# Point2Seq: Detecting 3D Objects as Sequences
## *–Supplementary Material–*

## 1. Details of the Spatial Sampler.

In this section, we will formulate the spatial sampler $\mathcal{S}$ in detail. We formulate the $\mathcal{S}(W^R, W^L)$, $\mathcal{S}(W^R, W^L, W^O)$, and $\mathcal{S}(W^R, W^L, W^O, W^S)$ as follows:

$$\left[p_1^1\right] = \mathcal{S}(W^R, W^L) = \begin{bmatrix} R_x + L_x R_l & R_y + L_y R_w \end{bmatrix}, \tag{1}$$

where $[R_x, R_y]$ is the BEV center coordinate of the region from $W^O$, $[R_l, R_w]$ are the additional parameters introduced to describe the spatial range of the region on the BEV feature map, and $L_x, L_y$ are the predicted offset from $W^L = [L_x, L_y, z] \in \mathbb{R}^3$.

$$\begin{aligned} \begin{bmatrix} p_1^2 \\ p_2^2 \\ p_3^2 \\ p_4^2 \end{bmatrix} = \mathcal{S}(W^R, W^L, W^O) &= \frac{1}{2} \begin{bmatrix} l_d & 0 \\ 0 & w_d \\ -l_d & 0 \\ 0 & -w_d \end{bmatrix} \begin{bmatrix} \sin\theta & \cos\theta \\ \cos\theta & -\sin\theta \end{bmatrix} + \begin{bmatrix} p_1^1 \\ p_1^1 \\ p_1^1 \\ p_1^1 \end{bmatrix} \\ &= \begin{bmatrix} \frac{1}{2} l_d \sin\theta + R_x + L_x R_l & \frac{1}{2} l_d \cos\theta + R_y + L_y R_w \\ \frac{1}{2} w_d \cos\theta + R_x + L_x R_l & -\frac{1}{2} w_d \sin\theta + R_y + L_y R_w \\ -\frac{1}{2} l_d \sin\theta + R_x + L_x R_l & -\frac{1}{2} l_d \cos\theta + R_y + L_y R_w \\ -\frac{1}{2} w_d \cos\theta + R_x + L_x R_l & \frac{1}{2} w_d \sin\theta + R_y + L_y R_w \end{bmatrix} \end{aligned} \tag{2}$$

where $l_d$ and $w_d$ are respectively the average length and width of the ground truth boxes from the training set, and $\theta$ is the rotation angle of the predicted box decoded from $W^O = [sin(\theta), cos(\theta)] \in \mathbb{R}^2$.

$$\begin{aligned} \begin{bmatrix} p_1^3 \\ p_2^3 \\ p_3^3 \\ p_4^3 \end{bmatrix} = \mathcal{S}(W^R, W^L, W^O, W^S) &= \frac{1}{2} \begin{bmatrix} l & 0 \\ 0 & w \\ -l & 0 \\ 0 & -w \end{bmatrix} \begin{bmatrix} \sin\theta & \cos\theta \\ \cos\theta & -\sin\theta \end{bmatrix} + \begin{bmatrix} p_1^1 \\ p_1^1 \\ p_1^1 \\ p_1^1 \end{bmatrix} \\ &= \begin{bmatrix} \frac{1}{2} l \sin\theta + R_x + L_x R_l & \frac{1}{2} l \cos\theta + R_y + L_y R_w \\ \frac{1}{2} w \cos\theta + R_x + L_x R_l & -\frac{1}{2} w \sin\theta + R_y + L_y R_w \\ -\frac{1}{2} l \sin\theta + R_x + L_x R_l & -\frac{1}{2} l \cos\theta + R_y + L_y R_w \\ -\frac{1}{2} w \cos\theta + R_x + L_x R_l & \frac{1}{2} w \sin\theta + R_y + L_y R_w \end{bmatrix} \end{aligned} \tag{3}$$

where $l$ and $w$ are respectively the length and width of the predicted box decoded from $W^S = [log(l), log(w), log(h)] \in \mathbb{R}^3$.

