3D Vehicle Detection Using Multi-Level Fusion From Point Clouds and Images

Kun Zhao, Lingfei Ma, *Member, IEEE*, Yu Meng, Li Liu, Junbo Wang, José Marcato Junior, *Member, IEEE*, Wesley Nunes Gonçalves, *Member, IEEE*, and Jonathan Li, *Senior Member, IEEE*

***Abstract*— 3D vehicle detectors based on point clouds generally have higher detection performance than detectors based on multi-sensors. However, with the lack of texture information, point-based methods get many missing detection of occluded and distant vehicles, and false detection with high-confidence of similarly shaped objects, which is a potential threat to traffic safety. Therefore, in the long run, fusion-based methods have more potential. This paper presents a multi-level fusion network for 3D vehicle detection from point clouds and images. The fusion network includes three stages: data-level fusion of point clouds and images, feature-level fusion of voxel and Bird's Eye View (BEV) in the point cloud branch, and feature-level fusion of point clouds and images. Besides, a novel coarse-fine detection header is proposed, which simulates the two-stage detectors, generating coarse proposals on the encoder, and refining them on the decoder. Extensive experiments show that the proposed network has better detection performance on occluded and distant vehicles, and reduces the false detection of similarly shaped objects, proving its superiority over some state-of-the-art detectors on the challenging KITTI benchmark. Ablation studies have also demonstrated the effectiveness of each designed module.**

***Index Terms*— 3D vehicle detection, deep learning, autonomous driving, false detection, point cloud processing, data fusion.**

Manuscript received March 25, 2021; revised November 26, 2021 and December 11, 2021; accepted December 16, 2021. This work was supported in part by the National Key Research and Development Program of China under Grant 2018YFE0192900 and in part by the National Natural Science Foundation of China under Grant 41871380 and Grant 42101451. The Associate Editor for this article was T.-H. Kim. *(Corresponding authors: Jonathan Li; Li Liu.)*

Kun Zhao is with the College of Mechanical Engineering, University of Science and Technology Beijing, Beijing 100083, China, and also with the Department of Geography and Environmental Management, University of Waterloo, Waterloo, ON N2L 3G1, Canada (e-mail: zhaokun1244@163.com).

Lingfei Ma is with the Engineering Research Center of State Financial Security, Ministry of Education, and the School of Statistics and Mathematics, Central University of Finance and Economics, Beijing 102206, China (e-mail: l53ma@cufe.edu.cn).

Yu Meng and Li Liu are with the College of Mechanical Engineering, University of Science and Technology Beijing, Beijing 100083, China (e-mail: myu@ustb.edu.cn; liliu@ustb.edu.cn).

Junbo Wang is with the College of the Environment and Ecology, Xiamen University, Xiamen 361102, China, and also with the Department of Geography and Environmental Management, University of Waterloo, Waterloo, ON N2L 3G1, Canada (e-mail: jeremywang@vip.163.com).

José Marcato Junior is with the Faculty of Engineering, Architecture and Urbanism and Geography, Federal University of Mato Grosso do Sul, Campo Grande 79070-900, Brazil (e-mail: jose.marcato@ufms.br).

Wesley Nunes Gonçalves is with the Faculty of Computer Science and the Faculty of Engineering, Architecture and Urbanism and Geography, Federal University of Mato Grosso do Sul, Campo Grande 79070-900, Brazil (e-mail: wesley.goncalves@ufms.br).

Jonathan Li is with the Department of Geography and Environmental Management, University of Waterloo, Waterloo, ON N2L 3G1, Canada (e-mail: junli@uwaterloo.ca).

Digital Object Identifier 10.1109/TITS.2021.3137392

I. Introduction

AS THE 'eyes' of autonomous driving systems, object detection is a prerequisite to ensure the safe operation of the system [1]. In recent years, with the development of deep learning techniques and the application of large-scale traffic scene data sets [2]–[4], the research on object detection has made great progress. Many detectors with high detection accuracy were proposed, which laid the foundation for their applications. However, some problems remain to be solved in the traffic scene, such as missing detection of occluded and distant objects [5], and false detection of similarly shaped objects [6], which threaten traffic safety. Therefore, these particular problems need to be further studied.

According to the different sensors used, there are two mainstream detection methods, the point-based methods [7], [8] and the fusion-based methods [9], [10]. Previous methods convert the 3D points into 2D views [7], [11]–[13] and then directly obtain 3D coordinates, size, and heading information via a 3D RPN network. These methods make the processing of disordered points simple and can leverage the mature 2D detectors to detect objects. However, the projection operation can lose some geometrically-related spatial information, resulting in lower detection accuracy. To avoid information loss, the pioneering method, PointNet [14], directly takes raw point clouds as the network input. Accordingly, some methods [8], [15]–[17] divide the points into a 3D voxel grid and utilize PointNet-based to extract the feature of each voxel cell, which retain more spatial information and greatly improve the detection accuracy. Point clouds are disordered, sparse, and lack texture information, consequently, point-based methods tend to provide poor detection performance for distant and occluded objects, and also generate false detection for similarly shaped objects. Fusion-based methods [9], [10], [18] usually take the point clouds as the main branch and the image as the auxiliary branch. The abstract feature maps of each branch generated by the extractors is calibrated and fused to perform 3D object detection. In addition, there is a cascading fusion strategy [19], [20] to obtain 2D proposals on the image, and perform 3D refinement on the corresponding point frustum. With the rich texture information in images, the fusion-based methods can overcome the shortcomings of the point-based methods. However, the fusion strategy needs further investigation. The point clouds provided by LiDAR are a set of surface points sparsely distributed in 3D space. The detectors learn the shape characteristics according to

