Influence of Camera-LiDAR Configuration on 3D Object Detection for Autonomous Driving

Ye Li[1]* Hanjiang Hu[2]* Zuxin Liu[2] Ding Zhao[2]

***Abstract*— Cameras and LiDARs are both important sensors for autonomous driving, playing critical roles for 3D object detection. Camera-LiDAR Fusion has been a prevalent solution for robust and accurate autonomous driving perception. In contrast to the vast majority of existing arts that focus on how to improve the performance of 3D target detection through cross-modal schemes, deep learning algorithms, and training tricks, we devote attention to the impact of sensor configurations on the performance of learning-based methods. To achieve this, we propose a unified information-theoretic surrogate metric for camera and LiDAR evaluation based on the proposed sensor perception model. We also design an accelerated high-quality framework for data acquisition, model training, and performance evaluation that functions with the CARLA simulator. To show the correlation between detection performance and our surrogate metrics, We conduct experiments using several camera-LiDAR placements and parameters inspired by self-driving companies and research institutions. Extensive experimental results of representative algorithms on NuScenes dataset validate the effectiveness of our surrogate metric, demonstrating that sensor configurations significantly impact point-cloud-image fusion based detection models, which contribute up to 30% discrepancy in terms of average precision.**

I. Introduction

Multi-sensor fusion plays an important role in autonomous driving perception. Existing 3D object detection algorithms based on sensor fusion mainly use cameras and LiDAR. Cameras capture the rich texture and features of 3D objects in the environment [1], [2], and LiDARs obtain the geometric characteristics of 3D objects [3]–[5]. In this paper, we study the perception system from the perspective of the physical design of multiple sensors, rather than fusion algorithms, and focus on the influence of camera-LiDAR configurations on 3D object detection performance.

Well-designed sensor configurations are critical for autonomous driving, as improper camera and LiDAR configurations can lead to poor input data quality, which can affect detection performance [6]. Previous works [7]–[9] have explored a number of novel camera-LiDAR fusion perception algorithms, achieving excellent accuracy on autonomous driving datasets, such as NuScenes [10] and Waymo [11]. However, only a few preliminary arts [12]–[15] study the perception problem from the sensor-configuration perspective, e.g. different placements or parameters of sensors. Most current research focuses on LiDAR [16], [17] and camera [18] configurations for sensing performance separately, but rarely establishes consistent criteria for both sensors. To this end, we aim to investigate the unified evaluation methods for both camera and LiDAR configurations, as shown in Fig. 1.

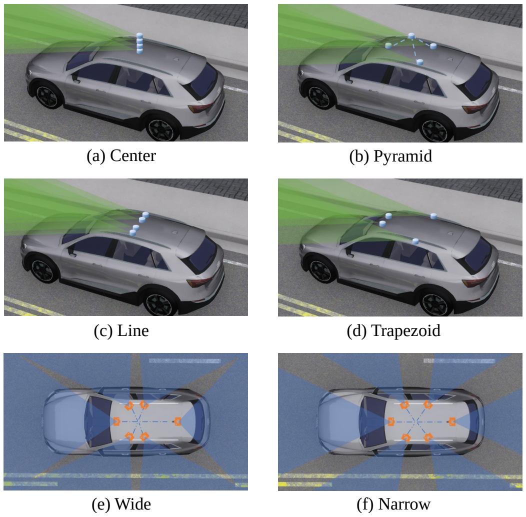

Fig. 1. LiDAR configurations of (a) Center, (b) Pyramid, (c) Line, (d) Trapezoid, and camera configurations of (e) Wide and (f) Narrow.

Fast evaluation of 3D detection performance under different camera and LiDAR configurations in the real world is quite challenging due to the laboriousness of data acquisition, model training, and performance testing [12]. Besides, efficiently comparing different sensor configurations for better 3D perception remains an open and critical question under a general trend of using more multi-modal sensors in autonomous driving [19]. To this point, this paper investigates the impact of camera-LiDAR configuration on 3D object detection performance, and proposes a novel and unified framework for accelerating the evaluation of different camera-LiDAR configurations. The main contributions of this paper are summarized as follows:

- We establish a new systematic framework to efficiently evaluate the 3D detection performance of different camera-LiDAR configurations without the effort-costly loop of data collection, model training and evaluation.
- We propose an easy-to-compute unified surrogate metric based on the sensing mechanisms of both cameras and LiDARs, effectively characterizing the sensing procedure and accelerating the evaluation of perception performance.
- Extensive experimental results in CARLA validate the correlation between our surrogate metric and the detection

*The first two authors contributed equally

[1] Ye Li is with University of Michigan, Ann Arbor, MI 48109, USA. yeyli@umich.edu. Work done while interning at Carnegie Mellon University, prior to current affiliation

[2] Hanjiang Hu, Zuxin Liu and Ding Zhao are with Carnegie Mellon University, Pittsburgh, PA 15213, USA.

