LiDAR Snowfall Simulation for Robust 3D Object Detection (Supplementary Materials)

Martin Hahner [1] Christos Sakaridis [1] Mario Bijelic [2]
Felix Heide [2] Fisher Yu [1] Dengxin Dai [1,3] Luc Van Gool [1,4]

[1] ETH Zürich [2] Princeton University [3] MPI for Informatics [4] KU Leuven

1. Introduction

In this supplemental document we aim to provide additional information supporting the results in the main manuscript. We provide additional information about our snowfall simulation in Sec. 2 and our wet ground simulation in Sec. 3. In Sec. 4, we provide additional insights into the SeeingThroughFog (STF) [1] dataset. In Sec. 5 and Sec. 6, we provide additional quantitative and qualitative results, respectively. In Sec. 7 we discuss known limitations of our work.

2. Additional Snowfall Simulation Details

For each row in Tab. 1 we sample $n_l = 64$ snowflake distribution patterns in 2D according to Algorithm 1. For this sampling, we use $R_{\text{max}} = 80$m, since the real data provided by the SeeingThroughFog (STF) dataset [1] suggests that the effective range of the Velodyne HDL-S3D is only 80m and not 120m as stated in its manual [17]. Two example snowflake distribution patterns are shown in Fig. 1. During training, we then sample uniformly at random from the rows given in Tab. 1 and load the corresponding precomputed snowflake distribution patterns from disk.

The precipitation rate for snow r_s expressed in rate of equivalent water, depends on the precipitation rate for rain, r_r, through

$$r_s = 487 \rho_s D_0 v_s \sqrt[3]{r_r^2}, \tag{1}$$

with the average snowflake diameter D_0 in m, the average terminal velocity of the snowflakes v_s in m/s and the average density of snow ρ_s in g/cm^3 [12]. Conversely, the rainfall rate r_r can be derived from the snowfall rate r_s

$$r_r = \sqrt{\left(\frac{r_s}{487 \rho_s D_0 v_s} \right)^3}. \tag{2}$$

We use this connection *e.g.* to compare our simulation(s) to LISA [9], which takes as input the rainfall rate r_r and not directly the snowfall rate r_s as our snowfall simulation (see Algorithm 1 in the main manuscript).

In this supplemental document, we also present how we sample snowflakes in Algorithm 1 and compute the beam occlusions in Algorithm 2.

3. Additional Wet Ground Simulation Details

This section provides more implementation details and theoretical background for the wet ground augmentation. As explained in the main manuscript, LiDAR measurements are affected by wet water films on road surfaces causing a reduction of road reflectance. If there is a water film present in the scene, it causes the light rays to be increasingly specularly reflected away from the measuring LiDAR sensor [2]. Those reflection properties are modeled by the Fresnel equations, which can be described as follows,

$$t^{\perp} = \frac{2 n_{\text{in}} \cos(\alpha_{\text{in}})}{n_{\text{in}} \cos(\alpha_{\text{in}}) + n_{\text{out}} \cos(\alpha_{\text{out}})}, \tag{3}$$

$$t^{=} = \frac{2 n_{\text{in}} \cos(\alpha_{\text{in}})}{n_{\text{out}} \cos(\alpha_{\text{in}}) + n_{\text{in}} \cos(\alpha_{\text{out}})}, \tag{4}$$

r_s [mm/h]	v_s [m/s]	r_s/v_s	r_r [mm/h] $\approx$
0.5	2.0	0.250	2
0.5	1.2	$0.41\overline{6}$	5
1.0	1.6	0.625	9
2.0	2.0	1.000	18
2.5	1.6	2.000	35
1.5	0.6	2.500	70
1.5	0.4	3.750	130
1.0	0.2	5.000	200

Table 1. Parameters used for our snowfall simulation.

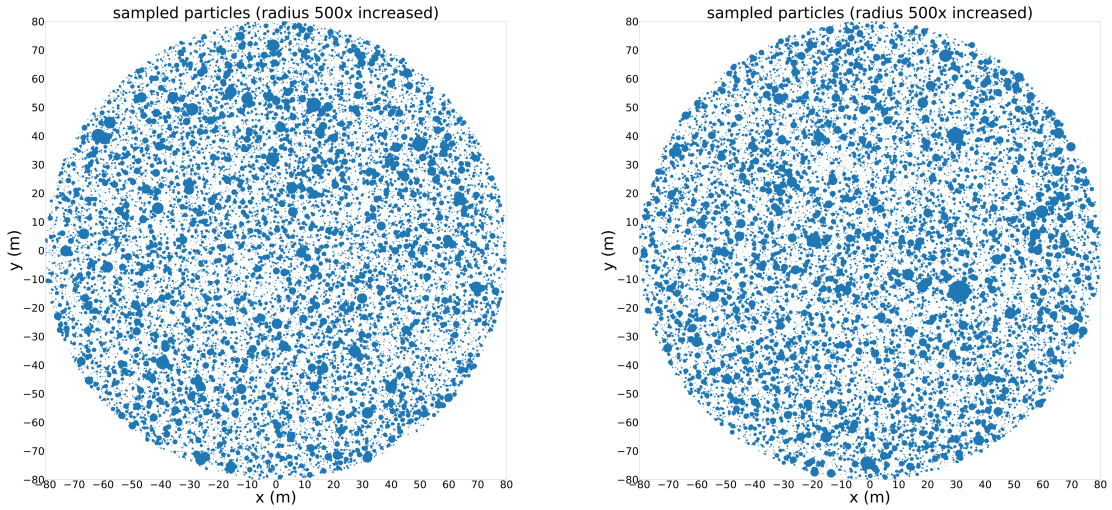


Figure 1. Two example snowflake distribution patterns for $r_s = 1$mm/h and $v_s = 1.6$m/s (third row in Tab. 1).

for the transmission t and for the reflection r,

$$r^{\perp} = \frac{n_{\text{in}} \cos \alpha_{\text{in}} - n_{\text{out}} \cos \alpha_{\text{out}}}{n_{\text{in}} \cos \alpha_{\text{in}} + n_{\text{out}} \cos \alpha_{\text{out}}}, \tag{5}$$

$$r^{=} = \frac{n_{\text{out}} \cos \alpha_{\text{in}} - n_{\text{in}} \cos \alpha_{\text{out}}}{n_{\text{out}} \cos \alpha_{\text{in}} + n_{\text{in}} \cos \alpha_{\text{out}}}. \tag{6}$$

The Fresnel equations explicitly model the change in amplitude of the sent laser pulse and depend on the polarization written as superscript, with $^{\perp}$ for perpendicular polarization and $^{=}$ for parallel polarization. To model the total transmitted power T_{total} from the ground as shown in Eq. (15), (16), and (17) in Sec. 3.1 in the main manuscript, the transferred power T_k^p and the reflection power R_k^p factors have to be derived as described in Eq. (13) to (14) in the main manuscript. The resulting total transmitted power T_{total}^p and reflectivity R_{total}^p with $\rho_0 = 1$ and a sensor at height $h = 1.55\,\text{m}$ are shown as plots per distance in Figure 2.

To run the wet ground simulation (Algorithm 2 in the main manuscript), the estimation of the ground plane and the incident angles from the clear measurements on the ground are required. Therefore, we first filter points in close vicinity to the ground plane. Those points $\mathbf{p} = [x, y, z]$ are part of the set:

$$P_f := \{\, (x, y, z) \mid\ (-1.86 - 0.01 \cdot x)\,\text{m} < z < -1.55\,\text{m} \ \wedge\ 10\,\text{m} < x < 70\,\text{m} \ \wedge -3\,\text{m} < y < 3\,\text{m} \,\}. \tag{7}$$

Then, a linear plane is fitted using the RANSAC algorithm [5], giving us the normal $\mathbf{w}$ and intercept h of the ground plane. Afterwards, the final points that we estimate to belong to the ground are being augmented by Algorithm 2 in the main manuscript:

$$P_g := \{\, \mathbf{p} \mid\ |\mathbf{p} \cdot \mathbf{w} - h| < \epsilon_g \,\}, \tag{8}$$

