World Electric Vehicle Journal

MDPI

Review

Emerging Trends in Autonomous Vehicle Perception: Multimodal Fusion for 3D Object Detection

Simegnew Yihunie Alaba *, Ali C. Gurbuz and John E. Ball

Department of Electrical and Computer Engineering, James Worth Bagley College of Engineering, Mississippi State University, Starkville, MS 39762, USA; gurbuz@ece.msstate.edu (A.C.G.); jeball@ece.msstate.edu (J.E.B.)

* Correspondence: sa1724@msstate.edu

Abstract: The pursuit of autonomous driving relies on developing perception systems capable of making accurate, robust, and rapid decisions to interpret the driving environment effectively. Object detection is crucial for understanding the environment at these systems' core. While 2D object detection and classification have advanced significantly with the advent of deep learning (DL) in computer vision (CV) applications, they fall short in providing essential depth information, a key element in comprehending driving environments. Consequently, 3D object detection becomes a cornerstone for autonomous driving and robotics, offering precise estimations of object locations and enhancing environmental comprehension. The CV community's growing interest in 3D object detection is fueled by the evolution of DL models, including Convolutional Neural Networks (CNNs) and Transformer networks. Despite these advancements, challenges such as varying object scales, limited 3D sensor data, and occlusions persist in 3D object detection. To address these challenges, researchers are exploring multimodal techniques that combine information from multiple sensors, such as cameras, radar, and LiDAR, to enhance the performance of perception systems. This survey provides an exhaustive review of multimodal fusion-based 3D object detection methods, focusing on CNN and Transformer-based models. It underscores the necessity of equipping fully autonomous vehicles with diverse sensors to ensure robust and reliable operation. The survey explores the advantages and drawbacks of cameras, LiDAR, and radar sensors. Additionally, it summarizes autonomy datasets and examines the latest advancements in multimodal fusion-based methods. The survey concludes by highlighting the ongoing challenges, open issues, and potential directions for future research.

Keywords: 3D object detection; autonomous vehicles; deep learning; LiDAR; multimodal fusion; perception; vision transformers

Citation: Alaba, S.Y.; Gurbuz, A.C.; Ball, J.E. Emerging Trends in Autonomous Vehicle Perception: Multimodal Fusion for 3D Object Detection. *World Electr. Veh. J.* **2024**, *15*, 20. https://doi.org/10.3390/wevj15010020

Academic Editor: Joeri Van Mierlo

Received: 14 December 2023
Revised: 28 December 2023
Accepted: 4 January 2024
Published: 7 January 2024

1. Introduction

The National Highway Traffic Safety Administration (NHTSA) 2021 US report showed 42,915 motor-vehicle deaths in 2021, marking a 10.5% increase from the 38,824 fatalities recorded in 2020 [1]. Despite a reduction in total mileage driven in 2020 due to pandemic lockdowns, the annual estimated death and fatality rates have risen from previous years, primarily attributed to high-speed driving. Autonomous driving systems aim to reduce these fatalities, enhancing overall driving safety and improving general traffic flow. Therefore, developing an autonomous driving system that is accurate, robust enough to operate under various environmental conditions, capable of fast and effective decision-making for high-speed driving, and compliant with safety standards can be instrumental in reducing the fatalities caused by human drivers. To enhance the robustness of autonomous vehicles (AVs), they must comprehend their surrounding driving environments, including other vehicles, pedestrians, cyclists, lane information, and constraints like speed limits on streets. Perception systems, particularly object detection, play a vital role in understanding these

environments and their locations. The perception system needs to provide accurate and precise information about the driving environment, demonstrate robustness in adverse weather conditions like snow, rain, fog, and during night driving, and make real-time decisions suitable for high-speed driving [2,3].

Object detection aims to identify object labels and predict their locations in images, videos, or other sensory data. Because of the additional dimension, 3D object detection gives a more thorough understanding of an environment than 2D. The 3D sensors are crucial to achieving these goals, such as Light Detection and Ranging (LiDAR), Radio Detection and Ranging (radar), and depth cameras (RGB-D) [4]. However, environmental variations, resolution differences, occlusions, and sensor limitations are crucial in the performance of the perception system.

The LiDAR sensor is widely used in autonomous vehicle applications due to its superior performance in adverse weather conditions compared to cameras, its high azimuth resolution, and its ability to provide accurate 3D information. In contrast, cameras lack inherent 3D data and are rich in color and texture details, which are crucial for object recognition and classification. Deep Learning (DL), with its ability to learn features directly from images without the need for engineered features, has significantly advanced the development of autonomous driving and other computer vision applications. However, despite LiDAR's importance, processing its data points presents more challenges due to their unstructured nature and sparsity. Radar sensors, another vital component, excel in adverse weather conditions and offer better long-range detection capabilities than LiDARs. While radars provide precise distance and velocity measurements, they have lower cross-range resolution than cameras or LiDARs. Consequently, sensor fusion techniques have been employed to harness the strengths of various sensors [5,6]. The major contributions of the paper are:

1. This work comprehensively analyzes multimodal fusion-based 3D object detection, including state-of-the-art methods and comparative results.
2. DL-based multimodal fusion methods are categorized into data (early) fusion, feature (middle) fusion, and decision (late) fusion.
3. Transformer-based multimodal fusion for 3D object detection methods and the challenges and shortcomings of vision transformers are discussed.
4. The paper identifies research challenges, open issues, and gaps in 3D object detection.
5. An overview of commonly used sensors and new datasets relevant to 3D object detection in autonomous driving is provided.

The structure of the remainder of this paper is as follows: Section 2 discusses related work. Section 3 summarizes commonly used sensors, datasets for 3D object detection, attention mechanisms in CNNs, and vision transformers. Section 4 provides an overview of multimodal fusion 3D object detection methods, including a comparative analysis of selected techniques. Challenges, open issues, and future research directions are outlined in Section 5. The final Section 6 offers a summary of the survey paper.

2. Related Work

Due to the rapid advancement of DL algorithms, numerous papers on 3D object detection for autonomous driving and robotics applications have been published. This paper comprehensively reviews DL-based multimodal fusion 3D object detection models specifically for autonomous driving. While radar is an essential sensor in autonomous driving, known for its robustness in adverse weather and long-range velocity detection capabilities, the volume of radar-based research is relatively small, primarily due to the scarcity of public datasets. This work addresses this gap by presenting state-of-the-art methods and analyses of radar-based multimodal fusion for 3D object detection. 3D object detection is rapidly growing in the computer vision community, leading to a significant yearly increase in published papers on multimodal fusion 3D object detection. However, more comprehensive survey papers on these multimodal fusion 3D object detection methods are still needed.

Feng et al. [3] focused primarily on general object detection and semantic segmentation datasets and autonomous driving methodologies, emphasizing 2D object detection and semantic segmentation. Arnold et al. [4] and Rahman et al. [7] offered a brief overview of 3D object detection methods specifically for autonomous driving applications. Li et al. [8], Guo et al. [9], and Fernandes et al. [10] discussed deep learning applications for LiDAR point clouds in autonomous driving, covering aspects like object detection, segmentation, and classification. Additionally, Li et al. [11], Lateef and Ruichek [12], and Yu et al. [13] covered general semantic segmentation work. Qian et al. [14] reviewed 3D object detection methods for autonomous driving applications. More recently, Cui et al. [15] and Wang et al. [16] explored image and LiDAR fusion in autonomous driving, reviewing camera-LiDAR fusion across 2D/3D object detection, semantic segmentation, and tracking. Although their survey encompasses fusion work, it includes topics beyond 3D object detection. Hence, our survey paper is distinct, focusing on multimodal fusion 3D object detection. Alaba et al. [2,17] reviewed 3D object detection methods using LiDAR and camera images for autonomous driving applications.

This comprehensive review meticulously examines state-of-the-art methods and provides detailed analyses of various fusion techniques, including radar-camera, LiDAR-camera, and radar-camera-LiDAR integrations. Fusion-based works are methodically categorized into groups based on data, feature, and decision fusion approaches. Unlike previous surveys discussing fusion techniques, this review uniquely classifies each work into specific categories according to the fusion method employed. A comparative analysis of several state-of-the-art multimodal fusion 3D object detection methods, including multimodal vision transformer-based methods, based on accuracy and processing speed is presented. Furthermore, the survey highlights commonly used sensors and introduces new datasets for autonomous driving. Additionally, ten significant research challenges, open issues, and gaps in multimodal fusion and general 3D object detection are explored. For related methodologies, such as feature extraction, various stages of autonomous driving, encoding techniques for 3D bounding boxes, and evaluation metrics for 3D object detection, refer [2,17]. This thorough review aims to provide a clear and structured understanding of the current methodologies and future directions in autonomous driving in multimodal fusion 3D object detection.

3. Background

3.1. Sensors in Autonomous Driving

The object detection system usually depends on multiple sensors to overcome the shortcomings of individual sensors. These sensors can be divided into passive sensors that do not emit signals for taking measurements, such as visual or stereo cameras, and active sensors that emit signals to sense the environment, such as LiDARs, radars, and ultrasonics. This section presents the various automotive sensors used in 3D object detection. Table 1 compares commonly used sensors in autonomous driving.

3.1.1. Monocular and Stereo Cameras

Monocular and Stereo cameras provide an object's precise color, texture, and appearance information. Monocular cameras are commonly used for 2D CV applications. In 3D object detection, their usage is limited because of their lack of depth information, which is crucial for understanding the surrounding environment. Using stereo cameras or applying motion structure helps solve the lack of depth information [18,19]. Stereo cameras, which produce pairs of images, provide more information to infer depth than monocular cameras. This leads to superior detection performance of stereo-based methods over monocular-based methods. However, achieving precise calibration and synchronization of stereo cameras, which is required for accurate results, is more challenging. The other disadvantage of monocular and stereo cameras is that they are poor in adverse weather, such as snow, rain, and fog. These cameras are also poor at low luminosity in the nighttime and extreme brightness in the daytime. We can reduce the shortcomings of monocular and

stereo cameras by fusing them with other sensors. Some works, such as [5,6], combined images and LiDAR point cloud data to further enhance object detection by using the color and texture information from cameras and the 3D data from LiDARs. The fusion techniques are detailed in Section 4. RGB cameras typically output at standard frame rates between 24–60 fps depending on the camera model. However, specialized cameras in scientific or industrial applications may have higher frame rates.

