3D-DFM: Anchor-Free Multimodal 3-D Object Detection With Dynamic Fusion Module for Autonomous Driving

Chunmian Lin, Daxin Tian, *Senior Member, IEEE*, Xuting Duan, *Member, IEEE*, Jianshan Zhou, Dezong Zhao, *Senior Member, IEEE*, and Dongpu Cao

***Abstract*— Recent advances in cross-modal 3D object detection rely heavily on anchor-based methods, and however, intractable anchor parameter tuning and computationally expensive post-processing severely impede an embedded system application, such as autonomous driving. In this work, we develop an anchor-free architecture for efficient camera–light detection and ranging (LiDAR) 3D object detection. To highlight the effect of foreground information from different modalities, we propose a dynamic fusion module (DFM) to adaptively interact images with point features via learnable filters. In addition, the 3D distance intersection-over-union (3D-DIoU) loss is explicitly formulated as a supervision signal for 3D-oriented box regression and optimization. We integrate these components into an end-to-end multimodal 3D detector termed 3D-DFM. Comprehensive experimental results on the widely used KITTI dataset demonstrate the superiority and universality of 3D-DFM architecture, with competitive detection accuracy and real-time inference speed. To the best of our knowledge, this is the first work that incorporates an anchor-free pipeline with multimodal 3D object detection.**

***Index Terms*— 3D object detection, autonomous driving, deep learning, intelligent transportation systems, multimodal fusion.**

I. Introduction

OBJECT detection is a fundamental and essential task for a wide range of real-world applications, including autonomous driving. In general, 2-D object detection encodes object location as the coordinates of the box on the image plane, but without truly spatial information, depicting the physical location of the object of interest is insufficient. To this end, 3D object detection recently emerged as a viable option, due to the widespread availability of various sensors and advanced data-processing techniques. It detects a 3D-oriented bounding box around the target and provides an accurate perception result to guide path planning in such autonomous-driving system.

Existing 3D object detection works have shown promising results, and they can be divided into single-modal and multimodal methods. A single-modal detector typically uses a single camera or light detection and ranging (LiDAR) sensor to understand the 3D properties of objects, i.e., size and orientation, suffering from several limitations. Due to a lack of depth information, a camera-based 3D detector fails to accurately localize spatial objects, whereas the LiDAR-based method cannot distinguish semantic categories of similar structures, particularly in such crowded or distant scenes. Consequently, several studies focused on cross-modal 3D object detection using both camera images and LiDAR points, and they investigated a variety of schemes aggregating modality features at different stages, as shown in Fig. 1.

Early-level fusion occurs during data input, and pixel segmentation labels [1], [4], pseudo-LiDAR signals [5], [6], and spatial transformation operations [7] are used to supplement the point representation. They must, however, ensure data alignment from various sensors through some complex operations. Late fusion is much easier to build and deploy because it incorporates pretrained 2-D and 3D detectors directly and fuses respective detection results by adding or concatenating association [3]. Though it avoids several time-consuming operations at the data level, false candidates from the single-modal method may be mistakenly regarded as valuable cues during result fusion, thereby reducing detection performance. Middle-level fusion generally performs image-point complementarities on the intermediate feature map, via multiview combination [8], region-proposal aggregation [9], bird's-eye-view (BEV) feature fusion [2], [10]–[12], and so on. However, these fusion modules produce coarse or imperfect correspondence at a fixed location, and background noise also produces a large number of false positive results. Furthermore, existing cross-modal fusion methods follow an anchor-based pipeline with many prior boxes and postprocessing, resulting in high architecture complexity and a large computation budget.

Manuscript received September 12, 2021; revised February 17, 2022; accepted April 27, 2022. This work was supported in part by the National Natural Science Foundation of China under Grant U20A20155, Grant 62061130221, and Grant 62173012; in part by the Beijing Municipal Natural Science Foundation under Grant L191001; in part by the Zhuoyue Program of Beihang University (Postdoctoral Fellowship); in part by the China Postdoctoral Science Foundation under Grant 2020M680299; and in part by the Beijing Municipal Science and Technology Commission under Grant Z211100001921004. *(Corresponding author: Daxin Tian.)*

Chunmian Lin, Daxin Tian, Xuting Duan, and Jianshan Zhou are with the Beijing Key Laboratory for Cooperative Vehicle Infrastructure Systems and Safety Control, Beijing Advanced Innovation Center for Big Data and Brain Computing, School of Transportation Science and Engineering, Beihang University, Beijing 100191, China (e-mail: dtian@buaa.edu.cn).

Dezong Zhao is with the James Watt School of Engineering, University of Glasgow, Glasgow G12 8QQ, U.K.

Dongpu Cao is with the Department of Mechanical and Mechatronics Engineering, University of Waterloo, Waterloo, ON N2L 3G1, Canada.

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TNNLS.2022.3171553.

Digital Object Identifier 10.1109/TNNLS.2022.3171553

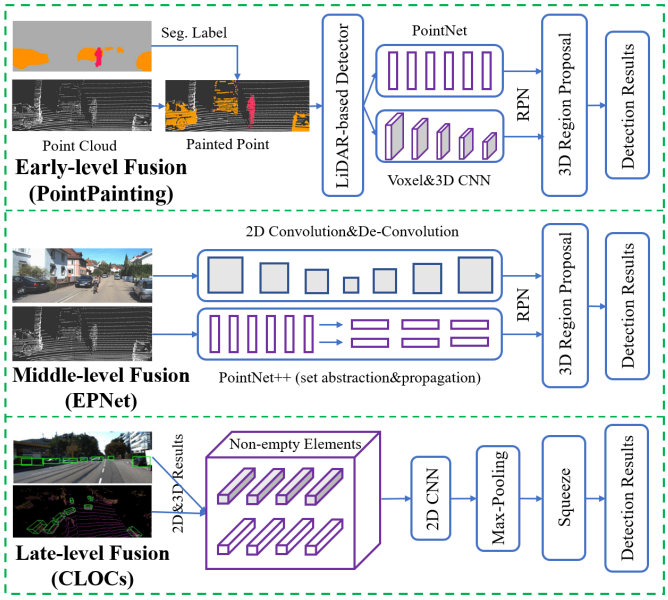

Fig. 1. Schematic of different fusion methods including early-level (PointPainting [1]), middle-level (EPNet [2]), and late-level (CLOCs [3]) fusion. Commonly, these approaches require to define prior box parameters to generate region proposals for 3D box prediction via RPN architecture, and postprocessing NMS is also indispensable to remove redundant candidates, significantly hindering the inference efficiency.

Box regression in 3D object detection is a more difficult and complex problem because it involves more spatial and size parameters. $L1$-norm distance is commonly used in the 2-D object detection to evaluate the similarity of ground truth (GT) and predicted box. However, this type of loss ignores the geometric property of the bounding box and generates different candidate boxes for the same GT, leading to the suboptimal box regression result [14]. Because of the increased degree of freedom in 3D object detection, this phenomenon would be amplificated. Recently, the intersection of union (IoU) [15] and its variant generalized-IoU (GIoU) [16] have been proposed as another evaluation metric of the distance between two boxes and have been successfully extended from 2-D to 3D object detection community, e.g., 3D-IoU [17] and 3D-GIoU [18]. Unfortunately, the IoU metric cannot evaluate two nonoverlap boxes, and the divergence of GIoU during model training remains unresolved. These problems motivate us to look into an alternative to guide 3D box regression and optimization.