Algorithm 1 Snowflake Sampling

procedure SAMPLE_SNOWFLAKES($R_{\text{max}}, r_s, v_s, \rho_s$) ▷ R_{max} in m, r_s and v_s in m/s, ρ_s in g/cm^3
  $A_{\text{2D}} \leftarrow \pi {R_{\text{max}}}^2$ ▷ area of our 2D sampling circle
  $A_{\text{snow}} \leftarrow \frac{r_s}{\rho_s v_s} A_{\text{2D}}$
  $r_r \leftarrow \left(\frac{r_s}{487 \rho_s D_0 v_s}\right)^{2/3}$ ▷ see Eq. 2
  $\Lambda \leftarrow 2.55 r_r^{0.48}$ ▷ rate parameter of exponential distribution from Gunn and Marshall [7]
  $\mathbf{s} \leftarrow \varnothing$
  $A_{\text{occ}} \leftarrow 0$
  while $A_{\text{occ}} < A_{\text{snow}}$ **do**
    Sample particle center (x, y) uniformly at random inside a 2D sampling circle centered at $(0, 0)$ and radius R_{max}.
    Sample particle diameter $D \sim \text{Exp}(\Lambda)$.
    Sample offset $\Delta \sim U(-D/2, D/2)$. ▷ offset between **3D** particle center and **2D** sampling circle
    $r \leftarrow \sqrt{(\frac{D}{2})^2 - \Delta^2}$ ▷ cross-section radius between **3D** particle and **2D** sampling circle
    if no $s \in \mathbf{s}$ intersects with (x, y, r) **then**
      $A_{\text{occ}} \leftarrow A_{\text{occ}} + \pi r^2$ ▷ update occupied space
      $\mathbf{s} \leftarrow \mathbf{s} \cup \{(x, y, r)\}$ ▷ add new snowflake
    end if
  end while
  return $\mathbf{s}$
end procedure

Algorithm 2 Compute Beam Occlusions

procedure GET_PARTICLES_IN_BEAM($\mathbf{s}, x, y, R_0, \Theta$)
  $\mathbf{t} \leftarrow \varnothing$
  $\alpha \leftarrow \arctan(y, x)$
  $\alpha_1, \alpha_2 \leftarrow \alpha \pm \frac{\Theta}{2}$ ▷ angles of beam boundaries
  for x_s, y_s, r in $\mathbf{s}$ **do**
    $\beta \leftarrow \arctan(y_s, x_s)$ ▷ angle of particle center
    $\gamma_1, \gamma_2 \leftarrow$ COMPUTE_TANGENT_ANGLES(x_s, y_s, r) ▷ angles of tangent rays from the origin to the particle
    particle_in_beam $\leftarrow \alpha_1 \leq \beta \leq \alpha_2$ ▷ standard case
    particle_at_boundary $\leftarrow$ CHECK_BOUNDARY_OVERLAP($\alpha_1, \alpha_2, x_s, y_s, r$) ▷ seldom case
    if *particle_in_beam* **or** *particle_at_boundary* **then**
      if *particle_at_boundary* **then** ▷ correct tangent angles if they exceed beam boundary angles
        $\gamma_1 \leftarrow \max(\alpha_1, \gamma_1)$
        $\gamma_2 \leftarrow \min(\alpha_2, \gamma_2)$
      end if
      $R \leftarrow \|(x_s, y_s)\|_2$
      $\theta \leftarrow |\gamma_1 - \gamma_2|$
      $\mathbf{t} \leftarrow \mathbf{t} \cup \{(R, \theta)\}$
    end if
  end for
  $\mathbf{t} \leftarrow$ SORT(R) ▷ sort by ascending distance to sensor
  return $\mathbf{t}$
end procedure

with ϵ_g being the maximum distance offset from the ground plane.

For those points, the necessary incident angle α_{in} can be calculated as follows,

$$\alpha_{\text{in}}(\mathbf{p}) = \arccos\left(\frac{\mathbf{pw}}{||\mathbf{p}||_2 ||\mathbf{w}||_2}\right). \tag{9}$$

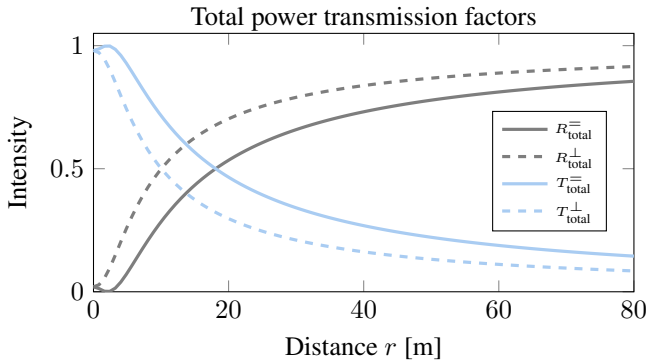


Figure 2. Total power transmission T^p_{total} and reflectivity R^p_{total} factors for road surfaces with $\rho_0 = 1$ and a sensor at height $h = 1.55\,\text{m}$. $p \in (\perp, =)$ refers to perpendicular and parallel polarization, respectively.

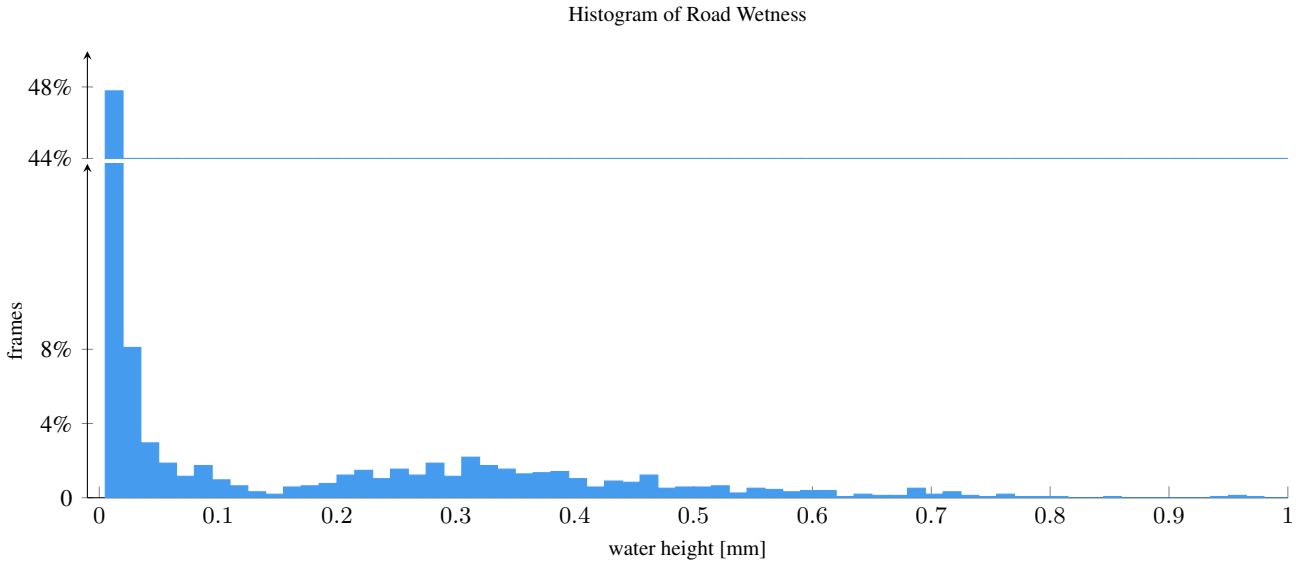


Figure 3. Road wetness measured by the proprietary road friction sensor utilized in STF [1].

4. Additional Dataset Details

In the course of our work we use the STF dataset [1]. In this section, we provide additional details of the dataset. Firstly, in Section 4.1, we show the distribution of road wetness heights motivating our wet ground simulation. Secondly, in Section 4.2, additional details about the creation of the light and heavy snowfall splits are provided. Finally, in Section 4.3, we present statistics regarding the ground truth annotations that are available in each split.

4.1. Ground Wetness Distribution

As pointed out in Sec. 3.1 in the main manuscript, we argue that a significant amount of the LiDAR scans in the STF [1] dataset is affected by wet road surfaces. To analyze how often that is the case, we can take a look at the road wetness readings provided by a proprietary road friction sensor. This sensor reports water height on the ground (see Fig. 3). We find that for more than 35 % of all the frames available in the STF [1] dataset, the road friction sensor reports a water height of at least 0.05 mm, which is enough to alter the intensity of the LiDAR points on the ground significantly (see Fig. 5 in the main manuscript).

4.2. Snowfall Splits

Qualitatively looking through the samples of the snowy test set with 4468 frames provided by the authors of the STF dataset [1], we found that many of those frames actually contain no or only very little snowfall clutter in the LiDAR scans.

