2024 18th International Conference on
Control, Automation, Robotics and Vision (ICARCV)
December 12-15, 2024. Dubai

CP-RCNN: Lidar Object Detection with Feature Pooling and Abstraction

Hang Zhou[1] and Yiding Ji[1*]

***Abstract*— LiDAR-based object detection is a challenging task for autonomous navigation systems, especially in pedestrian-rich environments. Recently, the integration of deep learning techniques with lidar-generated point cloud data has advanced object detection and segmentation in many scenarios. However, current lidar-based methods usually struggle to accurately detect small-sized objects, such as pedestrian and cyclist, causing severe safety and reliability concerns for autonomous vehicles. This study refines structural design of lidar based neural networks to enhance precision and recall metrics for the identification of small entities. Specifically, we introduce CP-RCNN, a novel lidar object detection framework that combines state of the art voxelization and feature extraction techniques. Extensive ablation experiments demonstrate that our method has improved performance in the detection of pedestrians and cyclists. Furthermore, this paper also proposes a novel neural network structure named Centerpoint-RCNN, which not only maintains high precision in vehicle classification but also achieves an impressive inference speed of 15Hz on the NVIDIA RTX 4090 graphics processing unit.**

I. INTRODUCTION

In contrast to image-based deep neural network architectures, point cloud data also imposes challenges for the direct application of Convolution Neural Networks (CNNs) [1] due to its unstructured nature. Furthermore, the challenge of detecting small objects like pedestrians and cyclists is magnified by the sparse representation of these entities in the point cloud data. Typically, a pedestrian may be represented by as few as 20 points, a number that can decrease to less than five in scenarios involving significant distance (over 50 meters) or occlusion, complicating the Lidar-based detection of small objects. Current algorithms face challenges in achieving a balance between high accuracy and efficient inference speeds in detecting these entities.

Based on their prepossessing approaches, deep neural network methods for processing point cloud data can be roughly categorized as: Voxelization [2] [3], Point-based [4] [5], Depth View methods [6] [7], and Transformer [8]. Recent research efforts have concentrated on integrating these strategies to create more comprehensive feature maps. For example, PV-RCNN [9] merges voxel-based and point-based methods, with the latter providing multi-scale semantic feature maps and object proposals for the detection process of the former. Despite its slower inference speed, PV-RCNN has shown unmatched accuracy. In contrast, Voxel R-CNN [10] improves inference speeds significantly by eliminating the point-based feature extractor found in PV-RCNN and directly sampling features from the 3D voxel feature map.

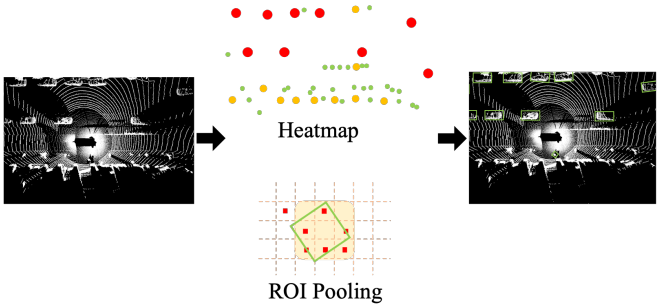

Fig. 1. General Articture of CP-RCNN that refines headtmap with region-of-interest pooling. The proposed methods is able to distinguish pedestrian from its similar-shaped true-negative neighbor.

Detecting small objects like pedestrians and cyclists using LiDAR technology poses distinct challenges, mainly stemming from the sparse nature of the generated point cloud data. Lidar sensors emit laser beams that bounce off surfaces to calculate distances, but when it comes to smaller objects, fewer points of reflection result in less data. This sparsity becomes more prominent as the distance grows, complicating the precise capture of details for smaller objects in the surroundings. As a result, pedestrians and cyclists, being significantly smaller than vehicles and structures, are frequently underrepresented in point cloud data, leading to challenges in their accurate detection and classification.

