sustainability

MDPI

Article

DFA-SAT: Dynamic Feature Abstraction with Self-Attention-Based 3D Object Detection for Autonomous Driving

Husnain Mushtaq [1], Xiaoheng Deng [1,*], Mubashir Ali [2], Babur Hayat [3] and Hafiz Husnain Raza Sherazi [4]

[1] School of Computer Science and Engineering, Central South University, Changsha 410083, China; husnainmushtaq911@gmail.com
[2] School of Computer Science, University of Birmingham, Birmingham B15 2TT, UK; mubashir.ali@unibg.it
[3] Department of Computer Science, University of Chenab, Gujrat 50700, Pakistan; babur@cs.uchenab.edu.pk
[4] School of Computing and Engineering, University of West London, London W5 5RF, UK; hafiz.sherazi@uwl.ac.uk
* Correspondence: dxh@csu.edu.cn

Abstract: Autonomous vehicles (AVs) play a crucial role in enhancing urban mobility within the context of a smarter and more connected urban environment. Three-dimensional object detection in AVs is an essential task for comprehending the driving environment to contribute to their safe use in urban environments. Existing 3D LiDAR object detection systems lose many critical point features during the down-sampling process and neglect the crucial interactions between local features, providing insufficient semantic information and leading to subpar detection performance. We propose a dynamic feature abstraction with self-attention (DFA-SAT), which utilizes self-attention to learn semantic features with contextual information by incorporating neighboring data and focusing on vital geometric details. DFA-SAT comprises four modules: object-based down-sampling (OBDS), semantic and contextual feature extraction (SCFE), multi-level feature re-weighting (MLFR), and local and global features aggregation (LGFA). The OBDS module preserves the maximum number of semantic foreground points along with their spatial information. SCFE learns rich semantic and contextual information with respect to spatial dependencies, refining the point features. MLFR decodes all the point features using a channel-wise multi-layered transformer approach. LGFA combines local features with decoding weights for global features using matrix product keys and query embeddings to learn spatial information across each channel. Extensive experiments using the KITTI dataset demonstrate significant improvements over the mainstream methods SECOND and PointPillars, improving the mean average precision (AP) by 6.86% and 6.43%, respectively, on the KITTI test dataset. DFA-SAT yields better and more stable performance for medium and long distances with a limited impact on real-time performance and model parameters, ensuring a transformative shift akin to when automobiles replaced conventional transportation in cities.

Keywords: smart cities; 3D object dejection; semantic features leaning; self-attention

Citation: Mushtaq, H.; Deng, X.; Ali, M.; Hayat, B.; Raza Sherazi, H.H. DFA-SAT: Dynamic Feature Abstraction with Self-Attention-Based 3D Object Detection for Autonomous Driving. *Sustainability* **2023**, *15*, 13667. https://doi.org/10.3390/su151813667

Academic Editors: Juneyoung Park and Marc A. Rosen

Received: 9 July 2023
Revised: 6 September 2023
Accepted: 7 September 2023
Published: 13 September 2023

1. Introduction

Smart sustainable cities use ICT for efficient operations, information sharing, better government services, and citizen well-being, prioritizing technological efficiency over availability for improved urban life [1–4]. Autonomous vehicles offer immersive user experiences, shaping future human–machine interactions in smart cities [5,6]. Mobility as a service is set to transform urban mobility in terms of sustainability [7]. Cities seek smart mobility solutions to address transport issues [8]. AVs' benefits drive their adoption, mitigating safety concerns. AVs promise traffic improvements, enhanced public transport, safer streets, and better quality of life in eco-conscious digital cities [9].

At the core of AV technology lies 3D object detection, a fundamental capability enabling AVs to perceive their surroundings in three dimensions. This 3D object detection is vital for safe autonomous vehicle navigation in smart cities [10,11]. It identifies and comprehends surrounding objects in 3D, enabling obstacle avoidance, path planning, and collision prevention [12]. Advancements in this technology enhance urban life through improved autonomous vehicle perception [13,14]. Autonomous vehicles are equipped with various sensors, including cameras, LiDAR (light detection and ranging), radar, and sometimes ultrasonic sensors. These sensors capture data about the surrounding environment [15].

Recent advancements in autonomous driving technology have significantly propelled the development of sustainable smart cities [16–18]. Notably, 3D object detection has emerged as a pivotal element within autonomous vehicles, forming the basis for efficient planning and control processes in alignment with smart city principles of optimization and enhancing citizens' quality of life, particularly in ensuring the safe navigation of autonomous vehicles (AVs) [19–21]. LiDAR, an active sensor utilizing laser beams to scan the environment, is extensively integrated into AVs to provide 3D perception in urban environments. Various autonomous driving datasets, such as KITTI, have been developed to enable mass mobility in smart cities [22,23]. Although 3D LiDAR point cloud data are rich in depth and spatial information and less susceptible to lighting variations, it possesses irregularities and sparseness, particularly at longer distances, which can jeopardize the safety of pedestrians and cyclists. Traditional methods for learning point cloud features struggle to comprehend the geometrical characteristics of smaller and distant objects in AVs [24,25].

To overcome geometric challenges and facilitate the use of deep neural networks (DNNs) for processing 3D smart city datasets to ensure safe autonomous vehicle (AV) navigation, custom discretization or voxelization techniques are employed [26–34]. These methods convert 3D point clouds into voxel representations, enabling the application of 2D or 3D convolutions. However, they may compromise geometric data and suffer from quantization loss and computational bottlenecks, posing sustainability challenges for AVs in smart cities. Region proposal network (RPN) backbones exhibit high accuracy and recall but struggle with average precision (AP), particularly for distant or smaller objects. The poor AP score hinders AV integration in sustainable smart cities due to its direct impact on object detection at varying distances [35,36].

Most RPN backbones, including region proposal networks, rely on convolutional neural networks (CNNs) for Euclidean data feature extraction [34,37]. However, CNNs are ill-suited for handling unstructured point clouds [38]. To address this, self-attention mechanisms from transformers are introduced to capture long-range dependencies and interactions, enhancing distant object representation and reducing false negatives [2,39,40]. By combining self-attention with CNNs, the performance of 3D object detection in AVs can be enhanced, even with limited point cloud data [2,41,42]. The proposed DFA-SAT approach shows promising results, addressing smart city challenges such as urban space management, pedestrian and cyclist safety, and overall quality of life improvement, aligning with eco-conscious city development goals. Figure 1 illustrates DFA-SAT's performance with a reduced number of point features.

This study aims to enhance 3D object detection in autonomous vehicles (AVs) to address the challenges posed by smart cities, including pedestrian and cyclist safety and reducing vehicle collisions [6,8,18]. It emphasizes the importance of foreground global points for learning better semantic and contextual information among points, a crucial aspect of 3D object detection. The study aims to overcome the limitations caused by insufficient semantic information in point clouds, improving AVs' 3D object detection capabilities, which is essential for their adoption in smart cities [9,11]. To achieve this, two key observations are made. First, a unified module can be developed to address weak semantic information by leveraging both voxel-based and point-based methods. Second, enhancing interactions between global and local object features can promote better feature association. The proposed solution, called dynamic feature abstraction with self-

attention (DFA-SAT), combines CNNs and self-attention mechanisms to augment semantic information in both voxel-based and point-based methods. The proposed approach aims to improve the effectiveness of 3D object detection by addressing the issue of insufficient semantic information.

DFA-SAT is composed of four primary components: object-based down-sampling (OBDS), semantic and contextual features extraction (SCFE), multi-level feature re-weighting (MLFR), and local and global features aggregation (LGFA). The OBDS module preserves more semantic foreground points based on the basis of spatial information as shown in Figure 2. SEFE learns rich semantic and contextual information with respect to spatial dependencies to refine the local point features information. MLFR decodes all the point features using the channel-wise multi-layered transformer approach to enhance the relationship among local features. It adjusts the weights of these relationships, emphasizing the most significant connections. In scenarios with sparse point clouds, distant points tend to be far apart from their neighbors, potentially hindering detection accuracy. LGFA combines local features with decoding weights for global features using matrix product key and query embedding to learn the spatial information across each channel. Figure 3 illustrates DFA-SAT, and Figure 4 demonstrates how it re-weights local and global encoded features.

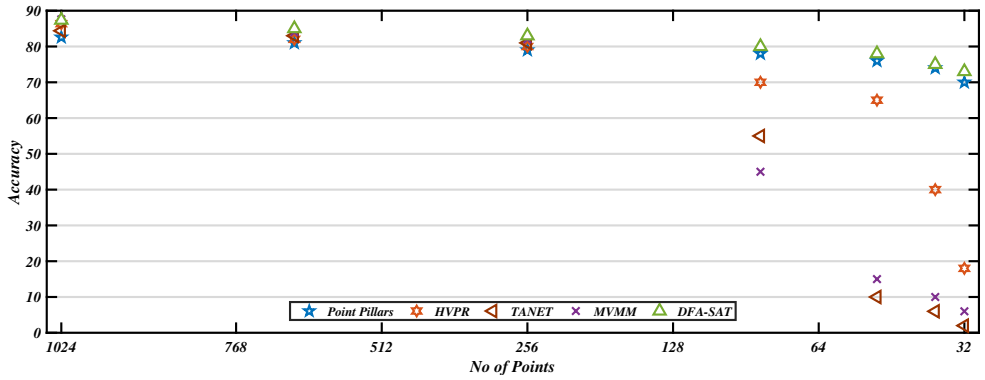

Figure 1. Comparison between various methods under varying point features.

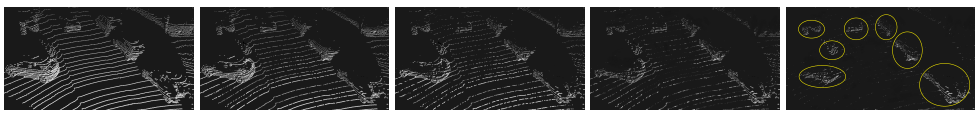

Figure 2. Object-based down-sampling by passing the point features from multiple set abstraction layers. The preserved points are further fed into the progressive DFA-SAT modules for semantic feature learning.

