Equirectangular Point Reconstruction for Domain Adaptive Multimodal 3D Object Detection in Adverse Weather Conditions (Supplementary Material)

Anonymous submission

Supplementary Material

The supplementary material provides a detailed explanation of our methods, including visual results and additional experiments. Section A introduces the processes of our methods, such as the auxiliary 2D detection loss function, the object-centric point generator, domain adaptation, and weather noise matching algorithms. Section B includes additional experiments at easy and hard difficulty levels, and other classes. Section C presents extensive ablation studies on the weather noise matching method, the DA module, and hyperparameters. Section D presents visual results of equirectangular point reconstruction and compares restoration and detection results. This section also includes t-SNE visualizations and weather noise matching results. Section E offers a broader discussion on EquiDetect. Finally, Section F presents limitations and future work on EquiDetect.

A Method Details

A.1 Auxiliary 2D Detection Loss Function

For auxiliary 2D detection in an image, a keypoint-based object detection method inspired by CenterNet (Zhou, Wang, and Krähenbühl 2019) was used. The 2D detection loss is defined as follows:

$$\mathcal{L}_{2Ddet} = \mathcal{L}_{key} + \mathcal{L}_{size} + \mathcal{L}_{off}, \tag{1}$$

where $\mathcal{L}_{key}$ denotes the keypoint loss for the center points using the focal loss, and $\mathcal{L}_{size}$ represents the object size loss using the $L1$ loss, which regresses the height and width of the bounding box. Moreover, $\mathcal{L}_{off}$ indicates the offset loss using the $L1$ loss.

A.2 Object-centric Ray Generation

We provide pseudocode for understanding the process of object-centric ray generation in Algorithm 1.

A.3 Domain Adaptation

We provide pseudocode for understanding the training stage and inference stage processes of the domain adaptation module in Algorithms 2 and 3.

Algorithm 1: Object-centric Ray Generation

Input: Range image $\hat{r}$, 2D bounding boxes $B_{2D_{i\in\{1,2,\dots,I\}}}$
Output: Updated range image $\hat{r}'$ with generated points

1: **for** each $i \in \{1, 2, ..., I\}$ **in parallel do**
2: Extract object patch O_i from $\hat{r}$ using B_{2D_i}.
3: Estimate the distance $\hat{d}_i$ of O_i by averaging the values at N central coordinates (u_n, v_n) within O_i.

$$\hat{d}_i = \frac{1}{N}\sum_{n=1}^{N} O_i(u_n, v_n)$$

4: Determine the optimal number of points P_{opt} based on $\hat{d}_i$.
5: Insert zero-padding between the pixels of O_i based on P_{opt}, resulting in O_{pad_i}.
6: Create a zero-padding mask M_i corresponding to O_{pad_i}.
7: Apply inpainting to the $1-M_i$ regions of O_{pad_i} using an inpainting model $U(\cdot)$, resulting in the generated patch O_{gen_i}.

$$O_{gen_i} = O_{pad_i} \odot M_i + U(O_{pad_i}) \odot (1 - M_i)$$

8: Update the range image $\hat{r}$ with O_{gen_i} to produce the final range image $\hat{r}'$.
9: **end for**
10: **Return** $\hat{r}'$

Algorithm 2: Domain Adaptation in Training Stage

Input: Source data, Target data, Pretrained Feature extractor on source data
Output: Fine-tuned feature extractor

1: Extract source feature f^s and target feature f^t from source data and target data using the pretrained feature extractor, repectively.
2: Calculate the maximum mean discrepancy loss ($\mathcal{L}_{mmd}$) between f^s and f^t.

$$\mathcal{L}_{mmd} \leftarrow \left\| \frac{1}{n_s}\sum_{p=1}^{n_s} f_p^s - \frac{1}{n_t}\sum_{q=1}^{n_t} f_q^t \right\|^2$$

3: Fine-tune feature extractor using $\mathcal{L}_{mmd}$.
4: **return** Fine-tuned feature extractor

Algorithm 3: Domain Adaptation in Inference Stage

Input: Fine-tuned feature extractor, Target data, Pretrained discriminator on f^s

Output: Perturbed target feature $\hat{f}^t$

Extract f^t from the target data using fine-tuned feature extractor.
for $h = 1$ to $\{1, 2, \cdots, g\}$ **do**
  Compute gradient of discriminator loss.

$$\nabla_{f^t}\mathcal{L}_{dis} \leftarrow \frac{1}{n^s}\sum_{p=1}^{n^s}\|\Psi(f_h^t;\mathcal{W}) - c\|_2$$

  Perturbation update with step size α.

$$f_h^t \leftarrow f_h^t - \alpha \cdot sign(\nabla_{f^t}\mathcal{L}_{dis})$$

  Mapping in the ϵ range.

$$f_{h+1}^t \leftarrow \Pi_{f^t,\epsilon}(f_h^t)$$

end for
$\hat{f}^t \leftarrow f_g^t$
Return: Perturbed target feature $\hat{f}^t$

A.4 Weather Noise Matching via Radial Manipulation

We present pseudocode in Algorithm 4 to clarify the process of the noise matching method.

B Additional Experiments

Tables 1 and 2 present the evaluation of 3D object detection models on the S-KITTI dataset, focusing on the car class at easy and hard difficulty levels. These models were trained using the S-KITTI training dataset, consisting of 14,848 frames with an equal ratio of clean, rain, snow, and fog conditions. Experimental results demonstrate that our proposed model outperforms other 3D object detection models in all weather conditions not only for the moderate difficulty level but also for easy and hard objects.