Furthermore, the difficulty in distinguishing small objects like pedestrians and cyclists from other urban fixtures such as poles, trees, and bins complicates their detection. These objects can share similar sizes and shapes, leading to confusion in the LiDAR-generated point cloud where they may appear indistinguishable. This similarity poses a significant challenge for algorithms tasked with interpreting the data, as accurately distinguishing between a pedestrian and a nearby pole becomes a complex task. Improvements in LiDAR resolution and sophisticated data processing techniques are crucial for enhancing the detection and classification of these small, yet critical, elements in urban environments. Despite recent algorithms [11] [9] [12] significantly improves small-size object detection, the overall accuracy and recall rate are

[1]Robotics and Autonomous Systems Thrust, Systems Hub, The Hong Kong University of Science and Technology (Guangzhou), Guangzhou, China hzhou269@connect.hkust-gz.edu.cn; jiyiding@hkust-gz.edu.cn

This work is funded by National Natural Science Foundation of China under grants 62303389 and 62373289, Guangdong Basic and Applied Basic Research Funding under grants 2024A1515012586 and 2022A151511076, Guangzhou Basic and Applied Basic Research Scheme under grant 2023A04J1067 and Guangzhou Municipality-University Joint Funding under grants 2024A03J0618 and 2023A03J0678.

*Corresponding Author

still significantly lower than vehicle class.

To address the aforementioned challenges, we introduce a lidar-based neural network framework for object detection called CP-RCNN that fulfills joint feature pooling and abstraction. The main contributions of this work are as follows:

- Our method applies a voxel-based lidar detection architecture which adds heatmap and ROI pooling to sample 3D structures from the feature map to improve the performance for small object detection.
- Our network uses a heatmap structure to enhance the detection of small objects, while maintaining a balance between detection accuracy and speed.

II. Related Work

In the growing field of point cloud processing, Deep Neural Network related methods are typically categorized into three main groups: Voxelization [2], [13], Point-based methods [4], [14], [15], and Depth View methods s [6].

The Voxelization approach, exemplified by VoxNet [2] and VoxelNet [16], divides the spatial domain into a grid of voxels and assigns point cloud data to these grids. This technique was improved by Yan et al. [3] with the introduction of SECOND, which used a Sparse Convolution Library to boost accuracy and computational efficiency and has become a key framework for voxel-based point cloud processing.

Simultaneously, the Point-based method saw significant progress with the introduction of PointNet by Qi et al [4]. This novel framework used 1x1 convolution and max pooling operations to directly process raw point cloud data within a neural network. PointNet++ [5] later extended the method to foster the extraction of regional features on different scales by employing the farthest point sampling, enhancing the fidelity of point sampling representations.

The Depth View method, as demonstrated by RangeNet [7], introduces a unique strategy of converting point cloud data into depth images. This conversion enables semantic segmentation, followed by point-cloud reconstruction and post-processing, which successfully transfers segmentation outcomes from depth images back to the point cloud domain.

In the domain of voxel-based detection methods, SECOND [3] is a fundamental framework that includes voxelization, voxel feature extraction, sparse convolution, and the utilization of a Regional Proposal Network (RPN) [17].

As for anchor-free Lidar detection methods, CenterPoint [11] is inspired by CenterNet [18], a method initially designed for image detection. CenterPoint adapts the structure of SECOND, specifically the RPN, to forecast object central points and extract details on object size and category through the regression head. In addition, Voxel R-CNN [10] significantly contributes to the advancement of two-stage, anchor-free, voxel-based methods. It simplifies the PV-RCNN architecture by removing the PointNet feature extraction component. The introduction of Voxel ROI Pooling by the authors of Voxel R-CNN is a novel concept that conducts set abstraction on the 3D feature map within the region proposed by the RPN, demonstrating a notable improvement in the efficiency and accuracy of point cloud processing techniques.

III. Preliminary

The voxelization techniques transform raw and unstructured point cloud data into a structured grid-like representation called voxels. Similar to pixels in 2D imaging, voxels represent values on a regular grid in a 3D space. This process enables traditional neural network architectures, like Convolutional Neural Networks (CNNs), to efficiently process point cloud data in a structured format. Key components in voxelization include the Voxel Feature Extractor (VFE) and the Pillar Feature Extractor (PFE) [19].