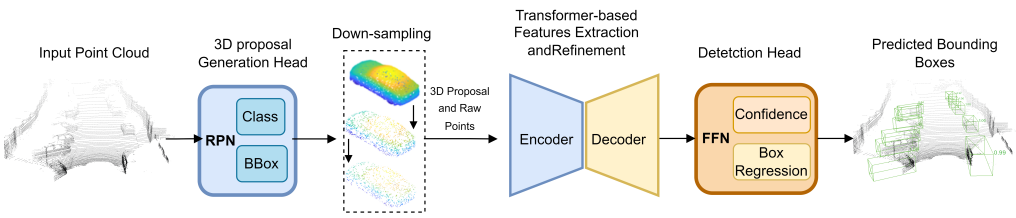

Figure 3. The proposed DFA-SAT is illustrated. It has several abstraction layers for down-sampling objectives to reduce the foreground global features in an engineered manner to control the computational and memory costs. Then, the abstract points are fed into the transformer–encoder module along with their corresponding proposal to perform, proposal-to-point embedding. It learns global and contextual information of points and transfers them to the transformer-based decoding module. The encoded points features are decoded and re-weighted with the global and local features for accurate prediction and bounding box regression to complete the end-to-end learning process.

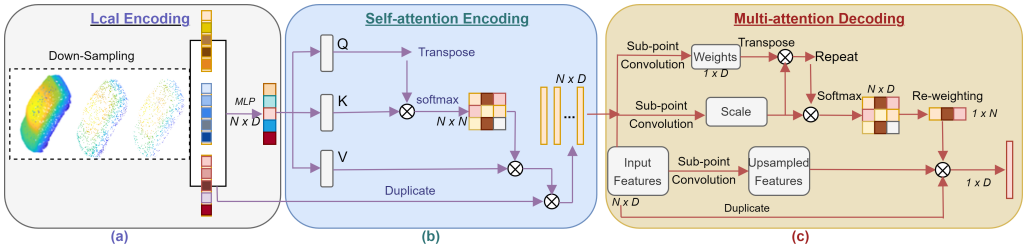

Figure 4. (**a**) The proposal-to-point embedding module is represented by a local encoding. (**b**) A self-attention encoding module performs proposal-aware refinement. A multi-head self-attention encoding layer, which is followed by a feed-forward network (FFN) with residual structure, receives the proposal-guided point features to learn rich semantic and contextual information and spatial dependencies to refine the point features. (**c**) The encoded point features are decoded and re-weighted with the aggregation of the global and local features using a linear projection method. We apply a straightforward approach to compute the decoding weights for each channel instead of overall aggregation at the final stage of the decoder.

To validate the effectiveness of the proposed DFA-SAT module, it was integrated into popular baseline algorithms such as SECOND [34] and PointPillars [37] which provide a base into which to incorporate 3D object detection in AVs to achieve the perceived benefits of smart mobility [2]. Through comprehensive experiments conducted on the widely recognized dataset KITTI [43], the results substantiate the benefits of DFA-SAT. KITTI and similar datasets play a significant role in the development of autonomous vehicles, which are integral to the advancement of smart cities' transportation infrastructure and sustainability goals [44]. Our module enhances the extraction of semantic information and significantly improves detection accuracy in AVs, especially for objects located at medium and long distances, to increase the safety of cyclists and pedestrians in sustainable smart cities. Importantly, the incorporation of the DFA-SAT module has a minimal impact on both the number of model parameters required and the run-time performance. In summary, the key contributions of this research can be outlined as follows:

1. We propose DFA-SAT, a versatile module that improves the detection of 3D objects by preserving maximum foreground features and enhancing weak semantic information of objects around AVs.
2. DFA-SAT performs semantic and contextual feature extraction and decodes these features to refine these relationships by assigning weights to meaningful connections, thus reinforcing their importance.
3. This module can be seamlessly integrated into both voxel-based and point-based methods.
4. Empirical evaluations on the benchmark dataset KITTI. We validate its efficacy in improving detection accuracy, especially for distant and sparse objects, to contribute to sustainability in urban environments.

2. Literature Review

2.1. Sustainable Transportation and Urban Planning

Sustainability has become a paramount concern across industries, with particular focus on the transportation sector. Numerous studies have addressed the implications of autonomous vehicles (AVs) and their potential to revolutionize urban living in smart cities [1–9,11]. Shi et al. [2] introduced a semantic understanding framework that enhances detection accuracy and scene comprehension in smart cities. Yigitcanlar et al. [6] highlighted the need for urban planners and managers to formulate AV strategies for addressing the challenges of vehicle automation in urban areas. Manfreda et al. [8] emphasized that the perceived benefits of AVs play a significant role in their adoption, especially when it comes to safety concerns. Campisi et al. [9] discussed the potential of the AV revolution to improve traffic flow, enhance public transport, optimize urban space, and increase safety for pedestrians and cyclists, ultimately enhancing the quality of life in cities. Duarte et al. [10]

explored the impact of AVs on the road infrastructure and how they could reshape urban living and city planning, akin to the transformative shift brought about by automobiles in the past. Heinrichs et al. [11] delved into the unique characteristics and prospective applications of autonomous transportation, which has the potential to influence land use and urban planning in distinct ways. Stead et al. [18] conducted scenario studies to analyze the intricate effects of AVs on urban structure, including factors like population density, functional diversity, urban layout, and accessibility to public transit. Li et al. [26] proposed a deep learning method combining LiDAR and camera data for precise object detection, while Seuwou et al. focused on smart mobility initiatives, emphasizing the significance of CAVs in sustainable development within intelligent transportation systems. Seuwou et al. [45] present a study that examines smart mobility initiatives and challenges within smart cities, focusing on connected vehicles and AVs. Xu et al. [46] introduced a fusion strategy utilizing LiDAR, cameras, and radar to enhance object detection in dense urban areas. These studies collectively underscore the importance of developing 3D object detection methods to ensure safe and efficient transportation systems in smart cities, addressing critical sustainability challenges.

2.2. Point Cloud Representations for 3D Object Detection

LiDAR is vital for AVs, generating unstructured, unordered, and irregular point clouds. Processing these raw points conventionally is challenging. Numerous 3D object detection methods have emerged in recent years [2,26–31,33,34,37,47–51]. These methods are categorized based on their approach to handling 3D LiDAR point cloud input.

2.2.1. Voxel-Based Methods

Studies have aimed to convert irregular point clouds into regular voxel grids and use CNNs to learn geometric patterns [25,30,34,37]. Early research used high-density voxelization and CNNs for voxel data analysis [26,50,51]. Yan et al. introduced the SECOND architecture for improved memory and computational efficiency using 3D sub-manifold sparse convolution [34]. PointPillars simplified voxel representation to pillars [37]. Existing single-stage and two-stage detectors often lack accuracy, especially for small objects [29,32]. ImVoxelNet by Danila et al. increased the memory and computational costs for image to voxel projection [25]. Zhou et al. transformed point clouds into regularly arranged 3D voxels, adding 3D CNN for object detection [30]. Noh et al. integrated voxel-based and point-based features for efficient single-stage 3D object detection [50]. Shi et al. proposed a voxel-based roadside LiDAR feature encoding module that voxelizes and projects raw point clouds into BEV for dense feature representation with reduced computational overhead [2]. Voxel-based approaches offer reasonable 3D object detection performance with efficiency but may suffer from quantization loss and structural complexity, making optimal resolution determination challenging for local geometry and related contexts.

2.2.2. Point-Based Methods

Different to voxel-based methods, point-based methods generate the 3D objects by direct learning of unstructured geometry from raw point clouds [28,49]. To deal with the unordered nature of 3D point clouds, point-based methods incorporate PointNet [48] and its different variants [29,39] to aggregate the point-wise features employing symmetric functions. Shi et al. [29] presented a regional proposal two-staged 3D object detection framework: Point-RCCN. This method works in quite an interesting way as it generates object proposals from foreground point segments and then exploits the local spatial and semantic features to regress the high-quality 3D bounding boxes.

Qi et al. [52] proposed voteNet, a deep Hough voting-based one-stage point 3D object detector to predict the centroid of an instance. Yang et al. [53] proposed 3DSSD, a single-staged 3D object detection framework. It uses farthest point sampling (FPS), a very popular approach, and Euclidean space as a fusion sampling strategy. PointGNN [54] is a generalized graph neural network for 3D object detection. Point-based methods are

not as resource intensive as voxel-based methods. Point-based methods are intuitive and straightforward and do not require any extra pre-processing and simply take raw point clouds as input. The drawback of point-based methods is their limited efficiency and insufficient learning ability.

2.2.3. Weak Semantic Information for 3D Object Detection

In autonomous driving, point cloud sampling often yields sparse coverage. For example, when aligning KITTI dataset color images with raw point clouds, only about 3% of pixels have corresponding points [42,55]. This extreme sparsity challenges high-level semantic perception from point clouds. Existing 3D object detection methods [29–31,33,34,37] typically extract local features from raw point clouds but struggle to capture comprehensive feature information and feature interactions. Sparse point cloud data, limitations in local feature extraction, and insufficient feature interactions lead to weak semantic information in 3D object detection models, notably affecting performance for distant and smaller objects.

Both voxel-based [30,34,37] and point-based [29,48] methods face weak semantic information challenges in sparse point clouds. For example, Yukang et al. [56] proposed a complex approach with focus sparse convolution and multi-modal expansion but with high computational costs and complexity. Qiuxiao et al. [57] introduced a sparse activation map (SAM) for voxel-based techniques, and Pei et al. [58] developed range sparse net (RSN) for real-time 3D object detection from dense images but with spatial depth information issues. Mengye et al. [59] introduced a sparse blocks network (SBNet) for voxel-based methods. Shi et al. [2] incorporated multi-head self-attention and deformable cross-attention for interacting vehicles. Existing methods focus on downstream tasks, under-utilize object feature information, and are often limited to either voxel-based or point-based models, reducing their generalizability.