Table 3 shows the mulimodal 3D object detection results on another classes (pedestrian and cyclist) at a moderate difficulty level. We trained and evaluated the 3D object detection models on S-KITTI. The proposed model consistently achieves the highest performance in all weather conditions (clean, snow, rain, and fog). These experimental results further demonstrate the effectiveness of our model across different object classes.

C Additional Ablation Studies

In Table 4, we analyze the performance on the real-world datasets (Dense (Bijelic et al. 2020) and CADC (Pitropov et al. 2021)) when models are trained on synthetic data generated by existing simulators compared to synthetic data generated using our method. For all models, training with the dataset generated using our matching method results in better performance on the real-world datasets. This demonstrates that the proposed method produces more realistic data that closely resembles the real-world datasets.

Table 5 presents a comparison of the effects according to the DA modules (training stage and inference stage). Experiments show that both the training stage and the reference stage achieve the best performance when applied. From this result, Table 5 reveals the domain alignment performance through MMD and feature perturbations.

Table 6 analyze the effects of the filter size according to R in the distance-constrained denoising. The experiment is evaluated on the S-KITTI dataset as an average across all weather conditions. As the filter size increases, the performance decreases, and the best performance is achieved when the filter size is 3.

Algorithm 4: Weather Noise Matching via Radial Manipulation

Input: Weather particles $M = \{m_1, m_2, \cdots, m_l\}$,
Noise points $Q = \{q_1, q_2, \cdots, q_n\}$

Output: Manipulated points P_M

Generate projected mask-point image represented M and Q.
for o in $\{1, 2, \cdots, l\}$ **do**
  Select k-nearest candidate points $candi_k$ in noise points for m_o.
  for e in $\{1, 2, \cdots, k\}$ **do**
    Manipulate radial direction of $candi_e$ by $\Delta\beta_e$.

$$candi'_e \leftarrow (r_{candi_e} + \Delta\beta_e, \theta_{candi_e}, \phi_{candi_e})$$

    if $r_{candi_e} + \Delta\beta_e$ in LiDAR measurement range **then**
      Transform spherical coordinates (r, θ, ϕ) of $candi'_e$ onto the 2D image coordinates (u, v).

$$candi''_e \leftarrow proj(candi'_e)$$

      Select $\Delta\hat{\beta}_e$ the optimal value of $\Delta\beta_e$ to minimize the distance to the m_o.

$$\Delta\hat{\beta}_e = \underset{\Delta\beta_e}{\operatorname{argmin}}\|(u_{m_o}, v_{m_o}) - candi''_e\|_2$$

      Manipulate radial direction of $candi'_e$ by $\hat{\Delta}\beta_e$.

$$candi'_e \leftarrow (r_{candi_e} + \Delta\hat{\beta}_e, \theta_{candi_e}, \phi_{candi_e})$$

    end if
  end for
  Select the noise point $noise_e$ with minimal distance to m_o.

$$noise_e \leftarrow \underset{k}{\operatorname{argmin}}\|center(m_o) - proj(candi'_k)\|_2$$

  if $noise_e$ overlap with m_o **then**
    Manipulated points$P_{m_o} \leftarrow noise_e$
  end if
end for
Return: Manipulated points P_M

Metric	$3D_{R40}\,AP(\%)$			
Weather	Clean	Snow	Rain	Fog
LiDAR-based				
PV-RCNN (Shi et al. 2020)	91.82	82.57	82.74	83.50
VoxelNeXt (Chen et al. 2023)	92.79	79.33	77.14	82.14
GD-MAE (Yang et al. 2023)	91.49	78.57	78.90	81.48
HINTED (Xia et al. 2024)	92.12	81.67	82.58	85.24
Multimodal				
Focals Conv (Chen et al. 2022)	91.94	84.29	85.43	84.24
Graph-VoI (Yang et al. 2022)	93.31	82.67	84.00	85.19
SFD (Wu et al. 2022)	92.77	84.08	84.62	85.27
TED-M (Wu et al. 2023a)	92.43	84.94	85.95	88.49
VirConv-S (Wu et al. 2023b)	**94.17**	86.22	86.91	88.74
Ours	93.06	**91.93**	**92.09**	**91.62**

Table 1: Results of 3D object detectors on S-KITTI for various weather conditions based on the car class at **easy** difficulty.

Metric	$3D_{R40}\,AP(\%)$			
Weather	Clean	Snow	Rain	Fog
LiDAR-based				
PV-RCNN (Shi et al. 2020)	81.72	61.11	61.15	66.68
VoxelNeXt (Chen et al. 2023)	76.71	52.14	51.60	65.75
GD-MAE (Yang et al. 2023)	77.97	51.57	52.76	62.32
HINTED (Xia et al. 2024)	82.41	64.32	65.48	68.15
Multimodal				
Focals Conv (Chen et al. 2022)	80.81	64.25	64.42	66.37
Graph-VoI (Yang et al. 2022)	82.16	63.88	64.79	67.63
SFD (Wu et al. 2022)	81.26	64.30	66.06	68.04
TED-M (Wu et al. 2023a)	85.38	67.61	63.38	73.08
VirConv-S (Wu et al. 2023b)	85.83	69.41	70.01	75.87
Ours	**86.48**	**73.47**	**73.72**	**80.05**

Table 2: Results of 3D object detectors on S-KITTI for various weather conditions based on the car class at **hard** difficulty.

