Deep Multi-Modal Object Detection and Semantic Segmentation for Autonomous Driving: Datasets, Methods, and Challenges

Di Feng, *Member, IEEE*, Christian Haase-Schütz, *Member, IEEE*, Lars Rosenbaum, Heinz Hertlein, *Member, IEEE*, Claudius Gläser, Fabian Timm, Werner Wiesbeck, *Life Fellow, IEEE*, and Klaus Dietmayer, *Member, IEEE*

***Abstract*— Recent advancements in perception for autonomous driving are driven by deep learning. In order to achieve robust and accurate scene understanding, autonomous vehicles are usually equipped with different sensors (e.g. cameras, LiDARs, Radars), and multiple sensing modalities can be fused to exploit their complementary properties. In this context, many methods have been proposed for deep multi-modal perception problems. However, there is no general guideline for network architecture design, and questions of "what to fuse", "when to fuse", and "how to fuse" remain open. This review paper attempts to systematically summarize methodologies and discuss challenges for deep multi-modal object detection and semantic segmentation in autonomous driving. To this end, we first provide an overview of on-board sensors on test vehicles, open datasets, and background information for object detection and semantic segmentation in autonomous driving research. We then summarize the fusion methodologies and discuss challenges and open questions. In the appendix, we provide tables that summarize topics and methods. We also provide an interactive online platform to navigate each reference: https://boschresearch.github.io/multimodalperception/.**

***Index Terms*— Multi-modality, object detection, semantic segmentation, deep learning, autonomous driving.**

I. Introduction

SIGNIFICANT progress has been made in autonomous driving since the first successful demonstration in the 1980s [1] and the DARPA Urban Challenge in 2007 [2]. It offers high potential to decrease traffic congestion, improve road safety, and reduce carbon emissions [3]. However, developing reliable autonomous driving is still a very challenging task. This is because driverless cars are intelligent agents that need to perceive, predict, decide, plan, and execute their decisions in the real world, often in uncontrolled or complex environments, such as the urban areas shown in Fig. 1. A small error in the system can cause fatal accidents.

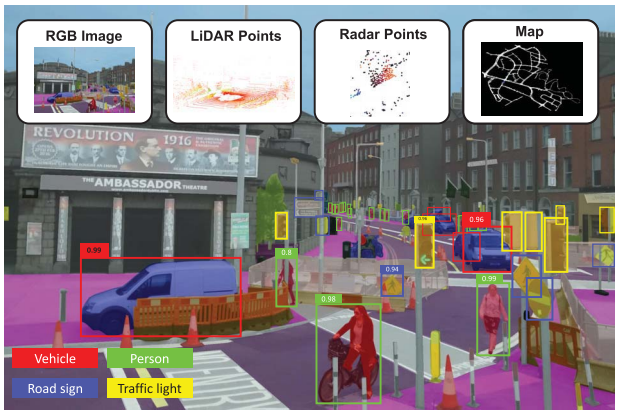


Fig. 1. A complex urban scenario for autonomous driving. The driverless car uses multi-modal signals for perception, such as RGB camera images, LiDAR points, Radar points, and map information. It needs to perceive all relevant traffic participants and objects accurately, robustly, and in real-time. For clarity, only the bounding boxes and classification scores for some objects are drawn in the image. The RGB image is adapted from [4].

Perception systems in driverless cars need to be (1). *accurate*: they need to give precise information of driving environments; (2). *robust*: they should work properly in adverse weather, in situations that are not covered during training (open-set conditions), and when some sensors are degraded or even defective; and (3). *real-time*: especially when the cars are driving at high speed. Towards these goals, autonomous cars are usually equipped with multi-modal sensors (e.g. cameras, LiDARs, Radars), and different sensing modalities are fused so that their complementary properties are exploited (cf. Sec. II-A). Furthermore, deep learning has been very successful in computer vision. A deep neural network is a powerful tool for learning hierarchical feature representations given a large amount of data [5]. In this regard, many methods have been proposed that employ deep learning to fuse

Manuscript received April 25, 2019; revised November 15, 2019; accepted January 24, 2020. Date of publication February 17, 2020; date of current version March 1, 2021. This work was supported by Robert Bosch GmbH. The Associate Editor for this article was H. G. Jung. *Di Feng and Christian Haase-Schütz contributed equally to this work. (Corresponding author: Di Feng.)*

Di Feng is with the Driver Assistance Systems and Automated Driving, Corporate Research, Robert Bosch GmbH, 71272 Renningen, Germany, and also with the Institute of Measurement, Control and Microtechnology, Ulm University, 89081 Ulm, Germany (e-mail: di.feng@de.bosch.com).

Christian Haase-Schütz is with the Engineering Cognitive Systems, Automated Driving, Chassis Systems Control, Robert Bosch GmbH, 74232 Abstatt, Germany, and also with Institute of Radio Frequency Engineering and Electronics, Karlsruhe Institute of Technology, 76131 Karlsruhe, Germany.

Lars Rosenbaum, Claudius Gläser, and Fabian Timm are with the Driver Assistance Systems and Automated Driving, Corporate Research, Robert Bosch GmbH, 71272 Renningen, Germany.

Heinz Hertlein is with the Engineering Cognitive Systems, Automated Driving, Chassis Systems Control, Robert Bosch GmbH, 74232 Abstatt, Germany.

Werner Wiesbeck is with the Institute of Radio Frequency Engineering and Electronics, Karlsruhe Institute of Technology, 76131 Karlsruhe, Germany.

Klaus Dietmayer is with the Institute of Measurement, Control and Microtechnology, Ulm University, 89081 Ulm, Germany.

This article has supplementary downloadable material available at https://ieeexplore.ieee.org, provided by the authors.

Digital Object Identifier 10.1109/TITS.2020.2972974

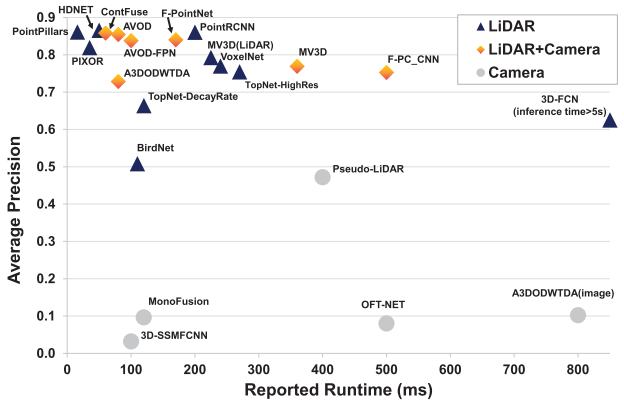


Fig. 2. Average precision (AP) vs. runtime. Visualized are deep learning approaches that use LiDAR, camera, or both as inputs for car detection on the KITTI bird's eye view test dataset. Moderate APs are summarized. The results are mainly based on the KITTI leader-board [6] (visited on Apr. 20, 2019). On the leader-board only the published methods are considered.

multi-modal sensors for scene understanding in autonomous driving. Fig. 2 shows some recently published methods and their performance on the KITTI dataset [6]. All methods with the highest performance are based on deep learning, and many methods that fuse camera and LiDAR information produce better performance than those using either LiDAR or camera alone. In this paper, we focus on two fundamental perception problems, namely, **object detection** and **semantic segmentation**. In the rest of this paper, we will call them **deep multi-modal perception** unless mentioned otherwise.

When developing methods for deep multi-modal object detection or semantic segmentation, it is important to consider the input data: Are there any multi-modal datasets available and how is the data labeled (cf. Tab. IV in the appendix)? Do the datasets cover diverse driving scenarios (cf. Sec. VI-A1)? Is the data of high quality (cf. Sec. VI-A2)? Additionally, we need to answer several important questions on designing the neural network architecture: Which modalities should be combined via fusion, and how to represent and process them properly ("What to fuse" cf. Sec. VI-B1)? Which fusion operations and methods can be used ("How to fuse" cf. Sec. VI-B2)? Which stage of feature representation is optimal for fusion ("When to fuse" cf. Sec. VI-B2)?

A. Related Works

Despite the fact that many methods have been proposed for deep multi-modal perception in autonomous driving, there is no published summary examining available multi-modal datasets, and there is no guideline for network architecture design. Yin and Berger, [7] summarize 27 datasets for autonomous driving that were published between 2006 and 2016, including the datasets recorded with a single camera alone or multiple sensors. However, many new multi-modal datasets have been released since 2016, and it is worth summarizing them. Ramachandram and Taylor [8] provide an overview on deep multi-modal learning, and mention its applications in diverse research fields, such as robotic grasping and human action recognition. Janai *et al.* [9] conduct a comprehensive summary on computer vision problems for autonomous driving, such as scene flow and scene construction. Recently, Arnold *et al.* [10] survey the 3D object detection problem in autonomous driving. They summarize methods based on monocular images or point clouds, and briefly mention some works that fuse vision camera and LiDAR information.

B. Contributions

To the best of our knowledge, there is no survey that focuses on deep multi-modal object detection (2D or 3D) and semantic segmentation for autonomous driving, which makes it difficult for beginners to enter this research field. Our review paper attempts to narrow this gap by conducting a summary of newly-published datasets (2013-2019), and fusion methodologies for deep multi-modal perception in autonomous driving, as well as by discussing the remaining challenges and open questions.

We first provide background information on multi-modal sensors, test vehicles, and modern deep learning approaches in object detection and semantic segmentation in Sec. II. We then summarize multi-modal datasets and perception problems in Sec. III and Sec. IV, respectively. Sec. V summarizes the fusion methodologies regarding "what to fuse", "when to fuse" and "how to fuse". Sec. VI discusses challenges and open questions when developing deep multi-modal perception systems in order to fulfill the requirements of "accuracy", "robustness" and "real-time", with a focus on data preparation and fusion methodology. We highlight the importance of data diversity, temporal and spatial alignment, and labeling efficiency for multi-modal data preparation. We also highlight the lack of research on fusing Radar signals, as well as the importance of developing fusion methodologies that tackle open dataset problems or increase network robustness. Sec. VII concludes this work. In addition, we provide an interactive online platform for navigating topics and methods for each reference. The platform can be found here: https://boschresearch.github.io/multimodalperception/.

II. Background

This section provides the background information for deep multi-modal perception in autonomous driving. First, we briefly summarize typical automotive sensors, their sensing modalities, and some vehicles for test and research purposes. Next, we introduce deep object detection and semantic segmentation. Since deep learning has most-commonly been applied to image-based signals, here we mainly discuss image-based methods. We will introduce other methods that process LiDAR and Radar data in Sec. V-A. For a more comprehensive overview on object detection and semantic segmentation, we refer the interested reader to the review papers [11], [12]. For a complete review of computer vision problems in autonomous driving (e.g. optical flow, scene reconstruction, motion estimation), cf. [9].

A. Sensing Modalities for Autonomous Driving

1) Visual and Thermal Cameras: Images captured by visual and thermal cameras can provide detailed texture information of a vehicle's surroundings. While visual cameras are sensitive

to lighting and weather conditions, thermal cameras are more robust to daytime/nighttime changes as they detect infrared radiation that relates to heat from objects. However, both types of cameras however cannot directly provide depth information.

2) LiDARs: LiDARs (Light Detection And Ranging) give accurate depth information of the surroundings in the form of 3D points. They measure reflections of laser beams which they emit with a certain frequency. LiDARs are robust to different lighting conditions, and less affected by various weather conditions such as fog and rain than visual cameras. However, typical LiDARs are inferior to cameras for object classification since they cannot capture the fine textures of objects, and their points become sparse with distant objects. Recently, flash LiDARs were developed which can produce detailed object information similar to camera images. Frequency Modulated Continuous Wave (FMCW) LiDARs can provide velocity information.

3) Radars: Radars (Radio Detection And Ranging) emit radio waves to be reflected by an obstacle, measures the signal runtime, and estimates the object's radial velocity by the Doppler effect. They are robust against various lighting and weather conditions, but classifying objects via Radars is very challenging due to their low resolution. Radars are often applied in adaptive cruise control (ACC) and traffic jam assistance systems [13].

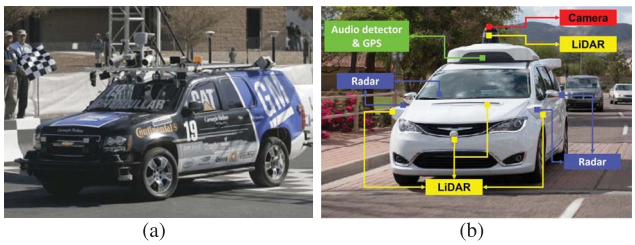
4) Ultrasonics: Ultrasonic sensors send out high-frequency sound waves to measure the distance to objects. They are typically applied for near-range object detection and in low speed scenarios, such as automated parking [13]. Due to the sensing properties, Ultrasonics are largely affected by air humidity, temperature, or dirt.

5) GNSS and HD Maps: GNSS (Global Navigation Satellite Systems) provide accurate 3D object positions by a global satellite system and the receiver. Examples of GNSS are GPS, Galileo and GLONASS. First introduced to automotive as navigation tools in driver assistance functions [13], currently GNSS is also used together with HD Maps for path planning and ego-vehicle localization for autonomous vehicles.

6) IMU and Odometers: Unlike sensors discussed above which capture information in the external environment (i.e. "exteroceptive sensors"), Inertial Measurement Units (IMU) and odometers provide vehicles' internal information (i.e. "proprioceptive sensors") [13]. IMU measure the vehicles' accelerations and rotational rates, and odometers the odometry. They have been used in vehicle dynamic driving control systems since the 1980s. Together with the exteroceptive sensors, they are currently used for accurate localization in autonomous driving.

B. Test Vehicle Setup

Equipped with multiple sensors introduced in Sec. II-A, many autonomous driving tests have been conducted. For example, the Tartan Racing Team developed an autonomous vehicle called "Boss" and won the DARPA Urban Challenge in 2007 (cf. Fig. 3(a)) [2]. The vehicle was equipped with a camera and several Radars and LiDARs. Google (Waymo) has tested their driverless cars in more than 20 US cities



(a) (b)

Fig. 3. (a) The Boss autonomous car at DARPA 2007 [2], (b) Waymo self-driving car [14].

driving 8 million miles on public roads (cf. Fig. 3(b)) [14]; BMW has tested autonomous driving on highways around Munich since 2011 [15]; Daimler mounted a stereo camera, two mono cameras, and several Radars on a Mercedes Benz S-Class car to drive autonomously on the Bertha Benz memorial route in 2013 [16]. Our interactive online platform provides a detailed description for more autonomous driving tests, including Uber, Nvidia, GM Cruise, Baidu Apollo, as well as their sensor setup.

Besides driving demonstrations, real-world datasets are crucial for autonomous driving research. In this regard, several research projects use *data* vehicles with multi-modal sensors to build open datasets. These data vehicles are usually equipped with cameras, LiDARs and GPS/IMUs to collect images, 3D point clouds, and vehicle localization information. Sec. III provides an overview of multi-modal datasets in autonomous driving.

C. Deep Object Detection

Object detection is the task of recognizing and localizing multiple objects in a scene. Objects are usually recognized by estimating a classification probability and localized with bounding boxes (cf. Fig. 1). Deep learning approaches have set the benchmark on many popular object detection datasets, such as PASCAL VOC [17] and COCO [18], and have been widely applied in autonomous driving, including detecting traffic lights [19]–[22], road signs [23]–[25], people [26]–[28], or vehicles [29]–[33], to name a few. State-of-the-art deep object detection networks follow one of two approaches: the two-stage or the one-stage object detection pipelines. Here we focus on image-based detection.

