IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 29 June 2024, accepted 14 July 2024, date of publication 2 August 2024, date of current version 12 August 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3437642

RESEARCH ARTICLE

ContextNet: Leveraging Comprehensive Contextual Information for Enhanced 3D Object Detection

CAIYAN PEI[1], SHUAI ZHANG[1,2], LIJUN CAO[1], AND LIQIANG ZHAO[1]

[1]School of Mathematics and Information Technology, Hebei Normal University of Science and Technology, Qinhuangdao, Hebei 066004, China
[2]School of Information Science and Engineering, Yanshan University, Qinhuangdao, Hebei 066004, China

Corresponding authors: Shuai Zhang (zhs3124@hevttc.edu.cn) and Liqiang Zhao (zlq1977@hevttc.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 61672448, in part by Hebei Natural Science Foundation under Grant F2023203058 and Grant F2022203045, in part by the Science and Technology (S&T) Program of Hebei under Grant 22310301D, and in part by the Project of Hebei Key Laboratory of Software Engineering under Grant 2256763H.

ABSTRACT The progress in object detection for autonomous driving using LiDAR point cloud data has been remarkable. However, current voxel-based two-stage detectors have not fully capitalized on the wealth of contextual information present in the point cloud data. Typically, Voxel Feature Encoding (VFE) layers tend to focus exclusively on internal voxel information, neglecting the broader context. Additionally, the process of extracting 3D proposal features through Region of Interest (ROI) spatial quantization and pooling downsampling results in a loss of spatial detail within the proposed regions. This limitation in capturing contextual details presents challenges for accurate object detection and positioning, particularly over long distances. In this paper, we propose ContextNet, which leverages comprehensive contextual information for enhanced 3D object detection. Specifically, it comprises two modules: the Voxel Self-Attention Encoding module (VSAE) and the Joint Channel Self-Attention Re-weight module (JCSR). VSAE establishes dependencies between voxels through self-attention, expanding the receptive field and introducing substantial contextual information. JCSR employs joint attention to extract both local channel information and global context information from the raw point cloud within the RoI region. By integrating these two sets of information and re-weighting the point features, the 3D proposal is refined, enabling a more accurate estimation of the object's position and confidence. Extensive experiments conducted on the KITTI dataset demonstrate that our approach outperforms voxel-based two-stage methods, particularly with a 9.5% improvement in the mAP compared to the baseline on the nuScenes test dataset, and an improved 1.61% hard AP compared to the baseline on the KITTI benchmark.

INDEX TERMS Autonomous driving, 3D object detection, LiDAR sensor.

I. INTRODUCTION

In recent years, there has been a significant focus on autonomous driving technology, with particular attention given to the rapid advancement and extensive research on 3D object detection within the perception system of autonomous vehicles, recognized as a crucial component [1], [2]. The application of 2D object detection or Semantic Segmentation [3], [4], [5], [6] in autonomous driving perception has achieved some success. For example, Liang et al. [3] proposed an improved sparse R-CNN that integrates coordinate attention block with ResNeSt and builds a feature pyramid to modify the backbone, which enables the extracted features to focus on important information, and improves the detection accuracy. Liang et al. [4] proposes a category-assisted transformer object detector called Detect-Former for autonomous driving. Liang et al. [5] proposed an anchor-free lightweight object detector, ALODAD, for autonomous driving. Liang et al. [6] proposed network takes a BEV point cloud image generated by the MMS as input and directly segments map features, including line and area features, from that image. While 2D object detection

The associate editor coordinating the review of this manuscript and approving it for publication was Zhongyi Guo.

technology [7], [8], [9] has shown maturity, its inability to effectively analyze data using 2D convolutional neural networks is due to the sparse, chaotic, and non-structural characteristics of point clouds. Consequently, to overcome this obstacle, certain approaches [10], [11], [12], [13] have been employed to directly extract pertinent information from individual pixel regions of the raw point cloud, converting it into 2D Bird's Eye View (BEV) feature maps, which are then processed using 2D CNNs. Nevertheless, this basic point-level statistical aggregation technique leads to a loss of important 3D spatial intricacies, resulting in suboptimal detector performance.

Spatial detail information equips the model with comprehensive 3D contextual details surrounding the object, aiding in the model's comprehension and localization of the object. In contrast to the aforementioned approaches relying on BEV representation, point-based and voxel-based techniques [14], [15], [16] have the advantage of directly capturing data from the point cloud space, thereby conserving 3D spatial intricacies to a significant degree.

In the pursuit of accurately preserving 3D spatial intricacies, certain methodologies opt to directly acquire data within the point cloud realm. These methodologies consist of two primary categories: point-based and voxel-based methods [2], [14], [15], [16], [17]. Point-based techniques [18], [19], [20], [21], [22], [23] employ point feature extractors [24], [25] to extract multi-scale point-level features from the raw point cloud, integrating surrounding contextual details for points via farthest point sampling (FPS) and ball query. However, this approach comes with a considerable computational burden. Voxel-based methods [13], [26], [27], on the other hand, segment 3D space into uniform cubic volumes (voxels). They project the point cloud onto voxel grids to convert discrete points into a compact voxel representation, followed by voxel feature encoding (VFE) [28] for encoding the voxels to facilitate 3D convolution-based feature extraction. In general, compared to point-based methods, voxel-based techniques tend to extract features more effectively. Nonetheless, the voxel feature encoder has a limited receptive field, capturing features solely within the voxel and neglecting broader contextual dependencies, thereby reducing object localization accuracy.

As foreground points and background points contribute differently to proposal refinement, we use channel attention and self-attention to decode the encoded features, integrating local channel information and global contextual information into the raw point features to re-weight the point features within the RoI region and refine the proposal features, improving the localization accuracy and detection performance of the model in 3D space. Both of these proposed modules are plug-and-play and can be easily applied to voxel-based methods. With the support of these two modules, the detection performance is significantly improved.

Single-modal 3D object detection methods [29], [30], [31], [32] can be divided into two categories: voxel-based and point-based methods. Point-based methods treat point clouds as a set of discrete points and sample and group them for each point. PointNet [24] and PointNet++ [25] were the first to use this approach to process point cloud data. Although these two methods cannot be directly used in 3D object detection tasks, they extract point-level features in a multi-scale manner as a component of point-based methods. PointRCNN [19] further optimize 3D RoIs use RoI Pooling and Point-wise MLPs (multi-layer perceptrons) based on PointNet outputs. Point-GNN [18] uses PointNet to extract point features and uses GNN to infer and aggregate contextual information of points in the local neighborhood graph. 3DSSD [21] proposes a new point cloud sampling method called Feature Distance Compounding (FDC), which samples point cloud data by compounding feature distances using Euclidean distance. STD [33] separates foreground and background points in point clouds using segmentation algorithms, to detect objects more effectively. Point-based methods need to query the local neighborhood of each point, and the computational cost of this operation increases quadratically with the query sphere radius and the number of points in the point cloud, resulting in high computational complexity.

Voxel-based methods are a method of dividing 3D space into equally sized cubic units (voxels). By mapping point cloud data onto a voxel grid, the discrete point cloud data is transformed into a regular and dense voxel representation. VoxelNet [28] is the first method to use a fully voxel-based representation, but due to the use of 3D convolution, the computation and memory costs of this method grow cubically with the number of voxels involved in the operation. SECOND [34] utilizes 3D sparse convolution and stores voxel indices in a hash table to speed up data lookup for specific positions. This improves the efficiency of the algorithm and reduces. Voxel RCNN [29] uses Voxel Query, RPN, and detection head joint training, and performs pooling operations on the 3D voxel grid using Voxel RoI Pooling to better capture the spatial features of objects. Although voxel-based methods are more suitable for feature extraction, these methods cannot accurately capture contextual information like point-based methods. The association between voxels is weaker, especially for distant voxels, which affects the model's learning and detection accuracy for the entire object. Although some methods [13], [26] utilize graph-based knowledge to solve this problem, the construction of the graph relies on manual design. For example, Song et al. [26] present a robust network voxel-as-point network (VP-Net) that views voxels as points to accurately detect 3-D objects in LiDAR point clouds and can capture objects' internal relationships.

As the evolution of detection algorithms continues, voxel-based methods [27], [29], [35], [36] have introduced proposal refinement networks into the network framework to form a two-stage architecture aimed at improving the model's detection performance. However, the requirement for voxel features in the proposal refinement process, which typically have lower resolutions, poses challenges [37], [38]. The conventional approach of dividing the Region of Interest

(RoI) into a grid and using pooling operations to aggregate features results in some loss of spatial information within the RoI, thereby limiting the ability to capture contextual information between points. Based on these limitations, we propose a voxel-based two-stage detection framework that leverages self-attention and channel attention mechanisms to aggregate contextual information between features, thus maximizing the retention of 3D spatial detail information.

In this paper, we introduce two principal contributions.

Our first contribution is the innovative Voxel Self-Attention Encoding (VSAE) module, an efficient extension of traditional voxel feature encoding (VFE) for voxel encoding. To effectively capture information, we construct self-attention between voxels, thereby enhancing the correlation between them. We apply multi-head attention to non-empty voxel positions, boosting the modeling of relationships among different voxels by learning their relative importance. This approach fosters contextual dependency information between voxels, augmenting the model's learning ability and comprehension of objects in 3D space.