2.3. Self-Attention Mechanism

The recent success of transformers in various computer vision domains [42,60] has led to a new paradigm in object detection. Transformers have proven to be highly effective in learning local context-aware representations. DETR [60] introduced this paradigm by treating object detection as a set prediction problem and employed transformers with parallel decoding to detect objects in 2D images. The application of point transformers [42] in self-attention networks for 3D point cloud processing and object classification has gained attention recently. Particularly, the point cloud transformer (PCT) framework [21] has been utilized for learning from point clouds and improving embedded input. PCT incorporates essential functionalities such as farthest-point sampling and nearest-neighbor searching. In the context of 3D object detection, Bhattacharyya et al. [61] proposed two variants of self-attention for contextual modeling. These variants augment convolutional features with self-attention features to enhance the overall performance of 3D object detection. Additionally, Jiageng et al. [62] introduced voxel transformer (VoTr), a novel and effective voxel-based transformer backbone specifically designed for point cloud 3D object detection. Shi et al. [2] employed multi-attention and cross-attention to establish a dense feature representation through feature re-weighting.

Overall, these studies highlight the importance of 3D object detection techniques in enhancing the perception capabilities of autonomous vehicles and contribute to the development of safer and more efficient transportation systems in smart cities. Our approach distinguishes itself from previous methods by incorporating the concepts of dynamic feature abstraction with self-attention (DFA-SAT) to preserve maximum foreground features and to improve the interaction between features to significantly reduce the number of accidents on the road and ensure cyclist and pedestrian safety when AVs meet. It utilizes self-attention to learn semantic features with contextual information by using neighboring information and focusing on vital relationships among the local and global point features. This novel approach leads to significant improvements in the detection of 3D objects and the extraction of meaningful semantic information.

3. Methodology

3.1. Overview

Three-dimensional object detection by AVs plays a crucial role in enhancing the capabilities and safety of smart cities. Existing 3D object detection networks seldom focus on point-wise dense 3D semantic segmentation, rather they target the smaller yet important and informative foreground points that do not require point-wise prediction, i.e., car, bike, pedestrian, etc. Sustainability demands the safety of pedestrians and cyclists and as few collisions in smart cities as possible to achieve the goal of real-time automation in transpositions. But, the current point-based 3D object detection networks follow sharp down-sampling and task-oriented feature selection like farthest point sampling (FPS) [35,53] or random sampling [63], which sharply reduces the important geometric information which is crucial for global feature learning. Following this finding, we introduce the DFA-SAT framework in this section, as shown in Figures 3 and 4. It includes four primary components: object-based down-sampling (OBDS), semantic and contextual features extraction (SCFE), multi-level feature re-weighting (MLFR), and local and global features aggregation (LGFA). The OBDS module preserves more semantic foreground points based on the basis of spatial information. SEFE learns rich semantic and contextual information with respect to spatial dependencies to refine the local point features information. MLFR decodes all the point features using the channel-wise multi-layered transformer approach to enhance the relationship among local features. It adjusts the weights of these relationships, emphasizing the most significant connections.

3.2. Object-Based Down-Sampling (OBDS)

It was discussed earlier that the object recall rate is inversely proportional to the number of points in the sample encoded features. Random down-sampling techniques sharply reduce foreground point features which results in a significant decrease in recall rate. It is observed that Feat-FPS [53] and D-FPS [35,63] yield good object recall rates at the early encoding stage but they do not preserve enough foreground global features for the final encoding stage. This reduces the precise object detection of the targeted object, especially in the case of distant or comparatively smaller objects like pedestrians and bikes, due to the limited availability of foreground global point features. Therefore, we have incorporated object-oriented down-sampling to preserve maximum foreground features by learning richer semantic information for the local encoding process followed by a further feature learning pipeline. This engineered down-sampling approach enables the learning of semantic features for each local point by the addition of two extra MLP layers to the encoding layers to learn the semantic category of point features. Here, a supervised semantic hot-label for each point is generated from the annotation of the original bounding box by the implementation of vanilla cross-entropy loss.

$$p = -\sum_{m=1}^{M} (y_i log(\hat{y}_i) + (1 - y_i) log(1 - \hat{y}_i)) \quad (1)$$

where M denotes the categories count, y_i is one hot-label, and $-\hat{y}_i$ is predicted logit. We have selected only the top k foreground points as representative features and fed them to the second encoding layer to achieve a comparatively higher recall ratio.

Moreover, this stage filters the points with respect to the central point feature and weights those point features according to their distance from the center of the object using the six spatial attributes of the bounding box as shown in Equation (2).

$$p_i = \sqrt[3]{\frac{min(b^*, f^*)}{max(b^*, f^*)} \times \frac{min(r^*, l^*)}{max(r^*, l^*)} \times \frac{min(d^*, u^*)}{max(d^*, u^*)}} \quad (2)$$

where coordinates values top, down, right, left, front, and back values of the masked point bounding box are represented as u^*, d^*, r^*, l^*, f^*, and b^* of the bounding box, respectively.

Points closer to the central point of instance probably have a higher mask value (*max* = 1) and the points closer to the surface are likely to have the lowest value (*min* = 0). We have implemented a soft-point-mask approach to allocate different weights to point features as per their spatial distance from the central point feature in the training pipeline. The spatial distance of each point is learned and loss is also calculated for this process and given in the coming section in Equation (10). At this level, our explicitly engineered down-sampling approach preserves high-scoring k points for the model training pipeline.

$$L_{sem} = -\sum_{m=1}^{M} (p_i.y_i log(\hat{y}_i) + (1 - y_i) log(1 - \hat{y}_i)) \tag{3}$$

3.3. Proposal Refinement (PR)

The proposal refinement process aims to reduce the deviation between the object proposal and its corresponding ground truth box by encompassing all possible and important object points. Such particular points are simply called RoI. Following the selection of important points from the proposal bounding box, we are going to target the RoI to refine the features learning process. The scaled RoI has a specific length l, width w, and height and radius $r = \alpha\sqrt{\left(\frac{w^m}{2}\right)^2 + \left(\frac{l^m}{2}\right)^2}$, where α is a hyperparameter. Then, we apply our customized sampling approach within the scaled RoI $((\mathcal{N} = \{p_i, \cdots, p_N\}))$ to obtain point features for further processing. In our object-based and class-based down-sampling strategy, we compute the spatial relationship between the center point and each surrounding point with respect to the given k value.

Considering the importance of key-points for geometric information learning [64,65], the down-sampled proposal information is taken for the key-point information learning process. Our key-point extraction strategy calculates the relative coordinate between a point and eight corners of the respective proposal. The relative coordinate calculation is performed as $\Delta p_i^c = p_i - p^c$, where c is the corner of the proposal p with the coordinate range $(1, \cdots, 8)$. Note that the distance information of l^m, h^m, w^m, θ^m is kept in a different dimension, as shown in the encoding module in Figure 4. Here, we have a better proposal representation of the newly generated Δp_i^c relative coordinate. Now, the proposal-oriented point features for each point p_i are given as:

$$f = \mathcal{J}\left(\left[\Delta p_i^m, \Delta p_i^1, \cdots, \Delta p_i^8, \Delta f^r\right]\right) \in \mathbb{R}^D \tag{4}$$

where $\mathcal{J}(\cdot)$ is the linear projection layer that embeds the proposal point features to a high-dimensional feature set.

3.4. Semantic and Contextual Features Extraction (SCFE)

A multi-head self-attention encoding layer, which is followed by a feed-forward network (FFN) with residual structure, receives the proposal guided point features to learn rich semantic and contextual information and spatial dependencies to refine the point features. Except for the position encoding scheme, our proposed self-attention encoding architecture has a similar structure to that of NLP transformer [66]. Let $A = [f_1, \cdots, f_N]^T \in \mathbb{R}^{N \times D}$ be an embedded D-dimensional point feature. We have query Q, key K, and value V an embedding and W_q, W_k, and $W_v \in \mathbb{R}^{N \times N}$ in the form of $Q = W_q A$, $K = W_k A$, and $V = W_v A$. The multi-head self-attention mechanism processes these three embeddings with h-head attentions. For each h-head attention, we have $Q_h, K_h, V_h \in \mathbb{R}^{N \times \acute{D}}$, where $h = 1, \cdots, H$. This multi-head self-attention applies softmax to each attention layer and gives the following output:

$$S^{emb}(A) = \mathcal{W}(\mathcal{F}(\mathcal{W}(S^{att}(Q, K, V)))) \tag{5}$$

where $\mathcal{W}(\cdot)$ and $\mathcal{F}(\cdot)$ represent normalization and ReLU activation, respectively. This multi-level self-attention approach constructs the encoding module of the proposed approach.

3.5. Multi-Level Feature Re-Weighting (MLFR)

We have established point features $\hat{A}$ from local encoding, and engineered the down-sampling strategy, RPN, and the multi-attention network so far. This subsection explains the decoding module of the proposed approach, i.e., how we decode all the point features A using the transformer approach. The standard transformer decoder uses self-attention and an encoder–decoder approach for N queries embedding, but our proposed transformer encoder–decoder approach transforms only one query embedding as N query embedding is computationally resource intensive and has high memory latency and is directly proportional to the number of proposals. The standard transformer approach embeds N objects while our approach needs to refine only one proposal for prediction.

3.5.1. Global Feature Decoding (GFD)

We aim to calculate the decoding weights allocated to each point feature and their sum to generate the final representation, which is equal to the weighted sum of all point features. We decided to apply and analyze the standard transformer decoding strategy and then implement and evaluate an improved decoding scheme to learn highly effective decoding weights and to measure the difference between the standard and improved decoding weight acquisition schemes.