Metric		$3D_{R40}\,AP(\%)$			
Weather		Clean	Snow	Rain	Fog
Pedestrian	Focals Conv	66.90	47.13	46.73	47.01
	Graph-VoI	68.87	46.67	47.11	49.03
	SFD	67.12	44.84	47.00	48.15
	TED-M	71.69	46.06	48.07	48.49
	VirConv-S	73.36	48.35	49.71	50.53
Ours		**74.03**	**55.89**	**57.10**	**52.75**
Cyclist	Focals Conv	75.45	50.95	50.91	54.86
	Graph-VoI	76.11	51.38	52.60	55.01
	SFD	72.93	50.54	49.86	52.78
	TED-M	75.81	51.06	52.97	53.90
	VirConv-S	80.63	53.18	54.41	57.11
Ours		**81.98**	**55.36**	**56.71**	**58.35**

Table 3: The results of multimodal 3D object detectors on S-KITTI for various weather conditions. The results are evaluated based on **the pedestrian and cyclist classes** at moderate difficulty.

In Table 7, we analyze the effects of the hyperparameter λ_{2D}^{aux} in the total loss function. In this experimental setting, when λ is 0, the auxiliary restoration task is not activated.

Metric		$3D_{R40}\,AP(\%)$			
Data		Dense			CADC
Weather		Snow	Rain	Fog	Snow
Physical-based simulator (Dong et al. 2023)	Focals Conv	29.26	30.01	22.95	29.73
	Graph-VoI	30.05	30.65	29.14	37.32
	SFD	22.03	22.87	20.61	27.03
	TED-M	28.97	27.09	22.73	36.64
	VirConv-S	34.81	33.29	32.11	37.51
	Ours	35.68	35.02	33.97	38.71
Physical-based simulator + Weather noise matching	Focals Conv	31.57	30.46	24.57	35.34
	Graph-VoI	32.72	32.64	31.75	42.41
	SFD	24.86	23.83	21.78	28.23
	TED-M	30.72	29.10	25.06	42.81
	VirConv-S	36.32	35.32	34.92	43.18
	Ours	**37.91**	**37.07**	**35.29**	**45.66**

Table 4: Ablation study for **the weather noise matching method** on the Dense and CADC datasets for weather conditions based on the car class at moderate difficulty.

Training stage	Inference stage	$3D_{R40}\,AP(\%)$	
		S-KITTI → Dense	S-KITTI → CADC
✓	✓	**45.80**	**52.09**
✓		42.78	50.61
	✓	41.88	48.06
		37.59	45.66

Table 5: Ablation study for **domain adaptation module** at moderate difficulty.

Filter Size	3	5	7	9
$3D_{R40}\,AP(\%)$	**81.71**	81.57	80.14	78.69

Table 6: Ablation study for **distance-constrained denoising filter size** on S-KITTI dataset at moderate difficulty.

λ_{2D}^{aux}	0	0.01	0.1	1	10
$3D_{R40}\,AP(\%)$	79.14	80.94	**81.71**	77.56	75.32

Table 7: Ablation study for λ_{2D}^{aux} on S-KITTI dataset at moderate difficulty.

ϵ	0.015	0.015	0.03	0.03	0.045	0.045
α	0.001	0.01	0.001	0.01	0.001	0.01
$3D_{R40}\,AP(\%)$	80.11	80.05	80.38	**81.71**	79.08	79.58

Table 8: Ablation study for ϵ and α on S-KITTI dataset at moderate difficulty.

The experimental results show the highest performance at $\lambda = 0.1$. The results indicate that the auxiliary restoration task contributes to object detection, as the performance is higher than when the restoration functionality is disabled. In contrast, when the value of λ is too low or too high, it hinders the primary detection task. If λ is too high, the model may focus too much on the restoration task, hindering detection. Conversely, if λ is too low, the auxiliary restoration functionality may not work effectively, offering little benefit to the detection performance.

In Table 8, we evaluate the effects of various settings for the parameters ϵ and α in feature perturbation. The best performance is observed when ϵ is set to 0.03 and α is set to 0.01. Hence, these parameters are configured to their opti-

mal values based on the results.

D Visual Results

Figures 1 and 2 exhibit the visual results regarding the pepper noise property observed in adverse weather conditions on S-KITTI and Dense datasets. The first row presents the range image transformed from the point cloud by equirectangular projection, and the second row presents the RGB image corresponding to each range image in the first row. In the clear weather condition, the background in the range image appears generally smooth. However, in adverse weather conditions such as snow and rain, the images exhibit a distribution of noise similar to pepper noise. This noise occurs because the laser beams are reflected by adverse weather elements before reaching the intended background or objects, reducing the radial value r of each point and causing them to appear darker in the range image. Based on these observations, we applied the distance-constrained median filtering in the range image to remove the adverse weather noise.

Figures 3 and 4 show the distance-constrained denoising results for snow and rain conditions, respectively, on the real-world weather dataset Dense. The proposed method effectively removes adverse weather noise with the pepper noise property in range images and demonstrates its effectiveness in the point cloud as well.

Figures 5 and 6 show detection results for snow and rain conditions, respectively, on S-KITTI dataset. Existing detection models (TED-M and VirConv-S) fail to predict distant objects, whereas the proposed method effectively removes noise across the entire scene and increases the points on distant objects, resulting in successful detection of these objects.