The Voxel Feature Extractor (VFE) is tasked with aggregating and encoding point features within each voxel. It groups points based on spatial locations into voxels and then applies operations to extract relevant features from the points within each voxel. These features may include the centroid of points, maximum height, or mean intensity.

The Pillar Feature Extractor (PFE) is a specialized version of VFE that streamlines voxelization by focusing on a single grid in the z-dimension, treating the point cloud as vertical ”pillars.” This method notably simplifies computational complexity by converting the three-dimensional data structure into a pseudo-two-dimensional format. After PFE processing, the data can be analyzed using 2D CNNs, known for efficiently extracting features from structured data.

IV. Methods

This section outlines the CP-RCNN architecture, a dual-stage voxel-centric framework for detecting 3D objects. It integrates 3D and 2D backbone networks, the latter paired with a Region Proposal Network (RPN). It also employs voxel Region of Interest (RoI) pooling alongside a detection subnet for enhancing box precision. Finally, Centerhead is designed for 2D features and RoI pooling aggreation.

A. Voxelization

Both voxelnet backbone and the pillar backbone are used in visualization for different purposes. The voxelnet backbone is suited for handling data processed by the VFE, accommodating the full three-dimensional structure of the voxelized point cloud. In contrast, the pillar backbone is tailored to work with data processed by the PFE, capitalizing on the reduced complexity of the pillar-based representation to facilitate efficient processing.

B. Backbone and RPN

The architectural foundation of the model adopts a structure analogous to that of SECOND, utilizing the Spconv library to facilitate sparse convolution operations. Furthermore, the integration of skip connections, a concept pioneered by the ResNet framework [10], has been incorporated into the backbone architecture to enhance the ease of optimization. This strategic incorporation of skip connections aids in mitigating the vanishing gradient problem, thereby streamlining the training process of deep neural networks.

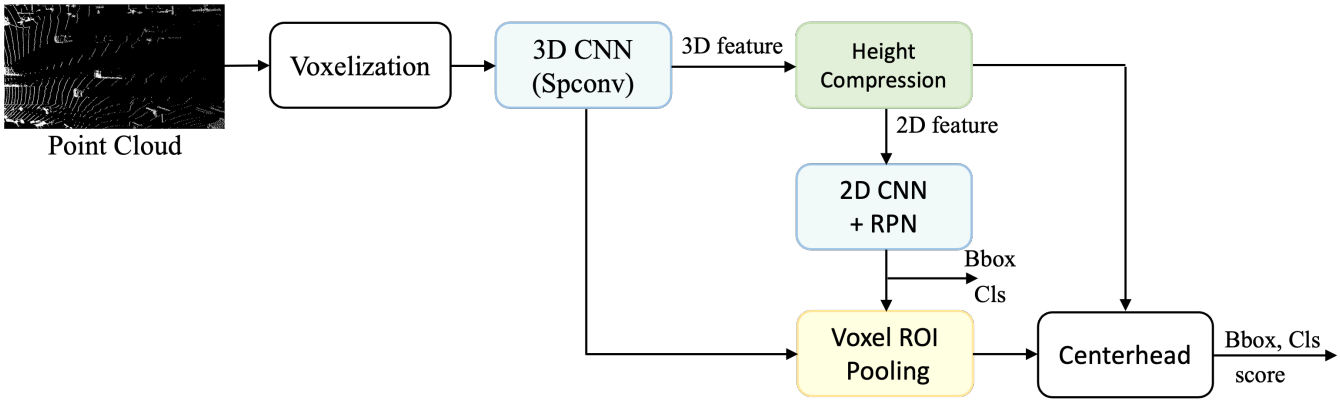

Fig. 2. The Overview of CP-RCNN for Lidar Object Detection. The first part is standard voxelization and 3D covolutional network. The extracted 3D feature are forward to Hight Compression for 2D feature and ROI Pooling. ROI Pooling also takes object proposal from RPN to get richer regional features. Finally, CenterHead takes 2D features to generate heatmap which is later combined with feature map from ROI pooling.