Our second contribution is the Joint Channel and Self-Attention Re-weight (JCSR) module, designed to refine 3D proposals. Specifically, we map the proposals to the raw point features and use self-attention for feature encoding. We then assign weights to the point features within the RoI region based on their importance to refine the point features and capture the correlation between points. Given the differing contributions of foreground and background points to proposal refinement, our module employs channel attention and self-attention to decode encoded features. This process integrates local channel and global contextual information into the raw point features, re-weights them within the RoI, and refines proposal features. The result is enhanced positioning accuracy and detection performance in 3D space. Both modules we propose are plug-and-play, readily applicable to voxel-based methodologies. The detection performance has seen substantial improvement with these modules in place.

II. RELATED WORK

A. LIDAR-BASED 3D OBJECT DETECTION

LiDAR sensors play a critical role in enhancing autonomous driving systems by enabling the perception of objects in 3D space, particularly in difficult lighting or adverse weather conditions [17], [39], [40]. They are known to outperform camera sensors in terms of reliability. Current methods for 3D object detection using LiDAR can be broadly categorized into three groups based on different point cloud encoding formats: point-based [18], [19], [21], [41], [42], voxel-based [29], [34], [43], [44], [45], and point-voxel fusion methods [20], [33], [46], [47], [48].

Voxel processing involves dividing 3D space into regular voxel grids with dimensions ($d_L \times d_W \times d_H$) in the x, y, and z directions. Only voxel units containing points are stored and utilized for feature extraction due to the sparse distribution of point clouds, resulting in many empty voxel units. Key works like VoxelNet [28] and its optimization in SECOND leverage 3D convolutional networks with sparse convolution. Later research has built upon these approaches by employing similar voxel encoding strategies. Furthermore, it is worth noting that Pillars can be considered a distinctive variant of voxels. Specifically, the point clouds are divided into a uniformly distributed grid on the x-y plane, with no binning performed along the z-axis. As the pioneering work in this series [43], [49], [50], [51], [52], PointPillar [43] was the first to introduce the pillar representation. Subsequent studies have expanded upon the concepts of 2D detection by incorporating the PointPillars approach. PillarNet [49] leverages an 'encoder-neck-head' detection structure to enhance the efficacy of pillar-based detection techniques. SWFormer [50] and ESS [53], inspired by the Swin Transformer [54], implement a multi-scale window strategy on pseudo-images, thus allowing the network to maintain a comprehensive receptive field. PillarNeXt [51] combines an array of established 2D detection methods to achieve a level of performance in line with voxel-based approaches.

In the realm of point-based 3D detection [18], [19], [21], [41], [42], [55], [56], early work extended the PointNet [24] backbone with a two-stage proposal refinement network to handle large-scale scenes with over 100k points. Recent studies [19], [41], [42] have addressed the computational burden by introducing semantic segmentation tasks during detection to filter out irrelevant background points.

Efforts have also been made to tackle the uncontrolled receptive field issue in PointNet and PointNet++ [24] by integrating Graph Neural Networks (GNN) or Transformer architectures. Notably, in point-voxel methods, PV-RCNN [20] utilizes SECOND as the first-stage detector and proposes a second-stage refinement step with a Region of Interest (RoI) grid pool for keypoint feature fusion. Subsequent research has focused on enhancing second-stage detection with attention mechanisms, scale-aware pooling, and point density-aware refinement modules.

These 3D detectors [18], [19], [42], [57] primarily using LiDAR data rely on sparse and noisy contexts provided by point cloud data. However, in challenging scenarios with low reflectivity, small objects, or severe occlusions, relying solely on point cloud data may lead to inaccurate detections. Therefore, the current focus is on exploring multi-modal contexts by integrating geometrically informed point clouds and semantically rich images to enhance 3D object detection capabilities.

B. CAMERA-BASED 3D OBJECT DETECTION

There are several types of sensors used for 3D object detection, with radars, cameras, and LiDAR sensors being the most commonly used. Radars are known for their long detection range and resilience to different weather conditions, and they can also provide velocity measurements due to the

Doppler effect. Cameras, on the other hand, are cost-effective and readily available, playing a crucial role in understanding semantics such as identifying traffic signs.

Camera-based 3D object detection serves as a foundational component for various downstream applications. Monocular 3D detection focuses on identifying 3D objects from a single input image. For instance, FCOS3D [58] expands upon the 2D FCOS detector to enable 3D detection by estimating 3D bounding boxes. Multi-view 3D object detection integrates multiple images to enhance geometric inference. PETR [59] enhances the sparse detector DETR [60] by introducing 3D positional encoding, while PETRv2 [61] further enhances this by including temporal modeling. StreamPETR introduces a unique query propagation algorithm to better utilize temporal information over long ranges.

BEV-based 3D object detectors transform multi-view images into a unified Bird's Eye View (BEV) representation for 3D object detection. BEVDet [62] and its subsequent work utilize LSS to compute BEV features and predict objects through convolutional heads. BEVFormer [63], on the other hand, leverages deformable attention operations [64] for computing BEV features and employs a DETR-style head [60] for object detection. These advancements in 3D object detection technologies contribute significantly to the field's progress and application in various real-world scenarios.

C. MULTI-MODEL 3D OBJECT DETECTION

The integration of cameras and LiDAR sensors for 3D object detection presents a significant opportunity to enhance detection accuracy by leveraging the complementary strengths of each sensor type. Cameras excel at capturing rich color information, which can be leveraged for extracting detailed semantic features. On the other hand, LiDAR sensors are proficient in providing precise 3D localization data, offering valuable insights into the spatial structure of the environment [65]. AVOD [66], MV3D [67] and F-Pointnet [68] are the pioneering proposal-level fusion works that perform the feature extraction of two modalities independently and simply concatenate multi-modal features via 2D and 3D RoI directly. CLOCs [69] directly combine the detection results from the pre-trained 2D and 3D detectors without integrating the features. They maintain instance semantic consistency in cross-modal fusion, while suffering from coarse feature aggregation and interaction. Since then, increasing attention has been paid to globally enhancing point cloud features through crossmodal fusion. Point decoration approaches [70], [71], [72] augment each LiDAR point with the semantic scores or image features extracted from the pre-trained segmentation network. 3D-CVF [12] and EPNet [73] explore crossmodal feature fusion with a learned calibration matrix. Recent studies have explored global fusion in the shared representation space based on the view transformation in the same way. These methods [74], [75], [76], [77] are less effective in exploiting the spatial cues of point cloud, and potentially compromise the quality of camera bird's-eye view (BEV) representation and cross-modal alignment. Besides, many concurrent approaches [78], [79] introduce the cross-attention module to adaptively align and fuse point cloud features with image features through the learned offset matrices. Addressing these challenges requires the development of efficient fusion strategies. Researchers are exploring various approaches to integrate multi-modal information effectively, aiming to minimize computational overhead while maintaining high detection accuracy. Despite the progress made, efficiently fusing camera and LiDAR data for 3D object detection remains an ongoing research challenge.

III. METHODS

We propose ContextNet, which comprises two modules: Voxel Self-Attention Encoding (VSAE) and Joint Channel and Self-Attention Re-weight (JCSR). As shown in Fig. 1, the Voxel Self-Attention Encoding (VSAE) module focuses on establishing dependencies between voxels using self-attention mechanisms. By doing so, it expands the receptive field and integrates significant contextual information. Essentially, VSAE enhances the understanding of voxel relationships within the point cloud data, contributing to improved object detection accuracy. The Joint Channel and Self-Attention Re-weight (JCSR) module employs joint attention to capture both local channel details and global context from raw point cloud data within the Region of Interest (RoI) region. By combining channel-wise and spatial attention mechanisms, JCSR effectively extracts relevant features while re-weighting the point features to refine the 3D proposal. This process aids in more accurately estimating the position and confidence of detected objects.

A. VOXEL SELF-ATTENTION ENCODING

In this section, we will delve into the intricacies of the Voxel Self-Attention Encoding (VSAE) module's design. To kick things off, we'll outline the key steps involved in crafting the VSAE module. Initially, the point cloud undergoes a process of voxelization, resulting in the creation of a dense voxel grid. Subsequently, self-attention mechanisms come into play to facilitate the establishment of relationships among these voxels.

Self-attention proves to be instrumental in fostering connections between voxels. This is achieved by singling out a query voxel, denoted as $V_i \in \mathbb{R}^{N_i \times C}$, and a participating voxel, labeled as $V_j \in \mathbb{R}^{N_j \times C}$, for the attention calculation process. Through this calculation, a feature termed f_i^{attn} is derived. Among them, N_i and N_j are the number of Voxel V_i and V_j and C is the dimension of V_i and V_j. The primary objective of applying self-attention is to model the interplay between non-empty voxels, thereby capturing crucial contextual information.

