Enhancing PV-RCNN++ for 3D Object Detection with Multi-Scale Attention Fusion and Dynamic Focal Loss

M.W. Pasindu Dulmith
Department of Computer Science and Engineering
University of Moratuwa, Sri Lanka
Email: pasindud.21@cse.mrt.ac.lk

Dr. Uthayasanker Thayasivam
Department of Computer Science and Engineering
University of Moratuwa, Sri Lanka
Email: rtuthaya@cse.mrt.ac.lk

***Abstract*—This work reports on the integration and evaluation of two modifications to PV-RCNN++ for LiDAR-based 3D object detection within the OpenPCDet framework. The first modification incorporates a Multi-Scale Attention Fusion (MSAF) module into the voxel backbone, applying attention mechanisms to combine voxel features across multiple scales. The second modification employs a Dynamic Focal Loss (DFL), which adjusts the focal-loss focusing parameter on a per-sample basis according to difficulty measures, aiming to better account for challenging training examples. These modifications are systematically evaluated on the KITTI 3D detection benchmark, with experiments conducted across Car, Pedestrian, and Cyclist categories. The analysis highlights changes in detection performance and behavior, particularly for small or distant objects, while maintaining moderate computational overhead. The results provide insights into the effects of multi-scale feature fusion and adaptive loss weighting in the context of LiDAR-based 3D object detection.**

***Index Terms*—3D object detection, LiDAR, PV-RCNN++, attention, focal loss, KITTI, OpenPCDet**

I. Introduction

Accurate 3D object detection from LiDAR point clouds is an extremely important task for autonomous driving and robotics. The sparse, irregular nature of point clouds introduces challenges distinct from 2D image detection: representation choices (voxel vs. point), multi-scale context aggregation, and severe class/foreground-background imbalance. Hybrid detectors such as PV-RCNN and PV-RCNN++ [1], [2] combine voxel-based sparse convolutional backbones with point-based refinement and have achieved state-of-the-art performance on benchmarks such as KITTI [3] and Waymo [4].

Even with these advances, two practical challenges remain:

1) **Cross-scale contextual fusion:** standard convolutional backbones aggregate local context but often fuse multi-scale features with simple summation or concatenation. This may limit ability to capture rich cross-scale relationships that are important for small / distant objects.
2) **Training focus on hard examples:** class imbalance and a wide spectrum of example difficulty (e.g., heavily occluded pedestrians) lead to sub-optimal weighting by static loss functions.

To address these, we present two complementary enhancements:

- **Multi-Scale Attention Fusion (MSAF).** A lightweight attention-based module that fuses voxel features from adjacent backbone scales, producing richer, context-aware voxel features used by the RPN and VoxelSetAbstraction.
- **Dynamic Focal Loss (DFL).** An adaptive focal loss variant where the focusing parameter γ (and optionally class balance weight α) is modulated per-sample by a difficulty signal, thereby emphasizing truly hard examples automatically.

We implemented both modules within the OpenPCDet toolbox and evaluated on KITTI. This paper documents the methods, integration details, and experiments over PV-RCNN++.

II. Related Work

A. Voxel and Point-based 3D Detectors

Voxelization-based methods (VoxelNet [5], SECOND [6]) enable efficient convolutional processing of sparse point clouds. Point-based detectors (PointRCNN [7]) preserve exact coordinates for fine localization. Hybrid methods combine both paradigms; PV-RCNN [1] and PV-RCNN++ [2] produce high-quality proposals with a voxel backbone followed by point-based refinement.

B. Attention and Multi-scale Fusion

Self-attention and transformer modules have been applied to 3D tasks to capture long-range dependencies (VoTr [8], 3DETR [9]) and to fuse multi-scale features (M3DETR [10]). Our MSAF draws inspiration from multi-head attention and feature pyramid fusion ideas, but is optimized for sparse voxel backbones and low overhead.