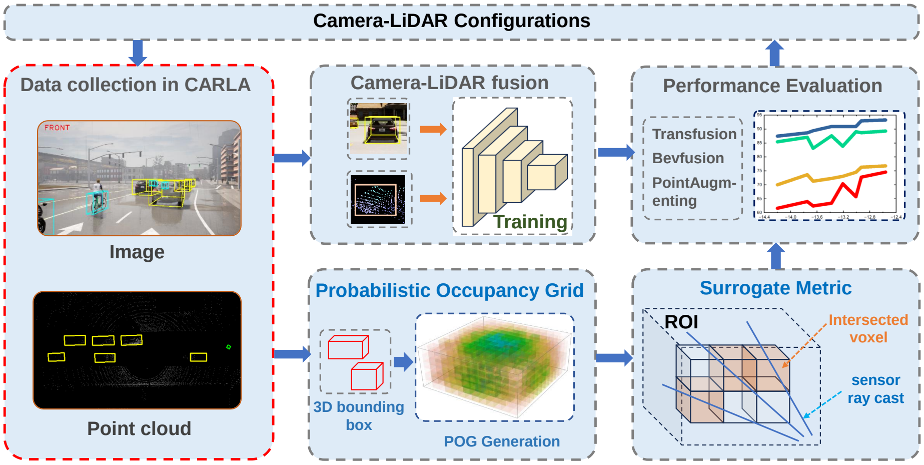

Fig. 2. Evaluation framework for camera-LiDAR configurations.

performance of several camera-LiDAR fusion algorithms. The code is available on https://github.com/safeai-lab/lidar-camera-placement.

II. Related Work

A. Multi-modal Sensor Configurations

Perception is an important sub-module of the autonomous driving system [20], which directly affects the decision-making and behavior of vehicles. With this respect, LiDARs and cameras are widely used for the perception of autonomous vehicles due to their capability to obtain rich information from the environment in real-time setting [21]–[23]. However, the performance of cameras and LiDARs is sensitive to the physical installation errors or motion perturbations [17], [24], [25]. There exist some works exploring configurations of cameras and LiDARs separately. Rahimian *et al.* [26] introduce a dynamic occlusion optimization method to estimate the configuration of the single camera.Puligandla *et al.* [15] optimize multi-camera placements for vehicle surround view in continuous domain using gradient-free black-box optimization. For LiDARs, Liu *et al.* [27] develop a non-detectable-space-based surrogate function and realize the optimum with Artificial Bee Colony algorithm. Instead of optimizing the surrounding view of sensors on the vehicle and to minimize undetectable space, recent works [12] quantitatively investigates the interaction between LiDAR placement and detection performance and proposes metrics to evaluate LiDAR perception. However, quantitative evaluation methods for both cameras and LiDARs associated with sensor fusion based 3D detection algorithms remain to be explored although [14] initiates the study of multiple sensors empirically. To this end, we present a novel unified metric for quantitatively studying both camera and LiDAR configurations in Section III.

B. Camera-LiDAR Detection Algorithms

3D object detection is an important task in autonomous driving. Recently, camera-LiDAR fusion has attracted increasing attention to boost more robust and accurate detection performance. Current camera-LiDAR fusion work can be classified into three categories: result-level, proposal-level, and point-level. The result-level methods [28]–[31] use 2D object detectors to prepare 3D detection proposals, with difficulty encountered when objects overlap in 2D planes. The proposal-level methods [32], [33] generate 3D proposals directly and perform fusion at the region-of-interest (ROI) level through ROIPool [34], but the granularity is poor. Most recent works use the point-level method and achieve promising detection performance. Point-level fusion methods established the association between 3D points and image pixels through calibration matrices and decorate LiDAR points input with image segmentation features, including input-level decoration [9], [35]–[39] and feature-level decoration. Moreover, some approaches [40]–[42] perform the decoration process on the bird's eye view (BEV) plane. The majority of the detection methods mentioned above are well-designed and evaluated on high-quality point cloud datasets, but they do not consider the influence of the placements and parameters of the camera and LiDAR sensing system. We adopt several representative latest released methods to evaluate the influence of camera-LiDAR configurations on detection performance in Section IV.

III. Unified Surrogate Metric

We propose a new surrogate metric derived from maximal information gain, leveraging camera and LiDAR ray-casting algorithms. First, we compute the probabilistic occupancy grid and the entropy of the joint distribution for scenes equipped with bounding boxes ground truth. Then we identified the voxels intersected by rays from both cameras and LiDARs to determine the conditional entropy and the unified surrogate metric of information gain, as shown in Fig. 5.

A. Problem Formulation

To evaluate the performance of different camera-LiDAR configurations, we only consider the objects in the region of interest (ROI) when calculating the detection accuracy

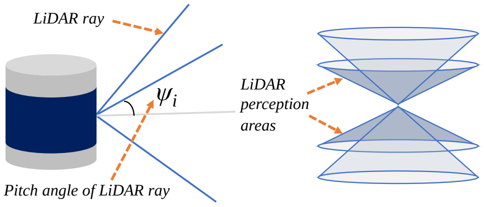

Fig. 3. LiDAR sensing model

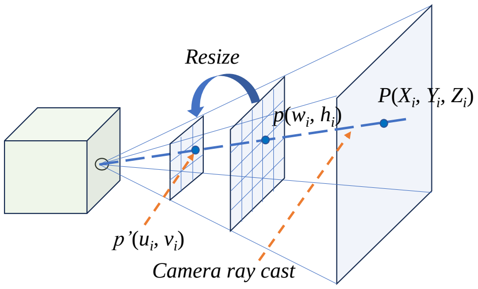

Fig. 4. Camera sensing model

metrics. We formulate the problem of camera-LiDAR configuration evaluation as comparing the 3D object detection performance with several state-of-the-art camera-LiDAR fusion methods and their corresponding surrogate metrics. Given the difficulty of evaluating the performance of camera-LiDAR detection in the real world, we propose a unified surrogate metric to accelerate the sensor configuration evaluation procedure. We introduce the camera-LiDAR perception model to calculate the unified surrogate metric.