1) Two-Stage Object Detection: In the first stage, several class-agnostic object candidates called regions of interest (ROI) or region proposals (RP) are extracted from a scene. Then, these candidates are verified, classified, and refined in terms of classification scores and locations. OverFeat [34] and R-CNN [35] are among pioneering works that employ deep learning for object detection. In these works, ROIs are first generated by the sliding window approach (OverFeat [34]) or selective search (R-CNN [35]) and then advanced into a regional CNN to extract features for object classification and bounding box regression. SPPnet [36] and Fast-RCNN [37] propose to obtain regional features directly from global feature maps by applying a larger CNN (e.g. VGG [38], ResNet [39], GoogLeNet [40]) on the whole image. Faster R-CNN [41] unifies the object detection pipeline and adopts the Region

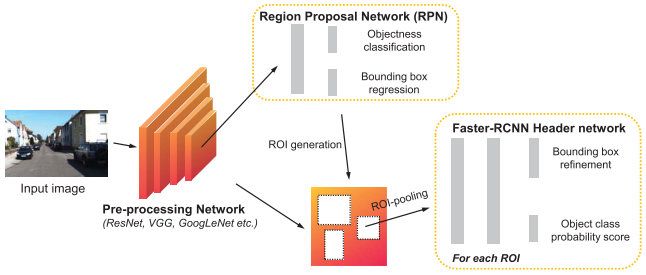


Fig. 4. The Faster R-CNN object detection network. It consists of three parts: a pre-processing network to extract high-level image features, a Region Proposal Network (RPN) that produces region proposals, and a Faster-RCNN head which fine-tunes each region proposal.

Proposal Network (RPN), a small fully-connected network, to slide over the high-level CNN feature maps for ROI generation (cf. Fig. 4). Following this line, R-FCN [42] proposes to replace fully-connected layers in an RPN with convolutional layers and builds a fully-convolutional object detector.

2) One-Stage Object Detection: This method aims to map the feature maps directly to bounding boxes and classification scores via a single-stage, unified CNN model. For example, MultiBox [43] predicts a binary mask from the entire input image via a CNN and infers bounding boxes at a later stage. YOLO [44] is a more complete unified detector which regresses the bounding boxes directly from the CNN model. SSD [45] handles objects with various sizes by regressing multiple feature maps of different resolution with small convolutional filters to predict multi-scale bounding boxes.

In general, two-stage object detectors like Faster-RCNN tend to achieve better detection accuracy due to the region proposal generation and refinement paradigm. This comes with the cost of higher inference time and more complex training. Conversely, one-stage object detectors are faster and easier to be optimized, yet under-perform compared to two-stage object detectors in terms of accuracy. Huang *et al.* [46] systematically evaluate the speed/accuracy trade-offs for several object detectors and backbone networks.

D. Deep Semantic Segmentation

The target of semantic segmentation is to partition a scene into several meaningful parts, usually by labeling each pixel in the image with semantics (pixel-level semantic segmentation) or by simultaneously detecting objects and doing per-instance per-pixel labeling (instance-level semantic segmentation). Recently, panoptic segmentation [47] is proposed to unify pixel-level and instance-level semantic segmentation, and it starts to get more attentions for autonomous driving [48]–[50]. Though semantic segmentation was first introduced to process camera images, many methods have been proposed for segmenting LiDAR points as well (e.g. [51]–[56]).

Many datasets have been published for semantic segmentation, such as Cityscape [57], KITTI [6], Toronto City [58], Mapillary Vistas [4], and ApolloScape [59]. These datasets advance the deep learning research for semantic segmentation in autonomous driving. For example, [54], [60], [61] focus on pixel-wise semantic segmentation for multiple classes including road, car, bicycle, column-pole, tree, sky, etc; [52] and [62] concentrate on road segmentation; and [51], [63], [64] deal with instance segmentation for various traffic participants.

Similar to object detection introduced in Sec. II-C, semantic segmentation can also be classified into two-stage and one-stage pipelines. In the two-stage pipeline, region proposals are first generated and then fine-tuned mainly for instance-level segmentation (e.g. R-CNN [65], SDS [66], Mask-RCNN [63]). A more common way for a semantic segmentation is the one-stage pipeline based on a Fully Convolutional Network (FCN) originally proposed by Long *et al.* [67]. In this work, the fully-connected layers in a CNN classifier for predicting classification scores are replaced with convolutional layers to produce coarse output maps. These maps are then up-sampled to dense pixel labels by backwards convolution (i.e. deconvolution). Kendall *et al.* [61] extend FCN by introducing an encoder-decoder CNN architecture. The encoder serves to produce hierarchical image representations with a CNN backbone such as VGG or ResNet (removing fully-connected layers). The decoder, conversely, restores these low-dimensional features back to original resolution by a set of upsampling and convolution layers. The restored feature maps are finally used for pixel-label prediction.

Global image information provides useful context cues for semantic segmentation. However, vanilla CNN structures only focus on local information with limited receptive fields. In this regard, many methods have been proposed to incorporate global information, such as dilated convolutions [68], [69], multi-scale prediction [70], as well as adding Conditional Random Fields (CRFs) as post-processing step [71].

Real-time performance is important in autonomous driving applications. However, most works only focus on segmentation accuracy. In this regard, Siam *et al.* [72] made a comparative study on the real-time performance among several semantic segmentation architectures, regarding the operations (GFLOPs) and the inference speed (fps).

III. Multi-Modal Datasets

Most deep multi-modal perception methods are based on supervised learning. Therefore, multi-modal datasets with labeled ground-truth are required for training such deep neural networks. In the following, we summarize several real-world datasets published since 2013, regarding sensor setups, recording conditions, dataset size and labels (cf. Tab. IV in the appendix). Note that there exist some virtual multi-modal datasets generated from game engines. We will discuss them in Sec. VI-A1.

A. Sensing Modalities

All reviewed datasets include RGB camera images. In addition, [6], [59], [73]–[88] provide LiDAR point clouds, and [89]–[91] thermal images. The KAIST Multispectral Dataset [92] provides both thermal images and LiDAR data. Bus data is included additionally in [86]. Only the very recently nuScenes [88], Oxford Radar RobotCar [84] and Astyx HiRes2019 Datasets [93] provide Radar data.

B. Recording Conditions

Even though the KITTI dataset [74] is widely used for autonomous driving research, the diversity of its recording conditions is relatively low: it is recorded in Karlsruhe - a mid-sized city in Germany, only during daytime and on sunny days. Other reviewed datasets such as [59], [77], [78], [81], [86]–[88] are recorded in more than one location. To increase the diversity of lighting conditions, [59], [79]–[81], [81], [83], [85], [87]–[91] collect data in both daytime and nighttime, and [92] considers various lighting conditions throughout the day, including sunrise, morning, afternoon, sunset, night, and dawn. The Oxford Dataset [73] and the Oxford Radar RobotCar Dataset [84] are collected by driving the car around the Oxford area during the whole year. It contains data under different weather conditions, such as heavy rain, night, direct sunlight and snow. Other datasets containing diverse weather conditions are [59], [85], [87], [88]. In [94], LiDAR is used as a reference sensor for generating ground-truth, hence we do not consider it a multi-modal dataset. However the diversity in the recording conditions is large, ranging from dawn to night, as well as reflections, rain and lens flare. The cross-season dataset [95] emphasizes the importance of changes throughout the year. However, it only provides camera images and labels for semantic segmentation. Similarly, the visual localization challenge and the corresponding benchmark [96] cover weather and season diversity (but no new multi-modal dataset is introduced). The recent Eurocity dataset [87] is the most diverse dataset we have reviewed. It is recorded in different cities from several European countries. All seasons are considered, as well as weather and daytime diversity. To date, the dataset is camera-only and other modalities (e.g. LiDARs) are announced.

C. Dataset Size

The dataset size ranges from only 1,569 frames up to over 11 million frames. The largest dataset with ground-truth labels that we have reviewed is the nuScenes Dataset [88] with nearly 1,4M frames. Compared to the image datasets in the computer vision community, the multi-modal datasets are still relatively small. However, the dataset size has grown by two orders of magnitudes between 2014 and 2019 (cf. Fig. 5(b)).

D. Labels

Most of the reviewed datasets provide ground-truth labels for 2D object detection and semantic segmentation tasks [59], [74], [87], [89]–[92]. KITTI [74] also labels tracking, optical flow, visual odometry, and depth for various computer vision problems. BLV3D [79] provides labels for tracking, interaction and intention. Labels for 3D scene understanding are provided by [59], [74], [78]–[83], [88].

Depending on the focus of a dataset, objects are labeled into different classes. For example, [89] only contains label for people, including distinguishable individuals (labeled as "Person"), non-distinguishable individuals (labeled as "People"), and cyclists; [59] classifies objects into five groups, and provides 25 fine-grained labels, such as truck, tricycle, traffic cone, and trash can. The Eurocity dataset [87] focuses on vulnerable road-users (mostly pedestrian). Instead of labeling objects, [76] provides a dataset for place categorization. Scenes are classified into forest, coast, residential area, urban area and indoor/outdoor parking lot. Reference [77] provides vehicle speed and wheel angles for driving behavior predictions. The BLV3D dataset [79] provides unique labeling for interaction and intention.

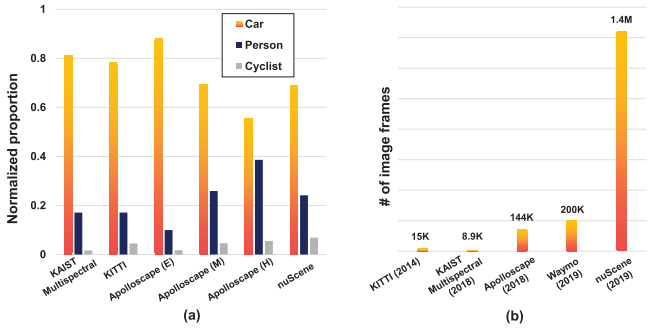


Fig. 5. (a) Normalized percentage of objects of car, person, and cyclist classes in KAIST Multispectral [92], KITTI [6], Apolloscape [59] (E: easy, M: moderate, and H: hard refer to the number of moveable objects in the frame - details can be found in [59]), and nuScene dataset [88]. (b) Number of camera image frames in several datasets. An increase by two orders of magnitude of the dataset size can be seen.

The object classes are very imbalanced. Fig. 5(a) compares the percentage of car, person, and cyclist classes from four reviewed datasets. There are much more objects labeled as car than person or cyclist.

IV. Deep Multi-Modal Perception Problems for Autonomous Driving

In this section, we summarize deep multi-modal perception problems for autonomous driving based on sensing modalities and targets. In the appendix for this work, we show an overview of the existing methods in Tab. V and Tab. VI, and an accuracy and runtime comparison among several methods in Tab. II and Tab. III.

A. Deep Multi-Modal Object Detection

1) Sensing Modalities: Most existing works combine RGB images from visual cameras with 3D LiDAR point clouds [97]–[115]. Some other works focus on fusing the RGB images from visual cameras with images from thermal cameras [90], [116]–[118]. Furthermore, Mees *et al.* [119] employ a Kinect RGB-D camera to fuse RGB images and depth images; Schneider *et al.* [60] generate depth images from a stereo camera and combine them with RGB images; Yang *et al.* [120] and Cascas *et al.* [121] leverage HD maps to provide prior knowledge of the road topology.

2) 2D or 3D Detection: Many works [60], [90], [98]–[100], [105], [107], [108], [110], [116]–[119], [122] deal with the 2D object detection problem on the front-view 2D image plane. Compared to 2D detection, 3D detection is more challenging since the object's distance to the ego-vehicle needs to be estimated. Therefore, accurate depth information provided by LiDAR sensors is highly beneficial. In this regard,

some papers including [97], [101]–[104], [106], [112], [114] combine RGB camera images and LiDAR point clouds for 3D object detection. In addition, Liang *et al.* [115] propose a multi-task learning network to aid 3D object detection. The auxiliary tasks include camera depth completion, ground plane estimation, and 2D object detection. How to represent the modalities properly is discussed in section V-A.

3) What to Detect: Complex driving scenarios often contain different types of road users. Among them, cars, cyclists, and pedestrians are highly relevant to autonomous driving. In this regard, [97], [98], [105], [107], [109] employ multi-modal neural networks for car detection; [100], [107], [108], [116]–[119] focus on detecting non-motorized road users (pedestrians or cyclists); [60], [90], [99], [101]–[104], [110], [114], [115] detect both.

B. Deep Multi-Modal Semantic Segmentation

Compared to the object detection problem summarized in Sec. IV-A, there are fewer works on multi-modal semantic segmentation: [91], [118], [123] employ RGB and thermal images, [60] fuses RGB images and depth images from a stereo camera, [124]–[126] combine RGB, thermal, and depth images for semantic segmentation in diverse environments such as forests, [122] fuses RGB images and LiDAR point clouds for off-road terrain segmentation and [127]–[131] for road segmentation. Apart from the above-mentioned works for semantic segmentation on the 2D image plane, [124], [132] deal with 3D segmentation on LiDAR points.

V. Methodology

When designing a deep neural network for multi-modal perception, three questions need to be addressed - *What to fuse*: what sensing modalities should be fused, and how to represent and process them in an appropriate way; *How to fuse*: what fusion operations should be utilized; *When to fuse*: at which stage of feature representation in a neural network should the sensing modalities be combined. In this section, we summarize existing methodologies based on these three aspects.

A. What to Fuse

LiDARs and cameras (visual cameras, thermal cameras) are the most common sensors for multi-modal perception in the literature. While the interest in processing Radar signals via deep learning is growing, only a few papers discuss deep multi-modal perception with Radar for autonomous driving (e.g. [133]). Therefore, we focus on several ways to represent and process LiDAR point clouds and camera images separately, and discuss how to combine them together. In addition, we briefly summarize Radar perception using deep learning.

1) LiDAR Point Clouds: LiDAR point clouds provide both depth and reflectance information of the environment. The depth information of a point can be encoded by its Cartesian coordinates $[x, y, z]$, distance $\sqrt{x^2 + y^2 + z^2}$, density, or HHA features (Horizontal disparity, Height, Angle) [65], or any other 3D coordinate system. The reflectance information is given by intensity.

There are mainly three ways to process point clouds. One way is by discretizing the 3D space into 3D voxels and assigning the points to the voxels (e.g. [29], [112], [134]–[136]). In this way, the rich 3D shape information of the driving environment can be preserved. However, this method results in many empty voxels as the LiDAR points are usually sparse and irregular. Processing the sparse data via clustering (e.g. [99], [105]–[107]) or 3D CNN (e.g. [29], [135]) is usually very time-consuming and infeasible for online autonomous driving. Zhou and Tuzel [134] propose a voxel feature encoding (VFE) layer to process the LiDAR points efficiently for 3D object detection. They report an inference time of 225 ms on the KITTI dataset. Yan *et al.* [137] add several sparse convolutional layers after the VFE to convert the sparse voxel data into 2D images, and then perform 3D object detection on them. Unlike the common convolution operation, the sparse convolution only computes on the locations associated with input points. In this way, they save a lot of computational cost, achieving an inference time of only 25 ms.

The second way is to directly learn over 3D LiDAR points in continuous vector space without voxelization. PointNet [138] and its improved version PointNet++ [139] propose to predict individual features for each point and aggregate the features from several points via max pooling. This method was firstly introduced in 3D object recognition and later extended by Qi *et al.* [104], Xu *et al.* [103] and Shin *et al.* [140] to 3D object detection in combination with RGB images. Furthermore, Wang *et al.* [141] propose a new learnable operator called Parametric Continuous Convolution to aggregate points via a weighted sum, and Li *et al.* [142] propose to learn a χ transformation before applying transformed point cloud features into standard CNN. They are tested in semantic segmentation or LiDAR motion estimation tasks.

A third way to represent 3D point clouds is by projecting them onto 2D grid-based feature maps so that they can be processed via 2D convolutional layers. In the following, we distinguish among spherical map, camera-plane map (CPM), as well as bird's eye view (BEV) map. Fig. 6 illustrates different LiDAR representations in 2D.