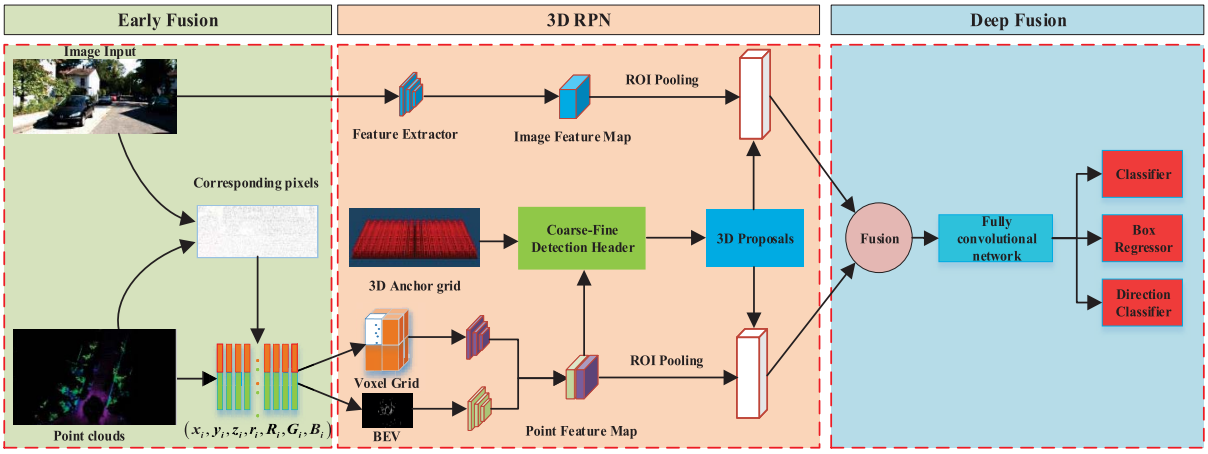

Fig. 1. The framework of our detector. It consists of three parts, early fusion module for the data-level fusion of point clouds and images; 3D RPN for proposal estimation with a high recall from point clouds branch, and feature extraction from the image; deep fusion module for proposal refinement from the features of point clouds and images.

the interaction relationship of each point, and predict the classification probability and the 3D bounding box. However, in complex traffic scenes, there are many objects similar to the shape of the vehicle. If only relying on the shape features, it is difficult for the detector to distinguish the similarly shaped objects correctly. Besides, for distant and occluded objects, LiDAR can only capture a small number of surface points associated with the objects. Since these surface points cannot provide enough semantic information, resulting in the missing detection.

Based on the above reasons, this paper proposes a multi-level fusion network, as shown in Fig. 1. A data-level fusion provides points with the rough texture information from RGB images in the early fusion module. Then the point clouds are encoded into two formats of voxel grid and Bird's Eye View (BEV), their abstract features are extracted and fused to output the proposals with high recall via a novel coarse-fine detection header. The proposed detection header simulates a two-stage detection network to obtain coarse proposals on the encoder and refine them on the decoder. Finally, the deep fusion module improves the confidence of positive samples by further fusing the image features, reducing the false detection. The experimental results prove that our fusion strategy, and coarse-fine detection header are effective for improving detection accuracy. The visualization results also show that our method can effectively reduce the missing detection of occluded and distant objects, and the false detection of similarly shaped objects.

The main contributions of this paper can be summarized as follows:

- The proposed multi-level fusion network can detect 3D vehicles from point clouds and images with competitive detection accuracy and efficiency in traffic scenes.
- Data-level and feature-level fusion strategy (early and deep fusion modules) can fully improve the efficiency of data utilization and greatly improve the detection performance of our network, especially for occluded and distant objects, reducing the false detection of similarly shaped objects.
- A novel coarse-fine detection header containing coarse and fine regressors is proposed, which can obtain the precise position and 3D shape information on the shallow features, and reset the precise semantic information on the deep features.

The paper is organized as follows. The related work is given in Section II. Our proposed method is presented in Section III, with the data set and experimental study given in Section IV. The conclusion is addressed in Section V.

II. Related Work

The existing work related to 3D detection can be grouped into three categories [1]: image-based, point-based, and fusion-based methods. With the lack of depth information in images, the performance of image-based methods is generally poor, so this section only introduces two other mainstream methods, i.e., point-based methods and fusion-based methods.

A. Point-Based Methods

The point-based methods can be subdivided into methods based on multi-view maps, raw point clouds, and voxel grid according to the processing methods for point clouds.

1) Methods Based on Multi-View Maps: To apply the mature 2D detection framework [21]–[23], some methods project the point cloud as pseudo images, and extend the detection result to the 3D space. [11], [24], [25] project the point cloud as a front view, while [7], [12], [26] project the point clouds as a BEV maps. Unlike the front views and RGB images, each object occupies an independent spatial position in BEV maps, which helps alleviate the problem of occlusion. BEV representation has become the mainstream of projection methods. The projection operation inevitably causes geometrically-related spatial information loss, multi-view maps have become auxiliary representations of the network inputs.

2) Methods Based on Raw Point Clouds: Avoiding the spatial information loss caused by projection, PointNet [14] is a pioneering work, which directly uses the raw point clouds as the network input. It utilizes two spatial transformation networks (STN) to deal with the rotation invariance of the point clouds. While the symmetric function, the max-pooling operation is beneficial for solving the disorder of point clouds. For abstracting the local feature, PointNet++ [27] constructs a set abstract layer to convert the raw point clouds into a set of local regions and fine features are extracted hierarchically. The disorder of the point clouds is the main problem that prevents the CNN network from directly operating on the raw points. Therefore, PointCNN [28] introduces a transformation matrix that can process the points in a specific order to obtain a feature that is independent of the order.

3) Methods Based on Voxel Gird: Some methods divide the points into a voxel grid to efficiently use the advantages of the CNN network. Previous works [29], [30] design the hand-crafted feature for each voxel cell and train an SVM classifier to detect objects by sliding window search. Their variants [31], [32] used 3D CNN instead of an SVM classifier, improving the detection performance. According to whether there are valid points in the voxel cell, Li *et al.* [16] code the point clouds as a binary grid and apply a 3D fully convolutional neural network to extract the global feature. Inspired by PointNet, VoxelNet [8] replaces the hand-crafted feature with a learning method that includes an MLP and a max-pooling to extract abstract features from each cell. The computation cost for such methods is usually quite high due to the expensive cost of 3D convolutions and large 3D search space. To improve the efficiency of 3D convolution, SECOND [15] introduces a sparse convolution network, which can avoid useless calculations for empty cells.

The point clouds are sparse and unevenly distributed; therefore, point-based methods cannot detect occluded and distant objects well. Besides, the detectors have poor robustness to discriminate the similarly shaped objects with the lack of texture information.

