itself.

This chapter extends generation from 2D image synthesis to 3D scene-level weather editing. The input is a clean reconstructed driving scene, such as the 3D scene asset produced by WeatherGS in Part II. The output is a weather-edited version of that scene, with controllable weather type, event, and intensity, while preserving multi-view and multi-frame consistency by construction. Together, WeatherGS and WeatherEdit form the 3D branch of the thesis framework: WeatherGS reconstructs clean 3D scene assets from weather-degraded real-world captures, whereas WeatherEdit injects controllable adverse weather effects into clean 3D scenes. This completes the pipeline from real-world data capture to simulation-ready adverse weather content.

The principal question of this chapter is whether realistic and controllable weather effects can be synthesised directly within a reconstructed 3D driving scene. Such a method must satisfy several requirements simultaneously: it should preserve the underlying scene structure, maintain consistency across views and frames, support dynamic weather events such as falling rain or snow, and provide user control over weather type and severity. I address this problem through **WeatherEdit** (Qian et al., 2026), a controllable 3D weather-editing framework built on a two-stage pipeline.

The first stage, weather background editing, adapts a pre-trained diffusion model to generate multi-view and multi-frame consistent weather backgrounds. This is achieved

through an All-in-One adapter and a parameter-free Temporal-View (TV) attention mechanism. The second stage, weather particle construction, injects dynamic weather particles into the reconstructed 3D scene using a Gaussian particle field with physics-inspired dynamics and local field alignment. The two stages are complementary: background editing captures global appearance changes such as fog, wetness, illumination shift, and snow accumulation, while particle construction models dynamic volumetric effects such as falling rain and snow.

Together, these components satisfy the realism and controllability requirements introduced in Section 7.1.2: distortion-free scene editing, temporal-view consistency, dynamic weather modelling, and controllable type, event, and severity. Experiments on four driving datasets, PandaSet (Xiao et al., 2021), Waymo Open (Sun et al., 2020), nuScenes (Caesar et al., 2020), and KITTI-360 (Liao et al., 2023), demonstrate state-of-the-art editing quality, including 84.5% user preference on foggy scenes. When the edited outputs are used for training-data augmentation, they further improve downstream semantic segmentation by up to 14.9% mIoU. The work presented in this chapter was accepted to the AAAI Conference on Artificial Intelligence (AAAI) 2026 (Qian et al., 2026).

A note on terminology is useful. The original paper uses the term “4D Gaussian field” to describe the particle representation, where the fourth dimension corresponds to time. In the thesis framework, however, WeatherEdit is categorised as a 3D generation method: the scene geometry is three-dimensional, the particles occupy 3D space, and temporal dynamics describe the evolution of those particles rather than an additional spatial dimension. For clarity, this chapter therefore uses the term “Gaussian particle field with temporal dynamics” when the distinction is relevant.

The remainder of this chapter is organised as follows. Section 7.1 introduces the motivation and formalises the realism and controllability requirements. Section 7.2 reviews the most directly relevant prior work. Section 7.3 presents the weather background editing stage, including the All-in-One adapter, semantic conditioning, and TV attention. Section 7.4 presents the weather particle construction stage, including the Gaussian particle field, attribute modelling, dynamic simulation, and local field alignment. Sections 7.5 to 7.7 report the experimental setup, results, and ablation studies. Finally, Section 7.8 concludes the chapter and leads into the thesis-wide discussion in Chapter 8.

7.1 Introduction

7.1.1 The Problem of 3D Weather Editing

A clean reconstructed 3D driving scene is more than a static record of a real-world capture. It is a reusable simulation asset whose value depends on the range of controlled variations that can be generated from it. For autonomous driving validation, many of

variations are often those that are difficult, unsafe, or rare to collect in the real world, such as heavy rain, snowstorms, fog, and their combinations. If a clean 3D reconstruction can be rendered under user-specified adverse weather, then each real-world drive can be transformed into a large set of virtual test scenarios with different atmospheric conditions, event states, and severity levels.

Achieving this capability is non-trivial. A straightforward solution is to edit rendered 2D frames independently and then aggregate the results. However, this approach does not provide the consistency required for simulation. Image editors such as ControlNet (Zhang et al., 2023b) and InstructPix2Pix (Brooks et al., 2023) can produce plausible weather effects in a single image, but they do not guarantee that the same effect remains coherent across cameras or over time. Puddles, reflections, fog density, and snow accumulation may change inconsistently from one view to another. Video editors (Deutch et al., 2024) improve temporal coherence for a single sequence, but they are not designed for synchronised multi-camera driving rigs. Moreover, purely 2D editing cannot explicitly model dynamic weather particles such as falling rain or drifting snow, since these effects occupy and evolve within 3D space.

A second direction is to model weather directly in a 3D representation. ClimateNeRF (Li et al., 2023a), for example, introduces static weather effects such as accumulated snow and flood water into NeRF-reconstructed scenes. This provides multi-view consistency by construction, since the weather becomes part of the 3D scene representation. However, static post-weather appearance is only one part of adverse weather simulation. Perception systems must also be tested under active weather events, including falling rain, blowing snow, and fog with varying visibility. These dynamic effects cannot be fully captured by static scene editing alone.

The missing capability is therefore a framework that combines the strengths of both directions: it should be 3D-consistent by construction, while also supporting dynamic weather effects and user-controllable weather type and severity.

7.1.2 Realism and Controllability Requirements

I formalise this goal through six requirements, grouped into realism and controllability.

Realism.

R1: Distortion-free editing. The edited scene should preserve the geometric and semantic content of the original reconstruction. Weather editing should not remove objects, introduce spurious objects, distort object shapes, or apply a global style change that alter the scene.

R2: Temporal-view consistency. Weather effects should remain coherent across both camera views and time steps. Multi-camera renderings at the same time should

observe the same weather event with appropriate parallax, and consecutive frames should exhibit physically plausible temporal evolution.

R3: Dynamic weather effects. The edited scene should support active weather phenomena such as falling rain, drifting snow, and volumetric fog. Static post-weather appearance, such as wet roads or accumulated snow, should be supported as one case rather than treated as the full problem.

Controllability.

C1: Type control. The user should be able to specify the weather type, including rain, snow, fog, or their combinations.

C2: Event control. The user should be able to distinguish active weather events from post-weather states, such as falling rain versus wet road surfaces, or falling snow versus accumulated snow.

C3: Severity control. The user should be able to adjust the intensity of the weather effect, such as light rain, heavy rain, light fog, or dense fog, preferably in a continuous manner.

WeatherEdit is designed to satisfy these requirements jointly, as illustrated in Fig. 7.1. Existing approaches typically address only a subset of them: 2D editors provide flexible appearance control but lack multi-view consistency, while 3D weather-editing methods provide consistency but are often limited to static post-weather effects.

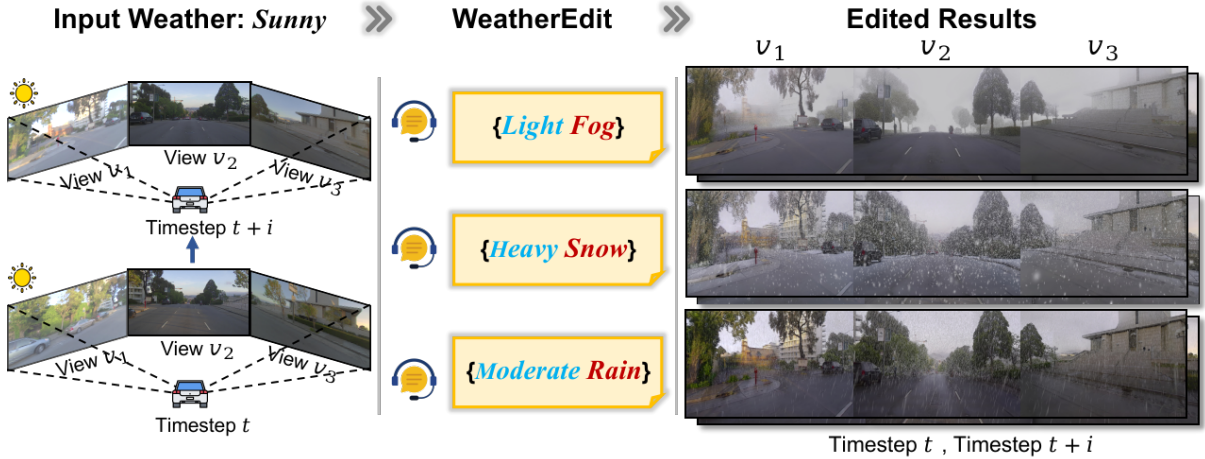


Figure 7.1: Given clear weather multi-view and multi-frame driving images rendered from an input 3D scene, WeatherEdit generates the same scene under user-specified weather conditions, including fog, snow, and rain, with controllable severity levels. The generated results preserve multi-view and multi-frame consistency while supporting both background appearance changes and dynamic weather effects.

7.1.3 The WeatherEdit Approach

WeatherEdit addresses the above requirements through a two-stage decomposition. The first stage, weather background editing, modifies the appearance of the reconstructed scene. It operates on rendered 2D views, but explicitly enforces consistency across cameras and frames. A pre-trained diffusion model is adapted to generate weather-specific background appearance, including foggy visibility, wet-road tones, snow-covered surfaces, and weather-dependent illumination. To avoid the inconsistency of independent frame editing, WeatherEdit introduces a consistency mechanism across temporal and view dimensions.

The second stage, weather particle construction, models dynamic weather directly in 3D. Weather particles, such as raindrops, snowflakes, and fog elements, are represented as Gaussian primitives in a particle field and rendered together with the reconstructed 3DGS scene. The particle field is equipped with physics-inspired dynamics so that particles evolve over time, and with local field alignment so that particle generation remains centred around the rendering camera. This allows the computational cost to scale with the visible field of view rather than the full spatial extent of the scene.

The two stages are complementary. Background editing captures persistent or slowly varying appearance changes, such as wet roads, foggy distance, altered illumination, and snow accumulation. Particle construction adds active dynamic effects, such as falling rain and drifting snow, at rendering time. In practice, the edited background views are first used to optimise a weather-styled 3DGS scene, while the Gaussian particle field is applied during rendering to overlay dynamic weather. Weather type and severity are controlled through text prompts in the background-editing stage and through explicit particle-field parameters in the dynamic-weather stage.

7.2 Related Work

2D weather synthesis. Image-level weather synthesis has been studied through both GAN-based and diffusion-based editing. CycleGAN-style methods (Hwang et al., 2022; Schmidt et al., 2019; Zhu et al., 2017) and GAN-based inpainting approaches (Schmidt et al., 2022) can translate individual images into different weather domains, but they operate independently on each frame and therefore cannot guarantee consistency across views or time. Recent diffusion-based editors, including ControlNet (Zhang et al., 2023b), T2I-Adapter (Mou et al., 2024), InstructPix2Pix (Brooks et al., 2023), and TurboEdit (Deutch et al., 2024), substantially improve visual realism and controllability. However, they inherit the same limitation: weather effects are generated in the image plane, so independently edited views may exhibit inconsistent fog density, snow accumulation, wet surfaces, or object appearance. This makes them unsuitable for simulation settings that require coherent multi-camera and multi-frame rendering. I include these diffusion-based editors

as direct baselines in the experiments.

3D weather synthesis. Physics-based simulation methods can model weather particles and fluid-like phenomena directly in 3D (Stomakhin et al., 2013; Zsolnai-Fehér, 2022; Feldman and O’Brien, 2002; Haas, 2014). They provide geometric consistency by construction, but their rendered outputs often lack the photorealistic appearance and scene-level semantic adaptation required for realistic driving simulation. Neural 3D weather synthesis has recently begun to address this gap. ClimateNeRF (Li et al., 2023a) incorporates static post-weather effects, such as accumulated snow and floodwater, into NeRF-reconstructed scenes. However, it does not model active weather events such as falling rain or drifting snow, and it is not formulated for 3D Gaussian Splatting. General 3D stylisation methods (Li et al., 2018b) can alter global scene appearance, but typically produce weak or style-oriented effects rather than controllable adverse weather phenomena. WeatherEdit instead combines weather-styled background editing with an explicit Gaussian particle field, enabling both static appearance changes and dynamic weather events within a 3DGS scene.

3D reconstruction for driving scenes. WeatherEdit builds on recent progress in 3D scene reconstruction, particularly 3D Gaussian Splatting (Kerbl et al., 2023). Driving scene reconstruction methods such as OmniRe (Chen et al., 2025) demonstrate that 3DGS can be used to construct scalable scene representations for autonomous driving environments. We previously introduced WeatherGS (Qian et al., 2025a), which reconstructs clean 3D scene assets from weather-degraded multi-view inputs. These reconstruction methods provide the scene representations on which WeatherEdit operates. In addition, dynamic 3DGS methods (Guo et al., 2024b; Luiten et al., 2024; Yang et al., 2024d; Chen et al., 2023; Zhang et al., 2024) show how Gaussian primitives can be extended with temporal behaviour. WeatherEdit draws on this representational flexibility to construct a dynamic weather particle field, while using OmniRe as the driving scene reconstruction backbone for the base scenes.

7.3 Stage 1: Weather Background Editing

7.3.1 All-in-One LoRA Adapter

A straightforward way to support multiple weather styles with a diffusion editor is to train a separate LoRA adapter (Hu et al., 2022) for each condition. CycleGAN-Turbo (Parmar et al., 2024) follows this style-specific design: each weather type, such as snow, rain, or fog, is represented by an independent LoRA module, and the corresponding adapter is selected at inference time. While this yields specialised style control, it scales linearly in training and storage cost with the number of supported weather types. More

importantly, independent adapters cannot share knowledge across related weather domains, even though they operate on the same driving scene distribution and often induce correlated appearance changes.

WeatherEdit instead introduces an All-in-One LoRA adapter that models multiple weather styles within a unified set of low-rank updates. The desired weather style is specified through the text prompt, while the adapter jointly learns shared and style-specific transformations. Let $W_0 \in \mathbb{R}^{d \times k}$ denote the pre-trained diffusion weights and x the input image. For weather style i (e.g., snowy, rainy, foggy), a low-rank matrix L_i is associated with the style, and the adapted forward pass computes

$$h = W_0 x + L_i x. \quad (7.1)$$

The style index i is recovered at inference from the text prompt by standard token matching. Joint training of the L_i matrices allows the model to share common weather-editing knowledge, such as driving scene structure and general surface appearance changes, while retaining style-specific capacity for effects such as snow accumulation, fog veiling, or rain-induced wetness.

Fig. 7.2 contrasts single-style adapters with the proposed All-in-One design.

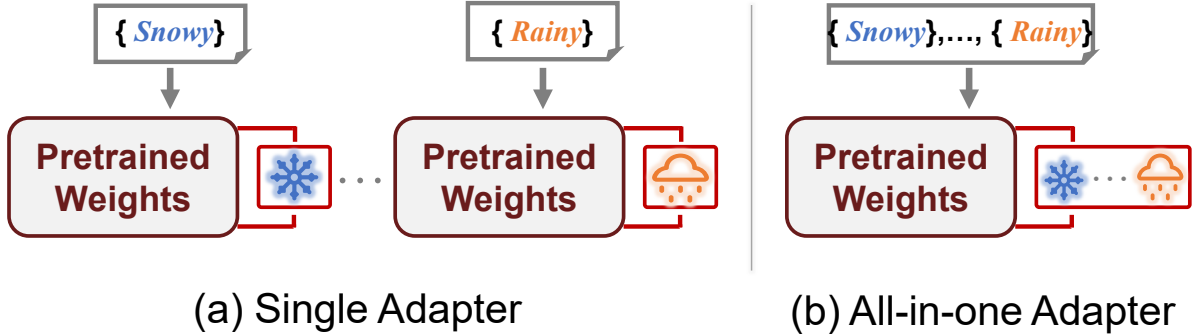


Figure 7.2: Single-style adapters versus the All-in-One adapter. The unified design supports multiple weather styles within one model, reducing training and storage cost while enabling knowledge sharing across related weather transformations.

Why a shared adapter rather than independent adapters. The All-in-One design is motivated by three considerations. First, weather-editing tasks share substantial structure. Snow, rain, and fog all modify the same underlying driving scene distribution, and their low-rank updates therefore contain common directions associated with scene layout, illumination shifts, and surface appearance changes. Joint adaptation acts as a cross-style regulariser, allowing shared factors to be learned collectively while reserving style-specific capacity for genuinely distinct effects. Second, a shared adapter naturally supports compositional editing. A prompt containing multiple weather-related descriptors can be handled within a single forward pass, whereas independent adapters would require heuristic adapter selection, interpolation, or ensembling. Third, the uni-

fied design reduces storage and serving overhead, which is important when the editor is deployed as part of a larger simulation or reconstruction pipeline. In this setting, joint adaptation is preferable to independent adaptation because the weather styles are related transformations of a common scene domain.

7.3.2 Semantic Conditioning

Weather effects are strongly scene-dependent. Snow accumulates on sidewalks, rooftops, and vegetation, but may leave active road surfaces wet or partially cleared. Rain produces reflections on smooth surfaces and puddles near road boundaries. Fog alters visibility in a depth-dependent manner and interacts differently with sky, buildings, and distant objects. These effects cannot be modelled reliably as spatially uniform style changes. The diffusion editor therefore requires structural guidance about the scene layout.

WeatherEdit conditions the adapter on a semantic segmentation map M of the input image x . The map is produced by an off-the-shelf segmentation model with Cityscapes-compatible labels, requiring no additional annotation. The segmentation-conditioned latent is denoted as $\mathbf{z}_0 = \mathcal{E}(x, M)$, where the Stable Diffusion VAE is augmented to accept the semantic channel. The training objective follows the standard diffusion reconstruction loss with semantic conditioning:

$$\mathcal{L}(\Delta\theta) = \mathbb{E}_{\epsilon, t} \left[\left\| \epsilon - \epsilon_{\theta + \Delta\theta} \left(\sqrt{a_t} \mathbf{z}_0 + \sqrt{1 - a_t} \epsilon, t, p \right) \right\|^2 \right], \quad (7.2)$$

where p is the text prompt and $\theta + \Delta\theta$ denotes the frozen diffusion backbone plus trainable LoRA parameters. Semantic conditioning encourages the editor to apply weather effects in geometrically and semantically plausible regions, for example placing snow on pavements and vegetation while preserving road surfaces as wet or partially visible rather than uniformly snow-covered.

7.3.3 Temporal-View Attention

Editing each camera view and time step independently leads to inconsistent outputs. Although each edited frame may be visually plausible, weather patterns, fog density, wet-surface reflections, and snow accumulation can vary incoherently across cameras and over time. Simulation-grade weather editing therefore requires information sharing during the diffusion denoising process across both the view and temporal dimensions.

To this end, WeatherEdit introduces a parameter-free Temporal-View (TV) attention mechanism that augments the diffusion model’s self-attention with view-aggregation and temporal-aggregation branches.

Spatial view attention. In a multi-camera driving rig, adjacent cameras typically share overlapping scene content. WeatherEdit uses the centre view as an anchor and

allows side-view queries to attend to the centre-view keys and values at the same time step:

$$\text{S_Attn}_t^v = \text{Softmax}\left(\frac{Q_t^v \cdot K_t^c, \top}{\sqrt{d}}\right) V_t^c, \quad (7.3)$$

where Q_t^v is the query from view v at time t , and K_t^c, V_t^c are the key and value from the centre view at the same time. This encourages weather appearance and scene-level edits to remain consistent across synchronised views.

Temporal attention. To reduce frame-to-frame flicker, each frame attends to its adjacent frames within the same view:

$$\text{T_Attn}_t^v = \text{Softmax}\left(\frac{Q_t^v \cdot K_v^{\mathcal{T}(t), \top}}{\sqrt{d}}\right) V_v^{\mathcal{T}(t)}, \quad (7.4)$$

with $\mathcal{T}(t) = \{t-1, t+1\}$. This local temporal aggregation propagates weather cues across neighbouring frames without requiring the diffusion model to process the full sequence jointly.

Combined attention. The original self-attention, spatial view attention, and temporal attention are combined as

$$\text{Attn}_t^v = \lambda \cdot \text{SelfAttn}_t^v + (1 - \lambda) \cdot (\text{S_Attn}_t^v + \text{T_Attn}_t^v). \quad (7.5)$$

The scalar λ balances local fidelity against cross-view and temporal consistency. A larger λ preserves the independence of each frame, while a smaller value increases information sharing across views and time. I use $\lambda = 0.5$ in my experiments, which provides a stable trade-off. Importantly, TV attention introduces no additional learnable parameters: it reuses the existing query, key, and value projections of the diffusion model and only changes the token set used during attention computation.

Fig. 7.3 illustrates the TV-attention mechanism.

7.4 Stage 2: Weather Particle Construction

7.4.1 Gaussian Particle Field

Real-world weather effects such as rain, snow, and fog contain a large number of local, dynamic particles. To reproduce these effects in a 3D scene, WeatherEdit models weather particles as explicit Gaussian primitives, analogous to the Gaussian representation used for the reconstructed scene but parameterised for transient weather phenomena rather than static geometry.

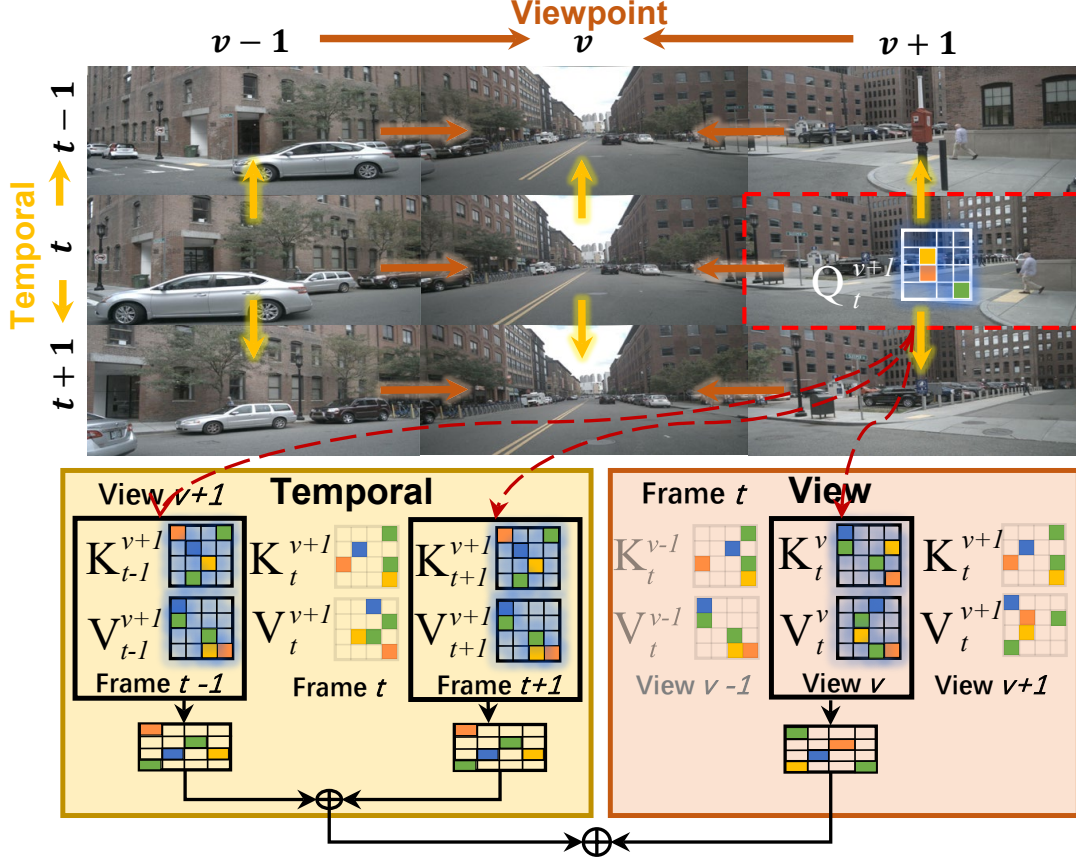


Figure 7.3: Temporal-View attention. Side cameras attend to the centre camera to improve view consistency, while each frame attends to adjacent frames to improve temporal consistency. The mechanism is parameter-free and augments the standard self-attention of the pre-trained diffusion model.