During the attention calculation phase, the query $Q_i \in \mathbb{R}^{N_i \times q_i \times H \times H_D}$ is generated based on the query voxel $V_i \in \mathbb{R}^{N_i \times v_i \times H \times H_D}$, whereas the key $K_j \in \mathbb{R}^{N_j \times k_j \times H \times H_D}$ and value

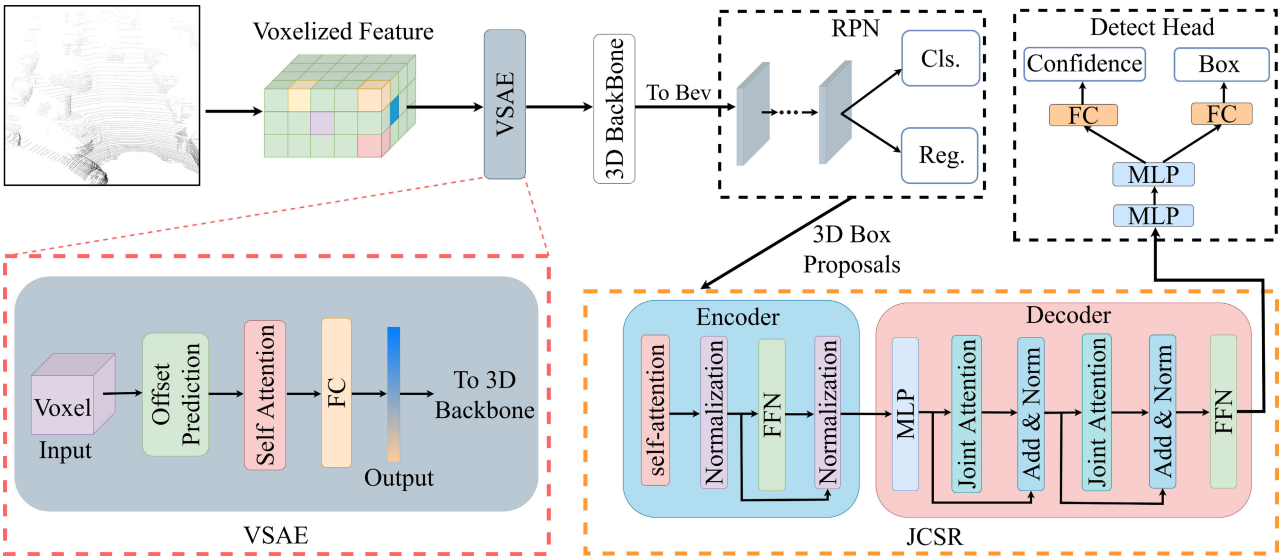

FIGURE 1. Overview of ContextNet. We propose ContextNet, which introduces attention mechanism to enrich contextual information of features. ContextNet consists of two modules: Voxel-based Feature Encode Module (VSAE) and Joint Channel and Self-Attention Reweighting Module (JCAR). VSAE takes voxel features as input and utilizes self-attention to build relationships between voxels, thereby enriching feature representation and improving proposal quality. JCAR takes raw point and 3D proposal as input, and extracts local and global fusion information by jointly using channel attention and self-attention. It then reassigns weights to point features and proposals to refine and optimize proposals.

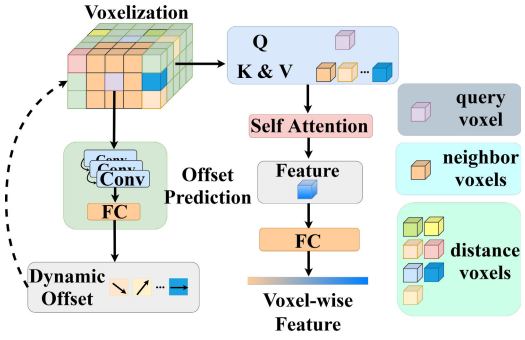

FIGURE 2. The framework of VSAE. VSAE module first uses convolution to learn dynamic offsets to locate positions of distance voxels. Then, it performs attention calculation between query voxels and participating voxels (composed of neighbor voxels and distance voxels) to aggregate context-dependent information among voxels. Finally, it encodes the aggregated features into a voxel-wise feature representation.

$V_j \in \mathbb{R}^{N_j \times C}$ are derived from the involved voxel. Among them, N_i and N_j are the number of Voxel V_i and V_j, and q_i, v_i, amd k_j are the dimension of query Q_i, value V_i, and key K_j. H represents the number of attention head. H_D represents the dimension in the attention head. This formalizes the process of self-attention, paving the way for a comprehensive understanding of the relationships between voxels and the contextual nuances encapsulated within them.

$$\mathbf{Q}_i = f_i \mathbf{W}_Q, \quad \mathbf{K}_j = f_j \mathbf{W}_K, \quad \mathbf{V}_j = f_j \mathbf{W}_V, \tag{1}$$

where $W_Q \in \mathbb{R}^{H \times H_D}$, $W_K \in \mathbb{R}^{H \times H_D}$, and $W_V \in \mathbb{R}^{H \times H_D}$ are the weight matrices corresponding to the query, key, and value, f_i and f_j are the feature vectors corresponding to the query voxel and the involved voxel respectively.

To compute the attention matrix $\mathcal{A}$, we first multiply the query vector Q_i with the key vector K_j. The next step involves normalizing the attention matrix $\mathcal{A}$ along the voxel direction to yield the normalized attention matrix $\mathcal{A}'$. Moving forward, we proceed to multiply the value V_j with the normalized attention matrix $\mathcal{A}'$. Subsequently, we aggregate the involved voxel features by employing the weighted sum of the attention weights. By integrating the voxel features that encapsulate the relationships and contextual information between voxels with the query voxel features, we enhance and update the query voxel features. This sequence of computations can be succinctly represented as follows:

$$\mathcal{A} = \frac{\mathbf{K_j^T Q_i}}{\sqrt{\mathbf{d_k}}}, \quad \mathcal{A}' = \sigma(\mathcal{A}), \quad f_i^{attn} = \mathbf{V_j} \mathcal{A}' \tag{2}$$

where $\sqrt{d_k}$ is a normalization factor, and $\sigma(\cdot)$is the softmax function. We perform a weighted sum of all values based on the attention weights and use a feedforward network(FFN) to generate the features, Where FFN is a simple fully connected (FC) layer.

The computational complexity associated with attention calculations is proportional to the square of the input size, resulting in considerable costs when performing global attention operations. Specifically, given the number of voxels N and the image feature dimensions $W \times H$, the complexity balloons to $O(NWH)$. To address this challenge, we sample the voxels participating in the attention computation, thereby curbing superfluous processing demands. In contrast to earlier approaches like Voxel RCNN [29] and VP-Net [26], which harbored substantial redundant information, we employ an attention mechanism. By utilizing KNN, we selectively focus on K dynamic, high-quality voxel features to enhance feature aggregation. Consequently, our method slashes the complexity from $O(NWH)$ to $O(NK^2)$, with K, the number of selected voxels, fixed at 32. In Figure 2,

the selected voxels comprise neighboring non-empty voxels as well as distant non-empty voxels. Sampling neighboring non-empty voxels is a relatively straightforward task, whereas sampling distant non-empty voxels poses a greater challenge. To tackle this challenge, we employ a learnable convolutional network to dynamically generate sampling offsets $(\Delta x, \Delta y, \Delta z)$ based on the spatial location of the current query voxel. These sampling offsets serve to precisely determine the location of distant voxels.

Overall, by strategically sampling voxels for attention calculation and leveraging a learnable convolutional network for generating sampling offsets, we aim to optimize the computational efficiency of the process while maintaining high detection performance.

$$\mathcal{P} = r \cdot (v + 0.5) \tag{3}$$

where $\mathcal{P}$ denotes the center coordinates of the actual voxel, r is the voxel size, and v is the index of the current voxel.

B. JOINT CHANNEL AND SELF-ATTENTION RE-WEIGHT

In the realm of two-stage object detection frameworks, there is a prevailing reliance on voxel features to engage in the process of proposal refinement. Nevertheless, these features are often characterized by their low resolution, a factor that can contribute to inaccuracies in object localization. Furthermore, the current methodology of feature pre-processing in proposal refinement is deemed laborious and intricate. As a response to these challenges, we advocate for the adoption of a novel proposal refinement strategy that streamlines the process through the incorporation of the Joint Attention method.

In this section, we aim to elucidate the intricacies of the Joint Channel and Self-Attention Re-weight (JCSR) module. Traditional methods have often utilized low-resolution voxel features to enhance proposals and pool features, inadvertently leading to a loss of crucial contextual information. To address this limitation, we introduce the innovative Joint Channel and Self-Attention Re-weight (JCSR) module.

Our approach involves refining proposals by identifying the Region of Interest (RoI) based on the proposal and subsequently sampling $\mathcal{N}$=256 points within the RoI. While some of these points may represent background elements, foreground points play a pivotal role in refining the proposal. To this end, the encoder within the JCSR module is instrumental in reassigning weights to these points and identifying the most salient ones. For a visual representation of the encoder in JCSR, please refer to Figure 1.

