Monocular 3D Object Detection with Pseudo-LiDAR Point Cloud Supplementary Material

Xinshuo Weng
Carnegie Mellon University
xinshuow@cs.cmu.edu

Kris Kitani
Carnegie Mellon University
kkitani@cs.cmu.edu

0. Overview

This document provides additional technical details, extra experiments, more visualization and justification of our idea. Each section in this document corresponds to the subsection of the approach section in the main paper.

1. Pseudo-LiDAR Generation

Additional Visualization of LiDAR vs. Pseudo-LiDAR

We provide the additional visual comparison between the LiDAR and pseudo-LiDAR point cloud in Figure 2, demonstrating again the *local misalignment* and *long tail* issues we have observed in the pseudo-LiDAR point cloud.

2. 2D Instance Mask Proposal Detection

Justification of Using Instance Mask Proposal for 3D Point Cloud Segmentation and 3D Box Estimation

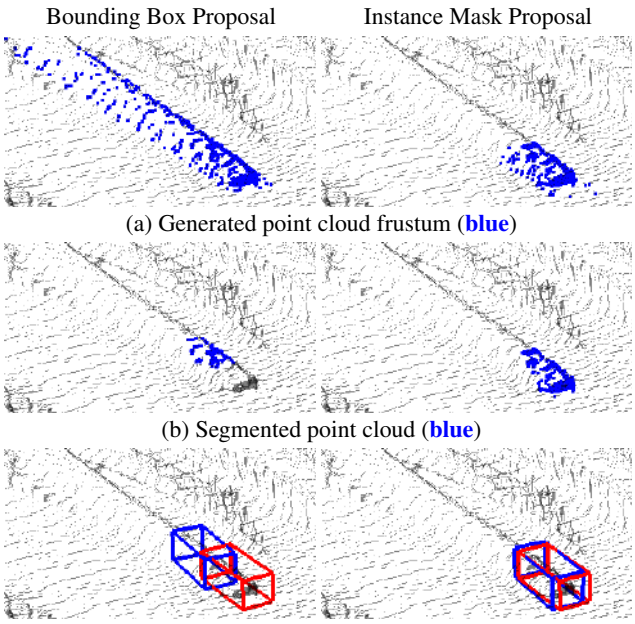
In the main paper, we justify the effectiveness of using instance mask proposal to generate the point cloud frustum with no tail. We provide further details here about how the generated point cloud frustum with no tail can improve the results in the subsequent 3D point cloud segmentation and 3D bounding box estimation module.

An example of visualization is shown in Figure 1. In the left column, the point cloud frustum is generated from the bounding box proposal and has a long tail, making the 3D point cloud segmentation task difficult (*e.g.*, in the middle left of the figure, the segmented point cloud misses lots of points belonging to the object and still contains background points). This further causes a poor 3D box estimation, especially a poor object center estimate. On the other hand, the point cloud frustum generated from the instance mask proposal with no tail, shown in the right column, can reduce a large number of background points so that the subsequent point cloud segmentation and 3D box estimation can be more accurate.

Quantitative Comparison of the 2D Instance Mask and Bounding Box Proposal

To compare our instance mask proposals with the bounding box proposals used in the baseline, we compute the minimum bounding rectangle (MBR) of our 2D mask proposals. We report the average precision (in %) of car category on val set of KITTI [1] 2D object detection benchmark as AP_{2D} in Table 1. IoU thresholds of 0.5 and 0.7 are used.

Unsurprisingly, we find that the MBR of our mask pro-



(a) Generated point cloud frustum (**blue**)

(b) Segmented point cloud (**blue**)

(c) 3D bounding box estimation (**blue**) and ground truth (**red**)

Figure 1: **Justification of Using Instance Mask Proposal.** We visualize the generated point cloud frustum (top row), segmented point cloud (middle row) and the 3D box prediction (bottom row) from the bounding box and instance mask proposal respectively. We show that the frustum from the instance mask with no tail makes the 3D point cloud segmentation easier and results in a better 3D box estimate.