B. Fusion-Based Methods

The RGB images contain rich texture information, and the point cloud can provide accurate depth information. Therefore, the fusion-based methods can make full use of the advantages of different sensors, which is essential to improve the detection performance of special objects (occluded and distant objects). MV3D [9] takes the BEV maps, front views, and images as the network inputs, constructs three independent feature extractors, and obtains the proposals on the BEV branch through a 3D RPN network. Finally, the feature regions corresponding to the proposals from three branches are fused to obtain the refined detection results via a deep fusion module. Different from the fusion strategy of MV3D, AVOD [10] only constructed two feature extractors for images and BEV branches. The RPN then uses both feature maps to generate non-oriented region proposals. Besides, an early fusion module is introduced to refine the detection results. F-PointNet [19] and F-ConvNet [20] adopt a novel hierarchical detection strategy. According to the 2D proposals generated from the RGB images, corresponding frustums on point cloud space are extracted to detect 3D objects through the point-based methods.

The fusion-based methods should have higher performance than other methods, but the opposite is true. This shows that the existing fusion strategy is inefficient, reducing the detection performance. Therefore, a more effective fusion strategy becomes the key to improving the performance of the fusion methods.

III. 3D Vehicle Detector

The proposed network, depicted in Fig. 1, consists of three components: (1) early fusion, data-level fusion from point clouds and images; (2) RPN network, bounding box prediction from BEV and voxel grid representations, and feature extraction from images; (3) deep fusion, bounding box refinement from fused features. We introduce each module in the following subsections.

A. Early Fusion

Previous works usually focus only on feature-level fusion of point clouds and images, while ignoring the data-level fusion. Data-level fusion can only achieve the fusion of points and limited image pixels due to the sparsity of point clouds. Nevertheless, it cannot be ignored that the limited pixels can provide rough texture information for the point clouds, which is very useful for improving point cloud representation for the scene.

Therefore, an early fusion module is designed for data-level fusion of point clouds and RGB images. The points and pixels are matched according to the sensor calibration parameters, and the color information of the pixels is concatenated with the point features. In this way, the spatial characteristics of the point clouds are maintained, and the semantic features of the surface points are enriched, avoiding the dependence of the feature extractor of point clouds on the object shape.

To improve real-time and ensure the matching of the images and the point clouds, irrelevant points outside the camera's field of view are filtered out, and the detection range is set to $\{[x, y, z]^T \mid x \in [0, 70.4]m, y \in [-40, 40]m, z \in [-3, 1]m\}$. Finally, each point includes not only the 3D coordinates and reflection intensity in the LiDAR coordinate system, but also the color information of the corresponding pixels on the image plane, which can be expressed as: $p_i = (x_i, y_i, z_i, r_i, R_i, G_i, B_i)$.

B. 3D Region Proposal Network

The 3D RPN network proposed in this work takes two different representations (voxel grid and BEV) as inputs. The abstract features of two branches are fused, and a coarse-fine detection header is introduced to obtain the proposals with high recall. The image branch uses VGG16-like network to extract abstract semantic features.

1) Point Cloud Representation: Different representations of point clouds are beneficial to improve the robustness of the detection network. Therefore, the point clouds are processed into voxel grid and BEV coding in this work.

a) Voxel grid: Voxel grid is the most popular form of coding in the current mainstream networks, which can efficiently represent large-scale traffic scenes, and provide richer spatial features for detectors. To generate the voxel grid, the detection area of point clouds is evenly divided into several cells with fixed size $0.2m \times 0.2m \times 0.4m$, and the points are allocated to the corresponding cells. Due to the influence of distance and occlusion, the number of points in each cell is extremely unbalanced. For the convenience of calculation, the density threshold of each cell is set to T. If the number of points in a given cell exceeds the threshold, T points are randomly sampled from the cell. Otherwise, zero filling is used. In order to make full use of the interaction among points, the initial point feature is augmented with $(x_i - x^m, y_i - y^m, z_i - z^m)$, where m denotes the arithmetic mean coordinate of all points within a cell. Finally, each point feature contains ten dimensions.

b) BEV: BEV generated by projection can be regarded as a pseudo image, which could be directly processed by traditional convolution structure. Although projection can cause information loss, this operation makes each object occupies an independent spatial position, which is conducive to reflect the relative position relationship between objects and alleviate the interference of overlapping and occlusion problems. According to the method described in [10], the 3D point clouds are converted into a six-channel BEV maps with five height maps and one density map.

The detection region is cut into several squares with a resolution of $0.1m$ on the x-y plane. The 3D point clouds within detection region are divided into five equal slices along the Z-axis, each is associated with a height map. For each height map, height features are encoded as the maximum height of the points within this slice. The density map is encoded by the number of points N within each cell, which is computed as:

$$min(1.0, \frac{\log(N+1)}{\log 16}) \tag{1}$$

2) Feature Extractors: Feature extractors for each branch, i.e. voxel grid, BEV and image, are adopted to extract abstract features. BEV and image branches adopt the traditional convolution structure, while feature extraction of voxel grid branch follows SECOND [15]. Firstly, voxel feature extractor (VFE) is used to extract the features of each cell, and then the global features of voxel grid are extracted by sparse convolution network.

a) Voxel grid branch: As shown in Fig.2, the voxel feature extractor with a linear layer [8] is used to obtain abstract features from each cell, generating the voxel feature grid with size $(10 \times 400 \times 352)$. To avoid invalid computation for empty voxels, sparse convolution network is adopted to extract global features of voxel grid. Here we use the parameter setting of SECOND. The convolution kernel is set to $(3 \times 1 \times 1)$, and the stride is set to $(2 \times 1 \times 1)$. Finally, tensor features with size $(64 \times 2 \times 400 \times 352)$ are generated after two sparse convolution layers, which can be extracted by traditional convolution operation after reshaping to $(128 \times 400 \times 352)$.

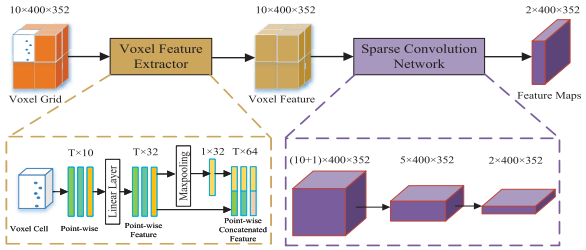

Fig. 2. The structure of voxel branch. It consists of an VFE and a sparse convolution network.