In our work, ROI refers to the limited 3D cuboid area $[L,W,H]$, where the sensor detects the target object. We divide the 3D ROI cuboid area Ω into voxels of the same resolution δ,

$$\Omega = \{\omega_1, \omega_2, ..., \omega_N\}, N = \frac{L}{\delta} \times \frac{W}{\delta} \times \frac{H}{\delta} \tag{1}$$

where N denotes the total number of divided voxels $\omega_i (i = 1, 2, ..., N)$ in the ROI.

B. Modeling Camera-LiDAR Perception

We first introduce the camera-LiDAR data sensing model. To evaluate the parameters of LiDAR and camera with a unified method, we propose the following ray cast method of both camera and LiDAR.

LiDAR perception model. Inspired by mechanically rotating LiDAR, we model the LiDAR perception model as a series of rotating vertically distributed rays. These rays are evenly distributed at certain angular intervals in the vertical direction, and they rotate synchronously around the vertical Z axis at a fixed angular velocity. Each ray has a fixed pitch angle to the horizontal plane and forms a conical surface by rotation. The sensing area of LiDAR is the collection in the ROI of all conical surfaces. For a LiDAR of $horizontal = \theta_0^L, vertical = \psi_0^L$, we calculate the yaw θ_{ij}^L and the pitch ψ_{ij}^L rotation angles for these rays as,

$$\theta_{ij}^L = \frac{\theta_0^L}{i}, \psi_{ij}^L = \frac{\psi_0^L}{j} + \frac{\psi_0^L}{2}, \quad i \in \{1, ..., I\}, j \in \{1, ..., J\} \tag{2}$$

where I and J represent the horizontal and vertical resolution for the specific LiDAR.

Camera perception model. Based on the pin-hole image camera model with image projection [43], as shown in Fig. 4, we can obtain the transformation matrix between the coordinates of each pixel and the points in the 3D world. Each pixel point $p(w_i, h_i)$ and the corresponding 3D world point $P(X_i, Y_i, Z_i)$ form a ray with its origin at the camera's optical center O. The RGB color channel of each pixel represents the exterior features of the object encountered by the ray. Thus, the total perception of a camera is the union of all ROI space traversed by the rays. Therefore, we see the fusion sensing model of LiDAR and camera as two sets of rays scanned in the ROI.

In our camera perception model, the number of rays does not exactly equal to the original pixel resolution $w_0 \times h_0$. The images obtained by the camera are dense and with high-resolution, but learning-based methods do not directly take the original images as inputs without down-sampling. So the original images are resized to a lower resolution $u_0 \times v_0$ to model the process of the camera ray casting. We first map the pixels on the resized image to the pixels on the original image as follows,

$$w_i = u_i \times \frac{w_0}{u_0}, h_i = v_i \times \frac{h_0}{v_0}, \quad i = 1, 2, ..., u_0 \times v_0 \tag{3}$$

Given pixel focal length of the camera f, we calculate the yaw θ_i^c and pitch ψ_i^c rotation angles for each ray $i = 1, 2, ..., u_0 \times v_0$,

$$\theta_i^c = acrcos \frac{w_i - \frac{w_0}{2}}{\sqrt{(w_i - \frac{w_0}{2})^2 + f^2}} \tag{4}$$

$$\psi_i^c = acrcos \frac{h_i - \frac{h_0}{2}}{\sqrt{(w_i - \frac{w_0}{2})^2 + (h_i - \frac{h_0}{2})^2 + f^2}} \tag{5}$$

C. Multi-sensor Probabilistic Occupancy Grid

Following [12], it is assumed that the more 3D objects are covered by the rays of our camera-LiDAR seeing model, the better the performance of multi-sensor 3D detection will be. Based on this intuition, we first propose Probabilistic Occupancy Grid (POG) to evaluate the probability that each voxel in ROI is occupied by the target object to be detected.

$$p_{POG} = p\{\omega_1, \omega_2, ..., \omega_N\} \tag{6}$$

where $\omega_i \sim p_\Omega$. Given a dataset with ground true of samples $Y_T = \{y_1, y_2, ..., y_T\}$, T denotes the total frames of ground truth samples. Each frame $y_t \in \{y_1, ..., y_T\}$ of the given samples contains $S^{(t)}$ ground-truth bounding boxes. We denote the bounding boxes of target objects as $\{b_1^{(t)}, b_2^{(t)}, ..., b_{S^{(t)}}^{(t)}\}$. We use $\omega_i \in y_t$ to denote the case where the voxel ω_i is contained by the ground truth bounding box of any target

TABLE I

UNIFIED SURROGATE METRIC FOR MULTI-SENSOR CONFIGURATION

Sensor Configurations	Camera S-MIG (10^3)			LiDAR S-MIG (10^3)			$\lambda = 0.1$, S-MS (10^3)		
	Car	Bicycle	Pedestrian	Car	Bicycle	Pedestrian	Car	Bicycle	Pedestrian
Wide + Center (W+C)	-73.08	-14.15	-13.13	-6.45	-1.07	-1.08	-13.75	-2.48	-2.39
Wide + Pyramid (W+P)	-73.08	-14.15	-13.13	-5.90	-0.96	-0.92	-13.21	-2.38	-2.23
Wide + Line (W+L)	-73.08	-14.15	-13.13	-5.62	-0.91	-0.88	-12.93	-2.32	-2.19
Wide + Trapezoid (W+T)	-73.08	-14.15	-13.13	-5.25	-0.87	-0.83	-12.56	-2.29	-2.14
Narrow + Center (N+C)	-77.56	-15.57	-14.09	-6.45	-1.07	-1.08	-14.20	-2.62	-2.49
Narrow + Pyramid (N+P)	-77.56	-15.57	-14.09	-5.90	-0.96	-0.92	-13.66	-2.52	-2.33
Narrow + Line (N+L)	-77.56	-15.57	-14.09	-5.62	-0.91	-0.88	-13.38	-2.46	-2.29
Narrow + Trapezoid (N+T)	-77.56	-15.57	-14.09	-5.25	-0.87	-0.83	-13.01	-2.43	-2.24