Upon the extraction of sparse point cloud features, the data is subsequently relayed to the Voxel R-CNN module and subjected to a process termed Height Compression. This process is meticulously designed to transmute the sparse point cloud features into a Bird's Eye View (BEV) representation. Initially, a voxel feature tensor, characterized by its dimensions (batch, channel, Density, Height, Width), is extracted from the sparse point cloud features. Following this extraction, a dimensional reconfiguration is performed, wherein the channel dimension is amalgamated with the Density dimension. This manipulation effectively converts the feature representation into a two-dimensional format under the BEV perspective.

The resulting two-dimensional feature map is then expediently utilized as input for subsequent processing stages, namely the Region Proposal Network (RPN) and the CenterHead module. These stages are integral to the model's capability to delineate and identify objects within the point cloud data, leveraging the structured BEV representation to achieve heightened accuracy and efficiency in object detection tasks. Through this sophisticated processing pipeline, the model harnesses the intrinsic spatial information contained within the point cloud data, enabling the precise localization and classification of objects in autonomous navigation systems.

C. Voxel ROI Pooling

This module draws conceptual inspiration from the seminal work presented in the Voxel R-CNN study. Upon the generation of 3D object proposals by the Region Proposal Network (RPN), the network undertakes a meticulous process of feature aggregation applied to these proposals. This involves a selective feature extraction process from each facet of the 3D proposals. Initially, the proposals are segmented into subvoxels, with each subvoxel functioning analogously to a grid within the proposal structure.

In a departure from conventional methodologies that aggregate voxel-wise features within the candidate frame in a straightforward manner, this module employs a technique known as Voxel Query. This technique is pivotal in sourcing the adjacent K voxel features for each grid by leveraging the set Manhattan distance. Consequently, the focal grid point alongside its K neighboring voxels are aggregated into a singular group, facilitating the utilization of the PointNet architecture for the purpose of feature aggregation.

Under the assumption that the feature dimension for each voxel is denoted as C, the module constructs relative position information $((x_p, y_p, z_p))$ for each neighboring voxel in relation to the grid. This relative positional data is then concatenated with the original voxel-wise features, resulting in an augmented feature dimension of (C+3). Following this augmentation, the PointNet architecture proceeds to aggregate these features, ultimately yielding an output feature vector with a dimension of (C').

The Voxel R-CNN module is instrumental in the extraction of regional features across diverse scales, thereby engendering a more enriched regional feature map. This sophisticated approach to feature aggregation not only enhances the granularity of the feature representation but also significantly contributes to the overall efficacy of the object detection process within the point cloud domain.

D. CenterHead

This module adheres to the implementation framework established by CenterPoint, integrating a bespoke CenterHead that processes a feature map output from the Voxel R-CNN alongside an ancillary 2-dimensional (2D) feature map. The latter is instrumental in the generation of a heatmap, which functions as a probabilistic indicator of object presence within the scene. The CenterHead mechanism is designed to accept a 3-channel input image of dimensions (W x H), and it subsequently yields a (K)-channel heatmap with reduced dimensions ((W/R) x (H/R)). Herein, (R) signifies the output stride, effectively diminishing the spatial resolution to emphasize areas of interest, whilst (K) denotes the quantified number of target classes designated for detection. This heatmap embodies a binary schema, with potential values of 0 and 1, where a value of 1 denotes the pixel's classification as the central point of a detection frame, indicative of object localization. Conversely, a value of 0 designates pixels categorized as belonging to the background.

Further, the regression head within this module is allocated the responsibility of cataloging and refining object attributes centered on the object's core feature. This encompasses adjustments to sub-voxel positioning, the ascertainment of the object's altitude relative to the ground plane, the computation of its three-dimensional (3D) extents, and the determination of its yaw rotation angle. The refinement of sub-voxel positioning endeavors to ameliorate the inaccuracies engendered by voxelization and stride quantization endemic to the backbone network. The incorporation of height above the ground $((h_g))$ is paramount for the precise 3D positioning of objects, counterbalancing the omission of elevation data resultant from map-view projections. Orientation estimation is approached through the predictive modeling of the yaw angle, employing the sine and cosine of said angle as continuous regression targets to ensure a comprehensive depiction of object heading. In conjunction with the parameters delineating the bounding box dimensions, these regression heads collectively furnish an exhaustive representation of the 3D bounding box state, thereby enhancing the precision of object detection and localization processes within the analyzed point cloud dataset.