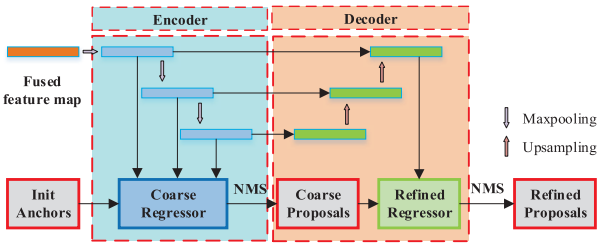

Fig. 3. The structure of coarse-fine detection header. It consists of an encoder and a decoder. Rough proposals are obtained on the encode and refined proposals are obtained on the decoder.

b) BEV and image branch: The BEV branch uses convolution operation and a maximum pooling to match the size of voxel features, generating a same feature vector with the size of $(128 \times 400 \times 352)$. The image branch adopts Feature Pyramid Network (FPN) [33] structure. First, feature encoder with $8\times$ down-sampling is used to extract abstract features, and then feature decoder with $8\times$ up-sampling is performed to restore the feature resolution.

3) Coarse-Fine Detection Header: The traditional FPN consists of an encoder and a decoder, and obtains the detection results on the decoder. Because of the translation invariance of convolution operation, the position information of the object becomes increasingly blurred with the increase in the number of convolution layers. Therefore, more accurate location information is retained on shallow feature maps. In this work, we improve the structure of FPN. Rough proposals are obtained from the last layers of all feature blocks of the encoder, expecting to get a higher recall. On the last convolutional block of the decoder with strong semantic information, the rough proposals are fine tuned. As shown in Fig. 3, after fusing voxel grid and BEV features, the rough regressor obtains proposals on the encoder feature maps of all scales to avoid losing positive samples. To improve the precision of the proposals, the refined regressor performs ROIpooling on the last feature layer of the decoder to refines the proposals. There are two Non-Maximum Suppressions (NMS) in the detection header to reduce the number of proposals after the coarse and the fine regressors.

C. Deep Fusion

The number of effective points reflected from the distant and occluded objects is small, which may not be enough to

detect the expected objects. Therefore, a deep fusion module is introduced to improve the confidence of positive proposals through fusing the feature blocks of image branch. As shown in Fig. 1, the two feature sources are merged to obtain the final detection result through a lightweight fully convolutional network (FCN) [38]. Since the NMS has already been performed on the 3D RPN stage, the number of proposals in this stage is small, not taking too much inference time.

D. Anchors and Targets

In the KITTI data set, the size of all labeled vehicles is usually approximately fixed. According to the statistics, the average size of anchors is set to $w_a = 1.6\ m$, $l_a = 3.9\ m$, $h_a = 1.56\ m$. Assuming that all vehicles are constrained on the road, all anchors are placed at $z_a = -1.0m$ with two rotations, i.e. 0 and 90 degrees. The objectness of anchors is defined by the intersection-over-union (IoU) between the ground truth on the BEV plane. If the IoU value exceeds the positive matching threshold, the anchors are considered positive. Conversely, if the IoU value is less than the negative matching threshold, the anchor is considered negative. The anchors with IoU between the positive and negative threshold are ignored. The threshold is usually an empirical value. Following the parameter settings of most detectors, the positive matching threshold is set to 0.6, and the negative matching threshold is set to 0.45 in this work.

In the dataset, the 3D ground truth boxes are denoted as $(x_g, y_g, z_g, l_g, w_g, h_g, \theta_g)$, where (x_g, y_g, z_g) is the central coordinates of the box, (l_g, w_g, h_g) is the dimension, and θ_g is the yaw rotation around Z-axis. Similarly, in this paper, the anchors are set to $(x_a, y_a, z_a, l_a, w_a, h_a, \theta_a)$. Following the parameter settings in SECOND, the regression target between anchors and ground truth is encoded by Eqs. (2):

$$\begin{aligned} &\Delta x = \frac{x_g - x_a}{d_a}, \quad \Delta y = \frac{y_g - y_a}{d_a}, \quad \Delta z = \frac{z_g - z_a}{d_a} \\ &\Delta l = \log(\frac{l_g}{l_a}), \quad \Delta w = \log(\frac{w_g}{w_a}), \quad \Delta h = \log(\frac{h_g}{h_a}) \\ &\Delta\theta = \theta_g - \theta_a \end{aligned} \tag{2}$$

where Δx, Δy, Δz are the offsets between center coordinates of anchor and ground truth. They are normalized by the diagonal of the base of anchors: $d_a = \sqrt{(l_a)^2 + (w_a)^2}$.

The reasonable regression target can achieve higher detection performance. Through the above normalized calculation, the detection robustness of vehicles with different sizes can be improved.

E. Loss Function

The loss function in this work includes three parts: loss for coarse proposals, refined proposals and refined results. Each part of the loss includes object classification loss, regression loss and direction classification loss. Focal loss [39] is used for object classification loss, Smooth L1 loss function for regression loss and Cross Entropy loss function for direction classification loss.

1) Object Classification Loss: In order to solve the imbalance of foreground and background in the samples, the Focal Loss function is used to construct the classification loss function as follows:

$$L_{cls} = -\alpha(1-p)^{\lambda}\log(p) \tag{3}$$

where p is the estimated category probability of each anchor, α is a weighting factor to balance the importance of positive and negative examples, and λ is a focusing parameter to down-weight the contribution of easily-classified examples and allow the model to focus on hard examples.

2) Regression Loss: In addition to classification task, anchors should be fine-tuned to obtain more accurate 3D information, including location, size, and orientation. Location and size offsets can be directly regressed, but the radian offset is subject to an adversarial example problem. For example, 0 and π radians correspond to the same box but lead to a large loss. A sine function [15] is used to solve this problem. The total regression loss is calculated by a Smooth L1 loss function:

$$L_{reg} = \Sigma_{b\in(x,y,z,w,l,h)} \ \text{SmoothL1}(\Delta b) \tag{4}$$

$$L_{\theta} = \text{SmoothL1}(\sin(\Delta\theta)) \tag{5}$$

where L_{reg} and Δb are the regression loss and offset for location and dimension, L_θ and $\Delta\theta$ are the special angle loss and offset.