object in the frame y_t. Then, for each voxel ω_i in the ROI, we traverse all the T frames of samples to obtain the probability that the voxel ω_i is occupied by the target objects in any given frame from Y_T as,

$$\hat{p}(\omega_i) = \frac{\sum_{t=1}^{T} \mathbb{1}(\omega_i \in y_t)}{T}, \ \omega_i \in y_t := \exists\, b^{(t)} \in y_t, s.t.\ \omega_i \in b^{(t)}$$

where $\mathbb{1}(\cdot)$ is an indicator function and N is the number of voxels in ROI. The POG can then be estimated by the joint probability of all occupied voxels in ROI. Since the presence of an object in one voxel does not imply presence in other voxels among all the frames in ROI, we could treat these voxels as independent and identically distributed random variables and calculate the joint distribution over all voxels in the set Ω as,

$$\hat{p}_{POG} = \hat{p}(\omega_1, \omega_2, ..., \omega_N) = \prod_{i=1,\hat{p}(\omega_i)\neq 0}^{N} \hat{p}(\omega_i) \qquad (7)$$

where N is the total number of non-zero voxels in ROI. Note that notations of $\hat{p}$ with ^ are the estimated distribution from observed samples, while notations of p without *hat* are the unknown non-random statistics to be estimated.

Considering the ray casting sensing model for a specific multi-sensor configuration, we define the conditional Probabilistic Occupancy Grid to represent the conditional probability that a voxel is occupied by the 3D bounding boxes of the target objects in the perceptual field of the sensor, with the assumption of conditional independence.

$$p_{POG|C=C_0} = p\{\omega_1, \omega_2, ..., \omega_N \mid C = C_0\} \qquad (8)$$

To make the notation compact and easy to read, we denote the occupied voxel random variable $\omega_i | C = C_0$ as $\omega_i^{C_0}$ and denote the conditional distribution as $p_{\Omega|C=C_0}$ as $p_{\Omega|C_0}$, so we have $\omega_i \sim p_{\Omega|C_0}$.

We use Bresenham's Line Algorithm [44] to deal with our proposed sensor ray casting sensing model, finding all voxels traversed by the ray generated from the given camera or LiDAR sensor. For the specific sensor configuration $C = C_0$, we denote the set of voxels traversed by rays as,

$$\Omega | C_0 = Brsenham(\Omega, C_0) = \{\omega_1^{C_0}, \omega_2^{C_0}, ..., \omega_N^{C_0}\} \qquad (9)$$

Given the specific sensor configuration $C = C_0$, the conditional Probabilistic Occupancy Grid can be expressed as,

$$\hat{p}_{POG|C_0} = \hat{p}(\omega_1^{C_0}, \omega_2^{C_0}, ..., \omega_N^{C_0}) = \prod_{i=1,\hat{p}(\omega_i^{C_0})\neq 0}^{N} \hat{p}(\omega_i^{C_0}) \qquad (10)$$

The conditional Probabilistic Occupancy Grid reflects the conditional joint distribution of voxels that intersect with the perceptual area of the sensor in the ROI. From the perspective of density estimation to find POG, the true POG and conditional POG given configuration C_0 can be estimated as

$$p_{POG} = \hat{p}_{POG}, p_{POG|C_0} = \hat{p}_{POG|C_0} \qquad (11)$$

D. Unified Surrogate Metric for Multi-sensor Fusion

In this section, we propose a unified surrogate metric for evaluating camera-LiDAR configurations, based on POG and information theory. We first define the surrogate metric for a single type of sensor. Given specific sensor configurations, We denote the total uncertainty in the joint voxel distribution by the total entropy of POG H_{POG}. Further, we can represent the conditional uncertainty with the entropy of the conditional POG given sensor configurations $C = C_0$.

$$H_{POG} = H(\Omega) = \mathbb{E}_{\omega_i \sim p_\Omega} \sum_{i=1}^{N} \hat{H}(\omega_i) \qquad (12)$$

$$H_{POG|C_0} = H(\Omega|C_0) = \mathbb{E}_{\omega_i \sim p_\Omega} \sum_{i=1}^{N} \hat{H}(\omega_i^{C_0}) \qquad (13)$$

Perception performance of a given configuration is maximized when the uncertainty of the voxel distribution to the sensor is minimized. We denote the information gain IG using the mutual information (**MI**) between the full entropy and the conditional entropy, which represents the decrease of the uncertainty of the voxel distribution given a specific sensor configuration.

$$IG_{\Omega,C_0} = H(\Omega) - H(\Omega|C_0) \qquad (14)$$

Since the total entropy $H(\Omega)$ is a constant given POG with a fixed voxel distribution and irrelevant to different sensor configurations, we denote our maximum-information-gain-based surrogate metric S-MIG as,

$$S_{MIG|C_0} = -H(\Omega|C_0) = -\mathbb{E}_{\omega_i \sim p_\Omega} \sum_{i=1}^{N} \hat{H}(\omega_i^{C_0}) \qquad (15)$$

Further, we introduce the Unified Surrogate Metric for Multi-sensor Fusion to evaluate the multi-sensor configuration. For camera configuration $C^c(\theta^c, \psi^c)$ and LiDAR configuration $C^L(\theta^L, \psi^L)$, we denote Unified Surrogate Metric for Multi-sensor (**S-MS**) as,

$$S_{MS} = S_{MIG|C^c(\theta^c,\psi^c)} + \lambda S_{MIG|C^L(\theta^L,\psi^L)} \tag{16}$$

where λ is a hyper-parameter weighting the difference in perceptual capbility between LiDAR and the camera, to fairly show the influence of cameras and LiDARs on 3D detection.