In this article, we present an anchor-free pipeline for multimodal 3D object detection, which waives complicated prior box and time-consuming NMS postprocessing. This is the first time, to the best of our knowledge, that anchor-free architecture has been combined with cross-modal 3D object detection. Using learnable filters, a dynamic fusion module (DFM) is proposed to aggregate image with point representation. It generates kernel parameters from the image feature map and then convolves point features with these generated filters to achieve adaptive image-point feature fusion and interaction. Background information is filtered out in this manner, and foreground features are preserved to contribute to object localization and detection. Furthermore, we develop a 3D distance intersection-over-union (3D-DIoU) metric that considers the attributes of center, overlap, and scale for two boxes, and formulate the 3D-DIoU loss for more accurate and consistent box regression. We incorporate these components into an end-to-end network termed 3D-DFM that is a flexible and general architecture that can utilize the existing voxel-based method to build a powerful cross-modal 3D detector. Extensive experiments are performed on publicly available KITTI benchmark [19], and the results demonstrate the superiority and universality of 3D-DFM. In particular, based on one-stage VoxelNet [20] and two-stage Part-A^2 [21], our proposed method reports real-time inference speed and state-of-the-art detection accuracy, which outperforms both single-modal and cross-modal 3D detectors by a remarkable margin.

The main contributions in this article can be summarized as follows.

1) First, we incorporate an anchor-free pipeline with camera–LiDAR 3D object detection, which simplifies postprocessing operations with much less engineering effort and accelerates model inference speed.
2) DFM is designed to combine the image with point features dynamically. It generates kernel parameters from the semantic map and interacts with point feature via these filters adaptively.
3) We investigate the effect of center point, overlapping area, and scale between two boxes and propose the 3D-DIoU loss for better 3D box optimization.
4) Empirical studies and ablation analysis are conducted on the KITTI dataset, and the results present the effectiveness and generalization of our proposed method.

II. Related Works

This section would review recent works on 3D object detection, anchor-free object detection, and box optimization in object detection.

A. 3D Object Detection

As previously stated, 3D object detection can be divided into three categories: camera-based, LiDAR-based, and cross-modal methods.

1) Camera-Based Method or Monocular 3D Detection: It usually builds on the design of 2-D object detector and uses geometric constraints to normalize bounding coordinates in the 3D space [22]. Recently, several image-based 3D detection works have been interested in how to recover depth information for assisting 3D object localization accuracy. A series of pseudo-LiDAR works [5], [23], [24] generates pseudo signal from depth image and performs 3D object detection on the pseudo point representation. Furthermore, DL4CN [25] proposes a dynamic depthwise dilated local convolutional network that learns depth maps automatically from images with different filters. CaDNN [26] is a differentiable end-to-end approach for monocular 3D detection that predicts categorical depth distribution for each pixel. Nevertheless, an image-based 3D detector achieves poor accuracy and only captures coarse object 3D boxes.

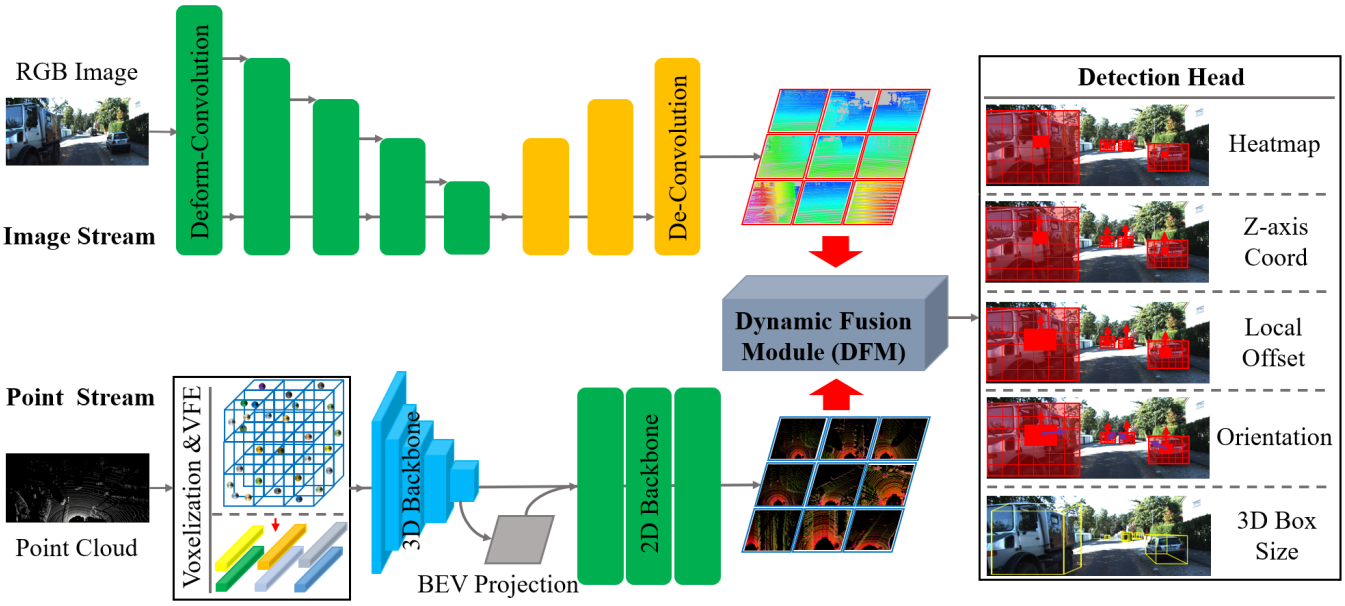

Fig. 2. Overview of 3D-DFM. Image stream: arbitrary 2-D CNN can be adopted for image feature learning. We design a simple architecture with five deformable convolutional layers with stride 2, followed by three deconvolutions to recover the resolution of semantic map. Point stream: for efficiency, any voxel-based detector can be utilized as a point cloud encoder that contains voxelization, VFE, 3D backbone, BEV projection, and 2-D backbone. DFM: it dynamically generates kernel parameters from the image map and convolves point features with these filters to aggregate multimodal information adaptively. Detection head: it directly predicts the center heatmap, Z-axis coordinate, local offset, orientation, and 3D box size from the joint feature map. Also, the whole architecture is supervised by the focal loss [13] and proposed 3D-DIoU loss for classification and box regression, respectively.

2) LiDAR-Based 3D Detector: It regresses the 3D box directly from the point cloud and reports the state-of-the-art detection performance in several public benchmarks. To handle the sparisity and irregularity of the point cloud, existing approaches either process voxel features via an efficient convolutional neural network (CNN) [20], [21], [27] or consume raw points by PointNet architectures [28]–[30]. It is noted that utilizing both data representations can result in improved detection performance. SA-SSD [31] builds a convolutional network to explicitly leverage the structured information of 3D point cloud while also guiding the backbone to be aware of object structure. Similarly, PV-RCNN [32] designs voxel set abstraction and keypoint set abstraction modules; it first summarizes voxel representation into a small set of keypoints and pools keypoints to representative region-of-interest (RoI)-grid points. However, without the assistance of semantics, LiDAR-based algorithms are prone to feature ambiguity of similar objects.

3) Cross-Modal Approach: It has received increasing attention due to the advantages of image-point complementarity. Early fusion typically introduces additional labels or features into a point cloud, such as PointPainting [1], PointAugment [4], and MVX-Net [11]. Many studies concentrate on feature combinations at the intermediate level. MV3D [8] and AVOD [9] fuse multiview proposal or RoI region using a pooling or crop module. 3D-CVF [12] performs autocalibrated projection to transform the image into a smoother BEV feature map and then applies simple concatenation on both points. EPNet [2] designs a Li-Fusion module inspired by the novel continuous convolution [10] and establishes pointwise correspondences for finer multimodal feature aggregation. Later fusion directly combines the 2-D and 3D detection results via a specific association or operation. CLOCs [3] adopts geometric and semantic consistency to convert 2-D and 3D candidates into a set of joint detection candidates, whereas these schemes frequently encounter issues with imperfect correspondence or data misalignment. In this work, we would develop a more accurate and adaptive approach to support multimodal feature fusion.