3) Direction Classification Loss: Since the regression loss cannot identity flipped boxes, a SoftMax classification loss, L_{dir}, is used to determine whether the orientation of prediction is inversed. The target of direction classifier is set as: in the x-y plane, if the heading angle of the vehicle is within the first or second quadrant, the result should be positive; otherwise, it should be negative.

In summary, the total loss function is:

$$L_{total} = \Sigma_{i\in(coarse,fine,fusion)} L_i \tag{6}$$

$$L_i = \beta_{cls}L_{cls} + \frac{1}{N_{pos}}(\beta_{reg}(L_{reg} + L_\theta) + \beta_{dir}L_{dir}) \tag{7}$$

where N_{pos} is the number of positive anchors, β_{reg}, β_{cls} and β_{dir} are respectively the weight of regression, classification, and direction classification. An experiment is also shown in the ablation study to illustrate that $\beta_{reg} = 2.0$, $\beta_{cls} = 1.0$, $\beta_{dir} = 0.2$ is most appropriate.

IV. Experiments

This section evaluates our proposed method and compares it with several state-of-the-art methods on the KITTI dataset. This dataset contains, respectively, 7481 and 7518 training and testing samples, which are divided into three difficulty levels (easy, moderate, and hard) based on the object pixel height, occlusion, and truncation. A total of 7481 training samples are divided into 3712 samples for training and 3769 samples for validation. Following the official evaluation protocol, the BEV and 3D detection results were evaluated in terms of the AP (IoU threshold is set to 0.7). In addition, some ablation experiments have been performed to demonstrate the effectiveness of some settings and novel modules.

TABLE I

PERFORMANCE COMPARISON IN 3D AND BEV VEHICLE DETECTION: AVERAGE PRECISION (AP) ON KITTI VAL SET

Methods	Inference Time(ms)	$AP_{3D}(\%)$			$AP_{BEV}(\%)$		
		Easy	Moderate	Hard	Easy	Moderate	Hard
MV3D	360	71.29	62.68	56.56	86.55	78.10	76.67
AVOD	100	84.41	74.44	68.65	-	-	-
F-PointNet	170	83.76	70.92	63.65	88.16	84.02	76.44
ContFusion	60	86.32	73.25	67.81	95.44	87.34	82.43
F-ConvNet	470	89.02	78.8	77.09	90.23	88.79	86.84
MMF[34]	80	87.90	77.86	75.57	96.66	88.25	79.60
PI-RCNN[35]	100	87.63	77.87	76.17	-	-	-
3D-CVF[36]	60	89.67	79.88	78.47	-	-	-
Ours-MLF	139	89.52	80.35	78.93	91.29	89.53	88.06
EPNet[37]	100	92.28	82.59	80.14	95.51	88.76	88.36

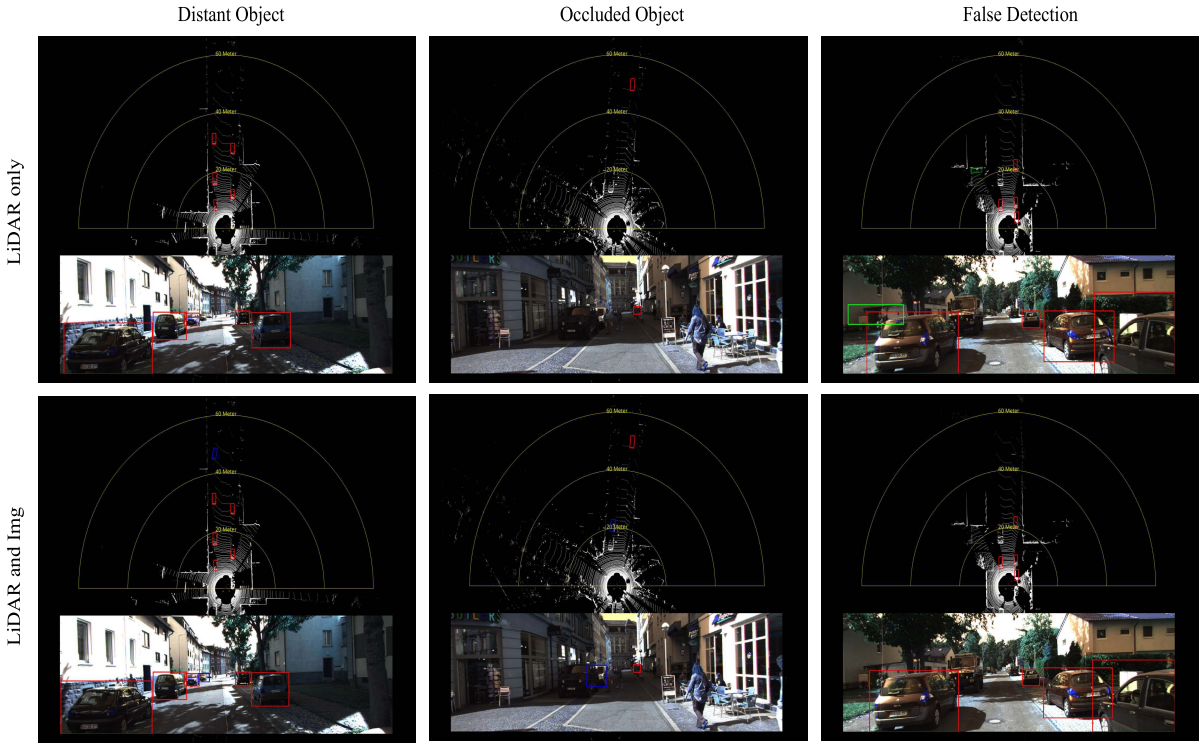

Fig. 4. Detection examples of 3D detection on the KITTI dataset. Top: Our network with point cloud only as input; Bottom: Our network with point cloud and image as inputs. Distant object detection (left), occluded object detection (middle), and false detection (right). The red boxes indicate the vehicles detected by both two networks. The blue boxes indicate the vehicles detected by the network with point cloud and image as input. The green box indicates the false detection sample. The radiuses of yellow semicircles are 20m, 40m, and 60m respectively. It can be seen that the network fused image has better robustness to alleviate the problems of occluded object detection, distant object detection and false detection. Note that the closest car in the middle sample is a truck, so it is not a missing detection.