Initially, establish the relationship between the feature of the proposal and the feature of each sampling point by computing the distance feature between every sample point and the proposal's center point, which can be represented as $p_d = p_i - p_c$. Subsequently, proceed to map the proposal features onto the point features, resulting in the feature f_i post-mapping. This process can be formally expressed as follows:

$$\mathbf{f_i} = \mathbb{A}([\mathbf{p_d}, \mathbf{l_c}, \mathbf{w_c}, \mathbf{h_c}, \theta_\mathbf{c}, \rho_\mathbf{i}]) \tag{4}$$

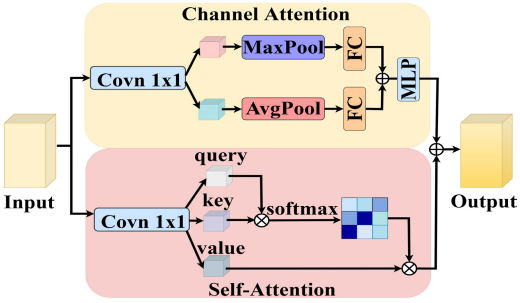

FIGURE 3. A detailed design illustration of the Joint Attention module, which is a commonly used and simple component that includes only MLP, MaxPool layer, AvgPool layer, and Self-Attention, is used for 3D proposal refinement.

where $\mathbb{A}(\cdot)$ is a fully-connected (FC) layer that lifts the dimensionality of the concatenated features to enrich the information. The features of the proposal, including length l_c, width w_c, height h_c, orientation angle θ_c, and the feature ρ_i of each sample point, are encoded as:

$$\mathbf{F} = [\mathbf{f_1}, \cdots, \mathbf{f}_\mathcal{N}] \tag{5}$$

These characteristics help to maintain the spatial details of the point cloud dataset [80]. Subsequently, the proposed features are inputted into a self-attention layer for encoding. During this stage, the values Q_e, K_e, and V_e are derived via linear mapping functions $F\mathbf{W}_Q$, $F\mathbf{W}_K$, and $F\mathbf{W}_V$, respectively. The encoding features are then computed through a sequence of operations. Initially, the attention calculation determines the weighted features, as denoted by the equation 2. Following this, for each weighted feature, we apply residual connection and Layer Normalization processes, resulting in the encoding features denoted as F^{enc} formally.

$$\mathbf{F_i}^{enc} = N(\mathcal{F}(N(\mathbf{f_i^{attn}}))) \tag{6}$$

where N(·) represents the normalization operator, and $\mathcal{F}(\cdot)$ denotes the FFN with a FC layer activation.

The conventional Transformer decoder has limitations in effectively capturing local point information and adapting to various scenes and objects within the RoI region. To overcome these challenges, we have developed a novel decoding module that integrates both local and global information to enhance the refinement of proposals. This is illustrated in Figure 1 through the incorporation of the Joint Channel and Spatial Relationship (JCSR) module.

The decoding module comprises key components such as Multilayer Perceptron (MLP), addition and normalization operations, Joint Attention mechanism, and Feedforward Network (FFN). Notably, the Joint Attention mechanism is a novel attention approach introduced by our team to address the shortcomings of traditional self-attention mechanisms.

By leveraging channel attention to combine local channel information with global contextual information, our proposed method effectively assigns weights to points within the RoI region, thereby improving the quality of proposal refinement. The innovative nature of our Joint Attention

mechanism allows for a more adaptive and comprehensive handling of diverse scenes and objects, enhancing the overall performance of the decoding process.

In summary, the integration of local and global information through the JCSR module, along with the introduction of the Joint Attention mechanism, represents a significant advancement in refining proposals within the decoding framework, as depicted in Fig. 1.

The detailed design of the Joint Attention module is illustrated in Fig. 3. It consists of two parallel pipelines. One pipeline applies global average pooling and global max pooling on the features, followed by a fully-connected layer. This introduces smooth channel attention through the global average pooling, and more sensitive attention to salient features through global max pooling. The pooled features are element-wise added and normalized to calculate channel weights. These weights are multiplied with the raw features to obtain a channel-weighted feature representation. The second pipeline employs self-attention on the input features to generate attention weights. This produces a global feature representation through interactions between the query embeddings. Finally, the features from the two branches are added to obtain features that fuse global contextual information and local channel information, providing a more comprehensive and accurate feature representation to re-allocating weights to points in the RoI:

$$\omega = \sigma(AP(F') + MP(F') + \frac{K_d^T Q_d}{\sqrt{d_k}}) \tag{7}$$

where $AP(\cdot)$ is the average pooling, $MP(\cdot)$ is the max pooling. The feature F' goes through nonlinear transformation by the MLP, then the Q_d and K_d of the decoder are generated through linear mapping functions. As a result, the features of the proposal can be represented as:

$$\mathcal{F}_{proposal} = \omega F' \tag{8}$$

The 3D proposal refinement network is designed to predict and pinpoint the object's location using the Region of Interest (RoI) feature of the provided proposal. This network leverages a Multilayer Perceptron (MLP) architecture to enhance the proposals and incorporates two fully connected (FC) branches. These branches are responsible for predicting confidence scores and bounding box regression, respectively.

IV. EXPERIMENTS

In this section, we present a comprehensive overview of the datasets used, training methodologies employed, and the evaluation criteria set forth. Our study includes a thorough examination of two modules within the framework, evaluated against the established KITTI benchmark [81] and nuScenes benchmark [82], with a comparative analysis against contemporary state-of-the-art detection models. Furthermore, our research incorporates a series of meticulous ablation studies aimed at deconstructing each element for a detailed analysis and validation of our proposed design.

A. KITTI DATASET

The KITTI [81] dataset is one of the most commonly used 3D object detection datasets for autonomous driving. It contains 7481 samples for training and 7518 samples for testing. Models are typically evaluated based on the mean Average Precision (mAP) metric. We conduct experiments on the most commonly detected car category and use an Average Precision (AP) with an IoU threshold of 0.7 as the evaluation metric. The ground truth is calculated with a recall of 40 positions (R40). To further compare the results with other methods on the KITTI 3D detection benchmark, we divide the KITTI training dataset into a 4:1 ratio for training and validation, and report the performance on the KITTI test dataset.

B. NUSCENES DATASET

The nuScenes [82] dataset is a large-scale 3D detection benchmark consisting of 700 training scenes, 150 validation scenes, and 150 testing scenes. The data are collected using six multi-view cameras and a 32-channel LiDAR sensor. It includes 360-degree object annotations for 10 object classes. To evaluate the detection performance, the primary metrics used are the mean Average Precision (mAP) and the nuScenes detection score (NDS).

C. IMPLEMENTATION DETAILS

For the KITTI dataset, we set the detection ranges on the X, Y, Z axes to [0, 70.4]m, [−40, 40]m, and [−3, 1]m respectively, and the voxel size is (0.05m, 0.05m, 0.1m). The detection range is set to [0, 70.4]m on the X-axis, [−40, 40]m on the Y-axis, and [−3, 1]m on the Z-axis, with voxel sizes of (0.05m, 0.05m, 0.1m) on each axis. We selected the one-stage method SECOND [34] as a baseline in the KITTI dataset. Moreover, we validate our ContextNet on the nuScenes [82] dataset using CenterPoint [95] as the baseline. The detection range for the X and Y axes is set at [−54m, 54m] and [−5m, 3m] for the Z axis. The input voxel size is set at (0.075m, 0.075m, 0.2m), and the maximum number of point clouds contained in each voxel is set to 10.

1) TRAINING

We employ the ADAM optimizer for training. Our training process involves utilizing 8 GTX 2080 Ti GPUs to train the complete network for 80 epochs with a batch size of 16. This process takes approximately 5 hours. For the learning rate, we utilize cosine annealing to decay it and set the initial value to 0.001. During the proposal refinement stage, following the design of PV-RCNN [20], we randomly sample 128 proposals and additionally sample N=256 points in the RoIs. The threshold of 3D IoU is 0.55. We use OpenPCDet [96] as our codebase for more detailed configuration information.

2) INFERENCE

We utilize non-maximum suppression to choose the best 100 proposals, while an IoU of 0.7 for filtering. After

TABLE 1. Comparison of mAP (R_{40}) for the car category on the KITTI dataset is reported.L+R represents the method that combines point cloud and image data, while L represents the LiDAR-only method.

Method	Modality	Publication	$AP_{3D}(\%)$			$AP_{BEV}(\%)$		
			Easy	Mod.	Hard	Easy	Mod.	Hard
EPNet [83]	L&R	ECCV2020	92.28	82.59	80.14	95.51	91.47	91.16
3D-CVF [12]	L&R	ECCV2020	89.67	79.88	78.47	_	_	_
AutoAlign [84]	L&R	IJCAI2022	88.16	78.01	74.90	_	_	_
VoPiFNet [85]	L&R	TITS2024	88.51	80.97	76.74	-	-	-
MSF-S [71]	L&R	TGRS2024	91.04	82.31	79.31	-	-	-
MSF-P [71]	L&R	TGRS2024	89.58	78.60	75.63	-	-	-
RoboFusion [78]	L&R	IJCAI2024	91.75	84.08	80.71	-	-	-
PPF-Det [86]	L&R	TITS2024	89.51	84.46	78.91	-	-	-
GraphAlign++ [36]	L&R	TCSVT2024	90.98	83.76	80.16	-	-	-
Point-GNN [18]	L	CVPR2020	87.89	78.34	77.38	89.82	88.31	87.16
PointRCNN [19]	L	CVPR2019	88.88	78.63	77.38	_	_	_
PV-RCNN [20]	L	CVPR2020	**92.57**	84.83	82.69	**95.76**	91.11	88.93
VoxelNet [28]	L	CVPR2018	81.97	65.46	62.85	_	_	_
SECOND [34]	L	Sensor2018	87.56	77.21	74.35	_	_	_
STD [33]	L	CVPR2019	89.70	79.80	79.30	90.51	88.50	88.11
SA-SSD [22]	L	CVPR2020	90.15	79.91	78.78	_	_	_
Part-A^2 [44]	L	TPAMI2020	89.47	79.47	78.54	90.42	88.61	87.31
PG-RCNN [87]	L	TGRS2023	89.38	82.13	77.33	-	-	-
Voxel RCNN (Baseline) [29]	L	AAAI2021	92.38	85.29	82.86	95.52	91.25	**88.99**
ContextNet (Ours)	L		92.08	**85.66**	**84.47**	95.47	**91.31**	88.97

refining the proposal stage, we utilize an IoU of 0.01 to eliminate boxes that do not meet the criteria. Training Details, we used the ADAM optimizer for end-to-end training of the entire network. We used a batch size of 16 and trained the network for 80 epochs on 8 GTX 2080 Ti GPUs, which took approximately 5 hours. During the data augmentation phase, we followed the strategy used in SECOND [34]. We set the learning rate to 0.001 and used the cosine annealing learning rate strategy for learning rate decay. For proposal generation, we used the RPN network provided by SECOND to generate high-quality proposals. During the proposal refinement phase, we randomly sampled 128 proposals and randomly sampled N=256 points from the RoI. We considered proposals with a 3D IoU of at least 0.55 with ground truth boxes as positive proposals for box refinement training.