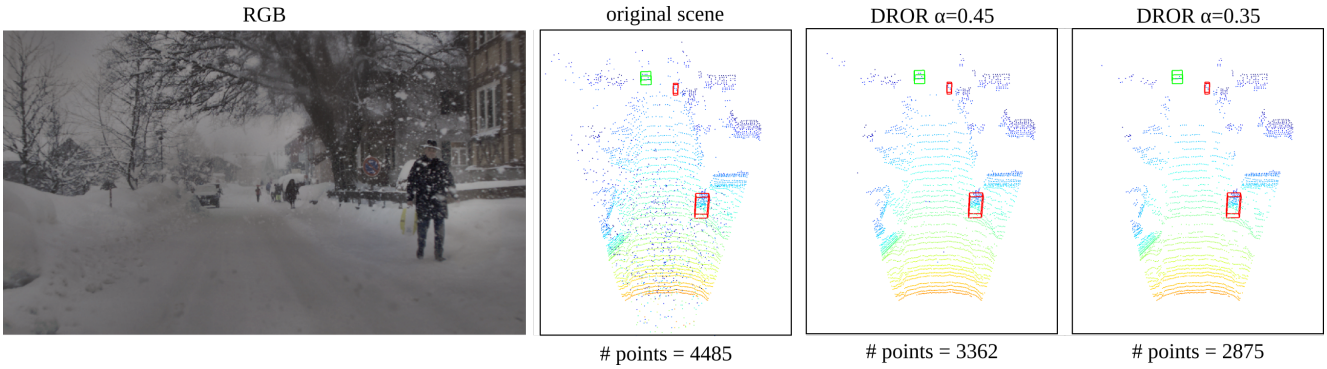


Figure 4. The figure shows the camera image of a scene from the heavy snowfall test set (first column) and the original point cloud including our proposed box (in blue) for classifying the snowy test set in two subsets (second column). The third column shows the original DROR [3] parametrization proposed by its authors and the fourth column shows our adjusted parametrization. Best viewed on a screen and zoomed in.

Hence, we were looking for a way to clean the snowy test set and filter out clear frames from this "snowy" test set.

For this task, we utilize the DROR [3] filter, which does a pretty good job at filtering out snowfall clutter from snowy scenes. However, we found that the proposed parametrization of the DROR [3] filter is too aggressive for the main LiDAR sensor, the Velodyne HDL-S3D, used to record the point clouds in STF [1]. When we set α to the horizontal angular resolution (which is 0.35° for the Velodyne HDL-S3D sensor [17]), as proposed by the DROR [3] authors, the filter removes noticeably more valid points compared to when we set $\alpha = 0.45$ (effectively softening the filter). This parametrization generally removes just as much snowfall clutter but keeps more valid points remaining in the scene (see Fig. 4).

Inspired by the authors of the Canadian Adverse Driving Conditions (CADC) dataset [11], we construct a $10\times2\times2$m box 3m in front of the ego vehicle and 1m below the LiDAR sensor position (see grey boxes in the top row of Fig. 5). In this box, we count the number of points that the DROR [3] filter removes. If DROR [3] does not remove any points from within this box, we treat this sample as "clear" and exclude it from the originally provided "snowy" test set. In fact, since the DROR [3] filter with our enhanced parametrization is still not perfect and *e.g.* still removes valid foreground points from objects within this box, we decided to not only exclude the frames where the DROR [3] filter removes no point, but also exclude all the frames from the "snowy" test set, where DROR [3] filters out up to nine points. This way we make sure that less clear samples make it to the corrected snowy test set. In the end, we exclude 552 frames from the initial "snowy" test set and are left with 3916 frames which contain at least ten points in our proposed box in front of the vehicle classified as clutter by the DROR [3] filter.

Then, since we already employ the DROR [3] filter and think the number of points DROR [3] classifies as clutter is a good proxy for the amount of snowfall occurring in the scene, we decide to construct two subsets out of these remaining 3916 real snowy frames. 2512 frames, where DROR [3] classifies 10-79 points from within the aforementioned box as clutter, construct the **light snowfall** test set and the remaining 1404 frames where DROR [3] classifies at least 80 (and in the most extreme scene up to 713) points as snowfall clutter construct the **heavy snowfall** test set. Fig. 5 shows an example with 10, 80, and the aforementioned, most extreme scene with 713 points counted by DROR [3] as clutter within our box, respectively.

Note that we first tried the box proposed in CADC [11] to count the number of points that DROR [3] classifies as clutter, but found that our proposed box gave a much better estimate on the actual snowfall clutter present in the scene.

4.2.1 Distribution of Snowfall Intensity

The distribution of snowfall intensity of the light snowfall and heavy snowfall frames are exponentially distributed and reported in Fig. 6 as log-scale plot. As extreme snowfall cases are rare, this dataset bias explains the size of the margins between the evaluations on the individual (snowy) test sets. Nevertheless, comparing the best performing method PV-RCNN [13] in Table 2, the "clear-only" baseline drops 7.29 in AP on the car class from clear weather to heavy snowfall (72.34 vs. 65.05 AP) in the distance from 0-30 m, which is the region most corrupted by snowfall. Our simulation is able to recover 45% of that loss, increasing performance by 3.34 AP, while only sacrificing 0.46 AP in *clear weather* (72.34 vs. 71.88 AP) in the aforementioned distance interval. On the pedestrian class, our simulation even recovers more than 50% without any sacrifice.

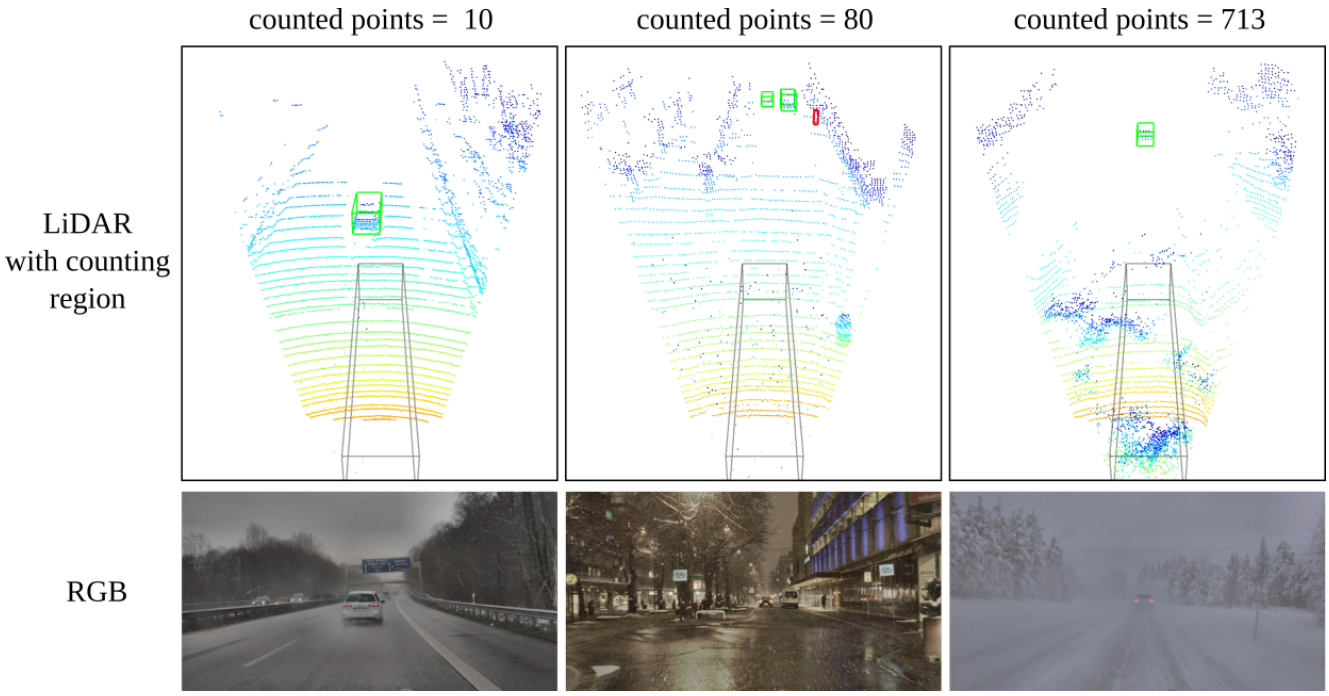


Figure 5. Example scenes with the thresholds used for creating the light and heavy snowfall test splits.

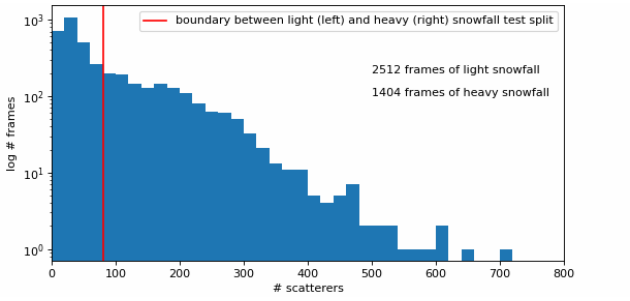


Figure 6. Histogram of scatterers per frame (with bin size 20).