A spherical map is obtained by projecting each 3D point onto a sphere, characterized by azimuth and zenith angles. It has the advantage of representing each 3D point in a dense and compact way, making it a suitable representation for point cloud segmentation (e.g. [51]). However, the size of the representation can be different from camera images. Therefore, it is difficult to fuse them at an early stage. A CPM can be produced by projecting the 3D points into the camera coordinate system, provided the calibration matrix. A CPM can be directly fused with camera images, as their sizes are the same. However, this representation leaves many pixels empty. Therefore, many methods have been proposed to up-sample such a sparse feature map, e.g. mean average [110], nearest neighbors [143], or bilateral filter [144]. Compared to the above-mentioned feature maps which encode LiDAR information in the front-view, a BEV map avoids occlusion problems because objects occupy different space in the map. In addition, the BEV preserves the objects' length and width, and directly provides the objects' positions on the ground

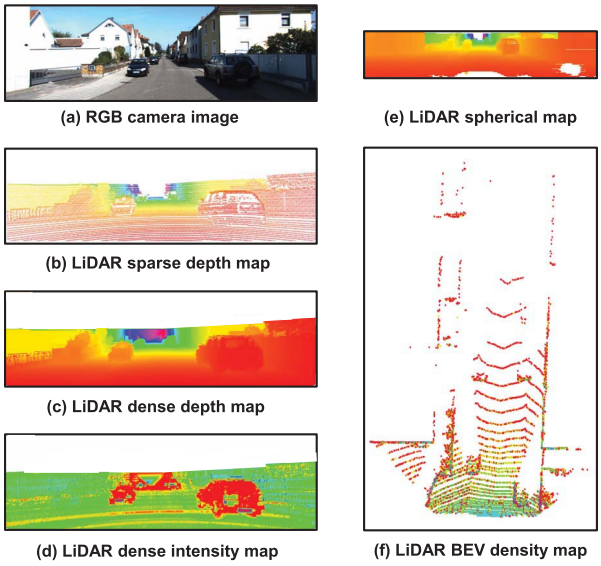


Fig. 6. RGB image and different 2D LiDAR representation methods. (a) A standard RGB image, represented by a pixel grid and color channel values. (b) A sparse (front-view) depth map obtained from LiDAR measurements represented on a grid. (c) Interpolated depth map. (d) Interpolation of the measured reflectance values on a grid. (e) Interpolated representation of the measured LiDAR points (surround view) on a spherical map. (f) Projection of the measured LiDAR points (front-facing) to bird's eye view (no interpolation).

plane, making the localization task easier. Therefore, the BEV map is widely applied to 3D environment perception. For example, Chen *et al.* [97] encode point clouds by height, density and intensity maps in BEV. The height maps are obtained by dividing the point clouds into several slices. The density maps are calculated as the number of points within a grid cell, normalized by the number of channels. The intensity maps directly represent the reflectance measured by the LiDAR on a grid. Lang *et al.* [145] argue that the hard-coded features for BEV representation may not be optimal. They propose to learn features in each column of the LiDAR BEV representation via PointNet [138], and feed these learnable feature maps to standard 2D convolution layers.

2) Camera Images: Most methods in the literature employ RGB images from visual cameras or one type of infrared images from thermal cameras (near-infrared, mid-infrared, far-infrared). Besides, some works extract additional sensing information, such as optical flow [119], depth [60], [124], [125], or other multi-spectral images [90], [124].

Camera images provide rich texture information of the driving surroundings. However, objects can be occluded and the scale of a single object can vary significantly in the camera image plane. For 3D environment inference, the bird's eye view that is commonly used for LiDAR point clouds might be a better representation. Roddick *et al.* [146] propose a Orthographic Feature Transform (OFT) algorithm to project the RGB image features onto the BEV plane. The BEV feature maps are further processed for 3D object detection from monocular camera images. Lv *et al.* [129] project each image pixel with the corresponding LiDAR point onto the BEV plane and fuse the multi-modal features for road segmentation. Wang *et al.* [147] and their successive work [148] propose to convert RGB images into pseudo-lidar representation by estimating the image depth, and then use state-of-the-art BEV LiDAR detector to significantly improve the detection performance.

3) Processing LiDAR Points and Camera Images in Deep Multi-Modal Perception: Tab. V and Tab. VI in the appendix summarize existing methods to process sensors' signals for deep multi-modal perception, mainly LiDAR points and camera images. From the tables we have three observations: (1). Most works propose to fuse LiDAR and camera features extracted from 2D convolution neural networks. To do this, they project LiDAR points on the 2D plane and process the feature maps through 2D convolutions. Only a few works extract LiDAR features by PointNet (e.g. [103], [104], [127]) or 3D convolutions (e.g. [122]); (2). Several works on multi-modal object detection cluster and segment 3D LiDAR points to generate 3D region proposals (e.g. [99], [105], [107]). Still, they use a LiDAR 2D representation to extract features for fusion; (3). Several works project LiDAR points on the camera-plane or RGB camera images on the LiDAR BEV plane (e.g. [129], [130], [149]) in order to align the features from different sensors, whereas many works propose to fuse LiDAR BEV features directly with RGB camera images (e.g. [97], [102]). This indicates that the networks implicitly learn to align features of different viewpoints. Therefore, a well-calibrated sensor setup with accurate spatial and temporal alignment is the prerequisite for accurate multi-modal perception, as will be discussed in Sec. VI-A2.

4) Radar Signals: Radars provide rich environment information based on received amplitudes, ranges, and the Doppler spectrum. The Radar data can be represented by 2D feature maps and processed by convolutional neural networks. For example, Lombacher *et al.* employ Radar grid maps made by accumulating Radar data over several time-stamps [150] for static object classification [151] and semantic segmentation [152] in autonomous driving. Visentin *et al.* show that CNNs can be employed for object classification in a post-processed range-velocity map [153]. Kim *et al.* [154] use a series of Radar range-velocity images and convolutional recurrent neural networks for moving objects classification. Amin and Erol [155] feed spectrogram from Time Frequency signals as 2D images into a stacked auto-encoders to extract high-level Radar features for human motion recognition. The Radar data can also be represented directly as "point clouds" and processed by PointNet++ [139] for dynamic object segmentation [156]. Besides, Woehler *et al.* [157] encode features from a cluster of Radar points for dynamic object classification. Chadwick *et al.* [133] first project Radar points on the camera plane to build Radar range-velocity images, and then combine with camera images for distant vehicle detection.

B. How to Fuse

This section summarizes typical fusion operations in a deep neural network. For simplicity we restrict our discussion to two sensing modalities, though more still apply. Denote M_i and M_j as two different modalities, and $f_l^{M_i}$ and $f_l^{M_j}$ their feature

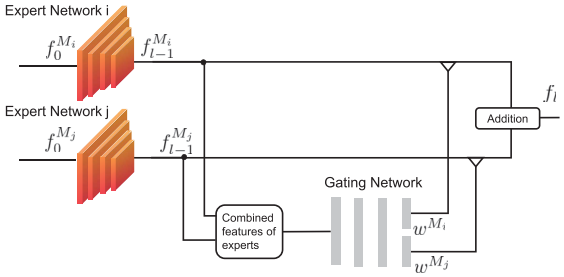


Fig. 7. An illustration of the Mixture of Experts fusion method. Here we show the combined features which are derived from the output layers of the expert networks. They can be extracted from the intermediate layers as well.

maps in the l^{th} layer of the neural network. Also denote $G_l(\cdot)$ as a mathematical description of the feature transformation applied in layer l of the neural network.

1) Addition or Average Mean: This join operation adds the feature maps element-wise, i.e. $f_l = G_{l-1}\left(f_{l-1}^{M_i} + f_{l-1}^{M_j}\right)$, or calculates the average mean of the feature maps.

2) Concatenation: Combines feature maps by $f_l = G_{l-1}\left(f_{l-1}^{M_i} \frown f_{l-1}^{M_j}\right)$. The feature maps are usually stacked along their depth before they are advanced to a convolution layer. For a fully connected layer, these features are usually flattened into vectors and concatenated along the rows of the feature maps.

3) Ensemble: This operation ensembles feature maps from different sensing modalities via $f_l = G_{l-1}\left(f_{l-1}^{M_i}\right) \cup G_{l-1}\left(f_{l-1}^{M_j}\right)$. As will be introduced in the following sections (Sec. V-C4 and Sec. V-C5), ensembles are often used to fuse ROIs in object detection networks.

4) Mixture of Experts: The above-mentioned fusion operations do not consider the informativeness of a sensing modality (e.g. at night time RGB camera images bring less information than LiDAR points). These operations are applied, hoping that the network can *implicitly* learn to weight the feature maps. In contrast, the Mixture of Experts (MoE) approach *explicitly* models the weight of a feature map. It is first introduced in [158] for neural networks and then extended in [119], [125], [159]. As Fig. 7 illustrates, the feature map of a sensing modality is processed by its domain-specific network called "expert". Afterwards, the outputs of multiple expert networks are averaged with the weights w^{M_i}, w^{M_j} predicted by a gating network which takes the combined features output by the expert networks as inputs h via a simple fusion operation such as concatenation:

$$f_l = G_l\left(w^{M_i} \cdot f_{l-1}^{M_i} + w^{M_j} \cdot f_{l-1}^{M_j}\right), \text{ with } w^{M_i} + w^{M_j} = 1. \tag{1}$$

C. When to Fuse

Deep neural networks represent features hierarchically and offer a wide range of choices to combine sensing modalities at early, middle, or late stages (Fig. 8). In the sequel, we discuss the early, middle, and late fusions in detail. For each fusion scheme, we first give mathematical descriptions using the same notations as in Sec. V-B, and then discuss their properties. Note that there exists some works that fuse features from the early stage till late stages in deep neural networks (e.g. [160]). For simplicity, we categorize this fusion scheme as "middle fusion". Compared to the semantic segmentation where multi-modal features are fused at different stages in FCN, there exist more diverse network architectures and more fusion variants in object detection. Therefore, we additionally summarize the fusion methods specifically for the object detection problem. Finally, we discuss the relationship between the fusion operation and the fusion scheme.

Note that we do not find conclusive evidence from the methods we have reviewed that one fusion method is better than the others. The performance is highly dependent on sensing modalities, data, and network architectures.

1) Early Fusion: This method fuses the raw or pre-processed sensor data. Let us define $f_l = f_{l-1}^{M_i} \oplus f_{l-1}^{M_j}$ as a fusion operation introduced in Sec. V-B. For a network that has $L+1$ layers, an early fusion scheme can be described as:

$$f_L = G_L\Big(G_{L-1}\Big(\cdots G_l\Big(\cdots G_2\big(G_1(f_0^{M_i} \oplus f_0^{M_j})\big)\Big)\Big)\Big), \tag{2}$$

with $l = [1, 2, \cdots, L]$. Early fusion has several pros and cons. First, the network learns the joint features of multiple modalities at an early stage, fully exploiting the information of the raw data. Second, early fusion has low computation requirements and a low memory budget as it jointly processes the multiple sensing modalities. This comes with the cost of model inflexibility. As an example, when an input is replaced with a new sensing modality or the input channels are extended, the early fused network needs to be retrained completely. Third, early fusion is sensitive to spatial-temporal data misalignment among sensors which are caused by calibration error, different sampling rate, and sensor defect.

2) Late Fusion: This fusion scheme combines decision outputs of each domain specific network of a sensing modality. It can be described as:

$$f_L = G_L^{M_i}\Big(G_{L-1}^{M_i}\big(\cdots G_1^{M_i}(f_0^{M_i})\big)\Big) \oplus G_L^{M_j}\Big(G_{L-1}^{M_j}\big(\cdots G_1^{M_j}(f_0^{M_j})\big)\Big). \tag{3}$$

Late fusion has high flexibility and modularity. When a new sensing modality is introduced, only its domain specific network needs to be trained, without affecting other networks. However, it suffers from high computation cost and memory requirements. In addition, it discards rich intermediate features which may be highly beneficial when being fused.

3) Middle Fusion: Middle fusion is the compromise of early and late fusion: It combines the feature representations from different sensing modalities at intermediate layers. This enables the network to learn cross modalities with different feature representations and at different depths. Define $l^\star$ as the layer from which intermediate features begin to be fused. The middle fusion can be executed at this layer only once:

$$f_L = G_L\Big(\cdots G_{l^\star+1}\Big(G_{l^\star}^{M_i}\big(\cdots G_1^{M_i}(f_0^{M_i})\big) \oplus G_{l^\star}^{M_j}\big(\cdots G_1^{M_j}(f_0^{M_j})\big)\Big)\Big). \tag{4}$$

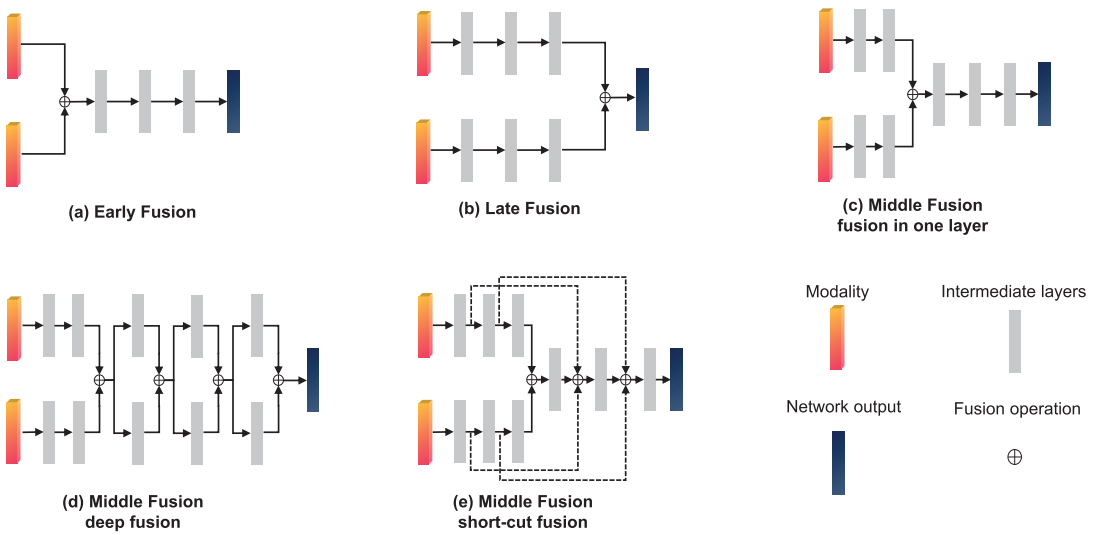


Fig. 8. An illustration of early fusion, late fusion, and several middle fusion methods.

Alternatively, they can be fused hierarchically, such as by deep fusion [97], [161]:

$$f_{l^\star+1} = f_{l^\star}^{M_i} \oplus f_{l^\star}^{M_j},$$
$$f_{k+1} = G_k^{M_i}(f_k) \oplus G_k^{M_j}(f_k),\ \forall k : k \in \left\{l^\star+1, \cdots, L\right\}. \quad (5)$$

or "short-cut fusion" [91]:

$$f_{l+1} = f_l^{M_i} \oplus f_l^{M_j},$$
$$f_{k+1} = f_k \oplus f_{k^\star}^{M_i} \oplus f_{k^\star}^{M_j},$$
$$\forall k : k \in \{l+1, \cdots, L\};\ \exists k^\star : k^\star \in \{1, \cdots, l-1\}. \quad (6)$$

Although the middle fusion approach is highly flexible, it is not easy to find the "optimal" way to fuse intermediate layers given a specific network architecture. We will discuss this challenge in detail in Sec. VI-B3.

4) Fusion in Object Detection Networks: Modern multi-modal object detection networks usually follow either the two-stage pipeline (RCNN [35], Fast-RCNN [37], Faster-RCNN [41]) or the one-stage pipeline (YOLO [44] and SSD [45]), as explained in detail in Sec. II-C. This offers a variety of alternatives for network fusion. For instance, the sensing modalities can be fused to generate regional proposals for a two-stage object detector. The regional multi-modal features for each proposal can be fused as well. Ku *et al.* [102] propose AVOD, an object detection network that fuses RGB images and LiDAR BEV images both in the region proposal network and the header network. Kim and Ghosh [108] ensemble the region proposals that are produced by LiDAR depth images and RGB images separately. The joint region proposals are then fed to a convolutional network for final object detection. Chen *et al.* [97] use LiDAR BEV maps to generate region proposals. For each ROI, the regional features from the LiDAR BEV maps are fused with those from the LiDAR front-view maps as well as camera images via deep fusion. Compared to object detections from LiDAR point clouds, camera images have been well investigated with larger labeled dataset and better 2D detection performance. Therefore, it is straightforward to exploit the predictions from well-trained image detectors when doing camera-LiDAR fusion. In this regard, [103], [104], [106] propose to utilize a pre-trained image detector to produce 2D bounding boxes, which build frustums in LiDAR point clouds. Then, they use these point clouds within the frustums for 3D object detection. Fig. 9 shows some exemplary fusion architectures for two-stage object detection networks. Tab. V in the appendix summarizes the methodologies for multi-modal object detection.

