Searching Efficient 3D Architectures with Sparse Point-Voxel Convolution

A.1 Implementation Details

A.1.1 Backbone Networks

We introduce two backbone networks which are used as starting points for SPVNAS.

Based on MinkowskiNet [2], we build our backbone network by wrapping Sparse Convolution layers in the 34-layer MinkowskiNets [2] using Sparse Point-Voxel Convolution. Specifically, the U-Net consists of 4 SPVConv layers. The first layer voxelizes the input data and devoxelizes at the end of the stemming stage of MinkowskiNet (*i.e.* before downsampling). The second SPVConv generates the volumetric representation subsequently and turns back to point-based representation after all four downsampling stages in MinkowskiNet. Each of the following two SPVConv layers consists of two upsampling stages in MinkowskiNet.

Based on PVCNN [5], we also design our smaller backbone by directly replacing the volumetric convolutions with a small Sparse Convolution network containing one convolution layer (followed by normalization and activation layers) and two residual blocks.

A.1.2 SPVNAS Details

On SemanticKITTI [1], we train SPVNAS super network for 15 epochs that supports fine-grained channel setting with a starting learning rate 0.24 and cosine learning rate decay. Then we train for another 15 epochs to incorporate variable network depth with a starting learning rate 0.096 and cosine learning rate decay. We perform evolutionary architecture search with $P = 50$ for 20 generations on sequence 08 (the official validation set). Best architecture is directly extracted from the super network and submitted to the test server after finetuning for 10 epochs with a starting learning rate of 0.032 and cosine learning rate schedule.

A.2 More Results

A.2.1 3D Scene Segmentation

We present detailed result comparisons between SPVNAS, SPVCNN and MinkowskiNets [2] on both the official test set and validation set (sequence 08) of SemanticKITTI [1] in Table 1 and Table 2. For the results on the validation set in Table 2, we run SPVNAS pipeline again on sequences 00-07 and 09, leaving out sequence 10 as the mini-validation set and report the results on sequence 08. We observe similar trends on both test and validation results: both a better 3D module (SPVConv) and the 3D Neural Architecture Search (3D-NAS) pipeline improves the performance of MinkowskiNet [2].

	#Params (M)	#MACs (G)	Latency (ms)	Mean IoU
MinkowskiNet [2]	2.2	11.3	115.7	57.5
SPVCNN (Ours)	2.2	11.9	124.3	58.5
SPVNAS (Ours)	2.6	15.0	**110.4**	**63.6**
MinkowskiNet [2]	5.5	28.5	152.0	60.0
SPVCNN (Ours)	5.5	30.0	160.9	61.6
SPVNAS (Ours)	**4.2**	**20.0**	**132.6**	**64.5**
MinkowskiNet [2]	8.8	45.9	207.4	62.8
SPVCNN (Ours)	8.8	47.4	214.3	64.4
SPVNAS (Ours)	**5.1**	**24.4**	**144.3**	**65.2**
MinkowskiNet [2]	21.7	113.9	294.0	61.1
SPVCNN (Ours)	21.8	118.6	317.1	63.8
SPVNAS (Ours)	**7.5**	**34.1**	**166.1**	66.0
SPVNAS (Ours)	12.5	73.8	259.9	**66.4**

Table 1. Detailed results of 3D scene segmentation on SemanticKITTI [1] test set. All results are obtained after 5-view rotation augmentation.

A.2.2 3D Object Detection

We further present the results of SPVCNN on KITTI [4] validation set in Table 3. We train both SECOND [6] and our SPVCNN on the KITTI training set for three times and average the results. Similar to the results on the test split, our SPVCNN has consistent improvement over almost all classes on the KITTI validation set.

A.3 More Visualizations

We provide more visualizations for SPVNAS and MinkowskiNets [2] in Figure A1 to show that the improvements brought by SPVConv on small objects and region boundaries are general. For example, in the first row, MinkowskiNets wrongly segments the entire traffic sign and bicycle instances. However, our SPVNAS is capable of making almost no mistake on these very small objects. Also, we observe in next two rows that MinkowskiNets don't perform well on the sidewalk-building or sidewalk-vegetation boundary where SPVNAS has a clear advantage.