4.3. Object Statistics

In this section we provide more context for the 3D object detection results presented in the main manuscript by providing statistics for the main class(es) of the STF [1] dataset. Fig. 7 shows the available 3D bounding box annotations per split. We can see that in contrast to the number of cars and pedestrians, there are not enough cyclists available in heavy and light snowfall to draw any valid conclusions on this class, as they are unfortunately too underrepresented in such harsh conditions (two orders of magnitude lower than in clear conditions and also two orders of magnitude lower than the car and pedestrian class in the same snowfall test set). Admittedly, the number of cars and pedestrians is also one order of magnitude lower than the annotations available in clear weather, but we are convinced that the absolute numbers of cars and pedestrians available in heavy and light snowfall are well sufficient and adequate to draw meaningful conclusions, especially for the car class.

Fig. 8 shows annotation statistics of cars in the heavy snowfall, the light snowfall and the clear test set (from left to right). Whereas the distributions for yaw-rotations and dimensions look very much the same for all three splits, we can see that in heavy snowfall, the annotations available in the distance interval 50-80m is relatively sparser compared to the other two test sets. This can be explained by less LiDAR points available at such distance in such harsh conditions. That is to say that every 3D bounding box at least requires one LiDAR point in order to be valid. Thus, if there are no points belonging to objects, there are also no 3D bounding box annotations. We can also see that the cars present in the clear test set tend to be a little closer to the ego vehicle / sensor and therefore, typically easier to detect and predict.

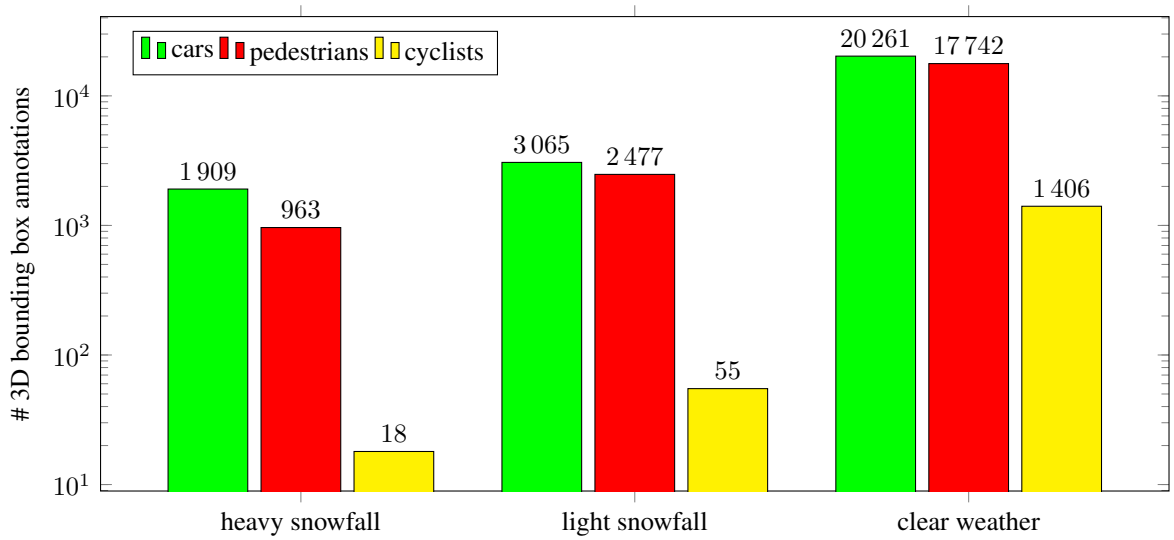


Figure 7. Number of available 3D bounding box annotations per class and test split. The heavy snowfall test split contains 1404 frames, the light snowfall test split contains 2512 frames and the clear weather test set contains 1816 frames.

5. Additional Quantitative Results

In Tab. 2, we present the quantitative results of the best performing method PV-RCNN [13] on the pedestrian class. For completeness, the results on the car class from the main manuscript are prepended. To mitigate potential statistical fluctuations, again, we report for each experiment the average performance over three independent training runs. To avoid confusion, PV-RCNN [13] is a multi-class method and all three runs predict both classes, cars and pedestrians, at the same time. So there is no need to train the method for every class separately. We can see that similar to the car class, our snowfall augmentation also outperforms all competing methods and the clear-weather baseline on the heavy and light snowfall test sets while gaining performance on the clear weather test set, too. The overall performance on the pedestrian class however still is noticeably lower than the performance on the car class (*e.g.* 41.61 vs. 18.24 for the training runs using our snowfall simulation). This indicates that vulnerable road users (VRUs) continue to deserve more attention by our research community.

In Tab. 3, we report the evaluation results using the original KITTI [6] evaluation framework (including the enhancement proposed in [16] to use 40 instead of 11 recall positions). We can see that with the exception of easy cars on the clear weather test set, either our snowfall simulation or our combined snowfall and wet ground simulation always achieves state-of-the-art results on all KITTI difficulty levels, supporting our findings reported on moderate cars for various distance intervals in the main manuscript. The distance-based evaluation of the remaining difficulty levels easy and hard is shown in Tab. 4. For completeness, the moderate difficulty reported in the main manuscript is included here as well. We can observe that our proposed simulation method shows superior performance on all difficulty levels and all individual distance intervals in heavy snowfall.

On a general note, we can see that while the performance difference between the easy and the moderate difficulty is relatively small, the difference in performance between the moderate and hard difficulty is larger. Note that according to the definition of the difficulty levels in [6], the hard difficulty level does also include all moderate and easy samples, and the moderate difficulty level does also include all easy samples. This means that the samples added going from moderate to hard seem to be significantly more difficult to predict correctly, otherwise there would not be such a noticeable gap between the moderate and hard results.

Finally, Tab. 5 shows an ablation study using different parametrizations of the DROR [3] filter. We can see that our adjusted parameterization of DROR [3] ($\alpha = 0.45$) almost always outperforms the parameterization suggested by the authors of [3] ($\alpha = 0.35$) for all 3D object detectors utilized in the main manuscript.

6. Additional Qualitative Results

In Fig. 9, 10 and 11, we present additional qualitative detection results of our full simulation method used for training the PV-RCNN [13] object detector, compared to using competing simulation and pre-processing methods for the same purpose.