Table 1: 2D proposal evaluation. AP_{2D} performance on KITTI **val** set for car category at IoU = 0.5 / 0.7.

Proposal Type	AP_{2D} (in %), **IoU = 0.5 / 0.7**		
	Easy	Moderate	Hard
Bounding Box	97.2 / 96.5	97.3 / 90.3	90.0 / 87.6
Instance Mask	96.0 / 87.6	89.6 / 75.7	80.3 / 59.4

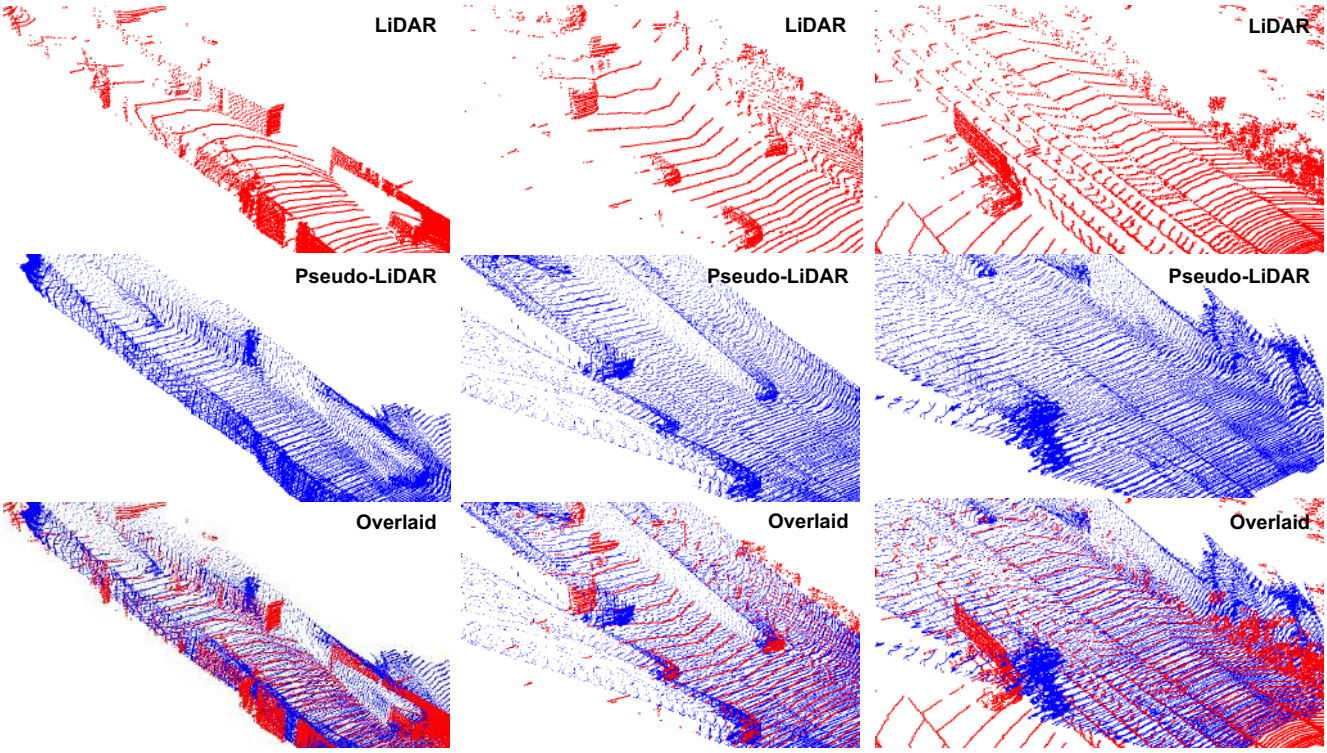


Figure 2: Additional visual comparison between the **LiDAR** (top), **pseudo-LiDAR** (middle) and an overlaid version (bottom).

