Supplementary Material

A Point Cloud Sequence Representation

This section provides details on our range image representation. We convert each 3D LiDAR point $\boldsymbol{p} = (x, y, z)$ to spherical coordinates and map them to image coordinates (u, v) resulting in a projection $\Pi : \mathbb{R}^3 \mapsto \mathbb{R}^2$

$$(u, v)^\top = \left(\frac{1}{2} \left[1 - \arctan(y, x)\, \pi^{-1} \right] W, \left[1 - \left(\arcsin(z\, r^{-1}) + \mathrm{f}_{\mathrm{up}} \right) \mathrm{f}^{-1} \right] H \right)^\top, \tag{5}$$

with (H, W) as the height and width of the range image and $\mathrm{f} = \mathrm{f}_{\mathrm{up}} + \mathrm{f}_{\mathrm{down}}$ is the vertical field-of-view of the sensor. Each range image pixel stores the range value $r = ||\boldsymbol{p}||_2$ of the projected point or zero if there exist no projected point in the corresponding pixel. In case multiple points project to the same pixel due to rounding, we keep the closer one since closer points are more important than far-away points. A range image pixel with coordinates u and v and range r is re-projected by solving Eq. (5) for pitch $\theta = \arctan(y, x)$ and yaw $\gamma = \arcsin(z\, r^{-1})$ and computing

$$(x, y, z)^\top = (r \cos(\theta) \cos(\gamma), r \cos(\theta) \sin(\gamma), r \sin(\theta))^\top. \tag{6}$$

In contrast to the existing work using range images [38, 39], we found that additionally using x, y, z, and intensity values did not improve the performance of point cloud prediction. We concatenate the range images along the temporal dimension to obtain the input 3D tensor with a size of (P, H, W) which is further processed by our encoder-decoder network.

B Experimental Setting

As explained in the experimental section of our main paper, we follow the experimental setup of MoNet [9]. Thus, we train the models on sequences 00 to 05 of the KITTI Odometry dataset [35], validate on sequences 06 and 07, and test on sequences 08 to 10. We project the point clouds from the Velodyne HDL-64E laser scanner into range images of size $H \times W = 64 \times 2048$ according to Eq. (5). As explained in Sec. 3.1, we map the normalized output values of the network to a predicted range interval defined before training. Based on the minimum and maximum range values of the training data, we set the minimum predicted range to 1 m and the maximum range to 85 m for KITTI Odometry [35] and to 110 m for Apollo [37].

C Amount of Training Data

This experiment demonstrates the advantage of a large amount of data for training our point cloud prediction method. Since the approach is self-supervised, data obtained from the LiDAR sensor can be directly used for training without expensive labeling. Fig. 6 shows the total validation loss on sequences 06 and 07 after 50 epochs with respect to the number of training iterations. Note that the final number of training iterations within 50 epochs depends on the total amount of samples in the training data. Fig. 6 shows the validation loss with the previously explained experimental setting on the left (*yellow*). If we train a model on a larger dataset containing the KITTI sequences 00 to 05 plus 11 to 15, the validation loss decreases faster and converges to a lower minimum (*blue*). For the largest dataset containing sequences 00 to 05 plus 11 to 21 (*green*), the validation loss is further reduced indicating that more data improves the model's performance on the unseen validation set. The individual loss components in Tab. 3 show that we can improve all losses by using more data. This emphasizes that self-supervised point cloud prediction profits from a large amount of sensor data without the need for expensive labeling.

D Additional Qualitative Results

In this section, we provide additional analysis of our predicted masked range images and the re-projected 3D point clouds. Our method can estimate the ego-motion of the vehicle resulting in an accurate prediction of the environment in Fig. 7. The moving cyclist in front of the ego vehicle is consistently predicted in all five future frames which demonstrates that our predicted range images like in Fig. 8 preserve details in the scene.

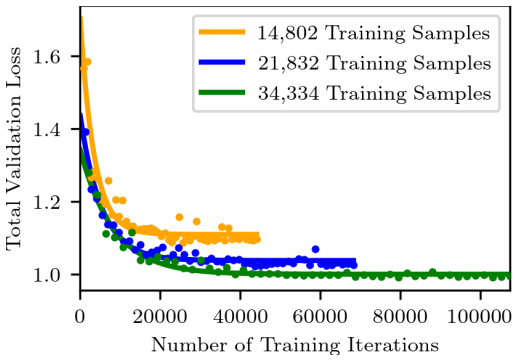

Figure 6: Total validation loss for a given number of training iterations with different training data sizes. The solid line is a fitted exponential curve.