5) Fusion Operation and Fusion Scheme: Based on the papers that we have reviewed, feature concatenation is the most common operation, especially at early and middle stages. Element-wise average mean and addition operations are additionally used for middle fusion. Ensemble and Mixture of Experts are often used for middle to decision level fusion.

VI. Challenges and Open Questions

As discussed in the Introduction (cf. Sec. I), developing deep multi-modal perception systems is especially challenging for autonomous driving because it has high requirements in accuracy, robustness, and real-time performance. The predictions from object detection or semantic segmentation are usually transferred to other modules such as maneuver prediction and decision making. A reliable perception system is the prerequisite for a driverless car to run safely in uncontrolled and complex driving environments. In Sec. III and Sec. V we have summarized the multi-modal datasets and fusion methodologies. Correspondingly, in this section we discuss the remaining challenges and open questions for multi-modal data preparation and network architecture design. We focus on how to improve the accuracy and robustness of the multi-modal perception systems while guaranteeing real-time performance. We also discuss some open questions, such as evaluation metrics and network architecture design. Tab. I summarizes the challenges and open questions.

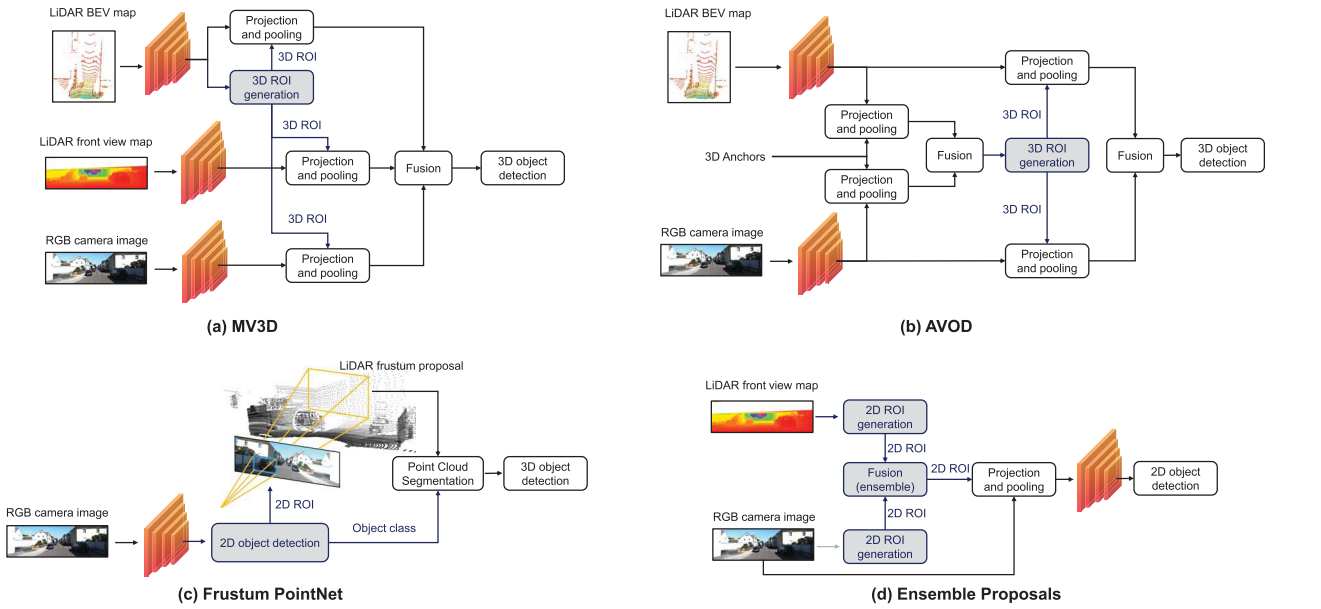


Fig. 9. Examplary fusion architectures for two-stage object detection networks. (a). MV3D [97]; (b). AVOD [102]; (c). Frustum PointNet [104]; (d). Ensemble Proposals [108].

TABLE I

AN OVERVIEW OF CHALLENGES AND OPEN QUESTIONS

Topics		Challenges	Open Questions
Multi-modal data preparation	Data diversity	• Relative small size of training dataset. • Limited driving scenarios and conditions, limited sensor variety, object class imbalance.	• Develop more realistic virtual datasets. • Finding optimal way to combine real- and virtual data. • Increasing labeling efficiency through cross-modal labeling, active learning, transfer learning, semi-supervised learning etc. Leveraging lifelong learning to update networks with continual data collection.
	Data quality	• Labeling errors. • Spatial and temporal misalignment of different sensors.	• Teaching network robustness with erroneous and noisy labels. • Integrating prior knowledge in networks. • Developing methods (e.g. using deep learning) to automatically register sensors.
Fusion methodology	"What to fuse"	• Too few sensing modalities are fused. • Lack of studies for different feature representations.	• Fusing multiple sensors with the same modality. • Fusing more sensing modalities, e.g. Radar, Ultrasonic, V2X communication. • Fusing with physical models and prior knowledge, also possible in the multi-task learning scheme. • Comparing different feature representation w.r.t informativeness and computational costs.
	"How to fuse"	• Lack of uncertainty quantification for each sensor channel. • Too simple fusion operations.	• Uncertainty estimation via e.g. Bayesian neural networks (BNN). • Propagating uncertainties to other modules, such as tracking and motion planning. • Anomaly detection by generative models. • Developing fusion operations that are suitable for network pruning and compression.
	"When to fuse"	• Fusion architecture is often designed by empirical results. No guideline for optimal fusion architecture design. • Lack of study for accuracy/speed or memory/robustness trade-offs.	• Optimal fusion architecture search. • Incorporating requirements of computation time or memory as regularization term. • Using visual analytics tool to find optimal fusion architecture.
Others	Evaluation metrics	• Current metrics focus on comparing networks' accuracy.	• Metrics to quantify the networks' robustness should be developed and adapted to multi-modal perception problems.
	More network architectures	• Current networks lack temporal cues and cannot guarantee prediction consistency over time. • They are designed mainly for modular autonomous driving.	• Using Recurrent Neural Network (RNN) for sequential perception. • Multi-modal end-to-end learning or multi-modal direct-perception.

A. Multi-Modal Data Preparation

1) Data Diversity: Training a deep neural network on a complex task requires a huge amount of data. Therefore, using large multi-modal datasets with diverse driving conditions, object labels, and sensors can significantly improve the network's accuracy and robustness against changing environments. However, it is not an easy task to acquire real-world data due to cost and time limitations as well as hardware

constraints. The size of open multi-modal datasets is usually much smaller than the size of image datasets. As a comparison, KITTI [6] records only 80,256 objects whereas ImageNet [162] provides 1,034,908 samples. Furthermore, the datasets are usually recorded in limited driving scenarios, weather conditions, and sensor setups (more details are provided in Sec. III). The distribution of objects is also very imbalanced, with much more objects being labeled as car than person or cyclist (Fig. 5). As a result, it is questionable how a deep multi-modal perception system trained with those public datasets performs when it is deployed to an unstructured environment.

One way to overcome those limitations is by data augmentation via simulation. In fact, a recent work [163] states that the most performance gain for object detection in the KITTI dataset is due to data augmentation, rather than advances in network architectures. Pfeuffer and Dietmayer [110] and Kim *et al.* [109] build augmented training datasets by adding artificial blank areas, illumination change, occlusion, random noises, etc. to the KITTI dataset. The datasets are used to simulate various driving environment changes and sensor degradation. They show that trained with such datasets, the network accuracy and robustness are improved. Some other works aim at developing virtual simulators to generate varying driving conditions, especially some dangerous scenarios where collecting real-world data is very costly or hardly possible. Gaidon *et al.* [164] build a virtual KITTI dataset by introducing a real to virtual cloning method to the original KITTI dataset, using the Unity Game Engine. Other works [165]–[170] generate virtual datasets purely from game engines, such as GTA-V, without a proxy of real-world datasets. Griffiths and Boehm [171] create a purely virtual LiDAR only dataset. In addition, Dosovitskiy *et al.* [172] develop an open-source simulator that can simulate multiple sensors in autonomous driving and Hurl *et al.* [173] release a large scale, virtual, multi-modal dataset with LiDAR data and visual camera. Despite many available virtual datasets, it is an open question to which extend a simulator can represent real-world phenomena. Developing more realistic simulators and finding the optimal way to combine real and virtual data are important open questions.

Another way to overcome the limitations of open datasets is by increasing the efficiency of data labeling. When building a multi-modal training dataset, it is relatively easy to drive the test vehicle and collect many data samples. However, it is very tedious and time-consuming to label them, especially when dealing with 3D labeling and LiDAR points. Lee *et al.* [174] develop a collaborative hybrid labeling tool, where 3D LiDAR point clouds are firstly weakly-labeled by human annotators, and then fine-tuned by pre-trained network based on F-PointNet [104]. They report that the labeling tool can significantly reduce the "task complexity" and "task switching", and have a 30× labeling speed-up (Fig. 10(a)). Piewak *et al.* [132] leverage a pre-trained image segmentation network to label LiDAR point clouds without human intervention. The method works by registering each LiDAR point with an image pixel, and transferring the image semantics predicted by the pre-trained network to the corresponding LiDAR points (cf. Fig. 10(b)). In another work, Mei *et al.* [175] propose a semi-supervised learning method to do 3D point segmentation labeling. With only a few manual labels together with pair-wise spatial constraints between adjacent data frames, a lot of objects can be labeled. Several works [176]–[178] propose to introduce active learning in semantic segmentation or object detection for autonomous driving. The networks iteratively query the human annotator some most informative samples in an unlabeled data pool and then update the networks' weights. In this way, much less labeled training data is required while reaching the same performance and saving human labeling efforts. There are many other methods in the machine learning literature that aim to reduce data labeling efforts, such as transfer learning [179], domain adaptation [180]–[184], and semi-supervised learning [185]. How to efficiently label multi-modal data in autonomous driving is an important and challenging future work, especially in scenarios where the signals from different sensors may not be matched (e.g. due to the distance some objects are only visible by visual camera but not by LiDAR). Finally, as there can always be new driving scenarios that are different from the training data, it is an interesting research topic to leverage lifelong learning [186] to update the multi-modal perception network with continual data collection.

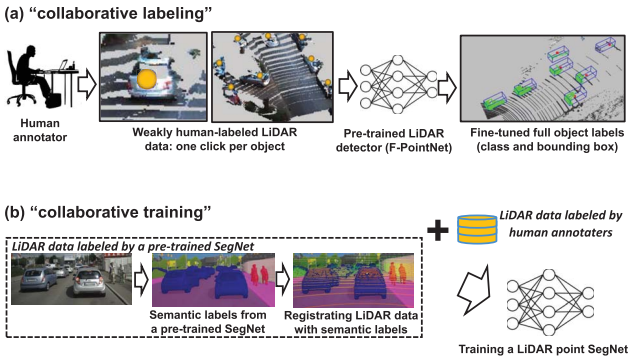


Fig. 10. Two examples of increasing data labeling efficiency in LiDAR data. (a) Collaborative labeling LiDAR points for 3D detection [174]: the LiDAR points within each object are firstly weakly-labeled by human annotators, and then fine-tuned by a pre-trained LiDAR detector based on the F-PointNet. (b) Collaborative training a semantic segmentation network (SegNet) for LiDAR points [132]: To boost the training data, a pre-trained image SegNet can be employed to transfer the image semantics.

2) Data Quality and Alignment: Besides data diversity and the size of the training dataset, data quality significantly affects the performance of a deep multi-modal perception system as well. Training data is usually labeled by human annotators to ensure the high labeling quality. However, humans are also prone to making errors. Fig. 11 shows two different errors in the labeling process when training an object detection network. The network is much more robust against labeling errors when they are randomly distributed, compared to biased labeling from the use of a deterministic pre-labeling. Training networks with erroneous labels is further studied in [187]–[190]. The impact on weak or erroneous labels on the performance of deep learning based semantic segmentation is investigated in [191], [192]. The influence of labelling errors on the accuracy of object detection is discussed in [193], [194].

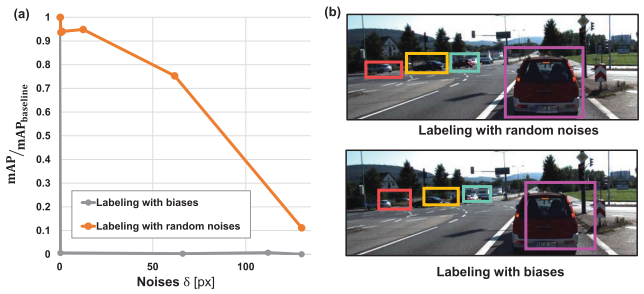


Fig. 11. (a) An illustration for the influence of label quality on the performance of an object detection network [195]. The network is trained on labels which are incrementally disturbed. The performance is measured by mAP normalized to the performance trained on the undisturbed dataset. The network is much more robust against random labeling errors (drawn from a Gaussian distribution with variance σ) than biased labeling (all labels shifted by σ) cf. [193], [194]. (b) An illustration of the random labeling noises and labeling biases (all bounding boxes are shifted in the upper-right direction).

Well-calibrated sensors are the prerequisite for accurate and robust multi-modal perception systems. However, the sensor setup is usually not perfect. Temporal and spatial sensing misalignments might occur while recording the training data or deploying the perception modules. This could cause severe errors in training datasets and degrade the performance of networks, especially for those which are designed to implicitly learn the sensor alignment (e.g. networks that fuse LiDAR BEV feature maps and front view camera images cf. Sec. V-A3). Interestingly, several works propose to calibrate sensors by deep neural networks: Giering *et al.* [196] discretize the spatial misalignments between LiDAR and visual camera into nine classes, and build a network to classify misalignment taking LiDAR and RGB images as inputs; Schneider *et al.* [197] propose to fully regress the extrinsic calibration parameters between LiDAR and visual camera by deep learning. Several multi-modal CNN networks are trained on different de-calibration ranges to iteratively refine the calibration output. In this way, the feature extraction, feature matching, and global optimization problems for sensor registration could be solved in an end-to-end fashion.

B. Fusion Methodology

1) What to Fuse: Most reviewed methods combine RGB images with thermal images or LiDAR 3D points. The networks are trained and evaluated on open datasets such as KITTI [6] and KAIST Pedestrian [92]. These methods do not specifically focus on sensor redundancy, e.g. installing multiple cameras on a driverless car to increase the reliability of perception systems even when some sensors are defective. How to fuse the sensing information from multiple sensors (e.g. RGB images from multiple cameras) is an important open question.

Another challenge is how to represent and process different sensing modalities appropriately before feeding them into a fusion network. For instance, many approaches exist to represent LiDAR point clouds, including 3D voxels, 2D BEV maps, spherical maps, as well as sparse or dense depth maps (more details cf. Sec. V-A). However, only Pfeuffer and Dietmayer [110] have studied the pros and cons for several LiDAR front-view representations. We expect more works to compare different 3D point representation methods.

In addition, there are very few studies for fusing LiDAR and camera outputs with signals from other sources such as Radars, ultrasonics or V2X communication. Radar data differs from LiDAR data and it requires different network architecture and fusion schmes. So far, we are not aware of any work fusing Ultrasonic sensor signals in deep multi-modal perception, despite its relevance for low-speed scenarios. How to fuse these sensing modalities and align them temporally and spatially are big challenges.