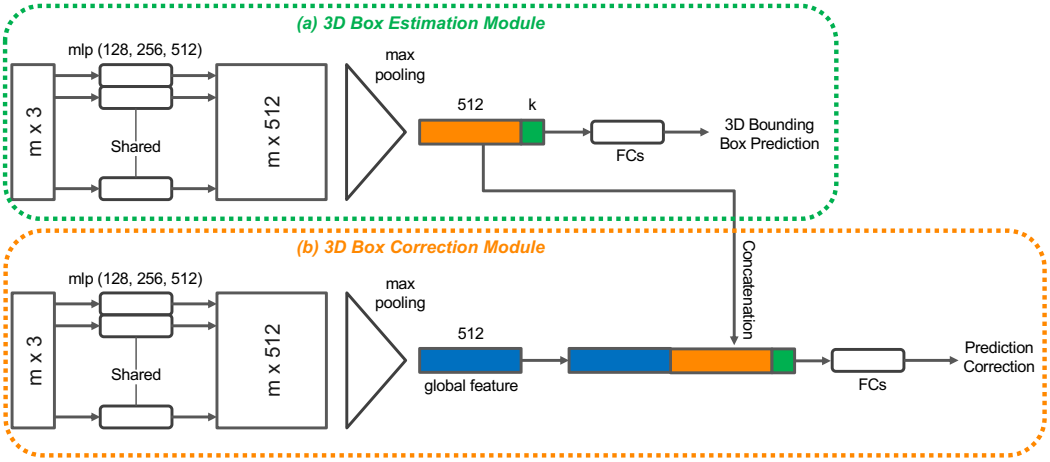


Figure 3: **Network Architecture of the 3D Box Estimation and 3D Box Correction Module.** Both modules take the point cloud as the input. The 3D box estimation module outputs the full 3D box parameters and the 3D box correction module outputs the residual of the parameters. k is the number of class (*e.g.*, 3 in KITTI). The length-k vector (in **green**) is a one-hot vector denotes which class the input point cloud belongs to. mlp denotes the multi-layer perceptron.

posal performs worse than the 2D bounding box proposal due to the lack of the pixel-level instance segmentation annotation on KITTI (only 200 images annotated with instance masks compared to 7500 images annotated with bounding boxes). However, the performance of the bird eye view and 3D object detection when using these 2D mask proposals is surprisingly higher than when using 2D bounding box proposals, which is shown in Ours (baseline) and Ours+Mask of Table 3 in the main paper. This further strengthens the effectiveness of using the instance mask proposal for 3D box estimation, *i.e.* detecting the 3D bounding box from the frustum with no tail is much easier.

3. Amodal 3D Object Detection

Network Architecture of the 3D Box Correction Module

We show the network architecture in Figure 3. Similar to the 3D box estimation module proposed in [2], we also use a PointNet-based network for our 3D box correction module.