The loss during training calculates the loss of RPN and loss of Centerhead and optimizer tries to optimize the sum of those two loss.

$$L_a = L_{RPN} + L_{Centerhead} \tag{1}$$

E. Data Augmentation

In the pursuit of enhancing the robustness and generalizability of the model, several sophisticated data augmentation techniques have been meticulously employed. The initial strategy involves the rotation of Light Detection and Ranging (LiDAR) data by arbitrary degrees during each augmentation iteration, thereby introducing rotational variability into the dataset. Subsequently, to infuse the dataset with a broader spectrum of variations, random perturbations are applied to both the spatial positioning and orientation of the LiDAR data points. This method ensures the model's resilience to slight deviations in object placement and alignment.

The third technique employed encompasses the modulation of the scale of LiDAR data points through random amplification or diminution. This approach is instrumental in accustoming the model to variations in object sizes within the LiDAR scans, thereby enhancing its scale invariance. The fourth strategy, termed Data Set Flipping, entails the duplication of the LiDAR dataset followed by its horizontal or vertical inversion, effectively generating novel data configurations from existing scans.

Furthermore, an additional layer of complexity is introduced by superimposing random noise onto the LiDAR data, a maneuver designed to mimic the real-world inaccuracies inherent in LiDAR measurements. This particular augmentation simulates the potential measurement variances and sensor noise, thus preparing the model for effective operation under realistic conditions.

Collectively, these data augmentation techniques are pivotal in cultivating a model that exhibits heightened adaptability and improved predictive accuracy across a diverse array of scenarios, thereby mitigating the risk of overfitting to the nuances of the training dataset.

V. Experiments

A. Implementation detail

The implementation, training and benchamrk are done on OpenPCDet [21] for fast and fair comparison. The process of voxelization, pivotal in the transformation of point clouds into structured formats conducive to machine learning applications, can be effectuated through the utilization of either a Voxel Feature Extractor (VFE) or a Pillar Feature Extractor (PFE). The PFE operates on a principle akin to that of the VFE, with the notable distinction being its confinement to a singular grid along the z-dimension. This structural simplification of PFE permits the subsequent application of two-dimensional Convolutional Neural Networks (CNNs) for feature processing. The architectural frameworks corresponding to these extractors can be categorized into two paradigms: the voxelnet backbone and the pillar backbone, respectively.

The dimension of the voxels or pillars is a critical parameter within network. The default configuration for voxel dimensions is set at [0.1m, 0.1m, 0.15m]. It is observed that an increase in the size of voxels or pillars results in a reduction in the total number of voxels required to represent the space, which typically correlates with a decrease in computational complexity and, consequently, expedited inference times. However, this efficiency gain is often counterbalanced by a degradation in model performance due to the coarser granularity of the spatial representation. Conversely, reducing the voxel size enhances the model's spatial resolution and, potentially, its detection accuracy, at the cost of increased computational demand. The spatial domain for detection activities is delineated by a range of -75.2 meters to 75.2 meters along both the x and y axes, and -2 meters to 4 meters along the z-axis. This detection schema is uniformly applied in the context of PFE, albeit with an adjusted voxel dimension of [0.32m, 0.32m, 6.0m].

Regarding the architectural backbone utilized for processing the voxelized data, two variants are predominantly employed: the original voxelnet backbone and a modified version incorporating residual connections, herein referred to as the residual voxelnet backbone. These backbones serve as the foundational computational structures upon which the subsequent layers of the neural network are constructed, each offering unique characteristics in terms of computational efficiency and the ability to capture and propagate relevant features through the network.