B. Anchor-Free Object Detection

Anchor-based detectors, in the context of 2-D object detection, depend on the design of prior boxes or predefined parameters to guide the bounding-box regression, which results in heavy computational cost and model complexity. Conversely, anchor-free architecture avoids the complicated engineering effort and instead predicts box size and confidence score from feature maps, such as DenseBox [33] and YOLOv1 [34]. Recently, keypoint estimation has been used for object detection. CornerNet [35] estimates a pair of bounding-box corners and designs corner pooling for better object localization. CenterNet [36] is a simpler and faster approach for locating the object's center point and predicting all other object properties, such as size, location, and orientation. The anchor-free paradigm has been widely studied and popularized in the 2-D object detection domain due to its efficient and accurate detection performance.

Several works introduce an anchor-free pipeline to 3D object detection. PointRCNN [30] creates the 3D proposal generation subnetwork for box refinement and confidence prediction based on point cloud segmentation. AFDet [37] simplifies postprocessing by removing anchor and NMS designs.

CenterPoint [38] and CenterNet3D [39] develop center-based 3D object detection architectures: the former estimates the center point for simultaneous detection and tracking, while the latter proposes an auxiliary corner attention module to pay more attention to object boundaries. In this article, we would investigate the potential of combining anchor-free architecture with multimodal 3D object detection.

C. Box Optimization in Object Detection

For 2-D object detection, $L1$ loss and its derivate are commonly used for box optimization, which measures the Euclidean distance between the GT and predicted boxes. Nonetheless, the independent assumption of four corner points may result in poor localization results. Alternatively, the IoU loss is introduced for bounding-box prediction [15], and the GIoU [16] is explored to tackle with two nonoverlapping boxes optimization.

It is intractable for a 3D-oriented bounding box with more coordinate, size, and pose parameters. More importantly, geometric correlation is also of vital importance for corner point prediction, which implies that the $L1$ distance loss is unsuitable for 3D object detection. Consequently, the 3D-IoU loss [17] and the 3D-GIoU loss [18] are proposed to evaluate the similarity of two rotated boxes in the 3D space. In general, box optimization in 3D space remains an open problem, and in this article, we will investigate a novel loss function to facilitate the 3D box regression.

III. Methodology

This section would introduce our proposed 3D-DFM architecture, as shown in Fig. 2, that primarily consists of image stream, point stream, DFM, and detection head.

A. Image Stream

A CNN is used to learn dense semantic features from RGB images in an image stream. We introduce deformable convolution [40] to capture local offset for spatial location in order to improve geometric transformation modeling. Furthermore, residual architecture [41] is required for stable model training and convergence. Therefore, we design a simple image stream composed of five stacked convolutional blocks with shortcut connections, each of which contains a 3×3 deformable filter with stride 2, followed by batch normalization [42] and ReLU activation function. The number of channels is 32, 64, 128, 256, and 512. To recover the size of feature map, we append three deconvolutional layers with 256, 128, and 64 channels. Finally, the image stream produces a high-resolution semantic map with stride 4 of input. It should be noted that any convolutional backbone can be adopted as an image stream, and we simply present a lightweight architecture for semantic feature extraction.

B. Point Stream

To efficiently encode point cloud features, the point stream architecture follows the previous voxel-based works that include voxelization, voxel feature encoding (VFE), 3D backbone, BEV projection, and 2-D backbone. To be specific, the raw point cloud is grouped and quantized into regular voxels with a fixed length, width, and height dimensions. Subsequently, a point-based architecture (i.e., PointNet) encodes the 3D information within each voxel, and the pooled feature is fed into 3D backbone (e.g., 3D sparse CNN [43]) for voxel abstraction. Finally, we compress the voxel feature into a BEV representation along the height dimension and adopt 2-D dense backbone to produce the point feature map. More importantly, most off-the-shelf voxel-based approaches can be served as a point feature encoder, and we would introduce one- and two-stage 3D detectors into our proposed architecture.

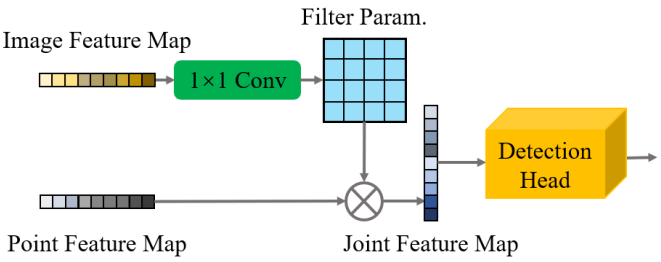

Fig. 3. Architecture of DFM. It generates kernel parameters from the image map via 1×1 convolution and dynamically interacts point features with these filters to produce the joint feature map. In this way, optimal pixel-point correspondences would be obtained to aggregate the multimodal information adaptively.

C. Dynamic Fusion Module

Much prior research aggregates image and point feature maps using a mathematical operation (e.g., summation or concatenation), and however, these approaches simply perform coarse correspondences at the targeted location without information interaction in local and global regions. The dynamic filter network [44] proposes a new convolutional framework in which kernel parameters are generated dynamically based on the input rather than fixed filters as in the standard convolutional layer. It contains a filter-generating module and a dynamic filtering layer. The former is a learnable architecture that provides custom parameters for different input samples, and the latter further applies these kernels to the input. A dynamic filter network has been applied in many complex tasks and showed promising performance [45], [46] due to its unique architecture and dynamic mechanism.

As shown in Fig. 3, we investigate the dynamic filtering mechanism on cross-modal feature combination and design DFM conditioned on image feature $f_I \in R^{H_I \times W_I \times C_I}$ and point feature $f_P \in R^{H_P \times W_P \times C_P}$, where H, W, and C are height, width, and the number of channels of feature map, respectively. Specifically, DFM adopts 1×1 convolution or linear unit as filter-generating network at first and dynamically generates filter parameters $F_\theta \in R^{N \times K \times K \times C_P}$ from the image map, where N is the number of filters, K is the size of the filter, and C_P is the number of channels. Also, in the dynamic filtering layer, these kernels are adaptively convolved with the point feature to obtain one-to-one correspondence results.

As a consequence, the joint feature map $f_D \in R^{H\times W\times N}$ is fed into an anchor-free detection head for box prediction in the next stage. The fused feature at (i, j) position can be mathematically described as follows:

$$f_D(i, j) = F_\theta \bigotimes (f_P(i, j)) \tag{1}$$

where $\bigotimes$ denotes the convolved multiplication.

There are several advantages of DFM design. On the one hand, DFM can be regarded as an attention mechanism for attending to the RoI feature of the size of $H \times W$. The point feature is dynamically reweighted by the customized kernel parameters generated from the image map, to obtain the significant fused representations. In this way, ineffective bins with background information are filtered out, and the model would pay more attention on the majority of foreground features during object localization and classification. On the other hand, DFM can adaptively provide better pixel-point correspondences via learnable filters. These sample-specific parameters are interacted with point feature to produce a more discriminative joint feature. Furthermore, DFM is also a light-weight architecture that only involves simple convolutional or linear operations, resulting in a marginal computation overhead for the network.

D. Detection Head

Our anchor-free detection head takes the joint feature map $f_D \in R^{H\times W\times N}$ as input and predicts center-point heatmap, z-axis coordinate, local offset, orientation, and 3D box size of objects, which is similar to the previous center-based works in [41] and [45].