Figures 7 and 8 show detection results on real-world adverse weather dataset Dense. Additionally, Fig. 9 presents detection results on real-world snow weather dataset CADC. The compared models (TED-M and VirConv-S) fail to detect most objects or even produce incorrect predictions, whereas the proposed model demonstrates robust prediction results even in real-world adverse weather conditions. Notably, the proposed model shows effective restoration performance not only on synthetic data but also in real-world environments.

Figures 10 and 11 present t-SNE (Van der Maaten and Hinton 2008) visualization results by DA module on the Dense dataset. The figures represent the source (S-KITTI) and target (Dense) domain features as red and blue points, respectively. As shown on the left side of the figures, the input distributions of the two domains are independent. In the center of the figures, after the training stage using MMD Loss, the distance of the mean discrepancy between the two domains has decreased. However, there are limitations in aligning the fine-grained domain regions. Subsequently, when feature perturbation is applied during the inference stage, as shown on the right side of the figures, the fine-grained distributions of the two domains are aligned. These figures demonstrate the superior performance of our DA module.

Figures 12 and 13 present the visual comparison between existing adverse weather dataset using physical-based simulator (Dong et al. 2023) and S-KITTI dataset which is applied noise matching method. The top row of the figures shows the results from the existing simulator, while the bottom row shows the results using our matching method. As can be seen in the figures, when using two independent simulators, the weather-element alignment is inconsistent. In contrast, when using our method, the alignment is adjusted. As a result, our weather noise matching method generates realistic multimodal synthetic dataset.

E Discussion

E.1 Reconstruction Complexity in Object Detection

Reconstructing degraded data in adverse weather conditions introduces the substantial complexity in both 2D and 3D object detection. The traditional approaches often employ complex models that are designed to process full resolution and entire scenes, leading to significant computational overhead. In contrast, our approach utilizes the characteristics of LiDAR data to implement a non-learning-based noise removal method and an object-centric generation method. Specifically, the distance-constrained denoising method for noise removal offers a simple structure that requires no learning, unlike conventional learning-based methods. Additionally, the object-centric ray generator focuses on processing specific regions, making it highly effective in reducing complexity. When traditional 2D image restoration methods are applied to detection tasks, their independent structure requires fully restored outputs. This necessitates the extraction of new features from the complete image for object detection, increasing complexity. By contrast, despite the inherent complexity of the backbone (Yang et al. 2022), our proposed auxiliary restoration network mitigates overall complexity by performing restoration at a quarter size and utilizing features from the restoration process, rather than re-extracting new features from a fully restored image. Consequently, our approach integrates efficiently with multimodal 3D object detection, leading to a significant reduction in overall complexity.

E.2 Restoration Dependency in Object Detection

For object detection, existing methods typically involve performing a complete restoration of the entire scene before extracting features for detection. This sequential process increases complexity and dependency because it requires a full restoration step followed by feature extraction, which not only adds computational burden but also introduces a high dependency on the restoration accuracy. If the restoration step is flawed, it directly affects the detection performance. On the other hand, our approach reduces dependency on restoration by incorporating the proposed auxiliary loss. This allows the detection model to learn robust feature representations that are less sensitive to imperfections in the restoration process. Consequently, even if the restoration is incomplete or contains errors, the detection performance remains stable, making our method more effective and reliable in adverse conditions.

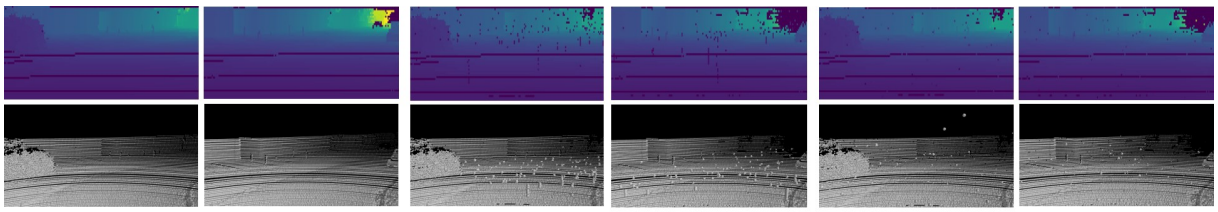

Figure 1: Visual results of range images regarding the **pepper noise** in synthetic weather conditions on **S-KITTI dataset**. The first row presents range images and second row presents corresponding point clouds. The range images in clear weather condition is smooth, whereas range images in adverse weather conditions (snow and rain) exhibit noise similar to pepper noise.

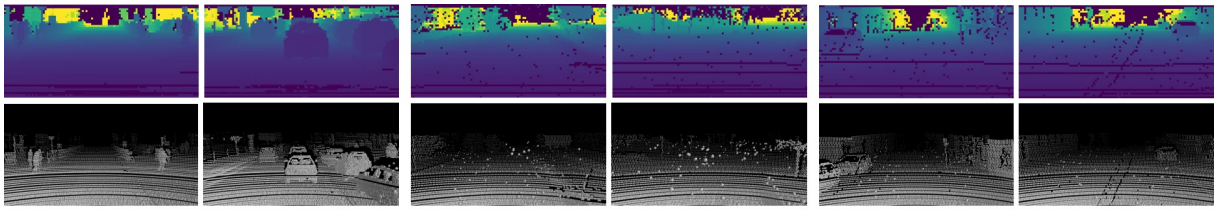

Figure 2: Visual results of range images regarding the **pepper noise** in real-world weather conditions on **Dense dataset**. The first row presents range images and second row presents corresponding point clouds. The range images in clear weather condition is smooth, whereas range images in adverse weather conditions (snow and rain) exhibit noise similar to pepper noise.