IV. Experiments

In this section, we design a framework for automatic camera-LiDAR data collection and evaluation based on the CARLA simulator [45] to avoid time-consuming, low-efficiency and high-cost real-world experiments to validate our method. To ensure the fairness of the experimental comparison, all images and point clouds collected in CARLA are with fixed scenarios, including the route of the data-collection ego vehicle, driving scenarios, traffic flow, etc. We conducted comprehensive experiments to show two key points: the impact of camera and LiDAR configurations on 3D object detection performance, and the correlation between our proposed unified surrogate metric and learning-based perception performance.

A. *Experimental Setup*

We collect data in the CARLA simulator and calculate the S-MS for each camera-LiDAR configuration based on the bounding box of the 3D target objects.

CARLA simulator and dataset. In order to verify our proposed method, we design an automatic camera-LiDAR data collection pipeline based on the realistic CARLA simulator. Following [12], we collect data in Town 1, 3, 4, and 6 and each town contains 8 manually recorded routes. Only camera and LiDAR configurations are changed for all the experiments. We collected 8 Nuscenes-like datasets with different camera-LiDAR configurations, and each dataset contains about 38000 frames. Each frame of the dataset contains six images and one set of point clouds with 3D bounding boxes of Car, Bicycle, and Pedestrian. We use the CARLA simulator to collect point cloud and image data in the NuScenes format [46]. We use 72% frames as the training set, 18% frames as the validation set, and the remaining 10% frames as the test set.

Camera-LiDAR fusion detection algorithms and metrics. In order to fairly demonstrate the performance under different camera-LiDAR configurations on 3D object detection, we adopt the three representative open-source camera-LiDAR fusion detection method. Specifically, we use two two-stage detection methods, Transfusion [7] and Bevfusion [8], and one single-stage detection method, Pointaugmenting [9]. We follow the default training pipeline and hyperparameters of these methods. For two-stage methods, We first train the LiDAR-only detection model, then fine-tune the fusion methods with pretrained image detection model. For the one-stage method, we train two backbones, VoxelNet [47], and Pointpillars [48], respectively. We adopt the mean Average Precision (mAP) metric from NuScenes [46], which defines thresholds by considering the 2D distance on the ground rather than Intersection over Union (IoU) based ones [21]. For mAP, The 2D distances are at 4 different thresholds: 0.5, 1.0, 2.0, and 4.0, respectively.

Different sensor configurations. The sensor configurations in this work adopt 4 LiDARs and 6 cameras following the NuScenes dataset [49]. Four different LiDAR configurations inspired by famous autonomous driving companies [12] and two camera configurations are used to explore the influence on object detection performance. For LiDAR configurations, the beams are be equally distributed in the vertical FOV [-25.0, 25.0] degrees, as shown in Fig. 1. Center placement is achieved by vertically stacking four LiDARs together at the roof center (1a). Pyramid placement (Fig. 1b) includes 1 front LiDAR and 3 back ones, with a higher one in the middle. Line has (1c) 4 LiDARs placed in a horizontal line symmetrically. Trapezoid installs 4 LiDARs in the parallel front and back (Fig. 1d). Each camera has the same resolution 1600×900. Wide configuration is the same as the setup of NuScenes with 5 general cameras (FOV = 70°) and a wide-angle camera (FOV = 110°) on the roof of the ego car (Fig. 1e). As contrast to the wide one, we design another Narrow configuration by changing the FOV to 55 degrees for all cameras in Fig. 1f.

B. *Experiment Results and Analysis*

We first show the influence of camera-LiDAR configurations on 3D object detection average precision, and then analyze the relation between the detection performance and our proposed unified surrogate metric.

Influence of sensor configurations on 3D object detection. In Fig. 5 and Table II, we show the 3D target detection performance with different representative algorithms using different camera-LiDAR configurations. The configurations of sensors drastically influence the detection performance, with a maximum fluctuation of 30%. Specifically, for the effects of the camera and LiDAR configurations respectively, it is not difficult to find that for the same LiDAR configuration, the Wide camera configurations significantly outperform the Narrow camera configurations; and for the same camera configuration, Trapezoid has the best detection performance for the majority of algorithms and objects. Under the Line LiDAR configuration, Pointaugmenting is the best for Bicycle while Bevfusion is the best for the Pedestrian, due to the point cloud detection characteristics of specific algorithms and the influence of POG for different objects.