The heatmap branch is responsible for the center-point estimation that finds where the object center $\tilde{p}(x, y)$ is in BEV. It learns to predict an $H \times W$ heatmap $\tilde{M}_{xyc} \in [0, 1]^{H\times W\times C}$ for C categories, with the rendered Gaussian kernels $G = \exp(-((x - \tilde{p}_x)^2 + (y - \tilde{p}_y)^2/2\sigma_p^2))$ at each GT object center $p(x, y)$, where σ_p denotes the object size adaptive standard deviation. Then, a z-axis coordinate branch is used to determine the z-axis location $\tilde{Z} \in R^{H\times W\times 1}$ for each center point. The local offset branch aims to recover discretization errors caused by the voxelization process and downsample stride and find the more accurate object center. It directly predicts an offset map $\tilde{O} \in R^{H\times W\times 3}$ for each center point. Considering the assumption that objects are shared with the same ground plane in most autonomous-driving scenes, we simply estimate the orientation map $\tilde{R} \in R^{H\times W\times 2}$ around the z-axis and encode the rotation as $(\sin(\theta), \cos(\theta))$. Finally, the 3D box size feature map $\tilde{S} \in R^{H\times W\times 3}$ is regressed to obtain the width, length, and height for each object. During model training, we adopt logistic regression with focal loss [13] for classification and the newly developed loss function to supervise 3D box regression, as described in Section III-E.

E. Loss Function

Overall, the loss function L for classification and box regression in the proposed method can be expressed as follows:

$$L = L_{\text{cls}} + L_{\text{reg}}. \tag{2}$$

Algorithm 1 3D-DIoU

Input: Ground-truth box: $\boldsymbol{B_g}(x_g, y_g, z_g, w_g, l_g, h_g, \theta_g)$ and the predicted box: $\boldsymbol{B_p}(x_p, y_p, z_p, w_p, l_p, h_p, \theta_p)$

2D-IoU of Boxes

1. Boxes in BEV Projection

$\boldsymbol{B_g^{bev}}(x_g^{bev}, y_g^{bev}, w_g^{bev}, l_g^{bev}, \theta_g^{bev})$

$\boldsymbol{B_p^{bev}}(x_p^{bev}, y_p^{bev}, w_p^{bev}, l_p^{bev}, \theta_p^{bev})$

2. 2D IoU Calculation

Area B_g^{bev} : $\boldsymbol{A_g} = w_g^{bev} \times h_g^{bev} \times sin(\theta_g^{bev})$

Area B_p^{bev} : $\boldsymbol{A_p} = w_p^{bev} \times h_p^{bev} \times sin(\theta_p^{bev})$

$\boldsymbol{I_{gp}}$: calculates the intersection area by sorting vertexes of two overlapping boxes in anticlockwise order, else $\boldsymbol{I_{gp}} = 0$

$\boldsymbol{IoU_{2D} = I_{gp}/(A_g + A_p - I_{gp})}$

3D-DIoU of Boxes

1. 3D-IoU Calculation

Volume B_g: $\boldsymbol{V_g} = w_g \times l_g \times h_g$

Volume B_p: $\boldsymbol{V_p} = w_p \times l_p \times h_p$

Intersection of Height: $\boldsymbol{I_h} = \boldsymbol{min}(z_g + h_g/2, z_p + h_p/2) - \boldsymbol{max}(z_g - h_g/2, z_p - h_p/2)$

Intersection of Volume: $\boldsymbol{I_v = I_{gp} \times I_h}$

$\boldsymbol{IoU_{3D} = I_v/(V_g + V_p - I_v)}$

2. The Distance of Center Points

$\boldsymbol{D} = \sqrt{(x_g - x_p)^2 + (y_g - y_p)^2 + (z_g - z_p)^2}$

3. The Diagonal of Smallest Closing Volume

$\boldsymbol{C} = \sqrt{\boldsymbol{max}(w_g, w_p)^2 + \boldsymbol{max}(l_g, l_p)^2 + \boldsymbol{max}(h_g, h_p)^2}$

4. 3D-DIoU Calculation

$\boldsymbol{DIoU_{3D} = IoU_{3D} - D^2/C^2}$

Output: $\boldsymbol{DIoU_{3D}}$

The classification loss adopts a focal loss to handle the class imbalance in heatmap prediction, as illustrated as follows:

$$L_{\text{cls}} = \frac{-1}{N} \sum_{xyc} \begin{cases} (1 - \tilde{M}_{xyc})^\alpha log(\tilde{M}_{xyc}), & M_{xyc} = 1 \\ (1 - M_{xyc})^\beta (\tilde{M}_{xyc})^\alpha & \\ \quad \times \log(1 - \tilde{M}_{xyc}), & \text{otherwise} \end{cases} \tag{3}$$

where N is the number of the center point and M_{xyc} denotes the center heatmap generated from Gaussian kernel. In all experiments, the hyperparameters α and β are set to 2 and 4, respectively.

Inspired by 2-D distance IoU [49], we investigate the effect of center distance, overlap, and aspect ratio factors on box regression, and reformulate the 3D distance IoU (3D-DIoU) metric to guide 3D bounding-box optimization, as defined in Algorithm 1. Given the predicted (B_p) and its GT (B_g) boxes, we first calculate the 2-D IoU ($\text{IoU}_{\text{2-D}}$) and the intersection area (I_{gp}) in the BEV projection. In the case of 3D box regression, we further compute the 3D-IoU (IoU_{3D}) between two boxes by quantizing the box and intersection volumes (V_g, V_p, and I_v). To evaluate the effect of geometric attributes, we consider box center distance (D) and diagonal of the smallest closing volume (C), and finally, the 3D-DIoU (DIoU_{3D}) is formulated as the subtraction between 3D-IoU and the ratio square of center point and diagonal distance. In contrast to previous box regression methods, the newly developed

TABLE I

3D AP PERFORMANCE OF 3D-DFM AND THE STATE-OF-THE-ART DETECTORS ON THE KITTI TEST SET. ALL 3D DETECTION METHODS IN KITTI LEADERBOARD ARE RANKED BY THE AP PERFORMANCE AT MODERATE LEVEL UNDER THE IOU THRESHOLD OF 0.7. NOTED THAT "MOD.," "L," "C + L," "E," "M," AND "H" ARE THE ABBREVIATION OF "MODALITY," "LIDAR," "CAMERA + LIDAR," "EASY," "MODERATE," AND "HARD," RESPECTIVELY. IN PARTICULAR, THE RED-BOLD FRONTS DENOTE THE PERFORMANCE GAIN OR DROP COMPARED WITH VOXELNET AND PART-A^2 METHODS

Mod.	Methods	Car (AP %)			Pedestrian (AP %)			Cyclist (AP %)			mAP (%)	FPS
		E	M	H	E	M	H	E	M	H		
L	VoxelNet (CVPR2018) [20]	77.47	65.11	57.73	39.48	33.69	31.51	61.22	48.36	44.37	50.99	12
	Pointpillar (CVPR2019) [48]	82.58	74.31	68.99	51.45	*41.92*	*38.89*	77.10	58.65	51.92	60.65	**62**
	PointRCNN (CVPR2019) [30]	85.94	75.76	68.32	49.43	41.78	38.63	73.93	59.60	53.59	60.78	10
	SA-SSD (CVPR2020) [31]	88.75	*79.79*	74.16	-	-	-	-	-	-	-	*25*
	CenterNet3D (IEEE TITS2020) [39]	86.20	77.90	73.03	-	-	-	-	-	-	-	*25*
	Part-A^2 (IEEE TPAMI2020) [21]	87.81	78.49	73.51	**53.10**	**43.35**	**40.06**	*79.17*	*63.52*	**56.93**	**63.99**	12
C+L	MMF (CVPR2019) [6]	88.40	77.43	70.22	-	-	-	-	-	-	-	13
	PI-RCNN (AAAI2020) [49]	84.37	74.82	70.03	-	-	-	-	-	-	-	10
	PointPainting (CVPR2020) [1]	82.11	71.70	67.08	*50.32*	40.97	37.87	77.63	**63.78**	55.89	60.81	3
	CLOCs (IROS2020) [3]	86.38	78.45	72.45	-	-	-	-	-	-	-	10
	EPNet (ECCV2020) [2]	**89.81**	79.28	*74.59*	-	-	-	-	-	-	-	10
	3D-CVF (ECCV2020) [12]	*88.84*	79.72	72.80	-	-	-	-	-	-	-	12
	3D-DFM-VoxelNet	79.05	68.87	63.68	41.34	36.66	34.60	71.16	56.34	50.70	55.82	20
	vs. VoxelNet	**+1.58**	**+3.76**	**+5.95**	**+1.86**	**+2.97**	**+3.09**	**+9.94**	**+7.98**	**+6.33**	**+4.83**	**+8**
	3D-DFM-Part-A^2	87.75	**80.93**	**76.12**	46.93	39.68	37.31	**79.65**	63.38	*56.61*	*63.15*	16
	vs. Part-A^2	**-0.06**	**+2.44**	**+2.61**	**-6.17**	**-3.67**	**-2.75**	**+0.48**	**-0.14**	**-0.32**	**-0.84**	**+4**