Table 1. Comparison of commonly used sensors in autonomous driving.

Sensors	Advantages	Disadvantages
Monocular Cameras	(a) Provide detailed texture information, color, and appearance. (b) Low cost.	(a) Not suitable for 3D detection due to lack of depth information. (b) Vulnerable to adverse light and weather.
Stereo Cameras	(a) Provide detailed texture information, color, and appearance. (b)Provide more information to infer depth.	(a) Vulnerable to adverse light and weather. (b) Calibration and synchronization are more challenging.
Depth cameras	(a) Provide detailed texture information, color, and appearance. (b) Provide depth information.	(a) Some of them might have limited range and accuracy. (b) Environmental interference from bright sunlight, reflective surfaces, or other infrared sources.
Thermal Cameras	Robust to low luminosity at night.	(a) Lack of depth information. (b) Low resolution
LiDARs	(a) Provide 3D information. (b) Less affected by adverse light and weather. (c) Precise measurements of distance and velocity.	(a) Does not provide fine texture information. (b) The data points are sparse and unstructured. (c) Expensive and large.
Radars	(a) Provide 3D information. (b) Robust to adverse light and weather. (c) Long-range distance and velocity measurement. (d) Typically used for adaptive cruise control in most AVs.	(a) Lower resolutions (b) Interference to other radar or communication systems. (c) High false-negative rate for static targets.
Ultrasonics	Good for close-range detection.	Affected by temperature and humidity.

3.1.2. Depth Cameras

Depth cameras provide visual data (color imagery) and depth information and utilize various technologies to measure the distance from the camera to objects in the environment. These cameras use various technologies, and one of the most commonly used technologies is time-of-flight (ToF). ToF technology-based depth cameras emit infrared light pulses and measure the time it takes for these pulses to travel to objects in the scene and return to the camera. The distance to objects is measured by calculating the time it takes for the emitted light to return to the sensor. An RGB-D camera, as the name shows, is a depth camera that combines RGB imaging and depth information in a single device. This combination enhances spatial understanding of the environment. RGB-D camera serves as the basis for constructing three-dimensional representations of the objects in the captured environment due to the additional captured depth information. Depth cameras, mainly ToF cameras, operate at high frame rates of up to 100 Hz by emitting light pulses to measure the time light returns, which enables depth calculations.

3.1.3. Thermal Cameras

Thermal cameras detect infrared energy, also called heat, and transform it into a visual image. The cameras are robust to low light luminosity at night because they do not rely on visible light. In addition, thermal cameras exhibit robustness in adverse weather conditions, including fog, smoke, or light precipitation. These cameras consume less power as they do not require additional lighting sources to capture images, making them energy efficient. They can detect heat emissions, enabling them to identify hidden objects or individuals. Thermal cameras often have frame rates ranging from 9 Hz to 60 Hz. Lower-end models may have slower rates, while high-performance ones can achieve higher frame rates for real-time monitoring.

One drawback of thermal cameras is that they have a lower resolution than traditional cameras. Although they have the advantage of detecting temperature variations and producing a visual representation of heat, the images they produce may have a different level of detail and clarity than those captured by traditional cameras. This could make it more challenging to identify smaller objects or specific details in the imagery. Also, thermal cameras lack inherent depth information, like monocular cameras, making it challenging to perceive three-dimensional objects.

3.1.4. LiDARs

LiDARs emit short laser energy pulses and measure the time between emitting and detecting the pulse to estimate distance. The sensor readings are given in a set of 3D coordinates called a Point Cloud (PCL), which gives accurate depth information of each point's surroundings and intensity levels. Unlike images, point clouds are sparse and unstructured but more robust to different lighting conditions and adverse weather. LiDAR can detect objects from over 200 m, but the number of points in a point cloud on a similarly sized object decreases. However, they do not generally provide precise textural information like monocular cameras. The flash LiDAR [20] is a solid-state LiDAR that uses an optical flash operation like a standard digital camera to provide precise texture information. Generally, the narrow-pulsed time of flight (ToF) based beam steering in LiDAR systems can be classified into three categories: Mechanical LiDAR, Solid-state LiDAR [20], and Semi-solid state LiDAR. A mechanical LiDAR has a 360-degree field of view (FOV) through high-grade optics and a rotating assembly. On the other hand, a solid-state LiDAR has no spinning mechanical components, which reduces FOV. Solid-state LiDARs utilize solid-state components such as silicon photonics or micro-electro-mechanical systems (MEMS) to generate and direct laser beams without needing moving parts. Because of these changes from mechanical LiDARs, solid-state LiDAR is usually less expensive. Semi-solid-state LiDARs utilize solid-state components and traditional mechanical elements to perform their functions. This combination balances the benefits of solid-state technology, such as compactness, reliability, and mechanical LiDAR systems like range and resolution. Pure solid-state LiDARs are smaller than semi-solid LiDARs, as the latter includes mechanical components that can affect the overall form factor and cost.

Furthermore, new LiDAR technologies, such as Frequency-Modulated Continuous Wave (FMCW) LiDAR and 4D LiDAR, have been introduced to improve the capabilities of traditional LiDARs in autonomous driving. FMCW LiDAR operates like continuous wave radar by using a laser signal that varies in frequency over time. Unlike pulsed LiDAR, which emits brief light bursts and measures their return time, FMCW LiDAR emits a continuous modulated laser signal. There is also an amplitude-modulated continuous wave (AMCW) LiDAR. FMCW LiDARs measure the frequency of the waves, but AMCW LiDARs measure the amplitude of the waves. A 4D LiDAR system can operate in three spatial dimensions: range, azimuth, elevation, and the fourth dimension of time. By including time, it provides an additional layer of information related to objects' motion and changes over time, which helps to measure the speed of an object directly. LiDAR system's output frequency varies based on the technology used. Mechanical LiDARs have lower scanning frequencies, ranging from a few hundred kilohertz to a few megahertz.

Newer technologies like solid-state LiDARs can achieve higher scan rates, reaching several megahertz or gigahertz ranges. The detailed performance comparisons of different LiDAR sensors are given in Lambert et al. [21].

The 3D object detection methods use a fusion technique to use the best out-of-camera and LiDAR data. However, the cameras need to be calibrated before fusion and co-registered to get a single spatial frame of reference. Pusztai and Hajder proposed calibrating LiDAR-Camera systems using ordinary boxes [22] with known sizes. Similarly, Kim and Park developed an extrinsic calibration method using a planar chessboard with multiple 3D matching planes [23]. Bai et al. [24] proposed a calibration method by finding the line correspondences between the LiDAR point clouds and camera images. An et al. [25] developed a geometric calibration method to estimate the extrinsic parameters of the LiDAR and camera system. The method provides 2D–3D and 3D–3D correspondences. Zhou et al. put forth an extrinsic calibration [26] technique using line and plane correspondences. Recently, Pusztai et al. [27] and Zhu et al. [28] proposed more LiDAR-camera calibration techniques. The main limitations of LiDAR sensors are the high price and the struggle to detect objects at close distances [20] or far distances. However, the mass production of solid-state LiDARs may reduce the price significantly.

3.1.5. Radars

Radar units emit electromagnetic waves to measure multiple objects' range, velocity, and direction of arrival. Most automotive radars use the Frequency Modulated Continuous Wave (FMCW) modulation technique [29], called linear FMCW, to measure range and velocity estimation simultaneously. We can classify the automotive radars into short-range (0.15–30 m), medium-range (1–100 m), and Long-range (10–250 m) based on range measurement capability [29]. Radars can measure an object's radial velocity directly using a phase shift. Radars are also more robust to lighting and adverse weather conditions than LiDARs. Still, their elevation resolutions are as low as an angular resolution in cross-range, making it harder to classify objects using radars. LiDAR has better vertical FOV (elevation) as well as angular resolution (for both azimuth and elevation) than radar [20]. A 4D rada technology also provides range, azimuth, elevation, and time. Another potential disadvantage of radars is the signal interference with other radars or communication systems. Radar systems operate at different frequencies for various applications, ranging from MHz to GHz. Automotive radars for collision detection typically use 24 or 77 GHz frequency bands.

3.1.6. Ultrasonic Sensors

Ultrasonic sensors emit high-frequency sound waves to estimate the distance from objects. Because of their short-range object detection capacity, they are suitable for slow-speed operations, such as parallel parking in autonomous vehicles. The limitation of ultrasonic sensors is that adverse weather affects these sensors, such as temperature and humidity. Ultrasonic sensors emit sound waves in the ultrasonic frequency range between tens of kHz and several hundred kHz, depending on the sensor design.

Autonomous vehicles require the integration of multiple sensors, including cameras, radars, and LiDARs, for advanced perception and complete autonomy. Table 1 shows each sensor has advantages and disadvantages. Relying on a single sensor type is often insufficient for all conditions and tasks, making the combination of different sensors essential for fully autonomous driving. This approach leverages the redundant data from these sensors to effectively respond to varying conditions. The necessity of multimodal fusion lies in using the redundancy of these sensors to address specific problems, as discussed in [30]. Table 2 illustrates how these sensors perform across various perception tasks.

Table 2. The camera, radar, and LiDAR sensors' performance on different perception systems. We use good, fair, and poor for relative comparison among sensors. Good means the sensor performs well for the given task.

Tasks	Camera	Radar	LiDAR
Classification	Good	Poor	Fair
Velocity Measurement	Poor	Good	Fair
Lane Tracking	Good	Poor	Poor
Sign Recognition	Good	Poor	Poor
Performance on Adverse Weather	Poor	Good	Fair
Performance on Night Vision	Poor	Good	Good
Distance Estimation	Fair	Good	Good

Cameras excel in classification, lane tracking, and sign recognition tasks but fall short in measuring velocity, operating under adverse weather conditions, and during night vision scenarios. Conversely, radars stand out in object detection, velocity measurement, functioning in adverse weather, night vision capabilities, and estimating distances, yet they underperform in classification, lane tracking, and sign recognition. LiDARs perform well in object detection, night vision applications, and precise distance estimation.

