Supplementary Material for
MAELi: Masked Autoencoder for Large-Scale LiDAR Point Clouds

Georg Krispel[1] David Schinagl[1,2] Christian Fruhwirth-Reisinger[1,2] Horst Possegger[1] Horst Bischof[1,2]
{georg.krispel,david.schinagl,christian.reisinger,possegger,bischof}@icg.tugraz.at
[1] Graz University of Technology [2] Christian Doppler Laboratory for Embedded Machine Learning

This supplementary material presents further details, results and insights into MAELi. We state further results in Section 1, describe the detailed architecture of our decoder in Section 2 and illustrate the motivation behind spherical masking in Section 3. Furthermore, we evaluate the impact of different amounts of masked voxels in Section 4 and discuss potential limitations in Section 5. Finally, we discuss additional insights on reconstruction results and data efficiency in Section 6.

1. Additional Results

We provide results for CenterPoint [8] and PV-RCNN [4] to illustrate that our pre-trained initialization significantly enhances these baseline models. Following insights from the main manuscript (Section 4.1), we observe in Table 1 that our MAELi pre-training effectively improves detection performance in a low-data regime where only a limited number of annotated samples are available for fine-tuning. In Table 2, we report AP scores for Waymo Open Dataset (WOD) [5], extending the results from Table 2 in the main manuscript. Additionally, in Table 3, we present our findings on the KITTI 3D dataset using the R_{11} metric, and make comparisons with ALSO [1] and Occupancy-MAE [3].

2. Sparse Reconstruction Decoder for 3D Object Detection

To describe the architecture of our decoder in detail, we group operations with the same *voxel/tensor stride* into a *block*. In Table 4, we list the different decoder blocks in addition to the preceding bird's-eye view (BEV) encoder (summarized as single entry) and the required *reshaping+sampling* step to transform the dense feature representation back to a sparse 3D tensor.

Each block comprises an upsampling step using *generative transposed convolution* and a pruning step via $1 \times 1 \times 1$ *submanifold sparse convolution*. The operations are listed in Table 5.

3. Spherical Masking - Illustration

As discussed in the main manuscript (Section 3.3), spherical masking reduces the angular resolution in azimuth and inclination by subsampling the LiDAR's range image. Figure 1 illustrates the effect of this sampling on the LiDAR's range image. We sample objects as if they were located at a larger distance. Since nearby objects are more densely sensed by the LiDAR, we have more knowledge about the actual occupancy and thus, can induce a stronger self-supervision signal. This helps to improve the model's ability to generalize to objects located farther away.

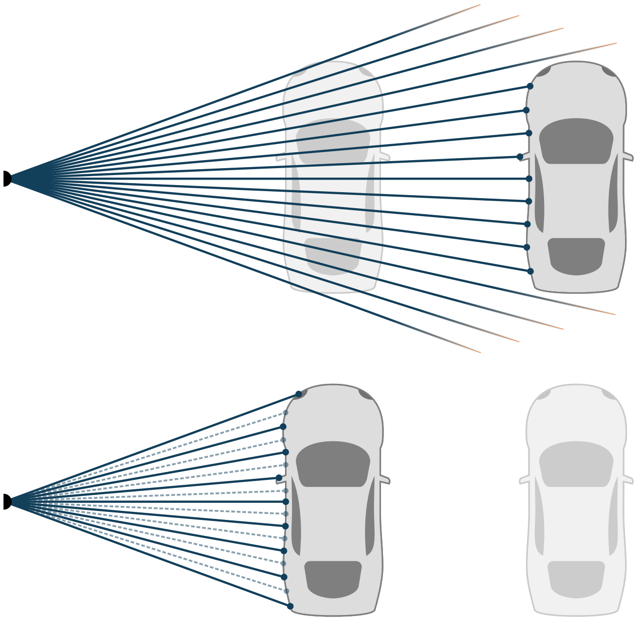

Figure 1. Spherical masking reduces the angular resolution of the LiDAR (bottom). The resulting sampling is thus similar to objects that are farther away (top).