D. COMPARISON WITH STATE-OF-THE-ARTS

1) KITTI BENCHAMRK

In the comparison with state-of-the-arts, we evaluated the performance of our proposed method on the KITTI Benchmark dataset to assess its efficiency and accuracy in object detection tasks.Table 1 illustrates the evaluation results of the mean average precision (mAP) performance of various state-of-the-art 3D object detectors on the KITTI dataset [81]. The data indicates that our proposed method has shown significant enhancements in performance compared to other leading approaches. Specifically, our method demonstrated outstanding mAP performance of 92.08%, 85.66%, 84.47%, 95.47%, 91.31%, and 88.97% respectively. In comparison to the baseline SECOND [34], our method exhibited improvements in AP_{3D} performance by 4.52% mAP, 8.51% mAP, and 10.12% mAP on the car category across three difficulty levels. Additionally, our model outperformed other voxel-based techniques, showing performance enhancements of −0.3%, 0.37%, and 1.61% over the voxel-based method with the best results, Voxel RCNN [29], respectively.

The results in Table 1 also highlight that our model surpassed point-based models and multi-modal models, particularly excelling at the Hard level. Notably, compared to the advanced point-based method PV-RCNN [20] and the multi-modal method EPNet [73], our model achieved a 1.78% and 4.33% higher mAP, respectively. This success can be attributed to our approach of modeling voxel relationships through self-attention and leveraging joint attention to re-weight the point cloud in the Region of Interest (RoI), thereby incorporating rich contextual information to enhance the performance of the detection network. Consequently, our model achieved superior 3D object detection results in sparser point clouds and under more severe occlusions.

2) NUSCENES BENCHAMRK

As shown in Table 2, we conducted comparative experiments on the nuScenes test benchmark. As a unimodal solution,

TABLE 2. Comparison with the SOTA methods on the nuScenes test set. "C.V.", "Motor.", "Ped.", and "T.C." are short for construction vehicle, motorcycle, pedestrian, and traffic cone, respectively.

Method	Publication	mAP	NDS	Car	Truck	C.V.	Bus	Trailer	Barrier	Motor.	Bike	Ped.	T.C.
PointPillars [43]	CVPR2019	30.5	45.3	68.4	23.0	4.1	28.2	23.4	38.9	27.4	1.1	59.7	30.8
InfoFocus [88]	ECCV2020	39.5	39.5	77.9	31.4	10.7	44.8	37.3	47.8	29.0	6.1	63.4	46.5
AFDetV2 [89]	AAAI2022	62.4	68.5	86.3	54.2	26.7	62.5	58.9	71.0	63.8	34.3	85.8	80.1
VISTA [90]	CVPR2022	63.0	69.8	84.4	55.1	25.1	63.7	54.2	71.4	70.0	45.4	82.8	78.5
PointPainting [70]	CVPR2020	46.4	58.1	77.9	35.8	15.8	36.2	37.3	60.2	41.5	24.1	73.3	62.4
MVP [91]	CVPRW2019	66.4	70.5	86.8	58.5	26.1	67.4	57.3	74.8	70.0	49.3	**89.1**	85.0
VP-Net (Voxel) [26]	TGRS2023	57.5	67.5	81.8	52.9	24.3	63.8	56.9	66.2	46.8	32.6	79.4	70.7
PointAugmenting [72]	CVPR2021	66.8	71.0	87.5	57.3	28.0	65.2	60.7	72.6	**74.3**	50.9	87.9	83.6
AutoAlign [84]	IJCAI2022	65.8	70.9	85.9	55.3	29.6	**67.7**	55.6	-	71.5	**51.5**	86.4	-
GraphAlign [35]	ICCV2023	66.5	70.6	**87.6**	57.7	26.1	66.2	57.8	74.1	72.5	49.0	87.2	**86.3**
Far3D [92]	AAAI2024	63.5	68.7	-	-	-	-	-	-	-	-	-	-
VoxelNeXt [93]	CVPR2023	64.5	70.0-	-	-	-	-	-	-	-	-	-	
APVR [94]	TAI2023	58.6	65.9-	-	-	-	-	-	-	-	-	-	
CenterPoint (Baseline) [95]	CVPR2021	58.0	65.5	84.6	51.0	17.5	60.2	53.2	70.9	53.7	28.7	83.4	76.7
ContextNet (Ours)		**67.5**	**71.1**	85.2	**58.9**	**32.4**	67.2	**61.5**	**77.3**	71.3	50.3	86.7	84.4

TABLE 3. Ablation experiments of VSAE and SCR. The results are evaluated with 3D AP calculated for car class on the moderate level.

VSAE	JCSR	$AP_{3D}(\%)$		
		Easy	Mod.	Hard
		87.56	77.21	74.35
✓		89.86	81.27	79.93
	✓	90.07	82.32	81.17
✓	✓	**92.08**	**85.66**	**84.47**

our ContextNet achieved an average precision (AP) of 67.5% and a NuScenes Detection Score (NDS) of 71.1%, surpassing the unimodal state-of-the-art (SOTA) methods [26], [43], [88], [89], [90], [91], [95], as well as the multi-modal SOTA methods [35], [70], [72], [84]. Moreover, it outperformed other SOTA methods in specific categories, such as Truck, Construction Vehicle, Trailer, and Barrier. As a LiDAR-based method, our method outperforms multi-modal and LiDAR-based methods in performance. For example, compared to the SOTA multi-modal methods of AutoAlign [84] and GraphAlign [35], our ContextNet surpasses 1.7%, 1.0% on mAP, 0.2%, and 0.5% on NDS, respectively. Compared to SOTA LiDAR-based methods, VP-Net [26] and CenterPoint [95] exceed 10.0%, 9.5% on mAP, 3.6%, and 5.6% on NDS, respectively.

The success of our ContextNet in exceeding the performance of point-based and multi-modal models, especially at the Hard level, can be attributed to our novel approach. We model voxel relationships using self-attention and employ joint attention to re-weight the point cloud within the Region of Interest (RoI). We incorporate contextual information and significantly enhance the detection network's performance. This approach allowed our model to achieve superior 3D object detection results in sparser point clouds and under more severe occlusions, demonstrating the effectiveness of our method in handling challenging scenarios in object detection tasks.

E. ABLATION STUDY

In order to comprehensively assess each component of our proposed framework, we undertook thorough ablation studies using the KITTI dataset as a reference [81]. The dataset was divided equally for training and validation purposes, with a 1:1 split for all models. Subsequently, we measured the AP_{3D} metric across 40 recall positions on the validation set to provide a detailed evaluation of our framework's performance.

1) EFFECTS OF VSAE AND JCSR MODULES

In this section, we conducted a validation of the effectiveness of two key components: the VSAE module and the JCSR module. The data presented in Table 3 clearly demonstrates that both components play a crucial role in enhancing the detection proficiency of the baseline method SECOND [34]. It is worth noting that while both components contribute to performance improvements, the JCSR module stands out for its more pronounced impact.

Specifically, in comparison to the baseline experiment, our model exhibited enhancements of 2.51%, 5.11%, and 6.82% in mAP performance across the easy, moderate, and hard difficulty levels, respectively. The JCSR module, in particular, yielded a more significant gain effect, underscoring its importance in enhancing model performance.

In contrast, the VSAE module, while delivering more moderate improvements, also played a significant role in enhancing performance. Notably, it resulted in mAP performance improvements of 2.3%, 4.06%, and 5.58% across the easy, moderate, and hard difficulty levels, respectively. These improvements, although slightly more moderate than those achieved by the JCSR module, remain substantial and should not be overlooked in their contribution to overall model enhancement.

2) DISTANCES ANALYSIS

This section evaluates the model's effectiveness across various distance intervals. The distances were categorized

TABLE 4. Comparison of detection accuracy across different distance ranges indicates that the mAP at the moderate level is reported.

Method	VSAE	JCSR	$AP_{3D}(\%)$			
			Overall	0-20m	20-40m	40m-inf
PV-RCNN [20]			84.83	95.12	78.03	39.43
Voxel RCNN [29]			85.29	**95.57**	79.05	41.75
SECOND [34]			77.21	89.98	71.51	34.55
	✓		81.27	91.31	75.83	37.91
		✓	83.33	93.38	77.63	39.25
	✓	✓	**85.66**	95.41	**79.45**	**42.49**

TABLE 5. Comparison of detection accuracy across different hyperparameters of $\mathcal{N}$ demonstrates that the mAP at the moderate level is reported.