3.2. Datatsets for 3D Object Detection

Most DL object detection methods utilize supervised learning. So, they need labeled and annotated images for training. Additionally, most of the datasets in autonomous vehicles are authentic images generated from sensors, but few synthetic datasets are generated from game engines and simulators. This section presents commonly used datasets for 3D object detection of autonomous driving. The comparison of 3D public datasets is provided in Table 3.

Table 3. Comparison of the public dataset for autonomous driving.

Data	Year	Sensors	Classes	Annotation	Variety	Recording Regions
KITTI	2012	LiDAR & Camera	8	2D & 3D	daytime	Karlsruhe, Germany
Cityscapes	2016	Camera	30	2D	spring, summer, fall, day, night, sunrise, & different weather	Primarily Germany
RobotCar	2017	Camera, LIDAR, & GPS	-	2D & 3D	heavy rain, night, direct sunlight, & snow	UK
KAIST	2018	Camera, LiDAR, & GPS/IMU	-	2D & 3D	daytime	Korea
nuScenes	2019	LiDAR, Camera, & Radar	23	2D & 3D	daytime & nighttime	Boston, USA

Table 3. *Cont.*

Data	Year	Sensors	Classes	Annotation	Variety	Recording Regions
HD3	2019	Camera, LiDAR, & GPS/IMU	8	3D	daytime	San Francisco, USA
ApolloScape	2019	Camera, LiDAR, & IMU/GNSS	35	2D & 3D	daytime & nighttime	4 different regions, China
Astyx radar	2019	Camera, LiDAR, & Radar	7	3D	daytime	Germany
Lyft L5	2019	Camera & Radar	9	3D	daytime	Palo Alto
Drivingstere	2019	Camera, LiDAR, & IMU/GNSS	6	2D & 3D	sunny, rainy, cloudy, foggy, & dusky	China
A*3D	2020	Camera & LiDAR	7	3D	daytime, nighttime, fog, & rain	Singapore
LIBRE	2020	LiDAR	-	3D	fog, rain, & strong-light	Nagoya University, Japan
Pandaset	2020	Camera& LiDAR	28	3D	daytime & nighttime	San Francisco
Waymo	2020	Camera & LiDAR	4	2D & 3D	daytime, nighttime, dawn & dusk	San Francisco, Phoenix, & Mountain View, USA
ONCE	2021	Camera & LiDAR	5	3D	daytime, nighttime, & rain	China

3.2.1. KITTI

KITTI is one of the most well-known multisensor datasets for autonomous driving applications for stereo, optical flow, visual odometry, 3D object detection, and 3D tracking tasks [31]. The dataset is recorded in the city of Karlsruhe, Germany, in rural areas using four Varifocal lenses, two-color and gray-scale high-resolution video cameras, one inertia navigation system GPS/IMU, and a 360° laser scanner Velodyne HDL-64E. The object detection and BEV benchmark contains 7481 training images, 7518 test images, corresponding point clouds, camera calibration files, and annotated 3D boxes around objects of interest for cars, pedestrians, and cyclists. They labeled the annotations easy, moderate, and hard according to the size, occlusion, and truncation levels, with a maximum truncation of 15%, 30%, and 50%, respectively.

Although KITTI is a commonly used dataset, it has limitations. The dataset is collected during the daytime and under sunny conditions. So, it is not a representative dataset for real-world applications because of different weather and environmental conditions in the real world. Furthermore, the data set is unbalanced [32], which comprises 75% cars, 4% cyclists and 15% pedestrians.

3.2.2. nuScenes

The nuScenes dataset is another commonly used autonomy dataset recorded with six cameras, five radars, and one LiDAR, all with 360° field of view [33]. The dataset comprises

1000 scenes, each 20 s long, and fully annotated with 3D bounding boxes for 23 classes and eight attributes. They recorded it in two places: Boston (USA) and Singapore during the daytime, nighttime, and rainy conditions, making it a more real-time representative dataset. This multisensor dataset comprises 1.4 M RGB images, 400 K LiDAR point clouds, 1.3 M radar point clouds, and 1.4 M 3D boxes. The dataset has seven times as many annotations and 100 times as many images as the KITTI [31].

3.2.3. A*3D Dataset

A*3D Dataset comprises RGB images and LiDAR data with 39 K frames, seven classes, and 230 K 3D object annotations [34]. The authors collected the dataset from many parts of Singapore and has a more diverse scene, time, and weather than other autonomous vehicle datasets. The dataset comprises ten times the high-density images of the KITTI [31] Dataset. It also consists of three times the nuScenes dataset [33] in terms of heavy occlusions and a number of nighttime frames, which adds a more challenging and highly diverse environment for autonomous driving applications.

3.2.4. H3D

The Honda Research Institute 3D Dataset (H3D) is 3D multiobject detection and tracking dataset collected using three cameras, Velodyne HDL-64E LiDAR, and GPS/IMU sensors in San Francisco Bay Area [35]. It comprises eight classes and 160 traffic scenes with one million bounding box labels in 27,721 frames with challenging settings, such as highly interactive, complex, and occluded traffic compared to the KITTI dataset [31]. The major limitation of the dataset is that it is recorded during the daytime.

3.2.5. LIBRE (Multiple 3D LiDAR)

The LIBRE dataset was collected using ten different LiDAR sensors covering several models and laser configurations in three different environments and configurations: static targets, adverse weather, and dynamic traffic around Nagoya University, Japan [36]. Other supporting sensors: RGB, IR, 360°, event cameras, IMU, GNSS, and CAN using ROS [37] used for the recording.

3.2.6. Waymo

The dataset comprises 1150 scenes, each lasting 20 s, captured in various urban and suburban geographies in San Francisco, Phoenix, and Mountain View [38]. The dataset contains 12 million 3D annotated ground truth bounding boxes for LiDAR data and around 12 million 2D annotated ground truth bounding boxes for Camera images. It comprises around 113 K LiDAR objects and 250 K camera image tracks.

3.2.7. Astyx Radar

The Astyx radar dataset [39] is a radar-centric automotive dataset. The dataset is recorded in Germany using Astyx 6455 HiRes radar, Velodyne VLP-16 LiDAR, and Point Grey Blackfly camera. It comprises ground truth data of seven classes: namely, Bus, Car, Cyclist, Motorcyclist, Person, Trailer, and Truck. The ground truth annotation of the dataset contains 3D position (x, y, z), rotation (rx, ry, rz), 3D dimension (weight, length, height), class information, occlusion indicator, and position and dimension uncertainty.

3.2.8. Lyft L5

The Lyft L5 dataset [40] is prepared using the nuScenes dataset [33] format. The dataset is recorded using 64-wire radars and multiple cameras. The dataset comprises over 55,000 human-labeled 3D annotated frames with nine classes, surface maps, and underlying HD spatial semantic maps.

3.2.9. PandaSet

The PandaSet dataset [41] is a public autonomous driving dataset for academic and commercial use provided by Hesai & Scale. They recorded it using one mechanical LiDAR, one solid-state LiDAR, onboard GPS/IMU, and six cameras. The dataset comprises 48,000 camera images, 16,000 LiDAR sweeps, over 100 scenes of eight seconds each, 28 annotation classes for object classification, and 37 semantic segmentation labels.

Most survey papers reviewed commonly used autonomous driving datasets, but none reviewed stereo-based datasets. In this survey, we also review stereo datasets or datasets with stereo data collections for autonomous driving.

3.2.10. Cityscapes

The Cityscapes dataset [42] consists of a set of stereo video sequences recorded from 50 different cities, primarily in Germany and neighboring countries' streets, with 5000 pixel-level annotations and 20,000 coarse pixel-level annotations. The dataset is recorded using a 22 cm baseline stereo camera with a frame rate of 17 Hz. The dataset comprises 30 classes: parking, sidewalk, pole, road, and person. It is recorded during different seasons, such as spring, summer, fall, and different weather. It consists of semantic segmentation, instance segmentation, and panoptic labels for urban street scenes and other classes, such as vehicle and person.

3.2.11. KAIST

The KAIST dataset [43] consists of stereo images from urban and residential regions for autonomous systems. The dataset is recorded using a coaligned RGB/thermal camera, RGB stereo, Velodyne HDL-32E 3D LiDAR, and GPS/IMU during the day, night, sunrise, morning, sunset, and down. It is essential for object detection, drivable region detection, localization, image enhancement, depth estimation, and colorization.

3.2.12. ApolloScape

The ApolloScape dataset [44] is more prominent and affluent than the KITTI dataset [31] both in semantic labeling and amount of data. It is crucial for 2D and 3D semantic segmentation, object detection, and self-localization. They recorded the dataset using two laser scanners, up to six video cameras, and a combined IMU/GNSS system in four regions of China. They developed tracking and a 2D/3D joint annotation pipeline to speed up the labeling process.

3.2.13. Drivingstereo

The Drivingstereo dataset [45] comprises over 180k stereo images for autonomous driving under diverse driving scenarios, e.g., urban, suburban, highway, elevated, and country roads. The dataset is recorded using color cameras, stereo camera pairs, 3D laser scanner Velodyne HDL-64E S3 LiDAR, and GPS/IMU. It is also collected under different temperatures and weather conditions, such as sunny, rainy, cloudy, foggy, and dusky. The stereo image disparity labels are transformed from multi-frame LiDAR points. The dataset consists of six classes: ground, nature, construction, vehicle, human, etc. The authors used distance-aware and semantic-aware evaluation metrics for stereo-matching evaluation of farther ranges and various classes.

3.2.14. RobotCar

The Oxford RobotCar dataset [46] is recorded in central Oxford, UK, from May 2014 to December 2015 using six cameras/trinocular stereo camera, LIDAR, and GPS/GLONASS (Global Navigation Satellite System). The dataset contains over 20 million images collected in different weather, such as heavy rain, night, direct sunlight, and snow. The dataset also includes many road and construction buildings over the year of driving.

3.2.15. SYNTHIA

The SYNTHIA dataset [47] is a synthetic dataset of urban scenes for semantic segmentation, object recognition, place identification, and change detection of autonomous driving. The dataset comprises pixel-level semantic annotations for 13 classes: sky, building, road, sidewalk, fence, vegetation, lane-marking, pole, car, traffic signs, pedestrians, cyclists, and miscellaneous. This dataset is a good example of generating a synthetic dataset that can represent the real-system driving scenario.