Fraction	Method	3D AP/APH (LEVEL 2)									
		Gain		Overall		Vehicle		Pedestrian		Cyclist	
		AP	APH	AP	APH	AP	APH	AP	APH	AP	APH
1% (791 frames)	Centerpoint [8]	-	-	39.64	36.50	41.01	40.32	40.01	32.63	37.90	36.55
	+ MAELi	+9.29	+8.75	48.93	45.25	49.99	49.24	51.92	43.07	44.89	43.43
	PV-RCNN [4]	-	-	43.93	30.72	51.34	48.70	41.59	20.35	38.86	23.11
	+ MAELi	+7.53	+4.89	51.46	35.61	56.24	55.38	49.41	25.32	48.73	26.14
‘ 5% (3952 frames)	Centerpoint [8]	-	-	53.91	51.16	53.04	52.45	52.73	46.51	55.96	54.53
	+ MAELi	+4.49	+4.21	58.40	55.37	57.62	57.01	59.01	51.83	58.57	57.27
	PV-RCNN [4]	-	-	56.98	38.98	61.66	60.86	53.28	27.15	56.00	28.92
	+ MAELi	+1.64	+1.39	58.62	40.37	62.77	62.04	57.07	29.05	56.02	30.02
10% (7904 frames)	Centerpoint [8]	-	-	58.09	55.41	56.95	56.40	56.97	50.90	60.35	58.94
	+ MAELi	+3.26	+3.07	61.35	58.48	59.93	59.36	62.06	55.30	62.06	60.78
	PV-RCNN [4]	-	-	60.09	41.89	63.73	63.05	57.32	30.09	59.23	32.53
	+ MAELi	+1.19	-0.05	61.28	41.84	64.63	63.99	59.82	30.90	59.40	30.62
20% (15808 frames)	Centerpoint [8]	-	-	61.81	59.15	60.59	60.06	61.19	55.03	63.64	62.36
	+ MAELi	+0.98	+0.90	62.79	60.05	61.79	61.23	63.47	57.04	63.11	61.87
	PV-RCNN [4]	-	-	62.15	42.99	65.01	64.35	60.40	30.30	61.05	34.32
	+ MAELi	+0.49	+7.21	62.65	50.20	65.45	64.85	61.54	35.52	60.95	50.24

Table 1. Quantitative results of our pre-training on Centerpoint and PV-RCNN on the Waymo *val* set. For each detector, we report the results of training from scratch (upper row) and the improved results utilizing a MAELi-pre-trained initialization (lower row), respectively. We use the first 399 sequences of the Waymo *train* set for pre-training and different fractions of the second 399 sequences for fine-tuning.

Method	3D AP/APH (LEVEL 2)									
	Gain		Overall		Vehicle		Pedestrian		Cyclist	
	AP	APH	AP	APH	AP	APH	AP	APH	AP	APH
SECOND [6]	-	-	58.26	54.35	62.58	62.02	57.22	47.49	54.97	53.53
+ Occ-MAE [3]	+0.85	+0.75	59.11	55.10	62.67	62.34	59.03	48.79	55.62	54.17
+ MAELi	+2.32	+2.35	60.57	56.69	63.75	63.20	60.71	50.93	57.26	55.95
CenterPoint [8]	-	-	64.51	61.92	63.16	62.65	64.27	58.23	66.11	64.87
+ Occ-MAE [3]	+1.35	+1.31	65.86	63.23	64.05	63.53	65.78	59.62	67.76	66.53
+ MAELi	+1.09	+1.08	65.60	63.00	64.22	63.70	65.93	59.79	66.66	65.52
PV-RCNN [4]	-	-	59.84	56.23	64.99	64.38	53.80	45.14	60.72	59.18
+ GCC-3D [2]	+1.46	+1.95	61.30	58.18	65.65	65.10	55.54	48.02	62.72	61.43
+ PropCont [7]	+2.78	+3.05	62.62	59.28	66.04	65.47	57.58	49.51	64.23	62.86
+ Occ-MAE [3]	+5.99	+5.74	65.82	61.98	67.94	67.34	64.91	55.57	64.62	63.02
+ MAELi	+5.88	+5.92	65.72	62.15	67.90	67.34	65.14	56.32	64.13	62.79

Table 2. Performance comparison on the Waymo *val* set trained on 20% of the Waymo *train* set including AP scores. We compare different detectors trained from scratch with their pendants utilizing pre-trained weights from GCC-3D, ProposalContrast, Occupancy-MAE and the proposed MAELi.