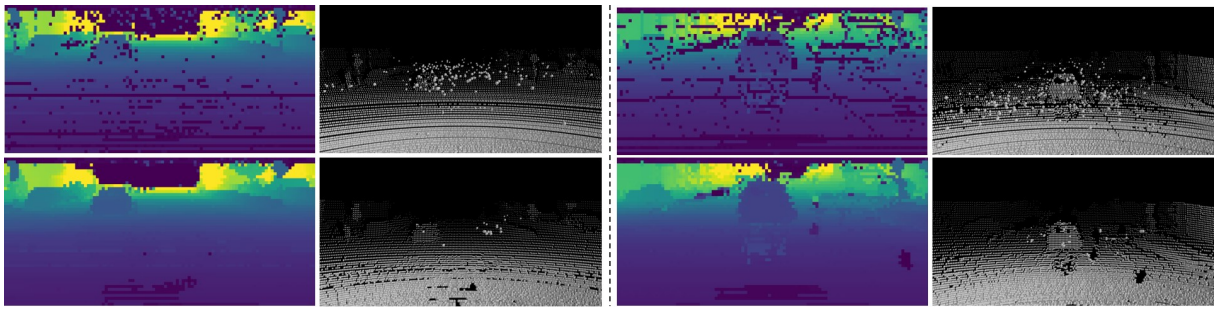

Figure 3: Visual results of **distance-constrained denoising** in real-world weather condition on Dense dataset. The first row presents point clouds in **snow** condition and corresponding range images. The second row presents the denoised point clouds and range images corresponding the first row.

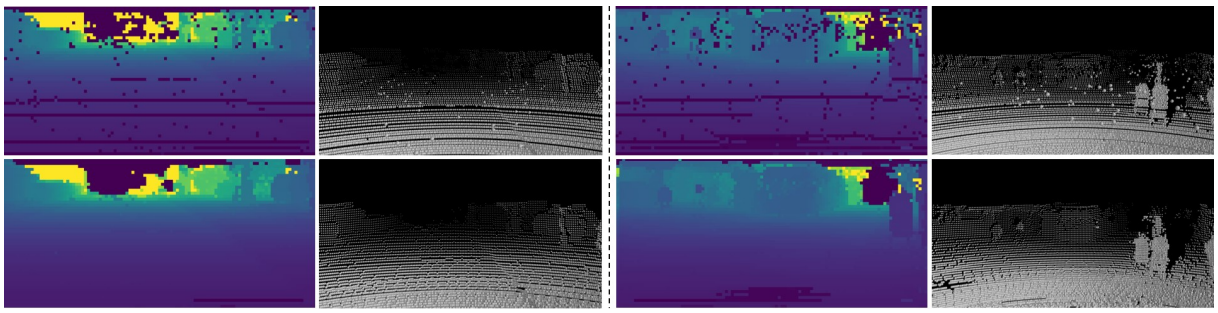

Figure 4: Visual results of **distance-constrained denoising** in real-world weather condition on Dense dataset. The first row presents point clouds in **rain** condition and corresponding range images. The second row presents the denoised point clouds and range images corresponding the first row.

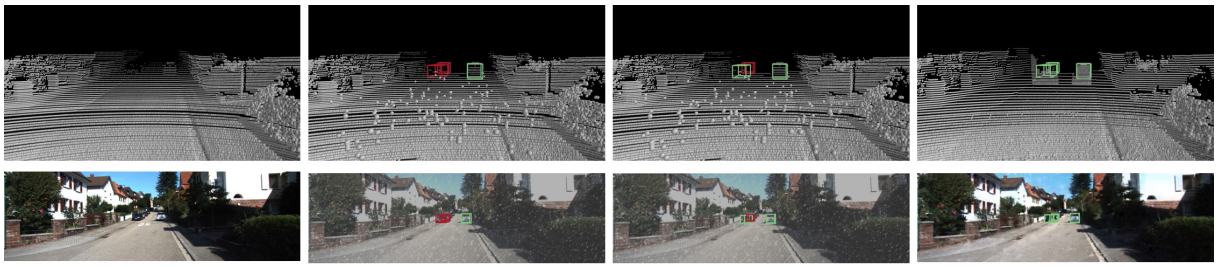

Figure 5: Comparison of **visual restoration and detection** results in **snow** condition on the S-KITTI dataset. (a) Ground truth point cloud and image. (b) Visual detection results of TED-M. (c) Visual detection results of VirConv-S. (d) Visual detection and restoration results of EquiDetect. The red boxes are defined as missed prediction and the green boxes are defined as correct prediction.