Each particle j is represented by an attribute tuple

$$A_j = \{C_j, P_j, R_j, S_j, O_j\}, \quad (7.6)$$

where C_j denotes RGB colour, $P_j \in \mathbb{R}^3$ is the 3D position, R_j is the rotation represented as a quaternion, $S_j \in \mathbb{R}^3$ is the anisotropic scale, and $O_j \in [0, 1]$ is the opacity. Particles are sampled within a bounded 3D particle field whose size B_i is specified for each weather type i . The total number of particles q controls the density of the field and therefore serves as the main severity parameter for precipitation-based weather. The weather field is defined as

$$W_i = \{\mathbb{G}(A_i), q, B_i\}, \quad (7.7)$$

where $\mathbb{G}(A_i)$ denotes the weather-specific distribution over particle attributes. This distribution is chosen according to the desired weather type: snow particles are approximately white with slight blue tint and irregular scale; raindrops are elongated and semi-transparent; fog particles are larger, grey, and more diffuse. The corresponding parameterisation is summarised in Table 7.1.

Fog is treated slightly differently from precipitation. Since fog is better described as

Table 7.1: Weather-specific particle parameterisation.

Weather	Colour	Shape	Dynamics	Severity driver
Rain	near-transparent grey	elongated (S high aspect ratio)	strong downward, small wind	q (particle count)
Snow	white, slight blue	roughly spherical	slow downward, strong lateral	q and drift intensity
Fog	uniform grey	large, high opacity	near-static, slight drift	β (attenuation coefficient)

volumetric attenuation than as a collection of visually distinct particles, WeatherEdit supplements the Gaussian field with the atmospheric transmission model $T(d) = e^{-\beta d}$, where β controls the attenuation strength. The fog particles are assigned large spatial scales and high opacity, producing soft volumetric scattering that complements the transmission-based visibility reduction.

7.4.2 Dynamic Simulation

To model active weather events, particle positions are updated over time using a constant-velocity motion model with weather-specific velocity distributions:

$$\mathbf{P}_j(t + \Delta t) = \mathbf{P}_j(t) + \mathbf{D}_j \cdot \Delta t, \quad (7.8)$$

where

$$\mathbf{P}_j(t) = [x(t) \ y(t) \ z(t)]^\top, \quad \mathbf{D}_j = [D_x \ D_y \ D_z]^\top. \quad (7.9)$$

The velocity vector $\mathbf{D}_j$ is sampled according to the selected weather type. Rain is assigned a strong downward component with limited horizontal drift; snow has a slower downward velocity and stronger lateral motion; fog is near-static, with only weak drift. This simple motion model captures the dominant temporal behaviour of common weather events while keeping the rendering cost low.

7.4.3 Particle Lifecycle: Emission, Motion, and Recycling

A finite particle field must maintain a stable particle density throughout the simulation. Without recycling, particles would eventually leave the field, reducing the apparent weather intensity over time. WeatherEdit therefore uses a torus-like recycling rule: when a particle exits the bounded field on one side, it is reintroduced from the opposite side, with a small randomised offset to avoid visible repetition:

$$\mathbf{P}_i(t) = \begin{cases} p_{\max} - \delta_p, & \text{if } p_i(t) < p_{\min}, \\ p_{\min} + \delta_p, & \text{if } p_i(t) > p_{\max}, \\ \mathbf{P}_i(t), & \text{otherwise,} \end{cases} \quad (7.10)$$

where $p_{\min}$ and $p_{\max}$ denote the bounds of the field along the corresponding dimension, and δ_p is a small offset. This recycling strategy keeps the particle count fixed at q , decoupling computational cost from sequence duration while preserving a stable weather

density.

7.4.4 Local Field Alignment

Driving scenes can extend over hundreds of metres. Placing particles throughout the entire reconstructed scene would require a very large number of Gaussians, most of which would not contribute to the rendered image because they lie outside the camera frustum. This is unnecessary for image formation and inefficient for rendering.

WeatherEdit instead uses a compact particle field aligned with the rendering camera. The field is positioned relative to the camera pose and moves with the camera over time. The local particle dynamics within the field remain unchanged, while the field’s global position follows the camera trajectory. Given the initial camera pose $\mathbf{T}_0$ and the camera pose $\mathbf{T}_t$ at time t , the relative transformation is

$$\Delta\mathbf{T} = \mathbf{T}_t \mathbf{T}_0^{-1}, \quad (7.11)$$

and the particle position is updated by

$$P_j(t) = \Delta\mathbf{R} P_j(t_0) + \Delta\mathbf{t}, \quad (7.12)$$

where $\Delta\mathbf{R}$ and $\Delta\mathbf{t}$ are the rotation and translation components of $\Delta\mathbf{T}$. This alignment keeps the particle field inside the visible region throughout the drive, allowing q to be set according to the particle density required for a single camera view rather than for the full scene volume. In typical driving scenes, this reduces the required number of particles by approximately two orders of magnitude compared with full-scene coverage.

Fig. 7.4 summarises the particle construction pipeline.

7.4.5 On the Constant-Velocity Approximation

The motion model in Eq. (7.8) uses a constant velocity for each particle, sampled at initialisation from a weather-specific distribution. This is a deliberate approximation. More detailed physical models, such as turbulence fields, wind shear, drag forces, or particle–surface interactions, could model weather dynamics more accurately, but would also substantially increase simulation and rendering cost.

Three considerations justify the constant-velocity approximation. First, the temporal window of a driving simulation is typically short. Over such horizons, precipitation particles often move close to terminal velocity, and the wind field can be treated as locally stable. A constant-velocity model therefore captures the dominant motion pattern of rain and snow at the level required for visual simulation. Second, although each particle follows a simple trajectory, the population-level motion remains diverse because velocities are sampled from a distribution. This introduces natural variation in falling speed and lateral drift without requiring an explicit turbulence simulator. Third, the update rule is

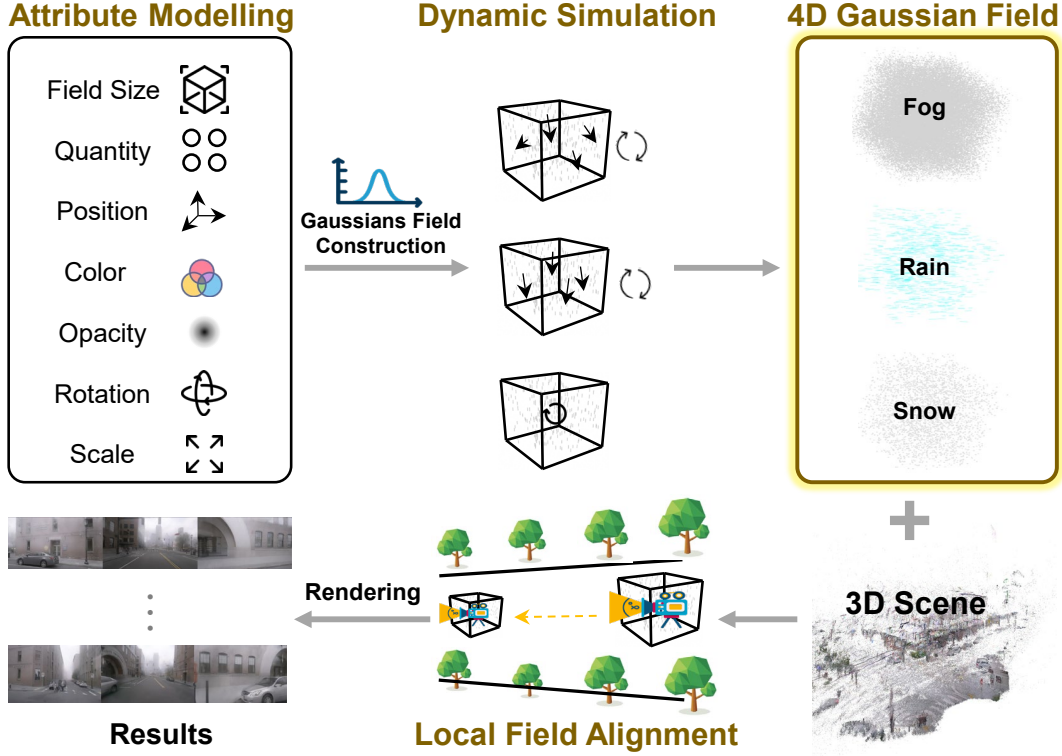


Figure 7.4: Weather particle construction pipeline. Attribute modelling defines weather-specific particle distributions; dynamic simulation updates particles over time; local field alignment keeps the particle field centred around the rendering camera. The resulting Gaussian particle field is rendered together with the reconstructed 3D scene to produce the final weather-edited view.

computationally lightweight, enabling dense particle fields to be rendered alongside the 3DGS scene without dominating the runtime.

This approximation also has clear limitations. Strong gusts of wind, which induce correlated motion among neighbouring particles, are not explicitly modelled. Particle-particle interactions are ignored, which is reasonable for moderate precipitation but less accurate in extreme storms. Particle-surface interactions, such as raindrops hitting a windshield or snow accumulating on a parked vehicle, are also not handled by the particle field. In WeatherEdit, such post-event appearance changes are instead captured by the Stage-1 background editing process. The framework therefore separates dynamic particle rendering from persistent appearance modification, with the limitation that fine-grained particle-surface interactions are not explicitly represented.

7.5 Experimental Setup

7.5.1 Datasets

Training data (adapter). The All-in-One adapter is trained on 1,237 image pairs extracted from BDD100K (Yu et al., 2020), MUSES (Brödermann et al., 2024), and ACDC (Sakaridis et al., 2021). Each pair consists of a normal-weather image and an

adverse weather image (snowy, rainy, or foggy), accompanied by a ground-truth semantic segmentation map.

Evaluation data. For evaluation I select eight scenes with multi-view image sequences from Pandaset (Xiao et al., 2021), Waymo Open (Sun et al., 2020), nuScenes (Caesar et al., 2020), and KITTI-360 (Liao et al., 2023). These cover urban streets, highways, and suburban roads, providing a diverse testbed for weather editing quality.

7.5.2 Baselines

I compare against two groups of prior methods. **2D-based editors:** ControlNet (Zhang et al., 2023b), InstructPix2Pix (Brooks et al., 2023), and TurboEdit (Deutch et al., 2024). **3D-based editors:** 3D Stylisation (Li et al., 2018b) and ClimateNeRF (Li et al., 2023a).

7.5.3 Evaluation Metrics

Editing quality. I use two CLIP-based (Radford et al., 2021) metrics:

- **CLIP-S** (CLIP similarity): cosine similarity between CLIP image embeddings of the original and edited images, measuring how well the scene content is preserved. Higher is better.
- **CLIP-DS** (Directional CLIP Similarity) (Gal et al., 2022): measures the alignment between the editing direction in image space and in text space. Higher is better; this penalises edits that are cosmetically different but do not move in the intended direction.

Consistency. Temporal consistency is measured by the warp error between adjacent frames (lower is better); spatial consistency by the Bhattacharyya distance between colour distributions across camera views (lower is better).

Downstream utility. To measure whether WeatherEdit-generated data improves perception robustness, I train a segmentation model with and without WeatherEdit augmentation and report **mIoU** on ACDC and MUSES adverse weather benchmarks.

Perceptual quality. A user study with 30 participants selects the most realistic result across WeatherEdit and baselines for each weather condition.

7.5.4 Implementation

The 2D editing stage is implemented on top of CycleGAN-Turbo (Parmar et al., 2024) as the diffusion backbone, with the All-in-One LoRA adapter as described in Section 7.3.

Training takes 12 hours on a single NVIDIA A6000. Inference runs at 3.2 seconds per frame. 3D scene reconstruction uses the OmniRe (Chen et al., 2025) driving-specific 3DGS framework. Gaussian particle field simulation and rendering together take approximately 0.21 seconds per frame on a single NVIDIA RTX 3090.

7.6 Results

7.6.1 Comparison with 2D-Based Editors

Fig. 7.5 and Table 7.2 compare WeatherEdit against three 2D-based editors on the 2D background editing alone (particle simulation disabled).

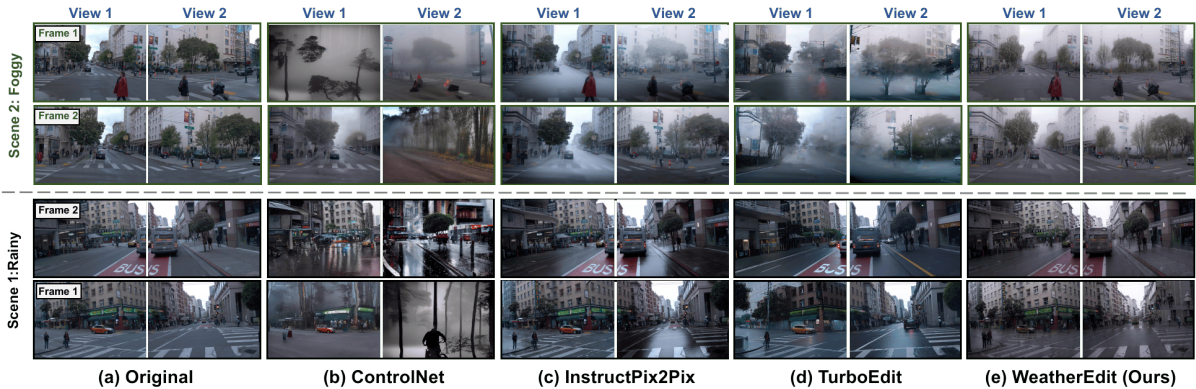


Figure 7.5: Qualitative comparison with 2D-based editors. ControlNet over-stylises and modifies scene content; InstructPix2Pix shows inconsistency across adjacent frames and views; TurboEdit stylises correctly but distorts scene content (pedestrians become reflections, trees disappear). WeatherEdit preserves scene integrity and achieves multi-view and multi-frame consistency.

Table 7.2: Comparison with 2D-based weather editors. Higher is better.

	ControlNet	IP2P	TurboEdit	Ours
CLIP-DS $\uparrow$	0.291	0.295	0.257	0.302
CLIP-S $\uparrow$	0.594	0.691	0.676	0.750

WeatherEdit achieves the highest CLIP-DS (0.302) and highest CLIP-S (0.750) simultaneously, indicating that it both moves in the correct editing direction and preserves scene content better than any of the 2D baselines. The qualitative results in Fig. 7.5 make the content-preservation advantage particularly visible: ControlNet tends to apply an overall colour wash that overwrites scene content, InstructPix2Pix produces frame-to-frame inconsistencies, and TurboEdit aggressively modifies scene elements (pedestrians, trees). WeatherEdit preserves the scene while modifying the atmosphere.

7.6.2 Comparison with 3D-Based Editors

Fig. 7.6 and Table 7.3 compare WeatherEdit against two 3D-based editors on the snowy editing task.

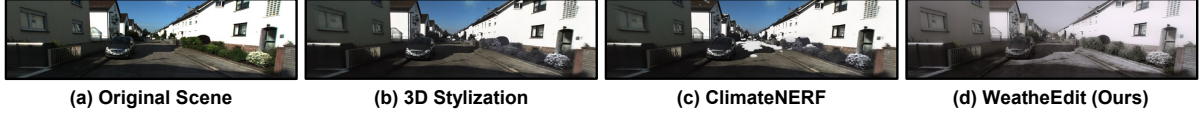


Figure 7.6: Qualitative comparison with 3D-based weather editors on snowy editing. 3D Stylisation produces a faint effect; ClimateNeRF produces unnatural snow cover without matching the atmospheric tone; WeatherEdit produces realistic snow with appropriate atmospheric context.

Table 7.3: Comparison with 3D-based weather editors. The high CLIP-S score of 3D Stylisation reflects minimal scene modification (it fails to edit) rather than editing quality.

	3D Stylisation	ClimateNeRF	Ours
CLIP-DS $\uparrow$	0.225	0.267	0.283
CLIP-S $\uparrow$	0.842	0.702	0.751

3D Stylisation achieves the highest CLIP-S because it barely modifies the scene; the high similarity reflects the failure to edit rather than successful editing. The CLIP-DS score, which is the more informative metric here since it captures whether the edit moves in the intended direction, clearly favours WeatherEdit (0.283 versus 0.267 and 0.225). ClimateNeRF produces a visible snow accumulation but does not modify the atmospheric tone, so its edits feel partial.

7.6.3 Controllability

A key differentiator of WeatherEdit is the explicit severity control exposed through the particle field’s q parameter. Fig. 7.7 demonstrates this across three driving datasets: for each original scene, WeatherEdit generates a 2D-edited background and then composites particle fields at light, moderate, and heavy severity. The user-facing interface is a single slider that adjusts q continuously; the underlying rendering produces visually coherent transitions from barely visible drizzle to dense storm.

The controllability spans all three C-requirements: type control via the text prompt to the All-in-One adapter; event control via the combination of background editing (for wet-road, accumulated-snow appearance) and particle field (for active rain, falling snow); and severity control via the q parameter (for all three weather types) and the fog attenuation coefficient β (for fog specifically).

7.6.4 Downstream Utility for Perception

Beyond visual quality, a useful weather-editing framework should improve the robustness of downstream perception models. I evaluate this by augmenting the training data with

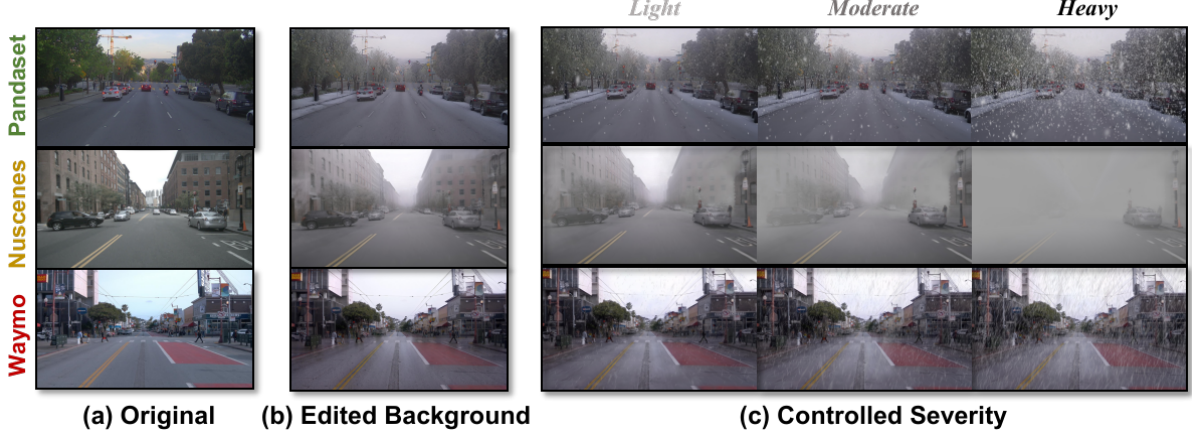


Figure 7.7: Controllable weather effect and severity across three driving datasets. WeatherEdit generates (a) the original scene, (b) the 2D-edited background, and (c) the composited particle fields at light, moderate, and heavy severity.

WeatherEdit-generated images and measuring semantic segmentation performance on real adverse weather benchmarks. Specifically, I train HRDA (Hoyer et al., 2024) and MIC (Hoyer et al., 2023) with and without WeatherEdit augmentation, and report mIoU on ACDC (Sakaridis et al., 2021) and MUSES (Brödermann et al., 2024) in Table 7.4.

Table 7.4: Semantic segmentation mIoU (%) with and without WeatherEdit-generated data augmentation.

	ACDC		MUSES	
	w/o	w	w/o	w
HRDA (Hoyer et al., 2024)	46.3	58.7 (+ 12.4)	42.2	57.1 (+ 14.9)
MIC (Hoyer et al., 2023)	50.0	59.0 (+ 9.0)	48.8	55.4 (+ 6.8)

WeatherEdit augmentation yields consistent gains across both segmentation backbones and both evaluation datasets. The improvement reaches +12.4 mIoU on ACDC and +14.9 mIoU on MUSES for HRDA, and remains substantial for MIC, with gains of +9.0 and +6.8 mIoU, respectively. These results show that the generated weather scenes are not only visually plausible, but also useful for training models that generalise to real adverse weather data.

The downstream gains also complement the findings of WeatherDG in Chapter 6. While WeatherDG generates independent 2D images for per-frame augmentation, WeatherEdit produces multi-view and temporally consistent weather-edited scenes. This makes WeatherEdit more suitable for simulation-oriented and multi-camera perception settings, where consistency across views and frames is required. The fact that augmentation improves performance on real benchmarks further indicates that WeatherEdit does not merely introduce synthetic visual diversity, but captures weather-related appearance changes that transfer to real adverse weather domains.

7.6.5 User Study

I further assess perceptual realism through a user study. It was given ethical approval by the University of Leeds Faculty Research Ethics Committee, reference number BESS+ FREC - 2025 2539-3449. The participants were recruited from the research groups of two of the co-authors, excluding any individuals involved in the research project. For each weather condition, participants were shown nine scenes edited by WeatherEdit and by a randomised set of baseline methods, and were asked to select the most realistic result. The preference scores are reported in Fig. 7.8.