Method	Pre-train	mAP	Car	Pedestrian	Cyclist
SECOND [6]	-	66.25	78.62	52.98	67.15
+ Occ-MAE [3]	KITTI 3D	66.71	78.90	53.14	68.08
+ ALSO [1]	nuScenes	67.29	78.65	55.17	68.05
+ ALSO [1]	KITTI 3D	66.86	78.78	53.57	68.22
+ ALSO [1]	KITTI360	67.40	78.63	54.23	69.35
+ MAELi	Waymo	68.31	78.44	55.72	70.78
+ MAELi	KITTI 3D	67.51	78.20	55.48	68.86
+ MAELi	KITTI360	68.74	78.44	56.00	71.79
PV-RCNN [4]	-	70.66	83.61	57.90	70.47
+ Occ-MAE [3]	KITTI 3D	71.73	83.82	59.37	71.99
+ ALSO [1]	nuScenes	72.20	83.77	58.49	74.35
+ ALSO [1]	KITTI 3D	71.96	83.67	58.48	73.74
+ ALSO [1]	KITTI360	72.69	83.39	60.83	73.85
+ MAELi	Waymo	71.79	83.38	58.53	73.45
+ MAELi	KITTI 3D	70.70	79.22	60.02	72.87
+ MAELi	KITTI360	73.03	83.99	62.43	72.67

Table 3. Quantitative results of our pre-training on SECOND and PV-RCNN on the KITTI 3D *val* set using the R_{11} metric.

Description	# Channels	Voxel/Tensor Stride	Spatial Dimension
Output BEV Encoder	512	-	188 × 188
Reshaping + Sampling	256	8 × 8 × 16	188 × 188 × 2
DBlock 1	64	8 × 8 × 8	188 × 188 × 5
DBlock 2	64	4 × 4 × 4	376 × 376 × 11
DBlock 3	32	2 × 2 × 2	752 × 752 × 21
DBlock 4	16	1 × 1 × 1	1504 × 1504 × 41

Table 4. Architecture of our decoder. We state the number of channels, the voxel stride and the maximum spatial dimension for the Waymo Open Dataset *after* each block. Stride and spatial dimensions are depicted in the format $x \times y \times z$. Each decoder block *inverts* one downsampling step from the sparse 3D encoder, eventually resulting in the original voxel stride.

4. Ablation Study

Analyzing Pipeline Components: We perform various experiments to investigate specific aspects of our pipeline, such as assessing the influence of our *distance weighting*

Operation	**Kernel Size**	**Stride**
Generative Transposed Convolution	$2 \times 2 \times 2^{\dagger}$	$2 \times 2 \times 2^{\dagger}$
Batch Norm	-	-
ReLU	-	-
Submanifold Sparse Convolution	$3 \times 3 \times 3$	$1 \times 1 \times 1$
Batch Norm	-	-
ReLU	-	-
Submanifold Sparse Convolution	$1 \times 1 \times 1$	$1 \times 1 \times 1$
Pruning	-	-

Table 5. Structure of each decoder block. We additionally state the operation's kernel size and stride, each in the format $x \times y \times z$. The upper part depicts the upsampling and feature transformation. The lower part uses the final feature representation from above and decides via classification whether a voxels is pruned or not. †These values deviate for DBlock 1, where it has a kernel size of $1 \times 1 \times 3$ and a stride of $1 \times 1 \times 2$ to invert the encoder's respective downsampling step.

for empty voxels, evaluating our *LiDAR-aware reconstruction* objective, and comparing the effectiveness of the *voxel-* and *spherical masking* strategies.

Therefore, we pre-train according to the *Data Efficiency* protocol depicted in the main manuscript (Section 4.1) and fine-tune a SECOND [6] model on 1% of the latter 399 sequences of the Waymo *train* set. To disable *distance weighting*, we set $w_j^{D,\mathbf{s}} = 1$ for all *empty* voxels $\mathbf{v}_j^{D,\mathbf{s}}$. To disable our *LiDAR-aware reconstruction*, we additionally consider all *unknown* voxels as *empty*.