3.2.16. ONCE

The One millioN sCenEs (ONCE) [48] dataset designed for 3D object detection in autonomous driving scenarios encompasses 1 million LiDAR scenes and 7 million related camera images. This dataset is derived from 144 h of driving data, a duration that is 20 times greater than the most extensive existing 3D autonomous driving datasets, such as nuScenes [33] and Waymo [38]. Additionally, it features a diverse collection of data acquired from various locations, times, and weather conditions.

Moreover, there are additional Radar datasets for autonomous driving, such as SeeingThroughFog [49], AIODrive [50], VoD [51], TJ4DRadSet [52], K-Radar [53] and aiMotive [54].

3.3. Attention Mechanisms and Vision Transformers

3.3.1. Attention Mechanism in CNNs

Convolutional neural networks process input features uniformly. Nevertheless, not all features hold equal importance for the network's prediction. Some parts of the input may have more semantically significant features than others. Hence, an attention mechanism becomes necessary to dynamically weigh the features based on their significance and select the most crucial part of the input. Selectively focusing on the important features improves the ability of the network to learn discriminative feature representation, which potentially reduces the computational load. Attention methods can be classified into five categories [55]: channel attention, spatial attention, temporal attention, branch channel, and hybrid attention. Hybrid attention methods combine fundamental approaches, such as channel and spatial or spatial and temporal attention. Channel attention directs attention toward specific input portions (what to pay attention to); spatial attention indicates where attention should be directed (where to pay attention to); temporal attention specifies when to prioritize essential features (when to pay attention to); and branch attention delineates which segment of the input deserves attention (which to pay attention to), primarily used in multi-branch architectures like highway networks [56].

Channel attention, spatial attention, and the hybrid channel & spatial attention are commonly employed attention methods in object detection. Among the channel attention methods, SENet [57] stands out as one of the pioneering attention modules. Channel attention can be seen as an object selection mechanism since each channel corresponds to a distinct object [58]. Each channel is regarded as a distinct feature, allowing channel attention to dynamically adjust the importance of different channels (or feature maps) within a network by learning channel-wise attention weights. The channel attention mechanism involves Global Average Pooling (GAP) to summarize feature maps across spatial dimensions, followed by fully connected or convolutional layers to generate attention weights. These weights are then applied to the original feature maps, amplifying informative channels and suppressing less relevant ones.The SENet attention module comprises two components: the squeeze and excitation modules. The squeeze module employs global average pooling to aggregate global spatial information, as shown in Figure 1a. Conversely, the excitation module utilizes fully connected layers, ReLU activation, and Sigmoid activation functions to generate an attention vector by capturing channel-wise relationships. Subsequently, this attention vector's input features are weighted to compute the attention weight. This attention module can seamlessly integrate with CNNs to acquire robust and discriminative feature representations. However, a primary limitation of this module is that global

average pooling may not capture complex global information, and the fully connected layer increases the number of parameters, potentially elevating the network's complexity. Several other attention modules, such as GSoP-Net [59], ECANet [60], FcaNet [61], and GCT [62], have enhanced the SENet module by refining either the squeeze, excitation, or both submodules.

Similarly, spatial attention can be viewed as a mechanism for selecting regions where attention is needed. Modules like GENet [63] and PSANet [64] are spatial attention methods designed using depth-wise convolution [65] and subnetworks for feature aggregation. Furthermore, attention modules that seamlessly integrate with non-local networks [66] have been introduced by GCNet [67] and A^2-Nets [68]. The channel & spatial attention module is an adaptive mechanism for selecting objects and regions to leverage the advantages of both channel and spatial attention modules. Some hybrid attention modules, such as the residual attention network [69] and SCNet [70], jointly predict channel and spatial attention networks. In contrast, Triplet attention [71], DAN [72], SCA-CNN [58], and CBAM [73] independently predict channel and spatial attention. CBAM [73] uses channel and spatial attention to improve region selection and object selection capability of backbone networks, as shown in Figure 1b.

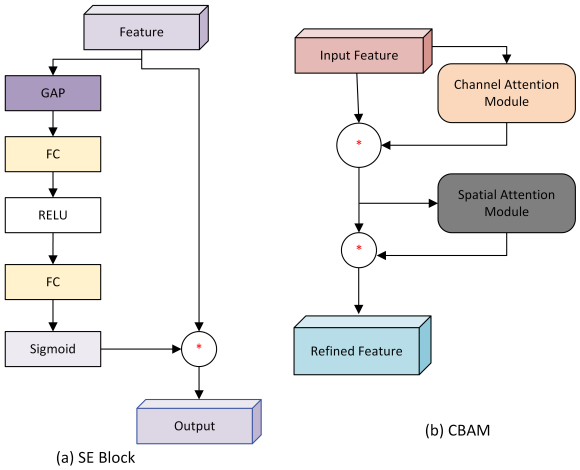

Figure 1. (**a**) Squeeze and Excitation block. (**b**) Convolutional Block Attention Module (CBAM).

3.3.2. Vision Transformers

The success of transformers in natural language processing [74] has inspired the introduction of vision transformers (ViT) [75]. The central component of ViT is the multi-head attention (MHA) mechanism, enabling parallelized attention computations. Each attention head possesses its set of learnable parameters, allowing it to capture distinct patterns within the input sequence. As illustrated in Figure 2, the scaled-dot product attention entails queries (Q), keys (K), and values (V) as inputs, with queries and keys having dimensions of d_k, and values having dimensions of d_v. Subsequently, the matrix output of the attention for each head can be calculated as follows:

$$Attention(Q, K, V) = softmax(\frac{Q_i K_i^T}{\sqrt{d_k}})V_i, \quad (1)$$

Utilizing multi-head attention in parallel and concatenating the outcomes of each head facilitates computation and boosts performance. Following each head's attention operation, the final multi-head attention can be calculated using the transformation matrix W^O.

$$MultiHead(Q, K, V) = Concat(head_0, ..., head_N)W^O, \quad (2)$$

where N is the number of heads and Concat(.) is the concatenation operation. While vision transformers have revolutionized computer vision, these networks still have shortcomings

that need to be addressed to enhance their performance for real-time applications. In multimodal fusion, substituting the transformer network's encoder with a voxel encoder network is a common practice to alleviate the computational burden. This voxel encoder proves particularly effective for handling voxel features [76] in multimodal fusion tasks conducted within the voxel representation.

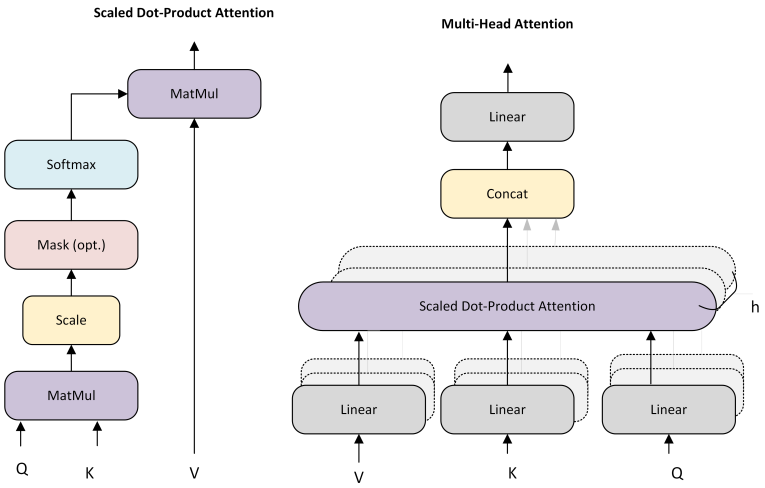

Figure 2. (**left**) Scaled Dot-Product Attention. (**right**) Multi-head attention consists of several attention layers running in parallel.

What are the Shortcomings of Vision Transformer Networks?

ViTs generally require more computational resources and memory than CNNs. This problem is more pronounced when high-resolution images are used. The attention calculation specifies a matrix multiplication between Q and K, as specified in (1). For an input feature map $X \in \mathbb{R}^{H \times W \times C}$, this matrix multiplication incurs a computational cost of $O(H \times W \times H \times W = H^2W^2C)$. Here, $H \times W$ denotes the number of input image patches (n) in images processed as a sequence of patches. Therefore, the computational complexity of multi-head attention for an input feature map $X \in \mathbb{R}^{H \times W \times C}$ exhibits quadratic growth with respect to the number of input patch sizes, scaling as $O(n^2)$. This high computational burden is a bottleneck for vision transformer-based networks.

Furthermore, the performance of ViTs in dense prediction tasks, like object detection and segmentation, is compromised when trained on small datasets, mainly due to their limited inductive bias. In contrast, CNNs possess distinct inductive biases, such as local connectivity and shared weights, which are not inherent in ViTs. Local connectivity in CNNs is based on the assumption that neighboring pixels are related, allowing the network to efficiently detect local features like edges and patterns using small, localized kernels. Similarly, the shared weight bias enables the consistent application of the same kernel across the entire image, facilitating the detection of significant features regardless of location. On the other hand, ViTs lack such inherent biases, treating all input tokens equally. This characteristic allows ViTs to adapt to various datasets and tasks but requires large datasets to establish relationships among features effectively. Consequently, ViTs are data-hungry and might not converge swiftly with small datasets, potentially leading to generalization problems and overfitting in such scenarios. To fully leverage the strengths of ViTs, including their capability to detect long-range dependencies, it is essential to address their shortcomings (refer to potential research directions in Section 5).

Since the introduction of the first ViT [75], numerous vision transformer networks have been proposed. Some of these networks are specifically designed to handle sparse data, such as LIDAR point clouds. Examples include SST [77], RoITr [78], FlatFormer [79], PCT [80], PointASNL [81], PointTransformer [82], Fast Point Transformer [83], Voxel Transformer [84], Voxel Set Transformer [85], Point Transformer v2 [86], Point Transformer [86], SST [77], RoITr [78], and Swformer [87].