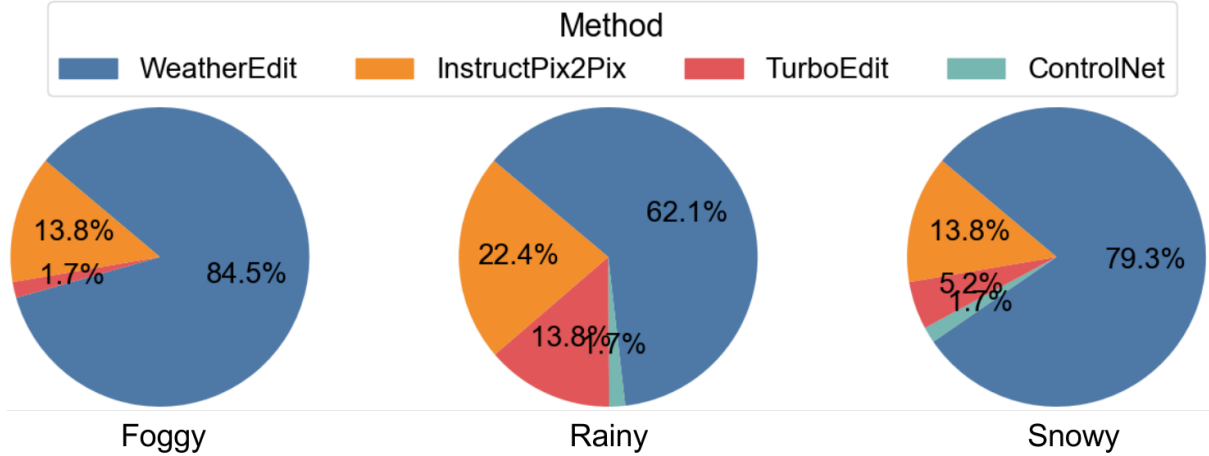


Figure 7.8: User preference for the most realistic weather editing results across three weather conditions. WeatherEdit receives the majority of votes in all settings: 84.5% for fog, 62.1% for rain, and 79.3% for snow.

WeatherEdit is preferred across all three weather types. The margin is largest for fog, where the 3D-consistent atmospheric appearance is especially important. The preference score is lower for rain, although WeatherEdit still receives the majority of votes. This is expected because some visually salient rain effects, such as streak-like patterns, can be partially reproduced by strong 2D editors. In contrast, fog requires coherent scene-level visibility changes, and snow requires spatially distributed accumulation, both of which benefit more directly from 3D-consistent editing.

7.7 Ablation Studies

7.7.1 Effect of Semantic Conditioning

I first evaluate the contribution of semantic conditioning in the weather background editor. Fig. 7.9 compares editing results with and without the semantic segmentation map as conditioning input.

Semantic conditioning improves the spatial plausibility of the edited weather. With semantic guidance, snow accumulates on sidewalks, vegetation, and other appropriate surfaces, while road regions remain more visible or wet. Without semantic conditioning,

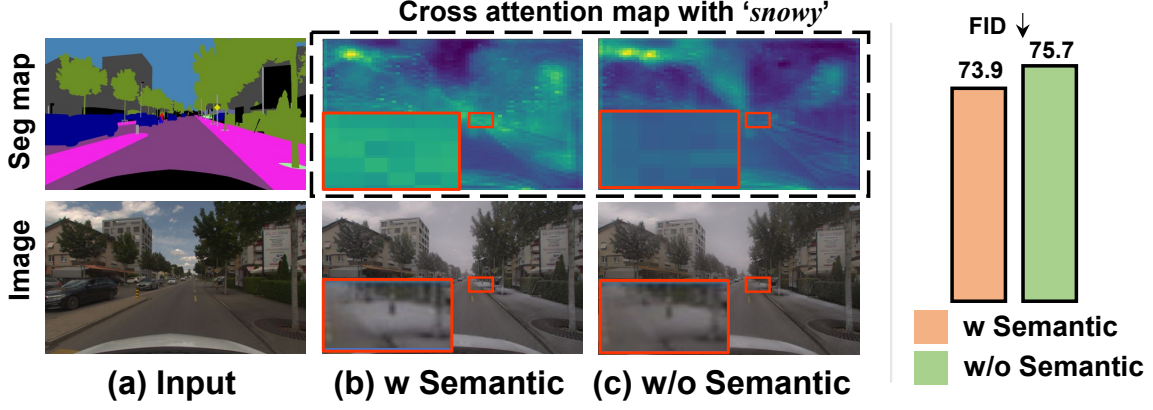


Figure 7.9: Effect of semantic conditioning. (a) Input image and semantic map. (b) With semantic conditioning, snow is placed in semantically plausible regions such as sidewalks, trees, and non-active surfaces. (c) Without semantic conditioning, snow is distributed more uniformly, including over road regions where it should be reduced or cleared.

the editor tends to distribute snow uniformly across the image, including over active road surfaces. This confirms that semantic maps provide useful structural constraints for weather placement, preventing weather effects from becoming a purely global style transfer.

7.7.2 Effect of Temporal-View Attention

I next analyse the contribution of Temporal-View attention. Fig. 7.10 qualitatively compares editing results with and without TV attention, while Table 7.5 reports quantitative consistency metrics.

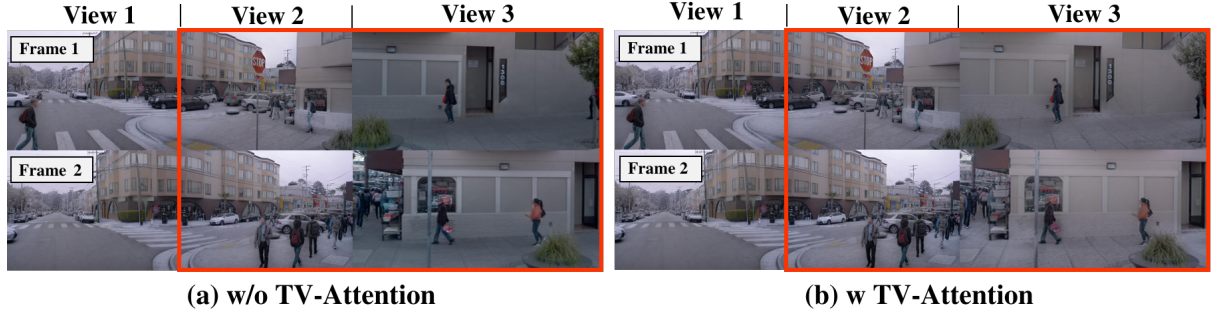


Figure 7.10: Effect of Temporal-View attention. (a) Without TV attention, the right-most view has a noticeably darker tone than the adjacent view, leading to cross-view inconsistency. (b) With TV attention, the edited appearance remains more consistent across views.

The results show that the two attention branches target complementary consistency axes. Temporal attention produces the lowest warp error, indicating improved frame-to-frame coherence. View attention reduces cross-view distribution mismatch, as reflected by the lower Bhattacharyya distance. The full TV-attention mechanism achieves the best cross-view consistency while maintaining competitive temporal consistency, confirming that view and temporal aggregation provide complementary benefits.

Table 7.5: Effect of TV attention components. Temporal attention reduces warp error between adjacent frames, while spatial view attention reduces Bhattacharyya distance across views. The full mechanism provides the best overall consistency.

	Self-attn	+View	+Temp	Full
Warp error ↓	0.043	0.042	0.039	0.041
Bhattacharyya dist ↓	0.272	0.253	0.261	0.245

7.7.3 Effect of the Particle-Field Modules

Finally, I ablate the components of the weather particle construction stage in Fig. 7.11.

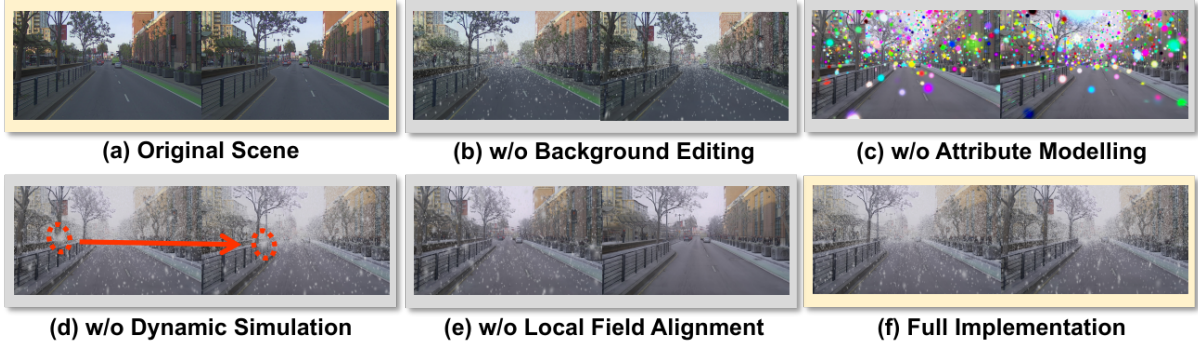


Figure 7.11: Ablation of the particle-field modules. (a) Original scene. (b) Without 2D background editing, dynamic particles appear over an otherwise clear scene. (c) Without attribute modelling, particles appear as implausible coloured blobs. (d) Without dynamic simulation, particles are visible but static. (e) Without local field alignment, the camera moves out of the particle field and the weather effect disappears. (f) The full system produces natural, camera-aligned, and dynamic weather.

Each component addresses a distinct failure mode. Without background editing, the dynamic particles are not supported by the expected atmospheric tone or post-weather appearance, making the result visually inconsistent. Without weather-specific attribute modelling, particles lack realistic colour, scale, opacity, and shape. Without dynamic simulation, the particles remain static and fail to represent active weather. Without local field alignment, the particle field does not follow the camera, causing the weather density to decay as the camera moves through the scene. The full model combines these components to produce dynamic weather that remains visually plausible, temporally coherent, and aligned with the rendered viewpoint.

7.8 Summary

This chapter presented WeatherEdit, a controllable 3D weather-editing framework for synthesising realistic, multi-view and multi-frame consistent weather effects in reconstructed driving scenes. WeatherEdit follows a two-stage design. The first stage performs weather background editing using an All-in-One LoRA adapter and Temporal-View attention, producing weather-styled scene appearance while preserving cross-view

and temporal consistency. The second stage constructs dynamic weather effects through a Gaussian particle field with physics-inspired motion and local field alignment. Together, these components satisfy the realism and controllability requirements defined at the beginning of the chapter: distortion-free editing, temporal-view consistency, dynamic weather effects, and controllable weather type, event state, and severity. Experiments across four driving datasets demonstrate state-of-the-art editing quality, including 84.5% user preference under foggy conditions. When used for data augmentation, WeatherEdit-generated images further improve downstream semantic segmentation by up to 14.9 mIoU points on adverse weather benchmarks.

WeatherEdit completes the 3D generation component of the thesis framework. Together with AllWeatherNet, WeatherDG, and WeatherGS, it occupies the last of the four positions in the framework introduced in Chapter 3: 2D enhancement, 2D generation, 3D enhancement, and 3D generation. These four methods address different but complementary aspects of weather-robust autonomous driving. AllWeatherNet enhances degraded 2D observations at inference time; WeatherDG expands the training distribution through 2D adverse weather generation; WeatherGS reconstructs clean 3D scene assets from weather-degraded multi-view captures; and WeatherEdit synthesises controllable adverse weather effects within reconstructed 3D scenes for simulation and testing.

The next and final chapter of the thesis returns to the framework as a whole. It analyses the common principles that connect the four methods, discusses their limitations and broader implications, and outlines future directions for generative and spatial modelling of adverse weather in autonomous driving systems.

Chapter 8

Discussion and Conclusion

The preceding four chapters have presented the technical contributions of this thesis, each addressing one component of the framework introduced in Chapter 3. Part II focused on Enhancement: AllWeatherNet (Chapter 4) restores weather-degraded images at inference time, while WeatherGS (Chapter 5) reconstructs clean 3D scene assets from weather-degraded multi-view captures. Part III focused on the complementary capability of Generation: WeatherDG (Chapter 6) synthesises adverse weather images for domain generalisation, while WeatherEdit (Chapter 7) generates controllable weather effects within reconstructed 3D scenes.

Each method was developed and evaluated in its own chapter. Taken individually, the four chapters constitute separate contributions to weather-robust perception, reconstruction, data augmentation, and simulation. Considered jointly, however, they support a broader argument: weather robustness in autonomous driving is not a single problem, but a set of coupled problems spanning observation enhancement, data generation, image-level modelling, and scene-level modelling. The primary contribution of this thesis is therefore not only the four methods themselves, but also the organising framework that connects them.

This final chapter examines the four contributions as a single body of work. It identifies the common principles that recur across the methods, analyses the limitations that remain, and discusses the broader implications for autonomous driving safety, simulation, and weather-robust computer vision.

The remainder of the chapter is organised as follows. Section 8.1 summarises the four technical contributions and the artefacts produced by each. Section 8.2 develops the cross-chapter themes that emerge from analysing the methods jointly. Section 8.3 discusses limitations of the framework and its individual components. Section 8.4 considers broader implications for autonomous driving safety, simulation, and adverse weather perception. Section 8.5 outlines promising directions for future research, and Section 8.6 concludes the thesis.

8.1 Summary of Contributions

The thesis makes four methodological contributions with corresponding results chapters. I summarise each.

Part II — Enhancement

8.1.1 AllWeatherNet: Unified 2D Image Enhancement

AllWeatherNet (Chapter 4) is a unified generative-adversarial framework that enhances images degraded by rain, snow, fog, and low light, within a single network. Its three architectural contributions (the Scaled Illumination-Aware Attention Mechanism, the hierarchical scene-object-texture discriminator, and the Ranked Adaptive Window Pairing training strategy) together produce, on ACDC, a 38.2% mIoU (up from the 32.9% mIoU of the uncorrected baseline) and the lowest NIQE and highest SSIM among all evaluated methods. The method works at inference time, requires no change to the downstream segmentation model, and generalises to held-out datasets (Foggy Zurich, Nighttime Driving) without retraining. AllWeatherNet was published at ICPR 2024 (Qian et al., 2024) and received the Piero Zamperoni Overall Best Student Paper Award.

8.1.2 WeatherGS: 3D Reconstruction from Weather-Degraded Observations

WeatherGS (Chapter 5) is the first 3D Gaussian Splatting framework for clean-scene reconstruction from multi-view images corrupted by adverse weather. Its two preprocessing contributions, the Atmospheric Effect Filter (AEF) for dense weather particles and the Lens Effect Detector (LED) for sparse lens occlusions, combined with mask-guided 3DGS optimisation, produce clean reconstructions on a new benchmark comprising six synthetic and two real-world weather-degraded scenes. Compared with NeRF-based alternatives, WeatherGS achieves approximately $14\times$ faster training and $1,600\times$ faster rendering, with better reconstruction quality on all three metrics (PSNR, SSIM, LPIPS). WeatherGS was published at ICRA 2025 (Qian et al., 2025a).

Part III — Generation

8.1.3 WeatherDG: LLM-Assisted 2D Data Generation

WeatherDG (Chapter 6) is a data-generation framework that produces photorealistic adverse weather training images without requiring any real adverse weather data. Its two architectural contributions (DreamBooth fine-tuning of Stable Diffusion on a single source-domain image, and a three-agent LLM pipeline (Instance Sampler, Scene Composer, Scene Descriptor) for procedural prompting) together enable the generation of

driving scene images with diverse weather conditions, balanced object classes, and detailed scene descriptions. On three target benchmarks (ACDC, BDD100K, Dark Zurich), segmentation models trained with WeatherDG augmentation improve by up to 13.9, 7.6, and 9.5 mIoU points respectively over clean-trained baselines, and by 3–9 points over other diffusion-based data augmentation methods. WeatherDG was published in IEEE Robotics Automation Letters (Qian et al., 2025b).

8.1.4 WeatherEdit: Controllable 3D Weather Generation

WeatherEdit (Chapter 7) is a controllable weather-editing framework that synthesises rain, snow, and fog into reconstructed 3D driving scenes with full multi-view and multi-frame consistency, dynamic particle effects, and fine-grained severity control. Its three architectural contributions (the All-in-One LoRA adapter for unified multi-weather style synthesis, the parameter-free Temporal-View (TV) attention for multi-camera multi-frame consistency, and the Gaussian particle field with local camera alignment for efficient dynamic-particle rendering) together deliver state-of-the-art editing quality across four driving datasets. User studies show 84.5% preference on foggy scenes; downstream semantic-segmentation mIoU improves by up to 14.9 points when WeatherEdit-generated data is used for training augmentation. WeatherEdit was accepted to AAAI 2026 (Qian et al., 2026).

Table 8.1: The four methods of the thesis, grouped by Part (capability) and then by dimensionality, characterised along five axes.

Part	Method	Cell	Function	Model stack	System-level role	Headline result
II. Enhancement	AllWeatherNet	2D Enh.	Inference-time restoration	GAN	Perception enhancement	+5.3 mIoU on ACDC
	WeatherGS	3D Enh.	Clean-scene recovery	Diffusion + 3DGS	Scene reconstruction	14× faster training
III. Generation	WeatherDG	2D Gen.	Training-data synthesis	LLM + Stable Diffusion	Scene understanding	+13.9 mIoU on ACDC
	WeatherEdit	3D Gen.	Controllable weather synthesis	LoRA + 3DGS + particles	Scene simulation	+14.9 mIoU on MUSES

The framework is the thesis’s interpretive contribution. It is what makes the four individual publications, each standing on its own merits, into a coherent research programme organised around two complementary capabilities. The title of the thesis, *Generative and Spatial Modelling*, names the two modelling families that together span the framework: the 2D cells use only generative modelling, the 3D cells combine generative modelling with an explicit spatial representation.

8.2 Cross-Chapter Analysis

Five cross-cutting themes emerge when the four methods are considered jointly.

8.2.1 Theme 1: From Enhancement to Generation

A recurring empirical pattern across the thesis is that training-time generation produces larger downstream gains than inference-time enhancement. As shown in Table 8.2, All-

Table 8.2: Comparison between inference-time enhancement and training-time generation for adverse-weather semantic segmentation on ACDC. Values report mIoU, with improvements over the source-only baseline shown in parentheses.

Backbone	Source-only	AllWeatherNet <i>Enhancement</i>	WeatherDG <i>Generation</i>
DAFormer	47.6	48.4 (+0.8)	53.3 (+5.7)
HRDA	46.3	53.2 (+6.9)	60.2 (+13.9)

WeatherNet, the 2D enhancement method, provides moderate improvements over the source-only baseline, whereas WeatherDG, the 2D generation method, leads to substantially larger gains under the same evaluation setting. This pattern suggests that adverse weather robustness benefits more from expanding the training distribution than from restoring degraded observations only at inference time.

Root cause versus symptom. Enhancement addresses the immediate symptom of adverse weather: the degraded input image. It attempts to map a corrupted observation back towards a cleaner visual domain before the image is processed by the segmentation model. However, it does not change the training distribution from which the segmentation model learned its representations. If adverse weather appearances are under-represented during training, the model may remain sensitive to weather-induced features even after restoration. Generation addresses the root cause more directly by expanding the training distribution. By exposing the segmentation model to diverse weather conditions during training, it encourages the model to learn representations that are robust to weather variation rather than relying on a separate restoration stage at inference time.

Coverage versus restoration. Enhancement assumes that a degraded input can be transformed into a distribution already familiar to the downstream model, typically the clear weather distribution. This assumption is inherently limited when the deployment domain contains visual patterns that cannot be perfectly restored, such as low visibility, wet reflections, snow-covered surfaces, or night-time illumination. Generation instead expands the set of conditions represented during training. In this view, weather robustness is primarily a coverage problem: downstream performance depends on whether the training distribution spans the visual conditions encountered at deployment.

Cascaded versus direct learning. Enhancement also introduces a cascaded error structure. Any artefact introduced during restoration becomes part of the input to the segmentation model and can propagate into the final prediction. Generation avoids this inference-time dependency. The segmentation model receives the original test image directly, but has been trained to interpret similar adverse weather patterns through synthetic or edited training examples. This makes the learned robustness part of the segmentation model itself, rather than a property of an upstream preprocessing module.

Together, these factors explain the transition across the thesis from enhancement-only solutions towards generation-based methods. Enhancement remains useful when degraded observations must be improved directly, but generation provides a stronger mechanism for improving the robustness of downstream perception models.

8.2.2 Theme 2: Generative and Spatial Modelling Are Complementary

A second theme concerns the distinction between 2D and 3D methods. WeatherDG demonstrates that 2D generation can substantially improve single-image semantic segmentation. However, because each generated image is independent, the method cannot guarantee consistency across multiple cameras or consecutive frames. This limits its use in simulation, multi-camera perception, and closed-loop validation, where consistency across observations is essential.

WeatherEdit addresses this limitation by combining generative modelling with an explicit spatial representation. Like WeatherDG, it uses generative models to synthesise adverse weather appearance. Unlike WeatherDG, it embeds this generation process within a 3D scene representation, allowing rendered views to share a common underlying state. Multi-view and multi-frame consistency are therefore not post-processing objectives, but structural consequences of the representation.

This distinction reflects the core motivation behind the thesis title. In the 2D cells of the framework, the primary modelling tool is generative: GAN-based enhancement in AllWeatherNet and diffusion-based image generation in WeatherDG. In the 3D cells, generative models are combined with spatial representations, particularly 3D Gaussian Splatting. The generative component provides appearance synthesis or restoration, while the spatial component provides geometric coherence across viewpoints and time.

The broader implication is that generative models alone are insufficient for tasks requiring consistency across observations. For multi-camera perception, video perception, and simulation-based testing, generative methods must be paired with spatial representations that impose consistency at the level of the scene. The thesis therefore argues not for generative modelling or spatial modelling in isolation, but for their combination.

8.2.3 Theme 3: Composability Across the Framework

The four methods are not mutually exclusive alternatives; they are composable components within a broader weather-robust perception and simulation pipeline. The most direct composition occurs along the 3D axis: WeatherGS reconstructs clean 3D scene assets from weather-degraded captures, and WeatherEdit subsequently injects controllable weather effects into those reconstructed assets. In this sense, the output of 3D enhancement becomes the input to 3D generation.

Composability also appears along the enhancement–generation axis. WeatherDG and WeatherEdit both improve the training distribution, but serve different deployment settings. WeatherDG produces independent adverse weather images for single-camera or per-frame segmentation training, while WeatherEdit produces multi-view and temporally consistent renderings suitable for multi-camera perception and simulation. AllWeatherNet is orthogonal to both: it can be used at inference time to enhance degraded inputs regardless of whether the segmentation model was trained with WeatherDG or WeatherEdit augmentation.

A complete weather-robust system could therefore combine all four capabilities. WeatherDG- or WeatherEdit-generated data could be used during training to improve adverse weather robustness; WeatherGS could reconstruct clean 3D assets from real-world captures; WeatherEdit could use those assets for controlled simulation; and AllWeatherNet could serve as an inference-time enhancement module when the deployed system encounters degraded visual inputs. This composability is what makes the framework practically useful. The four methods are not competing choices, but stackable capabilities.

8.2.4 Theme 4: The Increasing Role of Foundation Models

The four methods also reflect the increasing role of foundation models in computer vision over the period covered by this thesis. AllWeatherNet uses no foundation model; it trains a task-specific GAN from scratch using illumination-aware attention, hierarchical discrimination, and robust patch pairing. This reflects an earlier paradigm in which weather restoration methods were largely specialised architectures trained for a particular degradation setting.

WeatherDG shifts towards foundation-model reuse. It uses Stable Diffusion as a generative image prior and an LLM for prompt construction. DreamBooth fine-tuning adapts the diffusion model with a small number of trainable parameters, while the LLM is used through prompting rather than weight updates. WeatherDG therefore demonstrates how large pre-trained models can be adapted and orchestrated for a specific downstream objective, namely adverse weather domain generalisation.