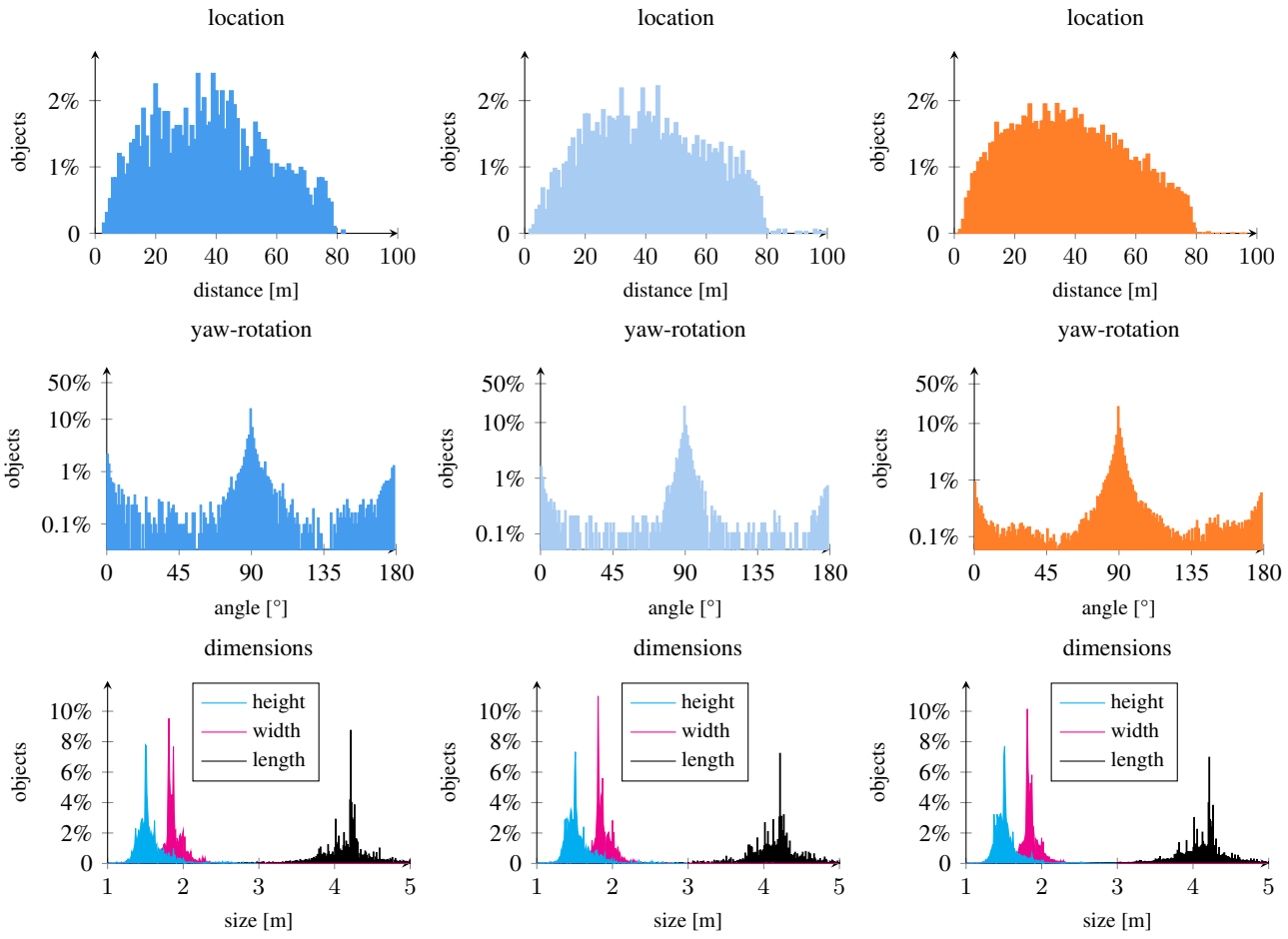


Figure 8. 3D bounding box statistics of the **car** class for the heavy snowfall test split with 1404 samples (left column), the light snowfall test split with 2512 samples (middle column) and the clear weather test split with 1816 samples (right column). The top row shows the distance to the object, middle row the birds-eye-view (BEV) yaw-rotation and the bottom row the objects dimensions.

Detection method	**Simulation method**	**heavy snowfall ↑**				**light snowfall ↑**				**clear weather ↑**			
		0-80m	0-30m	30-50m	50-80m	**0-80m**	0-30m	30-50m	50-80m	**0-80m**	0-30m	30-50m	50-80m
	None	39.69	65.05	36.14	8.03	41.13	69.24	39.72	**11.68**	45.36	**72.34**	42.48	10.53
	Fog [8]	38.19	64.72	33.38	7.49	39.82	68.41	38.68	9.65	43.37	71.05	40.03	9.90
	DROR [3]	38.57	64.27	35.40	8.07	39.33	66.73	38.14	10.51	41.44	67.76	38.48	9.44
PV-RCNN [13]	LISA [9]	39.21	64.21	35.34	8.64	41.60	69.15	**41.08**	11.15	45.30	71.06	42.86	11.45
cars	Ours-wet	40.03	65.34	35.82	**9.31**	41.07	68.49	40.03	11.02	44.81	71.60	42.71	10.63
	Ours-snow	41.61	67.44	**37.47**	8.84	41.20	68.79	40.20	11.13	45.61	72.14	**43.40**	11.21
	Ours-snow+wet	**41.79**	**68.39**	37.14	8.85	**41.79**	**70.30**	41.01	11.28	**45.71**	71.88	43.31	**11.69**
	None	16.48	31.20	12.69	1.71	19.25	38.86	11.90	2.19	18.95	36.45	12.15	2.10
	Fog [8]	16.79	31.03	12.91	1.51	18.13	36.83	10.58	1.63	17.10	33.31	11.05	1.48
	DROR [3]	16.83	32.29	11.94	1.38	18.75	38.25	11.06	1.93	18.33	35.02	12.07	1.77
PV-RCNN [13]	LISA [9]	17.43	33.08	13.37	**1.83**	19.40	38.46	11.93	**2.31**	18.21	34.90	12.06	1.92
pedestrians	Ours-wet	17.35	33.24	12.62	1.51	19.15	38.59	**12.00**	2.21	18.98	35.66	12.87	2.08
	Ours-snow	**18.24**	**34.06**	**13.46**	1.66	19.62	39.42	11.91	2.21	**19.23**	**36.78**	**12.84**	**2.24**
	Ours-snow+wet	17.50	33.22	13.03	1.32	**19.73**	**39.48**	11.84	2.24	18.91	35.70	12.69	1.92

Table 2. 3D average precision (AP) of moderate **cars** and moderate **pedestrians** on three STF splits: the heavy snowfall test split with 1404 samples, the light snowfall test split with 2512 samples and the clear weather test split with 1816 samples. "Ours-wet": our wet ground simulation, "Ours-snow": our snowfall simulation, "Ours-snow+wet": cascaded application of our snowfall and wet ground simulation.

Detection method	Simulation method	heavy snowfall ↑			light snowfall ↑			clear weather ↑		
		easy	moderate	hard	easy	moderate	hard	easy	moderate	hard
	None	40.36	39.69	35.29	42.90	41.13	37.57	**47.79**	45.36	41.45
	Fog [8]	38.70	38.19	33.74	41.85	39.82	36.32	46.06	43.37	39.22
	DROR [3]	39.74	38.57	33.72	41.17	39.33	35.57	44.23	41.44	37.46
PV-RCNN [13]	LISA [9]	39.40	39.21	34.87	43.07	41.60	37.99	47.07	45.30	41.34
cars	Ours-wet	40.03	40.03	35.63	42.75	41.07	37.46	46.97	44.81	40.64
	Ours-snow	42.13	41.61	36.79	42.84	41.20	37.59	47.49	45.61	41.74
	Ours-snow+wet	**42.42**	**41.79**	**36.94**	**43.31**	**41.79**	**38.29**	47.59	**45.71**	**41.80**
	None	17.92	16.79	16.22	19.91	19.25	18.59	19.59	18.95	17.75
	Fog [8]	17.68	16.48	16.00	18.59	18.13	17.18	17.92	17.10	16.02
	DROR [3]	17.81	16.83	16.08	19.40	18.75	17.79	19.11	18.33	17.20
PV-RCNN [13]	LISA [9]	18.65	17.43	16.87	19.88	19.40	18.52	18.85	18.21	17.16
pedestrians	Ours-wet	18.57	17.35	16.67	19.79	19.15	18.35	19.83	18.98	17.76
	Ours-snow	**19.30**	**18.24**	**17.58**	20.25	19.62	**18.94**	**19.97**	**19.23**	**18.02**
	Ours-snow+wet	18.47	17.50	17.05	**20.26**	**19.73**	18.91	19.70	18.91	17.64

Table 3. 3D average precision (AP) over the entire 0-80m range for easy, moderate and hard **cars** and **pedestrians** on three STF splits: the heavy snowfall test split with 1404 samples, the light snowfall test split with 2512 samples and the clear weather test split with 1816 samples. "Ours-wet": our wet ground simulation, "Ours-snow": our snowfall simulation, "Ours-snow+wet": cascaded application of our snowfall and wet ground simulation.