In Figure A2 we show the comparison between our smaller SPVNAS and DarkNet53 [1]. DarkNets use spherical projections to project 3D point clouds to a 2D plane such that part of geometry information is lost. Our SPVNAS, in contrast, directly learns on 3D data and is more powerful in geometric modeling. We observe that SPVNAS has significant advantages in both large regions (last 3 rows) and smaller objects (first row).

We finally demonstrate the superior performance of SPVCNN over SECOND [6], a state-of-the-art single stage 3D detector in Figure A3. SPVCNN has big advantage in scenes with crowded small objects. In the first row of Figure A3, our SPVCNN is capable of detecting a challenging small pedestrian instance

	#Params (M)	#MACs (G)	Latency (ms)	Mean IoU
MinkowskiNet [2]	5.5	28.5	152.0	58.9
SPVCNN (Ours)	5.5	30.0	160.9	60.7
SPVNAS (Ours)	**4.5**	**24.6**	158.1	**62.9**
MinkowskiNet [2]	8.8	45.9	207.4	60.3
SPVCNN (Ours)	8.8	47.4	214.3	61.4
SPVNAS (Ours)	**7.0**	**34.7**	**175.8**	**63.5**
MinkowskiNet [2]	21.7	113.9	294.0	61.1
SPVCNN (Ours)	21.8	118.6	317.1	63.8
SPVNAS (Ours)	**10.8**	**64.5**	**248.7**	**64.7**

Table 2. Results of 3D scene segmentation on SemanticKITTI [1] validation set. All results are obtained from single view.

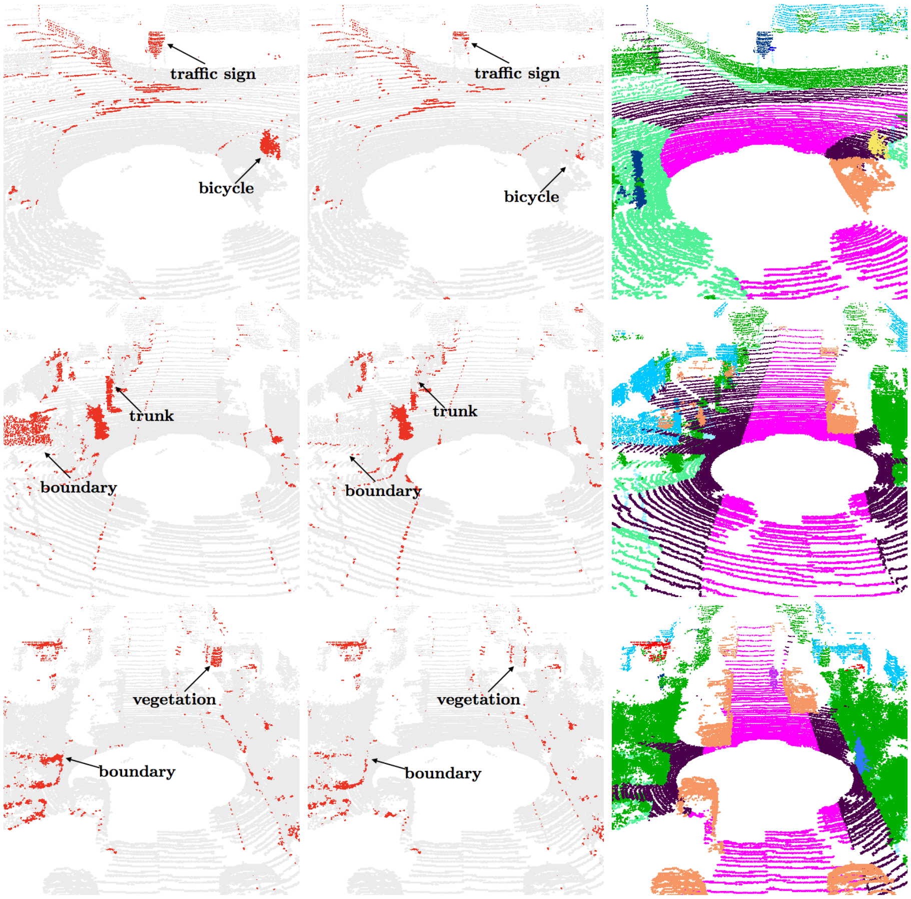
	Car			Cyclist			Pedestrian		
	Easy	Mod.	Hard	Easy	Mod.	Hard	Easy	Mod.	Hard
SECOND [6]	89.8	80.9	78.4	82.5	62.8	58.9	**68.3**	60.8	55.3
SPVCNN (Ours)	**90.9**	**81.8**	**79.2**	**85.1**	**63.8**	**60.1**	68.2	**61.6**	**55.9**

Table 3. Results of 3D object detection on the validation set of KITTI [3].

missed by SECOND and SPVCNN also avoids duplicate detections made by SECOND in the next three rows.