The standard decoder applies a query embedding approach or learnable vector of D-dimensions for the point features aggregation process across the channels of attention heads. The final decoding vector calculates the weights for all the point features received from each attention head as shown in Figure 4. This weight aggregation is calculated as:

$$z_h^{(N)} = \sigma\left(\frac{\hat{q_h}\hat{K_h^T}}{\sqrt{\acute{D}}}\right), h = 1, \cdots, H \tag{6}$$

where $\hat{q_h}$ and $\hat{K_h}$ are the corresponding query embedding and key embedding, respectively, of h attention heads which are computed by encoder output projects. Here, each key embedding or individual point is a global aggregation of each point of the $\hat{q_h}\hat{K_h^T}$ vector, and then each vector is assigned a decoding value by the subsequent $softmax$ function according to the normalized vector probability. However, these decoding weights are directly derived from global aggregations and lack the local information of each h-channel. Along with global aggregation, local channel-wise information plays a vital role in 3D point cloud data to obtain important geometric information about each channel.

3.5.2. Local Feature Decoding (LFD)

We have decided to apply a straightforward approach to compute the decoding weights for each h-channel of $\hat{K_h^T}$ instead of overall aggregation at the final stage of the decoder. This is obtained by generating D decoding vectors to obtain the corresponding decoding values for each. These decoding vector values are unified using a linear projection to form an aggregated decoding vector. The new multi-vector decoding scheme is summarized below:

$$z_h^{(V)} = r.\hat{\sigma}\left(\frac{\hat{K_h^T}}{\sqrt{\acute{D}}}\right), h = 1, \cdots, H \tag{7}$$

where r linear projection generates aggregated D decoding values into a re-weighting scalar along the N-dimension to compute $softmax$ using $\hat{\sigma}$. $\hat{\sigma}$ computes decoding weights of local features associated with each channel and omits the global aggregation for each point. Thus, both decoding approaches have local and global targets that do not encompass the overall feature point domain. Therefore, we have incorporated the combined

approach to collectively target the potential of both local and global features to compute the decoding weights.

3.6. Local and Global Features Aggregation (LGFA)

This strategy maintains the differences in the decoding weights calculation of points for each channel and then combines them with decoding weights for global points using the matrix product of the key embedding and query embedding to learn the spatial information across each *h*-channel. This combined re-weighting scheme creates the following vector decoding weights for all the feature points.

$$z_h^{(lg)} = r.\hat{\sigma}\left(\frac{\hbar(\hat{q_h}\hat{K_h^T}) \circ \hat{K_h^T}}{\sqrt{\acute{D}}}\right), h = 1, \cdots, H \tag{8}$$

where the $\hbar(\cdot)$ operation repeatedly creates $\mathbb{R}^{1\times N} \rightarrow \mathbb{R}^{\acute{D}\times N}$. In this way, we have calculated the global encoding weights and each local *h*-channel's information as compared to individual local and global encoding schemes. As compared to the standard transformer encoding–decoding approach, our global and local aggregated approach increases the minimum computational resources, which is easily adjustable as an accuracy vs. computation trade-off. The final embedding vector as the deciding proposal is given as:

$$y = \left[z_1^{(lg)}.\hat{V}_1, \cdots, z_H^{(lg)}.\hat{V}_H\right] \tag{9}$$

where the $\hat{V}$ value's embedding is the linear projection which is obtained from $\hat{A}$.

3.7. 3D Detection Head

The *D*-dimensional embedded decoding weights as the vector y are fed to the proposal generation head (FFN) for bounding boxes confidence prediction according to their corresponding classes. We have utilized IoU evaluation as the training targets between the 3D proposals and their corresponding 3D boxes. The given IoU between the 3D proposal representation and the corresponding ground truth 3D box. We incorporate scaling, location, and orientation information to encode multi-dimensional encoding representation. We have filtered all the proposals using post-processing 3D-NMS under specific IoU threshold values.

End to End Learning

We have trained our proposed 3D object detection for AVs in an end-to-end fashion. A multi-task loss strategy is employed in the form of summation for overall optimization. The total loss L_{total} comprises a customized down-sampling strategy loss $L_{dsample}$, confidence prediction loss L_{conf}, and box regression loss L_{reg}.

$$L_{total} = L_{dsmaple} + L_{conf} + L_{reg} \tag{10}$$

The binary cross-entropy loss is incorporated to predict the IoU-guided confidence loss. The box generation loss is also calculated under the size, location, angle-res, angle-bin, and corner areas:

$$L_{total} = L_{size} + L_{loc} + L_{ang\text{-}res} + L_{ang\text{-}bin} + L_{corner} \tag{11}$$

4. Experiment and Results

In this segment, we outline our experimental setup and present a sequence of validation test outcomes. We integrate the newly suggested DFA-SAT module into two established 3D object detection frameworks: PointPillars [37] and SECOND [34]. The outcomes generated by different versions of this model are succinctly summarized as follows.

4.1. Experimental Setup

We build our DFA-SAT based on an encoder–decoder architecture to achieve better accuracy without sacrificing efficiency. We incorporate set-abstraction layers for point-wise feature extraction. We target and steadily extract local geometric features by multi-scale grouping. We target the features in a layer-wise manner as we add two MLP layers before the RPN [34] to perform object-based down-samplings at each layer using custom down-sampling instead of standard down-sampling approaches like D-FPS. We refine the region proposal using a transformer-based decoding weights calculation by utilizing combined local and global points along each attention layer. We empirically evaluate DFA-SAT on the well-known publicly available dataset KITTI [55]. We also verify the effectiveness and contribution of each module of DFA-SAT through comprehensive ablation studies.

4.1.1. Dataset

The KITTI benchmark has three levels of difficulty ("hard", "moderate", and "easy") and objects are classified into car, cyclist, and pedestrian. Most of the studies consider moderate-level results as the main indicator. The KITTI 3D dataset contains a total of 15,062 LiDAR samples with 7481 training and 7518 testing LiDAR samples. For experimental training and evaluation, we split the KITTI training dataset into 3713 samples and 3769 samples for training and validation, respectively, following previous work [67]. The easy, moderate, and hard classification sets are part of KITTI and evaluated with different difficulty levels as well as maximum occlusion, minimum height, bounding box (Bbox), and maximum truncation.

The 3D detection results were assessed by measuring 3D and bird's eye view (BEV) average precisions (APs) at a 0.7 intersection over union (IoU) threshold specifically for the car class. The evaluation involved calculating the average precision (AP) for 40 recall positions on both the validation and test sets. For a fair comparison with prior works, a validation AP was also determined using 11 recall positions. Server submissions were evaluated using a training to validation ratio of 9:1, where 6733 samples were randomly selected from the training point cloud, and the remaining 784 samples were used for validation purposes. To compare the results, the PointPillars [37] and SECOND [34] baselines were utilized with a comprehensive network setup. This setup involved employing non-maximum suppression (NMS) with an overlap threshold of 0.7 IoU, applying a range filter of [(0, 70.4), (40, 40), (3, 1)] specifically for cars, and utilizing an anchor filter of [3.9, 1.6, 1.56] for cars. OpenPCDet [68] was employed to implement data augmentation techniques. Overall, these evaluation procedures and comparisons aimed to assess the performance and effectiveness of various 3D detection approaches in accurately detecting and localizing objects, particularly cars, in the given datasets.

4.1.2. Implementation Details

Our custom down-sampling approach provides up-to-the-mark inference and good-quality proposal boundary selection without an aggressive reduction in features. For the KITTI dataset, we set the x-, y-, and z-coordinate ranges as $(0, 71.21)$, $(-40, 40)$, $(-3, 1)$ and the x-axis, y-axis, and z-axis of the voxels are set as 0.05 m, 0.05 m, and 0.1 m. We conduct our experiments using the OpenPCDet [68] toolbox and readers are encouraged to read this reference for more details.

4.1.3. Network Configuration

For the KITTI dataset training, we set a batch size of 24 and used eight V100 GPUs to train the entire DDFA-SAT network. The whole DFA-SAT is trained in an end-to-end fashion from scratch with transformer channels $H = 4$, the ADAM optimizer, learning rate = 0.0001 with cosine annealing strategy, and epochs = 100. We select random proposals to calculate the confidence loss and regression loss using the IoU measurement.

4.2. Main Results

4.2.1. Detection Results

We compare the proposed DFA-SAT with existing studies on the KITTI benchmark dataset. Following [27,29,34,37], the average precision (AP) calculations are performed for the test set and value set with 40 recall positions and 11 recall positions, respectively, to conduct a comparison with previous methods. Tables 1 and 2 illustrate a performance comparison between our method and existing studies. It can be seen that our method achieves good performance in 3D object detection in comparison to other methods also shown in Table 3. This was achieved only by preserving global foreground point features using a customized down-sampling approach before the incorporation of the RPN module. Our DFA-SAT also yields better results for 'cyclist' detection against other point-based methods, as shown in Tables 1 and 2. Our DFA-SAT also shows better efficiency along with competitive object detection performance and it is worth mentioning that the mean AP is improved by 8.03% and 6.86% using the SECOND RPN for BEV and 3D detection, respectively. Also, the mean AP was improved by 5.22% and 6.43% using the PointPillars RPN for BEV and 3D detection, respectively. Our DFA-SAT also achieved the highest AP, of 80.54%. This shows its efficiency by detecting at a speed of 32 FPS on an Intel I9-10900X CPU@3.7 GHz with a single NVIDIA RTX 2080Ti. Our DFA-SAT is based on a custom down-sampling strategy for LiDAR point cloud and transformer-based encoder-decoder architecture, which is able to be trained with multiple classes at the same time instead of separate training for each object type in the training dataset. Tables 1 and 2 and Figure 5 demonstrate the empirical and qualitative results achieved by DFA-SAT in comparison with other methods from various perspectives. It is clear from Tables 1 and 2 and Figure 5 that DFA-SAT demonstrates a good ability to detect smaller as well as far away objects like cyclists and pedestrians. DFA-SAT also yields better detection results in comparison to different down-sampling approaches as shown in Table 4. Except for test split detection results, the KITTI dataset validation set performance comparison is also reported in Table 5. Among point-based detectors, DFA-SAT performs better in the detection of all classes. Our DFA-SAT is efficient in working with larger and smaller objects and decreasing points of faraway objects. DFA-SAT achieves better results at a moderate and easy level for car detection with the LiDAR modality. Many studies share the same RPN, SECOND and PointPillars, as ours; DFA-SAT achieves good results and stands comparable to them in terms of time efficiency for parameter tuning, as shown in Table 6. Figure 5 shows the visualization results with considerably better visualization and refinement. The empirical results and comparison verify the effectiveness of our proposed method in terms of better context information of feature points with respect to each other in an object. In addition to quantitative results, we also provide some visualizations of the 3D object detection results in Figure 5, which show accurate 3D bounding box prediction for objects across the road for both BEV and 3D objects. This is achieved by the proposed method due to its ability to absorb foreground semantic features and sharp geometry feature learning, which are distinctions of our method as compared to [26,27,29,30,34].