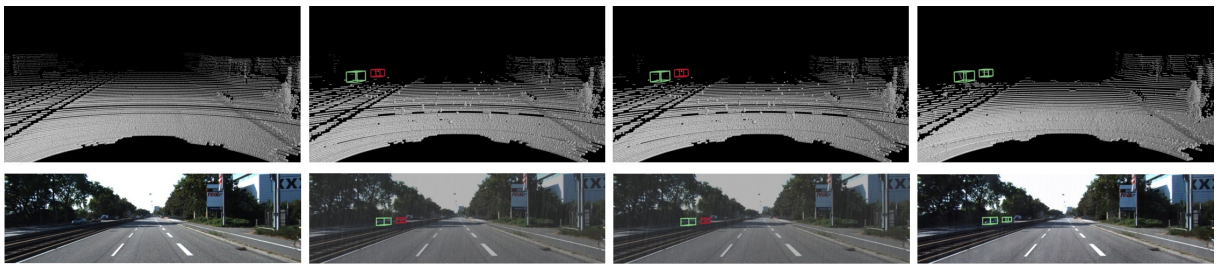

Figure 6: Comparison of **visual restoration and detection** results in **rain** condition on the S-KITTI dataset. (a) Ground truth point cloud and image. (b) Visual detection results of TED-M. (c) Visual detection results of VirConv-S. (d) Visual detection and restoration results of EquiDetect. The red boxes are defined as missed prediction and the green boxes are defined as correct prediction.

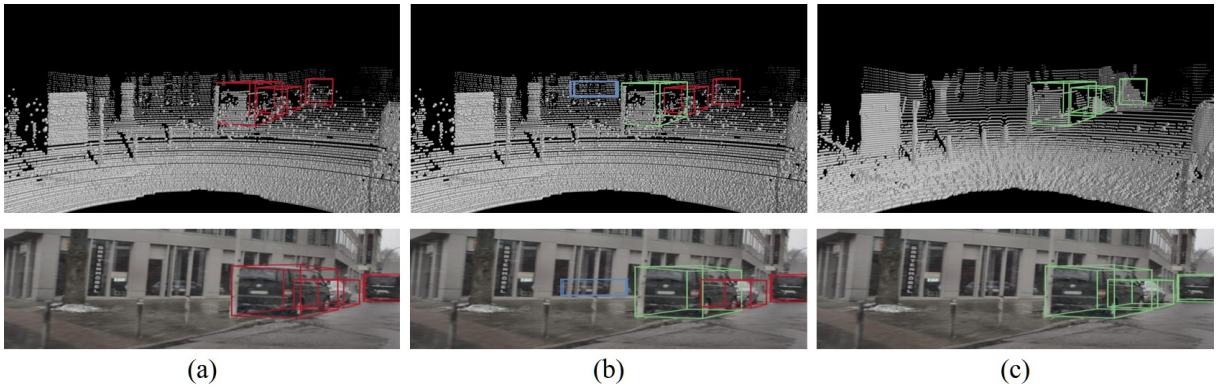

Figure 7: Comparison of **visual restoration and detection** results in real-world **snow** condition on the Dense dataset. (a) Visual detection results of TED-M. (b) Visual detection results of VirConv-S. (c) Visual detection and restoration results of EquiDetect. The red boxes are defined as missed prediction, the blue box is defined as incorrect prediction, and the green boxes are defined as correct prediction.

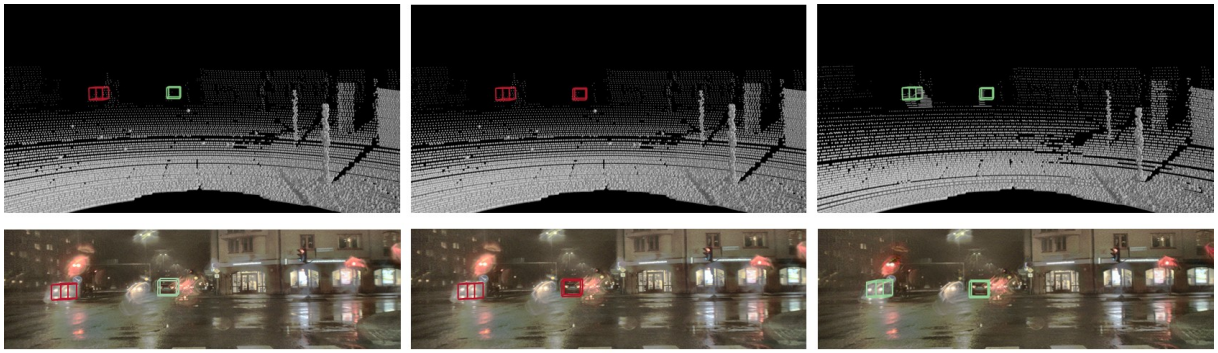

Figure 8: Comparison of **visual restoration and detection** results in real-world **rain** condition on the Dense dataset. (a) Visual detection results of TED-M. (b) Visual detection results of VirConv-S. (c) Visual detection and restoration results of EquiDetect. The red boxes are defined as missed prediction and the green boxes are defined as correct prediction.

Figure 9: Comparison of **visual restoration and detection** results in real-world **snow** condition on the CADC dataset. (a) Visual detection results of TED-M. (b) Visual detection results of VirConv-S. (c) Visual detection and restoration results of EquiDetect. The red boxes are defined as missed prediction and the green boxes are defined as correct prediction.