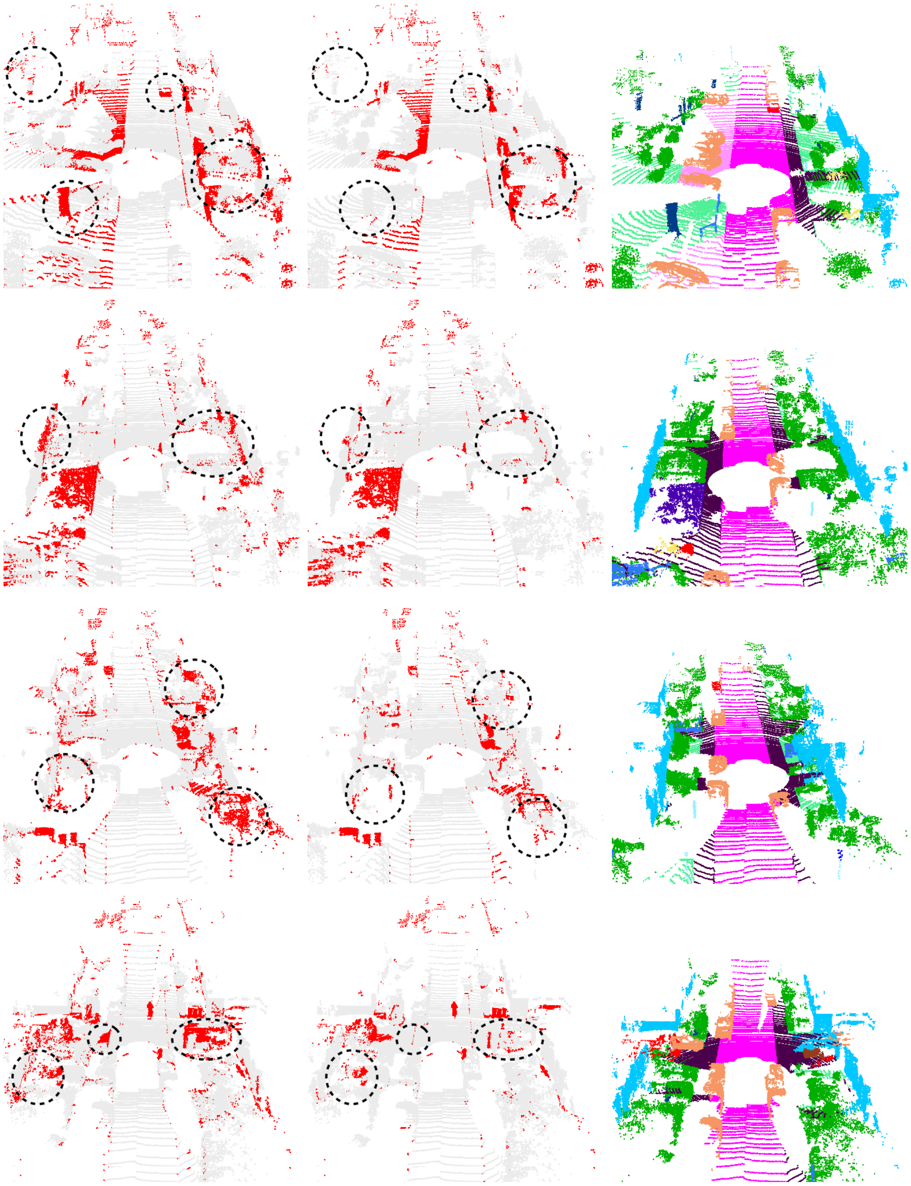
References

1. Behley, J., Garbade, M., Milioto, A., Quenzel, J., Behnke, S., Stachniss, C., Gall, J.: SemanticKITTI: A Dataset for Semantic Scene Understanding of LiDAR Sequences. In: ICCV (2019)
2. Choy, C., Gwak, J., Savarese, S.: 4D Spatio-Temporal ConvNets: Minkowski Convolutional Neural Networks. In: CVPR (2019)
3. Geiger, A., Lenz, P., Stiller, C., Urtasun, R.: Vision meets Robotics: The KITTI Dataset. IJRR (2013)
4. Geiger, A., Lenz, P., Urtasun, R.: Are we ready for autonomous driving? the kitti vision benchmark suite. In: CVPR (2012)