We state the results in Table 6 on *Vehicle* LEVEL 2 across the distance ranges [0m, 30m), [30m, 50m) and [50m, $+\infty$). Our *LiDAR-aware reconstruction* objective improves the overall results across all ranges. While *voxel masking* naturally has a nearly equal impact over all ranges, our *spherical masking* is especially beneficial for the range [30m, 50m) with a gain of 1.87AP and 2.01APH. The overall lower impact on the far distance range (above 50m) is also reasonable, since at this distance only very few points are sampled on the same object.

Masking: We evaluated our *voxel masking* for different amounts of voxels. We maintain the training and evaluation scheme from above and vary the amount of kept voxels. The results are shown in Table 7. Keeping 60% of the voxels leads to the best overall results, eventually used for all other experiments with MAELi. However, in combination with *spherical masking* significantly fewer points than this fraction actually remain. To get an estimate, we evaluated the amount of effectively used points over 10000 iterations, resulting in a fraction of 15.57% on average.

5. Limitations

Even though our sparse decoder allows for a memory efficient reconstruction, the amount of reconstructed voxels is obviously constrained by the available compute infrastructure. Especially during the first iterations of our pre-training, while not sufficiently trained, some samples may lead to an uncontrolled reconstruction. In order to regulate the amount of reconstructed voxels and to avoid training breakdowns, we introduce two limiting factors. First, we estimate an average ground plane for each dataset and prune all reconstructed voxels that are $0.1m$ below this plane, as these generally do not contribute valuable information. Second, we introduce a threshold for the maximum amount of reconstructed voxels to ensure that we do not run into memory issues. If an upsampling step would generate more voxels than this limit, we randomly prune before the upsampling. For these pruned voxels, simply no loss is induced, which only slightly *delays* the training effect for these rare cases. For the detection experiments, we set the maximum number of total voxels to 6 million, which are easily processable, *e.g.* on an NVIDIA® GeForce® RTX 3090 GPU. We counted only 62 limit exceedances within the first 10k iterations of a random experiment.

6. Additional Insights

Reconstruction Capabilities: In Figure 2, we visualize the reconstruction capabilities of our *LiDAR-aware loss* on a full point cloud. It encourages the network to fill up gaps in the wall and reconstruct the occluded areas of cars.

Figure 3 highlights that reconstruction outcomes can vary across different objects, with some objects such as the less frequent trucks presenting greater challenges. However, the MAELi model demonstrates an advanced understanding of object semantics, enabling it to complete objects beyond the visible LiDAR input point cloud.

In Figure 4, we show the individual layers of two different reconstructed cars. We can see that our pre-training approach indeed encourages the model to go beyond the sampled LiDAR surface, reconstructing the entire car, also showing some hints for the correct placement of tires. Furthermore, especially parts of the interior, which are often surrounded by glass and thus, sometimes traversed by LiDAR beams, are seemingly left out during the reconstruction.

AP vs APH Data Efficiency: In Figure 5, we plot the data efficiency results on Waymo [5] using our MAELi-based pre-training on the SECOND [6] detector. All three classes benefit from our pre-training (solid lines) compared to the vanilla version trained from scratch (dotted lines). With little data, the vanilla detector especially struggles to estimate the heading, which can be clearly seen for the smaller, less represented classes *Pedestrian* and *Cyclist*. However, utilizing our initialization, a proper estimation and significant detection improvements are possible already, early on, with only few annotated data samples.