Detection method	Simulation method	heavy snowfall ↑				light snowfall ↑				clear weather ↑			
		0-80m	0-30m	30-50m	50-80m	**0-80m**	0-30m	30-50m	50-80m	**0-80m**	0-30m	30-50m	50-80m
	None	40.36	65.13	36.15	8.80	42.90	69.43	41.97	**12.70**	**47.79**	**73.94**	44.51	12.36
	Fog [8]	38.70	63.68	33.24	8.65	41.85	69.69	41.23	10.32	46.06	72.27	41.85	11.56
	DROR [3]	39.74	63.00	36.88	9.66	41.17	67.13	40.16	11.66	44.23	68.92	40.83	11.46
PV-RCNN [13]	LISA [9]	39.40	63.83	34.96	9.25	43.07	70.36	42.85	12.08	47.07	72.51	44.36	13.15
easy	Ours-wet	40.03	64.41	35.63	**9.87**	42.75	69.29	42.27	11.91	46.97	72.58	44.07	12.36
	Ours-snow	42.13	66.47	38.04	9.66	42.84	69.43	42.42	12.00	47.49	72.65	**44.82**	13.09
	Ours-snow+wet	**42.42**	**67.39**	**38.22**	9.83	**43.31**	**70.52**	**42.87**	12.06	47.59	73.24	44.62	**13.41**
	None	39.69	65.05	36.14	8.03	41.13	69.24	39.72	**11.68**	45.36	**72.34**	42.48	10.53
	Fog [8]	38.19	64.72	33.38	7.49	39.82	68.41	38.68	9.65	43.37	71.05	40.03	9.90
	DROR [3]	38.57	64.27	35.40	8.07	39.33	66.73	38.14	10.51	41.44	67.76	38.48	9.44
PV-RCNN [13]	LISA [9]	39.21	64.21	35.34	8.64	41.60	69.15	**41.08**	11.15	45.30	71.06	42.86	11.45
moderate	Ours-wet	40.03	65.34	35.82	**9.31**	41.07	68.49	40.03	11.02	44.81	71.60	42.71	10.63
	Ours-snow	41.61	67.44	**37.47**	8.84	41.20	68.79	40.20	11.13	45.61	72.14	**43.40**	11.21
	Ours-snow+wet	**41.79**	**68.39**	37.14	8.85	**41.79**	**70.30**	41.01	11.28	**45.71**	71.88	43.31	**11.69**
	None	35.29	62.75	30.75	6.22	37.57	66.85	35.07	**9.45**	41.45	**69.89**	37.36	9.21
	Fog [8]	33.74	61.76	27.71	6.12	36.32	65.08	34.05	7.78	39.22	68.44	34.94	8.05
	DROR [3]	33.72	61.78	29.09	6.05	35.57	64.60	32.86	8.41	37.46	65.88	33.03	7.47
PV-RCNN [13]	LISA [9]	34.87	62.40	30.59	6.74	37.99	66.60	**36.36**	9.10	41.34	68.78	38.41	9.64
hard	Ours-wet	35.63	62.62	30.99	**7.25**	37.46	65.87	35.96	9.10	40.64	69.21	37.65	8.94
	Ours-snow	36.79	64.92	**32.33**	7.00	37.59	66.41	35.36	9.00	41.74	69.87	**38.63**	9.44
	Ours-snow+wet	**36.94**	**66.02**	32.16	7.01	**38.29**	**67.90**	36.13	9.17	**41.80**	69.58	38.51	**9.88**

Table 4. Comparison of simulation methods for 3D object detection in snowfall on STF [1]. We report 3D average precision (AP) of **easy**, **moderate** and **hard** cars on three STF splits: the heavy snowfall test split with 1404 samples, the light snowfall test split with 2512 samples and the clear weather test split with 1816 samples. "Ours-wet": our wet ground simulation, "Ours-snow": our snowfall simulation, "Ours-snow+wet": cascaded application of our snowfall and wet ground simulation.

In Fig. 9, we observe that our method consistently detects distant cars which are cluttered by snowfall, demonstrating a high recall, without introducing false positives. Moreover, our method also detects salient pedestrians in the second and third scene, whereas competing methods and the clear-weather baseline do not detect these pedestrians in general.

In Fig. 10, only our method detects the car in front of the ego-vehicle in the first scene, which is affected by severe spray. LISA [9] misses the pedestrian in the third scene, while DROR [3] outputs a false positive detection in the fourth scene and predicts an incorrect orientation for the car in that scene. On the contrary, our method correctly handles both of these scenes.

In Fig. 11, the clear weather baseline and LISA [9] produce false positives in the first scene which are mostly caused by clutter due to intense snowfall, whereas our method eliminates all false positives while predicting the two cars in the scene correctly at the same time. Our method also handles the second scene correctly, as it accurately detects the pedestrian on the left side of the road without any false positives.

Detection method	DROR [3] α	heavy snowfall ↑				light snowfall ↑				clear weather ↑			
		0-80m	0-30m	30-50m	50-80m	**0-80m**	0-30m	30-50m	50-80m	**0-80m**	0-30m	30-50m	50-80m
PV-RCNN [13]	0.35	37.77	64.02	33.94	**8.26**	37.63	65.83	34.86	9.81	40.28	64.99	37.77	9.36
	0.45	**38.57**	**64.27**	**35.40**	8.07	**39.33**	**66.73**	**38.14**	**10.51**	**41.44**	**67.76**	**38.48**	**9.44**
VoxelRCNN-Car [4] (single class method)	0.35	37.27	62.70	**33.63**	**7.55**	36.96	64.23	34.78	9.88	38.97	65.46	35.08	**8.95**
	0.45	**38.16**	**64.97**	33.23	6.83	**38.48**	**66.93**	**35.68**	**9.97**	**40.65**	**67.94**	**36.85**	8.45
CenterPoint [19]	0.35	36.18	62.21	31.80	6.48	36.84	65.37	33.96	9.37	38.73	65.32	34.50	8.03
	0.45	**38.42**	**64.47**	**34.31**	**8.27**	**38.69**	**65.62**	**37.59**	**10.26**	**40.80**	**67.62**	**36.61**	**8.69**
Part-A² [15]	0.35	33.70	61.10	25.96	5.81	33.11	59.18	31.14	7.32	36.85	63.46	32.07	8.32
	0.45	**35.85**	**65.36**	**27.99**	**6.13**	**35.43**	**63.50**	**32.87**	**7.95**	**39.48**	**66.92**	**35.18**	**8.61**
PointRCNN [14]	0.35	34.37	60.05	30.10	**5.78**	34.73	62.14	32.98	**8.05**	37.88	64.01	34.96	7.13
	0.45	**36.14**	**62.64**	**31.64**	5.28	**36.31**	**63.52**	**36.62**	7.77	**39.08**	**64.96**	**36.54**	**7.70**
SECOND [18]	0.35	33.80	60.12	27.76	5.99	33.44	60.05	30.02	8.52	36.73	61.86	32.98	8.72
	0.45	**35.04**	**60.72**	**28.79**	**7.88**	**35.09**	**62.24**	**32.09**	**8.85**	**38.96**	**64.74**	**35.50**	**9.76**
PointPillars [10]	0.35	**29.42**	53.70	**22.87**	**5.11**	30.06	56.56	26.30	6.66	32.75	58.21	27.54	**6.79**
	0.45	29.32	**54.52**	21.88	4.82	**30.99**	**57.17**	**28.43**	**6.95**	**34.72**	**60.59**	**30.34**	6.72

Table 5. Ablation study on the parametrization of the DROR [3] filter. We report 3D average precision (AP) of moderate cars on three STF splits: the heavy snowfall test split with 1404 samples, the light snowfall test split with 2512 samples and the clear weather test split with 1816 samples. "Ours-wet": our wet ground simulation, "Ours-snow": our snowfall simulation, "Ours-snow+wet": cascaded application of our snowfall and wet ground simulation.

The third example shows a case where our method is the only one correctly predicting the cyclist in the scene. Cyclists are, as mentioned in Sec. 4.3, compared to cars and pedestrians, fairly rare in both snowy test sets. In this example, we can also see a failure case where the second parked car on the left-hand side of the road is not detected by any method, including ours. Finally, for the fourth scene, only DROR [3] and our method detects the pedestrian on the right side of the road, despite the intense noise in the point cloud.

We could not find any specific "failure mode" of the 3D Object Detection methods trained on our simulation pipeline. The 3D Object Detection methods trained on our simulation pipeline so to speak make the "same" mistakes as the ones trained on competing approaches, just less of them, especially so in the 0-30m range, which is the region most corrupted by snowfall noise. One example showcasing that the 3D Object Detection methods trained on our simulation pipeline make the "same" mistakes is given in the third row of Fig. 11.

7. Discussion of Limitations

In this section, we discuss the limitations of our work. First of all, we do not foresee any potential negative societal impact by our work. However, we want to take the chance to address the limitations we are aware of.

A minor limitation of our current snowfall simulation is its runtime. Depending on the number of points per scene, our simulation can take up to multiple seconds per point cloud. This limitation could be overcome by *e.g.* enhancing internal data structures or entirely switching from an implementation in Python to an implementation in C. To allow reproducibility of our results and to allow the community to make and suggest adjustments, we plan to open-source our code.

Another limitation could be the instantiation choice of our simulation parameters (snowfall rate r_s and water height d_w). They highly depend on the prevailing weather in the test sets, which cannot be easily deduced from the raw point clouds themselves. A different instantiation of these hyperparameters might achieve better results. We chose the parameters according to their commonly used ranges and did not tune them explicitly on the underlying dataset.

The main goal of our simulation pipeline is to convert LiDAR point clouds captured in clear weather into point clouds as if they were captured under real snowfall. Our qualitative and quantitative results indicate that we are indeed getting closer to the actual appearance of LiDAR point clouds captured in real snowfall. However, we are not (yet) able to completely mimic/re-produce these measurements entirely. As one of our reviewers correctly pointed out, one limiting factor we did not address to this end, is the "wetness of objects". Simulating wet objects is an interesting extension of our model but requires additional semantic understanding, *e.g.* pedestrians with or without umbrellas, but it's definitely a good proposal as another possible direction of future work.