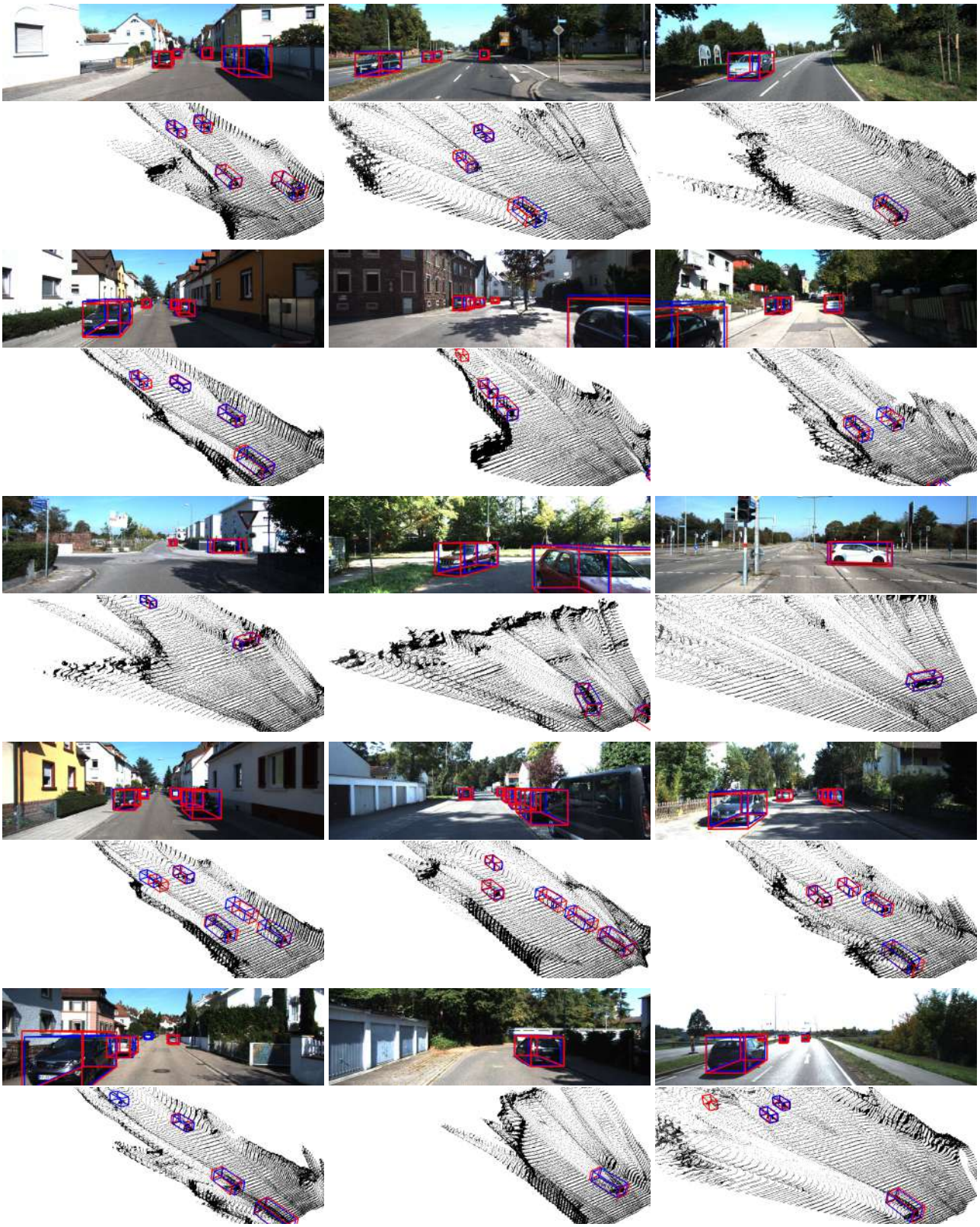


Figure 4: Additional qualitative results of our method on KITTI **val** set. We visualize our 3D bounding box estimate (in **blue**) and ground truth (in **red**) on the frontal images (1st and 3rd rows) and pseudo-LiDAR point cloud (2nd and 4th rows).