4. Multimodal Fusion 3D Object Detection Methods

4.1. CNN Multimodal Fusion 3D Object Detection

An autonomous driving perception system must meet three key criteria [3]: Accuracy: To provide error-free insights into the surrounding environment. Robustness: To function effectively in adverse weather conditions and in the event of sensor failures. Timeliness: To support high-speed driving with real-time processing. Attaining these objectives is possible by using multiple sensors and capitalizing on their strengths. Although most fusion methods combine image and LiDAR data, only a handful integrate radar point clouds with image and LiDAR data. Sensor fusion advances the development of a more resilient system by employing redundant data from various sensors. However, processing these diverse data representations is a complex challenge. Fusion techniques can be classified into three categories [5]: early (data) fusion, middle (feature) fusion, and late (decision) fusion, as depicted in Figure 3.

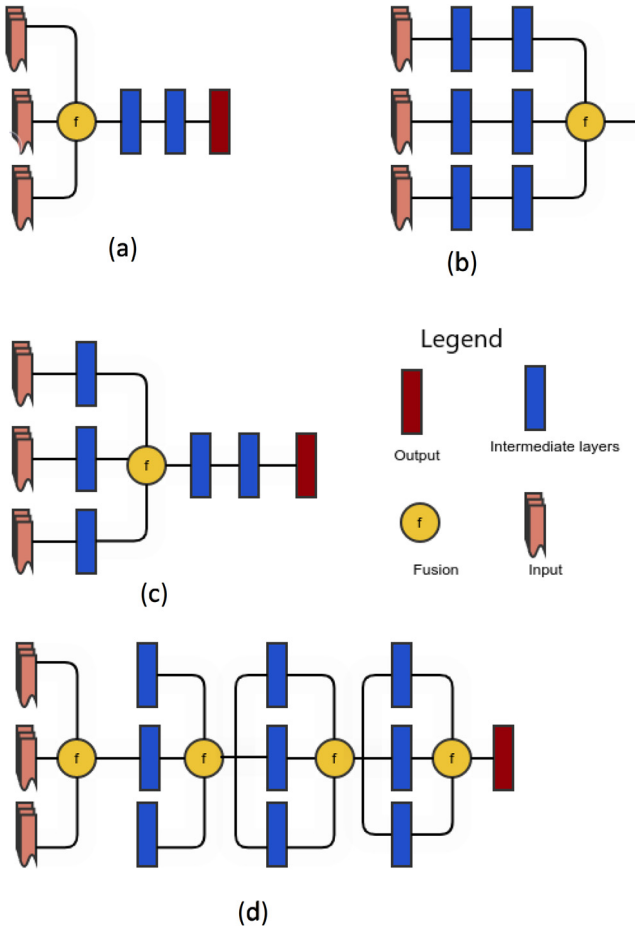

Figure 3. Different fusion techniques: (**a**) Data (Early) fusion, (**b**) Decision-level (Late) fusion, (**c**) One layer middle (feature) fusion, and (**d**) Deep (Feature) fusion.

4.1.1. Data (Early) Fusion

Early fusion incorporates all sensor data from the onset of operations, presenting a challenge when modifications are needed within a specific portion of the fusion network. One often has to overhaul the entire network to adjust any part of the fusion system. Additionally, sensor data discrepancies, such as misalignment, varying data representations, and differing sparsity levels, compound the complexity of early fusion. Common techniques for fusion include concatenation and summation methods. Summation-based fusion, mainly when designed similarly to the identity mapping technique in ResNet [88], can alleviate computational demands. However, this method may lead to information loss due to the reduction in feature map width that can occur during summation. However, while concate-

nation enlarges the width of feature maps and broadens the range of inputs, it also leads to a higher computational load. Beyond direct sensor data fusion, determining the relative importance of features for specific tasks could be pivotal, especially during sensor failure. Although quantifying the input contribution to a given application is complex, designing a model capable of such feature weighting is essential for maintaining performance when sensors malfunction.

The AVOD network [6] integrates LiDAR point clouds with camera RGB images to produce features utilized by both an RPN and a subsequent detector network. As inputs traverse through the network layers, a pooling operation reduces their size, posing a challenge for detecting small objects. AVOD comprises an encoder-decoder network within its feature extraction process to address this issue. The encoder AVOD uses is the same as that in MV3D [5] (details of which are provided in the section on other fusion techniques Section 4.1.4). AVOD introduces a bottom-up decoder that employs FPN [89] to upsample the feature map to the original size of the input. The RPN employs an early fusion technique to merge feature maps from images, a 3D anchor grid, and BEV, generating high-recall 3D proposals. AVOD has introduced a novel bounding box encoding that utilizes four corners and two heights. MV3D, on the other hand, faced significant limitations, the primary being its reliance on the assumption that the orientation vector points towards the longer side of the box, which becomes problematic for objects like pedestrians that do not conform to this rule. Moreover, MV3D's orientation information was only accurate up to an additive constant of $\pm\pi$ radians [6], leading to potential ambiguities in corner order and orientation details during corner matching. AVOD addresses these challenges using a regressed orientation vector represented by $\sin(\theta)$ and $\cos(\theta)$. The performance of AVOD on the KITTI dataset surpasses previous efforts such as MV3D [5] in terms of accuracy, detection speed, and memory efficiency.

Wang et al. [90] proposed an early fusion method for a BEV of LiDAR data and front view of camera images to generate 3D proposals. The proposed sparse nonhomogeneous pooling layer is used to build an efficient fusion method by mapping features between BEV and front view features. The result on the KITTI [31] dataset outperforms AVOD [89] for pedestrian detection. Liu et al. [91] proposed a 2D-driven 3D object detection model by fusing LiDAR data and RGB images. The developed scoring mechanism using prior knowledge gives a score for each object to select the target object. The model was tested using the Hong Kong Airport [91] dataset. Meyer et al. [92] proposed a radar and RGB image fusion 3D object detection network. The network is like AVOD [6], but FPN is not used as a feature extractor. It trained with the dataset collected from West Germany [39]. The point cloud radar is projected onto BEV with six height maps and one density map, but the Doppler information of the point cloud data is not used. A modified VGG network [93] is used to generate 3D object proposals from camera images and BEV representation. The output on the dataset shows that radar and camera fusion perform better than LiDAR and camera fusion.

Nobis et al. [94] put forth RadarVoxelFusionNet (RVF-Net), a fusion model combining LiDAR, camera, and radar data for 3D object detection. LiDAR points are projected onto the image plane and combined with camera images to create a simulated depth camera, generating 3D data points. Subsequently, the radar, LiDAR, and simulated depth camera information are merged within a discrete voxel framework before the main network processes them. To efficiently process the sparse voxel grid input, sparse 3D submanifold convolutions are employed. The resultant sparse tensors form the BEV representations inputted into the detection head. This detection head consists of a classification head that delivers class scores for each anchor box, a regression head for bounding box localization, and a direction head that provides additional classification values for yaw angle estimation.

The heterogeneity of raw data from multiple sensors, such as the pixel values of RGB images and the sparse and continuous data from LiDAR, along with dimensional disparities, presents a significant challenge for early fusion. Before fusing data sources, they need to be transformed into a single, synchronized feature vector. Managing data

from different sources during early fusion becomes complex without this conversion. Furthermore, disparities in the sampling rates of sensor recordings can complicate early fusion, sometimes necessitating data discarding to standardize it with other sensors, which can result in information loss. Despite these challenges, studies like Nobis et al. [94] have successfully achieved a direct fusion of camera, LiDAR, and radar inputs without preliminary processing. A robust model is required to manage the discrepancies in input data effectively and accomplish direct fusion within the network.

The early fusion technique combines data from various sources or sensors at the beginning before further processing occurs. This integration happens at the input level, which allows the model to have a complete and comprehensive view of the combined data right from the start. By combining different data sources early, the approach aims to create a unified representation that the model can use directly for subsequent processing. From the beginning, this approach aids in creating connections and context among various data streams, which may lead to a thorough understanding and better decision-making. Despite the comprehensive data integration at the input stage, early fusion methodologies have challenges. These may include potential noise amplification or information loss from different sensor sources. The challenges associated with fusion techniques are further explored in Section 5.

4.1.2. Feature (Middle) Fusion

Divergent data representations across sensors pose a significant challenge to early fusion techniques. Moreover, early fusion lacks the flexibility to accommodate updates at specific fusion levels; any change to a segment of the fusion network often necessitates a complete overhaul. In contrast, late fusion processes data from different sensors separately until the designated fusion layer, potentially underutilizing the information available from multiple sensors. Middle (feature) fusion, on the other hand, whether as a single-layer or deep fusion process, is crucial for effectively harnessing features from diverse sensors. A well-executed middle fusion approach can address the limitations inherent in early and late fusion strategies. Nonetheless, there are instances where early and late fusion methodologies surpass feature fusion in performance, as detailed in the open issues Section 5.

PointFusion [95] is a 3D object detection network that employs a dense fusion strategy for combining LiDAR data with RGB images. This network is comprised of three main components: PointNet [96] for processing LiDAR data, ResNet [88] for processing RGB images, and a fusion network that merges the two processed outputs to predict 3D bounding boxes. Two fusion architectures are proposed within PointFusion: a vanilla global fusion that directly regresses the locations of box corners and its dense fusion mechanism, which predicts the eight corner locations of a 3D box relative to each 3D input point. A scoring function then selects the optimal detection by leveraging spatial anchors [97] and dense prediction techniques [98]. Liang et al. [99] introduced a new ContFuse model that seamlessly combines LiDAR data points with RGB images through a continuous fusion network. They utilized a multilayer perceptron as a parametric kernel function for continuous convolution [100] to extract image features and map them onto BEV. The network then fuses these projected multiscale image features, similar to the FPN [89] representation, with LiDAR point cloud data. Their continuous convolution, employing an MLP, directly produces features without requiring explicit storage in GPU memory. A lightweight version of ResNet18 [88] serves as the backbone for the image network. The results from this network surpass those from MV3D [5], AVOD [6], and other previous studies on the KITTI dataset [31].

HDNET [101] is a single-stage 3D object detector that extracts geometric and semantic features from the high definition (HD) maps and fuses with BEV representation to improve the 3D object detection performance. The model tested on the KITTI [31] dataset and TOR4D [102] benchmark dataset. HD map is not part of the detection module. So, it is not an optimized network. Liang et al. [103] developed a feature fusion network from

LiDAR data and RGB imagery to learn an end-to-end architecture for multiple tasks: 2D and 3D object detection, ground estimation, and depth completion. The model also uses a geometric map like HDNET [101], but it is part of the detection module to be trained end-to-end. The image network uses pretrained ResNet-18 [88] as a backbone, whereas the LiDAR stream uses a customized residual network. Both networks use FPN [89] for multiscale feature representation. Point-wise feature fusion is done between multiscale image features and BEV of LiDAR data. The 3D bounding box is generated from the final BEV feature map.