Method	$\mathcal{N}$	$AP_{3D}(\%)$		
		Easy	Mod.	Hard
ContextNet (Ours)	128	91.26	83.98	82.63
	256	92.08	**85.66**	**84.47**
	512	**92.38**	85.36	83.69

TABLE 6. Effect of the quantity of voxels K used in attention calculation is examined, which shows that the mAP at the moderate level is reported.

Method	K	$AP_{3D}(\%)$		
		Easy	Mod.	Hard
ContextNet (Ours)	9	90.42	82.59	81.14
	16	91.77	83.82	81.67
	25	91.56	84.03	82.85
	32	**92.08**	**85.66**	**84.47**
	50	91.86	85.35	83.94

into three levels: 0-20m, 20-40m, and 40m-inf. As illustrated in Table 4, our approach demonstrates enhanced performance across different distance ranges. When compared to Voxel RCNN [29] and PV-RCNN [20], our method excels particularly at medium (20-40m) and long distances (40m-inf). This improvement can be primarily attributed to the incorporation of rich contextual information within the modules, enabling the preservation of intricate spatial details within the point cloud data. Consequently, this enhancement empowers the model to accurately infer and predict objects located at greater distances.

3) EFFECT OF DIFFERENT HYPERPARAMETER $\mathcal{N}$

The table presented (Table 5) demonstrates that opting for 256 raw point clouds within the Region of Interest (ROI) yields the optimal results, leading to the model achieving peak performance. As the number of point clouds increases beyond this threshold, the enhancement in mean Average Precision (mAP) appears to plateau, suggesting diminishing returns. Additionally, augmenting the sampling number has shown to have a negligible impact on the overall outcome.

TABLE 7. Compare the parameters, GFLOPs, and inference speed between our ContextNet and the baseline (Voxel RCNN). We tested the FPS using an NVIDIA GTX 2080Ti.

Method	Parameters	FPS	GFLOPs
Baseline (Voxel RCNN [29])	7.59M	22	1545.2
ContextNet (Ours)	8.02M	20	1571.3

4) EFFECT OF THE QUANTITY OF VOXELS K

Table 6 presented in the study illustrates a notable trend in the detection performance of the model concerning the quantity of involved voxels. It is evident that an increase in the number of involved voxels correlates with an improvement in detection accuracy. Specifically, the model exhibits performance gains of 1.66%, 3.07%, and 3.73% when the quantity of involved voxels escalates from 9 to 32. This enhancement can be attributed to the incorporation of rich contextual information for the querying voxel facilitated by self-attention mechanisms.

However, it is noteworthy that a subsequent increase in the quantity of involved voxels from 32 to 50 results in a slight decrease in detection accuracy. This decline could potentially be ascribed to the introduction of noise stemming from the augmentation of voxel quantity. Hence, it becomes evident that the selection of an optimal quantity of involved voxels holds paramount importance in ensuring the optimal performance of the model.

In conclusion, the findings underscore the critical role of striking a balance in the number of involved voxels to achieve the desired detection accuracy. By carefully considering the trade-off between contextual information enrichment and noise introduction, researchers can enhance the model's performance effectively.

5) COMPARE THE PARAMETERS, GFLOPS, AND INFERENCE SPEED

We compared our method with baseline Voxel RCNN in Table 7, particularly in terms of model parameters, FPS, and GFLOPs, which helps to further understand the model. We have added our strategies including VSAE and JCSR on the basis of Voxel RCNN, although the model parameters have slightly increased from 7.59M to 8.02M.

We conducted FPS testing on NVIDIA GTX 2080Ti and found no significant impact, decreasing from 22 to 20. However, as one of the indicators for measuring hardware, GFLOPs (Giga Floating point Operations Per Second) did not show significant changes, ranging from 1542.2 to 1571.3. Overall, our method, as a very simple strategy, is very suitable for addressing the issue of insufficient distance in previous Voxel based systems, and there has been no significant change in GPU testing and model parameter count.

V. CONCLUSION

The study delves into the limitations of current voxel-based two-stage methods and suggests a novel approach incorporating context-dependent information. Unlike existing techniques, the proposed method employs a self-attention mechanism during the voxel encoding stage to establish connections between voxels, facilitating the capture of context information within the specified voxel query range. Furthermore, joint attention is utilized in the proposal refinement stage to amalgamate global context and local channel information within the Region of Interest (RoI), thereby enhancing detection accuracy. The experimental results demonstrate that the proposed method outperforms existing approaches. The primary objective of this research is to introduce innovative concepts for voxel-based 3D object detection.

A. FUTURE WORK

In future research, we aim to delve into self-attention mechanisms to create connections among voxels for the purpose of 3D object detection in multi-modal fusion. Also, in our endeavor to enhance performance, we will investigate the efficacy of integrating visual foundational models, such as Depth Anything [97], into our approach.

ACKNOWLEDGMENT

The authors are grateful that Yanshan University has provided hardware resources, such as GPUs.

REFERENCES

[1] K. Huang, B. Shi, X. Li, X. Li, S. Huang, and Y. Li, "Multi-modal sensor fusion for auto driving perception: A survey," 2022, *arXiv:2202.02703*.
[2] Y. Wang, Q. Mao, H. Zhu, J. Deng, Y. Zhang, J. Ji, H. Li, and Y. Zhang, "Multi-modal 3D object detection in autonomous driving: A survey," *Int. J. Comput. Vis.*, vol. 131, no. 8, pp. 2122–2152, Aug. 2023.
[3] T. Liang, H. Bao, W. Pan, and F. Pan, "Traffic sign detection via improved sparse R-CNN for autonomous vehicles," *J. Adv. Transp.*, vol. 2022, pp. 1–16, Mar. 2022.
[4] T. Liang, H. Bao, W. Pan, X. Fan, and H. Li, "DetectFormer: Category-assisted transformer for traffic scene object detection," *Sensors*, vol. 22, no. 13, p. 4833, Jun. 2022.
[5] T. Liang, H. Bao, W. Pan, and F. Pan, "ALODAD: An anchor-free lightweight object detector for autonomous driving," *IEEE Access*, vol. 10, pp. 40701–40714, 2022.
[6] T. Liang, W. Pan, H. Bao, X. Fan, and H. Li, "Bird's eye view semantic segmentation based on improved transformer for automatic annotation," *KSII Trans. Internet Inf. Syst. (TIIS)*, vol. 17, no. 8, pp. 1996–2015, 2023.
[7] Z. Song, Y. Zhang, Y. Liu, K. Yang, and M. Sun, "MSFYOLO: Feature fusion-based detection for small objects," *IEEE Latin Amer. Trans.*, vol. 20, no. 5, pp. 823–830, May 2022.
[8] Z. Song, L. Wang, G. Zhang, C. Jia, J. Bi, H. Wei, Y. Xia, C. Zhang, and L. Zhao, "Fast detection of multi-direction remote sensing ship object based on scale space pyramid," in *Proc. 18th Int. Conf. Mobility, Sens. Netw. (MSN)*, Dec. 2022, pp. 1019–1024.
[9] Z. Song, P. Wu, K. Yang, Y. Zhang, and Y. Liu, "MsfNet: A novel small object detection based on multi-scale feature fusion," in *Proc. 17th Int. Conf. Mobility, Sens. Netw. (MSN)*, Dec. 2021, pp. 700–704.
[10] G. Zhang, J. Xie, L. Liu, Z. Wang, K. Yang, and Z. Song, "Urformer: Unified representation LiDAR-camera 3D object detection with transformer," in *Proc. Chin. Conf. Pattern Recognit. Comput. Vis. (PRCV)*, 2023, pp. 401–413.
[11] L. Liu, Z. Song, Q. Xia, F. Jia, C. Jia, L. Yang, and H. Pan, "Sparsedet: A simple and effective framework for fully sparse LiDAR-based 3D object detection," 2024, *arXiv:2406.10907*.
[12] J. H. Yoo, Y. Kim, J. Kim, and J. W. Choi, "3D-CVF: Generating joint camera and LiDAR features using cross-view spatial feature fusion for 3D object detection," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2020, pp. 720–736.
[13] L. Wang, Z. Song, X. Zhang, C. Wang, G. Zhang, L. Zhu, J. Li, and H. Liu, "SAT-GCN: Self-attention graph convolutional network-based 3D object detection for autonomous driving," *Knowl.-Based Syst.*, vol. 259, Jan. 2023, Art. no. 110080.
[14] J. Mao, S. Shi, X. Wang, and H. Li, "3D object detection for autonomous driving: A comprehensive survey," 2022, *arXiv:2206.09474*.
[15] R. Qian, X. Lai, and X. Li, "3D object detection for autonomous driving: A survey," *Pattern Recognit.*, vol. 130, Oct. 2022, Art. no. 108796.
[16] E. Arnold, O. Y. Al-Jarrah, M. Dianati, S. Fallah, D. Oxtoby, and A. Mouzakitis, "A survey on 3D object detection methods for autonomous driving applications," *IEEE Trans. Intell. Transp. Syst.*, vol. 20, no. 10, pp. 3782–3795, Oct. 2019.
[17] Z. Song, L. Liu, F. Jia, Y. Luo, G. Zhang, L. Yang, L. Wang, and C. Jia, "Robustness-aware 3D object detection in autonomous driving: A review and outlook," 2024, *arXiv:2401.06542*.
[18] W. Shi and R. Rajkumar, "Point-GNN: Graph neural network for 3D object detection in a point cloud," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 1708–1716.
[19] S. Shi, X. Wang, and H. Li, "PointRCNN: 3D object proposal generation and detection from point cloud," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 770–779.
[20] 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 *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 10526–10535.
[21] Z. Yang, Y. Sun, S. Liu, and J. Jia, "3DSSD: Point-based 3D single stage object detector," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 11037–11045.
[22] C. He, H. Zeng, J. Huang, X.-S. Hua, and L. Zhang, "Structure aware single-stage 3D object detection from point cloud," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 11870–11879.
[23] W. Zheng, W. Tang, S. Chen, L. Jiang, and C.-W. Fu, "CIA-SSD: Confident IoU-aware single-stage object detector from point cloud," in *Proc. AAAI Conf. Artif. Intell.*, 2021, vol. 35, no. 4, pp. 3555–3562.
[24] R. Q. Charles, H. Su, M. Kaichun, and L. J. Guibas, "PointNet: Deep learning on point sets for 3D classification and segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 77–85.
[25] C. R. Qi, L. Yi, H. Su, and L. J. Guibas, "PointNet++: Deep hierarchical feature learning on point sets in a metric space," in *Proc. Annu. Conf. Neural Inf. Process. Syst. (NeurIPS)*, 2017, pp. 5099–5108.
[26] Z. Song, H. Wei, C. Jia, Y. Xia, X. Li, and C. Zhang, "VP-Net: Voxels as points for 3D object detection," *IEEE Trans. Geosci. Remote Sens.*, vol. 1, 2023, Art. no. 5701912.
[27] Z. Song, G. Zhang, J. Xie, L. Liu, C. Jia, S. Xu, and Z. Wang, "VoxelNextFusion: A simple, unified, and effective voxel fusion framework for multimodal 3-D object detection," *IEEE Trans. Geosci. Remote Sens.*, vol. 61, 2023, Art. no. 5705412.
[28] Y. Zhou and O. Tuzel, "VoxelNet: End-to-end learning for point cloud based 3D object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 4490–4499.
[29] 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 *Proc. AAAI Conf. Artif. Intell.*, 2021, pp. 1201–1209.