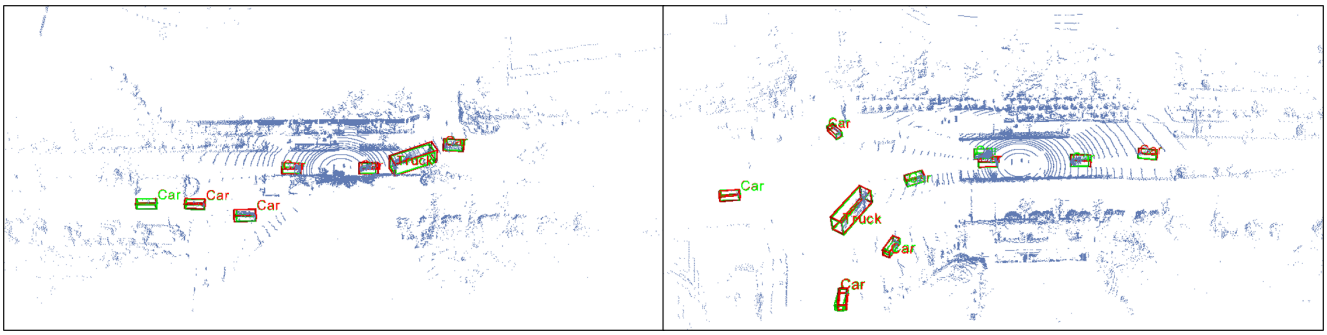


Figure 1. Visualization of detection results on the ONCE dataset. Red boxes are the ground truth boxes, and green boxes are the boxes predicted by Point2Seq

## 2. Additional Experiment.

In this section, we introduce the additional experiment we have conducted to evaluate Point2Seq on the commonly-used Waymo Open Dataset [8] and the ONCE dataset [4].

### 2.1. Comparison on the Waymo Open Dataset.

Table 1 and Table 2 show the detection results of the two extra classes on the Waymo validation set. For pedestrian detection, our method attains 78.33% LEVEL 1 mAP and 69.21% LEVEL 2 mAP. Moreover, for cyclist detection, our method attains 72.53% LEVEL 1 mAP and 71.29% LEVEL 2 mAP. Switching from the anchor and center head to our Point2Seq provides 7.21% and 2.19% LEVEL 1 mAP improvements on pedestrian detection and 10.51% and 1.26% LEVEL 1 mAP improvements on cyclist detection, respectively. Our approach outperforms those time-consuming two-stage 3D detectors [3, 6, 7] on all classes, which further indicates the effectiveness of the scene-to-sequence decoder.

### 2.2. Comparison on the ONCE Dataset.

Table 3 shows the detection results of Point2Seq on the ONCE dataset test split. As can be observed, our Point2Seq attains the state-of-the-art results on all classes on the ONCE dataset test split, with 73.17% mAP for vehicle detection, 56.62% mAP for pedestrian detection, and 69.72% for cyclist detection. The overall mAP of our approach is 66.50%, 5.26% higher than the center-based 3D object detector [12] and 14.60% higher than the anchor-based 3D object detector [11]. The observations on the ONCE dataset are consistent with previous experiments.

### 2.3. Comparison between Sequential word decoding and multi-stage refinements..

From Figure **??** we can see that (1) simple multi-step box refinements cannot outperform our proposed method. The best performance 77.11% comes from 1-step refinement, but it is still lower than the Point2Seq baseline (77.52%). (2) Constantly sampling features to refine boxes cannot yield consistent performance boosts, mAP drops from 77.11% to 76.39% when increasing the refinement steps. Those results indicate that the performance gain of our method is not simply brought by refinements. Predicting objects as sequences and encoding the sequential relationships of object words by updating the hidden state feature map are also significant, and bring considerable performance gain.