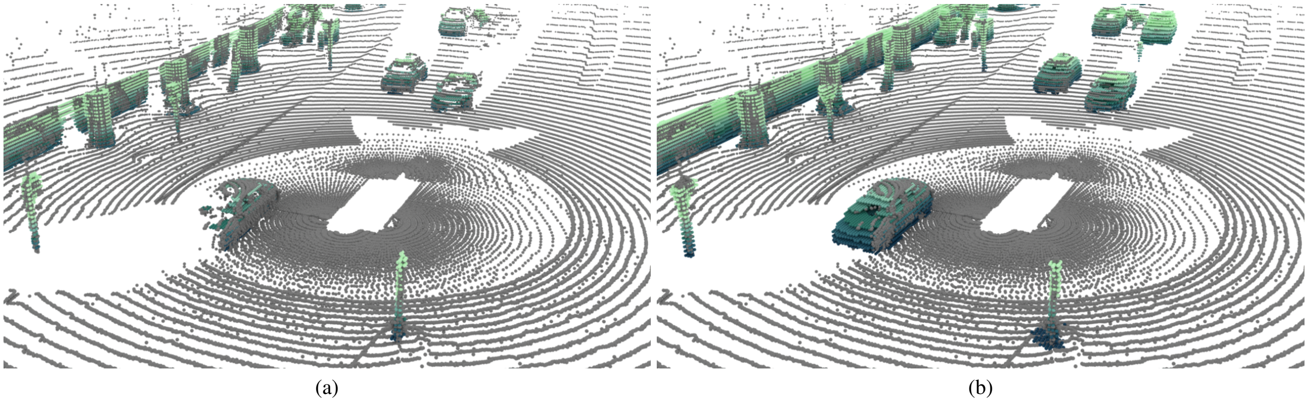

Figure 2. Completion results (a) without and (b) with our *LiDAR-aware reconstruction* on a full point cloud (gray). For visualization purposes, we color-coded the output points by their z-coordinate and removed the reconstructed ground plane.

Method	**3D AP/APH (LEVEL 2)**							
	Overall		**[0m, 30m)**		**[30m, 50m)**		**[50m, +inf)**	
	AP	**APH**	**AP**	**APH**	**AP**	**APH**	**AP**	**APH**
MAELi	51.05	50.11	80.22	79.37	48.86	47.64	21.09	19.98
w/o DW	50.91	49.85	79.95	79.04	48.62	47.20	21.47	20.31
w/o LAR	50.05	48.87	79.33	78.20	47.77	46.27	20.66	19.56
w/o VM	48.99	47.86	78.45	77.46	46.43	45.05	19.47	18.23
w/o SM	49.90	48.85	79.50	78.54	46.99	45.63	20.53	19.37
Baseline	41.64	40.02	72.59	70.79	37.09	34.86	13.14	11.96

Table 6. Impact of the components of MAELi evaluated on the Waymo *val* set for *Vehicle*. We disable *distance weighting* (w/o DW), *LiDAR-aware reconstruction* (w/o LAR), *voxel masking* (w/o VM) and *spherical masking* (w/o SM). We use the first 399 sequences of the Waymo *train* set for pre-training and 1% of the second 399 sequences for fine-tuning. We utilize a SECOND [6] model and state a version trained from scratch as baseline.

Fraction Voxels	**3D AP/APH (LEVEL 2)**							
	Overall		**Vehicle**		**Pedestrian**		**Cyclist**	
	AP	**APH**	**AP**	**APH**	**AP**	**APH**	**AP**	**APH**
0.8	45.34	32.84	50.26	49.10	48.03	24.33	37.74	25.09
0.7	44.91	32.10	50.35	49.19	47.54	24.29	36.83	22.83
0.6	46.01	33.05	51.05	50.11	48.13	24.65	38.86	24.38
0.5	45.60	32.27	50.87	49.79	47.08	23.72	38.86	23.31
0.4	45.79	31.85	50.36	49.24	48.27	24.50	38.74	21.81
Baseline	31.09	22.25	41.64	40.02	33.39	17.45	18.24	9.29

Table 7. Impact of different amounts of voxels kept during *voxel masking* evaluated on the Waymo *val* set for *Vehicle*. We use the first 399 sequences of the Waymo *train* set for pre-training and 1% of the second 399 sequences for fine-tuning. We utilize a SECOND [6] model and state a version trained from scratch as baseline.