WeatherGS uses foundation-model components more selectively. Its Atmospheric Effect Filter relies on a pre-trained diffusion-based restoration prior, while its Lens Effect Detector builds on an attention-based raindrop detection model. These modules serve as preprocessing components for a 3DGS reconstruction pipeline, rather than as the defining representation of the method. The 3DGS backbone itself is not a foundation model, but functions as a general-purpose spatial representation that can be combined with learned restoration modules.

WeatherEdit uses the most foundation-model-oriented stack. It adapts a pre-trained diffusion editor through LoRA, uses text prompts for controllable weather style, and combines this generative prior with a 3DGS-based scene representation and a Gaussian

particle field. It therefore represents the strongest example in the thesis of model composition: rather than training a single monolithic model, the method combines pre-trained diffusion, parameter-efficient adaptation, semantic conditioning, attention-based consistency mechanisms, and explicit 3D particle modelling.

This progression reflects a broader shift in the field. Weather-robust vision has moved from training specialised models for individual degradations towards composing pre-trained models and spatial representations for task-specific objectives. In this setting, research value increasingly lies in how models are adapted, constrained, and composed, rather than solely in the design of a standalone network architecture.

8.2.5 Theme 5: Relation to General-Purpose World Models

Recent general-purpose world models provide a broader context for this thesis. Atlas, introduced by World Labs in 2026, integrates text, images, camera poses, video frames, and depth maps within a shared spatial context, supporting controllable generation, sparse-view reconstruction, space-time simulation, and explicit 3D outputs (World Labs Team, 2026). It therefore reflects the convergence of generative and spatial modelling that also motivates this thesis.

However, Atlas is a general-purpose world model, whereas the methods presented here address weather-robust autonomous driving specifically. AllWeatherNet restores weather-degraded images; WeatherGS separates atmospheric and lens effects during clean 3D reconstruction; WeatherDG generates class-balanced adverse-weather data for semantic segmentation; and WeatherEdit provides explicit control over weather type, severity, and dynamics. These weather-specific mechanisms and downstream perception evaluations distinguish this thesis from Atlas. A promising future direction is to integrate such domain-specific constraints into general-purpose world models, combining broad spatial priors with controllable weather simulation and perception-oriented evaluation.

8.3 Limitations

This thesis has several limitations, which can be grouped into method-specific limitations and broader limitations of the overall framework. I first discuss the limitations of each technical contribution.

8.3.1 Method-Specific Limitations

AllWeatherNet. AllWeatherNet relies on loosely paired adverse weather and clear weather images during training. Although Ranked Adaptive Window Pairing relaxes the need for pixel-level alignment, the method still requires some form of correspondence between degraded and clean images. It is therefore less directly applicable to datasets in which no paired or approximately corresponding observations are available. In addition,

the GAN-based formulation inherits the optimisation instability commonly associated with adversarial training. While the method performs robustly under the hyperparameter settings used in this thesis, applying it to substantially different datasets or weather regimes may require additional tuning. Finally, AllWeatherNet operates on individual frames and does not explicitly model temporal consistency. When applied to video sequences, this frame-wise formulation may lead to temporal flickering or inconsistent enhancement across consecutive frames.

WeatherDG. The effectiveness of WeatherDG is limited by the quality of pseudo-labels produced during UDA-style training. The ablation study shows that performance saturates at approximately 1,000 generated images, suggesting that the main bottleneck is not the number of synthetic samples but the reliability of the downstream training signal. This limitation is particularly visible on Dark Zurich, where performance remains lower than on ACDC or BDD100K. Since the labelled source domain is predominantly daytime, pseudo-labels generated for synthetic nighttime images can be noisy, reducing the benefit of additional nighttime data. WeatherDG also depends on the ability of the LLM pipeline to follow class-balancing and scene-composition instructions. Less capable or less controllable language models may generate prompts with weaker coverage, reducing the transferability of the approach to settings where only lightweight LLMs are available.

WeatherGS. WeatherGS depends on the coverage of its preprocessing modules. The Atmospheric Effect Filter uses a diffusion prior trained for a fixed set of weather degradations; extending it to unseen conditions, such as dust storms, smoke, or volcanic ash, would require additional task-specific training. Similarly, the Lens Effect Detector is based on raindrop detection and may need retraining for lens artefacts with substantially different appearance, such as mud, insects, or ice. The reconstruction backbone also assumes static scene content. Dynamic objects in the training views, including moving vehicles, pedestrians, or vegetation, are treated as view inconsistencies and may be averaged, pruned, or reconstructed as ghost artefacts. Extending WeatherGS to dynamic driving scenes would therefore require incorporating dynamic 3DGS representations (Guo et al., 2024b; Luiten et al., 2024; Yang et al., 2024d).

WeatherEdit. WeatherEdit simplifies the dynamics of weather particles. The Gaussian particle field uses constant per-particle velocities, which capture the dominant motion of rain and snow but do not model more complex atmospheric dynamics such as turbulence, wind shear, or gust-driven correlated motion. Wind is represented only through horizontal velocity components; more realistic weather dynamics would require a fluid-simulation or flow-field backend. The local field alignment strategy also introduces a trade-off. It greatly reduces the number of particles required for rendering, but because the field is camera-centred, rapid camera rotations or unusual viewpoints may expose the

boundary of the particle field before it re-centres. This artefact is uncommon in standard forward-facing driving simulations, but it remains a limitation for more aggressive camera trajectories or non-standard camera rigs.

8.3.2 Cross-Cutting Limitations

Benchmark-limited evaluation. All four methods are evaluated on established academic benchmarks, including ACDC, BDD100K, Dark Zurich, PandaSet, Waymo Open, nuScenes, KITTI-360, and MUSES. These datasets provide standardised and reproducible evaluation protocols, which are essential for fair comparison. However, they cannot fully represent the open-ended deployment envelope of autonomous driving. Real-world operation involves continuously varying weather, compound conditions such as rain at night or fog with snow, sensor noise, calibration drift, and rare edge cases that are inevitably under-represented in finite benchmarks. Moreover, none of the methods has been deployed on a physical vehicle. Practical system-level issues such as real-time scheduling, failure detection, sensor calibration, and integration with downstream planning and control remain outside the scope of this thesis.

Camera-only formulation. This thesis formulates weather-robust perception primarily as a camera-based vision problem, since its central tasks—image enhancement, generative augmentation, weather-aware 3D reconstruction, and controllable visual synthesis—are all defined in the RGB image domain. This setting is well suited to studying degradations such as reduced visibility, contrast loss, airborne particles, surface reflections, and lens occlusion. However, modern autonomous driving systems commonly combine cameras with LiDAR, radar, and sometimes thermal sensors, each of which responds differently to adverse weather. Rain, snow, and fog may introduce attenuation, backscatter, missing returns, or spurious points in LiDAR, while radar is generally more robust to visibility degradation but provides lower spatial resolution. Thermal cameras are less sensitive to illumination changes, although their effectiveness depends on object–background temperature contrast. Extending this thesis to multi-modal perception would therefore require modality-specific degradation models, calibrated multi-sensor datasets, and adaptive fusion strategies that account for the changing reliability of each sensor under different weather conditions.

Limited weather taxonomy. This thesis focuses on rain, snow, fog, and low light, which represent common and practically important adverse conditions while covering several distinct degradation mechanisms, including atmospheric scattering, airborne particles, surface accumulation, wet-road reflections, and reduced illumination. Nevertheless, these conditions do not cover the full environmental envelope encountered in autonomous driving. Hail, freezing rain, smoke, dust storms, strong glare, lens contamination, flooding, and mixed weather events may involve different particle properties, surface inter-

actions, temporal dynamics, or sensor responses. Extending the framework to these conditions would require additional training and evaluation data, condition-specific generative priors, and, in some cases, new physical assumptions or representations. The current taxonomy should therefore be viewed as a representative but incomplete subset of adverse driving conditions.

Computational cost. Although the 3D methods proposed in this thesis are substantially more efficient than NeRF-based alternatives, they are still primarily suited to offline reconstruction, simulation, and data-generation workflows rather than real-time on-vehicle deployment. WeatherGS achieves efficient rendering relative to NeRF, but the full pipeline includes preprocessing components such as diffusion-based atmospheric filtering, which are not yet real-time on typical deployment hardware. Similarly, WeatherEdit is designed for controllable simulation content generation rather than online perception. Real-time use would require additional engineering, including model compression, distillation, pruning, quantisation, or a clearer separation between offline asset construction and online inference.

Metric limitations. The evaluation relies on standard metrics such as mIoU for semantic segmentation and PSNR, SSIM, LPIPS, and NIQE for image or reconstruction quality. These metrics are useful for reproducible comparison, but they only partially capture the deployment-oriented objectives of the thesis. mIoU averages performance across classes and may under-emphasise small but safety-critical categories such as pedestrians, cyclists, riders, and traffic signs. Image-quality metrics similarly do not directly measure whether the visual changes introduced by enhancement or generation improve the evidence used by downstream perception models. Future evaluation should therefore incorporate safety-aware, class-weighted, uncertainty-aware, and task-driven metrics that better reflect the operational risks associated with adverse weather perception.

8.4 Broader Implications

8.4.1 Data-Centric AI Safety

The large mIoU gains delivered by WeatherDG (+13.9) and WeatherEdit (+14.9), compared with the smaller gains delivered by AllWeatherNet (+5.3), make a concrete empirical case for a broader principle in AI safety: model robustness under distribution shift is bounded by the coverage of the training distribution, and data-centric interventions (expanding that coverage) outperform model-centric interventions (improving the model’s response to a fixed coverage gap) when the coverage gap is large. This aligns with the growing consensus in the AI safety community that data quality, diversity, and representativeness are the primary levers for safety-critical applications (European Parliament

and Council of the European Union, 2024).

The thesis’s generative augmentation approach, which uses foundation models to synthesise diverse training data at scales no real collection effort could match, is applicable far beyond autonomous driving. Medical imaging under variable acquisition parameters, agricultural monitoring across seasons and regions, surveillance systems across day/night cycles: any safety-critical AI system that must operate under variable environmental conditions can benefit from similar generative data-augmentation strategies.

8.4.2 Ethical Considerations

Equitable performance across conditions. Autonomous vehicles should perform safely regardless of the geography and climate in which they operate. If perception systems are optimised primarily for clear weather, as is typical due to data-availability biases, then users in regions with frequent adverse weather, including economically disadvantaged communities in many northern and tropical climates, may experience disproportionately lower safety margins. The thesis contributes to mitigating this inequity by providing methods that extend perception to adverse weather without requiring the real-world data that such regions may not have. It remains important to evaluate and report per-condition performance, not only aggregate metrics, to ensure that improvements are equitably distributed across the full operating envelope.

Dual-use potential. The generative capabilities developed in this thesis, namely photorealistic synthetic driving scenes with controllable weather, could in principle be misused to fabricate misleading visual content. The methods are designed and optimised for the specific application of autonomous driving data augmentation; the broader societal implications of increasingly capable generative models are shared with the field as a whole. Continued development of watermarking techniques, provenance standards, and detection methods for synthetic imagery is the appropriate community-level response.

Environmental cost. The computational cost of training generative models, 3D scene reconstruction, and 4D particle simulation is non-trivial. Each training run of the diffusion and 3DGS components in this thesis consumes significant GPU-hours. These costs are modest compared with the training of foundation models themselves, but they are not negligible, and future work should treat computational efficiency as an ethical consideration rather than merely a convenience: the most carbon-efficient method that achieves a given task should be preferred, all else equal.

Sim-to-real over-reliance. Simulation-based testing is a powerful complement to real-world validation, but it is not a substitute. Any simulator, however realistic, is bounded by the assumptions baked into it; the gap between simulated and real conditions, the sim-to-real gap, remains an active research problem. The methods in this thesis reduce

the gap, particularly by grounding the simulation in real-world 3D scene reconstructions (WeatherGS) and physics-inspired weather dynamics (WeatherEdit), but they do not eliminate it. Regulatory frameworks should treat simulation-based evidence as necessary but not sufficient for deployment decisions.

8.5 Future Directions

The work presented in this thesis suggests several directions for future research.

8.5.1 Multi-Sensor Extension

A natural extension is to move beyond camera-only perception towards multi-sensor weather robustness. Different sensing modalities respond to adverse weather in different ways: cameras suffer from visibility degradation and illumination changes, LiDAR is affected by scattering and sparse returns, radar is more robust to weather but provides lower spatial resolution, and thermal cameras capture complementary information under low-light conditions. Extending the generative-and-spatial framework to these modalities would enable richer synthetic data and more realistic simulation. For example, a WeatherGS-like reconstruction pipeline could be extended to recover clean LiDAR geometry from degraded captures, while a WeatherEdit-like simulator could generate synchronised RGB, LiDAR, radar, and thermal observations under user-specified weather conditions. The explicit 3D scene representations used in Chapters 5 and 7 provide a natural starting point for such sensor-consistent generation.

8.5.2 Weather-Adaptive Perception Models

This thesis primarily treats weather robustness as an upstream problem: degraded inputs are enhanced before perception, or adverse weather data are generated before training. A complementary direction is to incorporate weather adaptation into the perception model itself. This could involve weather-conditioned segmentation architectures, uncertainty-aware modules that adjust their predictions according to estimated visibility, or test-time adaptation methods that update model statistics online under changing weather. The synthetic data produced by WeatherDG and WeatherEdit would provide a useful training resource for such models, since it offers controlled variation over weather type, severity, time of day, and scene content. This would allow future systems to learn not only from diverse weather examples, but also from explicit weather labels or control parameters.

8.5.3 Real-Time and Resource-Constrained Deployment

Another important direction is improving efficiency for real-time or resource-constrained deployment. Among the methods in this thesis, AllWeatherNet is the most directly

deployable on vehicle hardware because it requires only a single forward pass at inference time. Its runtime could be further reduced through model compression (Han et al., 2016), pruning (Han et al., 2015), quantisation (Jacob et al., 2018), or knowledge distillation (Hinton et al., 2015). The 3D methods are currently better suited to offline workflows, such as simulation asset creation, reconstruction, and data generation. However, real-time rendering with 3DGS is advancing rapidly, and future improvements in Gaussian compression, streaming, and hardware-aware rasterisation may make interactive weather-aware 3D simulation increasingly practical. A useful deployment strategy may therefore separate offline asset construction from online perception, using the heavier generative and 3D components to prepare data and scenarios before deployment.

8.5.4 Extension to Additional Adverse Conditions

The thesis focuses on rain, snow, fog, and low light, but autonomous vehicles may encounter a broader range of environmental degradations, including hail, smoke, dust storms, solar glare, dirty lenses, condensation, ice, and sensor ageing. Extending the proposed framework to these conditions would require both data and modelling advances. Some conditions, such as dirty lenses or condensation, can be addressed by extending the lens-occlusion detection and masking strategy used in WeatherGS. Others, such as smoke or dust, may require new generative priors and particle models with different scattering and motion properties. While the architectural principles of the thesis are broadly transferable, each new condition may require task-specific assumptions about appearance, geometry, and sensor interaction.

8.5.5 Foundation Models for Zero-Shot Weather Adaptation

The growing capability of vision-language and multimodal foundation models opens the possibility of zero-shot or few-shot weather adaptation. A future perception system could use a vision-language model to infer the current weather condition, estimate its severity, and select an appropriate enhancement, adaptation, or uncertainty-calibration strategy. In this thesis, foundation models are used primarily as components for generation, restoration, and prompting. Future work could move them closer to the perception loop itself, using them for weather diagnosis, prompt-conditioned adaptation, or automatic scenario generation. The interaction between foundation-model reasoning and generative data augmentation is especially promising: generated data could improve the robustness of perception models, while foundation models could guide what data should be generated next.

8.6 Closing Remarks

Adverse weather remains one of the most persistent challenges for autonomous driving perception. The difficulty arises from the interaction of several factors: the physics of image formation, the limited coverage of real-world training data, the brittleness of perception models under distribution shift, and the need for safety validation across a wide operating envelope. No single method can address this problem in full.

This thesis has argued that weather-robust perception is better understood as the composition of two complementary capabilities: enhancement and generation. Enhancement recovers cleaner representations from degraded observations that already exist; generation synthesises adverse weather observations that are missing from the training or testing distribution. These capabilities must also be developed at two representational granularities: the 2D image, where most perception models operate, and the reconstructed 3D scene, where multi-view consistency, novel-view rendering, and simulation become possible. Generative modelling provides the learned priors required for restoration and synthesis, while spatial modelling provides the explicit geometric structure required for consistent 3D reconstruction and simulation-grade rendering. Together, these two modelling principles define the framework that organises the thesis.

The four methodological contributions of the thesis instantiate this framework. AllWeatherNet addresses 2D enhancement by restoring adverse weather images at inference time. WeatherGS addresses 3D enhancement by reconstructing clean scene assets from weather-degraded multi-view captures. WeatherDG addresses 2D generation by synthesising adverse weather training images for domain generalisation. WeatherEdit addresses 3D generation by injecting controllable weather effects into reconstructed driving scenes. Each method addresses a distinct research question and advances a specific sub-problem. Considered together, however, they form a coherent pipeline: real-world degraded observations can be enhanced, clean 3D assets can be reconstructed, missing adverse weather data can be generated, and reconstructed scenes can be re-edited for controlled simulation.

The main contribution of the thesis is therefore not only the individual performance of these methods, but also the structure that connects them. WeatherGS produces clean 3D assets that can serve as inputs to WeatherEdit. WeatherDG improves the training distribution, while AllWeatherNet remains available as an inference-time enhancement module. WeatherEdit then closes the loop by enabling controlled simulation of adverse conditions in 3D. This composability suggests that deployment-ready robustness is unlikely to emerge from a single model or dataset. Instead, it will require the orchestration of multiple methods, each addressing a different part of the weather-robustness problem.

The framework proposed in this thesis is intended as one such organising structure. It is expected to evolve as generative models, spatial representations, and autonomous driving systems continue to develop. Additional sensing modalities, dynamic 3D re-

construction, zero-shot weather adaptation, and more realistic simulation engines will reshape the framework in future work. Nevertheless, the underlying decomposition into two capabilities and two representations provides a principled way to locate new methods and identify remaining gaps.

Appendix A

Extended Experimental Results

This appendix reports the extended experimental results that were summarised in the main chapters. For each of the four methods, I provide per-class breakdowns, additional qualitative examples, and failure-case analyses that complement the aggregate results reported in the respective chapters.

A.1 AllWeatherNet: Per-Class and Per-Condition Analysis

The main results of Chapter 4 reported aggregate mIoU. Table A.1 reports the per-class breakdown on the ACDC benchmark, revealing which semantic classes benefit most from AllWeatherNet enhancement. Road, sidewalk, and building show the largest absolute gains; small dynamic classes such as motorcycle and rider show smaller but consistent improvements.

Three patterns deserve comment. First, the “stuff” classes (road, sidewalk, building, wall, fence, sky, vegetation) consistently improve under AllWeatherNet, suggesting that the hierarchical discriminator’s scene-level component is effective at restoring the large, continuous regions whose appearance is most affected by weather. Second, the small dynamic “thing” classes (motorcycle, rider, bicycle) show mixed results: AllWeatherNet improves some (train, truck) and slightly under-performs on others (motorcycle, bicycle). This is consistent with the structural argument that enhancement at the pixel level cannot overcome under-representation of these classes in the segmentation model’s training data, a limitation that motivates WeatherDG. Third, low-light-specialised methods (SCI, PairLIE) achieve the best result on one or two classes (sidewalk, rider) but lose heavily on classes that are also affected by non-night weather (truck, bus); this confirms the chapter’s argument that specialisation to a single condition is self-defeating for driving.

Table A.1: Per-class mIoU (%) on the ACDC validation set, comparing no enhancement, two representative baselines, and AllWeatherNet. Best per-class result in **bold**.

Class	No enhancement	WGWS-Net	EnlightenGAN	SCI	Ours
Road	73.7	73.5	76.6	75.8	77.7
Sidewalk	27.0	26.0	31.6	36.8	30.3
Building	53.6	52.6	55.9	45.0	57.2
Wall	10.9	10.9	12.3	4.3	15.3
Fence	16.8	16.1	19.1	17.1	19.8
Pole	25.3	24.5	23.3	20.0	25.4
Tr. light	50.0	49.5	47.2	34.0	49.3
Sign	33.9	32.6	31.9	26.9	32.2
Veg.	70.5	68.7	72.7	59.5	73.6
Terrain	30.0	29.7	26.5	26.2	28.4
Sky	79.5	79.1	79.5	67.5	79.8
Person	36.3	35.6	30.2	33.3	33.3
Rider	9.8	13.6	10.8	16.3	10.1
Car	61.8	57.8	56.2	58.8	59.7
Truck	5.9	6.3	17.8	18.6	16.7
Bus	45.4	39.9	46.7	42.9	44.5
Train	34.4	33.4	33.3	22.0	36.1
M.cycle	15.2	16.3	12.1	0.3	12.7
Bicycle	26.2	26.6	23.0	10.2	23.6
mIoU	37.1	36.4	37.2	32.4	38.2

A.1.1 Per-Condition Breakdown

Table A.2 reports mIoU separately on the four ACDC conditions. The improvement is largest on snow (+8.7 mIoU) and smallest on fog (+1.3 mIoU), reflecting the relative difficulty of each condition for the underlying segmentation model.

Table A.2: Per-condition mIoU (%) of AllWeatherNet on the ACDC validation set.

Method	Snow	Rain	Fog	Night	Avg.
No enhancement	34.1	38.7	46.8	28.9	37.1
AllWeatherNet	42.8	41.4	48.1	31.3	38.2
Δ	+8.7	+2.7	+1.3	+2.4	+1.1

The pattern that fog yields the smallest absolute improvement does not mean fog is the easiest condition; rather, fog’s baseline is already the highest because fog preserves broad scene structure even as it reduces contrast. Conditions that more aggressively corrupt local detail (snow, night) have lower baselines and more headroom for improvement.

A.2 WeatherDG: Detailed Domain-Generalisation Results

A.2.1 Per-Condition Breakdown across Backbones

Table A.3 expands the weather-breakdown result of Chapter 6 across all three backbones and both encoder choices.

Table A.3: Per-condition mIoU (%) on the ACDC validation set for WeatherDG with three backbones and two encoder choices.

Method	Encoder	Snow	Rain	Fog	Night	Avg.
DAFormer + Ours	ResNet-50	45.9	46.8	52.1	22.4	45.2
HRDA + Ours	ResNet-50	45.2	46.0	51.3	22.1	44.5
MIC + Ours	ResNet-50	44.7	45.2	50.8	21.3	43.8
DAFormer + Ours	MiT-B5	54.2	54.1	66.8	27.3	53.3
HRDA + Ours	MiT-B5	59.7	64.6	75.9	38.7	60.2
MIC + Ours	MiT-B5	59.4	62.0	74.2	36.9	60.0

Three observations follow. First, the MiT-B5 encoder consistently outperforms ResNet-50 by 10–15 mIoU across all conditions and backbones, confirming the broader pattern that transformer encoders are better suited to the weather-robust setting. Second, among the three backbones at MiT-B5 scale, HRDA and MIC are within 1 mIoU of each other, with DAFormer trailing by approximately 7 mIoU, suggesting that the masked-consistency (MIC) and multi-resolution (HRDA) objectives are both effective but through different mechanisms. Third, nighttime is uniformly the hardest condition, confirming the pseudo-labelling bottleneck hypothesis discussed in Section 6.8.4.