3D-DIoU metric takes the box center, overlapping area, and aspect ratio into consideration, which is favorable for reflecting the similarity of two boxes and guiding the box optimization. Consequently, we further adopt the 3D-DIoU loss for box optimization as mathematically demonstrated in the following equation:

$$L_{\text{reg}} = 1 - \text{DIoU}_{\text{3D}}. \tag{4}$$

IV. EXPERIMENTS

In this section, we first describe implementation details. Subsequently, experimental results on the KITTI dataset are reported to evaluate our proposed 3D-DFM, and ablation studies are conducted to verify the effectiveness of each component. Finally, we visualize the detection results.

A. Implementations

1) Dataset Setup: The KITTI dataset [19] is one of the most challenging 3D object detection benchmarks for autonomous-driving applications, which provides 7481 training and 7518 testing samples. It is noted that both camera images and LiDAR points are available for model evaluation, and all training data are annotated with instance labels and calibration files. We split the training set into 3712 samples for training and 3769 samples for validation. The evaluation metric is the average precision (AP) at 0.7 IoU threshold. On the KITTI dataset, we also compare the detection performance of our proposed method to that of state-of-the-art single-modal and multimodal detectors at easy, moderate, and hard levels.

2) Training Details: The experimental platform is on Ubuntu18.04 LTS with NVIDIA RTX GPU. In the image stream, the input size of 1280×384 is reshaped to 1408×416 pixels for dimension alignment. For point stream, the range of point cloud is limited to $[0, 70.4]$ m in the x-axis, $[-40, 40]$ m in the y-axis, and $[-3, 1]$ m in the z-axis. As for point cloud extractor, we use one-stage VoxelNet and two-stage Part-A^2 detectors, with the default settings from official implementations. Our 3D-DFM is trained with a batch size of 8 for 80k steps, optimized by adamW [50] and a one-cycle learning rate policy, with a weight decay of 0.01, a momentum from 0.95 to 0.85, and a maximum learning rate of 0.003. Unless otherwise specified, all experiments would follow the same training setting.

B. Experimental Results

We primarily present the 3D detection performance in the car, pedestrian, and cyclist classes and then compare the proposed approach to other state-of-the-art single-model and multimodal detectors on the KITTI test set, as shown in Table I.

3D-DFM-VoxelNet takes VoxelNet for point feature encoding and replaces the convolutional middle layer with 3D sparse CNN for efficiency. It reports accurate and fast 3D detection performance with 55.82% mAP at 20 FPS, outperforming VoxelNet by a remarkable margin. To be specific, 3D-DFM-VoxelNet increases the baseline by 1.58%, 3.76%, and 5.95% car AP at three different levels, and similar performance gains are also reported in pedestrian and cyclist classes. It is highlighted that the inference speed of 3D-DFM-VoxelNet is up to 20 FPS, which approaches to the requirements of many real-time applications such as autonomous-driving systems.

As for 3D-DFM-Part-A^2, it first voxelizes the raw point cloud into regular grid and pools the candidates by RoI-aware operation at the refinement stage. 3D-DFM-Part-A^2 shows prominent performance with 80.93% and 76.12% AP in car class at moderate and hard levels, which outperforms all single-modal and multimodal detectors by a considerable

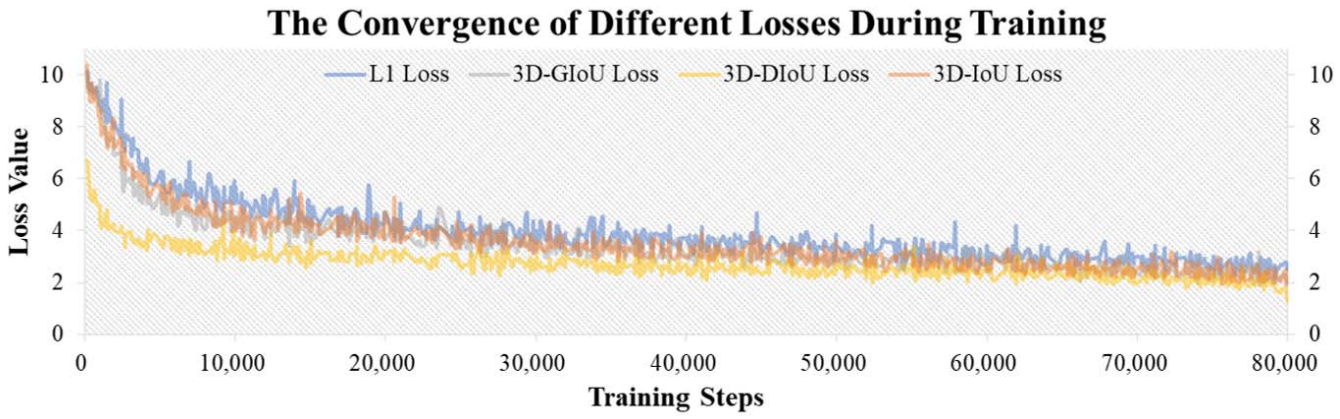

Fig. 4. Trend of $L1$-distance, 3D-IoU, 3D-GIoU, and 3D-DIoU loss functions during model training, and all experiments are performed on the basis of VoxelNet. It is observed that 3D-DIoU loss converges much faster than other three loss functions and finally drops at a smaller value. Best viewed in color.

TABLE II

3D CAR AP OF 3D-DFM AND OTHER DETECTORS ON THE KITTI VAL SET. NOTED THAT "MOD.," "L," "C + L," "E," "M," AND "H" ARE THE ABBREVIATIONS OF "MODALITY," "LIDAR," "CAMERA + LIDAR," "EASY," "MODERATE," AND "HARD," RESPECTIVELY. IN PARTICULAR, THE RED-BOLD FRONTS INDICATE THE PERFORMANCE GAIN OR DROP COMPARED WITH VOXELNET AND PART-A^2 BASELINES

Mod.	Methods	Car (AP%)		
		E	M	H
L	VoxelNet (CVPR2018) [20]	81.97	65.46	62.85
	Pointpillar (CVPR2019) [48]	83.62	75.22	72.40
	PointRCNN (CVPR2019) [30]	87.07	77.83	72.18
	Part-A^2 (IEEE TPAMI2020) [21]	89.33	81.59	76.05
C+L	PI-RCNN (AAAI2020) [49]	88.27	78.53	77.75
	CLOCs (IROS2020) [3]	**92.35**	*82.73*	78.10
	EPNet (ECCV2020) [2]	*92.28*	82.59	*80.14*
	3D-CVF (ECCV2020) [12]	89.67	79.88	78.47
	3D-DFM-VoxelNet	85.60	75.89	72.30
	vs. VoxelNet	**+3.63**	**+10.43**	**+9.45**
	3D-DFM-Part-A^2	91.94	**84.90**	**82.48**
	vs. Part-A^2	**+2.61**	**+3.31**	**+6.43**

TABLE III

ANALYSIS OF ANCHOR-FREE PIPELINE FOR VOXELNET AND PART-A^2 METHODS ON THE KITTI VAL SET. NOTED THAT "E," "M," AND "H" DENOTE "EASY," "MODERATE," AND "HARD" DIFFICULTY LEVELS, RESPECTIVELY

Methods	Car (AP%)			FPS
	E	M	H	
VoxelNet (CVPR2018)	81.97	65.46	62.85	12
	+0.81	**+1.68**	**+1.47**	**+13**
VoxelNet w. anchor-free	82.78	67.14	64.32	25
Part-A^2 (IEEE TPAMI2020)	89.33	81.59	76.05	12
	+0.13	**+0.44**	**+1.22**	**+8**
Part-A^2 w. anchor-free	89.46	82.03	77.27	20

margin. It also achieves the state-of-the-art accuracy with 79.65% AP on cyclist class at easy level among all approaches, demonstrating the superiority of the proposed method. In terms of speed, 3D-DFM-Part-A^2 runs at 16 FPS in one inference, which implies its possibility applied in a real-time system.