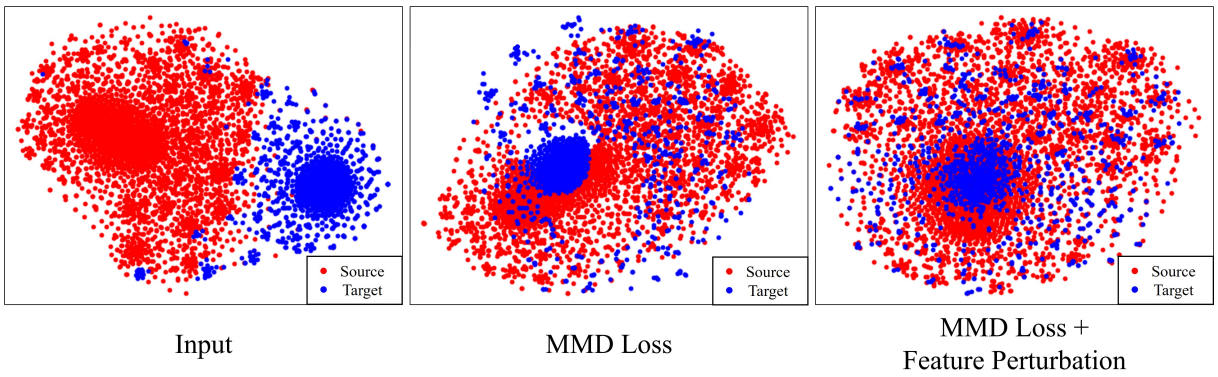

Figure 10: Visualization of t-SNE results by the **3D DA module** on the Dense dataset.

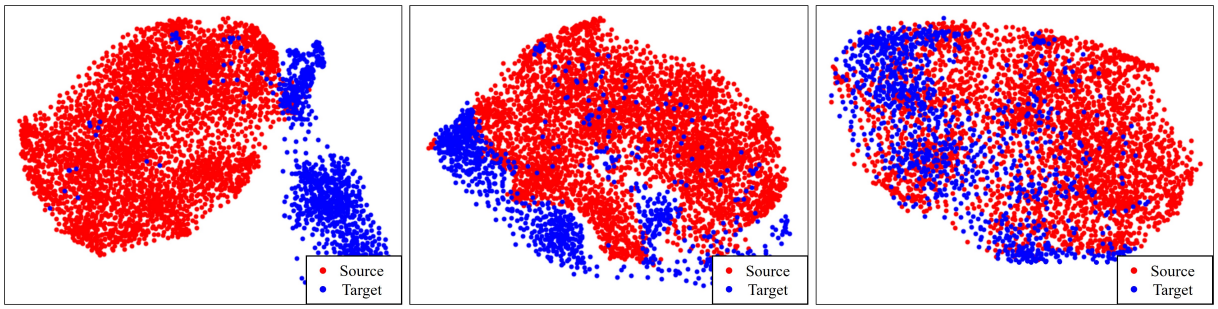

Figure 11: Visualization of t-SNE results by the **2D DA module** on the Dense dataset.

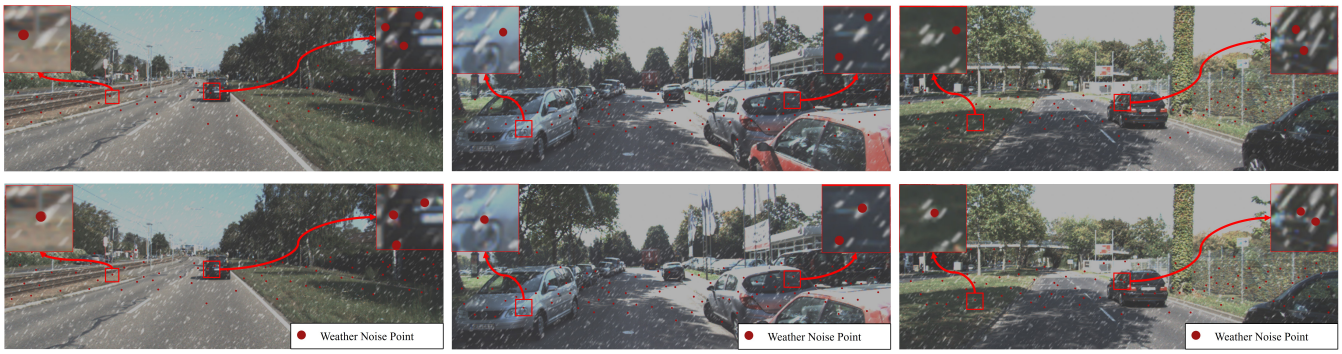

Figure 12: Comparison of simulation results in **snow** conditions on the S-KITTI dataset. The top row shows projected images with noise points and simulated images from the physics-based simulator. The bottom row shows projected images with noise points and simulated images from the physics-based simulator using our **weather noise matching method**.