A.2.2 Class Balance under Instance Sampling

The Instance Sampler ($\mathcal{E}_{IS}$) modifies the empirical class distribution of generated training images through inverse-frequency sampling. Table A.4 compares the class distribution of the WeatherDG generated set against the original Cityscapes training set and the ACDC target distribution.

The WeatherDG generated set substantially over-represents the minority classes, exactly as intended. Rider pixels, in particular, increase from 0.06% in ACDC to 3.12% in the generated set, a $52\times$ increase. This compensates for the rider class’s extreme under-representation in the real adverse weather distribution and contributes to the large mIoU gains observed on safety-critical classes such as riders and motorcycles.

Table A.4: Fractional pixel occupancy (%) of “thing” classes in the Cityscapes source, the ACDC target, and the WeatherDG generated set.

Class	Cityscapes	ACDC	WeatherDG
Person	1.82	0.73	4.21
Rider	0.21	0.06	3.12
Car	7.34	5.81	8.45
Truck	0.18	0.11	2.81
Bus	0.42	0.19	3.02
Train	0.03	0.01	1.85
Motorcycle	0.05	0.03	2.64
Bicycle	0.52	0.28	2.73

A.3 WeatherGS: Scene-Level and Backbone Analysis

A.3.1 Scene Construction Details

The six synthetic scenes used in the WeatherGS benchmark (Section 5.7.1) are constructed from three base scenes (Tanabata, Factory, Pool) with two weather overlays (snow, rain). Scene construction details:

- **Tanabata:** a modestly scaled outdoor scene with wooden structures, vegetation, and varied lighting; used to test reconstruction of fine textural detail.
- **Factory:** an industrial scene with large structural elements and relatively uniform lighting; used to test reconstruction robustness on large continuous regions.
- **Pool:** an outdoor pool scene with reflective surfaces; used to test interaction between weather artefacts and specular reflections.
- **Snow overlay:** applied through Blender’s snow-material add-on for surface accumulation plus a particle system emitting approximately 2000 snowflakes per frame with random motion.
- **Rain overlay:** particle system emitting rain streaks with downward velocity; occasional droplets added to the camera lens using OpenCV post-processing.

Each scene was rendered at 1920×1080 resolution with motion blur enabled to simulate real-world camera behaviour. Weather-free ground-truth versions of each scene (used only for quantitative evaluation) were rendered from the same camera trajectories but with the weather overlay disabled.

A.3.2 Additional Real-World Scenes

The two real-world scenes are keyframe extractions from publicly available videos. Scene 1 (snowy) shows a snow-covered wooden cabin with conifers and ground snow; Scene 2 (rainy) shows an outdoor walkway with rain streaks and lens droplets. For both real-world scenes, weather-free ground truth is unavailable, and evaluation is qualitative only.

Appendix B

Implementation Details and Hyperparameters

This appendix records the full hyperparameter settings and training recipes for each of the four methods, to facilitate reproducibility. The settings reported here are those used to produce the results in the main chapters.

B.1 WeatherGS (Chapter 5)

B.1.1 Atmospheric Effect Filter

The AEF is built on the Diff-Plugin framework of Liu *et al.* (Liu et al., 2024b). I use the pre-trained task plugins for derain and desnow. The CLIP encoder used for plugin selection is CLIP-ViT-B/32 (Radford et al., 2021). The cosine-similarity threshold θ is set to 0.22, tuned on a held-out validation subset. Text prompts used for plugin selection are “Remove the rainy effect in the image” and “Remove the snowy effect in the image.”

B.1.2 Lens Effect Detector

The LED is based on the AttGAN attention module (Qian et al., 2018), which is a pre-trained network for raindrop detection. I use the original AttGAN weights without fine-tuning on our data. The mask threshold t is set to 0.5. Morphological dilation with a 3×3 kernel is applied to the binary mask to ensure full coverage of feathered droplet edges.

B.1.3 3DGS Training

B.1.4 Benchmark Generation

The synthetic scenes are generated in Blender (Hess, 2010). The base Deblur-NeRF (Ma et al., 2022a) scenes (Tanabata, Factory, Pool) are imported into Blender; the snow

Table B.1: WeatherGS 3DGS training hyperparameters.

Parameter	Value
SfM package	COLMAP (Schonberger and Frahm, 2016)
Initial point cloud	SfM output (raw views)
Gaussian max count	3M (default 3DGS)
Densification interval	100 iterations
Densification from iteration	500
Densification until iteration	15,000
Opacity reset interval	3,000 iterations
Loss weight λ (D-SSIM)	0.2
Training iterations	30,000
GPU	NVIDIA RTX 3090
Training time (per scene)	$\sim$ 28 minutes
Rendering rate	$\sim$ 50 fps

material add-on adds accumulated snow to surfaces; a custom particle system emits airborne snowflakes and rain streaks with physically motivated parameters (terminal velocity, gravity, small turbulent velocity noise). Camera trajectories follow the original Deblur-NeRF sequences. Motion blur is enabled at render time with a shutter-open duration of 0.5 frames. OpenCV post-processing adds per-frame droplets to simulate lens adhesion.

B.2 WeatherDG (Chapter 6)

B.2.1 Diffusion Fine-Tuning

Stable Diffusion v1.5 is the base model. DreamBooth (Ruiz et al., 2023) is used for fine-tuning with the parameters in Table B.2.

Table B.2: WeatherDG Stable Diffusion fine-tuning hyperparameters.

Parameter	Value
Base model	Stable Diffusion v1.5
Fine-tuning procedure	DreamBooth
Identifier	V^* (unique token)
Fine-tuning prompt	“A photo of V^* driving scene”
Fine-tuning image source	Single Cityscapes image (random)
Crop size	512×512
Learning rate	10^{-6}
Training steps	800
Batch size	1
GPU	NVIDIA A6000
Fine-tuning time	$\sim$ 10 minutes
GPU memory	$\sim$ 20 GB

B.2.2 LLM Agents

All three LLM agents (Instance Sampler, Scene Composer, Scene Descriptor) are instantiated with Llama-2-7B-chat (Touvron et al., 2023). The prompting templates are reported in Appendix D. The Instance Sampler’s class-probability table is computed once from the ACDC validation set and cached.

B.2.3 Image Generation

Generation uses the standard DDIM sampler with 50 denoising steps and classifier-free guidance scale 7.5. Each image is generated at 512×512 resolution. Per-image generation cost is approximately 15 seconds on the A6000. For the main experiments, 1,000 synthetic images are generated per condition per backbone, for a total of approximately 5 hours of generation time per backbone.

B.2.4 UDA Segmentation Training

Three UDA backbones are used: DAFormer (Hoyer et al., 2022), HRDA (Hoyer et al., 2024), and MIC (Hoyer et al., 2023). Each backbone is trained with its published hyperparameter recipe, substituting the WeatherDG-generated synthetic images for the backbone’s expected target-domain unlabelled data. No other modifications to the backbones are made. Training time ranges from 6 hours (DAFormer) to 14 hours (MIC) on a single NVIDIA A6000.

B.3 WeatherEdit (Chapter 7)

B.3.1 All-in-One Adapter

The base diffusion backbone is CycleGAN-Turbo (Parmar et al., 2024), an efficient diffusion variant suited for image-to-image tasks. I inject a LoRA (Hu et al., 2022) adapter with rank $r = 8$ into the attention layers of the U-Net backbone. One low-rank matrix L_i is trained per weather style $i \in \{\text{snowy}, \text{rainy}, \text{foggy}\}$, with shared base weights across styles.

B.3.2 Semantic Conditioning

Semantic segmentation maps are generated by an off-the-shelf Mask2Former (Cheng et al., 2022) pre-trained on Cityscapes. The map is encoded alongside the image through a shared VAE encoder with an augmented input channel for the semantic channel. The LoRA adapter receives the combined latent $\mathbf{z}_0 = \mathcal{E}(x, M)$ as its denoising target.

B.3.3 Training Hyperparameters

Table B.3: WeatherEdit training hyperparameters for the All-in-One adapter.

Parameter	Value
Base diffusion	CycleGAN-Turbo
LoRA rank	8
Number of styles	3 (snow, rain, fog)
Training pairs	1,237 (BDD100K + MUSES + ACDC)
Learning rate	10^{-4}
Batch size	4
Training iterations	50,000
Optimiser	AdamW
Weight decay	0.01
GPU	NVIDIA A6000
Training time	~ 12 hours
Inference (per frame)	3.2 seconds

B.3.4 TV Attention Parameters

The λ parameter controlling the mix of self-, view-, and temporal-attention is set to 0.5. View attention treats the three-camera setup (left, centre, right) as the canonical configuration; scenes with more or fewer cameras would require adaptation of the attention mask structure.

B.3.5 Gaussian Particle Field

Table B.4: WeatherEdit Gaussian particle field parameters.

Parameter	Rain	Snow	Fog
Particle count q (moderate)	4,000	8,000	implicit
Field size B (metres)	$20 \times 20 \times 15$	$20 \times 20 \times 15$	$30 \times 30 \times 20$
Colour C	transparent grey	white + blue	uniform grey
Shape aspect ratio (scale S)	1:1:6 elongated	1:1:1 spherical	1:1:1 large
Vertical velocity D_z (m/s)	-8 to -10	-0.5 to -1.5	0
Horizontal velocity $D_{x,y}$ (m/s)	± 0.5 wind	± 1.5 drift	± 0.1 drift
Fog attenuation β (moderate)	–	–	0.8

Local field alignment is computed at each frame from the camera’s pose relative to the initial frame; the resulting relative transformation is applied to all particle positions in the field.

B.4 Computational Environment

All experiments in the thesis were conducted on a combination of the following hardware resources:

- NVIDIA RTX 3090 (24 GB): used for 3DGS training and rendering (WeatherGS, WeatherEdit), and AllWeatherNet training.
- NVIDIA A6000 (48 GB): used for Stable Diffusion fine-tuning and inference (WeatherDG), LoRA adapter training (WeatherEdit), and large-batch UDA training.
- CPU: AMD Ryzen Threadripper PRO 5975WX (32 cores) for data loading and preprocessing.
- RAM: 256 GB, sufficient for in-memory caching of training datasets.

Software stack: Ubuntu 22.04, CUDA 11.8, PyTorch 2.0, Python 3.10. Diffusion experiments use the Diffusers library; 3DGS experiments use the official 3D Gaussian Splatting implementation with custom modifications.

Appendix C

Dataset Details and Evaluation Protocols

This appendix documents the datasets used throughout the thesis and the evaluation protocols applied to each. The information here is intended to be sufficient for exact reproduction of the main experiments, supplementing the brief dataset descriptions given in the individual chapters.

C.1 Training Datasets

C.1.1 Cityscapes

Cityscapes (Cordts et al., 2016) is the canonical clear weather source domain for the domain-generalisation experiments in this thesis. It contains 5,000 finely annotated images captured in 50 German cities during daylight and good weather, with pixel-level semantic segmentation across 19 classes (road, sidewalk, building, wall, fence, pole, traffic light, traffic sign, vegetation, terrain, sky, person, rider, car, truck, bus, train, motorcycle, bicycle). An additional 20,000 coarsely annotated images are not used in this thesis. The official train/val/test split is 2,975/500/1,525 images.

For the purposes of this thesis, Cityscapes serves three distinct roles. In Chapter 4, Cityscapes-trained PSPNet is the fixed downstream segmentation model whose performance on enhanced images is the primary evaluation metric. In Chapter 6, Cityscapes is the source domain whose labelled training split drives the UDA backbones’ supervised component. Also, Cityscapes is used for the DreamBooth fine-tuning of Stable Diffusion.

C.1.2 ACDC: Adverse Conditions Dataset with Correspondences

ACDC (Sakaridis et al., 2021) is the primary adverse weather evaluation benchmark. It contains 4,006 images evenly distributed across four conditions: fog (1,000), nighttime (1,006), rain (1,000), and snow (1,000). Each image is accompanied by a pixel-level

semantic segmentation label (in the same 19-class Cityscapes-compatible scheme) and a coarse GPS correspondence to a clear weather image of approximately the same scene. The official train/val/test split is 1,600/406/2,000 images in total.

ACDC is used in the thesis in three distinct roles: as the adverse weather training data for AllWeatherNet (Chapter 4); as the evaluation benchmark for all three 2D methods (AllWeatherNet, WeatherDG, WeatherEdit data augmentation evaluation); and as the source of class-frequency statistics for WeatherDG’s Instance Sampler.

C.1.3 Dark Zurich

Dark Zurich (Sakaridis et al., 2020) is the canonical nighttime-only driving benchmark. It contains 8,779 images captured in Zurich across nighttime, twilight, and daytime conditions, with coarse GPS correspondences between the three times of day. A 50-image nighttime validation subset is provided with pixel-level segmentation labels. The thesis uses Dark Zurich as (i) supplementary training data for AllWeatherNet’s nighttime condition and (ii) a held-out evaluation target for WeatherDG and WeatherEdit.

C.1.4 BDD100K

BDD100K (Yu et al., 2020) is a large-scale driving video dataset covering diverse weather, geographic, and time-of-day conditions. It contains 100,000 video clips, 10,000 of which have been converted to keyframe images with semantic segmentation labels. BDD100K’s weather distribution is not uniform: it is skewed towards clear weather, with approximately 10% of images captured under rain, snow, or fog, but it provides a larger-scale reality check than ACDC’s 406-image validation set.

C.1.5 Foggy Zurich and Nighttime Driving

Foggy Zurich (Sakaridis et al., 2018) and Nighttime Driving (Dai and Van Gool, 2018) are used exclusively as held-out benchmarks for AllWeatherNet’s generalisation test (Section 4.9.4). Foggy Zurich contains 3,808 foggy images; Nighttime Driving contains 50 densely annotated test images. Neither is seen during AllWeatherNet training.

C.1.6 MUSES

MUSES (Brödermann et al., 2024) is a multi-sensor urban-scene segmentation dataset with synchronised camera, LiDAR, and radar modalities. The thesis uses MUSES only for the downstream-perception evaluation of WeatherEdit-augmented training data (Section 7.6.4).

C.2 3D Reconstruction Datasets

C.2.1 Synthetic Scenes for WeatherGS

The three base scenes (Tanabata, Factory, Pool) are drawn from Deblur-NeRF (Ma et al., 2022a) and re-rendered in Blender (Hess, 2010) with weather overlays. Each scene provides:

- 30–50 multi-view frames along a camera trajectory.
- Weather-degraded versions under snow and rain, with full ground-truth (weather-free) reference images at the same camera poses.
- Camera intrinsics and extrinsics in COLMAP-compatible format.
- Blender source files, enabling re-generation with different weather parameters.

C.2.2 Real-World Scenes for WeatherGS

Two real-world scenes are constructed from publicly available videos:

- **Snowy cabin:** approximately 40 keyframes extracted from a video of a snow-covered wooden cabin. Camera trajectory recovered via COLMAP; no ground truth available.
- **Rainy walkway:** approximately 30 keyframes from a handheld video of an outdoor walkway under moderate rain. Camera trajectory recovered via COLMAP; no ground truth available.

C.2.3 Driving Scenes for WeatherEdit

WeatherEdit is evaluated on eight driving scenes drawn from four datasets:

- **Pandaset** (Xiao et al., 2021): 3 urban-street scenes, 10 seconds each.
- **Waymo Open** (Sun et al., 2020): 2 highway scenes, 10 seconds each.
- **nuScenes** (Caesar et al., 2020): 2 urban scenes with heavy traffic.
- **KITTI-360** (Liao et al., 2023): 1 suburban scene.

All scenes use multi-view (left, centre, right) camera configurations. 3D scene reconstruction is performed with OmniRe (Chen et al., 2025) using the vendor-provided camera calibration and ego-motion.

C.3 Evaluation Protocols

C.3.1 Semantic Segmentation

All semantic-segmentation evaluations report the mean Intersection-over-Union (mIoU) across the 19 Cityscapes-compatible classes, following the standard protocol (Cordts et al., 2016). For a predicted segmentation $\hat{Y}$ and ground-truth Y , per-class IoU is

$$\text{IoU}_c = \frac{|\{i : \hat{Y}_i = c\} \cap \{i : Y_i = c\}|}{|\{i : \hat{Y}_i = c\} \cup \{i : Y_i = c\}|}, \quad \text{mIoU} = \frac{1}{19} \sum_{c=1}^{19} \text{IoU}_c. \quad (\text{C.1})$$

Pixels with an ignore label (e.g., image boundaries, out-of-vocabulary objects) are excluded from both numerator and denominator. Evaluation is run at the dataset’s native resolution; inputs are rescaled to match the segmentation model’s expected input size only if required by the backbone.

For evaluations that cross datasets (e.g., Cityscapes-trained model on ACDC), the 19 classes are matched by label name; Cityscapes class indices and ACDC class indices are bijectively mapped before mIoU computation.

C.3.2 Image Quality Metrics

PSNR. Peak Signal-to-Noise Ratio, measured between the reconstructed/enhanced image and a reference (ground-truth or clear weather paired) image:

$$\text{PSNR}(I, J) = 10 \log_{10} \left(\frac{\text{MAX}^2}{\text{MSE}(I, J)} \right), \quad (\text{C.2})$$

with $\text{MAX} = 255$ for 8-bit pixels.

SSIM. Structural Similarity Index Measure (Wang et al., 2004), evaluated with default window size and parameters ($K_1 = 0.01$, $K_2 = 0.03$, window size 11×11).

NIQE. Natural Image Quality Evaluator (Mittal et al., 2013), a no-reference metric that scores image naturalness based on statistics of natural-scene features. Lower is better.

LPIPS. Learned Perceptual Image Patch Similarity (Zhang et al., 2018) using the pre-trained AlexNet backbone. Lower is better.

C.3.3 CLIP-Based Metrics

CLIP-S. Cosine similarity between CLIP image embeddings of the original and edited images:

$$\text{CLIP-S}(x, x') = \cos(\text{CLIP}_V(x), \text{CLIP}_V(x')). \quad (\text{C.3})$$

Higher indicates better content preservation.

CLIP-DS. Directional CLIP Similarity (Gal et al., 2022). Given original and edited image pairs and text pairs (t, t') , the directional similarity is

$$\text{CLIP-DS} = \cos(\text{CLIP}_V(x') - \text{CLIP}_V(x), \text{CLIP}_T(t') - \text{CLIP}_T(t)). \quad (\text{C.4})$$

Higher indicates better alignment between the editing direction and the textual instruction.

C.3.4 Consistency Metrics

Warp error. For adjacent frames x_t and x_{t+1} with optical flow $F_{t \rightarrow t+1}$, the warp error measures the residual after back-warping:

$$\text{WarpError} = \frac{1}{|\Omega|} \sum_{p \in \Omega} \|x_{t+1}(p) - x_t(p - F_{t \rightarrow t+1}(p))\|_2. \quad (\text{C.5})$$

Lower indicates better temporal consistency. Optical flow is computed by RAFT (Teed and Deng, 2020) or a similar pre-trained flow network.

Bhattacharyya distance. Between two colour distributions p_v and $p_{v'}$ (estimated from different camera views):

$$\text{BhattaDist}(p_v, p_{v'}) = -\ln \sum_c \sqrt{p_v(c) p_{v'}(c)}, \quad (\text{C.6})$$

with the sum over discrete colour bins. Lower indicates better view consistency.

C.3.5 User-Study Protocol

The user study reported in Section 7.6.5 was conducted with 30 participants recruited from within the research community. For each weather condition (foggy, rainy, snowy), 9 scenes were presented. For each scene, participants were shown the original image alongside edited versions from each of four methods (WeatherEdit, ControlNet, Instruct-Pix2Pix, TurboEdit) in random order and asked to select “the most realistic weather editing.” No compensation was offered; participation was voluntary. Results are reported as the percentage of trials in which each method was selected.

C.4 Dataset Licensing and Ethical Use

All datasets used in the thesis are publicly available under research licenses. The primary licenses are:

- **Cityscapes:** Cityscapes License (non-commercial research use).
- **ACDC, Dark Zurich, Foggy Zurich, Nighttime Driving:** Apache-2.0 derivative, research use.
- **BDD100K:** BDD Data License Agreement, research use.
- **Waymo Open, nuScenes, KITTI-360, Pandaset:** individual vendor licenses for research use.
- **MUSES:** research use only.

All usage in the thesis complies with the respective licenses. No personal data beyond what is provided in the public releases is retained. Facial regions, license plates, and other personally identifiable regions present in the original datasets are inherited as anonymised by the vendors; no de-anonymisation is attempted.

The synthetic 3D scenes generated by the methods in this thesis (WeatherGS reconstructions, WeatherEdit edits) are released alongside the thesis’s codebase for reproducibility purposes, under the same research-use licenses as the underlying source datasets.

Appendix D

LLM Prompt Templates and Generation Examples

This appendix documents the full text of the LLM prompting templates used by WeatherDG (Chapter 6) and WeatherEdit (Chapter 7), along with representative generation examples. The templates are given verbatim; text in [BRACKETS] denotes fillable slots that the upstream agent or the calling code provides.

D.1 WeatherDG Three-Agent Pipeline

The three WeatherDG agents (Instance Sampler, Scene Composer, and Scene Descriptor) each receive a system prompt that defines their role and a user prompt that carries the instance-specific data.

D.1.1 Instance Sampler ($\mathcal{E}_{IS}$)

System prompt:

You are an instance-sampling agent for autonomous driving data generation. Your job is to select one object class from the Cityscapes class list for inclusion in a driving scene generation prompt. Sampling probabilities are provided by the caller based on the pixel-frequency distribution of the target adverse weather domain. You must respect the provided probabilities exactly. Output the selected class name only, without explanation or additional text.

User prompt template:

Class probabilities: [PROBABILITIES].
Sample one class and output its name.

Example invocation: Class probabilities: {person:0.08, rider:0.25, car:0.04, truck:0.13, bus:0.14, train:0.18, motorcycle:0.22, bicycle:0.19}. Sample one class and output its name.

Example output: motorcycle

D.1.2 Scene Composer ($\mathcal{E}_{SC}$)

System prompt:

You are a scene-composer agent for autonomous driving data generation. Given an object class, compose a scene-descriptive prompt that includes the class, a balanced choice of weather condition (one of: snowy, rainy, foggy), and a balanced choice of time of day (one of: daytime, nighttime). The weather and time choices must be approximately uniform across repeated invocations. Output the prompt in the format: "A photo of V* [CLS], [WEATHER], [TIME]."

User prompt template:

Object class: [CLS].
Compose a prompt.