Loss	14,082 Samples	21,832 Samples	34,334 Samples
$\mathcal{L}_{\text{R}}$ [m]	0.798	0.735	**0.710**
$\mathcal{L}_{\text{Mask}}$	0.299	0.290	**0.285**
$\mathcal{L}_{\text{CD}}$ $\left[\text{m}^2\right]$	0.985	0.795	**0.768**

Table 3: Final validation loss components for different numbers of training dataset sizes. We train all methods for 50 epochs.

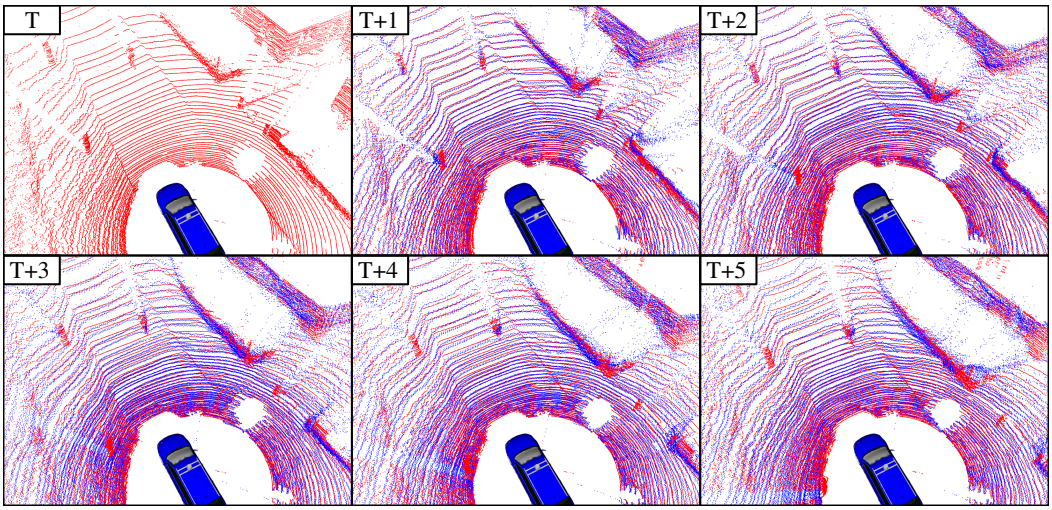

Figure 7: Last received point cloud at time T and the predicted next 5 future point clouds. Ground truth points are shown in red and predicted points in blue. The scene is taken from the KITTI Odometry sequence 08 that was not used during training.

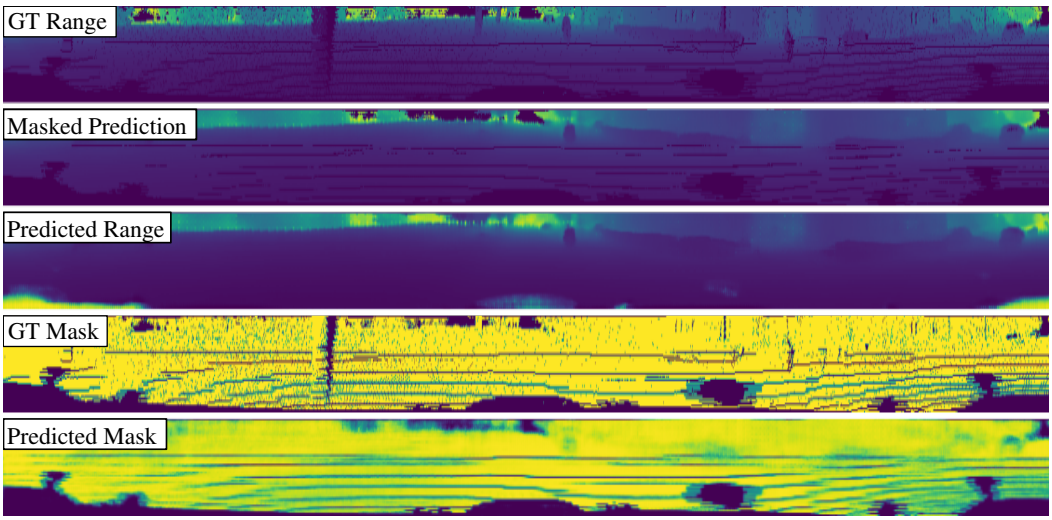

Figure 8: Ground truth and predictions of range image and re-projection mask at the last prediction step. The range is color-coded with increasing range from blue to yellow. Invalid points are shown in dark blue. For the mask, yellow indicates a valid point whereas dark blue represents invalid points. The masked prediction is the combination of predicted range and re-projection mask.

