Supplementary Material:
Walking Your LiDOG: A Journey Through Multiple Domains for LiDAR Semantic Segmentation
Cristiano Saltori\textsuperscript{1}, Aljoša Ošep\textsuperscript{2}, Elisa Ricci\textsuperscript{1,3}, Laura Leal-Taixe\textsuperscript{4}
\textsuperscript{1} University of Trento, \textsuperscript{2} TU Munich, \textsuperscript{3} Federazione Bruno Kessler, \textsuperscript{4} NVIDIA
cristiano.saltori@unitn.it

Abstract
We provide supplementary material in support of the main paper. We organize the content as follows:
• In Sec. 1, we report the implementation details of our experiments, the additional experimental evaluations on Synth4D-nuScenes→Real, and an additional ablation of LiDOG.
• In Sec. 2, we report and discuss qualitative results comparing the performance of LiDOG with all the compared baselines and ground truth annotations.

1. Experimental Evaluation
We provide the implementation details of our experiments in Sec. 1.1. In Sec. 1.2, we complete the set of experiments of the main paper and report the experimental results obtained on Synth4D-nuScenes→Real. In Sec. 1.3, we further ablate how the input resolution affects LiDOG performance.

1.1. Implementation details
We implement our method and all the baselines by using the PyTorch framework. We use MinkowskiNet [2] as a sparse convolutional backbone in all our experiments and train until convergence with voxel size 0.05m, total batch size of 16, learning rate 0.01 and ADAM optimizer [3]. In the experiments we use random rotation, scaling, and downsampling for better convergence of baselines and LiDOG. We set rotation bounds between $[-\pi/2, \pi/2]$, scaling between $[0.95, 1.05]$, and perform random downsampling for 80% of the patch points. We set projection bounds $B^{3D}$ based on the input resolution. We set $b_x$ and $b_z$ to 50m in the denser source domains of Synth4D-KITTI and SemanticKITTI and $b_x$ and $b_z$ to 30m in the sparser domains of Synth4D-nuScenes and nuScenes. Quantization parameters $x_q$ and $z_q$ are computed based on the spatial bounds in order to obtain BEV labels of $168 \times 168$ pixels. For downsampling dense features, we use a max pooling layer with window size 5, stride 3, and padding 1. The LiDOG 2D decoder is implemented with a series of three 2D convolutional layers interleaved by batch normalization layers. ReLU activation function is used on all the layers while softmax activation is applied on the last layer.

1.2. Synth4D-nuScenes→Real
In Tab. 1, we report the results for single-source Synth4D-nuScenes→Real. We observe a 41.82 mIoU gap (SemanticKITTI) between the source (19.71 mIoU) and target (61.53 mIoU) models on Synth4D-nuScenes→SemanticKITTI and a 23.92 mIoU gap (nuScenes), between source (24.57 mIoU) and target (48.49 mIoU) on Synth4D-nuScenes→nuScenes. Among the baselines, augmentation-based methods are the most effective with Mix3D achieving 31.64 mIoU (SemanticKITTI) and 31.23 mIoU (nuScenes). LiDOG reduces the domain gap between source and target models and outperforms all the compared baselines in all the scenarios. For example, on Synth4D-nuScenes→Real (Tab. 1), we obtain 34.79 mIoU, a $+15.08$ improvement over the source model.

1.3. BEV resolution
We study the impact of BEV image resolution on the LiDOG performance. We adapt the feature resolution by applying several pooling steps and the label resolution via the quantization step size. In Fig. 1, we report the results obtained on Synth4D-KITTI→Real, when using BEV features with 50% and 75% of the initial resolution (100%). Interestingly, on Synth4D-KITTI→nuScenes we observe a slight improvement with 75% resolution. However, we obtain consistently top performance on both domains with the full resolution (100%).
Table 1: Synth4D-nuScenes → Real, single-source. We train our model on Synth4D-nuScenes and test on SemanticKITTI and nuScenes. LiDOG improves over the source models by +15.08 mIoU on SemanticKITTI and by +9.21 mIoU on nuScenes. LiDOG outperforms all the compared baselines. Lower bound (red): a model trained in the source domain without the help of DG techniques. Upper bound (blue): a model directly trained on target data.