4.2.2. DFA-SAT Efficiency

We quantify the memory and computational efficiency of the proposed DFA-SAT. We have made a fair comparison of our proposed method with other existing methods in terms of hardware configuration variations and other parameter variations like speed and memory. We feed a similar quantity (16,384) of input point clouds and OpenPCDet [68] configuration is followed for memory. Our method also shows average results in GPU memory consumption reports, as shown in Table 6.

Table 1. BEV object detection quantitative performance evaluation using AP (%) on the KITTI test set and comparison with different methods.

Method	Category	Mod	Year	mAP	Car			Pedestrian			Cyclist		
					Easy	Mod.	Hard	Easy	Mod.	Hard	Easy	Mod.	Hard
MV3D [33]	Parallel	C+L	2017	78.45	86.62	78.93	69.80						-
Contfuse [47]	Sequential	C+L	2018	85.10	94.07	85.35	75.88						-
F-PointNet [48]	Parallel	C+L	2018	83.54	91.17	84.67	74.77	57.13	49.57	45.48	77.26	61.37	53.78
Avod [49]	Parallel	C+L	2018	85.14	90.99	84.82	79.62						-
PIXOR++ [31]	Multi-View	L	2018	83.68	89.38	83.70	77.97	-	-	-	-	-	-
VoxelNet[30]	Voxel	L	2018	82.0	89.35	79.26	77.39	46.13	40.74	38.11	66.70	54.76	50.55
SECOND [34]	Voxel	L	2018	81.80	88.07	79.37	77.95	55.10	46.27	44.76	73.67	56.04	48.78
PointPillars [37]	Voxel	L	2019	84.76	88.35	86.10	79.83	58.66	50.23	47.19	79.19	62.25	56.00
PSIFT+SENet [28]	Point	C+L	2019	82.99	88.80	83.96	76.21						-
PointRCNN [29]	Point	L	2019	87.41	92.13	87.39	82.72	54.77	46.13	42.84	82.56	67.24	60.28
PI-RCNN [27]	Parallel	C+L	2020	86.08	91.44	85.81	81.00	-	-	-	-	-	-
MVMM [26]	Point–Voxel	C+L	2023	88.78	92.17	88.70	85.47	53.75	46.84	44.87	81.84	70.17	63.84
SECOND+DFA-SAT	Point–Voxel	C+L	2023	89.83	93.55	89.04	83.91	55.14	47.28	45.41	83.68	71.82	64.27
PointPillars+DFA-SAT	Point–Voxel	C+L	2023	89.98	92.67	88.34	83.18	55.42	47.63	44.90	81.79	70.75	63.27

Table 2. Three-dimensional object detection quantitative performance evaluation using AP (%) on the KITTI test set and comparison with different methods.

Method	Category	Mod	Year	mAP	Car			Pedestrian			Cyclist		
					Easy	Mod.	Hard	Easy	Mod.	Hard	Easy	Mod.	Hard
MV3D [33]	Parallel	C+L	2017	64.20	74.97	63.63	54.0	-	-	-	-	-	-
Contfuse [47]	Sequential	C+L	2018	71.38	83.68	68.78	61.67	-	-	-	-	-	-
F-PointNet [48]	Parallel	C+L	2018	70.86	82.19	69.79	60.59	50.53	42.15	38.08	72.27	56.17	49.01
Avod [49]	Parallel	C+L	2018	67.70	76.39	66.47	60.23	36.10	27.86	25.76	57.19	42.08	38.29
Avod-FPN [49]	Parallel	C+L	2018	73.52	83.07	71.76	65.73	50.46	42.27	39.04	63.76	50.55	44.93
PointPillars [37]	Voxel	L	2019	74.11	82.58	74.31	68.99	51.45	41.92	38.89	77.10	58.65	51.92
VoxelNet[30]	Voxel	L	2018	66.77	77.47	65.11	57.73	39.48	33.69	31.5	61.22	48.36	44.37
SECOND [34]	Voxel	L	2018	74.33	83.34	72.55	65.82	48.96	38.78	34.91	71.33	52.08	45.83
PSIFT+SENet [28]	Point	C+L	2019	77.14	85.99	72.72	72.72	-	-	-	-	-	-
PointRCNN [29]	Point	C+L	2019	77.77	86.96	75.64	70.70	47.98	39.37	36.01	74.96	58.82	52.53
PI-RCNN [27]	Parallel	C+L	2020	76.41	84.37	74.82	70.03	-	-	-	-	-	-
HVPR [50]	Point–Voxel	L	2021	79.11	86.38	77.92	73.04	53.47	43.96	40.64			-
TANET [51]	Voxel	L	2020	76.38	84.39	75.94	68.82	53.72	44.34	40.49	75.70	59.44	52.53
MVMM [26]	Point–Voxel	C+L	2023	80.08	87.59	78.87	73.78	47.54	40.49	38.36	77.82	64.81	58.79
SECOND+DFA-SAT	Point–Voxel	C+L	2023	81.19	88.36	79.24	73.97	48.87	41.37	39.12	78.95	65.74	59.22
PointPillars+DFA-SAT	Point–Voxel	C+L	2023	80.54	88.45	79.05	74.16	48.91	41.37	38.43	78.36	66.82	59.09

Table 3. Performance comparison of AP of DFA-SAT with different RPNs under 3D and BEV.

Method	AP^{3D} (%)				AP^{BEV} (%)			
	Mean	Easy	Mod.	Hard	Mean	Easy	Mod.	Hard
SECOND	74.33	83.34	72.55	65.82	81.80	88.07	79.37	77.95
SECOND+DFA-SAT	81.19	88.36	79.24	73.97	89.83	93.55	89.04	83.91
Delta	+6.86	+5.02	+6.69	+8.15	+8.03	+5.48	+9.67	+5.96
PointPillars	74.11	82.58	74.31	68.99	84.76	88.35	86.10	79.83
PointPillars+DFA-SAT	80.54	88.45	79.05	74.16	89.98	92.67	88.34	83.18
Delta	+6.43	+5.84	+4.74	+5.17	+5.22	+4.32	+2.24	+3.35

Table 4. Ablation study of DFA-SAT with different down-sampling approaches. DFPS and Feat-FPS are traditionally used while we have incorporated the object-based down-sampling method and used AP (%) to demonstrate the results.

DFPS	Feat-FPS	Object-Based	Class-Based	Car Mod	Ped. Mod	Cyc. Mod
✓				77.24	42.98	66.72
✓	✓			79.09	44.43	73.24
	✓	✓	✓	79.54	45.96	70.36
✓	✓	✓		79.23	46.33	72.53
✓		✓	✓	80.75	46.98	72.89

Table 5. Three-dimensional object detection quantitative performance evaluation on the KITTI test set using AP (%) and comparison with different methods.

PointPillars	SECOND	2SR	Par (M)	Moderate AP	Car			Pedestrian			Cyclist		
					Easy	Mod.	Hard	Easy	Mod.	Hard	Easy	Mod.	Hard
✓			20	77.48	86.12	78.56	76.06	62.40	40.14	38.05	76.85	63.28	57.34
✓		✓	28	79.21	88.45	78.59	73.13	62.98	40.58	38.24	77.86	64.32	58.17
	✓		20	80.57	87.87	78.54	73.28	63.29	41.05	38.39	77.73	64.38	58.48
	✓	✓	32	81.19	88.36	79.24	73.97	48.87	41.37	39.12	78.95	65.74	59.22

Table 6. Ablation studies for different transformer-based encoding and decoding layers with two different RPNs to show the generalization ability of the proposed DFA-SAT approach.

Method	OBDS	SCFE	MLFR	LGFA	ms/Image	FPS	Param/MB
SECOND	✓		✓		17	63	71.5
	✓				19	58	71.8
	✓		✓	✓	22	49	72.3
	✓	✓	✓	✓	20	49	75.1
PointPillars	✓		✓		39	27	63.6
	✓				52	21	64.5
	✓			✓	57	19	64.8
	✓	✓		✓	61	20	66.2

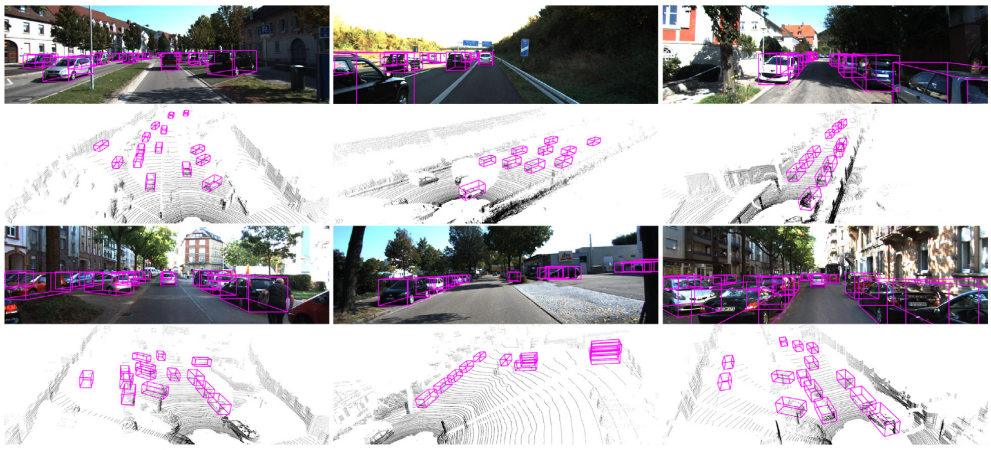

Figure 5. KITTI dataset visualization and prediction results. Sub-figures show predictions for PointPillars and SECOND RPNs at various angles with the DFA-SAT 3D object detection approach.