5. Liu, Z., Tang, H., Lin, Y., Han, S.: Point-Voxel CNN for Efficient 3D Deep Learning. In: NeurIPS (2019)
6. Yan, Y., Mao, Y., Li, B.: SECOND: Sparsely Embedded Convolutional Detection. Sensors (2018)



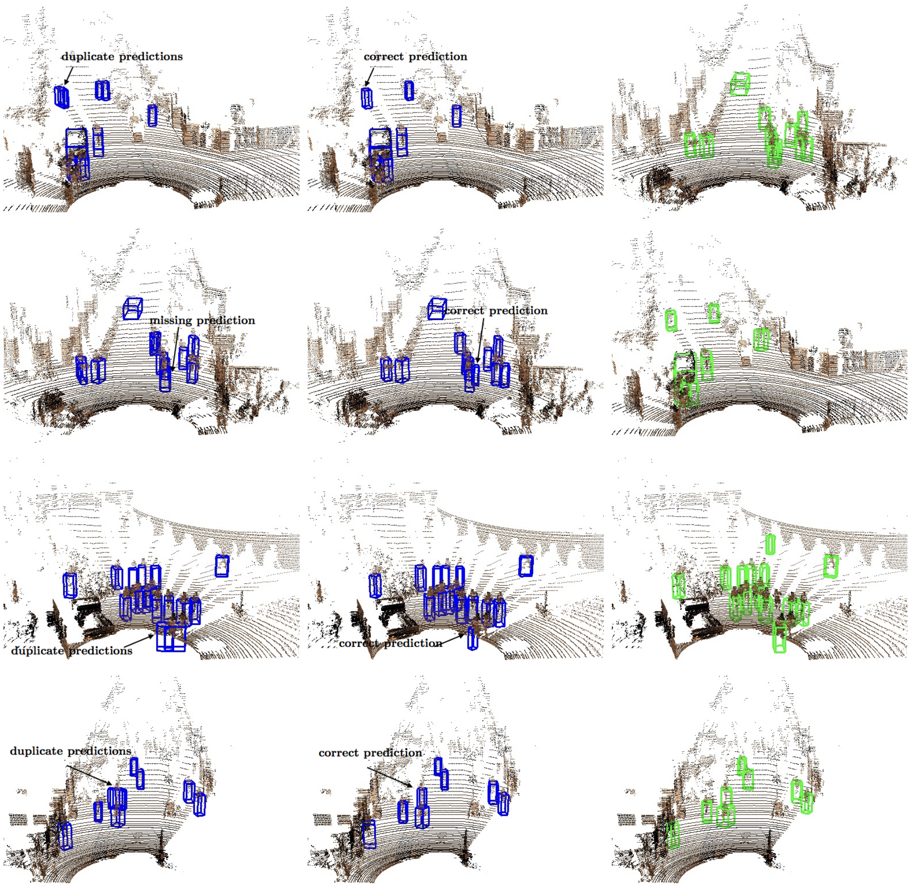
(a) Error by MinkNet **(b)** Less error by SPVNAS **(c)** Ground Truth

Fig. A1. More comparisons between MinkowskiNets [2] and our SPVNAS.



(a) Error by DarkNet **(b)** Less error by SPVNAS **(c)** Ground Truth

Fig. A2. Qualitative comparisons between DarkNet53 [1] and our smaller SPVNAS.



(a) Pred by SECOND **(b)** Pred by SPVCNN **(c)** Ground Truth

Fig. A3. Qualitative comparisons between SECOND [6] and our SPVCNN.