Du et al. [104] developed a generalized network for 3D SUV, Sedan, and Van vehicle detection. The 2D bounding boxes for each candidate vehicle are generated and fused with 3D point clouds to generate 3D box proposals following the random sample consensus (RANSAC) algorithm. Then, the three 3D box proposals of the candidate vehicles are compared to select the highest score boxes that are input to the next two-stage refinement network. The final two-stage refinement network carries out 3D box regression and classification to improve the detection accuracy. RoarNet [105] is a fusion method for RGB images and LiDAR data. Faster R-CNN [106] and Fast point R-CNN [97] are used for RGB image and pointNet [96] for LiDAR point as backbone networks. Inspired by [107], RoarNet uses a monocular image to generate a 3D pose of objects and 2D bounding boxes. The geometric agreement search and spatial scattering generate 3D object region proposals. Finally, the 3D bounding box regression is done using point clouds. LaserNet++ [108] presented a method for fusing LiDAR point cloud and RGB images for joint 3D object detection and semantic segmentation tasks. This work extends the LaserNet [109] LiDAR-based 3D object detection model. Each LiDAR point is projected onto the RGB image plane and fused with the extracted features using ResNet CNN [88]. It passed the fused features to the LaserNet model to train the model jointly for 3D object detection and semantic segmentation. The fusion of RGB images over the original LaserNet model improves the performance. Processing 3D object detection and semantic segmentation jointly reduce the computation and latency.

Lim et al. put forth FusionNet [110] 3D object detection model for autonomous driving. The radar feature extractor processes the radar range-azimuth image, and the camera feature extractor network processes the images before fusing. The training is done by partially freezing and fine-tuning the network to make sure the network learns from different signal sources. The fused features feed to the SSD [111] detection head for classification and regression. The *K*-means clustering is used to construct a set of anchor boxes for the car. This car network is trained and evaluated using the dataset collected from California highways. Zhou et al. [112] introduced a multiview fusion network from LiDAR BEV and perspective view at point level for 3D object detection. The BEV encoding technique uses vertical column voxelization like Pointpillars [113]. The proposed dynamic voxelization (DV) gives four advantages over hard voxelization [113,114]. (1) It does not need to sample points for each voxel because every point uses the model. (2) It avoids padding of voxels to a predefined size like VoxelNet and PointPillars. (3) It yields deterministic voxel embeddings. (4) It is important for a point-level fusion of context information from multiple views. The results on the Waymo [38] dataset and KITTI [31] dataset show performance improvement. PointPainting [115] is a sequential fusion method for 3D object detection. First, the authors use a semantic segmentation network for pixel-wise segmentation scores of images. Then, the LiDARs are projected onto the image plane and masked by the segmentation mask with a pixel-wise segmentation score. The PointPainting model shows performance improvement over VoxelNet [114], PointPillars [113], and PointRCNN [116] LiDAR-based methods.

Similarly, Wang et al. [117] introduced a two-step fusion of fusion network (FoFNet) for 3D object detection. It transformed the LiDAR data into voxel representation by averaging the vector representation of the points within a small voxel to avoid extra structure in transforming to voxels. Then, sparse convolution transforms the voxelized point clouds into 2D feature maps. In the first fusion step, features of multiple scales are fused using

a pyramidal network. In the second step, sizes are made uniform using deconvolution followed by channel-wise concatenation. The result shows that FoFNet fusion improves performance. MVAF-Net [118] proposed a single-stage multiview adaptive fusion network for 3D object detection. The architecture has three parts: a single view feature extraction (SVFE) to extract spatial features from BEV, range view (RV), and image in a point-wise manner; the multiview feature fusion (MVFF) to handle the feature fusion of the extracted features; and fusion feature detection (FFD) for box regression and classification. The MVAF-Net effectively fuses multiview features of BEV, RV, and image. FuseSeg [119] developed a fusion-based 3D object detection model by extending SqueezeSeg [120] model. The RGB and range image features are concatenated after feature warping using calibration to do the segmentation task. The model runs at 50 fps and improves the performance of SqueezeSeg up to 18%.

Likewise, CrossFusion net [121] is a two-stream BEV of point clouds and RGB images-based fusion 3D object detection model. The proposed CorssFusion layer manages the fusion between BEV and RGB image features and transforms feature maps of one stream to another based on the spatial relationship. An attention mechanism is applied to the transformed and original features to estimate the importance of feature maps from the two streams. RoIFusion [122] is a fusion of LiDAR point cloud and RGB images method for 3D object detection. RIOFusion architecture has three parts. The first part is the fused keypoints generation (FKG) layer. The point-guided and pixel-guided keypoint generation layers extract keypoints from point clouds and RGB images, respectively. Then, the FKG layer aggregates the key points before feeding into the second layer (ROI fusion layer). The ROI fusion layer generates geometric and texture features from 3D and 2D ROI. Finally, the 3D and 2D ROI features are fused for 3D bounding box prediction. The final prediction layer predicts the class category and the bounding box.

3D-CVF [123] is a cross-view spatial feature fusion method for 3D object detection. The model uses the auto-calibrated feature projection method to transform the camera-view features into dense BEV feature maps. A two-stage fusion is done: first, a gated fusion network is applied with spatial attention maps to aggregate the camera and LiDAR features. Then, the camera-LiDAR feature fusion is done using the 2D camera view features generated via 3D ROI pooling and the BEV feature for the proposal refinement stage. The result shows an improvement over the single-sensor implementation. Wang et al. [124] developed a high dimensional frustum PointNet fusion method using raw data from the camera, LiDAR, and radar for 3D object detection. The network consists of three parts: frustum proposal, 7D PointNet for radar point clouds, and 8D PointNet for radar point clouds. The same two-stage methods as F-PointNet [125] are used to generate 2D Frustum proposals, but they are generated from camera video (6Hz) instead of a single image. The 7D pointNet for LiDAR point cloud is generated using the same principle as of PointNet [96] and PointNet++ [126] with some modification. The 7D pointNet data structure comprises *XYZRGBT*, where *XYZ* denotes the Euclidean coordinates, RGB color information, and *T* a time stamp. Similarly, an 8D pointNet is designed for radar point clouds that can learn eight points, such as coordinates, velocity, dynamic property, false alarm probability, and validity states. This work is a good starting point to fuse LiDAR, radar, and camera.

GRIF Net [127] is a gated region of interest fusion network exploiting the data from radar and monocular images for 3D vehicle detection using the nuScene [33] dataset. FPN [89] and sparse block network (SBNet) [128] are used as an image backbone network and a radar backbone network, respectively. The fusion network adaptively fuses the two ROIs from the camera and radar to generate 3D proposals and regress 3D bounding boxes for vehicles. Most fusion methods combine distinct features in a model using concatenation, addition, or average. However, the GRIF Net fusion network uses a convolutional Mixture of Experts (MoE) by explicitly assigning weights to the camera and radar features. The detection performance on the nuScenes [33] dataset shows a good result on the vehicles with very sparse radar points and effective detection of vehicles from long distances. CenterFusion [129] introduced a method that exploits radar and RGB images for 3D object

detection. The center point detection network is used to identify the center points of objects in the image. Then, the association between the radar detection and the corresponding object's center point is done using a frustum-based method. The method also generates features to complement image features and regress bounding box properties, such as depth, velocity, and rotation. The model was trained and tested on the nuScenes [33] dataset.

LiDAR-SLAM [130] proposed a frustum-based probabilistic fusion network for 3D object detection. The visual and range information is combined into a frustum-based probabilistic framework to solve the sparse and noise problem. The 3D region probabilities are generated using the Simultaneous Dynamic Triangulation Mapping (SDTM) framework. Then, a descriptor-free method (G-PO) is applied for 3D bounding box prediction. SAANet [131] presented an adaptive spatial alignment (SAA) fusion network to establish automatically and align the correspondence between the point cloud features and image features. Sparse convolution is applied to the 3D voxels of point clouds to learn point cloud features. The image features are extracted using the ResNet [88] network. Then, the point cloud features and image features are fused using the SAA network. Finally, the 3D bounding boxes and classification are predicted. The results on the KITTI [31] dataset show average precision and inference time improvement over other methods. EPNet [132] put forth a fusion network using LiDAR and camera images for 3D object detection by solving the inconsistency between localization and classification confidence using consistency enforcing loss. The EPNet fusion network is a LiDAR-guided image fusion (LI-Fusion) network that fuses the row point cloud features and the image features and finds the point correspondence between LiDAR data points and image features in a point-wise manner. The network adaptively estimates the importance of semantic image segmentation features in enhancing point features. The network outperforms the state-of-the-art methods on the KITTI dataset and SUN-RGBD dataset.

Gao and Hu put forth MVDCANet [133], a 3D small object detection method based on multiview feature fusion. BEV and cylinder view (CYL) features are extracted from the point cloud for each point. The dual-channel feature extraction (DCFE) module is used for the point-wise fusion of extracted features from BEV and CYL before feeding to the point-wise attention feature encoder (PWAF) module. A pseudo-image (pillar) feature map is created by creating a series of voxels and mapping the point features extracted by the attention feature extraction (DCAFE) module, consisting of DCFE and PWAF modules. Then, the DO-Conv [134] module is applied to process the pseudo-image in the BackboneNet detection head. Finally, the classification and regression module predicts the detection results. Miao et al. put forth PVGNet [135], a one-stage point-voxel-grid (PVG) network for 3D object detection. Authors voxelized point clouds followed by feature encoding to nonempty voxels. The instance-aware focal loss is used to minimize the effect of an unbalanced class. The backbone network comprises sparse 3D and 2D convolution to extract multiscale voxel and grid-wise BEV features. The PVG fusion network fuses point, voxel, and grid-level features in a point-wise manner. Then, these features are used for point-wise segmentation and regression of the 3D bounding box for each foreground point. The final detection results were generated using a group voting-based method instead of Non-maximal suppression to merge redundant boxes. The experimental result on the KITTI dataset [31] shows that the model outperforms previous models, such as SASSD [136].