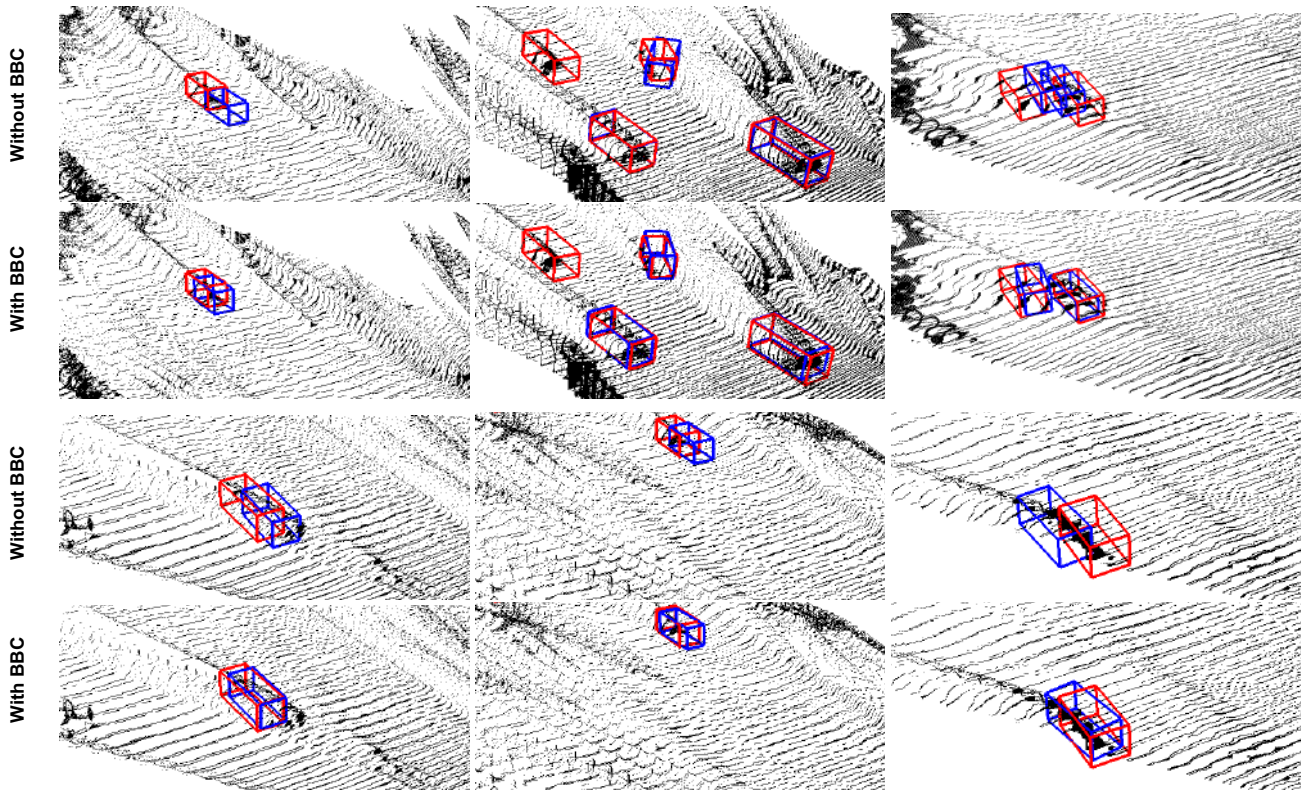


Figure 5: **Additional Visualization about the Effect of Bounding Box Consistency (BBC)**. We visualize our 3D bounding box estimate (**blue**) without BBC (in 1st and 3rd rows) and with BBC (in 2nd and 4th rows). Ground truth is shown in **red**. We show that using the bounding box consistency improves the 3D IoU between the 3D box estimate and the ground truth.

The major difference is that the 3D box estimation module predicts the 3D box parameters while our 3D box correction module outputs the correction to the prediction (*i.e.*, residual of the parameters). In addition, we concatenate the features extracted from the 3D box estimation module with the global feature extracted from the 3D box correction module for predicting the residual of the parameters.

4. 2D-3D Bounding Box Consistency (BBC)

Additional Visualization about the Effect of Bounding Box Consistency

We provide the extra visual comparison between the 3D bounding box estimate with and without using the BBC in Figure 5. The 3D bounding box results shown in the 2nd and 4th rows, which are estimated from the model trained with bounding box consistency loss and post-processed with bounding box consistency optimization, clearly improve the 3D IoU over the 3D bounding box results without using the BBC, shown in the 1st and 3rd rows.

5. Experiments

Additional Visualization of 3D Object Detection Results

We provide additional qualitative results in Figure 4. We show that, from only a single RGB image, the 3D bounding box detection for the car category can be very accurate, even for the challenging faraway objects (*e.g.*, in the 6th row 1st column and 8th row 2nd column of the Figure 4).

References

[1] A. Geiger, P. Lenz, and R. Urtasun. Are We Ready for Autonomous Driving? the KITTI Vision Benchmark Suite. *CVPR*, 2012. 1

[2] C. R. Qi, W. Liu, C. Wu, H. Su, and L. J. Guibas. Frustum PointNets for 3D Object Detection from RGB-D Data. *CVPR*, 2018. 2