Finally, it is an interesting topic to combine physical constraints and model-based approaches with data-driven neural networks. For example, Ramos *et al.* [198] propose to fuse semantics and geometric cues in a Bayesian framework for unexpected objects detections. The semantics are predicted by a FCN network, whereas the geometric cues are provided by model-based stereo detections. The multi-task learning scheme also helps to add physical constraints in neural networks. For example, to aid 3D object detection task, Liang *et al.* [115] design a fusion network that additionally estimate LiDAR ground plane and camera image depth. The ground plane estimation provides useful cues for object locations, while the image depth completion contributes to better cross-modal representation; Panoptic segmentation [47] aims to achieve complete scene understanding by jointly doing semantic segmentation and instance segmentation.

2) How to Fuse: Explicitly modeling uncertainty or informativeness of each sensing modality is important safe autonomous driving. As an example, a multi-modal perception system should show higher uncertainty against adverse weather or detect unseen driving environments (open-world problem). It should also reflect sensor's degradation or defects as well. The perception uncertainties need to be propagated to other modules such as motion planning [199] so that the autonomous vehicles can behave accordingly. Reliable uncertainty estimation can show the networks' robustness (cf. Fig 12). However, most reviewed papers only fuse multiple sensing modalities by a simple operation (e.g. addition and average mean, cf. Sec. V-B). Those methods are designed to achieve high average precision (AP) without considering the networks' robustness. The recent work by Bijelic *et al.* [111] uses dropout to increase the network robustness in foggy images. Specifically, they add pixel-wise dropout masks in different fusion layers so that the network randomly drops LiDAR or camera channels during training. Despite promising results for detections in foggy weather, their method cannot express which sensing modality is more reliable given the distorted sensor inputs. To the best of our knowledge, only the gating network (cf. Sec. V-B) explicitly models the informativeness of each sensing modality.

One way to estimate uncertainty and to increase network robustness is Bayesian Neural Networks (BNNs). They assume a prior distribution over the network weights and infer the posterior distribution to extract the prediction probability [200]. There are two types of uncertainties BNNs can model. *Epistemic uncertainty* illustrates the models' uncertainty when describing the training dataset. It can be obtained by

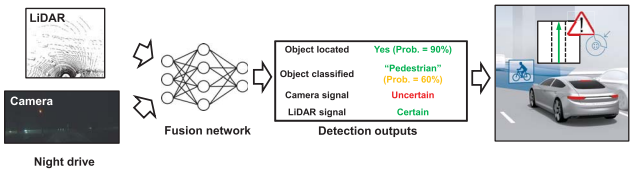


Fig. 12. The importance of explicitly modeling and propagating uncertainties in a multi-modal object detection network. Ideally, the network should produce reliable prediction probabilities (object classification and localization). It should e.g. depict high uncertainty for camera signals during a night drive. Such uncertainty information is useful for the decision making modules, such as maneuver planning or emergency braking systems.

estimating the weight posterior by variational inference [201], sampling [202]–[204], batch normalization [205], or noise injection [206]. It has been applied to semantic segmentation [207] and open-world object detection problems [208], [209]. *Aleatoric uncertainty* represents observation noises inherent in sensors. It can be estimated by the observation likelihood such as a Gaussian distribution or Laplacian distribution. Kendall and Gal [210] study both uncertainties for semantic segmentation; Ilg *et al.* [211] propose to extract uncertainties for optical flow; Feng *et al.* [212] examine the epistemic and aleatoric uncertainties in a LiDAR vehicle detection network for autonomous driving. They show that the uncertainties encode very different information. In the successive work, [213] employ aleatoric uncertainties in a 3D object detection network to significantly improve its detection performance and increase its robustness against noisy data. Other works that introduce aleatoric uncertainties in object detectors include [214]–[217]. Although much progress has been made for BNNs, to the best of our knowledge, so far they have not been introduced to multi-modal perception. Furthermore, few works have been done to propagate uncertainties in object detectors and semantic segmentation networks to other modules, such as tracking and motion planning. How to employ these uncertainties to improve the robustness of an autonomous driving system is a challenging open question.

Another way that can increase the networks' robustness is generative models. In general, generative models aim at modeling the data distribution in an unsupervised way as well as generating new samples with some variations. Variational Autoencoders (VAEs) [218] and Generative Adversarial Networks (GANs) [219] are the two most popular deep generative models. They have been widely applied to image analysis [220]–[222], and recently introduced to model Radar data [223] and road detection [224] for autonomous driving. Generative models could be useful for multi-modal perception problems. For example, they might generate labeled simulated sensor data, when it is tedious and difficult to collect in the real world; they could also serve to detect situations where sensors are defect or an autonomous car is driving into a new scenario that differs from those seen during training. Designing specific fusion operations for deep generative models is an interesting open question.

3) When to Fuse: As discussed in Sec. V-C, the choice of when to fuse the sensing modalities in the reviewed works is mainly based on intuition and empirical results. There is no conclusive evidence that one fusion scheme is better than the others. Ideally, the "optimal" fusion architecture should be found automatically instead of by meticulous engineering. Neural network structure search can potentially solve the problem. It aims at finding the optimal number of neurons and layers in a neural network. Many approaches have been proposed, including the bottom-up construction approach [225], pruning [226], Bayesian optimization [227], genetic algorithms [228], and the recent reinforcement learning approach [229]. Another way to optimize the network structure is by regularization, such as l_1 regularization [230] and stochastic regularization [231], [232].

Furthermore, visual analytics techniques could be employed for network architecture design. Such visualization tools can help to understand and analyze how networks behave, to diagnose the problems, and finally to improve the network architecture. Several methods have been proposed for understanding CNNs for image classification [233], [234]. So far, there has been no research on visual analytics for deep multi-modal learning problems.

4) Real-time Consideration: Deep multi-modal neural networks should perceive driving environments in real-time. Therefore, computational costs and memory requirements should be considered when developing the fusion methodology. At the "what to fuse" level, sensing modalities should be represented in an efficient way. At the "how to fuse" level, finding fusion operations that are suitable for network acceleration, such as pruning and quantization [235]–[237], is an interesting future work. At the "when to fuse" level, inference time and memory constraints can be considered as regularization term for network architecture optimization.

It is difficult to compare the inference speed among the methods we have reviewed, as there is no benchmark with standard hardware or programming languages. Tab. II and Tab. III in the appendix summarize the inference speed of several object detection and semantic segmentation networks on the KITTI test set. Each method uses different hardware, and the inference time is reported only by the authors. It is an open question how these methods perform when they are deployed on automotive hardware.

C. Others

1) Evaluation Metrics: The common way to evaluate object detection methods is mean average precision (mAP) [6], [238]. It is the mean value of average precision (AP) over object classes, given a certain intersection over union (IoU) threshold defined as the geometric overlap between predictions and ground truths. As for the pixel-level semantic segmentation, metrics such as average precision, false positive rate, false negative rate, and IoU calculated at pixel level [57] are often used. However, these metrics only summarize the prediction *accuracy* to a test dataset. They do not consider how sensor behaves in different situations. As an example, to evaluate the performance of a multi-modal network, the IoU thresholds should depend on object distance, occlusion, and types of sensors.

Furthermore, common evaluation metrics are not designed specifically to illustrate how the algorithm handles open-set

conditions or in situations where some sensors are degraded or defective. There exist several metrics to evaluate the quality of predictive uncertainty, e.g. empirical calibration curves [239] and log predictive probabilities. The detection error [240] measures the effectiveness of a neural network in distinguishing in- and out-of-distribution data. The Probability-based Detection Quality (PDQ) [241] is designed to measure the object detection performance for spatial and semantic uncertainties. These metrics can be adapted to the multi-modal perception problems to compare the networks' robustness.

2) More Network Architectures: Most reviewed methods are based on CNN architectures for single frame perception. The predictions in a frame are not dependent on previous frames, resulting in inconsistency over time. Only a few works incorporate temporal cues (e.g. [121], [242]). Future work is expected to develop multi-modal perception algorithms that can handle time series, e.g. via Recurrent Neural Networks. Furthermore, current methods are designed to propagate results to other modules in autonomous driving, such as localization, planning, and reasoning. While the modular approach is the common pipeline for autonomous driving, some works also try to map the sensor data directly to the decision policy such as steering angles or pedal positions (end-to-end learning) [243]–[245], or to some intermediate environment representations (direct-perception) [246], [247]. Multi-modal end-to-end learning and direct perception can be potential research directions as well.

VII. Conclusion and Discussion

We have presented our survey for deep multi-modal object detection and segmentation applied to autonomous driving. We have provided a summary of both multi-modal datasets and fusion methodologies, considering "what to fuse", "how to fuse", and "when to fuse". We have also discussed challenges and open questions. Furthermore, our interactive online tool allows readers to navigate topics and methods for each reference. We plan to frequently update this tool. Despite the fact that an increasing number of multi-modal datasets have been published, most of them record data from RGB cameras, thermal cameras, and LiDARs. Correspondingly, most of the papers we reviewed fuse RGB images either with thermal images or with LiDAR point clouds. Only recently has the fusion of Radar data been investigated. This includes nuScene dataset [88], the Oxford Radar RobotCar Dataset [84], the Astyx HiRes2019 Dataset [93], and the seminal work from Chadwick *et al.* [133] that proposes to fuse RGB camera images with Radar points for vehicle detection. In the future, we expect more datasets and fusion methods concerning Radar signals. There are various ways to fuse sensing modalities in neural networks, encompassing different sensor representations, cf. Sec. V-A, fusion operations cf. Sec. V-B, and fusion stages, cf. Sec. V-C. However, we do not find conclusive evidence that one fusion method is better than the others. Additionally, there is a lack of research on multi-modal perception in open-set conditions or with sensor failures. We expect more focus on these challenging research topics.

Acknowledgment

The authors would like to thank F. Duffhauss for collecting literature and reviewing the paper. They also like to thank Bill Beluch, Rainer Stal, Peter Möller and Ulrich Michael for their suggestions and inspiring discussions.

References

[1] E. Dickmanns and B. Mysliwetz, "Recursive 3-D road and relative ego-state recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 14, no. 2, pp. 199–213, Feb. 1992.

[2] C. Urmson *et al.*, "Autonomous driving in urban environments: Boss and the urban challenge," *J. Field Robot.*, vol. 25, no. 8, pp. 425–466, 2008.

[3] R. Berger. *Autonomous Driving*. Think Act. Accessed: 2014. [Online]. Available: http://www.rolandberger.ch/media/pdf/Roland_Berger_TABAutonomousDriving%final20141211

[4] G. Neuhold, T. Ollmann, S. R. Bulo, and P. Kontschieder, "The mapillary vistas dataset for semantic understanding of street scenes," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 5000–5009.

[5] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, p. 436, 2015.

[6] 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.

[7] H. Yin and C. Berger, "When to use what data set for your self-driving car algorithm: An overview of publicly available driving datasets," in *Proc. IEEE 20th Int. Conf. Intell. Transp. Syst. (ITSC)*, Oct. 2017, pp. 1–8.

[8] D. Ramachandram and G. W. Taylor, "Deep multimodal learning: A survey on recent advances and trends," *IEEE Signal Process. Mag.*, vol. 34, no. 6, pp. 96–108, Nov. 2017.

[9] J. Janai, F. Güney, A. Behl, and A. Geiger, "Computer vision for autonomous vehicles: Problems, datasets and state of the art," 2017, *arXiv:1704.05519*. [Online]. Available: https://arxiv.org/abs/1704.05519

[10] 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.

[11] L. Liu *et al.*, "Deep learning for generic object detection: A survey," 2018, *arXiv:1809.02165*. [Online]. Available: http://arxiv.org/abs/1809.02165

[12] A. Garcia-Garcia, S. Orts-Escolano, S. Oprea, V. Villena-Martinez, and J. Garcia-Rodriguez, "A review on deep learning techniques applied to semantic segmentation," 2017, *arXiv:1704.06857*. [Online]. Available: https://arxiv.org/abs/1704.06857

[13] K. Bengler, K. Dietmayer, B. Farber, M. Maurer, C. Stiller, and H. Winner, "Three decades of driver assistance systems: Review and future perspectives," *IEEE Intell. Transp. Syst. Mag.*, vol. 6, no. 4, pp. 6–22, Oct. 2014.

[14] Waymo. (2017). *Waymo Safety Report: On the Road to Fully Self-Driving*. [Online]. Available: https://waymo.com/safety

[15] M. Aeberhard *et al.*, "Experience, results and lessons learned from automated driving on Germany's highways," *IEEE Intell. Transp. Syst. Mag.*, vol. 7, no. 1, pp. 42–57, Jan. 2015.

[16] J. Ziegler *et al.*, "Making bertha drive—An autonomous journey on a historic route," *IEEE Intell. Transp. Syst. Mag.*, vol. 6, no. 2, pp. 8–20, Apr. 2014.

[17] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. *The PASCAL Visual Object Classes Challenge 2007 (VOC2007) Results*. [Online]. Available: http://www.pascal-network.org/challenges/VOC/voc2007/workshop/index.html

[18] T.-Y. Lin *et al.*, "Microsoft COCO: Common objects in context," in *Proc. Eur. Conf. Comput. Vis.* Zürich, Switzerland: Springer, 2014, pp. 740–755.

[19] M. Weber, P. Wolf, and J. M. Zollner, "DeepTLR: A single deep convolutional network for detection and classification of traffic lights," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Jun. 2016, pp. 342–348.

[20] J. Muller and K. Dietmayer, "Detecting traffic lights by single shot detection," in *Proc. 21st Int. Conf. Intell. Transp. Syst. (ITSC)*, Nov. 2018, pp. 342–348.

[21] M. Bach, S. Reuter, and K. Dietmayer, "Multi-camera traffic light recognition using a classifying Labeled Multi-Bernoulli filter," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Jun. 2017, pp. 1045–1051.

[22] K. Behrendt, L. Novak, and R. Botros, "A deep learning approach to traffic lights: Detection, tracking, and classification," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2017, pp. 1370–1377.

[23] Z. Zhu, D. Liang, S. Zhang, X. Huang, B. Li, and S. Hu, "Traffic-sign detection and classification in the wild," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 2110–2118.

[24] H. S. Lee and K. Kim, "Simultaneous traffic sign detection and boundary estimation using convolutional neural network," *IEEE Trans. Intell. Transp. Syst.*, vol. 19, no. 5, pp. 1652–1663, May 2018.

[25] H. Luo, Y. Yang, B. Tong, F. Wu, and B. Fan, "Traffic sign recognition using a multi-task convolutional neural network," *IEEE Trans. Intell. Transp. Syst.*, vol. 19, no. 4, pp. 1100–1111, Apr. 2018.

[26] S. Zhang, R. Benenson, M. Omran, J. Hosang, and B. Schiele, "Towards reaching human performance in pedestrian detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 4, pp. 973–986, Apr. 2018.

[27] L. Zhang, L. Lin, X. Liang, and K. He, "Is faster R-CNN doing well for pedestrian detection?" in *Proc. Eur. Conf. Comput. Vis.* Amsterdam, The Netherlands: Springer, 2016, pp. 443–457.

[28] X. Chen, K. Kundu, Y. Zhu, H. Ma, S. Fidler, and R. Urtasun, "3D object proposals using stereo imagery for accurate object class detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 5, pp. 1259–1272, May 2018.

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

[30] B. Li, T. Zhang, and T. Xia, "Vehicle detection from 3D lidar using fully convolutional network," in *Proc. Robot., Sci. Syst.*, Jun. 2016, pp. 1–8.

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

[32] J. Fang, Y. Zhou, Y. Yu, and S. Du, "Fine-grained vehicle model recognition using a coarse-to-fine convolutional neural network architecture," *IEEE Trans. Intell. Transp. Syst.*, vol. 18, no. 7, pp. 1782–1792, Jul. 2017.

[33] A. Mousavian, D. Anguelov, J. Flynn, and J. Kosecka, "3D bounding box estimation using deep learning and geometry," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 5632–5640.

[34] P. Sermanet, D. Eigen, X. Zhang, M. Mathieu, R. Fergus, and Y. LeCun, "OverFeat: Integrated recognition, localization and detection using convolutional networks," in *Proc. Int. Conf. Learn. Representations*, 2013, pp. 1–16.

[35] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 580–587.

[36] K. He, X. Zhang, S. Ren, and J. Sun, "Spatial pyramid pooling in deep convolutional networks for visual recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 9, pp. 1904–1916, Jan. 2015.