B. Evaluation Metric

The evaluation metric employed is delineated by the authoritative Waymo Open Dataset [22] guidelines. The principal metric under consideration is the Average Precision Weighted by Heading (APH), mathematically represented as: $\text{APH} = \int h(r) * dr$.

where (h(r)) signifies the precision-recall curve, which is subsequently weighted by heading accuracy. This metric is

Performance @(train with 20% Data)	Veh_L1/%	Veh_L2/%	Ped_L2/%	Ped_L2/%	Cyc_L1/%	Cyc_L2/%
Pointpillar [19]	70.43/69.83	62.18/61.64	66.21/46.32	58.18/40.64	55.26/51.75	53.18/49.80
CenterPoint [11]	71.33/70.76	63.16/62.65	72.09/65.49	64.27/58.23	68.68/67.39	66.11/64.87
Part-A2-Anchor [20]	74.66/74.12	65.82/65.32	71.71/62.24	62.46/54.06	66.53/65.18	64.05/62.75
SECOND [3]	70.96/70.34	62.58/62.02	65.23/54.24	57.22/47.49	57.13/55.62	54.97/53.53
PV-RCNN [9]	**77.61/77.14**	**69.18/68.75**	79.42/73.31	70.88/65.21	72.50/71.39	69.84/68.77
CP-RCNN (ours)	74.38/74.23	65.68/64.76	**80.46/74.87**	**72.37/67.09**	**74.82/73.13**	**70.80/69.68**

TABLE I

COMPARISON WITH STATE OF THE ART

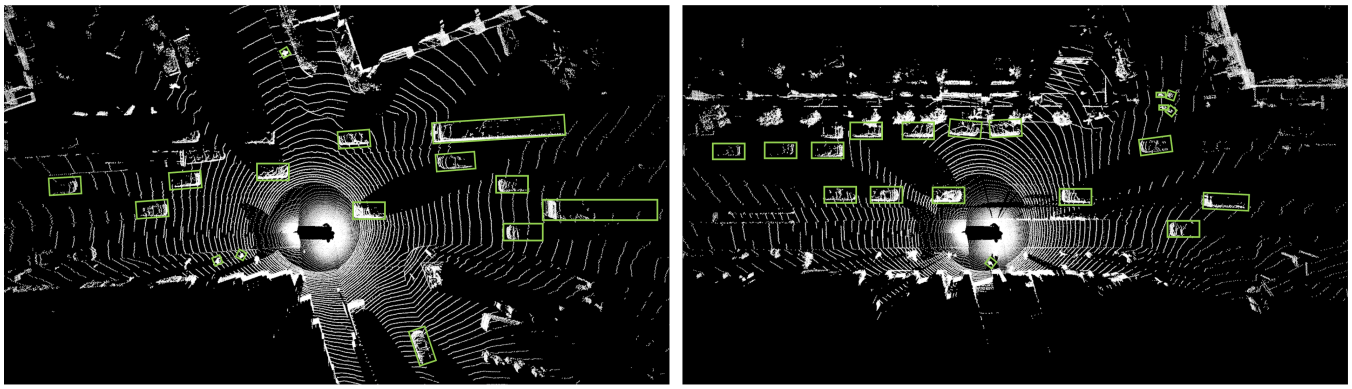

Fig. 3. Qualitative findings from the CP-RCNN analysis on Waymo validation. Where pointcloud are in white and green boxs are detection results.

	mAPH_L2 /%	Inference Speed/Hz
2D CNN	50.69	28.8
3D CNN	54.35	23.2
3D CNN + Centerhead	61.92	18.9
3D CNN + ROI Pooling + Centerhead	65.82	16.7
3D CNN(Resnet) + ROI Pooling + Centerhead	67.17	15.0

TABLE II

ABLATION STUDY. WHERE 2D CONVOLUTION IS AN ALTERNATIVE FEATURE EXTRACTOR TO 3D CONVOLUTION IN FIG3.

instrumental in quantifying the precision with which objects are identified, encapsulating the consideration of both false positives and false negatives within its assessment framework.