## 3. Qualitative Results.

In this section, we provide qualitative results on the ONCE dataset in Figure 1 and the Waymo Open Dataset in Figure 2. The figures show that our proposed Point2Seq can accurately detect 3D objects without post-processing like NMS. We also provide a video sequence for more qualitative results.

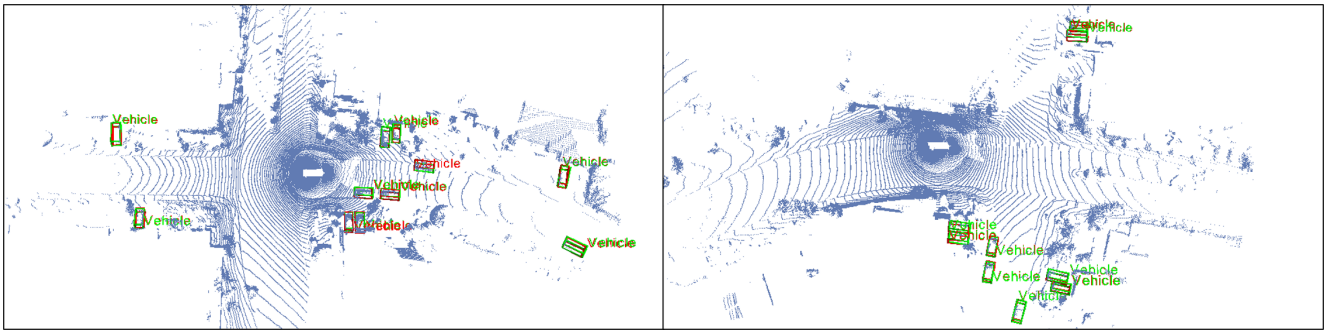


Figure 2. Visualization of detection results on the Waymo Open Dataset. Red boxes are the ground truth boxes, and green boxes are the boxes predicted by Point2Seq