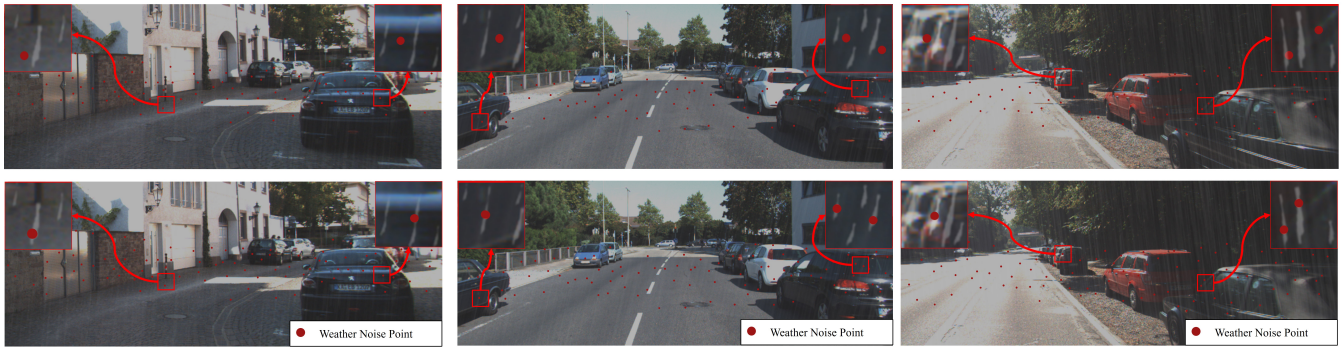

Figure 13: Comparison of simulation results in **rain** conditions on the S-KITTI dataset. The top row shows projected images with noise points and simulated images from the physics-based simulator. The bottom row shows projected images with noise points and simulated images from the physics-based simulator using our **weather noise matching method**.

E.3 Other Datasets

Our research aims to develop a robust model for autonomous driving in adverse weather conditions. Although there are typical multimodal datasets available, such as Waymo (Sun et al. 2020) or nuScenes (Caesar et al. 2020) for 3D object detection, these datasets were not collected with the purpose of addressing adverse weather conditions. Therefore, we focused on using datasets like Dense and CADC, which are specifically collected for adverse weather conditions and align with the primary goal of our work.

F Limitations and Future Work

Although the proposed weather noise matching method improves alignment of synthetic noise between modalities, it still faces challenges in addressing the realism of synthetic weather data for each modality. The method improves performance in real-world scenarios, but realism for each simulator remains an issue. Additionally, the generated data may struggle to model all possible corruptions encountered in real-world environments. This limitation can lead to dependency on the quality of the underlying simulators. As future work, we aim to develop a more robust fusion model that can withstand various weather conditions by utilizing diverse modalities, such as Radar, in autonomous driving environments.

References

Bijelic, M.; Gruber, T.; Mannan, F.; Kraus, F.; Ritter, W.; Dietmayer, K.; and Heide, F. 2020. Seeing through fog without seeing fog: Deep multimodal sensor fusion in unseen adverse weather. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 11682–11692.

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*, 11621–11631.

Chen, Y.; Li, Y.; Zhang, X.; Sun, J.; and Jia, J. 2022. Focal sparse convolutional networks for 3d object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 5428–5437.

Chen, Y.; Liu, J.; Zhang, X.; Qi, X.; and Jia, J. 2023. Voxelnext: Fully sparse voxelnet for 3d object detection and tracking. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 21674–21683.

Dong, Y.; Kang, C.; Zhang, J.; Zhu, Z.; Wang, Y.; Yang, X.; Su, H.; Wei, X.; and Zhu, J. 2023. Benchmarking robustness of 3d object detection to common corruptions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 1022–1032.

Pitropov, M.; Garcia, D. E.; Rebello, J.; Smart, M.; Wang, C.; Czarnecki, K.; and Waslander, S. 2021. Canadian adverse driving conditions dataset. *The International Journal of Robotics Research*, 40(4-5): 681–690.

Shi, S.; Guo, C.; Jiang, L.; Wang, Z.; Shi, J.; Wang, X.; and Li, H. 2020. Pv-rcnn: Point-voxel feature set abstraction for 3d object detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 10529–10538.

Sun, P.; Kretzschmar, H.; Dotiwalla, X.; Chouard, A.; Patnaik, V.; Tsui, P.; Guo, J.; Zhou, Y.; Chai, Y.; Caine, B.; et al. 2020. Scalability in perception for autonomous driving: Waymo open dataset. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2446–2454.

Van der Maaten, L.; and Hinton, G. 2008. Visualizing data using t-SNE. *Journal of machine learning research*, 9(11).

Wu, H.; Wen, C.; Li, W.; Li, X.; Yang, R.; and Wang, C. 2023a. Transformation-equivariant 3D object detection for autonomous driving. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, 2795–2802.

Wu, H.; Wen, C.; Shi, S.; Li, X.; and Wang, C. 2023b. Virtual Sparse Convolution for Multimodal 3D Object Detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 21653–21662.

Wu, X.; Peng, L.; Yang, H.; Xie, L.; Huang, C.; Deng, C.; Liu, H.; and Cai, D. 2022. Sparse fuse dense: Towards high quality 3d detection with depth completion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 5418–5427.

Xia, Q.; Ye, W.; Wu, H.; Zhao, S.; Xing, L.; Huang, X.; Deng, J.; Li, X.; Wen, C.; and Wang, C. 2024. HINTED: Hard Instance Enhanced Detector with Mixed-Density Feature Fusion for Sparsely-Supervised 3D Object Detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 15321–15330.

Yang, H.; He, T.; Liu, J.; Chen, H.; Wu, B.; Lin, B.; He, X.; and Ouyang, W. 2023. GD-MAE: generative decoder for MAE pre-training on lidar point clouds. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 9403–9414.

Yang, H.; Liu, Z.; Wu, X.; Wang, W.; Qian, W.; He, X.; and Cai, D. 2022. Graph r-cnn: Towards accurate 3d object detection with semantic-decorated local graph. In *European Conference on Computer Vision*, 662–679. Springer.

Zhou, X.; Wang, D.; and Krähenbühl, P. 2019. Objects as points. *arXiv preprint arXiv:1904.07850*.