Xu et al. presented FusionPainting [137], a multisensor fusion model at a semantic level with adaptive attention for 3D object detection. The semantic segmentation can be generated from the point cloud and 2D image using the 3D and 2D segmentation network. Then, the two segmentation results fused at the semantic level using the proposed adaptive attention module. Finally, the output of the adaptive attention module is fed to the 3D detectors for 3D object detection. The model shows a promising result on the nuScenes [33] dataset. Moreover, Zhang et al. proposed Faraway-Frustum [138], a LiDAR and RGB image fusion 3D object detection model to address the problem of distant object detection. Frustums in the point cloud space are generated for each object and the corresponding LiDAR points using the 2D instance semantic segmentation masks. Then,

the 3D centroid of the object is estimated using the point cloud clustering method. If the centroid is greater than the faraway threshold, the object is treated as a distant object, and the 3D bounding box is regressed using the proposed Faraway Frustum-Network (FF-Net). Learning representations are used for 3D bounding box regression if the object is not a faraway object. The model shows comparable performance on the KITTI [31] dataset.

Middle fusion provides enhanced flexibility by enabling the integration of features at multiple stages within the model. This approach encompasses a spectrum ranging from single-layer to deep fusion, as demonstrated in Figure 3. Such a technique facilitates the transformation of input data into more abstract feature representations across various layers during the training phase, empowering the model to effectively utilize data at each network stage. Despite its consistent data utilization throughout the training process, some instances have revealed certain early fusion models surpassing the performance achieved by middle fusion methodologies. Ascertaining the optimal fusion method tailored to specific tasks remains an unresolved challenge. The complexities inherent in different fusion techniques are elaborated upon in Section 5.

4.1.3. Decision (Late) Fusion

Late fusion is a technique where features obtained from multiple sensors are processed independently until they merge at the fusion point. This approach avoids the data representation challenges often encountered in early fusion, as it involves transforming the inputs into high-level features in the preceding layers before fusion. Consequently, the features employed in late fusion undergo a more advanced transformation than those in middle or early fusion methods. This technique, also known as decision fusion, uses its output for critical functions such as classification and localization. This section reviews late fusion methodologies in 3D object detection specifically for autonomous driving.

Dou et al. [139] enhanced 3D vehicle detection performance using a semantic segmentation network (SEG-Net) and an improved version of VoxelNet. The SEG-Net segment of their model employed MobileNet [140] as the foundational recognition and segmentation network. This network incorporated Spatial-Channel Squeeze and Excitation (SCSE) [141] alongside Receptive Field Block (RFB) [142] for semantic segmentation, offering both spatial information and an extensive receptive field. The enhanced VoxelNet combined the semantic map with LiDAR data to predict 3D vehicle bounding boxes. Lu et al. [143] introduced SCANET, a two-stage spatial-channel attention network for 3D object detection. They projected LiDAR point cloud data into BEV representation. The network featured a Spatial-Channel Attention (SCA) encoder and an Extension Spatial Upsample (ESU) decoder for generating 3D region proposals. A modified VGG-16 [93] integrated with SCA was used to extract multiscale and global context features, producing spatial and channel-wise attention for selective feature discrimination. The ESU decoder module enhanced the resolution of information lost in successive pooling operations. 3D proposals were then projected onto BEV and RGB image planes, with the model's multilevel fusion method combining region-based features before predicting 3D bounding boxes. The model's effectiveness was assessed using the KITTI [31] dataset. A comparison of commonly used fusion methods is presented in Table 4.

Table 4. BEV and 3D performance comparison of fusion based 3D object detection methods on the KITTI [31] test benchmark. E stands for easy, M for moderate, and H for hard.

Methods	AP_{BEV} (%)									AP_{3D} (%)								
	Car			Pedestrians			Cyclists			Car			Pedestrians			Cyclists		
	E	M	H	E	M	H	E	M	H	E	M	H	E	M	H	E	M	H
MV3D [5]	86.0	76.9	68.5	-	-	-	-	-	-	71.1	62.4	55.1	-	-	-	-	-	-
PointFusion [95]	-	-	-	-	-	-	-	-	-	77.9	63.0	53.3	33.4	28.0	23.4	49.3	29.4	27.0
AVOD-FPN [6]	88.5	83.8	77.9	58.8	51.1	47.5	68.1	57.5	50.8	81.9	71.9	66.4	50.8	42.8	40.9	64.0	52.2	46.6
ContFuse [99]	88.8	85.8	77.3	-	-	-	-	-	-	82.5	66.2	64.0	-	-	-	-	-	-
MVX-Net [144]	89.2	85.9	78.1	-	-	-	-	-	-	83.2	72.7	65.2	-	-	-	-	-	-
SAANET [131]	-	-	-	-	-	-	-	-	-	83.7	73.9	66.8	38.6	32.7	31.5	64.3	50.9	45.1
VPFNet [145]	-	-	-	-	-	-	-	-	-	88.5	81.0	76.7	54.7	48.4	45.0	77.6	64.1	58.0
RoIFusion [122]	-	-	-	-	-	-	-	-	-	88.3	79.5	74.5	-	-	-	-	-	-
3D-CVF [123]	-	-	-	-	-	-	-	-	-	88.8	79.7	72.8	-	-	-	-	-	-
FS-Net [146]	-	-	-	-	-	-	-	-	-	88.7	82.1	77.4	49.8	43.3	40.9	81.8	68.4	60.9
DenseVoxel [147]	-	-	-	-	-	-	-	-	-	91.0	82.4	77.4	-	-	-	-	-	-
CL3D [148]	-	-	-	-	-	-	-	-	-	87.5	80.3	76.2	47.3	39.4	37.0	77.3	62.0	55.5

Pang et al. put forth CLOCs [149], a Camera-LiDAR object candidates fusion 3D object detection model. Correctly detected objects have the same bounding boxes. Therefore, geometric consistency is built between 2D and 3D detectors to avoid false detections. Similarly, semantic consistency association is used to fuse detection only from the same category. A combination of 2D detectors, such as RRC [150], MS-CNN [151], and Cascade R-CNN [152] and 3D detectors, such as SECOND [153], Pointpillars [113], PointRCNN [116], and PV-RCNN [154] is used as a fusion network to show the flexibility of the model on the KITTI [31] dataset. Wang et al. [145] presented a Voxel-Pixel Fusion Network (VPFNET), which takes the geometric relation of a voxel-pixel pair and fuses the corresponding features. The LiDAR voxelized and image features are fused using the Voxel-pixel fusion layer. A cross-attention mechanism between the reweighted voxel and reweighted pixel features enhances the fusion performance.

Zhang et al. proposed the RangeLVDet [155] model for 3D object detection by fusing LIDAR range image and camera RGB image. Darknet-53 [156] is used as an RGB feature extraction network, and YOLOv3 [157] as a 2D detection framework. The high-level RGB semantic and range image features are converted to the point view and fused by point-wise concatenation. The point view helps each point get the corresponding high-level features of the range and RGB images. The point view can be easily converted to BEV by projecting the 3D point to the x-y plane. Finally, the 3D bounding boxes are predicted from the fused BEV features using 2D convolution-based RPN. Even though the model outperforms LiDAR-only based models, such as SECOND [153], and Pointpillars [113] on their dataset, the authors did not test the model on a public dataset.

In the late fusion approach, data from diverse sensors undergo separate processing until combined at the fusion stage. This strategy aims to minimize errors caused by discrepancies in the data and has been shown to improve performance compared to early fusion. Its capability is restricted since it separates data streams until the last fusion stage, failing to fully leverage the potential of various sensor inputs throughout the processing pipeline. While some instances demonstrate late fusion outperforming early and middle fusion techniques in certain tasks, its superiority over other fusion methodologies lacks conclusive evidence. This underscores that late fusion might require further refinement and enhancement to match or exceed the performance achieved by early- or middle-fusion strategies. Finding the best fusion technique that works optimally is still an ongoing challenge. Section 5 provides unresolved questions aiming to explore and elucidate the most effective sensor fusion techniques, a task as intriguing as motivating.

4.1.4. Other Fusion Techniques

This section discusses works that employ multiple fusion techniques. MV3D [5] implements a fusion of LiDAR point cloud data and RGB images for 3D object detection. This approach includes a 3D RPN that builds upon Faster R-CNN [97] and incorporates a region-based fusion network. A modified version of the VGG-16 [93] network serves as the feature extraction backbone. The 3D proposal network, using a BEV representation of the point cloud, generates 3D object proposals and projects these into BEV proposals, front view (FV) proposals, and image (RGB) proposals. Subsequently, the region-based fusion network combines each view proposal through Region of Interest (ROI) pooling, facilitating the joint prediction of object classes and 3D box regression. ROI pooling ensures feature vectors of consistent length for each view. In comparing early, deep, and late fusion techniques, MV3D found the deep fusion approach to yield the best results. However, a notable limitation of MV3D is its reduced effectiveness in detecting smaller objects.

Likewise, MVX-Net [144] used PointFusion and VoxelFusion approaches for 3D object detection. A pre-trained Faster RCNN [97] is used as an image feature extractor network. The voxel architecture processes the output of pointFusion, an early fusion between the projected LiDAR points onto the image plane and the extracted image features using Faster RCNN at the point level. Then, the voxel feature extractor pooled and encoded these features to feed the 3D RPN to generate 3D bounding boxes. Nonempty 3D voxels are projected onto the image plane and fused with the image features using voxelFusion at the voxel level (late fusion). The VoxelFusion performance is lower than PointFusion's performance. However, VoxelFusion is better than PointFusion in memory usage. Kuang et al. [158] put forth a multi-modality cascaded fusion model for autonomous driving. The fusion model is composed of two parts: intra-frame and inter-frame fusion. The intra-frame fusion module fuses each frame's RGB image and radar detection. Then, the inter-frame fusion performs a homogeneous and heterogeneous association between the tracklets at time $t-1$ and the fused detection at time t. Accurate object trajectories are generated from this association. Decision-level and feature-level fusion are combined for robust detection. The model is evaluated on the nuScenes [33] dataset.