[37] R. Girshick, "Fast R-CNN," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 1440–1448.

[38] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014, *arXiv:1409.1556*. [Online]. Available: http://arxiv.org/abs/1409.1556

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

[40] C. Szegedy *et al.*, "Going deeper with convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 1–9.

[41] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 6, pp. 1137–1149, Jun. 2017.

[42] J. Dai, Y. Li, K. He, and J. Sun, "R-FCN: Object detection via region-based fully convolutional networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2016, pp. 379–387.

[43] C. Szegedy, A. Toshev, and D. Erhan, "Deep neural networks for object detection," in *Proc. Adv. Neural Inf. Process. Syst.*, 2013, pp. 2553–2561.

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

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

[46] J. Huang *et al.*, "Speed/accuracy trade-offs for modern convolutional object detectors," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, vol. 4, Jul. 2017, pp. 7310–7311.

[47] A. Kirillov, K. He, R. Girshick, C. Rother, and P. Dollar, "Panoptic segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 9404–9413.

[48] A. Kirillov, R. Girshick, K. He, and P. Dollar, "Panoptic feature pyramid networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 6399–6408.

[49] Y. Xiong *et al.*, "UPSNet: A unified panoptic segmentation network," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 8818–8826.

[50] L. Porzi, S. R. Bulo, A. Colovic, and P. Kontschieder, "Seamless scene segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 8277–8286.

[51] B. Wu, A. Wan, X. Yue, and K. Keutzer, "SqueezeSeg: Convolutional neural nets with recurrent CRF for real-time road-object segmentation from 3D LiDAR point cloud," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2018, pp. 1887–1893.

[52] L. Caltagirone, S. Scheidegger, L. Svensson, and M. Wahde, "Fast LIDAR-based road detection using fully convolutional neural networks," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Jun. 2017, pp. 1019–1024.

[53] Q. Huang, W. Wang, and U. Neumann, "Recurrent slice networks for 3D segmentation of point clouds," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 2626–2635.

[54] A. Dewan, G. L. Oliveira, and W. Burgard, "Deep semantic classification for 3D LiDAR data," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Sep. 2017, pp. 3544–3549.

[55] A. Dewan and W. Burgard, "DeepTemporalSeg: Temporally consistent semantic segmentation of 3D LiDAR scans," 2019, *arXiv:1906.06962*. [Online]. Available: http://arxiv.org/abs/1906.06962

[56] A. Milioto, I. Vizzo, J. Behley, and C. Stachniss, "RangeNet++: Fast and accurate LiDAR semantic segmentation," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Nov. 2019, p. 1.

[57] M. Cordts *et al.*, "The cityscapes dataset for semantic urban scene understanding," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 3213–3223.

[58] S. Wang *et al.*, "TorontoCity: Seeing the world with a million eyes," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 3028–3036.

[59] X. Huang *et al.*, "The ApolloScape dataset for autonomous driving," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2018, pp. 954–960.

[60] L. Schneider *et al.*, "Multimodal neural networks: RGB-D for semantic segmentation and object detection," in *Proc. Scandin. Conf. Image Anal.* Tromsø, Norway: Springer, 2017, pp. 98–109.

[61] V. Badrinarayanan, A. Kendall, and R. Cipolla, "SegNet: A deep convolutional encoder-decoder architecture for image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 12, pp. 2481–2495, Dec. 2017.

[62] M. Teichmann, M. Weber, M. Zollner, R. Cipolla, and R. Urtasun, "MultiNet: Real-time joint semantic reasoning for autonomous driving," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Jun. 2018, pp. 1013–1020.

[63] K. He, G. Gkioxari, P. Dollar, and R. Girshick, "Mask R-CNN," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 2980–2988.

[64] J. Uhrig, E. Rehder, B. Frohlich, U. Franke, and T. Brox, "Box2Pix: Single-shot instance segmentation by assigning pixels to object boxes," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Jun. 2018, pp. 292–299.

[65] S. Gupta, R. Girshick, P. Arbeláez, and J. Malik, "Learning rich features from RGB-D images for object detection and segmentation," in *Proc. Eur. Conf. Comput. Vis.* Zürich, Switzerland: Springer, 2014, pp. 345–360.

[66] B. Hariharan, P. Arbeláez, R. Girshick, and J. Malik, "Simultaneous detection and segmentation," in *Proc. Eur. Conf. Comput. Vis.* Zürich, Switzerland: Springer, 2014, pp. 297–312.

[67] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 3431–3440.

[68] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "DeepLab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 4, pp. 834–848, Apr. 2018.

[69] A. Paszke, A. Chaurasia, S. Kim, and E. Culurciello, "ENet: A deep neural network architecture for real-time semantic segmentation," 2016, *arXiv:1606.02147*. [Online]. Available: http://arxiv.org/abs/1606.02147

[70] A. Roy and S. Todorovic, "A multi-scale CNN for affordance segmentation in RGB images," in *Proc. Eur. Conf. Comput. Vis.* Amsterdam, The Netherlands: Springer, 2016, pp. 186–201.

[71] S. Zheng *et al.*, "Conditional random fields as recurrent neural networks," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 1529–1537.

[72] M. Siam, M. Gamal, M. Abdel-Razek, S. Yogamani, M. Jagersand, and H. Zhang, "A comparative study of real-time semantic segmentation for autonomous driving," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2018, pp. 587–597.

[73] W. Maddern, G. Pascoe, C. Linegar, and P. Newman, "1 year, 1000 km: The Oxford RobotCar dataset," *Int. J. Robot. Res.*, vol. 36, no. 1, pp. 3–15, Jan. 2017.

[74] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun, "Vision meets robotics: The KITTI dataset," *Int. J. Robot. Res.*, vol. 32, no. 11, pp. 1231–1237, Sep. 2013.

[75] J.-L. Blanco-Claraco, F.-Á. Moreno-Dueñas, and J. González-Jiménez, "The Málaga urban dataset: High-rate stereo and LiDAR in a realistic urban scenario," *Int. J. Robot. Res.*, vol. 33, no. 2, pp. 207–214, Feb. 2014.

[76] H. Jung, Y. Oto, O. M. Mozos, Y. Iwashita, and R. Kurazume, "Multi-modal panoramic 3D outdoor datasets for place categorization," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Oct. 2016, pp. 4545–4550.

[77] Y. Chen *et al.*, "LiDAR-video driving dataset: Learning driving policies effectively," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 5870–5878.

[78] A. Patil, S. Malla, H. Gang, and Y.-T. Chen, "The H3D dataset for full-surround 3D multi-object detection and tracking in crowded urban scenes," in *Proc. Int. Conf. Robot. Autom. (ICRA)*, May 2019, pp. 9552–9557.

[79] J. Xue *et al.*, "BLVD: Building a large-scale 5D semantics benchmark for autonomous driving," in *Proc. Int. Conf. Robot. Autom. (ICRA)*, May 2019, pp. 6685–6691.

[80] R. Kesten. (2019). *Lyft Level 5 AV Dataset 2019*. [Online]. Available: https://level5.lyft.com/dataset/

[81] M.-F. Chang *et al.*, "Argoverse: 3D tracking and forecasting with rich maps," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 8748–8757.

[82] (2019). *PandaSet: Public Large-Scale Dataset for Autonomous Driving*. [Online]. Available: https://scale.com/open-datasets/pandaset

[83] (2019). *Waymo Open Dataset: An Autonomous Driving Dataset*. [Online]. Available: https://www.waymo.com/open

[84] D. Barnes, M. Gadd, P. Murcutt, P. Newman, and I. Posner, "The Oxford radar RobotCar Dataset: A radar extension to the Oxford RobotCar dataset," 2019, *arXiv:1909.01300*. [Online]. Available: http://arxiv.org/abs/1909.01300

[85] Q.-H. Pham *et al.*, "A*3D Dataset: Towards autonomous driving in challenging environments," 2019, *arXiv:1909.07541*. [Online]. Available: http://arxiv.org/abs/1909.07541

[86] J. Geyer (2019) *A2D2: AEV Autonomous Driving Dataset*. [Online]. Available: https://www.audi-electronics-venture.de/aev/web/en/driving-dataset.html

[87] M. Braun, S. Krebs, F. Flohr, and D. M. Gavrila, "EuroCity persons: A novel benchmark for person detection in traffic scenes," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 8, pp. 1844–1861, Aug. 2019.

[88] H. Caesar *et al.*, "NuScenes: A multimodal dataset for autonomous driving," 2019, *arXiv:1903.11027*. [Online]. Available: http://arxiv.org/abs/1903.11027

[89] S. Hwang, J. Park, N. Kim, Y. Choi, and I. S. Kweon, "Multispectral pedestrian detection: Benchmark dataset and baseline," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 1037–1045.

[90] K. Takumi, K. Watanabe, Q. Ha, A. Tejero-De-Pablos, Y. Ushiku, and T. Harada, "Multispectral object detection for autonomous vehicles," in *Proc. Thematic Workshops ACM Multimedia Thematic Workshops*, 2017, pp. 35–43.

[91] Q. Ha, K. Watanabe, T. Karasawa, Y. Ushiku, and T. Harada, "MFNet: Towards real-time semantic segmentation for autonomous vehicles with multi-spectral scenes," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Sep. 2017, pp. 5108–5115.

[92] Y. Choi *et al.*, "KAIST multi-spectral day/night data set for autonomous and assisted driving," *IEEE Trans. Intell. Transp. Syst.*, vol. 19, no. 3, pp. 934–948, Mar. 2018.

[93] M. Meyer and G. Kuschk, "Automotive radar dataset for deep learning based 3D object detection," in *Proc. 16th Eur. Radar Conf.*, 2019, pp. 129–132.

[94] D. Kondermann *et al.*, "Stereo ground truth with error bars," in *Proc. 12th Asian Conf. Comput. Vis.* Singapore: Springer, 2014, pp. 595–610.

[95] M. M. Larsson, E. Stenborg, L. Hammarstrand, M. Pollefeys, T. Sattler, and F. Kahl, "A cross-season correspondence dataset for robust semantic segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 9532–9542.

[96] T. Sattler *et al.*, "Benchmarking 6DOF outdoor visual localization in changing conditions," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 8601–8610.

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

[98] A. Asvadi, L. Garrote, C. Premebida, P. Peixoto, and U. J. Nunes, "Multimodal vehicle detection: Fusing 3D-LIDAR and color camera data," *Pattern Recognit. Lett.*, vol. 115, pp. 20–29, Nov. 2018.

[99] S.-I. Oh and H.-B. Kang, "Object detection and classification by decision-level fusion for intelligent vehicle systems," *Sensors*, vol. 17, no. 12, p. 207, Jan. 2017.

[100] J. Schlosser, C. K. Chow, and Z. Kira, "Fusing LIDAR and images for pedestrian detection using convolutional neural networks," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2016, pp. 2198–2205.

[101] Z. Wang, W. Zhan, and M. Tomizuka, "Fusing Bird's eye view LIDAR point cloud and front view camera image for 3D object detection," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Jun. 2018, pp. 1–6.

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

[103] D. Xu, D. Anguelov, and A. Jain, "PointFusion: Deep sensor fusion for 3D bounding box estimation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 244–253.

[104] 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/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 918–927.

[105] X. Du, M. H. Ang, and D. Rus, "Car detection for autonomous vehicle: LIDAR and vision fusion approach through deep learning framework," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Sep. 2017, pp. 749–754.

[106] X. Du, M. H. Ang, S. Karaman, and D. Rus, "A general pipeline for 3D detection of vehicles," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2018, pp. 3194–3200.

[107] D. Matti, H. K. Ekenel, and J.-P. Thiran, "Combining LiDAR space clustering and convolutional neural networks for pedestrian detection," in *Proc. 14th IEEE Int. Conf. Adv. Video Signal Based Surveill. (AVSS)*, Aug. 2017, pp. 1–6.

[108] T. Kim and J. Ghosh, "Robust detection of non-motorized road users using deep learning on optical and LIDAR data," in *Proc. IEEE 19th Int. Conf. Intell. Transp. Syst. (ITSC)*, Nov. 2016, pp. 271–276.

[109] J. Kim, J. Koh, Y. Kim, J. Choi, Y. Hwang, and J. W. Choi, "Robust deep multi-modal learning based on gated information fusion network," in *Proc. Asian Conf. Comput. Vis.*, 2018, pp. 90–106.

[110] A. Pfeuffer and K. Dietmayer, "Optimal sensor data fusion architecture for object detection in adverse weather conditions," in *Proc. 21st Int. Conf. Inf. Fusion (FUSION)*, Jul. 2018, pp. 2592–2599.

[111] M. Bijelic, F. Mannan, T. Gruber, W. Ritter, K. Dietmayer, and F. Heide, "Seeing through fog without seeing fog: Deep sensor fusion in the absence of labeled training data," in *Proc. IEEE Conf. Computer Vision*, 2019, pp. 1–11.

[112] V. A. Sindagi, Y. Zhou, and O. Tuzel, "MVX-Net: Multimodal VoxelNet for 3D object detection," in *Proc. Int. Conf. Robot. Autom. (ICRA)*, May 2019, pp. 7276–7282.

[113] J. Dou, J. Xue, and J. Fang, "SEG-VoxelNet for 3D vehicle detection from RGB and LiDAR data," in *Proc. Int. Conf. Robot. Autom. (ICRA)*, May 2019, pp. 4362–4368.

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

[115] 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.

[116] J. Wagner, V. Fischer, M. Herman, and S. Behnke, "Multispectral pedestrian detection using deep fusion convolutional neural networks," in *Proc. 24th Eur. Symp. Artif. Neural Netw., Comput. Intell. Mach. Learn.*, 2016, pp. 509–514.

[117] J. Liu, S. Zhang, S. Wang, and D. Metaxas, "Multispectral deep neural networks for pedestrian detection," in *Proc. Procedings Brit. Mach. Vis. Conf.*, Sep. 2016, pp. 73.1–73.13.

[118] D. Guan, Y. Cao, J. Yang, Y. Cao, and M. Y. Yang, "Fusion of multispectral data through illumination-aware deep neural networks for pedestrian detection," *Inf. Fusion*, vol. 50, pp. 148–157, Oct. 2019.

[119] O. Mees, A. Eitel, and W. Burgard, "Choosing smartly: Adaptive multimodal fusion for object detection in changing environments," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Oct. 2016, pp. 151–156.

[120] B. Yang, M. Liang, and R. Urtasun, "HDNET: Exploiting HD maps for 3D object detection," in *Proc. 2nd Annu. Conf. Robot Learn.*, 2018, pp. 146–155.

[121] S. Casas, W. Luo, and R. Urtasun, "IntentNet: Learning to predict intention from raw sensor data," in *Proc. 2nd Annu. Conf. Robot Learn.*, 2018, pp. 947–956.

[122] D.-K. Kim, D. Maturana, M. Uenoyama, and S. Scherer, "Season-invariant semantic segmentation with a deep multimodal network," in *Field and Service Robotics*. Zürich, Switzerland: Springer, 2018, pp. 255–270.

[123] Y. Sun, W. Zuo, and M. Liu, "RTFNet: RGB-thermal fusion network for semantic segmentation of urban scenes," *IEEE Robot. Autom. Lett.*, vol. 4, no. 3, pp. 2576–2583, Jul. 2019.

[124] A. Valada, G. L. Oliveira, T. Brox, and W. Burgard, "Deep multispectral semantic scene understanding of forested environments using multimodal fusion," in *Proc. Int. Symp. Experim. Robot.* Tokyo, Japan: Springer, 2016, pp. 465–477.

[125] A. Valada, J. Vertens, A. Dhall, and W. Burgard, "AdapNet: Adaptive semantic segmentation in adverse environmental conditions," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2017, pp. 4644–4651.

[126] A. Valada, R. Mohan, and W. Burgard, "Self-supervised model adaptation for multimodal semantic segmentation," *Int. J. Comput. Vis.*, vol. 127, pp. 1–47, Jul. 2019.