Example invocations and outputs:

User: Object class: motorcycle. Compose a prompt.
Agent: A photo of V* motorcycle, rainy, daytime.
User: Object class: truck. Compose a prompt.
Agent: A photo of V* truck, snowy, nighttime.
User: Object class: bicycle. Compose a prompt.
Agent: A photo of V* bicycle, foggy, daytime.

D.1.3 Scene Descriptor ($\mathcal{E}_{SD}$)

System prompt:

You are a scene-descriptor agent for autonomous driving data generation. Given a scene-compositor prompt of the form "A photo of V* [CLS], [WEATHER], [TIME]," expand it into a richly detailed description that specifies atmospheric conditions, lighting, and environmental details appropriate to the weather and time of day. The description should be suitable for a photorealistic text-to-image diffusion model. Preserve the "A photo of V* [CLS]" prefix; append the detailed weather

and lighting description; do not change the object class. Typical output length: 20-40 words.

User prompt template:

Short prompt: [SHORT_PROMPT].

Expand with detailed weather and lighting description.

Example invocations and outputs:

User: Short prompt: A photo of V* motorcycle, rainy, daytime. Expand with detailed weather and lighting description.

Agent: A photo of V* motorcycle under a grey and overcast sky, with raindrops on the pavement and puddles reflecting the surrounding buildings, streetlights casting a warm glow through the late morning rain, the air misty and the road surface glistening.

User: Short prompt: A photo of V* truck, snowy, nighttime. Expand with detailed weather and lighting description.

Agent: A photo of V* truck on a snow-covered urban road at night, heavy snowflakes falling in the beams of headlights, accumulated snow on the pavement and vegetation, streetlights casting amber glows that illuminate the swirling snow, visibility reduced and the scene dimly lit.

User: Short prompt: A photo of V* bicycle, foggy, daytime. Expand with detailed weather and lighting description.

Agent: A photo of V* bicycle shrouded in dense morning fog on a suburban street, soft diffused daylight filtering through the mist, distant trees and buildings reduced to faint silhouettes, the road surface damp and the air heavy with moisture.

D.1.4 Pipeline Composition

The full pipeline calls the three agents in sequence. A worked example:

1. [IS output] motorcycle
2. [SC output] A photo of V* motorcycle, rainy, daytime.
3. [SD output] A photo of V* motorcycle under a grey and overcast sky, with raindrops on the pavement and puddles reflecting the surrounding buildings, streetlights casting a warm glow through the late morning rain, the air misty and the road surface glistening.

The final SD output is the prompt passed to the fine-tuned Stable Diffusion model. Generation with DDIM sampling, 50 steps, guidance scale 7.5 produces the corresponding synthetic training image.

D.2 WeatherEdit Prompts for the All-in-One Adapter

WeatherEdit’s All-in-One adapter uses simpler prompts than WeatherDG, because the adapter’s LoRA matrices are trained to respond to brief style cues rather than detailed descriptions. The prompts are of the form “[WEATHER_STYLE] weather,” with the style selected from {snowy, rainy, foggy}.

D.2.1 Adapter Training Prompts

During adapter training, each (clear weather, adverse weather) image pair is accompanied by a training prompt derived from the adverse weather class. Example training pairs:

- [Clear image, Snowy image] with prompt "snowy weather"
- [Clear image, Rainy image] with prompt "rainy weather"
- [Clear image, Foggy image] with prompt "foggy weather"

The prompt is processed through a text encoder (CLIP-Text) whose output conditions the U-Net cross-attention layers. At training time, a contrastive objective encourages the cross-attention maps to attend to style-appropriate regions of the image (snow on sidewalks, rain on road surfaces, fog across the full scene).

D.2.2 Inference Prompts

At inference, the user selects a target weather style through a simple prompt. The prompt can optionally include severity modifiers, although the severity is primarily controlled by the particle-field parameters rather than the prompt:

- "snowy weather" (default snowy style)
- "light snowy weather" (mild variation, adapter responds with subtler effect)
- "heavy foggy weather" (stronger variation, adapter responds with denser effect)

The adapter’s training does not include severity modifiers in the prompts, so severity prompts produce stylistic shifts (thicker veil, more reflections) but not the full-fledged intensity changes that the particle field delivers. Severity is best controlled through the particle count q and fog attenuation β , as documented in Appendix B.3.5.

D.3 Failure Modes and Failure Prompts

D.3.1 WeatherDG Failure Modes

Two classes of failure mode were observed in the WeatherDG generation pipeline.

Implausible scene composition. Approximately 5% of generated images contain scene elements that are implausible for a driving context (e.g., a motorcycle in the middle of an empty field, a truck seen from an aerial viewpoint). These arise when the diffusion model’s priors override the DreamBooth fine-tuning’s compositional constraints, typically on rare class-weather combinations (train in fog, bicycle at night). Such images were not filtered in the main experiments; their presence in the training set does not appear to degrade performance substantially.

LLM prompt hallucination. Approximately 2% of LLM prompts contain internally inconsistent details (e.g., “a motorcycle parked under a clear blue sky while heavy snow falls”). The diffusion model typically ignores the inconsistent component and generates a plausible image matching the dominant condition. A more conservative generation strategy would filter such prompts before image generation; I did not do so in the main experiments.

D.3.2 WeatherEdit Failure Modes

Edge artefacts at field boundary. When the camera rotates rapidly, the particle field’s local alignment can produce a brief edge artefact at the boundary between the old and new field positions. This is visible only in unusual camera trajectories (sharp turns, rapid rotations) and was not a concern in the driving scene experiments.

Segmentation-map dependence. The semantic conditioning in the All-in-One adapter relies on an off-the-shelf segmentation model (Mask2Former). On challenging scenes where the auxiliary segmentation model fails (unusual viewpoints, occluded objects), the weather effect is distributed incorrectly: snow may, for example, accumulate on road surfaces. In our experiments, Mask2Former’s accuracy was sufficient for the tested scenes, but a weather-editing deployment on more diverse scenes may need a more robust auxiliary segmentation model.

D.4 Sample Generated Images

The figures in the main chapters (Figs. 6.5 and 7.7) show representative generation examples under controlled conditions. The full training sets used in the experiments contain, by design, far more variation than can be displayed in print; the representative figures

are drawn to illustrate specific claims (the Scene Descriptor improves realism; severity is continuously controllable) rather than to span the full generation space.

For completeness, I list the per-weather-type generation statistics of the WeatherDG training set used in the main experiments:

Table D.1: WeatherDG generated training-set composition.

Weather	Snowy	Rainy	Foggy	Night	Total
Images generated	250	250	250	250	1,000
Unique prompts	240	243	247	245	975
Classes covered (thing)	8/8	8/8	8/8	8/8	8/8

The slightly lower unique-prompt counts compared with image counts arise from occasional duplicate prompts generated by the LLM pipeline; these duplicates produce slightly different images due to the stochastic nature of diffusion sampling, so they are retained.

Appendix E

Publication Timeline and Author Contributions

This appendix records the timeline of the four peer-reviewed publications that form the backbone of this thesis, along with the candidate’s specific contributions to each. The information here supplements the candidate contribution statement in the front matter of the thesis.

E.1 Publication Timeline

Table E.1 lists the four publications in chronological order of submission, with the dates of submission, acceptance, and publication.

Table E.1: Publication timeline of the four core papers.

Paper	Venue	Submitted	Accepted	Published
AllWeather-Net	ICPR 2024	May 2024	July 2024	December 2024
WeatherDG	IEEE RA-L	September 2024	February 2025	May 2025
WeatherGS	ICRA 2025	September 2024	January 2025	May 2025
WeatherEdit	AAAI 2026	August 2025	November 2025	February 2026

E.2 Candidate Contributions

For each paper, the candidate’s contributions are recorded below using the CRediT taxonomy (Contributor Roles Taxonomy). Roles not listed were not substantial; where multiple authors contributed to a role, the candidate’s share is indicated.

E.2.1 AllWeather-Net (ICPR 2024)

- **Conceptualisation:** lead. Candidate conceived the unified-enhancement framing and the three architectural innovations (SIAM, hierarchical discrimination, RAWP).

- **Methodology:** lead. Candidate designed SIAM’s quadratic scaling, the scene–object–texture discriminator hierarchy, and the RAWP search protocol.
- **Software:** lead. Candidate implemented the full AllWeatherNet codebase, including the generator, the three discriminators, the SIAM module, and the RAWP training loop.
- **Validation:** lead. Candidate designed and conducted all experiments, including the 10-method baseline comparison, per-condition breakdowns, and the unseen-dataset generalisation tests.
- **Formal analysis:** lead. Candidate produced all quantitative results, figures, and ablations.
- **Writing—original draft:** lead. Candidate drafted the full manuscript.
- **Writing—review & editing:** shared with co-authors (Rezaei, Anwar, Li, Husain, Azarmi, Wang).
- **Supervision:** Li (primary), supported by Rezaei, Anwar, Wang.

E.2.2 WeatherDG (IEEE RA-L 2025)

- **Conceptualisation:** lead. Candidate conceived the three-agent LLM pipeline architecture and the inverse-frequency class-balancing approach.
- **Methodology:** lead. Candidate designed the pipeline, the DreamBooth fine-tuning protocol, and the pseudo-target UDA training approach.
- **Software:** lead. Candidate implemented the full WeatherDG pipeline including the LLM prompting, the diffusion fine-tuning, the image generation, and the UDA training integration.
- **Validation:** lead. Candidate designed and conducted all experiments.
- **Formal analysis:** lead. Candidate produced the main-result tables, the diffusion-method comparison, the per-weather breakdown, and all ablations.
- **Writing—original draft:** lead.
- **Writing—review & editing:** shared with co-authors (Guo, Mo, Li).
- **Supervision:** Li (primary).

E.2.3 WeatherGS (ICRA 2025)

- **Conceptualisation:** equal with Guo. Candidate and Guo jointly conceived the dense-to-sparse preprocessing strategy; the candidate led the AEF design and the integration with 3DGS; Guo led the LED design and the masking protocol.
- **Methodology:** lead (AEF, 3DGS integration); shared (LED, benchmark construction).
- **Software:** shared with Guo.
- **Validation:** lead.
- **Formal analysis:** lead.
- **Writing—original draft:** lead.
- **Writing—review & editing:** shared with co-authors (Guo, Li, Markkula).
- **Supervision:** Markkula (primary), Li (secondary).

Because the candidate and the second author Guo made equal conceptual contributions to the dense-to-sparse framework, the paper is co-first-authored; both authors' contributions are individually substantial and the thesis records the candidate's portion accurately.

E.2.4 WeatherEdit (AAAI 2026)

- **Conceptualisation:** lead. Candidate conceived the two-stage architecture (2D background editing, 3D particle construction) and the requirement matrix (R1–R3, C1–C3) that drives the design.
- **Methodology:** lead. Candidate designed the All-in-One LoRA adapter, the Temporal-View attention, the Gaussian particle field, and the local camera alignment.
- **Software:** lead. Candidate implemented all four components above.
- **Validation:** lead. Candidate designed and conducted all experiments, including the multi-dataset evaluation, the user study, the downstream-perception augmentation experiments, and the four ablations.
- **Formal analysis:** lead.
- **Writing—original draft:** lead.
- **Writing—review & editing:** shared with co-authors (Li, Guo, Markkula).
- **Supervision:** Markkula (primary), Li (secondary).

E.3 Research Outputs Beyond the Four Core Papers

In addition to the four core publications, the research programme produced several secondary outputs, summarised here for completeness.

E.3.1 Code Releases

Code for each of the four core methods has been released under an open-source licence (MIT for WeatherDG and WeatherGS, Apache-2.0 for AllWeatherNet and WeatherEdit) at the candidate’s GitHub organisation. The repositories include:

- **AllWeatherNet**: full training and inference code for the GAN, including the SIAM module and the three-level hierarchical discriminator; pre-trained weights for the unified model covering all four conditions.
- **WeatherDG**: the LLM pipeline, the DreamBooth fine-tuning scripts, and the UDA training integration for DAFormer, HRDA, and MIC backbones; sample generated images.
- **WeatherGS**: the AEF and LED preprocessing modules, the mask-guided 3DGS training extension, and the benchmark-construction scripts.
- **WeatherEdit**: the All-in-One LoRA adapter, the TV-attention diffusion sampler, the 4D Gaussian particle field, and the integration with OmniRe for driving scene reconstruction.

E.3.2 Dataset Releases

The WeatherGS synthetic benchmark (six Blender-rendered scenes with weather and ground-truth variants) is released alongside the WeatherGS code. The WeatherDG synthetic training set (1,000 images, per-condition breakdown, with the prompts that generated each) is released with the WeatherDG code. Reconstruction assets from WeatherEdit (clean 3D scenes and edited variants at three severity levels) are released with the WeatherEdit code.

E.3.3 Conference Presentations

The candidate presented AllWeather-Net orally at ICPR 2024, WeatherDG as a poster at the IROS 2025, and WeatherGS orally at ICRA 2025. WeatherEdit was accepted for poster at AAAI 2026.

E.3.4 Awards and Recognition

AllWeather-Net received the Piero Zamperoni Overall Best Student Paper Award at ICPR 2024, recognising it as the top student-led paper at the conference.

Appendix F

Notation and Glossary

This appendix records the notation used throughout the thesis and provides a glossary of abbreviations. The notation is kept consistent across chapters where possible; where a symbol is reused with a chapter-specific meaning, both meanings are listed.

F.1 Mathematical Notation

F.1.1 Scalars, Vectors, Matrices

- Lowercase italic (x, t, λ): scalar quantities.
- Lowercase bold ($\mathbf{x}, \mathbf{z}$): vectors.
- Uppercase italic (I, J, C): scalar-valued functions or images.
- Uppercase bold ($\mathbf{I}, \mathbf{M}, \mathbf{T}$): matrices or tensors.
- Greek letters ($\alpha, \beta, \theta, \Sigma$): parameters, hyperparameters, covariance matrices, or network weights. Σ specifically denotes covariance in the 3DGS context.
- Calligraphic ($\mathcal{L}, \mathcal{E}, \mathcal{T}$): loss functions, LLM agents, or prompts.

F.1.2 Common Operators

- $\mathbb{E}$: expectation.
- $\|\cdot\|$: Euclidean norm unless otherwise indicated.
- $\circ$: element-wise (Hadamard) multiplication.
- $\otimes$: Kronecker product (not used in the main text; mentioned for completeness).
- $\mathbb{I}(\cdot)$: indicator function.
- D_{KL} : Kullback–Leibler divergence.

F.2 Notation Index by Chapter

F.2.1 Chapter 1: Introduction

- $I(\mathbf{x})$: observed pixel intensity at location $\mathbf{x}$.
- $J(\mathbf{x})$: clean scene radiance at $\mathbf{x}$ (in the atmospheric scattering model).
- A : global atmospheric light (airlight).
- $t(\mathbf{x})$: transmission at $\mathbf{x}$, $t(\mathbf{x}) = e^{-\beta d(\mathbf{x})}$.
- β : scattering coefficient.
- $d(\mathbf{x})$: scene depth at pixel $\mathbf{x}$.

F.2.2 Chapter 2: Literature Review

- $p_{\text{data}}(\mathbf{x})$: the distribution over natural images that a generative model aims to match.
- $q_{\phi}(\mathbf{z} \mid \mathbf{x})$: VAE encoder, parameterised by ϕ .
- $p_{\theta}(\mathbf{x} \mid \mathbf{z})$: VAE decoder, parameterised by θ .
- ϵ : sampled Gaussian noise in a diffusion process.
- $\bar{\alpha}_t = \prod_{s=1}^t (1 - \beta_s)$: cumulative signal retention in diffusion.
- $\mu, \Sigma, \alpha, \mathbf{c}$: 3DGS Gaussian primitive parameters (centre, covariance, opacity, colour).
- $G(\mathbf{x})$: 3D Gaussian evaluated at position $\mathbf{x}$.

F.2.3 Chapter 4: AllWeatherNet

- I_S, I_T : source (adverse) and target (clear) images.
- G : generator network.
- I' : enhanced output image, $I' = G(I_S) + I_S$.
- I_{ij} : illumination intensity at pixel (i, j) .
- $\text{Att}_{ij}, \text{S_Att}_{ij}$: naive illumination attention and scaled attention (SIAM).
- P^s, P^o, P^t : scene-level, object-level, and texture-level patches.
- D_s, D_o, D_t : scene, object, and texture discriminators.
- P_{Sc}^o : c -th candidate object-level patch in the RAWP search window.

- P_{best} : best-matched source object patch under RAWP.
- L_D^ℓ, L_G^ℓ : discriminator and generator loss at level ℓ .
- $\lambda_1, \lambda_2, \lambda_3$: loss weights for scene, object, texture levels.

F.2.4 Chapter 5: WeatherGS

- AEF: Atmospheric Effect Filter.
- LED: Lens Effect Detector.
- t : text prompt for AEF plugin selection.
- i, i' : input and output images of the AEF.
- E_T, E_V : CLIP text and vision encoders.
- θ : cosine-similarity threshold for plugin selection.
- $\mathcal{D}^*$: selected AEF task plugin.
- $C(x, y)$: LED confidence map.
- $\mathbf{M}$: binary occlusion mask.
- $\bar{a}_t$: cumulative signal retention (inherited from diffusion).

F.2.5 Chapter 6: WeatherDG

- V^* : DreamBooth unique identifier, bound to driving scene layout.
- $\mathbf{z}_0$: VAE-encoded latent of the input image.
- $\mathcal{E}_{IS}, \mathcal{E}_{SC}, \mathcal{E}_{SD}$: Instance Sampler, Scene Composer, Scene Descriptor (LLM agents).
- $\mathcal{T}$: final generated prompt.
- $\mathcal{P}$: class-sampling probability distribution.
- E_i : empirical pixel-count fraction of class i in a reference dataset.
- P_i : inverse-frequency sampling probability of class i .

F.2.6 Chapter 7: WeatherEdit

- W_0 : pre-trained diffusion weights.
- L_i : LoRA low-rank matrix for weather style i .
- SelfAttn, S_Attn, T_Attn: self-, spatial view-, and temporal attention outputs.
- λ : TV-attention weighting parameter.
- $A_j = \{C_j, P_j, R_j, S_j, O_j\}$: attribute tuple of particle j (colour, position, rotation, scale, opacity).
- B_i : particle-field extent for weather type i .
- q : particle count (primary severity knob).
- D_j : velocity vector of particle j .
- ΔT : relative camera pose between frames for local field alignment.

F.3 Glossary of Abbreviations

F.3.1 Methods and Frameworks

- **AllWeatherNet**: the 2D enhancement method of Chapter 4.
- **WeatherDG**: the 2D generation method of Chapter 6.
- **WeatherGS**: the 3D enhancement method of Chapter 5.
- **WeatherEdit**: the 3D generation method of Chapter 7.
- **3DGS**: 3D Gaussian Splatting.
- **NeRF**: Neural Radiance Fields.
- **GAN**: Generative Adversarial Network.
- **VAE**: Variational Autoencoder.
- **DDPM, DDIM**: Denoising Diffusion Probabilistic Model, Denoising Diffusion Implicit Model.
- **LoRA**: Low-Rank Adaptation.
- **DreamBooth**: subject-driven diffusion fine-tuning method.
- **UDA, DG**: Unsupervised Domain Adaptation, Domain Generalisation.
- **LLM**: Large Language Model.

F.3.2 Component-Level

- **SIAM**: Scaled Illumination-Aware Attention Mechanism (AllWeatherNet).
- **RAWP**: Ranked Adaptive Window Pairing (AllWeatherNet).
- **AEF, LED**: Atmospheric Effect Filter, Lens Effect Detector (WeatherGS).
- **TV attention**: Temporal-View Attention (WeatherEdit).
- **SfM**: Structure-from-Motion.

F.3.3 Datasets and Benchmarks

- **ACDC**: Adverse Conditions Dataset with Correspondences.
- **Cityscapes**: clear weather urban driving benchmark.
- **BDD100K**: Berkeley DeepDrive 100K dataset.
- **MUSES**: Multi-Sensor Urban-scene Segmentation dataset.
- **Dark Zurich**: nighttime driving benchmark.
- **Pandaset, Waymo, nuScenes, KITTI-360**: driving datasets used for WeatherEdit evaluation.

F.3.4 Metrics

- **mIoU**: mean Intersection-over-Union.
- **PSNR**: Peak Signal-to-Noise Ratio.
- **SSIM**: Structural Similarity Index Measure.
- **LPIPS**: Learned Perceptual Image Patch Similarity.
- **NIQE**: Natural Image Quality Evaluator.
- **CLIP-S, CLIP-DS**: CLIP Similarity, Directional CLIP Similarity.
- **FID**: Fréchet Inception Distance.

F.3.5 Frameworks and Standards

- **ODD**: Operational Design Domain.
- **CRedit**: Contributor Roles Taxonomy.

F.3.6 General Computing

- **GPU**: Graphics Processing Unit.
- **CUDA**: Compute Unified Device Architecture (NVIDIA).
- **Adam, AdamW**: stochastic optimisers.
- **MLP, CNN**: Multi-Layer Perceptron, Convolutional Neural Network.
- **ReLU, LeakyReLU, GELU**: activation functions.

Bibliography

- Abdullah-Al-Wadud, M., Kabir, M. H., Dewan, M. A. A., and Chae, O. (2007). A dynamic histogram equalization for image contrast enhancement. *IEEE transactions on consumer electronics*, 53(2):593–600.
- Arjovsky, M., Chintala, S., and Bottou, L. (2017). Wasserstein generative adversarial networks. In *ICML*.
- Azuma, H., Matsui, Y., and Maki, A. (2024). ZoDi: Zero-shot domain adaptation with diffusion-based image transfer. arXiv:2403.13652.
- Barron, J. T., Mildenhall, B., Tancik, M., Hedman, P., Martin-Brualla, R., and Srinivasan, P. P. (2021). Mip-NeRF: A multiscale representation for anti-aliasing neural radiance fields. In *ICCV*.
- Barron, J. T., Mildenhall, B., Verbin, D., Srinivasan, P. P., and Hedman, P. (2022). Mip-NeRF 360: Unbounded anti-aliased neural radiance fields. *CVPR*.
- Becker, N., Rust, H. W., and Ulbrich, U. (2022). Weather impacts on various types of road crashes: a quantitative analysis using generalized additive models. *European transport research review*, 14(1):1–18.
- Benigmim, Y., Roy, S., Essid, S., Kalogeiton, V., and Lathuilière, S. (2024). Collaborating foundation models for domain generalized semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3108–3119.
- Benigmim, Y., Roy, S., Essid, S., Kalogeiton, V., and Lathuilière, S. (2023). One-shot unsupervised domain adaptation with personalized diffusion models. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 698–708.
- Berman, D., Avidan, S., et al. (2016). Non-local image dehazing. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1674–1682.
- Blattmann, A., Dockhorn, T., Kulal, S., Mendelevitch, D., Kilian, M., Lorenz, D., Levi, Y., English, Z., Voleti, V., Letts, A., et al. (2023a). Stable video diffusion: Scaling latent video diffusion models to large datasets. arXiv:2311.15127.