Correlation between the unified surrogate metric and detection performance. Based on the proposed unified surrogate metric, we now analyze the interplay between 3D detection performance and camera-LiDAR configurations. Fig. 5 shows the detection performance of Car, Bicycle, and Pedestrian at different S-MS values. While there are some fluctuations, the rising tendency for detection performance with increasing S-MS values is clear to see. The full config-

TABLE II
COMPARISON OF OBJECT DETECTION PERFORMANCE UNDER VARIOUS CAMERA-LIDAR CONFIGURATIONS USING DIFFERENT METHODS

Car mAP		Wide				Narrow			
Model	Backbone	Center	Pyramid	Line	Trapezoid	Center	Pyramid	Line	Trapezoid
Transfusion	PointPillars	87.08	83.87	89.08	**89.27**	85.44	83.1	87.62	**88.74**
Bevfusion	VoxelNet	88.65	90.93	92.95	**93.29**	87.49	89.42	**90.96**	90.91
PointAugmenting	PointPillars	73.68	73.07	76.35	**76.81**	70.04	71.36	72.35	**74.56**
PointAugmenting	VoxelNet	64.06	70.37	72.78	**74.62**	61.62	62.48	63.48	**67.8**

Bicycle mAP		Wide				Narrow			
Model	Backbone	Center	Pyramid	Line	Trapezoid	Center	Pyramid	Line	Trapezoid
Transfusion	PointPillars	78.45	80.63	83.98	**86.21**	72.79	77.87	79.04	**83.38**
Bevfusion	VoxelNet	83.62	89.55	90.53	**90.78**	79.95	83.89	86.22	**87.22**
PointAugmenting	PointPillars	64.88	67.17	**69.79**	68.98	53.97	61.66	**65.26**	63.04
PointAugmenting	VoxelNet	56.38	62.18	**65.35**	63.94	51.59	55.09	**57.46**	56.53

Pedestrian mAP		Wide				Narrow			
Model	Backbone	Center	Pyramid	Line	Trapezoid	Center	Pyramid	Line	Trapezoid
Transfusion	PointPillars	45.77	57.51	66.55	**68.06**	39.38	53.39	60.74	**65.97**
Bevfusion	VoxelNet	57.37	65.96	**72.22**	67.89	55.39	58.65	**62.36**	62.21
PointAugmenting	PointPillars	35.71	37.71	39.96	**43.77**	25.1	30.38	33.7	**38.14**
PointAugmenting	VoxelNet	29.67	30.48	31.27	**32.36**	25.63	27.92	26.81	**29.63**

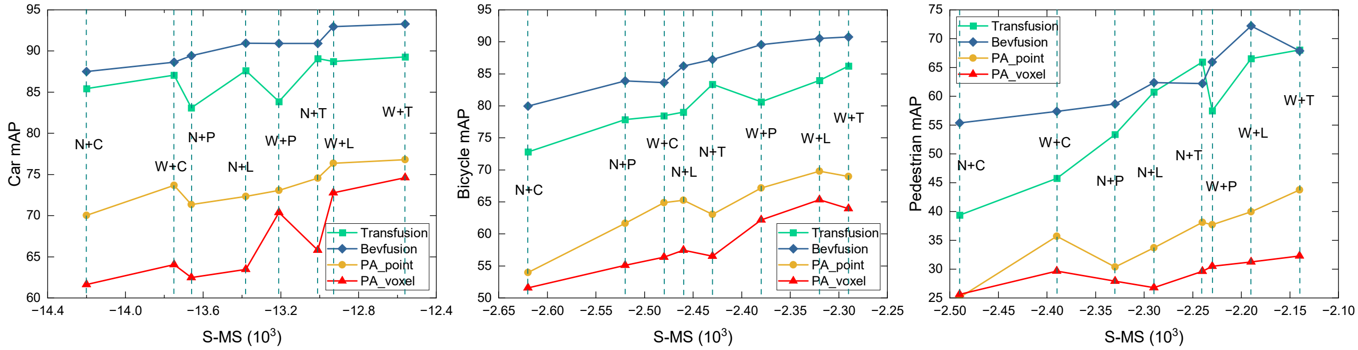

Fig. 5. The relationship between 3D detection mAP and unified surrogate metric (S-MS) under camera-LiDAR configurations, abbr. listed in Table I

urations with abbreviations can be found in Table I.

In addition to errors in data acquisition and stochastic model training, the fluctuations in the figure may be caused by two following factors. First, the same λ value is adopted for all algorithms and target objects when calculating the S-MS values, but the real weight constant λ may not be the same due to the differences in the characteristics of algorithms and target objects. Second, there exist some specific sensor configurations as preferences or attacks for specific learning-based algorithms [50], [51]. For instance, Line LiDAR configuration (L) has excellent detection performance for small objects (Bicycle and Pedestrian), as these objects are small in lateral dimension and Line configuration increases the point cloud density in the lateral direction. Under Pyramid LiDAR configuration (P), the performance of Transfusion to detect Car is drastically degraded due to the adversarial attack of Pyramid for Transfusion.

In addition, while the Wide camera configuration generally outperforms the Narrow camera configuration, detection performance with the Narrow + Trapezoid (N+T) configuration is sometimes better than with the Wide + other configurations, indicating that superior LiDAR sensor can compensate for camera deficiencies, and vice versa.

Potential application analysis. Our unified surrogate metric greatly accelerates the development, optimization, and evaluation of multi-sensor configurations for self-driving cars compared to the commonly-seen R&D process involving installing sensors, collecting data, training models, and evaluating performance. Our approach can efficiently evaluate different camera-LiDAR configurations by simply calculating the surrogate metric from the bounding boxes. Besides, the optimal solution of camera-LiDAR configurations for specific scenarios can be found based on our metrics.

V. CONCLUSION

In this paper, we investigate the influence of LiDAR and camera configurations on the performance of 3D object detection for autonomous driving. We propose a novel framework for evaluating LiDAR and camera configurations, including data acquisition, model training, and performance evaluation. We propose a unified surrogate metric that predicts 3D object detection performance for different camera and LiDAR configurations. We conduct extensive experiments with CARLA-collected data and representative camera-LiDAR fusion algorithms, and the results have shown high consistency between our metrics and detection performance, providing new directions for the optimization of multi-sensor configuration in self-driving cars.