[127] F. Yang, J. Yang, Z. Jin, and H. Wang, "A fusion model for road detection based on deep learning and fully connected CRF," in *Proc. 13th Annu. Conf. Syst. Syst. Eng. (SoSE)*, Jun. 2018, pp. 29–36.

[128] L. Caltagirone, M. Bellone, L. Svensson, and M. Wahde, "LIDAR–camera fusion for road detection using fully convolutional neural networks," *Robot. Auto. Syst.*, vol. 111, pp. 125–131, Jan. 2019.

[129] X. Lv, Z. Liu, J. Xin, and N. Zheng, "A novel approach for detecting road based on two-stream fusion fully convolutional network," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Jun. 2018, pp. 1464–1469.

[130] F. Wulff, B. Schaufele, O. Sawade, D. Becker, B. Henke, and I. Radusch, "Early fusion of camera and Lidar for robust road detection based on U-Net FCN," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Jun. 2018, pp. 1426–1431.

[131] Z. Chen, J. Zhang, and D. Tao, "Progressive LiDAR adaptation for road detection," *IEEE/CAA J. Autom. Sinica*, vol. 6, no. 3, pp. 693–702, May 2019.

[132] F. Piewak *et al.*, "Boosting lidar-based semantic labeling by cross-modal training data generation," in *Proc. Workshop Eur. Conf. Comput. Vis.*, 2018, pp. 497–513.

[133] S. Chadwick, W. Maddern, and P. Newman, "Distant vehicle detection using radar and vision," in *Proc. Int. Conf. Robot. Autom. (ICRA)*, May 2019, pp. 8311–8317.

[134] 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.

[135] 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.

[136] S. Shi, Z. Wang, X. Wang, and H. Li, "From points to parts: 3D object detection from point cloud with part-aware and part-aggregation network," 2019, *arXiv:1907.03670*. [Online]. Available: https://arxiv.org/abs/1907.03670

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

[138] R. Q. Charles, H. Su, M. Kaichun, and L. J. Guibas, "PointNet: Deep learning on point sets for 3D classification and segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 77–85.

[139] C. R. Qi, L. Yi, H. Su, and L. J. Guibas, "PointNet++: Deep hierarchical feature learning on point sets in a metric space," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 5099–5108.

[140] K. Shin, Y. P. Kwon, and M. Tomizuka, "RoarNet: A robust 3D object detection based on RegiOn approximation refinement," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Jun. 2019, pp. 2510–2515.

[141] S. Wang, S. Suo, W.-C. Ma, A. Pokrovsky, and R. Urtasun, "Deep parametric continuous convolutional neural networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 2589–2597.

[142] Y. Li, R. Bu, M. Sun, W. Wu, X. Di, and B. Chen, "PointCNN: Convolution on χ-transformed points," in *Proc. Adv. Neural Inf. Process. Syst.*, 2018, pp. 826–836.

[143] A. Asvadi, L. Garrote, C. Premebida, P. Peixoto, and U. J. Nunes, "DepthCN: Vehicle detection using 3D-LIDAR and ConvNet," in *Proc. IEEE 20th Int. Conf. Intell. Transp. Syst. (ITSC)*, Oct. 2017, pp. 1–6.

[144] C. Premebida, L. Garrote, A. Asvadi, A. P. Ribeiro, and U. Nunes, "High-resolution LIDAR-based depth mapping using bilateral filter," in *Proc. IEEE 19th Int. Conf. Intell. Transp. Syst. (ITSC)*, Nov. 2016, pp. 2469–2474.

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

[146] T. Roddick, A. Kendall, and R. Cipolla, "Orthographic feature transform for monocular 3D object detection," in *Proc. Brit. Mach. Vis. Conf.*, 2019, pp. 1–10.

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

[148] Y. You *et al.*, "Pseudo-LiDAR++: Accurate depth for 3D object detection in autonomous driving," 2019, *arXiv:1906.06310*. [Online]. Available: http://arxiv.org/abs/1906.06310

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

[150] K. Werber *et al.*, "Automotive radar gridmap representations," in *IEEE MTT-S Int. Microw. Symp. Dig.*, Apr. 2015, pp. 1–4.

[151] J. Lombacher, M. Hahn, J. Dickmann, and C. Wöhler, "Potential of radar for static object classification using deep learning methods," in *IEEE MTT-S Int. Microw. Symp. Dig.*, Apr. 2016, pp. 1–4.

[152] J. Lombacher, K. Laudt, M. Hahn, J. Dickmann, and C. Wohler, "Semantic radar grids," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Jun. 2017, pp. 1170–1175.

[153] T. Visentin, A. Sagainov, J. Hasch, and T. Zwick, "Classification of objects in polarimetric radar images using CNNs at 77 GHz," in *Proc. IEEE Asia Pacific Microw. Conf. (APMC)*, Nov. 2017, pp. 356–359.

[154] S. Kim, S. Lee, S. Doo, and B. Shim, "Moving target classification in automotive radar systems using convolutional recurrent neural networks," in *Proc. 26th Eur. Signal Process. Conf. (EUSIPCO)*, Sep. 2018, pp. 1482–1486.

[155] M. G. Amin and B. Erol, "Understanding deep neural networks performance for radar-based human motion recognition," in *Proc. IEEE Radar Conf. (RadarConf)*, Apr. 2018, pp. 1461–1465.

[156] O. Schumann, M. Hahn, J. Dickmann, and C. Wohler, "Semantic segmentation on radar point clouds," in *Proc. 21st Int. Conf. Inf. Fusion (FUSION)*, Jul. 2018, pp. 2179–2186.

[157] O. Schumann, C. Wohler, M. Hahn, and J. Dickmann, "Comparison of random forest and long short-term memory network performances in classification tasks using radar," in *Proc. Sensor Data Fusion, Trends, Solutions, Appl. (SDF)*, Oct. 2017, pp. 1–6.

[158] R. A. Jacobs, M. I. Jordan, S. J. Nowlan, and G. E. Hinton, "Adaptive mixtures of local experts," *Neural Comput.*, vol. 3, no. 1, pp. 79–87, Feb. 1991.

[159] D. Eigen, M. Ranzato, and I. Sutskever, "Learning factored representations in a deep mixture of experts," in *Proc. Workshop Int. Conf. Learn. Representations*, 2014, pp. 1–8.

[160] M. Bloesch, J. Czarnowski, R. Clark, S. Leutenegger, and A. J. Davison, "CodeSLAM–learning a compact, optimisable representation for dense visual SLAM," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 2560–2568.

[161] J. Wang, Z. Wei, T. Zhang, and W. Zeng, "Deeply-fused nets," 2016, *arXiv:1605.07716*. [Online]. Available: http://arxiv.org/abs/1605.07716

[162] O. Russakovsky *et al.*, "ImageNet large scale visual recognition challenge," *Int. J. Comput. Vis.*, vol. 115, no. 3, pp. 211–252, Dec. 2015.

[163] J. Ngiam *et al.*, "StarNet: Targeted computation for object detection in point clouds," 2019, *arXiv:1908.11069*. [Online]. Available: http://arxiv.org/abs/1908.11069

[164] A. Gaidon, Q. Wang, Y. Cabon, and E. Vig, "Virtual worlds as proxy for multi-object tracking analysis," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2016, pp. 4340–4349.

[165] S. R. Richter, V. Vineet, S. Roth, and V. Koltun, "Playing for data: Ground truth from computer games," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 102–118.

[166] G. Ros, L. Sellart, J. Materzynska, D. Vazquez, and A. M. Lopez, "The SYNTHIA Dataset: A large collection of synthetic images for semantic segmentation of urban scenes," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 3234–3243.

[167] S. R. Richter, Z. Hayder, and V. Koltun, "Playing for benchmarks," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 2232–2241.

[168] X. Yue, B. Wu, S. A. Seshia, K. Keutzer, and A. L. Sangiovanni-Vincentelli, "A lidar point cloud generator: From a virtual world to autonomous driving," in *Proc. ACM Int. Conf. Multimedia Retr.*, 2018, pp. 458–464.

[169] M. Wrenninge and J. Unger, "Synscapes: A photorealistic synthetic dataset for street scene parsing," 2018, *arXiv:1810.08705*. [Online]. Available: http://arxiv.org/abs/1810.08705

[170] P.-H. Huang, K. Matzen, J. Kopf, N. Ahuja, and J.-B. Huang, "DeepMVS: Learning multi-view stereopsis," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 2821–2830.

[171] D. Griffiths and J. Boehm, "SynthCity: A large scale synthetic point cloud," 2019, *arXiv:1907.04758*. [Online]. Available: http://arxiv.org/abs/1907.04758

[172] A. Dosovitskiy, G. Ros, F. Codevilla, A. Lopez, and V. Koltun, "CARLA: An open urban driving simulator," in *Proc. 1st Annu. Conf. Robot Learn.*, 2017, pp. 1–16.

[173] B. Hurl, K. Czarnecki, and S. Waslander, "Precise synthetic image and LiDAR (PreSIL) dataset for autonomous vehicle perception," 2019, *arXiv:1905.00160*. [Online]. Available: http://arxiv.org/abs/1905.00160

[174] J. Lee, S. Walsh, A. Harakeh, and S. L. Waslander, "Leveraging pre-trained 3D object detection models for fast ground truth generation," in *Proc. 21st Int. Conf. Intell. Transp. Syst. (ITSC)*, Nov. 2018, pp. 2504–2510.

[175] J. Mei, B. Gao, D. Xu, W. Yao, X. Zhao, and H. Zhao, "Semantic segmentation of 3D LiDAR data in dynamic scene using semi-supervised learning," *IEEE Trans. Intell. Transp. Syst.*, to be published.

[176] R. Mackowiak, P. Lenz, O. Ghori, F. Diego, O. Lange, and C. Rother, "CEREALS–cost-effective region-based active learning for semantic segmentation," in *Proc. Brit. Mach. Vis. Conf.*, 2018, pp. 1–21.

[177] S. Roy, A. Unmesh, and V. P. Namboodiri, "Deep active learning for object detection," in *Proc. Brit. Mach. Vis. Conf.*, 2018, p. 91.

[178] D. Feng, X. Wei, L. Rosenbaum, A. Maki, and K. Dietmayer, "Deep active learning for efficient training of a LiDAR 3D object detector," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Jun. 2019, pp. 667–674.

[179] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Trans. Knowl. Data Eng.*, vol. 22, no. 10, pp. 1345–1359, Oct. 2010.

[180] V. M. Patel, R. Gopalan, R. Li, and R. Chellappa, "Visual domain adaptation: A survey of recent advances," *IEEE Signal Process. Mag.*, vol. 32, no. 3, pp. 53–69, May 2015.

[181] Y. Chen, W. Li, X. Chen, and L. Van Gool, "Learning semantic segmentation from synthetic data: A geometrically guided input-output adaptation approach," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 1841–1850.

[182] Y. Chen, W. Li, C. Sakaridis, D. Dai, and L. Van Gool, "Domain adaptive faster R-CNN for object detection in the wild," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 3339–3348.

[183] K.-H. Lee, G. Ros, J. Li, and A. Gaidon, "Spigan: Privileged adversarial learning from simulation," in *Proc. Int. Conf. Learn. Representations*, 2019, pp. 1–14.

[184] J. Tremblay *et al.*, "Training deep networks with synthetic data: Bridging the reality gap by domain randomization," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2018, pp. 969–977.

[185] D. P. Kingma, S. Mohamed, D. J. Rezende, and M. Welling, "Semi-supervised learning with deep generative models," in *Proc. Adv. Neural Inf. Process. Syst.*, 2014, pp. 3581–3589.

[186] G. I. Parisi, R. Kemker, J. L. Part, C. Kanan, and S. Wermter, "Continual lifelong learning with neural networks: A review," *Neural Netw.*, vol. 113, pp. 54–71, May 2019.

[187] Y. Wang *et al.*, "Iterative learning with open-set noisy labels," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 8688–8696.

[188] M. Ren, W. Zeng, B. Yang, and R. Urtasun, "Learning to reweight examples for robust deep learning," in *Proc. Int. Conf. Mach. Learn.*, 2018, pp. 1–13.

[189] L. Jiang, Z. Zhou, T. Leung, L.-J. Li, and L. Fei-Fei, "MentorNet: Learning data-driven curriculum for very deep neural networks on corrupted labels," in *Proc. Int. Conf. Mach. Learn.*, 2018, pp. 2309–2318.

[190] X. Ma *et al.*, "Dimensionality-driven learning with noisy labels," in *Proc. Int. Conf. Mach. Learn.*, 2018, pp. 3361–3370.

[191] A. Zlateski, R. Jaroensri, P. Sharma, and F. Durand, "On the importance of label quality for semantic segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 1479–1487.

[192] P. Meletis and G. Dubbelman, "On boosting semantic street scene segmentation with weak supervision," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Jun. 2019, pp. 1334–1339.

[193] C. Haase-Schutz, H. Hertlein, and W. Wiesbeck, "Estimating labeling quality with deep object detectors," in *Proc. IEEE Intell. Vehicles Symp.*, Jun. 2019, pp. 33–38.

[194] S. Chadwick and P. Newman, "Training object detectors with noisy data," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Jun. 2019, pp. 1319–1325.

[195] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollar, "Focal loss for dense object detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 2, pp. 318–327, Feb. 2020.

[196] M. Giering, V. Venugopalan, and K. Reddy, "Multi-modal sensor registration for vehicle perception via deep neural networks," in *Proc. IEEE High Perform. Extreme Computing Conf. (HPEC)*, Sep. 2015, pp. 1–6.

[197] N. Schneider, F. Piewak, C. Stiller, and U. Franke, "RegNet: Multimodal sensor registration using deep neural networks," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Jun. 2017, pp. 1803–1810.

[198] S. Ramos, S. Gehrig, P. Pinggera, U. Franke, and C. Rother, "Detecting unexpected obstacles for self-driving cars: Fusing deep learning and geometric modeling," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Jun. 2017, pp. 1025–1032.

[199] H. Banzhaf, M. Dolgov, J. Stellet, and J. M. Zollner, "From footprints to beliefprints: Motion planning under uncertainty for maneuvering automated vehicles in dense scenarios," in *Proc. 21st Int. Conf. Intell. Transp. Syst. (ITSC)*, Nov. 2018, pp. 1680–1687.

[200] D. J. C. Mackay, "A practical Bayesian framework for backpropagation networks," *Neural Comput.*, vol. 4, no. 3, pp. 448–472, May 1992.

[201] G. E. Hinton and D. Van Camp, "Keeping the neural networks simple by minimizing the description length of the weights," in *Proc.6th Annu. Conf. Comput. Learn. Theory (COLT)*, 1993, pp. 5–13.

[202] Y. Gal, "Uncertainty in deep learning," Ph.D. dissertation, Univ. Cambridge, Cambridge, U.K., 2016.

[203] A. Graves, "Practical variational inference for neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2011, pp. 2348–2356.

[204] S. Mandt, M. D. Hoffman, and D. M. Blei, "Stochastic gradient descent as approximate Bayesian inference," *J. Mach. Learn. Res.*, vol. 18, no. 1, pp. 4873–4907, 2017.

[205] M. Teye, H. Azizpour, and K. Smith, "Bayesian uncertainty estimation for batch normalized deep networks," in *Proc. Int. Conf. Mach. Learn.*, 2018, pp. 1–30.

[206] J. Postels, F. Ferroni, H. Coskun, N. Navab, and F. Tombari, "Sampling-free epistemic uncertainty estimation using approximated variance propagation," in *Proc. IEEE Conf. Computer Vision*, 2019, pp. 2931–2940.

[207] A. Kendall, V. Badrinarayanan, and R. Cipolla, "Bayesian SegNet: Model uncertainty in deep convolutional encoder-decoder architectures for scene understanding," in *Proc. Brit. Mach. Vis. Conf.*, 2017, pp. 1–11.

[208] D. Miller, L. Nicholson, F. Dayoub, and N. Sunderhauf, "Dropout sampling for robust object detection in open-set conditions," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2018, pp. 1–7.