- Blattmann, A., Rombach, R., Ling, H., Dockhorn, T., Kim, S. W., Fidler, S., and Kreis, K. (2023b). Align your latents: High-resolution video synthesis with latent diffusion models. *CVPR*.
- Brock, A., Donahue, J., and Simonyan, K. (2019). Large scale GAN training for high fidelity natural image synthesis. In *ICLR*.
- Brödermann, T., Bruggemann, D., Sakaridis, C., Ta, K., Liagouris, O., Corkill, J., and Van Gool, L. (2024). Muses: The multi-sensor semantic perception dataset for driving under uncertainty. In *European Conference on Computer Vision*, pages 21–38. Springer.
- Brooks, T., Holynski, A., and Efros, A. A. (2023). InstructPix2Pix: Learning to follow image editing instructions. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 18392–18402.
- Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., et al. (2020). Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.
- Caesar, H., Bankiti, V., Lang, A. H., Vora, S., Liong, V. E., Xu, Q., Krishnan, A., Pan, Y., Baldan, G., and Beijbom, O. (2020). nuscenes: A multimodal dataset for autonomous driving. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11621–11631.
- Cao, A. and Johnson, J. (2023). HexPlane: A fast representation for dynamic scenes. In *CVPR*.
- Chen, A., Xu, Z., Geiger, A., Yu, J., and Su, H. (2022a). TensorRF: Tensorial radiance fields. In *ECCV*.
- Chen, L.-C., Papandreou, G., Kokkinos, I., Murphy, K., and Yuille, A. L. (2018a). 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.
- Chen, L.-C., Zhu, Y., Papandreou, G., Schroff, F., and Adam, H. (2018b). Encoder-decoder with atrous separable convolution for semantic image segmentation. arXiv:1802.02611 [cs].
- Chen, S., Ye, T., Bai, J., Chen, E., Lin, J., and Zhu, L. (2024a). Teaching tailored to talent: Adverse weather restoration via prompt pool and depth-anything constraint. In *European Conference on Computer Vision (ECCV)*.
- Chen, W. and Liu, L. (2024). Deblur-GS: 3D gaussian splatting from camera motion blurred images. *Proceedings of the ACM on Computer Graphics and Interactive Techniques*, 7(1):1–15.

- Chen, W.-T., Fang, H.-Y., Ding, J.-J., Tsai, C.-C., and Kuo, S.-Y. (2020). JSTASR: Joint size and transparency-aware snow removal algorithm based on modified partial convolution and veiling effect removal. In *European Conference on Computer Vision*.
- Chen, W.-T., Huang, Z.-K., Tsai, C.-C., Yang, H.-H., Ding, J.-J., and Kuo, S.-Y. (2022b). 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.
- Chen, Y., Chen, Z., Zhang, C., Wang, F., Yang, X., Wang, Y., Cai, Z., Yang, L., Liu, H., and Lin, G. (2024b). Gaussianeditor: Swift and controllable 3d editing with gaussian splatting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21476–21485.
- Chen, Y., Gu, C., Jiang, J., Zhu, X., and Zhang, L. (2023). Periodic vibration gaussian: Dynamic urban scene reconstruction and real-time rendering. arXiv:2311.18561.
- Chen, Z., Yang, J., Huang, J., de Lutio, R., Esturo, J. M., Ivanovic, B., Litany, O., Gojcic, Z., Fidler, S., Pavone, M., Song, L., and Wang, Y. (2025). OmniRe: Omni urban scene reconstruction. In *The Thirteenth International Conference on Learning Representations*.
- Cheng, B., Misra, I., Schwing, A. G., Kirillov, A., and Girdhar, R. (2022). Masked-attention mask transformer for universal image segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1280–1289.
- Choi, S., Jung, S., Yun, H., Kim, J. T., Kim, S., and Choo, J. (2021). Robustnet: Improving domain generalization in urban-scene segmentation via instance selective whitening. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11580–11590.
- Cordts, M., Omran, M., Ramos, S., Rehfeld, T., Enzweiler, M., Benenson, R., Franke, U., Roth, S., and Schiele, B. (2016). The cityscapes dataset for semantic urban scene understanding. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3213–3223.
- Creswell, A., White, T., Dumoulin, V., Arulkumaran, K., Sengupta, B., and Bharath, A. A. (2018). Generative adversarial networks: An overview. *IEEE signal processing magazine*, 35(1):53–65.
- Cui, Y., Ren, W., Cao, X., and Knoll, A. (2024a). AdaIR: Adaptive all-in-one image restoration via frequency mining and modulation. In *Advances in Neural Information Processing Systems (NeurIPS)*.

- Cui, Z., Gu, L., Sun, X., Ma, X., Qiao, Y., and Harada, T. (2024b). Aleth-NeRF: Illumination adaptive NeRF with concealing field assumption. In *Proceedings of the AAAI Conference on Artificial Intelligence*.
- Dai, D. and Van Gool, L. (2018). Dark model adaptation: Semantic image segmentation from daytime to nighttime. In *International Conference on Intelligent Transportation Systems (ITSC)*, pages 3819–3824. IEEE.
- Deutch, G., Gal, R., Garibi, D., Patashnik, O., and Cohen-Or, D. (2024). TurboEdit: Text-based image editing using few-step diffusion models.
- Dhariwal, P. and Nichol, A. (2021). Diffusion models beat gans on image synthesis. *Advances in neural information processing systems*, 34:8780–8794.
- Dong, X., Xu, W., Miao, Z., Ma, L., Zhang, C., Yang, J., Jin, Z., Teoh, A. B. J., and Shen, J. (2022). Abandoning the bayer-filter to see in the dark. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17431–17440.
- Dosovitskiy, A., Ros, G., Codevilla, F., Lopez, A., and Koltun, V. (2017). CARLA: An open urban driving simulator. In *Conference on robot learning*, pages 1–16. PMLR.
- European Parliament and Council of the European Union (2024). Regulation (EU) 2024/1689 laying down harmonised rules on artificial intelligence and amending regulations (EC) no 300/2008, (EU) no 167/2013, (EU) no 168/2013, (EU) 2018/858, (EU) 2018/1139 and (EU) 2019/2144 and directives 2014/90/EU, (EU) 2016/797 and (EU) 2020/1828. Official Journal of the European Union. Article 10: Data and data governance.
- Fahes, M., Vu, T.-H., Bursuc, A., Pérez, P., and de Charette, R. (2023). PØDA: Prompt-driven zero-shot domain adaptation. In *ICCV*.
- Fardo, F. A., Conforto, V. H., De Oliveira, F. C., and Rodrigues, P. S. (2016). A formal evaluation of PSNR as quality measurement parameter for image segmentation algorithms. arXiv:1605.07116.
- Feldman, B. E. and O’Brien, J. F. (2002). Modeling the accumulation of wind-driven snow. In *ACM SIGGRAPH 2002 conference abstracts and applications*, pages 218–218.
- Fridovich-Keil, S., Meanti, G., Warburg, F. R., Recht, B., and Kanazawa, A. (2023). K-planes: Explicit radiance fields in space, time, and appearance. In *CVPR*.
- Fridovich-Keil, S., Yu, A., Tancik, M., Chen, Q., Recht, B., and Kanazawa, A. (2022). Plenoxels: Radiance fields without neural networks. In *CVPR*.

- Fu, Z., Yang, Y., Tu, X., Huang, Y., Ding, X., and Ma, K.-K. (2023). Learning a simple low-light image enhancer from paired low-light instances. In *CVPR*, pages 22252–22261.
- Gal, R., Alaluf, Y., Atzmon, Y., Patashnik, O., Bermano, A. H., Chechik, G., and Cohen-Or, D. (2023). An image is worth one word: Personalizing text-to-image generation using textual inversion. In *ICLR*.
- Gal, R., Patashnik, O., Maron, H., Bermano, A. H., Chechik, G., and Cohen-Or, D. (2022). Stylegan-nada: Clip-guided domain adaptation of image generators. *ACM Transactions on Graphics (TOG)*, 41(4):1–13.
- Geiger, A., Lenz, P., and Urtasun, R. (2012). Are we ready for autonomous driving? the KITTI vision benchmark suite. In *Conference on Computer Vision and Pattern Recognition (CVPR)*.
- Goodfellow, I. J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. (2014). Generative adversarial networks.
- Guédon, A. and Lepetit, V. (2024). SuGaR: Surface-aligned gaussian splatting for efficient 3D mesh reconstruction and high-quality mesh rendering. *CVPR*.
- Gulrajani, I., Ahmed, F., Arjovsky, M., Dumoulin, V., and Courville, A. (2017). Improved training of wasserstein GANs. In *NeurIPS*.
- Guo, C., Li, C., Guo, J., Loy, C. C., Hou, J., Kwong, S., and Cong, R. (2020). Zero-reference deep curve estimation for low-light image enhancement. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1777–1786, Seattle, WA, USA. IEEE.
- Guo, Y., Gao, Y., Liu, W., Lu, Y., and Liu, R. W. (2024a). OneRestore: A universal restoration framework for composite degradation. In *European Conference on Computer Vision (ECCV)*.
- Guo, Z., Zhou, W., Li, L., Wang, M., and Li, H. (2024b). Motion-aware 3D gaussian splatting for efficient dynamic scene reconstruction. *IEEE Transactions on Circuits and Systems for Video Technology*, pages 1–1.
- Haas, J. K. (2014). A history of the unity game engine. Worcester Polytechnic Institute.
- Hahner, M., Dai, D., Sakaridis, C., Zaeck, J.-N., and Van Gool, L. (2019). Semantic understanding of foggy scenes with purely synthetic data. In *2019 IEEE Intelligent Transportation Systems Conference (ITSC)*, pages 3675–3681. IEEE.
- Han, S., Mao, H., and Dally, W. J. (2016). Deep compression: Compressing deep neural networks with pruning, trained quantization and Huffman coding. In *International Conference on Learning Representations (ICLR)*.

- Han, S., Pool, J., Tran, J., and Dally, W. J. (2015). Learning both weights and connections for efficient neural networks. In *Advances in Neural Information Processing Systems*.
- Haque, A., Tancik, M., Efros, A. A., Holynski, A., and Kanazawa, A. (2023). Instruct-NeRF2NeRF: Editing 3D scenes with instructions. In *ICCV*.
- He, K., Zhang, X., Ren, S., and Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778.
- Hertz, A., Mokady, R., Tenenbaum, J., Aberman, K., Pritch, Y., and Cohen-Or, D. (2023). Prompt-to-prompt image editing with cross-attention control. *ICLR*.
- Hess, R. (2010). *Blender Foundations: The Essential Guide to Learning Blender 2.6*. Focal Press.
- Higgins, I., Matthey, L., Pal, A., Burgess, C., Glorot, X., Botvinick, M., Mohamed, S., and Lerchner, A. (2017). beta-VAE: Learning basic visual concepts with a constrained variational framework. In *International Conference on Learning Representations (ICLR)*.
- Hinton, G., Vinyals, O., and Dean, J. (2015). Distilling the knowledge in a neural network. arXiv:1503.02531.
- Ho, J., Jain, A., and Abbeel, P. (2020). Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851.
- Ho, J. and Salimans, T. (2021). Classifier-free diffusion guidance. In *NeurIPS 2021 Workshop on Deep Generative Models and Downstream Applications*.
- Ho, J., Salimans, T., Gritsenko, A., Chan, W., Norouzi, M., and Fleet, D. J. (2022). Video diffusion models. *NeurIPS*.
- Hore, A. and Ziou, D. (2010). Image quality metrics: PSNR vs. SSIM. In *2010 20th international conference on pattern recognition*, pages 2366–2369. IEEE.
- Hoyer, L., Dai, D., and Gool, L. V. (2024). Domain adaptive and generalizable network architectures and training strategies for semantic image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 46(01):220–235.
- Hoyer, L., Dai, D., and Van Gool, L. (2022). DAFormer: Improving network architectures and training strategies for domain-adaptive semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9924–9935.

- Hoyer, L., Dai, D., Wang, H., and Van Gool, L. (2023). MIC: Masked image consistency for context-enhanced domain adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.
- Hu, E. J., Shen, Y., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., Wang, L., Chen, W., et al. (2022). Lora: Low-rank adaptation of large language models. *ICLR*, 1(2):3.
- Huang, B., Yu, Z., Chen, A., Geiger, A., and Gao, S. (2024). 2D gaussian splatting for geometrically accurate radiance fields. In *SIGGRAPH*.
- Huang, K., Shi, B., Li, X., Li, X., Huang, S., and Li, Y. (2022). Multi-modal sensor fusion for auto driving perception: A survey. arXiv:2202.02703.
- Huang, L., Chen, D., Liu, Y., Shen, Y., Zhao, D., and Zhou, J. (2023). Composer: Creative and controllable image synthesis with composable conditions. In *ICML*.
- Huang, X., Liu, M.-Y., Belongie, S., and Kautz, J. (2018). Multimodal unsupervised image-to-image translation. In *ECCV*.
- Huo, Z., Zhan, X., Qiao, Y., and Zhao, S. (2025). D3-dehaze: a divide-and-conquer framework for enhanced single image dehazing. *The Visual Computer*, pages 1–16.
- Hwang, S., Jeon, S., Ma, Y.-S., and Byun, H. (2022). WeatherGAN: Unsupervised multi-weather image-to-image translation via single content-preserving UResNet generator. *Multimedia Tools and Applications*, 81(28):40269–40288.
- Isola, P., Zhu, J.-Y., Zhou, T., and Efros, A. A. (2017). Image-to-image translation with conditional adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1125–1134.
- Jacob, B., Kligys, S., Chen, B., Zhu, M., Tang, M., Howard, A., Adam, H., and Kalenichenko, D. (2018). Quantization and training of neural networks for efficient integer-arithmetic-only inference. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*.
- Jia, Y., Hoyer, L., Huang, S., Wang, T., Van Gool, L., Schindler, K., and Obukhov, A. (2024). DGInStyle: Domain-generalizable semantic segmentation with image diffusion models and stylized semantic control. In *European Conference on Computer Vision (ECCV)*.
- Jiang, J., Zuo, Z., Wu, G., Jiang, K., and Liu, X. (2025). A survey on all-in-one image restoration: Taxonomy, evaluation and future trends. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

- Jiang, K., Wang, Z., Wang, Z., Chen, C., Yi, P., Lu, T., and Lin, C.-W. (2022). Degrade is upgrade: Learning degradation for low-light image enhancement. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 1078–1086.
- Jiang, X., Huang, J., Jin, S., and Lu, S. (2023). Domain generalization via balancing training difficulty and model capability. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 18993–19003.
- Jiang, Y., Gong, X., Liu, D., Cheng, Y., Fang, C., Shen, X., Yang, J., Zhou, P., and Wang, Z. (2021). Enlightengan: Deep light enhancement without paired supervision. *IEEE transactions on image processing*, 30:2340–2349.
- Jobson, D. J., Rahman, Z.-u., and Woodell, G. A. (1997a). A multiscale retinex for bridging the gap between color images and the human observation of scenes. *IEEE Transactions on Image processing*, 6(7):965–976.
- Jobson, D. J., Rahman, Z.-u., and Woodell, G. A. (1997b). Properties and performance of a center/surround retinex. *IEEE transactions on image processing*, 6(3):451–462.
- Jolicoeur-Martineau, A. (2018). The relativistic discriminator: a key element missing from standard GAN. arXiv:1807.00734.
- Jose Valanarasu, J. M., Yasarla, R., and Patel, V. M. (2022). TransWeather: Transformer-based restoration of images degraded by adverse weather conditions. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2343–2353, New Orleans, LA, USA. IEEE.
- Kang, L.-W., Lin, C.-W., and Fu, Y.-H. (2012). Automatic single-image-based rain streaks removal via image decomposition. *IEEE transactions on image processing*, 21(4):1742–1755.
- Karras, T., Aila, T., Laine, S., and Lehtinen, J. (2018). Progressive growing of GANs for improved quality, stability, and variation. In *ICLR*.
- Karras, T., Aittala, M., Aila, T., and Laine, S. (2022). Elucidating the design space of diffusion-based generative models. In *NeurIPS*.
- Karras, T., Aittala, M., Härkönen, E., Lehtinen, J., Laine, S., Hellsten, J., and Aila, T. (2021). Alias-free generative adversarial networks. In *Advances in Neural Information Processing Systems*, pages 852–863.
- Karras, T., Laine, S., and Aila, T. (2019). A style-based generator architecture for generative adversarial networks. In *CVPR*.
- Karras, T., Laine, S., Aittala, M., Hellsten, J., Lehtinen, J., and Aila, T. (2020). Analyzing and improving the image quality of StyleGAN. In *CVPR*.

- Kenk, M. A. and Hassaballah, M. (2020). DAWN: vehicle detection in adverse weather nature dataset. arXiv:2008.05402.
- Kerbl, B., Kopanas, G., Leimkühler, T., and Drettakis, G. (2023). 3D gaussian splatting for real-time radiance field rendering. *ACM Transactions on Graphics*, 42(4).
- Kerbl, B., Meuleman, A., Kopanas, G., Wimmer, M., Lanvin, A., and Drettakis, G. (2024). A hierarchical 3D gaussian representation for real-time rendering of very large datasets. *ACM Transactions on Graphics*.
- Kerim, A., Chamone, F., Ramos, W., Marcolino, L. S., Nascimento, E. R., and Jiang, R. (2022). Semantic segmentation under adverse conditions: A weather and nighttime-aware synthetic data-based approach.
- Kingma, D. P. (2013). Auto-encoding variational bayes. arXiv:1312.6114.
- Lee, D., Lee, M., Shin, C., and Lee, S. (2023). DP-NeRF: Deblurred neural radiance field with physical scene priors. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 12386–12396.
- Lee, H.-Y., Tseng, H.-Y., Huang, J.-B., Singh, M., and Yang, M.-H. (2018). Diverse image-to-image translation via disentangled representations. In *Proceedings of the European conference on computer vision (ECCV)*, pages 35–51.
- Lee, J. C., Rho, D., Sun, X., Ko, J. H., and Park, E. (2024). Compact 3D gaussian representation for radiance field. *CVPR*.
- Li, P., Tian, J., Tang, Y., Wang, G., and Wu, C. (2021). Deep retinex network for single image dehazing. *IEEE Transactions on Image Processing*, 30:1100–1115.
- Li, R., Tan, R. T., and Cheong, L.-F. (2020). 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.
- Li, X., Wu, J., Lin, Z., Liu, H., and Zha, H. (2018a). Recurrent squeeze-and-excitation context aggregation net for single image deraining. In *Proceedings of the European conference on computer vision (ECCV)*, pages 254–269.
- Li, Y., Lin, Z.-H., Forsyth, D., Huang, J.-B., and Wang, S. (2023a). ClimateNeRF: Extreme weather synthesis in neural radiance field. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*.
- Li, Y., Liu, H., Wu, Q., Mu, F., Yang, J., Gao, J., Li, C., and Lee, Y. J. (2023b). GLIGEN: Open-set grounded text-to-image generation. In *CVPR*.

- Li, Y., Liu, M.-Y., Li, X., Yang, M.-H., and Kautz, J. (2018b). A closed-form solution to photorealistic image stylization. In *Proceedings of the European conference on computer vision (ECCV)*, pages 453–468.
- Li, Y., Wu, J., Zhao, L., and Liu, P. (2024). DerainNeRF: 3D scene estimation with adhesive waterdrop removal. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pages 2787–2793.
- Li, Z., Wang, W., Li, H., Xie, E., Sima, C., Lu, T., Yu, Q., and Dai, J. (2022). BEVFormer: Learning bird’s-eye-view representation from multi-camera images via spatiotemporal transformers.
- Liao, Y., Xie, J., and Geiger, A. (2023). Kitti-360: A novel dataset and benchmarks for urban scene understanding in 2d and 3d. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(3):3292–3310.
- Lin, C.-H., Wang, Z., Liang, R., Zhang, Y., Fidler, S., Wang, S., and Gojcic, Z. (2025). Controllable weather synthesis and removal with video diffusion models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 13580–13591.
- Liu, K., Zhan, F., Xu, M., Theobalt, C., Shao, L., and Lu, S. (2024a). Stylegaussian: Instant 3d style transfer with gaussian splatting. In *SIGGRAPH Asia 2024 Technical Communications*, pages 1–4. ACM.
- Liu, M.-Y., Breuel, T., and Kautz, J. (2017). Unsupervised image-to-image translation networks. *Advances in neural information processing systems*, 30.
- Liu, R., Ma, L., Zhang, J., Fan, X., and Luo, Z. (2021). Retinex-inspired unrolling with cooperative prior architecture search for low-light image enhancement. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10561–10570.
- Liu, Y., Ke, Z., Liu, F., Zhao, N., and Lau, R. W. (2024b). Diff-plugin: Revitalizing details for diffusion-based low-level tasks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4197–4208.
- Liu, Y., Luo, C., Fan, L., Wang, N., Peng, J., and Zhang, Z. (2024c). CityGaussian: Real-time high-quality large-scale scene rendering with gaussians. *ECCV*.
- Liu, Y.-F., Jaw, D.-W., Huang, S.-C., and Hwang, J.-N. (2018). DesnowNet: Context-aware deep network for snow removal. *IEEE Transactions on Image Processing*, 27(6):3064–3073.
- Long, J., Shelhamer, E., and Darrell, T. (2015). Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3431–3440.

- Lu, T., Yu, M., Xu, L., Xiangli, Y., Wang, L., Lin, D., and Dai, B. (2024). Scaffold-GS: Structured 3D gaussians for view-adaptive rendering. *CVPR*.
- Luiten, J., Kopanas, G., Leibe, B., and Ramanan, D. (2024). Dynamic 3D gaussians: Tracking by persistent dynamic view synthesis. In *3DV*.
- Ma, L., Li, X., Liao, J., Zhang, Q., Wang, X., Wang, J., and Sander, P. V. (2022a). Deblur-NeRF: Neural radiance fields from blurry images.
- Ma, L., Ma, T., Liu, R., Fan, X., and Luo, Z. (2022b). Toward fast, flexible, and robust low-light image enhancement. In *CVPR*, pages 5637–5646.
- Mao, X., Li, Q., Xie, H., Lau, R. Y., Wang, Z., and Paul Smolley, S. (2017). Least squares generative adversarial networks. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*.
- Martin-Brualla, R., Radwan, N., Sajjadi, M. S., Barron, J. T., Dosovitskiy, A., and Duckworth, D. (2021). NeRF in the wild: Neural radiance fields for unconstrained photo collections. In *CVPR*.
- Meng, C., He, Y., Song, Y., Song, J., Wu, J., Zhu, J.-Y., and Ermon, S. (2021). Sdedit: Guided image synthesis and editing with stochastic differential equations. arXiv:2108.01073.
- Mildenhall, B., Srinivasan, P. P., Tancik, M., Barron, J. T., Ramamoorthi, R., and Ng, R. (2020). NeRF: Representing scenes as neural radiance fields for view synthesis.
- Mirza, M. and Osindero, S. (2014). Conditional generative adversarial nets. In *arXiv preprint arXiv:1411.1784*.
- Mittal, A., Soundararajan, R., and Bovik, A. C. (2013). Making a “completely blind” image quality analyzer. *IEEE Signal Processing Letters*, 20(3):209–212.
- Miyato, T., Kataoka, T., Koyama, M., and Yoshida, Y. (2018). Spectral normalization for generative adversarial networks. In *ICLR*.
- Mokady, R., Hertz, A., Aberman, K., Pritch, Y., and Cohen-Or, D. (2023). Null-text inversion for editing real images using guided diffusion models. In *CVPR*.
- Mou, C., Wang, X., Xie, L., Wu, Y., Zhang, J., Qi, Z., and Shan, Y. (2024). T2I-Adapter: learning adapters to dig out more controllable ability for text-to-image diffusion models. In *Proceedings of the Thirty-Eighth AAAI Conference on Artificial Intelligence and Thirty-Sixth Conference on Innovative Applications of Artificial Intelligence and Fourteenth Symposium on Educational Advances in Artificial Intelligence*, volume 38(5) of *AAAI’24/IAAI’24/EAAI’24*, pages 4296–4304. AAAI Press.