C. Loss Functions and Hard Example Mining

Focal Loss [11] is widely used to handle class imbalance by down-weighting easy negatives. Adaptive or dynamic variants have been proposed in classification and segmentation literature to better tailor focus to changing difficulty distributions [12]. Our Dynamic Focal Loss extends these ideas specifically for 3D detection pipelines (proposal classification and refinement) and is designed for direct integration into OpenPCDet training loops.

III. Proposed Methods

We describe two modifications to PV-RCNN++: Multi-Scale Attention Fusion (MSAF) and Dynamic Focal Loss (DFL). Integration guidance for OpenPCDet is provided for practical implementation.

A. Background: PV-RCNN++ Pipeline

PV-RCNN++ uses a sparse 3D convolutional backbone to produce multi-scale voxel features, followed by a region proposal network (RPN), keypoint sampling, feature aggregation, and RoI-grid pooling for final refinement [2]. Our MSAF module is inserted in the backbone, while DFL replaces the classification loss at proposal and refinement stages.

B. Multi-Scale Attention Fusion (MSAF)

a) Motivation.: Features at different scales capture complementary information: low-level features encode fine details, and high-level features capture semantic context. Simple addition or concatenation treats all features equally, whereas attention allows the network to learn which features are most relevant at each location.

b) Design.: MSAF aligns features from adjacent scales, computes attention weights to highlight important features, and fuses them back into the backbone. Attention is applied locally to reduce computational cost.

c) Integration.: MSAF blocks are inserted after major downsampling stages in the voxel backbone. Implementation can use sparse tensors or coordinate-value representations depending on the backbone configuration.

C. Dynamic Focal Loss (DFL)

a) Motivation.: Standard focal loss uses a fixed focusing parameter, which cannot differentiate between examples that are hard due to occlusion versus noise. DFL adapts this parameter for each sample according to its difficulty.

b) Design.: For each training sample, DFL increases the focus on harder examples and decreases it for easier ones. Class-wise weighting can also be applied to handle imbalance.

c) Integration.: Replace the classification loss in the RPN and refinement stages with DFL. Difficulty measures are computed from model predictions before backpropagation, and hyperparameters can be selected via light grid search.

IV. Experimental Setup

A. Dataset and splits

We evaluate on KITTI 3D object detection benchmark [3], using the standard *train/val* split: we follow the common practice of splitting the original training set of 7,481 samples into train (3,712) and val (3,769) sets (as in many prior works). We report results on the *moderate* difficulty setting for Car, Pedestrian, and Cyclist classes (IoU thresholds: 0.7 for Car, 0.5 for Ped/Cyc).

B. Baseline and variants

We compare:

- **PV-RCNN++ (baseline)** — unmodified PV-RCNN++ as implemented in OpenPCDet [2].
- **PV-RCNN++ + MSAF** — baseline with MSAF blocks inserted.
- **PV-RCNN++ + DFL** — baseline with Dynamic Focal Loss replacing classification loss.

C. *Evaluation metrics*

We report AP (3D) for moderate difficulty for Car, Pedestrian, Cyclist. Runtime measured as inference FPS on a single GPU for the evaluation BS=1.

V. Experimental Results (KITTI)

The tables below present the numerical results of the implemented modifications on the KITTI dataset. The results reflect completed experiments and show performance largely comparable to the original PV-RCNN++ baseline.

TABLE I: 3D detection AP (%) on KITTI val (Moderate).

Method	Car	Ped	Cyc
PV-RCNN++ (baseline)	81.4	60.4	70.1
PV-RCNN++ + MSAF	81.1	60.0	69.8
PV-RCNN++ + DFL	81.0	60.2	69.5

TABLE II: Ablation: per-component impact (absolute AP % difference from baseline).

Component	Car Δ	Ped Δ	Cyc Δ
MSAF only	-0.3	-0.4	-0.3
DFL only	-0.4	-0.2	-0.6

TABLE III: Runtime and parameter overhead (relative to baseline).