A. Quantitative Analysis

The performance comparison of our MLF and other SOTA methods in 3D detection and BEV detection for vehicles are presented in Table I. All methods use images and point cloud as inputs. In the 3D detection, the APs of our MLF can achieve 89.52%, 80.35%, and 78.93% in three difficulties respectively, which are 2.76%, 2.24%, and 1.21% lower than the best method, EPNet. Different from the data-level fusion of 3D points and image pixels in our MLF, EPNet has introduced a novel structure called LI-Fusion Module, which can realize the feature-level fusion of each 3D point and the corresponding pixel. This is the reason why our MLF performs worse than the EPNet. Compared with 3D-CVF, the detection accuracy of our MLF are better, 0.47% and 0.46% higher in moderate and hard levels, only 0.15% lower in easy level. In the BEV detection, the AP of MLF is 4.22% lower than that of EPNet in easy level, while in the moderate and hard levels, the gaps are reduced to 0.77% and 0.3%. This is due to the introduction of the BEV encoding from point cloud in our MLF, which can intuitively reflect the relative position between objects. In summary, although the detection accuracy and speed are not the best, compared with other SOTA methods, our MLF can still achieve a competitive detection performance via the multi-modal and multi-stage fusion of images and point cloud.

In order to intuitively compare the performance of fusion-based method and point-based method, Fig. 4 shows some results from three aspects: occluded object detection, remote object detection and false detection. It can be seen that the

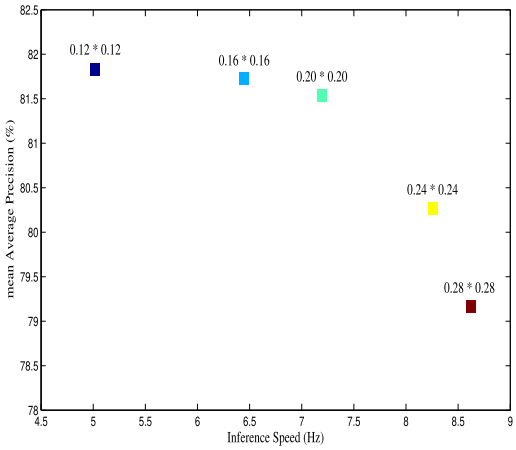

Fig. 5. Relationship between the detection performance and computational complexity of different spatial resolutions. The voxel cell sizes are set as $0.4m \times \{0.12^2, 0.16^2, 0.20^2, 0.24^2, 0.28^2\}m^2$ alone Z-Y-X axes, respectively.

TABLE II
RESULTS ON THE WEIGHT SETTING OF THE LOSS FUNCTION

classification weight	localization weight	direction weight	Average Precision (%)		
			Easy	Moderate	Hard
1.0	1.0	0.2	89.07	78.69	76.57
2.0	1.0	0.2	88.74	77.27	70.49
1.0	2.0	0.2	89.52	80.35	78.93

fusion-based method has better performance for the problems that often appear in traffic scenes.

B. Ablation Studies

1) Voxel Size: The spatial resolution of voxel grid has a significant impact on the accuracy and efficiency of the detector. Smaller cells allow finer localization, while larger cells make the network run faster. Fig. 5 shows the impact of different resolutions on accuracy and efficiency. It corroborates that the smaller voxel is beneficial to the precision while the larger voxel can reduce the computational complexity. In order to achieve a trade-off of accuracy and efficiency, the size of the voxel in this work is set to $0.4m \times 0.20^2 m^2$.

2) Loss Weight for Different Tasks: In this work, the loss function consists of three parts: objectness classification loss, regression loss and direction classification loss. The weight of each loss has a great impact on the detection results. A smaller weight is assigned to direction classification loss, since it is an auxiliary task to distinguish whether the direction is reversed. For objectness classification and regression loss, we set up experiments in three different situations: dominated by classification task, dominated by regression task, and balanced by both two. The results in Table II show that the network needs to pay more attention to the regression of 3D information of vehicles to obtain better performance.

3) Coarse-Fine Detection Header: Considering that the location information is more accurate on the shallow feature maps, while the semantic information is more explicit on the deep feature maps, this work proposes a coarse-fine detection header to obtain a higher recall, ensuring that most positive proposals can be detected. Table III compares our coarse-fine detection header with the single-scale detection header and the multi-scale detection header. The results prove that the coarse-fine detection header has more advantages, especially for challenging samples (hard difficulty).

TABLE III
COMPARISON OF DIFFERENT DETECTION HEADERS

Detection Header	Average Precision (%)			
	mAP	Easy	Moderate	Hard
Single-scale	80.67	87.34	78.84	75.82
Multi-scale	81.03	88.77	79.21	75.10
Coarse-fine	82.93	89.52	80.35	78.93

TABLE IV
COMPARISON OF DIFFERENT FUSION MODES

Fusion Methods	Average Precision (%)			
	mAP	Easy	Moderate	Hard
No Image	80.64	88.38	77.93	75.60
Early-Fusion	80.90	88.51	78.20	75.98
Deep-Fusion	81.21	88.69	78.45	76.48
Both-Fusion	82.93	89.52	80.35	78.93

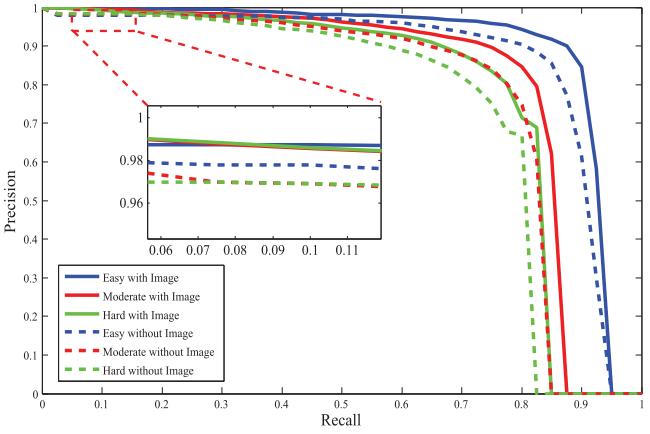

Fig. 6. PR curves of the point-based network and the fusion-based network in three difficulties respectively.