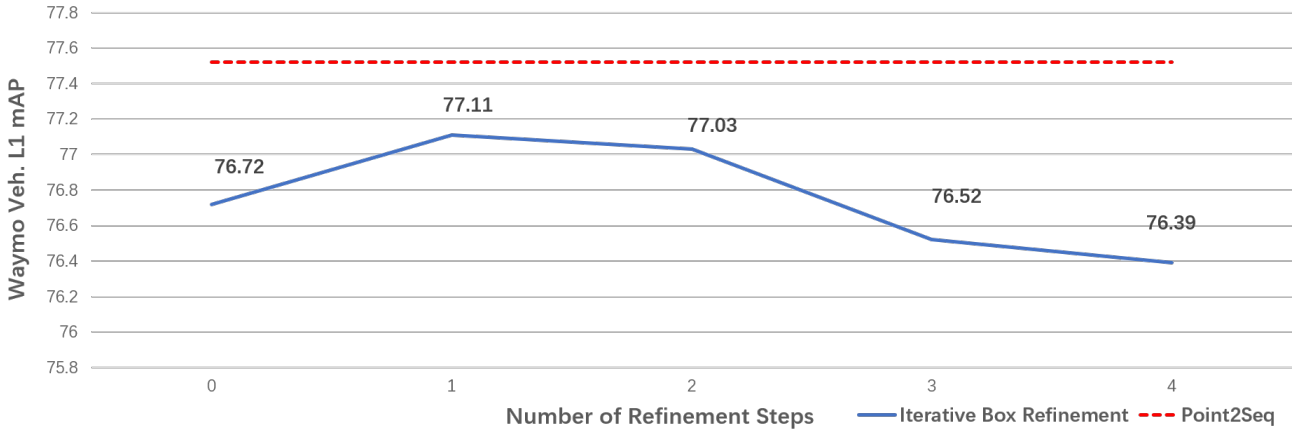


Figure 3. Sequential word decoding vs. multi-stage refinement

| Method | Backbone | Head | Pedestrian LEVEL 1 | | Pedestrian LEVEL 2 | |
|---|---|---|---|---|---|---|
| | | | 3D mAP(%) | 3D mAPH(%) | 3D mAP(%) | 3D mAPH(%) |
| LaserNet [5] | Range | Anchor | 63.4 | 73.47 | 61.55 | 42.69 |
| RangeDet [1] | Range | Center | 75.94 | - | - | - |
| RSN [9] | Range | Center | 77.8 | 72.7 | 68.3 | 63.7 |
| Pillar-OD [10] | Pillar | Anchor | 72.51 | - | - | - |
| MVF [13] | Voxel | Anchor | - | - | 65.33 | - |
| PV-RCNN [7] | Voxel | Anchor | 75.01 | 65.65 | 66.04 | 57.61 |
| CenterPoints [12] | Voxel | Center | 79.0 | 72.9 | 71.0 | 65.3 |
| SECOND† [11] | Voxel | Anchor | 71.12 | 61.28 | 63.66 | 54.42 |
| CenterPoints† [12] | Voxel | Center | 76.14 | 70.00 | 68.32 | 62.67 |
| Point2Seq (Ours) | **Voxel** | **Sequence** | **78.33** | **72.81** | **69.21** | **64.19** |

Table 1. Performance comparison on the Waymo Open Dataset with 202 validation sequences for pedestrian detection. †: re-implemented using the official code. Point2Seq maintains the same backbone, data augmentations, and training epochs with the re-implemented baselines.

| Method | Backbone | Head | Cyclist LEVEL 1 | | Cyclist LEVEL 2 | |
|---|---|---|---|---|---|---|
| | | | 3D mAP(%) | 3D mAPH(%) | 3D mAP(%) | 3D mAPH(%) |
| PV-RCNN [7] | Voxel | Anchor | 67.81 | 66.35 | 65.39 | 63.98 |
| CenterPoints [12] | Voxel | Center | - | - | - | 68.61 |
| SECOND† [11] | Voxel | Anchor | 62.02 | 60.58 | 59.62 | 58.34 |
| CenterPoints† [12] | Voxel | Center | 71.27 | 70.13 | 68.72 | 67.63 |
| Point2Seq (Ours) | **Voxel** | **Sequence** | **72.53** | **71.29** | **70.25** | **69.04** |

Table 2. Performance comparison on the Waymo Open Dataset with 202 validation sequences for cyclist detection. †: re-implemented using the official code. Point2Seq maintains the same backbone, data augmentations, and training epochs with the re-implemented baselines.

| Method | mAP(%) | Vehicle mAP(%) | | | | Pedestrian mAP(%) | | | | Cyclist mAP(%) | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | overall | 0-30m | 30-50m | 50m-inf | overall | 0-30m | 30-50m | 50m-inf | overall | 0-30m | 30-50m | 50m-inf |
| PointRCNN [7] | 28.74 | 52.00 | 74.44 | 40.72 | 22.14 | 8.73 | 12.20 | 6.96 | 2.96 | 34.02 | 46.48 | 27.39 | 11.45 |
| PointPillars [2] | 45.47 | 69.52 | 84.51 | 60.55 | 45.72 | 17.28 | 20.21 | 15.06 | 11.48 | 49.63 | 60.15 | 42.43 | 27.73 |
| PV-RCNN [7] | 53.85 | 76.98 | 89.89 | 69.35 | 55.52 | 22.66 | 27.23 | 21.28 | 12.08 | 61.93 | 72.13 | 56.64 | 37.23 |
| SECOND [11] | 51.90 | 69.71 | 86.96 | 60.22 | 43.02 | 26.09 | 30.52 | 24.63 | 14.19 | 59.92 | 70.54 | 54.89 | 34.34 |
| CenterPoints [12] | 61.24 | 66.35 | 83.65 | 56.74 | 41.57 | 51.80 | 62.80 | 45.41 | 24.53 | 65.57 | 73.02 | 62.85 | 44.77 |
| Point2Seq (Ours) | **66.50** | **73.17** | **87.85** | **63.50** | **50.31** | **56.62** | **68.80** | **50.65** | **26.37** | **69.72** | **78.24** | **65.95** | **47.19** |

Table 3. Performance comparison on the ONCE dataset test split. Point2Seq maintains the same backbone architecture and training configurations with the baselines on the ONCE benchmark.