4.3. Ablation Studies

In the ablation experiments, we have utilized the KITTI dataset for validation of our customized down-sampling and transformer-based encoder–decoder network which is trained with multi-class objects and 40 recall positions using the AP metric. DFA-SAT comprises four modules: object-based down-sampling (OBDS), semantic and contextual

features extraction (SCFE), multi-level feature re-weighting (MLFR), and local and global features aggregation (LGFA). OBDS is one of the major contributions of this study while SCFE and MLFR depend on feature extraction, which is conducted based on engineered down-sampling. Therefore, a series of ablation experiments was conducted by gradually inserting each module to demonstrate and verify the effectiveness of the DFA-SAT modules. We used similar settings and hyperparameters (64 filters, k = 10) for 80 epochs and evaluated the 3D network with the KITTI benchmark validation set. We have applied the 0.7 IoU threshold and 40 recall points on the official KITTI evaluation metric to the car class. Tables 4–8 present the ablation studies of DFA-SAT using different down-sampling approaches, RPN incorporation, semantic and contextual features extraction, multi-level feature re-weighting, and local and global features aggregation with two different RPNs, respectively.

Table 7. Ablation studies for different modules of GFA-SAT architecture using AP (%).

PR	OBDS	SCFE	MLFR	LGFA	Easy	Mod.	Hard
✓		✓			85.42	77.79	72.45
	✓	✓			86.79	78.1	72.81
✓	✓	✓			87.18	78.35	73.13
✓	✓		✓		87.75	79.06	73.65
✓	✓			✓	88.36	79.24	73.97

Table 8. Ablation study of DFA-SAT with various choices of k-nearest neighbors.

Method	k-Points	AP3D (%)			APBEV (%)		
		Easy	Mod.	Hard	Easy	Mod.	Hard
SECOND	9	87.36	79.24	73.97	93.55	89.04	83.91
	16	87.12	78.83	73.15	92.67	88.35	83.02
	32	88.36	79.24	73.97	93.55	89.04	83.91
PointPillars	9	86.47	78.35	72.98	92.46	88.15	82.86
	16	86.23	77.94	72.26	91.78	87.46	82.59
	32	88.45	79.05	74.16	92.67	88.34	83.18

4.3.1. Ablation on OBDS

We verify the effectiveness of our proposed customized down-sampling approach by replacing it with Feat-FPS and D-FPS. Our down-sampling approach gives better results even with a reduced number of point features due to better localization of instance foreground features, as shown in Table 4. This approach achieves good results for even smaller and faraway objects like cyclists, as shown in Tables 1 and 2. This shows that the proposed down-sampling approach is better able to preserve and learn object features to improve detection. Our sampling strategy focuses on the center of the 3D point cloud object and its foreground features and their relationship with the center point feature and does not learn the geometric details of the object with a larger aspect ratio. We have incorporated a customized down-sampling algorithm to explicitly learn relationships among the adjacent points in the k-nearest-neighbors feature domain. It is a distance adjustment approach that uses a dot product to multiply features to suppress the distance to tackle the issue of sparseness which occurs due to aggressive down-sampling.

4.3.2. Ablation on RPN Backbone

We verify the effectiveness of the RPN networks "SECOND [34] and PointPillars [37]" that we used as the backbones of our detection framework. We integrate our encoder–decoder architecture with the existing RPN "SECOND" and "PointPillars" as voxel-based representatives to verify that our DFA-SAT is able to be integrated on top of readily available RPNs for strong proposal refinement. Tables 5 and 6 show the ability of the proposed method to give good refinement results with controlled parameters. To verify the

efficiency and stability of the proposed approach DFA-SAT was added in the 'SECOND' and 'PointPillars' models following its two sparse convolutional layers and the pillar encoding, respectively. All the comparison tests and evaluations were conducted using the KITTI test and validation sets. These experiments proved our claim that our proposed approach is not bound to a specific RPN.

4.3.3. Ablation on SCFE

We validate our strategy of selecting key-points for proposal-to-point embedding. The key-point-subtraction approach is used against existing size-oriented strategies commonly used in proposal-to-point embedding approaches. Our approach to applying key-point subtraction is significantly important for fair results achievements as our results decreased when we replaced it with the existing approach, as shown in Tables 6 and 7.

4.3.4. Ablation on MLFR

The self-attention-based encoding scheme enables the DFA-SAT model to learn more critical features through local feature dependency learning and global features context aggregation. Tables 6 and 7 demonstrate the performance impact of the self-attention encoding approach. We add checkpoints at random epochs and analyze the different attention layers of the trained model by visualization of their attention maps. We observe that the trained model gives more attention to the object car even with sparse points and ignores the background points, as is the objective of this work, as shown in Figure 4. This study reveals the importance of foreground features as well as their prospective results in the trained model. It can be seen from Tables 1–3 that the APs of BEV and 3D for the PointPillars and SECOND (baseline RPNs) were improved when using the DFA-SAT module. The baseline performance was also enhanced by using key-points for geometric information learning using self-attention encoding. From Tables 6 and 7, we inferred that the number of parameters and run-time settings were similar to the baselines. For example, DFA-SAT as a core detection module outperformed the baselines with 57 FPS, with the fastest run-time for PointPillars.

4.3.5. Ablation on LGFA

We have employed three decoding schemes in our trained model and the combined transformer-based local and global features decoding scheme outperformed both individual local and global schemes with micro-margins. The combined features re-weighting approach performs effective weight decoding and integrates both local and global weights, as shown in Tables 6 and 7.

4.3.6. Ablation on the Choice of Different *k* Features in Neighboring

Table 8 illustrates the impact of the hyperparameter *k* in the DFA-SAT, where values of 9, 16, and 32 were chosen for additional validation. Notably, the KITTI dataset yielded the most favorable outcome when *k* was set to 9, which was subsequently employed as the hyperparameter in subsequent experiments. These findings lend further support to the notion that the sparsity of point clouds has a notable influence on the experimental results, as shown in Table 8.

5. Discussion

The incorporation of autonomous vehicles (AVs) into urban settings marks a pivotal development in the evolution of smart cities. At the core of AV technology lies 3D object detection, a fundamental capability enabling AVs to perceive their surroundings in three dimensions. The paper's primary contributions include the proposal of DFA-SAT, a versatile module for 3D object detection in AVs, and its integration into established frameworks like PointPillars and SECOND. DFA-SAT addresses the challenges of weak semantic information in point clouds, particularly for distant and sparse objects. It achieves

this by preserving foreground features, refining semantic and contextual information, and enhancing feature associations.

The significance of DFA-SAT lies in its potential to improve the safety of AVs in smart cities by better detecting pedestrians, cyclists, and other objects by 8.03% and 6.86% using the SECOND RPN for BEV and 3D detection, respectively. The module's minimal impact on the model's parameter (75.1 param/MB) and run-time performance (49 FPS) is crucial for practical applications. The experimental setup is comprehensive, and the authors provide detailed information about the dataset, implementation details, and network configuration. They use the KITTI dataset, a well-established benchmark for 3D object detection, and conduct evaluations on multiple difficulty levels (easy, mod., and hard) and use AP% and mAP% as evaluation metrics. The custom down-sampling approach, encoder–decoder architecture, and transformer-based decoding weight calculations distinguish DFA-SAT. The authors also emphasize the efficiency of their approach, demonstrating its suitability for real-world applications.

The paper presents extensive results comparing DFA-SAT with existing methods. It achieves competitive performance in 3D object detection, particularly for detecting smaller and distant objects like cyclists and pedestrians. The improvement in mean average precision (AP) for both the PointPillars and SECOND frameworks demonstrates the effectiveness of DFA-SAT. It also exhibits efficient performance, achieving high AP while running at 32 FPS. The paper's qualitative results showcase accurate 3D bounding box predictions and refined object detection, emphasizing the importance of semantic and contextual information with meticulous deliberation and strategic implementation. Three-dimensional object detection holds the potential to reshape the functioning of cities, rendering them more habitable, eco-conscious, and responsive to the needs of their inhabitants.

6. Conclusions

The proposed DFA-SAT dynamic feature abstraction with self-attention architecture for 3D object detection in autonomous vehicles has significant implications for smart city applications. By improving the detection performance of LiDAR 3D point-cloud-based object detectors, this research contributes to the advancement of autonomous driving technology, which is a vital component of smart cities. This study presents a dynamic feature abstraction with self-attention (DFA-SAT), encoding decoding architecture for 3D object detection in autonomous vehicles to assist autonomous driving using LiDAR 3D point clouds. It thoroughly examines existing issues with 3D object detectors and proposes a novel methodology called DFA-SAT to extract detailed geometric information among the local semantic features and applies a features re-weighing mechanism. DFA-SAT sets itself apart from existing methods by utilizing a convolutional neural network (CNN) and a self-attention mechanism to learn high-dimensional local features and combine them with low-dimensional global features, leading to significant improvements in detection performance. Experimental evaluations conducted on the KITTI 3D object detection dataset demonstrate the advantages of DFA-SAT, as it achieves noticeable performance enhancements. The research outcomes of DFA-SAT, evaluated on the KITTI 3D object detection dataset, highlight its potential in enhancing autonomous driving and smart city development. Improvements in 3D object detection methods are essential for safer, more efficient, and sustainable urban environments as autonomous vehicles become integrated into smart city infrastructures. The study's insights pave the way for future developments in object detection techniques, driving the progress of autonomous vehicles in urban planning and smart cities. Combining technological advancements with supportive policies and responsible adoption will lead to a more sustainable and environmentally friendly transportation future.