4) Fusion Mode: The image is rich in texture information, which is very useful for detecting occluded objects, distant objects and alleviating false detection. The image fusion in this work includes two parts: early fusion and deep fusions. Table IV shows the performance comparison of no image fusion, early fusion, deep fusion, and both fusion. Experimental results prove that the performance of the network has been greatly improved via integrating early fusion and deep fusion.

5) PR Curve: Fig. 6 shows the Precision-Recall (PR) curves of the point cloud-based network and the fusion-based network in three difficulties. It reveals that under the low recall (taking 0.1 as an example), the fusion method has a higher accuracy, which proves that the fusion based network can alleviate false detection.

V. Conclusion

In this work, we have proposed a multi-level fusion network for 3D vehicle detection based on images and point clouds to improve the detection performance of occluded and distant objects and reduce the false detection of similarly shaped objects. Different from other fusion-based methods, we introduce an early fusion module to perform a data-level fusion of images and point clouds, giving the point cloud rough texture information. In the point clouds branch, the 3D points are represented as voxel grid and BEV to enhance the ability for characterizing the traffic scene; a novel coarse-fine detection header is proposed to generate the coarse results on the encoder feature map with accurate position information, and the refined results on the decoder with high semantic information. In the deep fusion module, the feature maps of the image branch are further fused with the point cloud feature maps, which is essential to reduce the false detection of the similarly shaped object. Experimental results show that our method has higher performance than some SOTA methods, especially for occluded and distant objects, and also reduces the false detection of similarly shaped objects. The ablation experiments also prove that the modules proposed in this work is effective for improving the detection performance. The network structure of this work is more complicated, which leads to a slightly worse real-time performance than other methods. Further research is to consider the lightweight for network model.

Acknowledgment

Kun Zhao greatly acknowledges the China Scholarship Council (CSC) for the graduate fellowship.

References

[1] E. Arnold, O. Y. Al-Jarrah, M. Dianati, S. Fallah, D. Oxtoby, and A. Mouzakitis, "A survey on 3D object detection methods for autonomous driving applications," *IEEE Trans. Intell. Transp. Syst.*, vol. 20, no. 10, pp. 3782–3795, Oct. 2019.

[2] 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.

[3] H. Caesar *et al.*, "NuScenes: A multimodal dataset for autonomous driving," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 11621–11631.

[4] P. Sun *et al.*, "Scalability in perception for autonomous driving: Waymo open dataset," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2020, pp. 2446–2454.

[5] S. K. Kwon, E. Hyun, J.-H. Lee, J. Lee, and S. H. Son, "A low-complexity scheme for partially occluded pedestrian detection using LiDAR-radar sensor fusion," in *Proc. IEEE 22nd Int. Conf. Embedded Real-Time Comput. Syst. Appl. (RTCSA)*, Oct. 2016, p. 104.

[6] T.-F. Ju, W.-M. Lu, K.-H. Chen, and J.-I. Guo, "Vision-based moving objects detection for intelligent automobiles and a robustness enhancing method," in *Proc. IEEE Int. Conf. Consum. Electron.-Taiwan*, 2014, pp. 75–76.

[7] J. Beltrán, C. Guindel, F. M. Moreno, D. Cruzado, F. Garcia, and A. De La Escalera, "BirdNet: A 3D object detection framework from LiDAR information," in *Proc. 21st Int. Conf. Intell. Transp. Syst. (ITSC)*, 2018, pp. 3517–3523.

[8] 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.

[9] 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.*, Dec. 2017, pp. 1907–1915.

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

[11] B. Li, T. Zhang, and T. Xia, "Vehicle detection from 3D LiDAR using fully convolutional network," 2016, *arXiv:1608.07916*.

[12] M. Simony, S. Milzy, K. Amendey, and H.-M. Gross, "Complex-YOLO: An euler-region-proposal for real-time 3D object detection on point clouds," in *Proc. Eur. Conf. Comput. Vis. (ECCV) Workshops*, 2018, pp. 1–14.

[13] S.-L. Yu, T. Westfechtel, R. Hamada, K. Ohno, and S. Tadokoro, "Vehicle detection and localization on bird's eye view elevation images using convolutional neural network," in *Proc. IEEE Int. Symp. Saf., Secur. Rescue Robot. (SSRR)*, Oct. 2017, pp. 102–109.

[14] C. R. Qi, H. Su, K. Mo, and L. J. Guibas, "PointNet: Deep learning on point sets for 3D classification and segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 652–660.

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

[16] B. Li, "3D fully convolutional network for vehicle detection in point cloud," in *Proc. IEEE/RSJ Int. Conf. Intell. Robot. Syst. (IROS)*, Sep. 2017, pp. 1513–1518.

[17] A. H. Lang, S. Vora, H. Caesar, L. Zhou, J. Yang, and O. Beijbom, "PointPillars: Fast encoders for object detection from point clouds," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Oct. 2019, pp. 12697–12705.

[18] 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.

[19] C. R. Qi, W. Liu, C. Wu, H. Su, and L. J. Guibas, "Frustum pointnets for 3D object detection from rgb-d data," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 918–927.

[20] Z. Wang and K. Jia, "Frustum ConvNet: Sliding frustums to aggregate local point-wise features for amodal 3D object detection," in *Proc. IEEE/RSJ Int. Conf. Intell. Robot. Syst. (IROS)*, Nov. 2019, pp. 1742–1749.

[21] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 28, 2015, pp. 91–99.

[22] 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.*, 2016, pp. 779–788.

[23] W. Liu *et al.*, "SSD: Single shot multibox detector," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2016, pp. 21–37.

[24] W. Ali, S. Abdelkarim, M. Zidan, M. Zahran, and A. El Sallab, "YOLO3D: End-to-end real-time 3D oriented object bounding box detection from LiDAR point cloud," in *Proc. Eur. Conf. Comput. Vis. (ECCV) Workshops*, 2018, pp. 1–12.

[25] H. Su, S. Maji, E. Kalogerakis, and E. Learned-Miller, "Multi-view convolutional neural networks for 3D shape recognition," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 945–953.