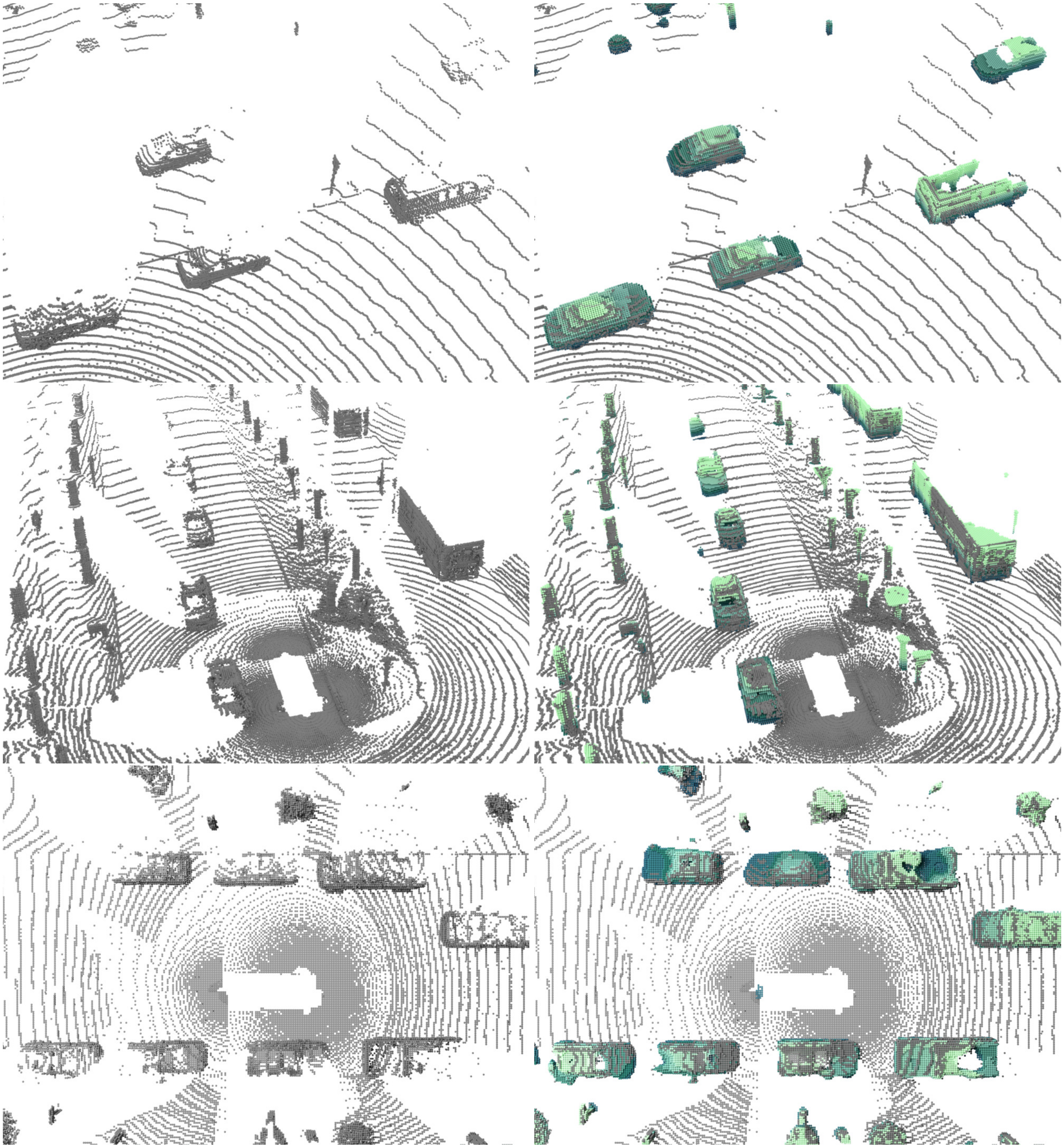

Figure 3. Further reconstruction results. We show the input point cloud on the left and the completed point cloud on the right. MAELi's reconstruction exhibits imperfections when reconstructing objects that are sparsely sampled or less frequent, such as trucks. However, it shows an apparent understanding of traffic scenes beyond a LiDAR's 2.5D sampling, *e.g*. by symmetrically completing occluded parts of cars and poles. For visualization purposes, we color-coded the output points by their z-coordinate and removed the reconstructed ground plane.