Limitations

The DFA-SAT model demonstrates impressive efficiency in detecting objects within extensive LiDAR point clouds. However, it is not without its limitations. Notably, the

semantic prediction of individual points can be problematic when dealing with imbalanced class distributions. Its accuracy may be hampered in the case of uneven distribution of points for a given semantic context. To address this challenge, in future research, we intend to explore and implement advanced techniques aimed at mitigating the effects of class imbalances. With this, we aim to enhance the model's overall performance and robustness in complex real-world scenarios to provide a more comprehensive understanding of the method's implications for smart city development.

Author Contributions: Conceptualization H.M., X.D. and M.A.; methodology, H.M.; software, H.M. and B.H.; validation, H.M., X.D., M.A. and B.H.; formal analysis, H.M. and H.H.R.S.; investigation, H.M. and H.H.R.S.; resources, H.M. and X.D.; data curation, H.M. and M.A.; writing—original draft preparation, H.M.; writing—review and editing, H.M. and X.D.; visualization, H.M., B.H. and H.H.R.S.; supervision, X.D.; project administration, H.M. and X.D.; funding acquisition, X.D. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The dataset created and examined in the present study can be accessed from the KITTI 3D object detection repository (https://www.cvlibs.net/datasets/kitti/eval_object.php?obj_benchmark=3d, accessed on 14 March 2023).

Acknowledgments: This work was supported by the National Natural Science Foundation of China Project (62172441, 62172449); the Local Science and Technology Developing Foundation Guided by the Central Government of China (Free Exploration project 2021Szvup166); the Opening Project of State Key Laboratory of Nickel and Cobalt Resources Comprehensive Utilization (GZSYS-KY- 2022- 018, GZSYS-KY-2022-024); Key Project of Shenzhen City Special Fund for Fundamental Research(202208183000751); and the National Natural Science Foundation of Hunan Province (2023JJ30696).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Mitieka, D.; Luke, R.; Twinomurinzi, H.; Mageto, J. Smart Mobility in Urban Areas: A Bibliometric Review and Research Agenda. *Sustainability* **2023**, *15*, 6754. [CrossRef]
2. Shi, H.; Hou, D.; Li, X. Center-Aware 3D Object Detection with Attention Mechanism Based on Roadside LiDAR. *Sustainability* **2023**, *15*, 2628. [CrossRef]
3. Lee, H.K. The Relationship between Innovative Technology and Driver's Resistance and Acceptance Intention for Sustainable Use of Automobile Self-Driving System. *Sustainability* **2022**, *14*, 10129. [CrossRef]
4. Zhang, D.; Li, Y.; Li, Y.; Shen, Z. Service Failure Risk Assessment and Service Improvement of Self-Service Electric Vehicle. *Sustainability* **2022**, *14*, 3723. [CrossRef]
5. Xia, T.; Lin, X.; Sun, Y.; Liu, T. An Empirical Study of the Factors Influencing Users' Intention to Use Automotive AR-HUD. *Sustainability* **2023**, *15*, 5028. [CrossRef]
6. Yigitcanlar, T.; Wilson, M.; Kamruzzaman, M. Disruptive Impacts of Automated Driving Systems on the Built Environment and Land Use: An Urban Planner's Perspective. *J. Open Innov. Technol. Mark. Complex.* **2019**, *5*, 24. [CrossRef]
7. Musa, A.A.; Malami, S.I.; Alanazi, F.; Ounaies, W.; Alshammari, M.; Haruna, S.I. Sustainable Traffic Management for Smart Cities Using Internet-of-Things-Oriented Intelligent Transportation Systems (ITS): Challenges and Recommendations. *Sustainability* **2023**, *15*, 9859. [CrossRef]
8. Manfreda, A.; Ljubi, K.; Groznik, A. Autonomous vehicles in the smart city era: An empirical study of adoption factors important for millennials. *Int. J. Inf. Manag.* **2021**, *58*, 102050. [CrossRef]
9. Campisi, T.; Severino, A.; Al-Rashid, M.A.; Pau, G. The Development of the Smart Cities in the Connected and Autonomous Vehicles (CAVs) Era: From Mobility Patterns to Scaling in Cities. *Infrastructures* **2021**, *6*, 100. [CrossRef]
10. Duarte, F.; Ratti, C. The Impact of Autonomous Vehicles on Cities: A Review. *J. Urban Technol.* **2018**, *25*, 3–18. [CrossRef]
11. Heinrichs, D. Autonomous Driving and Urban Land Use. In *Autonomous Driving: Technical, Legal and Social Aspects*; Maurer, M., Gerdes, J.C., Lenz, B., Winner, H., Eds.; Springer: Berlin/Heidelberg, Germany, 2016; pp. 213–231. [CrossRef]
12. Leonard, J.; How, J.; Teller, S.; Berger, M.; Campbell, S.; Fiore, G.; Fletcher, L.; Frazzoli, E.; Huang, A.; Karaman, S.; et al. A Perception-Driven Autonomous Urban Vehicle. In *The DARPA Urban Challenge: Autonomous Vehicles in City Traffic*; Springer: Berlin/Heidelberg, Germany, 2009; pp. 163–230. [CrossRef]
13. Dai, D.; Chen, Z.; Bao, P.; Wang, J. A review of 3d object detection for autonomous driving of electric vehicles. *World Electr. Veh. J.* **2021**, *12*, 139. [CrossRef]

14. Wang, K.; Zhou, T.; Li, X.; Ren, F. Performance and Challenges of 3D Object Detection Methods in Complex Scenes for Autonomous Driving. *IEEE Trans. Intell. Veh.* **2022**, *8*, 1699–1716. [CrossRef]
15. Rosique, F.; Navarro, P.J.; Fernández, C.; Padilla, A. A systematic review of perception system and simulators for autonomous vehicles research. *Sensors* **2019**, *19*, 648. [CrossRef] [PubMed]
16. Rahman, M.M.; Thill, J.C. What Drives People's Willingness to Adopt Autonomous Vehicles? A Review of Internal and External Factors. *Sustainability* **2023**, *15*, 11541. [CrossRef]
17. Yao, L.Y.; Xia, X.F.; Sun, L.S. Transfer Scheme Evaluation Model for a Transportation Hub based on Vectorial Angle Cosine. *Sustainability* **2014**, *6*, 4152–4162. [CrossRef]
18. Stead, D.; Vaddadi, B. Automated vehicles and how they may affect urban form: A review of recent scenario studies. *Cities* **2019**, *92*, 125–133. [CrossRef]
19. Pham Do, M.S.; Kemanji, K.V.; Nguyen, M.D.V.; Vu, T.A.; Meixner, G. The Action Point Angle of Sight: A Traffic Generation Method for Driving Simulation, as a Small Step to Safe, Sustainable and Smart Cities. *Sustainability* **2023**, *15*, 9642. [CrossRef]
20. Feng, D.; Haase-Schütz, C.; Rosenbaum, L.; Hertlein, H.; Gläser, C.; Timm, F.; Wiesbeck, W.; Dietmayer, K. Deep Multi-Modal Object Detection and Semantic Segmentation for Autonomous Driving: Datasets, Methods, and Challenges. *IEEE Trans. Intell. Transp. Syst.* **2021**, *22*, 1341–1360. [CrossRef]
21. Guo, Y.; Wang, H.; Hu, Q.; Liu, H.; Liu, L.; Bennamoun, M. Deep Learning for 3D Point Clouds: A Survey. *IEEE Trans. Pattern Anal. Mach. Intell.* **2021**, *43*, 4338–4364. [CrossRef]
22. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, L.U.; Polosukhin, I. Attention is All you Need. In Proceedings of the Advances in Neural Information Processing Systems, Long Beach, CA, USA, 4–9 December 2017; Guyon, I., Luxburg, U.V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., Garnett, R., Eds.; Curran Associates, Inc.: Red Hook, NY, USA, 2017; Volume 30.
23. Devlin, J.; Chang, M.W.; Lee, K.; Toutanova, K. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), Minneapolis, MN, USA, 2–7 June 2019; pp. 4171–4186. [CrossRef]
24. Yu, H.; Luo, Y.; Shu, M.; Huo, Y.; Yang, Z.; Shi, Y.; Guo, Z.; Li, H.; Hu, X.; Yuan, J.; et al. DAIR-V2X: A Large-Scale Dataset for Vehicle-Infrastructure Cooperative 3D Object Detection. In Proceedings of the 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), New Orleans, LA, USA, 18–24 June 2022; Volume 2022. [CrossRef]
25. Rukhovich, D.; Vorontsova, A.; Konushin, A. ImVoxelNet: Image to Voxels Projection for Monocular and Multi-View General-Purpose 3D Object Detection. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, Waikoloa, HI, USA, 3–8 January 2022. [CrossRef]
26. Li, S.; Geng, K.; Yin, G.; Wang, Z.; Qian, M. MVMM: Multi-View Multi-Modal 3D Object Detection for Autonomous Driving. *IEEE Trans. Ind. Inform.* **2023**, 1–9. [CrossRef]
27. Xie, L.; Xiang, C.; Yu, Z.; Xu, G.; Yang, Z.; Cai, D.; He, X. PI-RCNN: An efficient multi-sensor 3D object detector with point-based attentive cont-conv fusion module. In Proceedings of the AAAI Conference on Artificial Intelligence, New York, NY, USA, 7–12 February 2020; Volume 34, pp. 12460–12467.
28. Zhao, X.; Liu, Z.; Hu, R.; Huang, K. 3D object detection using scale invariant and feature reweighting networks. In Proceedings of the AAAI Conference on Artificial Intelligence, New York, NY, USA, 7–12 February 2019; Volume 33, pp. 9267–9274.
29. Shi, S.; Wang, X.; Li, H. PointRCNN: 3D object proposal generation and detection from point cloud. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 15–20 June 2019; Volume 2019. [CrossRef]
30. Zhou, Y.; Tuzel, O. VoxelNet: End-to-End Learning for Point Cloud Based 3D Object Detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018. [CrossRef]
31. Yang, B.; Luo, W.; Urtasun, R. Pixor: Real-time 3d object detection from point clouds. In Proceedings of the IEEE conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 7652–7660.
32. Xu, W.; Hu, J.; Chen, R.; An, Y.; Xiong, Z.; Liu, H. Keypoint-Aware Single-Stage 3D Object Detector for Autonomous Driving. *Sensors* **2022**, *22*, 1451. [CrossRef]
33. Chen, X.; Ma, H.; Wan, J.; Li, B.; Xia, T. Multi-view 3D object detection network for autonomous driving. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; Volume 2017. [CrossRef]
34. Yan, Y.; Mao, Y.; Li, B. Second: Sparsely embedded convolutional detection. *Sensors* **2018**, *18*, 3337. [CrossRef]
35. Qi, C.R.; Yi, L.; Su, H.; Guibas, L.J. PointNet++: Deep hierarchical feature learning on point sets in a metric space. *Adv. Neural Inf. Process. Syst.* **2017**, *2017*, 30.
36. Qi, C.R.; Su, H.; Mo, K.; Guibas, L.J. PointNet: Deep learning on point sets for 3D classification and segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; Volume 2017. [CrossRef]
37. Lang, A.H.; Vora, S.; Caesar, H.; Zhou, L.; Yang, J.; Beijbom, O. Pointpillars: Fast encoders for object detection from point clouds. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; Volume 2019. [CrossRef]