[30] H. Kuang, B. Wang, J. An, M. Zhang, and Z. Zhang, "Voxel-FPN: Multi-scale voxel feature aggregation for 3D object detection from LiDAR point clouds," *Sensors*, vol. 20, no. 3, p. 704, Jan. 2020.

[31] Z. Liu, H. Tang, Y. Lin, and S. Han, "Point-voxel CNN for efficient 3D deep learning," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 32, 2019, pp. 1–20.

[32] 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," *Int. J. Comput. Vis.*, vol. 131, no. 2, pp. 531–551, Feb. 2023.

[33] Z. Yang, Y. Sun, S. Liu, X. Shen, and J. Jia, "STD: Sparse-to-dense 3D object detector for point cloud," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 1951–1960.

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

[35] Z. Song, H. Wei, L. Bai, L. Yang, and C. Jia, "GraphAlign: Enhancing accurate feature alignment by graph matching for multi-modal 3D object detection," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2023, pp. 3358–3369.

[36] Z. Song, C. Jia, L. Yang, H. Wei, and L. Liu, "GraphAlign++: An accurate feature alignment by graph matching for multi-modal 3D object detection," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 34, no. 4, pp. 2619–2632, Apr. 2024.

[37] K. Yang and Z. Song, "Deep learning-based object detection improvement for fine-grained birds," *IEEE Access*, vol. 9, pp. 67901–67915, 2021.

[38] W. Xiang, Z. Song, G. Zhang, and X. Wu, "Birds detection in natural scenes based on improved faster RCNN," *Appl. Sci.*, vol. 12, no. 12, p. 6094, Jun. 2022.

[39] L. Wang, X. Zhang, Z. Song, J. Bi, G. Zhang, H. Wei, L. Tang, L. Yang, J. Li, C. Jia, and L. Zhao, "Multi-modal 3D object detection in autonomous driving: A survey and taxonomy," *IEEE Trans. Intell. Vehicles*, vol. 1, no. 1, pp. 1–19, Sep. 2023.

[40] Y. Wang, Q. Mao, H. Zhu, J. Deng, Y. Zhang, J. Ji, H. Li, and Y. Zhang, "Multi-modal 3D object detection in autonomous driving: A survey," *Int. J. Comput. Vis.*, vol. 131, no. 8, pp. 2122–2152, Aug. 2023.

[41] L. Xie, C. Xiang, Z. Yu, G. Xu, Z. Yang, D. Cai, and X. He, "PI-RCNN: An efficient multi-sensor 3D object detector with point-based attentive cont-conv fusion module," in *Proc. AAAI Conf. Artif. Intell.*, 2020, pp. 12460–12467.

[42] J. Li, S. Luo, Z. Zhu, H. Dai, A. S. Krylov, Y. Ding, and L. Shao, "3D IoU-net: IoU guided 3D object detector for point clouds," 2020, *arXiv:2004.04962*.

[43] A. H. Lang, S. Vora, H. Caesar, L. Zhou, J. Yang, and O. Beijbom, "PointPillars: Fast encoders for object detection from point clouds," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 12689–12697.

[44] S. Shi, Z. Wang, J. Shi, X. Wang, and H. Li, "From points to parts: 3D object detection from point cloud with part-aware and part-aggregation network," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 43, no. 8, pp. 2647–2664, Aug. 2021.

[45] J. S. K. Hu, T. Kuai, and S. L. Waslander, "Point density-aware voxels for LiDAR 3D object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 8459–8468.

[46] Z. Li, F. Wang, and N. Wang, "LiDAR R-CNN: An efficient and universal 3D object detector," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 7542–7551.

[47] J. Noh, S. Lee, and B. Ham, "HVPR: Hybrid voxel-point representation for single-stage 3D object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 14600–14609.

[48] J. Mao, M. Niu, H. Bai, X. Liang, H. Xu, and C. Xu, "Pyramid R-CNN: Towards better performance and adaptability for 3D object detection," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 2703–2712.

[49] G. Shi, R. Li, and C. Ma, "PillarNet: Real-time and high-performance pillar-based 3D object detection," in *Proc. Eur. Conf. Comput. Vis.*, 2022, pp. 35–52.

[50] P. Sun, M. Tan, W. Wang, C. Liu, F. Xia, Z. Leng, and D. Anguelov, "Swformer: Sparse window transformer for 3D object detection in point clouds," in *Proc. Eur. Conf. Comput. Vis.*, Oct. 2022, pp. 1–20.

[51] J. Li, C. Luo, X. Yang, and Q. Qcraft, "Pillarnext: Rethinking network designs for 3D object detection in LiDAR point clouds," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Oct. 2023, pp. 17567–17576.

[52] X. Feng, H. Du, H. Fan, Y. Duan, and Y. Liu, "Seformer: Structure embedding transformer for 3D object detection," in *Proc. AAAI Conf. Artif. Intell.*, vol. 37, 2023, pp. 632–640.

[53] L. Fan, Z. Pang, T. Zhang, Y.-X. Wang, H. Zhao, F. Wang, N. Wang, and Z. Zhang, "Embracing single stride 3D object detector with sparse transformer," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 8448–8458.

[54] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, and B. Guo, "Swin transformer: Hierarchical vision transformer using shifted windows," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 9992–10002.

[55] Y. Zhang, Q. Hu, G. Xu, Y. Ma, J. Wan, and Y. Guo, "Not all points are equal: Learning highly efficient point-based detectors for 3D LiDAR point clouds," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 18931–18940.

[56] C. Chen, Z. Chen, J. Zhang, and D. Tao, "SASA: Semantics-augmented set abstraction for point-based 3D object detection," in *Proc. AAAI Conf. Artif. Intell.*, 2022, vol. 36, no. 1, pp. 221–229.

[57] L. Wang, X. Zhang, F. Zhao, C. Wu, Y. Wang, Z. Song, L. Yang, B. Xu, J. Li, and S. S. Ge, "Fuzzy-NMS: Improving 3D object detection with fuzzy classification in NMS," *IEEE Trans. Intell. Vehicles*, vol. 1, no. 1, pp. 1–15, Sep. 2024.

[58] T. Wang, X. Zhu, J. Pang, and D. Lin, "FCOS3D: Fully convolutional one-stage monocular 3D object detection," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. Workshops (ICCVW)*, Oct. 2021, pp. 913–922.

[59] Y. Liu, T. Wang, X. Zhang, and J. Sun, "PETR: Position embedding transformation for multi-view 3D object detection," in *Proc. 17th Eur. Conf. Comput. Vis. (ECCV)*, 2022, pp. 531–548.

[60] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko, "End-to-end object detection with transformers," in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 213–229.

[61] Y. Liu, J. Yan, F. Jia, S. Li, A. Gao, T. Wang, and X. Zhang, "PETRv2: A unified framework for 3D perception from multi-camera images," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2023, pp. 3262–3272.

[62] J. Huang, G. Huang, Z. Zhu, Y. Ye, and D. Du, "BEVDet: High-performance multi-camera 3D object detection in bird-eye-view," 2021, *arXiv:2112.11790*.