## References


[1] Lue Fan, Xuan Xiong, Feng Wang, Naiyan Wang, and Zhaoxiang Zhang. Rangedet: In defense of range view for lidar-based 3d object detection. *arXiv preprint arXiv:2103.10039*, 2021. 3

[2] Alex H Lang, Sourabh Vora, Holger Caesar, Lubing Zhou, Jiong Yang, and Oscar Beijbom. Pointpillars: Fast encoders for object detection from point clouds. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12697–12705, 2019. 4

[3] Jiageng Mao, Minzhe Niu, Haoyue Bai, Xiaodan Liang, Hang Xu, and Chunjing Xu. Pyramid r-cnn: Towards better performance and adaptability for 3d object detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2723–2732, 2021. 2

[4] Jiageng Mao, Minzhe Niu, Chenhan Jiang, Hanxue Liang, Jingheng Chen, Xiaodan Liang, Yamin Li, Chaoqiang Ye, Wei Zhang, Zhenguo Li, et al. One million scenes for autonomous driving: Once dataset. *arXiv preprint arXiv:2106.11037*, 2021. 2

[5] Gregory P. Meyer, Ankita Gajanan Laddha, Eric Kee, Carlos Vallespi-Gonzalez, and Carl K. Wellington. Lasernet: An efficient probabilistic 3d object detector for autonomous driving. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 12669–12678, 2019. 3

[6] Hualian Sheng, Sijia Cai, Yuan Liu, Bing Deng, Jianqiang Huang, Xian-Sheng Hua, and Min-Jian Zhao. Improving 3d object detection with channel-wise transformer. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 2743–2752, October 2021. 2

[7] 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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10529–10538, 2020. 2, 3, 4

[8] Pei Sun, Henrik Kretzschmar, Xerxes Dotiwalla, Aurelien Chouard, Vijaysai Patnaik, Paul Tsui, James Guo, Yin Zhou, Yuning Chai, Benjamin Caine, et al. Scalability in perception for autonomous driving: Waymo open dataset. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2446–2454, 2020. 2

[9] Pei Sun, Weiyue Wang, Yuning Chai, Gamaleldin Elsayed, Alex Bewley, Xiao Zhang, Cristian Sminchisescu, and Dragomir Anguelov. Rsn: Range sparse net for efficient, accurate lidar 3d object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5725–5734, 2021. 3

[10] Yue Wang, Alireza Fathi, Abhijit Kundu, David A Ross, Caroline Pantofaru, Tom Funkhouser, and Justin Solomon. Pillar-based object detection for autonomous driving. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXII 16*, pages 18–34. Springer, 2020. 3

[11] Yan Yan, Yuxing Mao, and Bo Li. Second: Sparsely embedded convolutional detection. *Sensors*, 18(10):3337, 2018. 2, 3, 4

[12] Tianwei Yin, Xingyi Zhou, and Philipp Krahenbuhl. Center-based 3d object detection and tracking. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11784–11793, 2021. 2, 3, 4

[13] Yin Zhou, Pei Sun, Yu Zhang, Dragomir Anguelov, Jiyang Gao, Tom Ouyang, James Guo, Jiquan Ngiam, and Vijay Vasudevan. End-to-end multi-view fusion for 3d object detection in lidar point clouds. In *Conference on Robot Learning*, pages 923–932. PMLR, 2020. 3