However, a significant performance drop is found in the pedestrian class. The function of DFM is assumed to conflict with the pooling operation in the box refinement stage, and this contradiction is amplified in small and few object detection, such as pedestrian detection. Furthermore, model robustness may result in a notable performance gap between validation and test splits. We would further investigate these problems and provide the effective solutions in future work.

In addition, we evaluate the detection performance on the KITTI val split and list the 3D AP results on the main car class in Table II. Our proposed method provides consistent performance gains over the two baseline methods, and 3D-DFM-Part-A^2 still demonstrates the state-of-the-art detection accuracy among all algorithms. In general, our 3D-DFM architecture facilitates the accuracy and efficiency of both one- and two-stage voxel-based 3D detectors.

C. Ablation Studies

Extensive ablation studies are conducted to evaluate the effectiveness of each component in the proposed 3D-DFM. Anchor-free pipeline, DFM, and 3D-DIoU loss are appended into the one-stage (VoxelNet) and two-stage (Part-A^2) baselines, respectively. Noted that all experiments are conducted on the KITTI val set, and we only present the 3D detection results in car class for convenience.

1) Anchor-Free Pipeline: We detach the traditional RPN head in VoxelNet and Part-A^2 and adopt anchor-free architecture for bounding-box prediction. As shown in Table III, anchor-free pipeline dramatically accelerates the inference speed of two baselines by 13 and 8 FPS, respectively. Moreover, it offers considerable AP gains at three difficulty levels, indicating the effectiveness and efficiency of anchor-free pipeline.

2) Dynamic Fusion Module: We introduce the image stream into VoxelNet and Part-A^2 architectures and measure the effect of different fusion approaches on detection performance

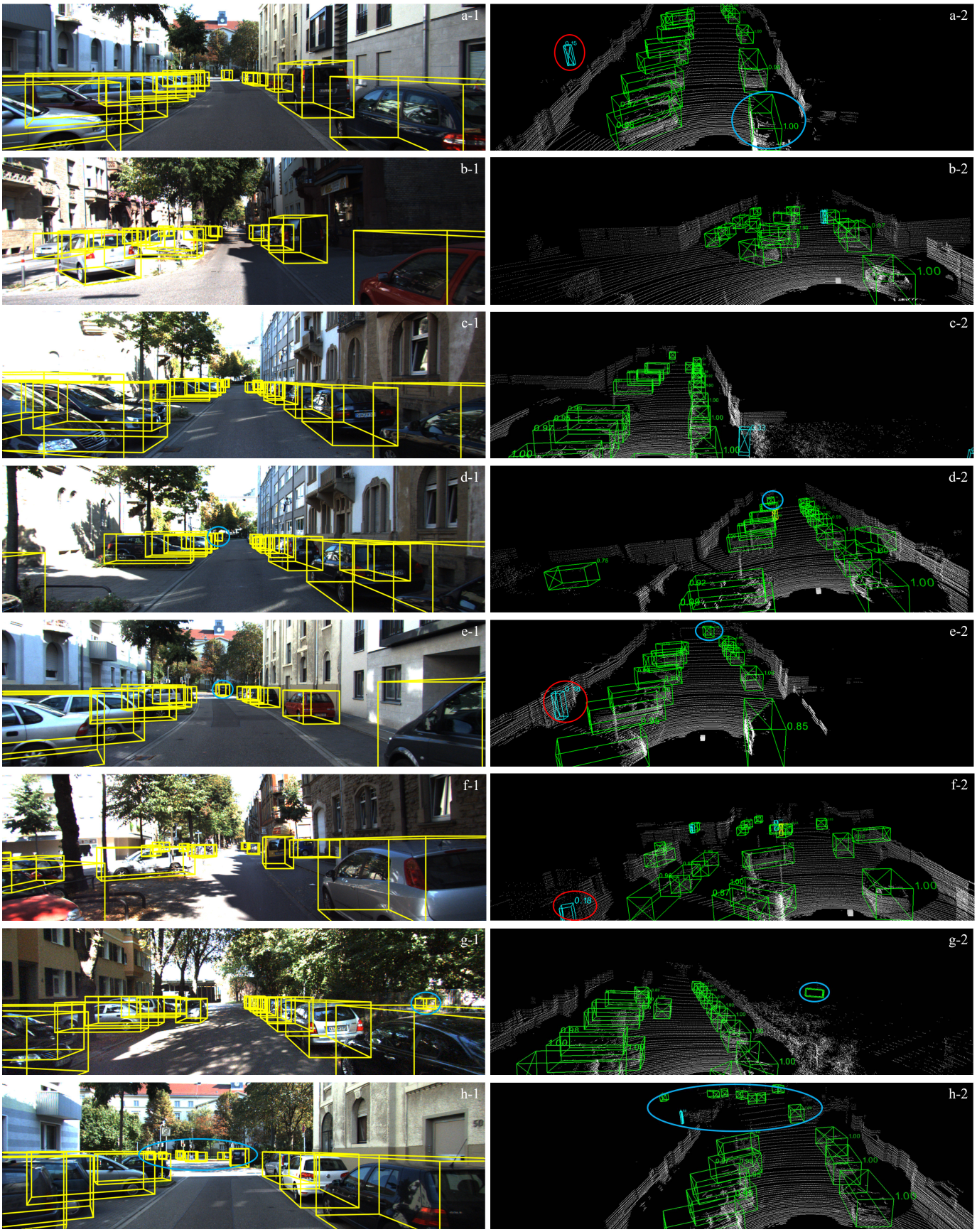

Fig. 5. Detection results of 3D-DFM-Part-A^2 on the KITTI test set. It is noted that our detector can find occluded or distant objects accurately, as highlighted in the blue circle. Also, some false positive detections are drawn in red. Best viewed in color.