[26] B. Yang, W. Luo, and R. Urtasun, "PIXOR: Real-time 3D object detection from point clouds," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 7652–7660.

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

[28] Y. Li, R. Bu, M. Sun, W. Wu, X. Di, and B. Chen, "PointCNN: Convolution on X-transformed points," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 31, 2018, pp. 820–830.

[29] D. Z. Wang and I. Posner, "Voting for voting in online point cloud object detection," in *Proc. Robot., Sci. Syst.*, Rome, Italy, 2015, vol. 1, no. 3, pp. 10–15.

[30] S. Song and J. Xiao, "Sliding shapes for 3D object detection in depth images," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2014, pp. 634–651.

[31] M. Engelcke, D. Rao, D. Z. Wang, C. H. Tong, and I. Posner, "Vote3Deep: Fast object detection in 3D point clouds using efficient convolutional neural networks," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2017, pp. 1355–1361.

[32] S. Song and J. Xiao, "Deep sliding shapes for amodal 3D object detection in RGB-D images," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Oct. 2016, pp. 808–816.

[33] T.-Y. Lin, P. Dollar, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 2117–2125.

[34] 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.
[35] 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.*, 2020, vol. 34, no. 7, pp. 12460–12467.
[36] 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. ECCV*. Cham, Switzerland: Springer, Aug. 2020, pp. 720–736.
[37] T. Huang, Z. Liu, X. Chen, and X. Bai, "EPNet: Enhancing point features with image semantics for 3D object detection," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2020, pp. 35–52.
[38] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 3431–3440.
[39] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 2980–2988.

Kun Zhao received the M.S. degree in vehicle engineering from the University of Science and Technology Beijing, Beijing, China, in 2016, where he is currently pursuing the Ph.D. degree in mechanical engineering. His research interests include machine learning, pattern recognition, computer vision, and intelligent vehicle.

Lingfei Ma (Member, IEEE) received the B.Sc., M.Sc., and Ph.D. degrees in geomatics engineering from the University of Waterloo, Waterloo, ON, Canada, in 2015, 2017, and 2020, respectively. He is currently an Assistant Professor of urban data science with the Central University of Finance and Economics, Beijing, China. He has published more than 30 papers in refereed journals and conferences, including IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS, IEEE TRANSACTIONS ON GEOSCIENCE AND REMOTE SENSING, *ISPRS Journal of Photogrammetry and Remote Sensing*, IEEE TRANSACTIONS ON NEURAL NETWORKS AND LEARNING SYSTEMS, and IEEE-CVPRW. His research interests include autonomous driving, mobile laser scanning, intelligent processing of point clouds, 3D scene modeling, and machine learning. He was a recipient of the 2020 National Best Ph.D. Thesis Award granted by the Canadian Remote Sensing Society. He serves as the Guest Editor for *International Journal of Applied Earth Observation and Geoinformation*.

Yu Meng received the M.S. and Ph.D. degrees in computer science and technology from Jilin University, Changchun, China, in 2007. He is currently an Associate Professor with the School of Mechanical Engineering, University of Science and Technology Beijing, Beijing, China. His research interests include computer vision and intelligent vehicle.

Li Liu received the Ph.D. degree in mechanical engineering from the University of Science and Technology Beijing, Beijing, China, in 2012. He is currently a Professor with the School of Mechanical Engineering, University of Science and Technology Beijing. His research interests focus on autonomous driving and mine intelligence.

Junbo Wang received the M.S. degree in marine geology from the Third Institute of Oceanography, State Oceanic Administration, Xiamen, China, in 2015. He is currently pursuing the Ph.D. degree in marine affairs with Xiamen University, Xiamen. His research interests include machine/deep learning, geospatial information data processing, and marine resource management.

José Marcato Junior (Member, IEEE) received the Ph.D. degree in cartographic science from São Paulo State University, Brazil. He is currently a Professor with the Faculty of Engineering, Architecture and Urbanism and Geography, Federal University of Mato Grosso do Sul, Campo Grande, Brazil. He has published more than 30 in refereed journals and more than 70 in conferences, including *ISPRS Journal of Photogrammetry and Remote Sensing* and IEEE JOURNAL OF SELECTED TOPICS IN APPLIED EARTH OBSERVATIONS AND REMOTE SENSING. His current research interests include UAV photogrammetry and deep neural networks for object detection, classification, and segmentation.

Wesley Nunes Gonçalves (Member, IEEE) received the Ph.D. degree in computational physics from the University of São Paulo, Brazil. He is currently a Professor with the Faculty of Computer Science and the Faculty of Engineering, Architecture and Urbanism and Geography, Federal University of Mato Grosso do Sul, Campo Grande, Brazil. He has published more than 30 in refereed journals and more than 60 in conferences, including *Pattern Recognition*, *Pattern Recognition Letters*, and *Neurocomputing*. His current research interests include computer vision, machine learning, deep neural networks for object detection, classification, and segmentation.

Jonathan Li (Senior Member, IEEE) received the Ph.D. degree in geomatics engineering from the University of Cape Town in 2000. He is currently a Professor with the Department of Geography and Environmental Management and cross-appointed with the Department of Systems Design Engineering, University of Waterloo, Canada. He is also a Founding Member of the Waterloo Artificial Intelligence Institute. He has coauthored more than 500 publications, over 300 of which were published in refereed journals, including IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS, *ISPRS Journal of Photogrammetry and Remote Sensing*, and *Remote Sensing of Environment*. His research interests include artificial intelligence (AI) techniques for information extraction from LiDAR point clouds and earth observation images and their applications in geospatial mapping, transportation, and urban digital twins. He was a recipient of the 2021 CIG Geomatica Award, the 2020 ISPRS Samuel Gamble Award, and the 2019 Outstanding Achievement Award in Mobile Mapping Technology. He is a fellow of the Engineering Institute of Canada (FEIC), the President-Elect of Canadian Institute of Geomatics (CIG), the Editor-in-Chief of the *International Journal of Applied Earth Observation and Geoinformation*, and an Associate Editor of IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS, IEEE TRANSACTIONS ON GEOSCIENCE AND REMOTE SENSING, and *Canadian Journal of Remote Sensing*.