References

[1] Mario Bijelic, Tobias Gruber, Fahim Mannan, Florian Kraus, Werner Ritter, Klaus Dietmayer, and Felix Heide. Seeing through fog without seeing fog: Deep multimodal sensor fusion in unseen adverse weather. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 1, 4, 5, 6, 9, 12, 13, 14

[2] Johan Casselgren, Mikael Sjödahl, and James LeBlanc. Angular spectral response from covered asphalt. *Applied Optics*, 46(20):4277–4288, 2007. 1

[3] Nicholas Charron, Stephen Phillips, and Steven Waslander. De-noising of lidar point clouds corrupted by snowfall. In *Conference on Computer and Robot Vision (CRV)*, 2018. 5, 7, 8, 9, 10, 12, 13, 14

[4] Jiajun Deng, Shaoshuai Shi, Peiwei Li, Wengang Zhou, Yanyong Zhang, and Houqiang Li. Voxel R-CNN: Towards high performance voxel-based 3d object detection. In *AAAI Conference on Artificial Intelligence*, 2021. 10

[5] Martin Fischler and Robert Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24:381–395, 1981. 2

[6] Andreas Geiger, Philip Lenz, and Raquel Urtasun. Are we ready for autonomous driving? the KITTI vision benchmark suite. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012. 7

[7] Kevin Gunn and John Marshall. The distribution with size of aggregate snowflakes. *Journal of Atmospheric Sciences*, 15(5):452 – 461, 1958. 3

[8] Martin Hahner, Christos Sakaridis, Dengxin Dai, and Luc Van Gool. Fog simulation on real lidar point clouds for 3d object detection in adverse weather. In *IEEE International Conference on Computer Vision (ICCV)*, 2021. 8, 9

[9] Velat Kilic, Deepti Hegde, Vishwanath Sindagi, Brinton Cooper, Mark Foster, and Vishal Patel. Lidar light scattering augmentation (LISA): physics-based simulation of adverse weather conditions for 3d object detection. *arXiv preprint 2107.07004*, 2021. 1, 8, 9, 12, 13, 14

[10] Alex Lang, Sourabh Vora, Holger Caesar, Lubing Zhou, Jiong Yang, and Oscar Beijbom. PointPillars: Fast encoders for object detection from point clouds. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019. 10

[11] Matthew Pitropov, Danson Evan Garcia, Jason Rebello, Michael Smart, Carlos Wang, Krzysztof Czarnecki, and Steven Waslander. Canadian adverse driving conditions dataset. *The International Journal of Robotics Research*, 40:681 – 690, 2021. 5

[12] Roy Rasmussen, Jothiram Vivekanandan, Jeffrey Cole, Barry Myers, and Charles Masters. The estimation of snowfall rate using visibility. *Journal of Applied Meteorology*, 38(10):1542 – 1563, 1999. 1

[13] Shaoshuai Shi, Chaoxu Guo, Li Jiang, Zhe Wang, Jianping Shi, Xiaogang Wang, and Hongsheng Li. PV-RCNN: Point-voxel feature set abstraction for 3d object detection. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 5, 7, 8, 9, 10, 12, 13, 14

[14] Shaoshuai Shi, Xiaogang Wang, and Hongsheng Li. PointRCNN: 3d object proposal generation and detection from point cloud. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019. 10

[15] Shaoshuai Shi, Zhe Wang, Jianping Shi, Xiaogang Wang, and Hongsheng Li. From points to parts: 3d object detection from point cloud with part-aware and part-aggregation network. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 2020. 10

[16] Andrea Simonelli, Samuel Rota Bulo, Lorenzo Porzi, Manuel Lopez-Antequera, and Peter Kontschieder. Disentangling monocular 3d object detection. In *IEEE International Conference on Computer Vision (ICCV)*, 2019. 7

[17] Velodyne. Hdl-64e user's manual, 2007. 1, 5

[18] Yan Yan, Yuxing Mao, and Bo Li. SECOND: Sparsely embedded convolutional detection. *Sensors*, 18, 2018. 10

[19] Tianwei Yin, Xingyi Zhou, and Philipp Krahenbuhl. Center-based 3d object detection and tracking. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021. 10

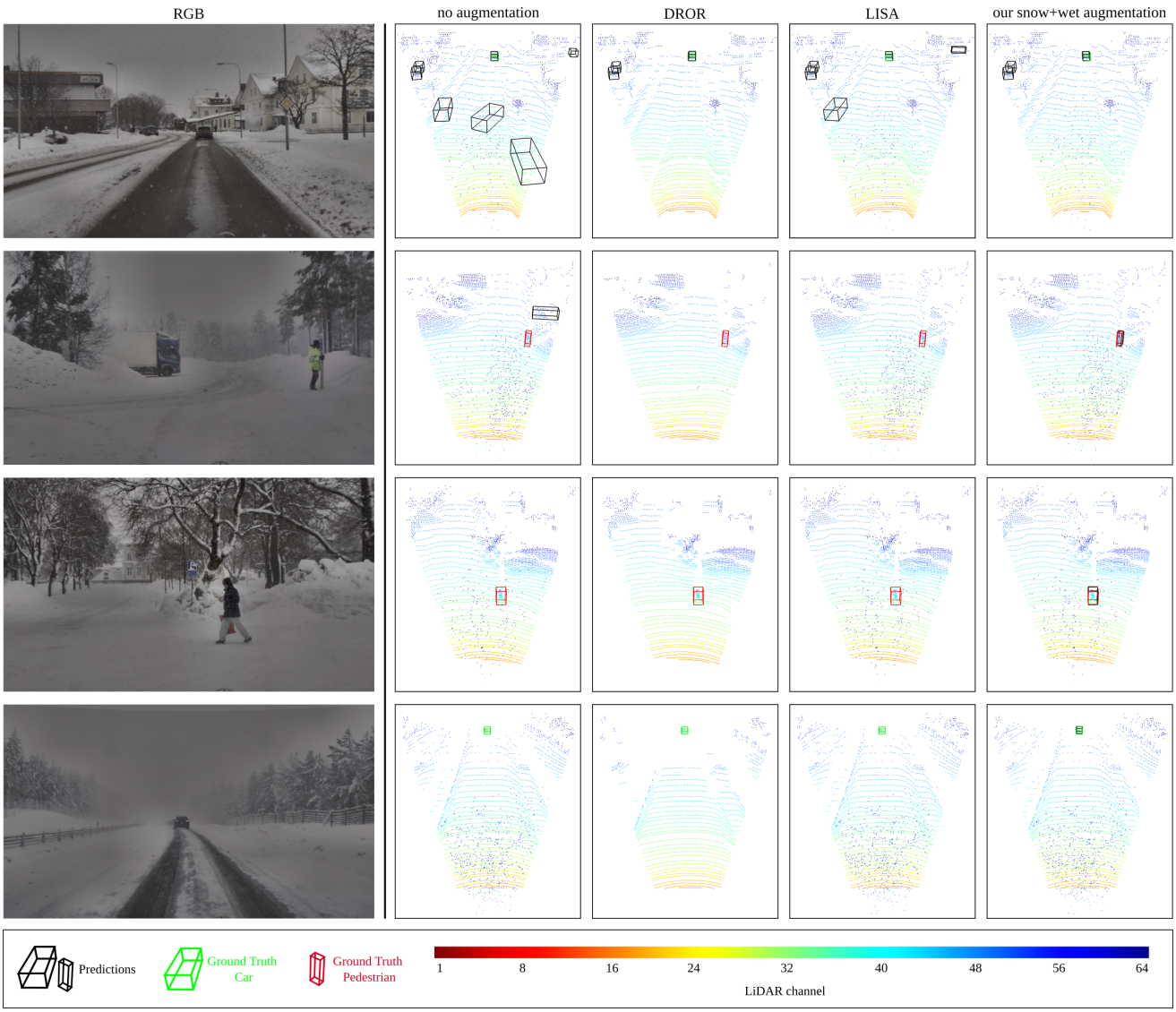


Figure 9. Qualitative comparison of PV-RCNN [13] on samples from STF [1] containing heavy snowfall. The leftmost column shows the corresponding RGB images. The rest of the columns show the LiDAR point clouds with ground-truth boxes and predictions using the clear-weather baseline ("no augmentation"), DROR [3], LISA [9], and our fully-fledged simulation ("our snow+wet augmentation"). Best viewed on a screen and zoomed in.