E Statistical Analysis

For a more detailed quantitative analysis for full-scale point cloud prediction, we compare box plots for each prediction step in Fig. 9 to Fig. 13 and across all prediction steps in Fig. 14. Each plot shows the median (orange line), minimum (lower bar), maximum (upper bar), first quartile (lower colorized rectangle), and third quartile (upper colorized rectangle) of the Chamfer distances computed between the predicted and ground truth point clouds for the test set. As shown in the main paper for the mean Chamfer distances, the constant velocity baseline achieves the lowest median Chamfer distances for short prediction horizons. Since all baselines ignore moving objects for the prediction, our method outperforms them for larger time steps resulting in a lower median and interquartile ranges. The statistical results support our claims made in the paper.

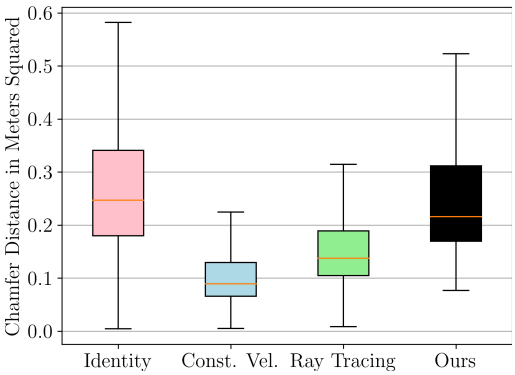

Figure 9: Prediction step 1

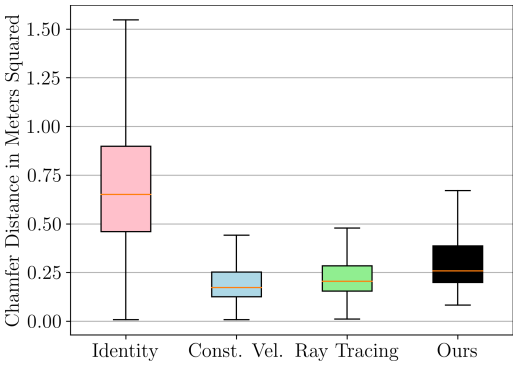

Figure 10: Prediction step 2

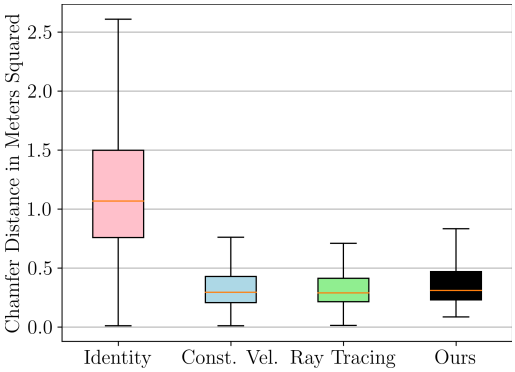

Figure 11: Prediction step 3

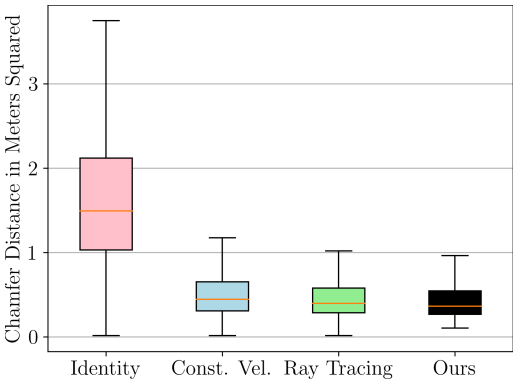

Figure 12: Prediction step 4

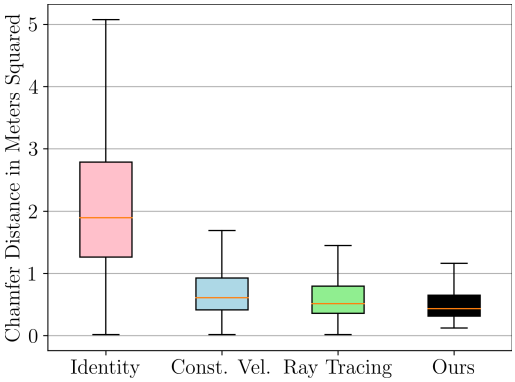

Figure 13: Prediction step 5

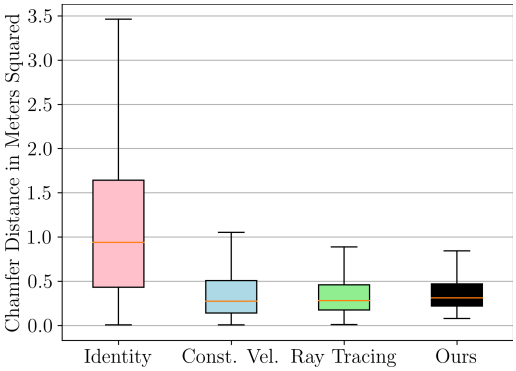

Figure 14: All steps