[209] D. Miller, F. Dayoub, M. Milford, and N. Sunderhauf, "Evaluating merging strategies for sampling-based uncertainty techniques in object detection," in *Proc. Int. Conf. Robot. Autom. (ICRA)*, May 2019, pp. 2348–2354.

[210] A. Kendall and Y. Gal, "What uncertainties do we need in Bayesian deep learning for computer vision?" in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 5574–5584.

[211] E. Ilg *et al.*, "Uncertainty estimates and multi-hypotheses networks for optical flow," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 652–667.

[212] D. Feng, L. Rosenbaum, and K. Dietmayer, "Towards safe autonomous driving: Capture uncertainty in the deep neural network for lidar 3D vehicle detection," in *Proc. 21st Int. Conf. Intell. Transp. Syst. (ITSC)*, Nov. 2018, pp. 3266–3273.

[213] D. Feng, L. Rosenbaum, F. Timm, and K. Dietmayer, "Leveraging heteroscedastic aleatoric uncertainties for robust real-time LiDAR 3D object detection," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Jun. 2019, pp. 1280–1287.

[214] G. P. Meyer, A. Laddha, E. Kee, C. Vallespi-Gonzalez, and C. K. Wellington, "LaserNet: An efficient probabilistic 3d object detector for autonomous driving," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 12677–12686.

[215] S. Wirges, M. Reith-Braun, M. Lauer, and C. Stiller, "Capturing object detection uncertainty in multi-layer grid maps," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Jun. 2019, pp. 1520–1526.

[216] M. T. Le, F. Diehl, T. Brunner, and A. Knol, "Uncertainty estimation for deep neural object detectors in safety-critical applications," in *Proc. 21st Int. Conf. Intell. Transp. Syst. (ITSC)*, Nov. 2018, pp. 3873–3878.

[217] D. Feng, L. Rosenbaum, C. Glaeser, F. Timm, and K. Dietmayer, "Can we trust you? On calibration of a probabilistic object detector for autonomous driving," 2019, *arXiv:1909.12358*. [Online]. Available: http://arxiv.org/abs/1909.12358

[218] D. P. Kingma and M. Welling, "Auto-encoding variational Bayes," in *Proc. Int. Conf. Learn. Representations*, 2014, pp. 1–14.

[219] I. Goodfellow *et al.*, "Generative adversarial nets," in *Proc. Adv. Neural Inf. Process. Syst.*, 2014, pp. 2672–2680.

[220] A. B. L. Larsen, S. K. Sønderby, H. Larochelle, and O. Winther, "Autoencoding beyond pixels using a learned similarity metric," in *Proc. Int. Conf. Mach. Learn.*, 2016, pp. 1558–1566.

[221] A. Deshpande, J. Lu, M.-C. Yeh, M. J. Chong, and D. Forsyth, "Learning diverse image colorization," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 2877–2885.

[222] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 5967–5976.

[223] T. A. Wheeler, M. Holder, H. Winner, and M. J. Kochenderfer, "Deep stochastic radar models," in *Proc. IEEE Intell. Vehicles Symp.*, Jun. 2017, pp. 47–53.

[224] X. Han, J. Lu, C. Zhao, S. You, and H. Li, "Semisupervised and weakly supervised road detection based on generative adversarial networks," *IEEE Signal Process. Lett.*, vol. 25, no. 4, pp. 551–555, Apr. 2018.

[225] J. L. Elman, "Learning and development in neural networks: The importance of starting small," *Cognition*, vol. 48, no. 1, pp. 71–99, Jul. 1993.

[226] J. Feng and T. Darrell, "Learning the structure of deep convolutional networks," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 2749–2757.

[227] D. Ramachandram, M. Lisicki, T. J. Shields, M. R. Amer, and G. W. Taylor, "Structure optimization for deep multimodal fusion networks using graph-induced kernels," in *Proc. 25th Eur. Symp. Artif. Neural Netw., Comput. Intell. Mach. Learn.*, 2017, pp. 1–6.

[228] D. Whitley, T. Starkweather, and C. Bogart, "Genetic algorithms and neural networks: Optimizing connections and connectivity," *Parallel Comput.*, vol. 14, no. 3, pp. 347–361, Aug. 1990.

[229] B. Zoph and Q. V. Le, "Neural architecture search with reinforcement learning," 2016, *arXiv:1611.01578*. [Online]. Available: https://arxiv.org/abs/1611.01578

[230] P. Kulkarni, J. Zepeda, F. Jurie, P. Pérez, and L. Chevallier, "Learning the structure of deep architectures using L1 regularization," in *Proc. Procedings Brit. Mach. Vis. Conf.*, 2015, pp. 1–12.

[231] C. Murdock, Z. Li, H. Zhou, and T. Duerig, "Blockout: Dynamic model selection for hierarchical deep networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 2583–2591.

[232] F. Li, N. Neverova, C. Wolf, and G. Taylor, "Modout: Learning multi-modal architectures by stochastic regularization," in *Proc. 12th IEEE Int. Conf. Autom. Face Gesture Recognit. (FG)*, May 2017, pp. 422–429.

[233] A. Bilal, A. Jourabloo, M. Ye, X. Liu, and L. Ren, "Do convolutional neural networks learn class hierarchy?" *IEEE Trans. Visual. Comput. Graph.*, vol. 24, no. 1, pp. 152–162, Jan. 2018.

[234] M. Liu, J. Shi, Z. Li, C. Li, J. Zhu, and S. Liu, "Towards better analysis of deep convolutional neural networks," *IEEE Trans. Visual. Comput. Graph.*, vol. 23, no. 1, pp. 91–100, Jan. 2017.

[235] S. Han, H. Mao, and W. J. Dally, "Deep compression: Compressing deep neural networks with pruning, trained quantization and Huffman coding," in *Proc. Int. Conf. Learn. Representations*, 2015, pp. 1–14.

[236] Y. Cheng, D. Wang, P. Zhou, and T. Zhang, "A survey of model compression and acceleration for deep neural networks," 2017, *arXiv:1710.09282*. [Online]. Available: https://arxiv.org/abs/1710.09282

[237] L. Enderich, F. Timm, L. Rosenbaum, and W. Burgard, "Learning multimodal fixed-point weights using gradient descent," in *Proc. 27th Eur. Symp. Artif. Neural Netw., Comput. Intell. Mach. Learn.*, 2019, pp. 1–6.

[238] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The Pascal visual object classes (VOC) challenge," *Int. J. Comput. Vis.*, vol. 88, no. 2, pp. 303–338, Jun. 2010.

[239] A. P. Dawid, "The well-calibrated Bayesian," *J. Amer. Stat. Assoc.*, vol. 77, no. 379, pp. 605–610, Sep. 1982.

[240] S. Liang, Y. Li, and R. Srikant, "Enhancing the reliability of out-of-distribution image detection in neural networks," in *Proc. Int. Conf. Learn. Representations*, 2017, pp. 1–27.

[241] D. Hall, F. Dayoub, J. Skinner, P. Corke, G. Carneiro, and N. Sünderhauf, "Probabilistic object detection: Definition and evaluation," 2018, *arXiv:1811.10800*. [Online]. Available: https://arxiv.org/abs/1811.10800

[242] W. Luo, B. Yang, and R. Urtasun, "Fast and furious: Real time end-to-end 3D detection, tracking and motion forecasting with a single convolutional net," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 3569–3577.

[243] M. Bojarski *et al.*, "End to end learning for self-driving cars," 2016, *arXiv:1604.07316*. [Online]. Available: http://arxiv.org/abs/1604.07316

[244] G.-H. Liu, A. Siravuru, S. Prabhakar, M. Veloso, and G. Kantor, "Learning end-to-end multimodal sensor policies for autonomous navigation," in *Proc. 1st Annu. Conf. Robot Learn.*, 2017, pp. 249–261.

[245] M. Bansal, A. Krizhevsky, and A. Ogale, "ChauffeurNet: Learning to drive by imitating the best and synthesizing the worst," in *Proc. Robotics: Sci. Syst.*, Dec. 2019, pp. 1–20.

[246] A. Sauer, N. Savinov, and A. Geiger, "Conditional affordance learning for driving in urban environments," in *Proc. 2st Annu. Conf. Robot Learn.*, 2018, pp. 237–252.

[247] C. Chen, A. Seff, A. Kornhauser, and J. Xiao, "DeepDriving: Learning affordance for direct perception in autonomous driving," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 2722–2730.

[248] S. Gu, T. Lu, Y. Zhang, J. M. Alvarez, J. Yang, and H. Kong, "3-D LiDAR+ monocular camera: An inverse-depth-induced fusion framework for urban road detection," *IEEE Trans. Intell. Veh.*, vol. 3, no. 3, pp. 351–360, Sep. 2018.

[249] 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.

[250] O. Erkent, C. Wolf, C. Laugier, D. S. Gonzalez, and V. R. Cano, "Semantic grid estimation with a hybrid Bayesian and deep neural network approach," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Oct. 2018, pp. 888–895.

[251] Y. Cai, D. Li, X. Zhou, and X. Mou, "Robust drivable road region detection for fixed-route autonomous vehicles using map-fusion images," *Sensors*, vol. 18, no. 12, p. 4158, Nov. 2018.

Di Feng (Member, IEEE) received the bachelor's degree in mechatronics (Honos.) from Tongji University in 2014, and the master's degree (Hons.) in electrical and computer engineering from the Technical University of Munich. He is currently pursuing the Ph.D. degree with the Corporate Research of Robert Bosch GmbH, Renningen, in cooperation with the Ulm University. During his studies, he was granted the opportunity to work in several teams with reputable companies and research institutes, such as BMW AG, the German Aerospace Center (DLR), and the Institute for Cognitive Systems (ICS) Technical University of Munich. His current research is centered on robust multimodal object detection using deep learning approach for autonomous driving. He is also interested in robotic active learning and exploration through tactile sensing and cognitive systems.

Christian Haase-Schütz (Member, IEEE) received the bachelor's degree in physics from the University of Stuttgart in 2013, and the master's degree in physics from Friedrich-Alexander-University Erlangen-Nuremberg. He is currently pursuing the Ph.D. degree with Chassis Systems Control, Robert Bosch GmbH, Abstatt, in cooperation with the Karlsruhe Institute of Technology. He did his bachelor's thesis with the Max Planck Institute for Intelligent Systems and his master's thesis with the Center for Medical Physics. During his master studies, he was granted a scholarship by the Bavarian state to visit Huazhong University of Science and Technology, Wuhan, China, from March 2015 to July 2015. His professional experience includes work with ETAS GmbH, Stuttgart, and Andreas Stihl AG, Waiblingen. His current research is centered on multimodal object detection using deep learning approaches for autonomous driving. He is further interested in challenges of AI systems in the wild. He is a member of the German Physical Society DPG.

Lars Rosenbaum received the Dipl.Inf. (M.S.) and the Dr. rer. nat. (Ph.D.) degrees in bioinformatics from the University of Tuebingen, Germany, in 2009 and 2013, respectively. During this time, he was working on machine learning approaches for computer-aided molecular drug design and analysis of metabolomics data. In 2014, he joined ITK Engineering in Marburg, Germany, working on driver assistance systems. Since 2016, he has been a Research Engineer with Corporate Research, Robert Bosch GmbH, Renningen, Germany, where he is currently doing research on machine learning algorithms in the area of perception for automated driving.

Heinz Hertlein (Member, IEEE) received the Dipl.Inf. (diploma in computer science) and Dr.Ing. (Ph.D.) degrees in biometric speaker recognition from the Friedrich-Alexander-University Erlangen-Nuremberg, Germany, in 1999 and 2010, respectively. From 2002, he was working on algorithms and applications of multimodal biometric pattern recognition at the company BioID in Erlangen-Tennenlohe and Nuremberg. From 2012, he was appointed at the University of Hertfordshire, U.K., initially as a Postdoctoral Research Fellow and later as a Senior Lecturer. He was teaching in the fields of signal processing and pattern recognition, and his research activities were mainly focused on biometric speaker and face recognition. Since 2015, he has been at Chassis Systems Control, Robert Bosch GmbH in Abstatt, Germany, where he is currently working in the area of perception for autonomous driving.

Claudius Gläser was born in Gera, Germany in 1982. He received the Diploma degree in computer science from the Technical University of Ilmenau, Germany, in 2006, and the Dr.Ing. degree (equivalent to Ph.D.) from Bielefeld University, Germany, in 2012. From 2006 he was a Research Scientist with the Honda Research Institute Europe GmbH, Offenbach/Main, Germany, working in the fields of speech processing and language understanding for humanoid robots. In 2011, he joined the Corporate Research of Robert Bosch GmbH in Renningen, Germany, where he developed perception algorithms for driver assistance and highly automated driving functions. He is currently the team lead for perception for automated driving and manages various related projects. His research interests include environment perception, multimodal sensor data fusion, multiobject tracking, and machine learning for highly automated driving.

Fabian Timm received the Ph.D. degree in computer science from the University of Lübeck, Germany, the Diploma degree in collabration with Philips Lighting Systems, Aachen, Germany, in 2006, and the Ph.D. degree in the field of machine vision and machine learning from the Institute for Neuro-and Bioinformatics, University of Lübeck, in collaboration with Philips Lighting Systems, in 2011. In 2012, he joined corporate research at Robert Bosch GmbH, where he worked on industrial image processing and machine learning. Afterwards, he worked in the business unit at Bosch and developed new perception algorithms, such as pedestrian and cyclist protection only with a single radar sensor. Since 2018, he has been leading the Automated Driving-Perception and Sensors Research Group, Bosch Corporate Research. His main research interests are machine and deep learning, signal processing, and sensors for automated driving.

Werner Wiesbeck (Life Fellow, IEEE) received the Dipl.Ing. (M.S.) and Dr.Ing. (Ph.D.) degrees in electrical engineering from the Technical University Munich, Germany, in 1969 and 1972, respectively. From 1972 to 1983, he was with product responsibility for mm-wave radars, receivers, direction finders, and electronic warfare systems in industry. From 1983 to 2007, he was the Director of the Institut für Höchstfrequenztechnik und Elektronik, University of Karlsruhe. He is currently a Distinguished Senior Fellow with the Karlsruhe Institute of Technology. His research topics include antennas, wave propagation, radar, remote sensing, wireless communication, and ultra wide band technologies. He has authored or coauthored several books and over 850 publications. He is also a Supervisor of over 90 Ph.D. students, a responsible supervisor of over 600 Diploma-/Master thesis. He holds over 60 patents. He is an Honorary Life Member of IEEE GRS-S and a member of the Heidelberger Academy of Sciences and Humanities, and the German Academy of Engineering and Technology. He was a recipient of a number of awards, including the IEEE Millennium Award, the IEEE GRS Distinguished Achievement Award, the Honorary Doctorate (Dr. h. c.) from the University Budapest/Hungary, the Honorary Doctorate (Dr.Ing. E. h.) from the University Duisburg/Germany, the Honorary Doctorate (Dr.Ing. E. h.) from Technische Universität Ilmenau, and the IEEE Electromagnetics Award in 2008. He is also the Chairman of the GRS-S Awards Committee. He was the Executive Vice President of IEEE GRS-S from 1998 to 1999 and the President of IEEE GRS-S from 2000 to 2001. He has been the general chairman of several conferences.

Klaus Dietmayer (Member, IEEE) was born in Celle, Germany, in 1962. He received the Diploma degree in electrical engineering from the Technical University of Braunschweig, Germany, in 1989, and the Dr.Ing. degree (equivalent to Ph.D.) from the University of Armed Forces, Hamburg, Germany, in 1994. In 1994, he joined the Philips Semiconductors Systems Laboratory, Hamburg, as a Research Engineer. Since 1996, he became a Manager in the field of networks and sensors for automotive applications. In 2000, he was appointed to a Professorship at the University of Ulm in the field of measurement and control. He is currently a Full Professor and the Director of the School of Engineering and Computer Science, Institute of Measurement, Control and Microtechnology, University of Ulm. His research interests include information fusion, multiobject tracking, environment perception, situation understanding, and trajectory planning for autonomous driving. He is a member of the German Society of Engineers VDI/VDE.