C. Comparison with SOTA

The ablation study results highlights the numerical superiority of the CP-RCNN model over other state-of-the-art counterparts with specifically focusing on pedestrian and cyclist detection. When evaluating the performance metrics, it is imperative to consider the precision in detection across various levels (Level 1 and Level 2) for both pedestrians and cyclists, which are critical categories for ensuring the safety and efficacy of autonomous navigation systems. Where Level 1 difficulty refers to objects that are easier to detect and typically have fewer occlusions, while Level 2 difficulty involves objects that are more challenging to detect, often due to partial occlusion or obstacles are at greater distance.

D. Ablation Study

For pedestrian detection at Level 1 Ped, CP-RCNN achieves an impressive score of 80.46%, surpassing the next best model, PV-RCNN, which scores 79.42%. At Level 2 Ped, CP-RCNN further extends its lead with a score of 74.87%, compared to PV-RCNN's 73.31%. This indicates CP-RCNN's enhanced capability to accurately identify pedestrians with a higher degree of precision, reducing the likelihood of false positives and false negatives, which are crucial for the reliable operation of autonomous systems in pedestrian-rich environments. In the domain of cyclist detection, the performance of CP-RCNN is equally commendable. At Level 1 Cyc, CP-RCNN records a score of 74.82%, outperforming CenterPoint, which achieves a score of 68.68%. At Level 2 Cyc, CP-RCNN maintains its superiority with a score of 70.80%, compared to CenterPoint's 66.11%. These results underscore CP-RCNN's adeptness at detecting cyclists, a challenging task given the cyclists' smaller size and more dynamic movement patterns compared to vehicles.

The numerical advantage of CP-RCNN in pedestrian and cyclist detection can be attributed to its approach to shared feature of multi-stage detection which allows for more nuanced and accurate representation of these smaller, dynamic objects. Furthermore, CP-RCNN's superior performance in Level 2 metrics, which are more stringent in evaluating detection precision, illustrates its robustness in complex scenarios and its potential to significantly enhance the safety measures of autonomous navigation systems.

The table presents a concise yet comprehensive ablation study focusing on the performance of various convolution-

based methods and their impact on the mean Average Precision at Level 2 (mAPH_L2/%) and inference speed (measured in Hertz, Hz). Each row of the table delineates a distinct configuration, starting from basic 2D convolution to more complex arrangements incorporating 3D convolution, Region of Interest (ROI) pooling, and a centerhead mechanism. From the simplest configuration, 2D convolution, which yields an mAPH_L2 of 50.69% and an inference speed of 28.8 Hz, there is a clear trend of increasing detection precision at the expense of reduced inference speed as the complexity of the model architecture escalates. The incorporation of 3D convolution enhances the mAPH_L2 to 54.35%, albeit with a slower inference speed of 23.2 Hz. This trend continues with the addition of a centerhead, boosting the mAPH_L2 significantly to 61.92% but further slowing down the inference speed to 18.9 Hz.

The combination of 3D convolution, ROI pooling and centerhead results in a further improved mAPH_L2 of 65.82%, with a corresponding decrease in inference speed to 16.7 Hz. The most complex configuration, which includes Residual 3D Convolution alongside ROI Pooling and a centerhead, achieves the highest mAPH_L2 of 67.17%. However, this configuration also exhibits the slowest inference speed of 15.0 Hz, underscoring the computational trade-offs inherent in increasing model complexity for enhanced detection accuracy. The results presented in this paper underscore the potential of CP-RCNN as a leading solution in the field of LiDAR-based object detection. Its high precision in detecting pedestrians and cyclists, coupled with a pragmatic approach to computational efficiency, positions CP-RCNN as a significant advancement in the development of autonomous driving technologies. Our future work will focus on further optimizing the balance between detection accuracy and inference speed, exploring new architectural innovations and computational strategies to enhance the practical applicability of CP-RCNN in real-world settings.

VI. CONCLUSIONS

In this study, we present CP-RCNN, an innovative LiDAR object detection approach that combines advanced voxelization techniques with refined feature extraction methods. Our model excels in setting new benchmarks for detecting pedestrians and cyclists, two challenging categories for autonomous navigation systems due to their small size and dynamic behavior.