38. Wang, Y.; Solomon, J.M. Deep Closest Point: Learning Representations for Point Cloud Registration. In Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), Seoul, Republic of Korea, 27 October–2 November 2019.
39. Wang, Y.; Sun, Y.; Liu, Z.; Sarma, S.E.; Bronstein, M.M.; Solomon, J.M. Dynamic graph Cnn for learning on point clouds. *ACM Trans. Graph.* **2019**, *38*, 1–12. [CrossRef]
40. Li, G.; Muller, M.; Thabet, A.; Ghanem, B. DeepGCNs: Can GCNs go as deep as CNNs? In Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), Seoul, Republic of Korea, 27 October–2 November 2019; Volume 2019. [CrossRef]
41. Andresini, G.; Appice, A.; Malerba, D. Nearest cluster-based intrusion detection through convolutional neural networks. *Knowl.-Based Syst.* **2021**, *216*, 106798. [CrossRef]
42. Engel, N.; Belagiannis, V.; Dietmayer, K. Point transformer. *IEEE Access* **2021**, *9*, 16259–16268. [CrossRef]
43. Geiger, A.; Lenz, P.; Urtasun, R. Are we ready for autonomous driving? the KITTI vision benchmark suite. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Providence, RI, USA, 16–21 June 2012. [CrossRef]
44. Murayama, K.; Kanai, K.; Takeuchi, M.; Sun, H.; Katto, J. Deep Pedestrian Density Estimation For Smart City Monitoring. In Proceedings of the 2021 IEEE International Conference on Image Processing (ICIP), Anchorage, AL, USA, 19–22 September 2021; pp. 230–234. [CrossRef]
45. Seuwou, P.; Banissi, E.; Ubakanma, G. The Future of Mobility with Connected and Autonomous Vehicles in Smart Cities. In *Digital Twin Technologies and Smart Cities*; Farsi, M., Daneshkhah, A., Hosseinian-Far, A., Jahankhani, H., Eds.; Springer International Publishing: Cham, Switzerland, 2020; pp. 37–52. [CrossRef]
46. Xu, X.; Dong, S.; Xu, T.; Ding, L.; Wang, J.; Jiang, P.; Song, L.; Li, J. FusionRCNN: LiDAR-Camera Fusion for Two-Stage 3D Object Detection. *Remote Sens.* **2023**, *15*, 1839. [CrossRef]
47. Liang, M.; Yang, B.; Wang, S.; Urtasun, R. Deep Continuous Fusion for Multi-sensor 3D Object Detection. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; Volume 11220, LNCS. [CrossRef]
48. Qi, C.R.; Liu, W.; Wu, C.; Su, H.; Guibas, L.J. Frustum PointNets for 3D Object Detection from RGB-D Data. In Proceedings of the IEEE conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018. [CrossRef]
49. Ku, J.; Mozifian, M.; Lee, J.; Harakeh, A.; Waslander, S.L. Joint 3D Proposal Generation and Object Detection from View Aggregation. In Proceedings of the 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS 2018, Madrid, Spain, 1–5 October 2018. [CrossRef]
50. Noh, J.; Lee, S.; Ham, B. HVPR: Hybrid Voxel-Point Representation for Single-stage 3D Object Detection. In Proceedings of the 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Nashville, TN, USA, 20–25 June 2021. [CrossRef]
51. Liu, Z.; Zhao, X.; Huang, T.; Hu, R.; Zhou, Y.; Bai, X. TANet: Robust 3D object detection from point clouds with triple attention. In Proceedings of the AAAI Conference on Artificial Intelligence, New York, NY, USA, 7–12 February 2020. [CrossRef]
52. Qi, C.R.; Litany, O.; He, K.; Guibas, L. Deep hough voting for 3D object detection in point clouds. In Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), Seoul, Republic of Korea, 27 October–2 November 2019; Volume 2019. [CrossRef]
53. Yang, Z.; Sun, Y.; Liu, S.; Jia, J. 3DSSD: Point-based 3d single stage object detector. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020. [CrossRef]
54. Shi, S.; Guo, C.; Jiang, L.; Wang, Z.; Shi, J.; Wang, X.; Li, H. PV-RCNN: Point-voxel feature set abstraction for 3D object detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020. [CrossRef]
55. Geiger, A.; Lenz, P.; Stiller, C.; Urtasun, R. Vision meets robotics: The KITTI dataset. *Int. J. Robot. Res.* **2013**, *32*, 1231–1237. [CrossRef]
56. Chen, Y.; Li, Y.; Zhang, X.; Sun, J.; Jia, J. Focal Sparse Convolutional Networks for 3D Object Detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 18–24 June 2022; pp. 5418–5427. [CrossRef]
57. Chen, Q.; Li, P.; Xu, M.; Qi, X. Sparse Activation Maps for Interpreting 3D Object Detection. In Proceedings of the 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), Nashville, TN, USA, 20–25 June 2021; pp. 76–84. [CrossRef]
58. Sun, P.; Wang, W.; Chai, Y.; Elsayed, G.; Bewley, A.; Zhang, X.; Sminchisescu, C.; Anguelov, D. RSN: Range Sparse Net for Efficient, Accurate LiDAR 3D Object Detection. In Proceedings of the 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), Nashville, TN, USA, 20–25 June 2021.
59. Ren, M.; Pokrovsky, A.; Yang, B.; Urtasun, R. SBNet: Sparse Blocks Network for Fast Inference. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018.
60. Zhu, X.; Su, W.; Lu, L.; Li, B.; Wang, X.; Dai, J. Deformable DETR: Deformable Transformers for End-to-End Object Detection. *arXiv* **2020**, arXiv:2010.04159.
61. Bhattacharyya, P.; Huang, C.; Czarnecki, K. SA-Det3D: Self-Attention Based Context-Aware 3D Object Detection. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, BC, Canada, 11–17 October 2021; pp. 3022–3031. [CrossRef]

62. Mao, J.; Niu, M.; Bai, H.; Liang, X.; Xu, H.; Xu, C. Pyramid R-CNN: Towards Better Performance and Adaptability for 3D Object Detection. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, BC, Canada, 11–17 October 2021. [CrossRef]
63. Hu, Q.; Yang, B.; Xie, L.; Rosa, S.; Guo, Y.; Wang, Z.; Trigoni, N.; Markham, A. Randla-Net: Efficient semantic segmentation of large-scale point clouds. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020. [CrossRef]
64. Zhou, D.; Fang, J.; Song, X.; Guan, C.; Yin, J.; Dai, Y.; Yang, R. IoU Loss for 2D/3D Object Detection. In Proceedings of the 2019 International Conference on 3D Vision (3DV), Quebec City, QC, Canada, 6–19 September 2019. [CrossRef]
65. Law, H.; Deng, J. CornerNet: Detecting Objects as Paired Keypoints. *Int. J. Comput. Vis.* **2020**, *128*, 734–750. [CrossRef]
66. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, Ł.; Polosukhin, I. Attention is all you need. *Adv. Neural Inf. Process. Syst.* **2017**, *2017*, 30.
67. Chen, X.; Kundu, K.; Zhu, Y.; Berneshawi, A.G.; Ma, H.; Fidler, S.; Urtasun, R. 3D Object Proposals for Accurate Object Class Detection. In Proceedings of the Advances in Neural Information Processing Systems, Lake Tahoe, NV, USA, 6–14 December 2015; Cortes, C., Lawrence, N., Lee, D., Sugiyama, M., Garnett, R., Eds.; Curran Associates, Inc.: Red Hook, NY, USA, 2015; Volume 28.
68. OpenPCDet Development Team. Openpcdet: An Opensource Toolbox for 3D Object Detection from Point Clouds. 2020. Available online: https://github.com/open-mmlab/OpenPCDet (accessed on 16 March 2023).