TABLE IV

ANALYSIS OF VARIOUS FUSION METHODS FOR VOXELNET AND PART-A^2 METHODS ON THE KITTI VAL SET. NOTED THAT "E," "M," AND "H" DENOTE "EASY," "MODERATE," AND "HARD" DIFFICULTY LEVELS, RESPECTIVELY. "C," "$\oplus$," AND "$\otimes$" ARE FEATURE CONCATENATION, SUMMATION, AND MULTIPLICATION OPERATIONS, RESPECTIVELY

Methods	Car (AP%)			FPS
	E	M	H	
VoxelNet (CVPR2018)	81.97	65.46	62.85	12
	+1.25	**+1.87**	**+2.29**	**-3**
VoxelNet w. C	83.22	67.33	65.14	9
	+1.38	**+1.71**	**+2.64**	**-3**
VoxelNet w. $\oplus$	83.35	67.17	65.49	9
	+1.62	**+2.93**	**+3.56**	**-3**
VoxelNet w. $\otimes$	83.59	68.39	66.41	9
	+2.08	**+6.14**	**+4.92**	**-4**
VoxelNet w. DFM	84.05	71.60	67.77	8
Part-A^2 (IEEE TPAMI2020)	89.33	81.59	76.05	12
	+0.73	**+1.04**	**+1.66**	**-3**
Part-A^2 w. C	90.06	82.63	77.71	9
	+0.95	**+1.30**	**+2.01**	**-3**
Part-A^2 w. $\oplus$	90.28	82.89	78.06	9
	+1.22	**+1.48**	**+2.37**	**-3**
Part-A^2 w. $\otimes$	90.55	73.08	78.42	9
	+1.58	**+1.91**	**+3.13**	**-4**
Part-A^2 w. DFM	90.91	83.50	79.18	8

using DFM, feature concatenation, feature summation, and feature multiplication. As elaborated in Table IV, feature aggregation methods sacrifice detection speed marginally while providing substantial accuracy improvements. Specifically, concatenation, summation, and multiplication operations improve the detection performance by 1%–3% AP boosts in both VoxelNet and Part-A^2. The proposed DFM results in better performance gains particularly for 6.14% in VoxelNet and 3.13% in Part-A^2, demonstrating the superiority of dynamic fusion mechanism. Instead of simple concatenation or multiplication in a local region, it interacts image with point features adaptively at different locations using a sample-specific filter, and more discriminative information can be preserved to contribute to object localization.

3) 3D Distance Intersection-Over-Union: We further replace the $L1$ loss in VoxelNet and Part-A^2, with 3D-IoU, 3D-GIoU, and 3D-DIoU losses to analyze their contribution for box optimization, as shown in Table V. When compared to 3D-IoU and 3D-GIoU metrics, our 3D-DIoU loss presents 0.5%–3.0% AP gains for VoxelNet and 0.9%–2.0% AP improvements for Part-A^2, suggesting its suitability for 3D box optimization. It considers the center-point distance, overlapping area, and aspect ratio of two boxes in regression and therefore receives more accurate box regression results.

D. Qualitative Results

1) Convergence of 3D-DIoU Loss: We investigate the convergence and stability of 3D-DIoU loss with other loss functions, i.e., $L1$ distance, 3D-IoU, and 3D-GIoU loss functions. As shown in Fig. 4, 3D-DIoU loss converges much faster

TABLE V

ANALYSIS OF DIFFERENT LOSS FUNCTIONS FOR VOXELNET AND PART-A^2 METHODS ON THE KITTI VAL SET. NOTED THAT "E," "M," AND "H" DENOTE "EASY," "MODERATE," AND "HARD" DIFFICULTY LEVELS, RESPECTIVELY

Methods	Car (AP%)			FPS
	E	M	H	
VoxelNet (CVPR2018)	81.97	65.46	62.85	12
	+0.52	**+1.81**	**+1.98**	**-2**
VoxelNet w. 3D-IoU	82.49	67.27	64.83	10
	+0.65	**+2.23**	**+2.55**	**-2**
VoxelNet w. 3D-GIoU	82.62	68.07	65.40	10
	+0.74	**+2.61**	**+3.06**	**-2**
VoxelNet w. 3D-DIoU	82.71	69.94	65.91	10
Part-A^2 (IEEE TPAMI2020)	89.33	81.59	76.05	12
	+0.28	**+0.31**	**+0.69**	**-2**
Part-A^2 w. 3D-IoU	89.61	81.90	76.74	10
	+0.61	**+0.43**	**+1.24**	**-2**
Part-A^2 w. 3D-GIoU	89.94	82.02	77.29	10
	+0.90	**+0.96**	**+2.08**	**-2**
Part-A^2 w. 3D-DIoU	90.23	82.55	78.13	10

than the other three losses and ends up with a smaller value. The trend illustrates that the 3D-DIoU loss is preferable for 3D object detection, which is consistent with the above conclusion.

2) Detection Visualization: We finally visualize the results predicted by our proposed method on the KITTI test set in Fig. 5. Intuitively, 3D-DFM-Part-A^2 can accurately recognize object category and localization, even in the severely occluded or truncated cases, i.e., blue circles in Fig. 5. Nevertheless, our proposed method still suffers from false positive or low-confidence detection results, e.g., red circles in Fig. 5. We would probe into these problems in the future to ensure more robust and stable detection performance.

V. CONCLUSION

In this article, we incorporate an anchor-free pipeline with multimodal 3D object detection task and design an end-to-end architecture called 3D-DFM with DFM and 3D-DIoU loss. DFM, in particular, performs adaptive image-point feature aggregation using dynamically generated filters, whereas 3D-DIoU loss considers geometric properties of two boxes for better box optimization. Extensive experimental results on the KITTI dataset demonstrate the superiority and universality of 3D-DFM architecture. Based on one-stage VoxelNet, it reports real-time inference speed and considerable detection accuracy; with two-stage Part-A^2, 3D-DFM-Part-A^2 achieves the state-of-the-art detection performance among all single-modal and multimodal 3D detectors.

However, our 3D-DFM still fails to detect small or few objects correctly, such as pedestrian, and occasionally produces false positive results in some challenging scenarios. It is assumed that the data fusion mechanism and model robustness remain to be further improved. In the future, we would investigate these drawbacks and provide the solutions to pursue more accurate and promising 3D detection performance.

REFERENCES

[1] 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. 4604–4612.

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

[3] 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.

[4] 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. 11794–11803.

[5] Y. Wang, W.-L. Chao, D. Garg, B. Hariharan, M. Campbell, and K. Q. Weinberger, "Pseudo-LiDAR from visual depth estimation: Bridging the gap in 3D object detection for autonomous driving," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 8437–8445.

[6] M. Liang, B. Yang, Y. Chen, R. Hu, and R. Urtasun, "Multi-task multi-sensor fusion for 3D object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 7345–7353.

[7] W. Zhang, Z. Wang, and C. C. Loy, "Exploring data augmentation for multi-modality 3D object detection," 2020, *arXiv:2012.12741*.

[8] 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. 1907–1915.

[9] J. Ku, M. Mozifian, J. Lee, A. Harakeh, and S. 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.

[10] M. Liang, B. Yang, S. Wang, and R. Urtasun, "Deep continuous fusion for multi-sensor 3D object detection," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 641–656.

[11] V. A. Sindagi, Y. Zhou, and O. Tuzel, "MVX-Net: Multimodal Voxelnet for 3D object detection," 2019, *arXiv:1904.01649*.

[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] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollar, "Focal loss for dense object detection," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 2980–2988.

[14] Y. Chen, H. Li, R. Gao, and D. Zhao, "Boost 3D object detection via point clouds segmentation and fused 3D GIoU-L_1 loss," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 33, no. 2, pp. 1–12, Feb. 2020.

[15] J. Yu, Y. Jiang, and Z. Wang, "Unitbox: An advanced object detection network," in *Proc. 24th ACM Int. Conf. Multimedia (ACM MM)*, 2020, pp. 516–520.

[16] H. Rezatofighi, Tsoi, J. Gwak, A. Sadeghian, I. Reid, and S. Savarese, "Generalized intersection over union: A metric and a loss for bounding box regression," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 658–666.

[17] D. Zhou *et al.*, "IoU loss for 2D/3D object detection," in *Proc. Int. Conf. 3D Vis. (3DV)*, Sep. 2019, pp. 85–94.

[18] J. Xu, Y. Ma, S. He, and J. Zhu, "3D-GIoU: 3D generalized intersection over union for object detection in point cloud," *Sensors*, vol. 19, no. 19, p. 4093, Sep. 2019.

[19] 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.

[20] 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.

[21] 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. 2020.