Our comprehensive ablation study demonstrates that CP-RCNN not only outperforms existing models in terms of detection accuracy but also strikes an impressive balance between precision and computational efficiency. It notably enhances mean Average Precision at Level 2 (mAPH_L2) for pedestrian and cyclist detection, showcasing its superior ability to accurately identify these critical object categories across diverse scenarios. Despite the inevitable impact of increased model complexity on inference speed, CP-RCNN maintains competitive performance, positioning it as a viable choice for real-time applications.

REFERENCES

[1] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.

[2] D. Maturana and S. Scherer, "Voxnet: A 3d convolutional neural network for real-time object recognition," *IEEE International Conference on Intelligent Robots and Systems (IROS)*, pp. 922–928, 2015.

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

[4] C. R. Qi, H. Su, K. Mo, and L. J. Guibas, "Pointnet: Deep learning on point sets for 3d classification and segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 652–660, 2017.

[5] C. R. Qi, L. Yi, H. Su, and L. J. Guibas, "Pointnet++: Deep hierarchical feature learning on point sets in a metric space," *arXiv preprint arXiv:1706.02413*, 2017.

[6] X. Chen, H. Ma, J. Wan, B. Li, and T. Xia, "Multi-view 3d object detection network for autonomous driving," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1907–1915, 2017.

[7] A. Milioto, I. Vizzo, J. Behley, and C. Stachniss, "Rangenet++: Fast and accurate lidar semantic segmentation," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2019.

[8] M.-H. Guo, J.-X. Cai, Z.-N. Liu, T.-J. Mu, R. R. Martin, and S.-M. Hu, "Pct: Point cloud transformer," *Computational Visual Media*, vol. 7, p. 187–199, Apr. 2021.

[9] S. Shi, 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*, pp. 10529–10538, 2020.

[10] J. Deng, S. Shi, P. Li, W. Zhou, Y. Zhang, and H. Li, "Voxel r-cnn: Towards high performance voxel-based 3d object detection," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 35, pp. 1206–1214, 2021.

[11] T. Yin, X. Zhou, and P. Krähenbühl, "Center-based 3d object detection and tracking," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11784–11793, 2021.

[12] B. Zhu, Z. Jiang, X. Zhou, Z. Li, and G. Yu, "Class-balanced Grouping and Sampling for Point Cloud 3D Object Detection," *arXiv e-prints*, p. arXiv:1908.09492, Aug 2019.

[13] Z. Liu, X. Zhao, T. Huang, R. Hu, Y. Zhou, and X. Bai, "TANet: Robust 3d object detection from point clouds with triple attention," in *Proceedings of the AAAI Conference on Artificial Intelligence*, pp. 11677–11684, 2020.

[14] Z. Yang, Y. Sun, S. Liu, and J. Jia, "3dssd: Point-based 3d single stage object detector," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11040–11048, 2020.

[15] Y. Chen, S. Liu, X. Shen, and J. Jia, "Fast point r-cnn," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 9774–9783, 2019.

[16] Y. Zhou and O. Tuzel, "Voxelnet: End-to-end learning for point cloud based 3d object detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4490–4499, 2018.

[17] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," in *Advances in neural information processing systems*, pp. 91–99, 2015.

[18] X. Zhou, D. Wang, and P. Krähenbühl, "Objects as points," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7281–7290, 2019.

[19] A. H. Lang, S. Vora, H. Caesar, L. Zhou, J. Yang, and O. Beijbom, "Pointpillars: Fast encoders for object detection from point clouds," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 12697–12705, 2019.

[20] S. Shi, Z. Wang, J. Shi, X. Wang, and H. Li, "Part-A^2 net: 3d part-aware and aggregation neural network for object detection from point cloud," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 12405–12414, 2019.

[21] O. D. Team, "OpenPCDet: An open-source toolbox for 3d object detection from point clouds." `https://github.com/open-mmlab/OpenPCDet`, 2020.

[22] P. Sun, H. Kretzschmar, X. Dotiwalla, A. Chouard, V. Patnaik, P. Tsui, J. Guo, Y. Zhou, Y. Chai, B. 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*, pp. 2446–2454, 2020.