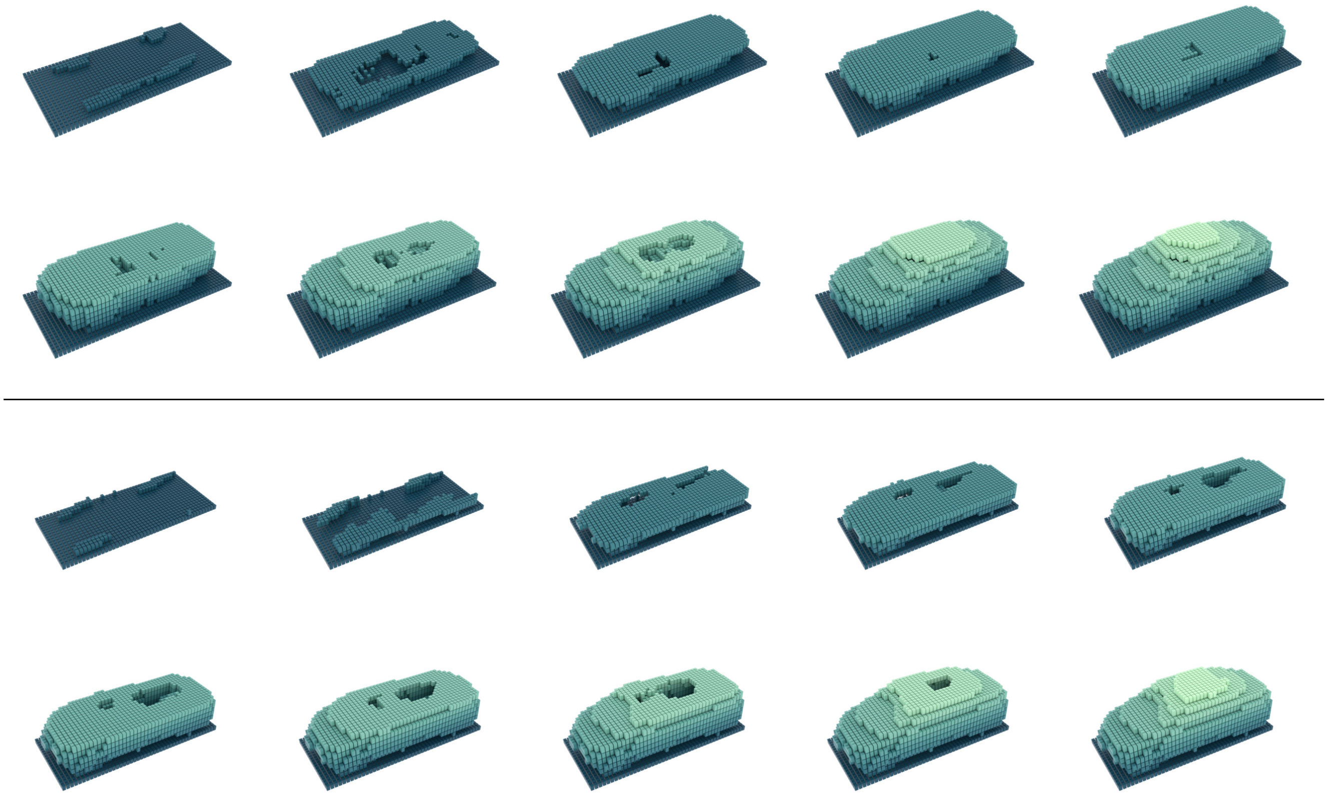

Figure 4. Different layers of two reconstructed cars. We observed that the reconstructed cars are often hollow. There are visible tendencies to leave parts of the interior free.