<table>
<thead>
<tr>
<th>Info</th>
<th>Method</th>
<th>Vehicle</th>
<th>Person</th>
<th>Road</th>
<th>Sidewalk</th>
<th>Terrain</th>
<th>Manmade</th>
<th>Vegetation</th>
<th>mIoU</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lower bound</td>
<td>Source</td>
<td>14.54</td>
<td>2.41</td>
<td>32.78</td>
<td>14.86</td>
<td>6.4</td>
<td>30.89</td>
<td>36.07</td>
<td>19.71</td>
</tr>
<tr>
<td>Aug</td>
<td>Mix3D [5]</td>
<td>37.38</td>
<td>7.26</td>
<td>56.90</td>
<td>21.06</td>
<td>10.60</td>
<td>34.46</td>
<td>53.79</td>
<td>31.64</td>
</tr>
<tr>
<td>2D DG</td>
<td>IBN [6]</td>
<td>47.15</td>
<td>7.81</td>
<td>50.74</td>
<td>4.51</td>
<td>15.20</td>
<td>29.53</td>
<td>51.35</td>
<td>29.47</td>
</tr>
<tr>
<td></td>
<td>RobustNet [1]</td>
<td>21.19</td>
<td>8.56</td>
<td>44.46</td>
<td>10.80</td>
<td>15.06</td>
<td>11.95</td>
<td>30.59</td>
<td>20.37</td>
</tr>
<tr>
<td>3D UDA</td>
<td>SN [8]</td>
<td>11.56</td>
<td>2.38</td>
<td>37.50</td>
<td>5.19</td>
<td>20.70</td>
<td>39.23</td>
<td>18.20</td>
<td>18.20</td>
</tr>
<tr>
<td>3D DG</td>
<td>Ours</td>
<td>55.08</td>
<td>11.42</td>
<td>59.48</td>
<td>26.10</td>
<td>2.78</td>
<td>34.83</td>
<td>53.82</td>
<td>34.79</td>
</tr>
<tr>
<td>Upper bound</td>
<td>Target</td>
<td>81.57</td>
<td>23.23</td>
<td>82.09</td>
<td>57.64</td>
<td>46.14</td>
<td>64.14</td>
<td>75.21</td>
<td>61.53</td>
</tr>
</tbody>
</table>

Figure 1: BEV image resolution: We compare the performance while changing the BEV image resolution on SemanticKITTI (left) and nuScenes (right), Source: Synth4D-KITTI.

2. Qualitative evaluation

We report additional qualitative results for each baseline and in all the studied generalization directions. In Fig. 2-4 we show qualitative results in the Synth→Real setting: Synth4D-kitti→Real (Fig. 2), Synth4D-nuScenes→Real (Fig. 3) and Synth4D-kitti+Synth4D-nuScenes→Real (Fig. 4). In Fig. 5-6 we show qualitative results in the SemanticKITTI→nuScenes and nuScenes→SemanticKITTI directions, respectively. Source predictions are often incorrect and spatially inconsistent. Baselines consistently improve over the source model performance. LiDOG achieves the overall best performance with improved and more precise predictions. This can be seen in all the reported results, both synth→real and real→real.

References


Figure 2: **Qualitative results.** Top: Synth4D-kitti→SemanticKITTI, bottom: Synth4D-kitti→nuScenes. LiDOG improves over source and baselines, e.g., we observe the improvements of *road* in SemanticKITTI and *vehicle* in nuScenes.
Figure 3: **Qualitative results.** *Left:* Synth4D-nuScenes $\rightarrow$ SemanticKITTI, *right:* Synth4D-nuScenes $\rightarrow$ nuScenes. LiDOG improves over source and baselines, e.g., we observe the improvements of *road* in SemanticKITTI and *manmade* in nuScenes.
Figure 4: **Qualitative results.** *Top:* Synth4D-kitti+Synth4D-nuScenes→SemanticKITTI, *bottom:* Synth4D-kitti+Synth4D-nuScenes→nuScenes. LiDOG improves over source and baselines, *e.g.*, we observe the improvements of *vegetation* in SemanticKITTI and *road* in nuScenes.
Figure 5: **Qualitative results.** Top: SemanticKITTI→nuScenes. LiDOG improves over source and baselines, e.g., we observe the improvements in *terrain*, *road*, and *manmade*.
Figure 6: **Qualitative results.** Top: nuScenes→SemanticKITTI. LiDOG improves over source and baselines, *e.g.*, we observe the improvements in *sidewalk* and *road*.