- Müller, T., Evans, A., Schied, C., and Keller, A. (2022). Instant neural graphics primitives with a multiresolution hash encoding. *ACM Transactions on Graphics*.
- Niedermayr, S., Stumpfegger, J., and Westermann, R. (2024). Compressed 3D gaussian splatting for accelerated novel view synthesis. *CVPR*.
- Niemeijer, J., Schwonberg, M., Termöhlen, J.-A., Schmidt, N. M., and Fingscheidt, T. (2024). Generalization by adaptation: Diffusion-based domain extension for domain-generalized semantic segmentation. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 2830–2840.
- Ost, J., Mannan, F., Thuerey, N., Knodt, J., and Heide, F. (2021). Neural scene graphs for dynamic scenes. In *CVPR*.
- Özdenizci, O. and Legenstein, R. (2023). Restoring vision in adverse weather conditions with patch-based denoising diffusion models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(8):10346–10357.
- Pan, X., Luo, P., Shi, J., and Tang, X. (2018). Two at once: Enhancing learning and generalization capacities via IBN-Net. In *ECCV*.
- Park, K., Sinha, U., Hedman, P., Barron, J. T., Bouaziz, S., Goldman, D. B., Martin-Brualla, R., and Seitz, S. M. (2021). HyperNeRF: A higher-dimensional representation for topologically varying neural radiance fields. *ACM Transactions on Graphics*.
- Park, T., Efros, A. A., Zhang, R., and Zhu, J.-Y. (2020). Contrastive learning for unpaired image-to-image translation. In *European Conference on Computer Vision*.
- Park, T., Liu, M.-Y., Wang, T.-C., and Zhu, J.-Y. (2019). Semantic image synthesis with spatially-adaptive normalization. In *CVPR*.
- Parmar, G., Park, T., Narasimhan, S., and Zhu, J.-Y. (2024). One-step image translation with text-to-image models. arXiv:2403.12036.
- Peng, D., Lei, Y., Hayat, M., Guo, Y., and Li, W. (2022). Semantic-aware domain generalized segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2594–2605.
- Poole, B., Jain, A., Barron, J. T., and Mildenhall, B. (2023). DreamFusion: Text-to-3D using 2D diffusion. *ICLR*.
- Pratt, S., Covert, I., Liu, R., and Farhadi, A. (2023). What does a platypus look like? generating customized prompts for zero-shot image classification. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15691–15701.

- Pumarola, A., Corona, E., Pons-Moll, G., and Moreno-Noguer, F. (2021). D-NeRF: Neural radiance fields for dynamic scenes. In *CVPR*.
- Qian, C., Guo, Y., Li, W., and Markkula, G. (2025a). WeatherGS: 3D scene reconstruction in adverse weather conditions via gaussian splatting. In *2025 IEEE International Conference on Robotics and Automation (ICRA)*, pages 185–191.
- Qian, C., Guo, Y., Mo, Y., and Li, W. (2025b). WeatherDG: LLM-assisted procedural weather generation for domain-generalized semantic segmentation. *IEEE Robotics and Automation Letters*, 10(6):5919–5926.
- Qian, C., Li, W., Guo, Y., and Markkula, G. (2026). WeatherEdit: Controllable weather editing with 4D gaussian field. *Proceedings of the AAAI Conference on Artificial Intelligence*, 40(10):8511–8519.
- Qian, C., Rezaei, M., Anwar, S., Li, W., Hussain, T., Azarmi, M., and Wang, W. (2024). *AllWeather-Net: Unified Image Enhancement for Autonomous Driving Under Adverse Weather and Low-Light Conditions*, page 151–166. Springer Nature Switzerland.
- Qian, R., Tan, R. T., Yang, W., Su, J., and Liu, J. (2018). Attentive generative adversarial network for raindrop removal from a single image. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2482–2491.
- Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al. (2021). Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR.
- Radford, A., Metz, L., and Chintala, S. (2016). Unsupervised representation learning with deep convolutional generative adversarial networks. In *ICLR*.
- Ranftl, R., Bochkovskiy, A., and Koltun, V. (2021). Vision transformers for dense prediction. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 12179–12188.
- Ren, D., Zuo, W., Hu, Q., Zhu, P., and Meng, D. (2019). Progressive image deraining networks: A better and simpler baseline. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 3937–3946.
- Ren, J., Pan, L., Tang, J., Zhang, C., Cao, A., Zeng, G., and Liu, Z. (2023). Dreamgaussian4d: Generative 4d gaussian splatting. arXiv:2312.17142.
- Ren, W., Tian, J., Han, Z., Chan, A., and Tang, Y. (2017). Video desnowing and deraining based on matrix decomposition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4210–4219.

- Rombach, R., Blattmann, A., Lorenz, D., Esser, P., and Ommer, B. (2021). High-resolution image synthesis with latent diffusion models.
- Ronneberger, O., Fischer, P., and Brox, T. (2015). 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.
- Ruiz, N., Li, Y., Jampani, V., Pritch, Y., Rubinstein, M., and Aberman, K. (2023). Dreambooth: Fine tuning text-to-image diffusion models for subject-driven generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*.
- Saharia, C., Chan, W., Chang, H., Lee, C., Ho, J., Salimans, T., Fleet, D., and Norouzi, M. (2022). Palette: Image-to-image diffusion models. In *SIGGRAPH*.
- Sakaridis, C., Dai, D., Hecker, S., and Van Gool, L. (2018). Model adaptation with synthetic and real data for semantic dense foggy scene understanding. In *ECCV*, pages 687–704.
- Sakaridis, C., Dai, D., and Van Gool, L. (2020). Map-guided curriculum domain adaptation and uncertainty-aware evaluation for semantic nighttime image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.
- Sakaridis, C., Dai, D., and Van Gool, L. (2021). ACDC: The adverse conditions dataset with correspondences for semantic driving scene understanding. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10765–10775.
- Schmidt, V., Luccioni, A., Mukkavilli, S. K., Balasooriya, N., Sankaran, K., Chayes, J., and Bengio, Y. (2019). Visualizing the consequences of climate change using cycle-consistent adversarial networks. arXiv:1905.03709.
- Schmidt, V., Luccioni, A., Teng, M., Zhang, T., Reynaud, A., Raghupathi, S., Cosne, G., Juraver, A., Vardanyan, V., Hernández-García, A., et al. (2022). ClimateGAN: Raising climate change awareness by generating images of floods. In *International Conference on Learning Representations*.
- Schonberger, J. L. and Frahm, J.-M. (2016). Structure-from-motion revisited. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4104–4113.
- Seiskari, O., Ylilammi, J., Kaatrasalo, V., Rantalankila, P., Turkulainen, M., Kannala, J., Rahtu, E., and Solin, A. (2024). Gaussian splatting on the move: Blur and rolling shutter compensation for natural camera motion. arXiv:2403.13327.

- Shah, S., Dey, D., Lovett, C., and Kapoor, A. (2018). Airsim: High-fidelity visual and physical simulation for autonomous vehicles. In *Field and Service Robotics: Results of the 11th International Conference*, pages 621–635. Springer.
- Singer, U., Polyak, A., Hayes, T., Yin, X., An, J., Zhang, S., Hu, Q., Yang, H., Ashual, O., Gafni, O., et al. (2023). Make-a-video: Text-to-video generation without text-video data. *ICLR*.
- Song, J., Meng, C., and Ermon, S. (2020). Denoising diffusion implicit models. arXiv:2010.02502.
- Song, Y. and Ermon, S. (2019). Generative modeling by estimating gradients of the data distribution. *NeurIPS*.
- Song, Y., Sohl-Dickstein, J., Kingma, D. P., Kumar, A., Ermon, S., and Poole, B. (2021). Score-based generative modeling through stochastic differential equations. In *ICLR*.
- Stomakhin, A., Schroeder, C., Chai, L., Teran, J., and Selle, A. (2013). A material point method for snow simulation. *ACM Transactions on Graphics (TOG)*, 32(4):1–10.
- Sun, P., Kretschmar, H., Dotiwalla, X., Chouard, A., Patnaik, V., Tsui, P., Guo, J., Zhou, Y., Chai, Y., Caine, B., Vasudevan, V., Han, W., Ngiam, J., Zhao, H., Timofeev, A., Ettinger, S., Krivokon, M., Gao, A., Joshi, A., Zhang, Y., Shlens, J., Chen, Z., and Anguelov, D. (2020). Scalability in perception for autonomous driving: Waymo open dataset. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.
- Sun, S., Ren, W., Gao, X., Wang, R., and Cao, X. (2024). Restoring images in adverse weather conditions via histogram transformer. In *European Conference on Computer Vision (ECCV)*.
- Tancik, M., Casser, V., Yan, X., Pradhan, S., Mildenhall, B., Srinivasan, P. P., Barron, J. T., and Kretschmar, H. (2022). Block-NeRF: Scalable large scene neural view synthesis. In *CVPR*.
- Tang, J., Chen, Z., Chen, X., Wang, T., Zeng, G., and Liu, Z. (2024). Lgm: Large multi-view gaussian model for high-resolution 3d content creation. In *European Conference on Computer Vision*, pages 1–18. Springer.
- Teed, Z. and Deng, J. (2020). RAFT: Recurrent all-pairs field transforms for optical flow. In *European Conference on Computer Vision (ECCV)*, pages 402–419.
- Touvron, H., Martin, L., Stone, K., Albert, P., Almahairi, A., Babaei, Y., Bashlykov, N., Batra, S., Bhargava, P., Bhosale, S., et al. (2023). Llama 2: Open foundation and fine-tuned chat models. arXiv:2307.09288.

- Turki, H., Ramanan, D., and Satyanarayanan, M. (2022). Mega-NeRF: Scalable construction of large-scale NeRFs for virtual fly-throughs. In *CVPR*.
- Turki, H., Zhang, J. Y., Ferroni, F., and Ramanan, D. (2023). SUDS: Scalable urban dynamic scenes. In *CVPR*.
- Vachha, C. and Haque, A. (2024). Instruct-GS2GS: Editing 3D gaussian splats with instructions.
- Vahdat, A. and Kautz, J. (2020). NVAE: A deep hierarchical variational autoencoder. In *Advances in Neural Information Processing Systems (NeurIPS)*.
- Valanarasu, J. M. J., Yasarla, R., and Patel, V. M. (2021). TransWeather: Transformer-based restoration of images degraded by adverse weather conditions.
- van den Oord, A., Vinyals, O., and Kavukcuoglu, K. (2017). Neural discrete representation learning. In *Advances in Neural Information Processing Systems (NeurIPS)*.
- Verbin, D., Hedman, P., Mildenhall, B., Zickler, T., Barron, J. T., and Srinivasan, P. P. (2022). Ref-NeRF: Structured view-dependent appearance for neural radiance fields. In *CVPR*.
- Voleti, V., Yao, C.-H., Boss, M., Letts, A., Pankratz, D., Tochilkin, D., Laforte, C., Rombach, R., and Jampani, V. (2024). Sv3d: Novel multi-view synthesis and 3d generation from a single image using latent video diffusion. In *European Conference on Computer Vision*, pages 439–457. Springer.
- Wang, H., Xu, X., Xu, K., and Lau, R. W. (2023a). Lighting up nerf via unsupervised decomposition and enhancement. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 12632–12641.
- Wang, T., Zhu, X., Pang, J., and Lin, D. (2021). FCOS3D: fully convolutional one-stage monocular 3D object detection.
- Wang, T.-C., Liu, M.-Y., Zhu, J.-Y., Tao, A., Kautz, J., and Catanzaro, B. (2018). High-resolution image synthesis and semantic manipulation with conditional GANs. In *CVPR*.
- Wang, Y., Ma, C., and Liu, J. (2023b). SmartAssign: Learning a smart knowledge assignment strategy for deraining and desnowing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3677–3686.
- Wang, Y., Yu, J., and Zhang, J. (2023c). Zero-shot image restoration using denoising diffusion null-space model. *ICLR*.

- Wang, Z., Bovik, A. C., Sheikh, H. R., and Simoncelli, E. P. (2004). Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4):600–612.
- Wang, Z., Cun, X., Bao, J., Zhou, W., Liu, J., and Li, H. (2022). 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.
- Wei, C., Wang, W., Yang, W., and Liu, J. (2018). Deep retinex decomposition for low-light enhancement. arXiv:1808.04560.
- Wen, Q., Wu, Y., and Chen, Q. (2023). Video waterdrop removal via spatio-temporal fusion in driving scenes. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 10003–10009. IEEE.
- World Labs Team (2026). Atlas: A world model for spatial intelligence. *World Labs Blog*.
- Wu, G., Yi, T., Fang, J., Xie, L., Zhang, X., Wei, W., Liu, W., Tian, Q., and Wang, X. (2024a). 4D gaussian splatting for real-time dynamic scene rendering. *CVPR*.
- Wu, H., Qu, Y., Lin, S., Zhou, J., Qiao, R., Zhang, Z., Xie, Y., and Ma, L. (2021). Contrastive learning for compact single image dehazing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10551–10560.
- Wu, R., Mildenhall, B., Henzler, P., Park, K., Gao, R., Watson, D., Srinivasan, P. P., Verbin, D., Barron, J. T., Poole, B., et al. (2024b). ReconFusion: 3D reconstruction with diffusion priors. In *CVPR*.
- Wu, W., Weng, J., Zhang, P., Wang, X., Yang, W., and Jiang, J. (2022). Uretinex-net: Retinex-based deep unfolding network for low-light image enhancement. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5901–5910.
- Xiao, P., Shao, Z., Hao, S., Zhang, Z., Chai, X., Jiao, J., Li, Z., Wu, J., Sun, K., Jiang, K., et al. (2021). Pandaset: Advanced sensor suite dataset for autonomous driving. In *2021 IEEE international intelligent transportation systems conference (ITSC)*, pages 3095–3101. IEEE.
- Xie, D., Hu, X., Zhou, Y., and Duan, S. (2026). All-in-one adverse weather removal via dual state space-based diffusion model with degradation-aware guidance. *Pattern Recognition*, 171:112081.
- Xie, E., Wang, W., Yu, Z., Anandkumar, A., Alvarez, J. M., and Luo, P. (2021). SegFormer: Simple and efficient design for semantic segmentation with transformers. In *Neural Information Processing Systems (NeurIPS)*.

- Xie, S., Xu, Y., Gong, M., and Zhang, K. (2023). Unpaired image-to-image translation with shortest path regularization. In *CVPR*, pages 10177–10187.
- Xu, X., Wang, R., Fu, C.-W., and Jia, J. (2022). SNR-aware low-light image enhancement. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 17714–17724.
- Yadav, S., Mehra, A., Rohmetra, H., Ratnakumar, R., and Narang, P. (2021). Derain-GAN: Single image deraining using wasserstein GAN. *Multimedia Tools and Applications*, 80:36491–36507.
- Yang, L., Gu, X., and Sun, J. (2023). Generalized semantic segmentation by self-supervised source domain projection and multi-level contrastive learning. *Proceedings of the AAAI Conference on Artificial Intelligence*, 37(9):10789–10797.
- Yang, W., Tan, R. T., Feng, J., Guo, Z., Yan, S., and Liu, J. (2020a). Joint rain detection and removal from a single image with contextualized deep networks. *IEEE transactions on pattern analysis and machine intelligence*, 42(6):1377–1393.
- Yang, W., Wang, S., Fang, Y., Wang, Y., and Liu, J. (2020b). From fidelity to perceptual quality: A semi-supervised approach for low-light image enhancement. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 3063–3072.
- Yang, Y., Wu, H., Aviles-Rivero, A. I., Zhang, Y., Qin, J., and Zhu, L. (2024a). Genuine knowledge from practice: Diffusion test-time adaptation for video adverse weather removal. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 25606–25616.
- Yang, Z., Gao, X., Zhou, W., Jiao, S., Zhang, Y., and Jin, X. (2024b). Deformable 3D gaussians for high-fidelity monocular dynamic scene reconstruction. *CVPR*.
- Yang, Z., Teng, J., Zheng, W., Ding, M., Huang, S., Xu, J., Yang, Y., Hong, W., Zhang, X., Feng, G., et al. (2024c). CogVideoX: Text-to-video diffusion models with an expert transformer. arXiv:2408.06072.
- Yang, Z., Yang, H., Pan, Z., and Zhang, L. (2024d). Real-time photorealistic dynamic scene representation and rendering with 4D gaussian splatting. In *International Conference on Learning Representations (ICLR)*.
- Ye, H., Zhang, J., Liu, S., Han, X., and Yang, W. (2023). IP-Adapter: Text compatible image prompt adapter for text-to-image diffusion models. arXiv:2308.06721.
- Ye, M., Danelljan, M., Yu, F., and Ke, L. (2024). Gaussian grouping: Segment and edit anything in 3D scenes. In *ECCV*.

- Yeh, S.-Y., Hsieh, Y.-G., Gao, Z., Yang, B. B., Oh, G., and Gong, Y. (2023). Navigating text-to-image customization: From LyCORIS fine-tuning to model evaluation. *arXiv:2309.14859*.
- Yu, F., Chen, H., Wang, X., Xian, W., Chen, Y., Liu, F., Madhavan, V., and Darrell, T. (2020). Bdd100k: A diverse driving dataset for heterogeneous multitask learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2636–2645.
- Yu, Z., Chen, A., Huang, B., Sattler, T., and Geiger, A. (2024). Mip-splatting: Alias-free 3D gaussian splatting. *CVPR*.
- Zamfir, E., Wu, Z., Mehta, N., Khan, S., Zhang, Y., and Timofte, R. (2025). Complexity experts are task-discriminative learners for any image restoration. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.
- Zang, S., Ding, M., Smith, D., Tyler, P., Rakotoarivelo, T., and Kaafar, M. A. (2019). The impact of adverse weather conditions on autonomous vehicles: How rain, snow, fog, and hail affect the performance of a self-driving car. *IEEE Vehicular Technology Magazine*, 14(2):103–111.
- Zhang, D., Wang, C., Wang, W., Li, P., Qin, M., and Wang, H. (2024). Gaussian in the wild: 3d gaussian splatting for unconstrained image collections. In *European Conference on Computer Vision*, pages 341–359. Springer.
- Zhang, H., Ba, Y., Yang, E., Mehra, V., Gella, B., Suzuki, A., Pfahnl, A., Chandrappa, C. C., Wong, A., and Kadambi, A. (2023a). WeatherStream: Light transport automation of single image deweathering. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 13499–13509, Vancouver, BC, Canada. IEEE.
- Zhang, H., Goodfellow, I., Metaxas, D., and Odena, A. (2019a). Self-attention generative adversarial networks. In *ICML*.
- Zhang, H., Sindagi, V., and Patel, V. M. (2019b). Joint transmission map estimation and dehazing using deep networks. *IEEE Transactions on Circuits and Systems for Video Technology*, 30(7):1975–1986.
- Zhang, J., Ren, W., Zhang, S., Zhang, H., Nie, Y., Xue, Z., and Cao, X. (2022). Hierarchical density-aware dehazing network. *IEEE Transactions on Cybernetics*, 52(10):11187–11199.
- Zhang, K., Li, R., Yu, Y., Luo, W., and Li, C. (2021a). Deep dense multi-scale network for snow removal using semantic and depth priors. *IEEE Transactions on Image Processing*, 30:7419–7431.

- Zhang, L., Rao, A., and Agrawala, M. (2023b). Adding conditional control to text-to-image diffusion models.
- Zhang, R., Isola, P., Efros, A. A., Shechtman, E., and Wang, O. (2018). The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 586–595.
- Zhang, Y., Guo, X., Ma, J., Liu, W., and Zhang, J. (2021b). Beyond brightening low-light images. *International Journal of Computer Vision*, 129:1013–1037.
- Zhang, Y., Zhang, J., and Guo, X. (2019c). Kindling the darkness: A practical low-light image enhancer. In *Proceedings of the 27th ACM international conference on multimedia*, pages 1632–1640.
- Zhao, H., Shi, J., Qi, X., Wang, X., and Jia, J. (2017). Pyramid scene parsing network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2881–2890.
- Zhao, L., Wang, P., and Liu, P. (2024). Bad-gaussians: Bundle adjusted deblur gaussian splatting. In *European Conference on Computer Vision (ECCV)*.
- Zhao, Y., Zhong, Z., Zhao, N., Sebe, N., and Lee, G. H. (2022). Style-hallucinated dual consistency learning for domain generalized semantic segmentation. In *Proceedings of the European Conference on Computer Vision (ECCV)*.
- Zheng, Q., Sun, H., Xu, H., and Xu, F. (2024). Learning novel view synthesis from heterogeneous low-light captures. arXiv:2403.13337.
- Zheng, Y., Zhan, J., He, S., Dong, J., and Du, Y. (2023). Curricular contrastive regularization for physics-aware single image dehazing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5785–5794.
- Zheng, Z., Wu, Y., Han, X., and Shi, J. (2020). Forkgan: Seeing into the rainy night. In *Computer Vision—ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part III 16*, pages 155–170. Springer.
- Zhou, X., Lin, Z., Shan, X., Wang, Y., Sun, D., and Yang, M.-H. (2024). DrivingGaussian: Composite gaussian splatting for surrounding dynamic autonomous driving scenes. *CVPR*.
- Zhu, J.-Y., Park, T., Isola, P., and Efros, A. A. (2017). Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Proceedings of the IEEE international conference on computer vision*, pages 2223–2232.

- Zhu, Y., Wang, T., Fu, X., Yang, X., Guo, X., Dai, J., Qiao, Y., and Hu, X. (2023). Learning weather-general and weather-specific features for image restoration under multiple adverse weather conditions. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 21747–21758.
- Zhuang, J., Wang, C., Liu, L., Lin, L., and Li, G. (2023). DreamEditor: Text-driven 3D scene editing with neural fields. *SIGGRAPH Asia*.
- Zsolnai-Fehér, K. (2022). The flow from simulation to reality. *nature physics*, 18(11):1260–1261.