[63] Z. Li, W. Wang, H. Li, E. Xie, C. Sima, T. Lu, Y. Qiao, and J. Dai, "Bevformer: Learning bird's-eye-view representation from multi-camera images via spatiotemporal transformers," in *Proc. Eur. Conf. Comput. Vis.*, 2022, pp. 1–18.

[64] X. Zhu, W. Su, L. Lu, B. Li, X. Wang, and J. Dai, "Deformable DETR: Deformable transformers for end-to-end object detection," 2020, *arXiv:2010.04159*.

[65] C. Chen, L. Z. Fragonara, and A. Tsourdos, "RoIFusion: 3D object detection from LiDAR and vision," *IEEE Access*, vol. 9, pp. 51710–51721, 2021.

[66] J. Ku, M. Mozifian, J. Lee, A. Harakeh, and S. L. Waslander, "Joint 3D proposal generation and object detection from view aggregation," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Oct. 2018, pp. 1–8.

[67] X. Chen, H. Ma, J. Wan, B. Li, and T. Xia, "Multi-view 3D object detection network for autonomous driving," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 6526–6534.

[68] A. Paigwar, D. Sierra-Gonzalez, Ö. Erkent, and C. Laugier, "Frustum-PointPillars: A multi-stage approach for 3D object detection using RGB camera and LiDAR," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. Workshops (ICCVW)*, Oct. 2021, pp. 2926–2933.

[69] S. Pang, D. Morris, and H. Radha, "CLOCs: Camera-LiDAR object candidates fusion for 3D object detection," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Oct. 2020, pp. 10386–10393.

[70] S. Vora, A. H. Lang, B. Helou, and O. Beijbom, "PointPainting: Sequential fusion for 3D object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 4603–4611.

[71] S. Xu, F. Li, Z. Song, J. Fang, S. Wang, and Z.-X. Yang, "Multi-sem fusion: Multimodal semantic fusion for 3-D object detection," *IEEE Trans. Geosci. Remote Sens.*, vol. 62, 2024, Art. no. 5703114.

[72] C. Wang, C. Ma, M. Zhu, and X. Yang, "PointAugmenting: Cross-modal augmentation for 3D object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 11789–11798.

[73] Z. Liu, T. Huang, B. Li, X. Chen, X. Wang, and X. Bai, "EPNet++: Cascade bi-directional fusion for multi-modal 3D object detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 7, pp. 1–18, Nov. 2022.

[74] Z. Song, L. Yang, S. Xu, L. Liu, D. Xu, C. Jia, F. Jia, and L. Wang, "GraphBEV: Towards robust BEV feature alignment for multi-modal 3D object detection," 2024, *arXiv:2403.11848*.

[75] Z. Song, F. Jia, H. Pan, Y. Luo, C. Jia, G. Zhang, L. Liu, Y. Ji, L. Yang, and L. Wang, "ContrastAlign: Toward robust BEV feature alignment via contrastive learning for multi-modal 3D object detection," 2024, *arXiv:2405.16873*.

[76] D. Xu, H. Li, Q. Wang, Z. Song, L. Chen, and H. Deng, "M2DA: Multi-modal fusion transformer incorporating driver attention for autonomous driving," 2024, *arXiv:2403.12552*.

[77] L. Bai, C. Jia, Z. Song, and C. Cui, "VGA: Vision and graph fused attention network for rumor detection," 2024, *arXiv:2401.01759*.

[78] Z. Song, G. Zhang, L. Liu, L. Yang, S. Xu, C. Jia, F. Jia, and L. Wang, "RoboFusion: Towards robust multi-modal 3D object detection via SAM," 2024, *arXiv:2401.03907*.

[79] J. Bi, H. Wei, G. Zhang, K. Yang, and Z. Song, "DyFusion: Cross-attention 3D object detection with dynamic fusion," *IEEE Latin Amer. Trans.*, vol. 22, no. 2, pp. 106–112, Feb. 2024.

[80] X. Zhang, L. Wang, J. Chen, C. Fang, L. Yang, Z. Song, G. Yang, Y. Wang, X. Zhang, J. Li, Z. Li, Q. Yang, Z. Zhang, and S. Sam Ge, "Dual radar: A multi-modal dataset with dual 4D radar for autonomous driving," 2023, *arXiv:2310.07602*.

[81] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? The KITTI vision benchmark suite," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2012, pp. 3354–3361.

[82] H. Caesar, V. Bankiti, A. H. Lang, S. Vora, V. E. Liong, Q. Xu, A. Krishnan, Y. Pan, G. Baldan, and O. Beijbom, "NuScenes: A multimodal dataset for autonomous driving," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 11618–11628.

[83] T. Huang, Z. Liu, X. Chen, and X. Bai, "EPNet: Enhancing point features with image semantics for 3D object detection," in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 35–52.

[84] Z. Chen, Z. Li, S. Zhang, L. Fang, Q. Jiang, F. Zhao, B. Zhou, and H. Zhao, "AutoAlign: Pixel-instance feature aggregation for multi-modal 3D object detection," 2022, *arXiv:2201.06493*.

[85] H. Zhu, J. Deng, Y. Zhang, J. Ji, Q. Mao, H. Li, and Y. Zhang, "VPFNet: Improving 3D object detection with virtual point based LiDAR and stereo data fusion," 2021, *arXiv:2111.14382*.

[86] G. Xie, Z. Chen, M. Gao, M. Hu, and X. Qin, "PPF-Det: Point-pixel fusion for multi-modal 3D object detection," *IEEE Trans. Intell. Transp. Syst.*, vol. 25, no. 6, pp. 5598–5611, Jun. 2024.

[87] I. Koo, I. Lee, S.-H. Kim, H.-S. Kim, W.-J. Jeon, and C. Kim, "PG-RCNN: Semantic surface point generation for 3D object detection," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2023, pp. 18142–18151.

[88] J. Wang, S. Lan, M. Gao, and L. S. Davis, "InfoFocus: 3D object detection for autonomous driving with dynamic information modeling," in *Proc. Comput. Vis. 16th Eur. Conf.*, Aug. 2020, pp. 405–420.

[89] Y. Hu, Z. Ding, R. Ge, W. Shao, L. Huang, K. Li, and Q. Liu, "AFDetV2: Rethinking the necessity of the second stage for object detection from point clouds," in *Proc. AAAI Conf. Artif. Intell.*, vol. 36, no. 1, 2022, pp. 969–979.

[90] S. Deng, Z. Liang, L. Sun, and K. Jia, "VISTA: Boosting 3D object detection via dual cross-VIew SpaTial attention," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 8438–8447.

[91] G. P. Meyer, J. Charland, D. Hegde, A. Laddha, and C. Vallespi-Gonzalez, "Sensor fusion for joint 3D object detection and semantic segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2019, pp. 1230–1237.

[92] X. Jiang, S. Li, Y. Liu, S. Wang, F. Jia, T. Wang, L. Han, and X. Zhang, "Far3D: Expanding the horizon for surround-view 3D object detection," 2023, *arXiv:2308.09616*.

[93] Y. Chen, J. Liu, X. Zhang, X. Qi, and J. Jia, "Voxelnext: Fully sparse voxelnet for 3D object detection and tracking," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Aug. 2023, pp. 21674–21683.

[94] J. Cao, C. Tao, Z. Zhang, Z. Gao, X. Luo, S. Zheng, and Y. Zhu, "Accelerating point-voxel representation of 3D object detection for automatic driving," *IEEE Trans. Artif. Intell.*, vol. 1, no. 1, pp. 1–13, Sep. 2023.

[95] T. Yin, X. Zhou, and P. Krähenbuhl, "Center-based 3D object detection and tracking," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 11779–11788.

[96] O. D. Team. (2020). *Openpcdet: An Open-source Toolbox for 3D Object Detection From Point Clouds*. [Online]. Available: https://github.com/open-mmlab/OpenPCDet

[97] L. Yang, B. Kang, Z. Huang, X. Xu, J. Feng, and H. Zhao, "Depth anything: Unleashing the power of large-scale unlabeled data," in *Proc. CVPR*, 2024, pp. 1–13.

CAIYAN PEI was born in Shenze, Hebei, China, in 1978. She received the M.S. degree from Yanshan University, China, in 2009. She is currently teaching with the School of Mathematics and Information Technology, Hebei Normal University of Science and Technology, China. Her research interests include computer vision, data analysis and mining, and computer education.

SHUAI ZHANG was born in Changli, Hebei, China, in 1975. He is currently pursuing the Ph.D. degree with the School of Information Science and Engineering, Yanshan University, China. He teaches with the School of Mathematics and Information Science and Technology, Hebei Normal University of Science and Technology, China, with the title of Associate Professor. His research interests include computer vision and data analysis and mining.

LIJUN CAO was born in Qinhuangdao, Hebei, China, in 1971. She is currently a Professor with the School of Mathematics and Information Science and Technology, Hebei Normal University of Science and Technology, China. Her research interests include computer education, data mining, and data analysis and processing.

LIQIANG ZHAO was born in Funing, Hebei, China, in 1968. He received the B.S. degree in applied mathematics from Jilin University, in 1990, and the Ph.D. degree in computer science and technology from Yanshan University, in 2010. He is currently a Professor with the School of Mathematics and Information Science and Technology, Hebei Normal University of Science and Technology, China. His research interests include intelligent computing, robotics, and the Internet of Things (IoT) engineering.