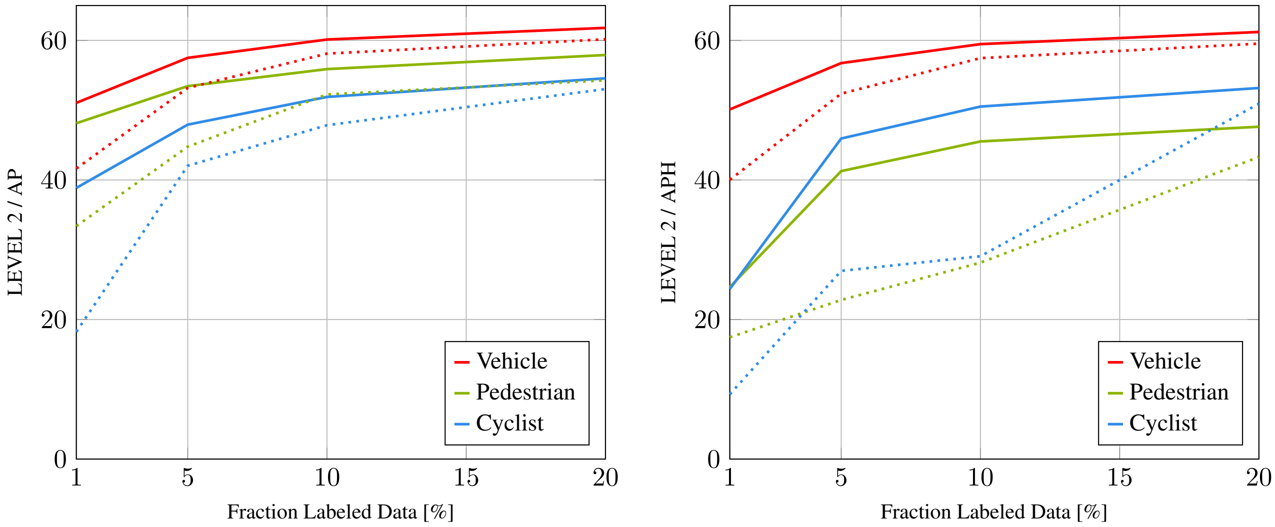

Figure 5. Results of our pre-training on SECOND [6] on the Waymo *val* set, using different amounts of labeled data. We use the first 399 sequences of the Waymo *train* set without any labels for pre-training and different fractions of the latter 399 sequences for fine-tuning. The *solid lines* are the results utilizing our pre-training with MAELi. The *dotted lines* denote the version trained from scratch.

References

[1] Alexandre Boulch, Corentin Sautier, Björn Michele, Gilles Puy, and Renaud Marlet. ALSO: Automotive Lidar Self-supervision by Occupancy estimation. In *Proc. CVPR*, 2023. 1, 2

[2] Hanxue Liang, Chenhan Jiang, Dapeng Feng, Xin Chen, Hang Xu, Xiaodan Liang, Wei Zhang, Zhenguo Li, and Luc Van Gool. Exploring Geometry-Aware Contrast and Clustering Harmonization for Self-Supervised 3D Object Detection. In *Proc. ICCV*, 2021. 2

[3] Chen Min, Xinli Xu, Dawei Zhao, Liang Xiao, Yiming Nie, and Bin Dai. Occupancy-MAE: Self-Supervised Pre-Training Large-Scale LiDAR Point Clouds With Masked Occupancy Autoencoders. *IEEE Transactions on Intelligent Vehicles*, 2023. 1, 2

[4] 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 *Proc. CVPR*, 2020. 1, 2

[5] Pei Sun, Henrik Kretzschmar, Xerxes Dotiwalla, Aurelien Chouard, Vijaysai Patnaik, Paul Tsui, James Guo, Yin Zhou, Yuning Chai, Benjamin Caine, Vijay Vasudevan, Wei Han, Jiquan Ngiam, Hang Zhao, Aleksei Timofeev, Scott Ettinger, Maxim Krivokon, Amy Gao, Aditya Joshi, Yu Zhang, Jonathon Shlens, Zhifeng Chen, and Dragomir Anguelov. Scalability in Perception for Autonomous Driving: Waymo Open Dataset. In *Proc. CVPR*, 2020. 1, 3

[6] Yan Yan, Yuxing Mao, and Bo Li. SECOND: Sparsely Embedded Convolutional Detection. *Sensors*, 18(10):3337, 2018. 2, 3, 4, 6

[7] Junbo Yin, Dingfu Zhou, Liangjun Zhang, Jin Fang, Cheng-Zhong Xu, Jianbing Shen, and Wenguan Wang. ProposalContrast: Unsupervised Pre-training for LiDAR-based 3D Object Detection. In *Proc. ECCV*, 2022. 2

[8] Tianwei Yin, Xingyi Zhou, and Philipp Krähenbühl. Center-Based 3D Object Detection and Tracking. In *Proc. CVPR*, 2021. 1, 2