References

[1] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask r-cnn," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2961–2969.

[2] J. Redmon and A. Farhadi, "Yolo9000: Better, faster, stronger," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 6517–6525.

[3] Y. Wang and J. M. Solomon, "Object dgcnn: 3d object detection using dynamic graphs," *Advances in Neural Information Processing Systems*, vol. 34, pp. 20 745–20 758, 2021.

[4] Z. Liu, S. Zhou, C. Suo, P. Yin, W. Chen, H. Wang, H. Li, and Y. Liu, "Lpd-net: 3d point cloud learning for large-scale place recognition and environment analysis," in *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019, pp. 2831–2840.

[5] Z. Qiao, H. Hu, W. Shi, S. Chen, Z. Liu, and H. Wang, "A registration-aided domain adaptation network for 3d point cloud based place recognition," in *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2021, pp. 1317–1322.

[6] Q. Xu, Y. Zhou, W. Wang, C. R. Qi, and D. Anguelov, "Spg: Unsupervised domain adaptation for 3d object detection via semantic point generation," in *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021, pp. 15 426–15 436.

[7] X. Bai, Z. Hu, X. Zhu, Q. Huang, Y. Chen, H. Fu, and C.-L. Tai, "Transfusion: Robust lidar-camera fusion for 3d object detection with transformers," in *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022, pp. 1080–1089.

[8] Z. Liu, H. Tang, A. Amini, X. Yang, H. Mao, D. Rus, and S. Han, "Bevfusion: Multi-task multi-sensor fusion with unified bird's-eye view representation," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2023.

[9] C. Wang, C. Ma, M. Zhu, and X. Yang, "Pointaugmenting: Cross-modal augmentation for 3d object detection," in *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021, pp. 11 789–11 798.

[10] H. Caesar, V. Bankiti, A. H. Lang, S. Vora, V. E. Liong, Q. Xu, A. Krishnan, Y. Pan, G. Baldan, and O. Beijbom, "nuscenes: A multimodal dataset for autonomous driving," in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020, pp. 11 618–11 628.

[11] P. Sun, H. Kretzschmar, X. Dotiwalla, A. Chouard, V. Patnaik, P. Tsui, J. Guo, Y. Zhou, Y. Chai, B. Caine, V. Vasudevan, W. Han, J. Ngiam, H. Zhao, A. Timofeev, S. Ettinger, M. Krivokon, A. Gao, A. Joshi, Y. Zhang, J. Shlens, Z. Chen, and D. Anguelov, "Scalability in perception for autonomous driving: Waymo open dataset," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

[12] H. Hu, Z. Liu, S. Chitlangia, A. Agnihotri, and D. Zhao, "Investigating the impact of multi-lidar placement on object detection for autonomous driving," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 2550–2559.

[13] S. Mou, Y. Chang, W. Wang, and D. Zhao, "An optimal lidar configuration approach for self-driving cars," *arXiv preprint arXiv:1805.07843*, 2021.

[14] T. Ma, Z. Liu, and Y. Li, "Perception entropy: A metric for multiple sensors configuration evaluation and design," *arXiv preprint arXiv:2104.06615*, 2021.

[15] V. A. Puligandla and S. Lončarić, "A continuous camera placement optimization model for surround view," *IEEE Transactions on Intelligent Vehicles*, pp. 1–11, 2023.

[16] Z. Liu, M. Arief, and D. Zhao, "Where should we place lidars on the autonomous vehicle? - an optimal design approach," in *2019 International Conference on Robotics and Automation (ICRA)*, 2019, pp. 2793–2799.

[17] H. Zhang, "Two-dimensional optimal sensor placement," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 25, no. 5, pp. 781–792, 1995.

[18] V. A. Puligandla and S. Lončarić, "A multiresolution approach for large real-world camera placement optimization problems," *IEEE Access*, vol. 10, pp. 61 601–61 616, 2022.

[19] Z. Wang, J. Zhan, C. Duan, X. Guan, P. Lu, and K. Yang, "A review of vehicle detection techniques for intelligent vehicles," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 34, no. 8, pp. 3811–3831, 2023.

[20] M.-F. Chang, J. Lambert, P. Sangkloy, J. Singh, S. Bak, A. Hartnett, D. Wang, P. Carr, S. Lucey, D. Ramanan, and J. Hays, "Argoverse: 3d tracking and forecasting with rich maps," in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019, pp. 8740–8749.

[21] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? the kitti vision benchmark suite," in *2012 IEEE Conference on Computer Vision and Pattern Recognition*, 2012, pp. 3354–3361.

[22] J. Zhang and S. Singh, "Loam: Lidar odometry and mapping in real-time," *Robotics: Science and Systems Conference (RSS)*, pp. 109–111, 01 2014.

[23] V. Ravi Kumar, S. Yogamani, H. Rashed, G. Sitsu, C. Witt, I. Leang, S. Milz, and P. Mäder, "Omnidet: Surround view cameras based multi-task visual perception network for autonomous driving," *IEEE Robotics and Automation Letters*, vol. 6, no. 2, pp. 2830–2837, 2021.