[22] X. Chen, K. Kundu, Z. Zhang, H. Ma, S. Fidler, and R. Urtasun, "Monocular 3D object detection for autonomous driving," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 2147–2156.

[23] X. Weng and K. Kitani, "Monocular 3D object detection with pseudo-LiDAR point cloud," 2019, *arXiv:1903.09847*.

[24] Y. You *et al.*, "Pseudo-LiDAR++: Accurate depth for 3D object detection in autonomous driving," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2020, pp. 1–22.

[25] M. Ding *et al.*, "Learning depth-guided convolutions for monocular 3D object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 11672–11681.

[26] C. Reading, A. Harakeh, J. Chae, and S. L. Waslander, "Categorical depth distribution network for monocular 3D object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 8555–8564.

[27] 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. (AAAI)*, 2021, pp. 1–9.

[28] C. R. Qi, 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. 652–660.

[29] 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. Adv. Neural Inf. Process. Syst. (NIPS)*, 2017, pp. 5099–5108.

[30] 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.

[31] C. He, H. Zeng, J. Huang, X. Hua, and L. Zhang, "Structure aware single-stage 3D object detection from point cloud," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 11873–11882.

[32] S. Shi *et al.*, "PV-RCNN: Point-voxel feature set abstraction for 3D object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 10529–10538.

[33] L. Huang, Y. Yang, Y. Deng, and Y. Yu, "DenseBox: Unifying landmark localization with end to end object detection," 2015, *arXiv:1509.04874*.

[34] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 779–788.

[35] H. Law and J. Deng, "CornerNet: Detecting objects as paired keypoints," *Int. J. Comput. Vis.*, vol. 128, no. 3, pp. 642–656, Mar. 2020.

[36] X. Zhou, D. Wang, and P. Krähenbühl, "Objects as points," 2019, *arXiv:1904.07850*.

[37] R. Ge *et al.*, "AFDet: Anchor free one stage 3D object detection," 2020, *arXiv:2006.12671*.

[38] T. Yin, X. Zhou, and P. Krahenbuhl, "Center-based 3D object detection and tracking," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 11784–11793.

[39] G. Wang, B. Tian, Y. Ai, L. Chen, and D. Cao, "Centernet3D: An anchor free object detector for autonomous driving," *IEEE Trans. Intell. Transp. Syst.*, 2020, doi: 10.1109/TITS.2021.3118698.

[40] J. Dai *et al.*, "Deformable convolutional networks," in *Proc. Int. Conf. Comput. Vis. (ICCV)*, 2017, pp. 764–773.

[41] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.

[42] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," 2015, *arXiv:1502.03167*.

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

[44] B. D. Brabandere, X. Jia, T. Tuytelaars, and L. V. Gool, "Dynamic filter networks," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, 2016, pp. 1–10.

[45] Y. Pang, L. Zhang, X. Zhao, and H. Lu, "Hierarchical dynamic filtering network for RGB-D salient object detection," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2020, pp. 235–252.

[46] P. Sun *et al.*, "Sparse R-CNN: End-to-end object detection with learnable proposals," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 14454–14463.

[47] 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 Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 12698–12705.

[48] L. Xie *et al.*, "PI-RCNN: An efficient multi-sensor 3D object detector with point-based attentive cont-conv fusion module," in *Proc. AAAI Conf. Artif. Intell. (AAAI)*, 2020, pp. 12467–12469.

[49] Z. Zheng, P. Wang, W. Liu, J. Li, R. Ye, and D. Ren, "Distance-IoU loss: Faster and better learning for bounding box regression," in *Proc. AAAI Conf. Artif. Intell.*, vol. 34, no. 7, Apr. 2020, pp. 12993–13000.

[50] I. Loshchilov and F. Hutter, "Decoupled weight decay regularization," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2019, pp. 1–19.

Chunmian Lin is currently pursuing the Ph.D. degree with the School of Transportation Science and Engineering, Beihang University, Beijing, China.

His current research interests include multimodal fusion perception, image processing, computer vision, autonomous driving, artificial intelligence, and deep learning, particularly their applications in intelligent transportation systems.

Daxin Tian (Senior Member, IEEE) received the Ph.D. degree in computer application technology from Jilin University, Changchun, China, in 2007.

He is currently a Professor with the School of Transportation Science and Engineering, Beihang University, Beijing, China. He was awarded the Changjiang Scholars Program (Young Scholar) of Ministry of Education of China in 2017, the National Science Fund for Distinguished Young Scholars in 2018, and the Distinguished Young Investigator of China Frontiers of Engineering in 2018. His research is focused on intelligent transportation systems, autonomous-connected vehicles, swarm intelligence, and mobile computing.

Dr. Tian served as the Technical Program Committee Member/the Chair/Co-Chair for several international conferences, including International Conference on Transportation Information and Safety (ICTIS) 2019, IEEE International Conference on Unmanned Systems (ICUS) 2019, IEEE International Workshop on High Mobility Wireless Communications (HMWC) 2020.

Xuting Duan (Member, IEEE) received the Ph.D. degree in traffic information engineering and control from Beihang University, Beijing, China in 2018.

He is currently an Assistant Professor with the School of Transportation Science and Engineering, Beihang University. His current research interests include vehicular ad hoc networks, cooperative vehicle infrastructure systems, and the Internet of vehicles.

Jianshan Zhou received the Ph.D. degree in traffic information engineering and control from Beihang University, Beijing, China, in 2020.

He is currently a Post-Doctoral Research Fellow supported by the Zhuoyue Program of Beihang University and the National Postdoctoral Program for Innovative Talents. His research interests include the modeling and optimization of vehicular communication networks and air–ground cooperative networks, the analysis and control of connected autonomous vehicles, and intelligent transportation systems.

Dezong Zhao (Senior Member, IEEE) received the B.Eng. and M.S. degrees from Shandong University, Jinan, China, in 2003 and 2006, respectively, and the Ph.D. degree from Tsinghua University, Beijing, China, in 2010, all in control science and engineering.

He is currently a Senior Lecturer in autonomous systems with the School of Engineering, University of Glasgow, Glasgow, U.K. His research interests include connected and autonomous vehicles, machine learning, and control engineering.

Dr. Zhao's work has been recognized by being awarded an EPSRC Innovation Fellowship and a Royal Society-Newton Advanced Fellowship in 2018 and 2020, respectively.

Dongpu Cao received the Ph.D. degree from Concordia University, Montreal, QC, Canada, in 2008.

He is currently the Canada Research Chair in Driver Cognition and Automated Driving and an Associate Professor and the Director of the Waterloo Cognitive Autonomous Driving (CogDrive) Laboratory, University of Waterloo, Waterloo, ON, Canada. He has contributed more than 200 articles and three books. His current research focuses on driver cognition, automated driving, and cognitive autonomous driving.

Dr. Cao received the SAE Arch T. Colwell Merit Award in 2012, the IEEE VTS 2020 Best Vehicular Electronics Paper Award, and three best paper awards from the ASME and IEEE conferences. He serves as the Deputy Editor-in-Chief for *IET Intelligent Transport Systems* journal and an Associate Editor for IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY, IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS, IEEE/ASME TRANSACTIONS ON MECHATRONICS, IEEE TRANSACTIONS ON INDUSTRIAL ELECTRONICS, IEEE/CAA JOURNAL OF AUTOMATICA SINICA, IEEE TRANSACTIONS ON COMPUTATIONAL SOCIAL SYSTEMS, and *Journal of Dynamic Systems, Measurement and Control* (ASME). He was a Guest Editor of *Vehicle System Dynamics*, IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS: SYSTEMS, and IEEE INTERNET OF THINGS JOURNAL. He serves on the SAE Vehicle Dynamics Standards Committee and acts as the Co-Chair of the IEEE ITSS Technical Committee on Cooperative Driving.