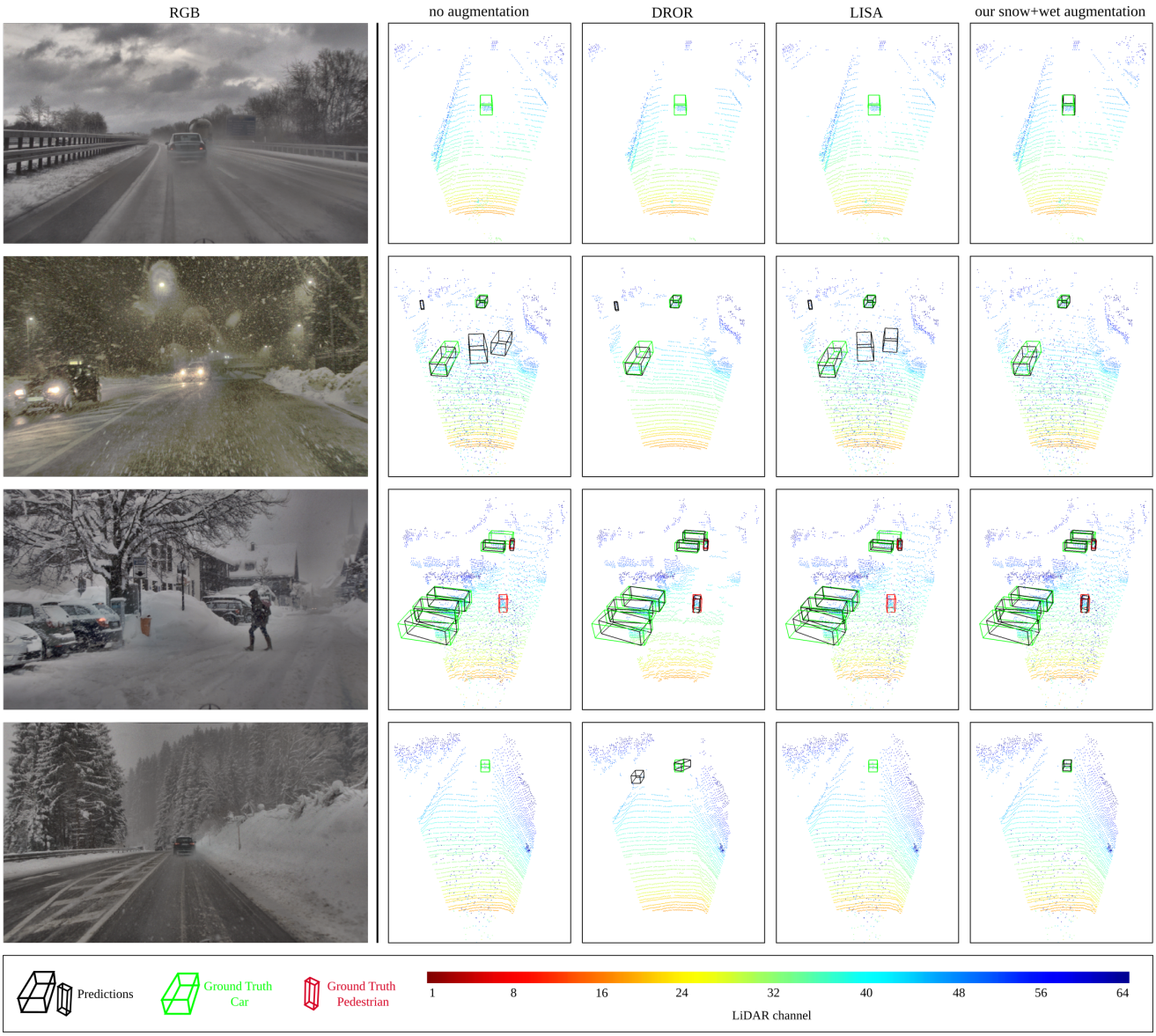


Figure 10. Qualitative comparison of PV-RCNN [13] on samples from STF [1] containing heavy snowfall. The leftmost column shows the corresponding RGB images. The rest of the columns show the LiDAR point clouds with ground-truth boxes and predictions using the clear-weather baseline ("no augmentation"), DROR [3], LISA [9], and our fully-fledged simulation ("our snow+wet augmentation"). Best viewed on a screen and zoomed in.

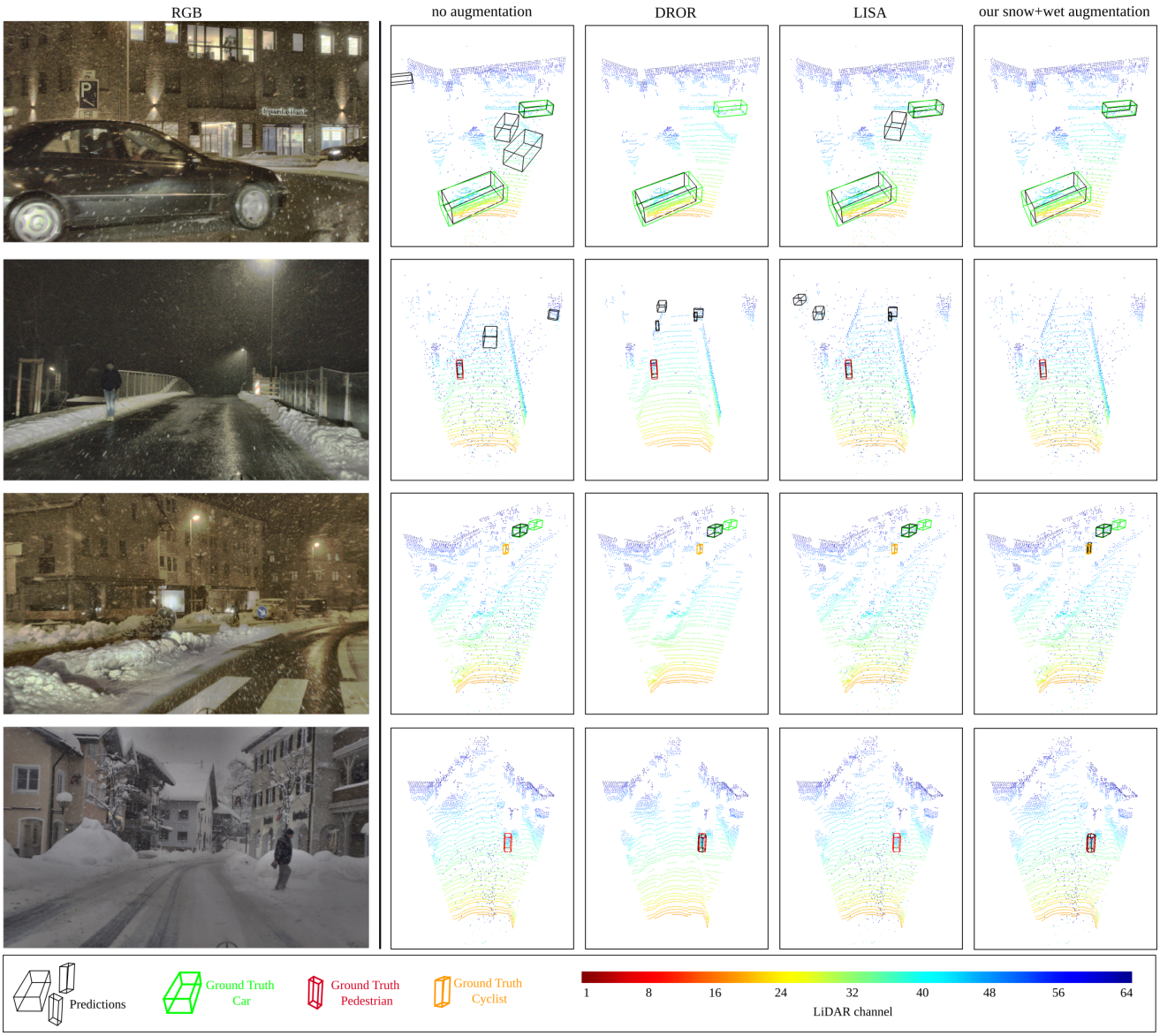


Figure 11. Qualitative comparison of PV-RCNN [13] on samples from STF [1] containing heavy snowfall. The leftmost column shows the corresponding RGB images. The rest of the columns show the LiDAR point clouds with ground-truth boxes and predictions using the clear-weather baseline ("no augmentation"), DROR [3], LISA [9], and our fully-fledged simulation ("our snow+wet augmentation"). Best viewed on a screen and zoomed in.