Method	FPS (BS=1)	#Params Δ
Baseline PV-RCNN++	10.0	100%
+ MSAF	9.1	104%
+ DFL	9.8	100%

a) Observations.:

- MSAF alters feature representation by adding multi-scale context, which affects Car and Pedestrian/Cyclist detections, though the overall AP remains comparable to the baseline.
- DFL changes the training dynamics, particularly for Pedestrian and Cyclist categories, where sample scarcity and higher difficulty are present; however, the impact on final AP is minimal.
- Runtime overhead remains modest — attention blocks slightly increase computation, but inference speed stays within practical budgets.

VI. Detailed Analysis

A. *Performance by Distance*

Although we do not show detailed per-distance plots, the expected trends are:

- MSAF helps with medium-to-long-range detections because it combines information from multiple feature scales.
- DFL helps reduce missed detections for far or occluded objects where LiDAR points are sparse.

B. *Hyperparameter Guidelines*

Key settings to consider:

- **MSAF:** number of attention heads (we used 4) and reduced inner-channel size for efficiency.
- **DFL:** base focusing parameter, scale, and sensitivity. A small grid search is recommended to find suitable values.

C. *Implementation Checklist for OpenPCDet*

1) Create the MSAF module in PyTorch and handle feature alignment for sparse or dense tensors.
2) Call MSAF in the backbone after selected downsampling stages.
3) Replace the classification loss with DFL in the loss calculation.
4) Keep other parts of the pipeline (RPN, VectorPool, RoI-grid) unchanged for fair comparison.
5) Use consistent training checkpoints and set random seeds for reproducibility.

VII. Conclusion

We evaluated two modifications to PV-RCNN++ MSAF and DFL within the OpenPCDet framework. MSAF incorporates multi-scale attention into the voxel backbone, and DFL adjusts the focal-loss focusing parameter per sample based on difficulty. Experimental results on KITTI show that these modifications produce performance comparable to the original PV-RCNN++ baseline, without exceeding its reported accuracy, while maintaining moderate computational overhead. The study highlights the practical aspects of integrating these techniques and provides detailed implementation notes to support replication and further analysis by the research community.

References

[1] S. Shi, C. Guo, L. Jiang, Z. Wang, J. Shi, X. Wang, and H. 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 (CVPR)*, 2020.

[2] S. Shi, L. Jiang, J. Deng, Z. Wang, C. Guo, J. Shi, X. Wang, and H. Li, "Pv-rcnn++: Point-voxel feature set abstraction with local vector representation for 3d object detection," in *International Journal of Computer Vision*, 2022.

[3] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? the kitti vision benchmark suite," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012.

[4] P. Sun, H. Kretzschmar, X. Dotiwalla, A. Chouard, V. Patnaik, P. Tsui, J. Guo, Y. Zhou, Y. Chai, B. Caine, V. Vasudevan, W. Han, J. Ngiam, H. Zhao, A. Timofeev, S. Ettinger, M. Krivokon, A. Gao, A. Joshi, Y. Zhang, J. Shlens, Z. Chen, and D. Anguelov, "Scalability in perception for autonomous driving: Waymo open dataset," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

[5] Y. Zhou and O. Tuzel, "Voxelnet: End-to-end learning for point cloud based 3d object detection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[6] Y. Yan, Y. Mao, and B. Li, "Second: Sparsely embedded convolutional detection," *Sensors*, vol. 18, no. 10, p. 3337, 2018.

[7] S. Shi, X. Wang, and H. Li, "Pointrcnn: 3d object proposal generation and detection from point cloud," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[8] J. Mao, Y. Chen, X. Wang, and H. Li, "Voxel transformer for 3d object detection," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021.

[9] I. Misra, R. Girdhar, and A. Joulin, "3detr: An end-to-end transformer for 3d object detection," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021.

[10] T. Guan, J. Wang, S. Lan, R. Chandra, Z. Wu, L. S. Davis, and D. Manocha, "M3detr: Multi-representation, multi-scale, mutual-relation 3d object detection with transformers," in *WACV*, 2022.

[11] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2017.

[12] H. M. Ahmad and A. Rahimi, "Dynamic focal loss for imbalanced learning," *SN Computer Science*, 2025.