[24] H. Hu, Z. Liu, L. Li, J. Zhu, and D. Zhao, "Robustness certification of visual perception models via camera motion smoothing," in *Proceedings of The 6th Conference on Robot Learning*, ser. Proceedings of Machine Learning Research, vol. 205. PMLR, 14–18 Dec 2023, pp. 1309–1320.

[25] H. Hu, C. Liu, and D. Zhao, "Robustness verification for perception models against camera motion perturbations," in *International conference on machine learning (ICML) Workshop on Formal Verification of Machine Learning (WFVML)*, 2023.

[26] P. Rahimian and J. K. Kearney, "Optimal camera placement for motion capture systems," *IEEE Transactions on Visualization and Computer Graphics*, vol. 23, no. 3, pp. 1209–1221, 2017.

[27] Z. Liu, M. Arief, and D. Zhao, "Where should we place lidars on the autonomous vehicle? - an optimal design approach," in *2019 International Conference on Robotics and Automation (ICRA)*, 2019, pp. 2793–2799.

[28] C. R. Qi, W. Liu, C. Wu, H. Su, and L. J. Guibas, "Frustum pointnets for 3d object detection from rgb-d data," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018, pp. 918–927.

[29] Z. Wang and K. Jia, "Frustum convnet: Sliding frustums to aggregate local point-wise features for amodal 3d object detection," in *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2019, pp. 1742–1749.

[30] K. Shin, Y. P. Kwon, and M. Tomizuka, "Roarnet: A robust 3d object detection based on region approximation refinement," in *2019 IEEE Intelligent Vehicles Symposium (IV)*, 2019, pp. 2510–2515.

[31] R. Q. Charles, H. Su, M. Kaichun, and L. J. Guibas, "Pointnet: Deep learning on point sets for 3d classification and segmentation," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 77–85.

[32] J. Ku, M. Mozifian, J. Lee, A. Harakeh, and S. L. Waslander, "Joint 3d proposal generation and object detection from view aggregation," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2018, pp. 1–8.

[33] X. Chen, H. Ma, J. Wan, B. Li, and T. Xia, "Multi-view 3d object detection network for autonomous driving," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 6526–6534.

[34] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 6, pp. 1137–1149, 2017.

[35] S. Vora, A. H. Lang, B. Helou, and O. Beijbom, "Pointpainting: Sequential fusion for 3d object detection," in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020, pp. 4603–4611.

[36] T. Yin, X. Zhou, and P. Krähenbühl, "Multimodal virtual point 3d detection," in *Advances in Neural Information Processing Systems (NeurIPS 2021)*, vol. 34, 2021, pp. 16 494–16 507.

[37] S. Xu, D. Zhou, J. Fang, J. Yin, Z. Bin, and L. Zhang, "Fusionpainting: Multimodal fusion with adaptive attention for 3d object detection," in *2021 IEEE International Intelligent Transportation Systems Conference (ITSC)*, 2021, pp. 3047–3054.

[38] Z. Chen, Z. Li, S. Zhang, L. Fang, Q. Jiang, F. Zhao, B. Zhou, and H. Zhao, "Autoalign: Pixel-instance feature aggregation for multi-modal 3d object detection," in *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence, IJCAI-22*, 7 2022, pp. 827–833.

[39] Y. Chen, Y. Li, X. Zhang, J. Sun, and J. Jia, "Focal sparse convolutional networks for 3d object detection," in *2022 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

[40] M. Liang, B. Yang, Y. Chen, R. Hu, and R. Urtasun, "Multi-task multi-sensor fusion for 3d object detection," in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019, pp. 7337–7345.

[41] L. Xie, C. Xiang, Z. Yu, G. Xu, Z. Yang, D. Cai, and X. He, "Pi-rcnn: An efficient multi-sensor 3d object detector with point-based attentive cont-conv fusion module," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, no. 07, 2020, pp. 12 460–12 467.

[42] 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 *Computer Vision – ECCV 2020*, 2020, pp. 720–736.

[43] R. Hartley and A. Zisserman, *Multiple view geometry in computer vision*. Cambridge university press, 2003.

[44] J. E. Bresenham, "Algorithm for computer control of a digital plotter," *IBM Systems Journal*, vol. 4, no. 1, pp. 25–30, 1965.

[45] A. Dosovitskiy, G. Ros, F. Codevilla, A. Lopez, and V. Koltun, "Carla: An open urban driving simulator," in *Conference on robot learning*. PMLR, 2017, pp. 1–16.

[46] H. Caesar, V. Bankiti, A. H. Lang, S. Vora, V. E. Liong, Q. Xu, A. Krishnan, Y. Pan, G. Baldan, and O. Beijbom, "nuscenes: A multimodal dataset for autonomous driving," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 11 621–11 631.

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

[48] A. H. Lang, S. Vora, H. Caesar, L. Zhou, J. Yang, and O. Beijbom, "Pointpillars: Fast encoders for object detection from point clouds," in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019, pp. 12 689–12 697.

[49] H. Caesar, V. Bankiti, A. H. Lang, S. Vora, V. E. Liong, Q. Xu, A. Krishnan, Y. Pan, G. Baldan, and O. Beijbom, "nuscenes: A multimodal dataset for autonomous driving," *arXiv preprint arXiv:1903.11027*, 2019.

[50] C. Kanbak, S.-M. Moosavi-Dezfooli, and P. Frossard, "Geometric robustness of deep networks: analysis and improvement," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 4441–4449.

[51] L. Engstrom, B. Tran, D. Tsipras, L. Schmidt, and A. Madry, "Exploring the landscape of spatial robustness," in *International conference on machine learning*. PMLR, 2019, pp. 1802–1811.