Xie et al. put forth PI-RCNN [159], a point-based attentive Cont-Conv fusion (PACF) 3D object detection module. This fusion network is different from deep-continuous fusion [99] and multitasks multisensor fusion [103] because it handles point-wise continuous convolution directly on the LiDAR points rather than projecting the LiDAR points into BEV. Based on the PACF module, Point-Image RCNN (PIRCNN), a multisensor multitask module that combines image segmentation and 3D object detection, is proposed. The PI-RCNN comprises two sub-networks: the segmentation sub-network and the point-based 3D detection sub-network. The segmentation sub-network extracts semantic features from RGB images, whereas the point-based 3D detection sub-network generates and refines 3D proposals from raw LIDAR points. The PACF module fuses semantic features from RGB images with raw LIDAR points features. They conducted early (data) fusion and middle (feature) fusion. Additionally, two operations are added to make the fusion more robust. The experimental result on the KITTI [31] dataset shows that feature fusion performs slightly better than data fusion.

Jiao et al. [160] proposed an extrinsic uncertainty-based two-stage multi-LiDAR 3D object detection (MLOD) model to predict the states of an object. The first stage uses SECOND [153] to generate 3D object proposals with three fusion schemes: input, feature, and decision fusion. Then, the PointNet [96] network is adopted at the second stage to handle extrinsic uncertainty and box refinement. This network is used to eliminate highly uncertain points. Uncertainties are quantified using the trace [161] covariance quantifying method. Extra features, such as the scores and parameters of each proposal, are concatenated with the features extracted using the SECOND [153] network. Finally, classification scores and refinement are done using fully connected layers. The model is evaluated on the LYFT [40] dataset.

A single sensor often fails to perform optimally across all conditions and tasks, underscoring the need to employ multiple sensors in 3D object detection for AVs. The redundancy provided by data from various sensors is essential for mitigating potential failures due to adverse weather or limited sensor information. However, sensor fusion systems can generate vast amounts of data, necessitating significant computational power for processing. This vast amount of data can lead to increased processing and training time, and the quality of data and annotations can adversely affect the performance of supervised multisensor fusion in 3D object detection. Before deploying models into real-world systems, they must undergo additional steps, such as evaluating interpretability and resistance to adversarial attacks. Converting multisensor data into a unified frame of reference, particularly in early fusion, poses another challenge. It is essential to carefully evaluate the possibility of sensor failure and the necessary measures to address it before deploying the models. The challenges associated with sensor fusion, including adversarial attacks and interpretability, fall outside the scope of this survey. Additionally, most models are trained and tested on the KITTI dataset, which is imbalanced and not fully representative of real-world scenarios. Many KITTI datasets focus on cars, with less representation of pedestrians and cyclists, potentially biasing detection decisions during network training.

Table 5 offers a comparative analysis of fusion-based 3D object detection models, focusing on the inputs used for training and evaluation, dataset types, fusion techniques, and mAP. Most models combine RGB images and LiDAR data, with few numbers combining RGB images with radar or integrating RGB images, radar, and LiDAR. Developing an efficient fusion object detection network is crucial for building effective pipeline models in autonomous driving, including tasks like path planning and localization.

Table 5. Fusion methods comparison based on the mean average precision (mAP), fusion techniques, and input types. NDS is the nuScenes dataset evaluation metric. The custom dataset is collected from California highway [110].

Method	Dataset	Input	Fusion Type	mAP-BEV/NDS (%)	mAP-3D/mAP (%)
MV3D [5]	KITTI [31]	RGB image& LiDAR	early, middle, late	-	-
PointFusion [95]	KITTI [31]	RGB image& LiDAR	early	-	40.13
AVOD-FPN [6]	KITTI [31]	RGB image &LiDAR	middle	64.03	55.63
SAANET [131]	KITTI [31]	RGB imag & LiDAR	middle	-	52.5
3D-CVF [123]	KITTI [31]	RGB image &LiDAR	middle (gated based)	-	-
CenterFusion [129]	nuScenes [33]	RGB image & radar	middle	45.30	33.20
RVF-Net [94]	nuScenes [33]	RGB image, radar, & LiDAR	early	54.86	-
FusionNet [110]	custom [110]	RGB image & radar	early	-	73.5
Meyer et al. [92]	Astyx [39]	RGB image & radar	late	-	48.0
VPFNet [145]	KITTI [31]	RGB image & LiDAR	middle	-	64.48

4.2. Transformer-Based Multimodal Fusion 3D Object Detection

Because of the transformer's ability to manage long-range dependencies and adapt to diverse datasets and tasks, vision transformation has garnered attention in multimodal fusion for 3D object detection tasks. Some of these approaches employ CNNs for feature extraction and incorporate transformers for intermediate processing and the detection head [76,162–166]. In contrast, others develop end-to-end Transformer-based multimodal and multi-task fusion networks [167].

One of the primary challenges in multimodal fusion arises from the differing input representations of various sensors, such as cameras and LiDAR. Achieving a unified representation is essential before fusing features from these multiple sensors. One approach to achieving this fusion is by converting the camera image into a voxel representation, sampling the LiDAR point cloud into voxels, and then merging the features within this unified voxel representation, as demonstrated in methods like UVTR [165] and multisensor fusion [76]. These techniques typically employ a 2D backbone for processing images and 3D backbones for handling LiDAR point cloud data to extract respective features. These extracted features are subsequently transformed into a unified voxel representation before fusion. Following this, a choice can be made between applying a transformer encoder or a voxel encoder before the transformer decoder. However, it's worth noting that employ-

ing both a transformer encoder and decoder may increase the network's computational demands. As a result, voxel encoders are frequently preferred over transformer encoders to enhance local feature interactions, given their demonstrated effectiveness in handling voxels in multiple 3D object detection tasks. Then, the transformed decoder takes inputs from the encoders to generate predictions and bounding box information for each class. Generally, due to the sparse nature of 3D point clouds, voxels remain empty for many volumes. Consequently, handling these vacant cells during processing reduces overall performance [2]. Another drawback of the voxel representation lies in its utilization of 3D convolution, which increases computational expenses.

Another method of avoiding input representation is projecting the point cloud data into BEV representation, converting image features to BEV, and then fusing features from multiple sensors in a unified BEV after the features are extracted [162–165,167]. Projecting the point cloud data into BEV keeps geometric structure and semantic density. However, it still causes height compression, which may potentially cause information loss. Token-Fusion [162] represents a multimodal fusion network that dynamically identifies tokens and replaces them with projected inter-modal features. Additionally, it incorporates residual positional alignment to optimize intermodal alignments more effectively. The Bridge-transformer network [163], on the other hand, is explicitly designed to bridge the learning processes between image and point cloud data. It introduces conditional object queries and points-to-patch projection techniques to enhance the interaction between image and point cloud data. DeepFusion [166] introduces a multimodal fusion approach employing InverseAug and LearnableAlign mechanisms. These techniques facilitate geometric alignment between LiDAR points and image pixels while capturing the correlation between dynamic features in LiDAR and images. BevFusion [167] is designed as an end-to-end transformer-based multimodal and multi-task fusion 3D object detection and segmentation model shown in Figure 4. BEVFusion uses VoxelNet [114] and Swin transformer [168] as LiDAR point cloud data and camera image backbones. The transformed features are fused in BEV before employing task-specific heads for 3D object detection and semantic segmentation.

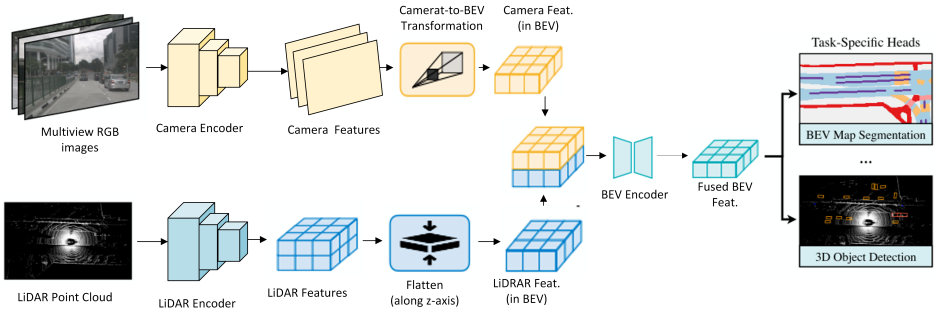

Figure 4. BEVFusion processes features from various input modes, transforming them into a unified BEV representation through view transformations. This approach fuses the BEV features using a fully convolutional BEV encoder and accommodates various tasks by employing specialized task-specific heads.

Table 6 shows the evaluation metrics results using NDS (%) and mAP (%) on the nuScenes test set benchmark. In the table, "V0.075" refers to a voxel grid size of 0.075 m, used with the VoxelNet backbone, and R-101, R18, and R-50 [88] denotes the ResNet network with the corresponding layers. Additionally, DLA34 [169] and V2-99 [170] serve as backbone networks for extracting features from camera images. Transformer-based models such as UVTR [165], MFusion [76], TransFusion [164], and BEVFusion [167] are contrasted with other models that utilize CNN-based multimodal fusion, all of which have been trained and tested on the nuScenes dataset. While BEVFusion [167] demonstrates superior performance compared to other methods, it does exhibit high computational complexity due to its end-to-end transformer network. As previously mentioned in Section 3, trans-

formers are computationally intensive, primarily due to their self-attention mechanism. Consequently, there is a need to find a solution that balances the computational load of transformers and their adaptability.

Table 6. The NDS (%) and mAP (%) comparison of multimodal 3D Fusion Methods on the nuScenes test set benchmark. Here, 'L' and 'C' signify LiDAR point cloud and camera data, respectively.

Method	Backbone	Modality	NDS (%)	mAP (%)
PointPainting [115]	-	LC	58.1	46.4
3D-CVF [123]	VoxelNet-R18	LC	62.3	52.7
FUTR3D [171]	VoxelNet-R101	LC	68.0	64.2
MVP [172]	V0.075-DLA34	LC	70.5	66.4
PointAugmenting [173]	V0.075-DLA34	LC	71.0	66.8
AutoAlign [174]	VoxelNet-R50	LC	71.1	66.6
AutoAlignv2 [175]	VoxelNet-R50	LC	72.4	68.4
UVTR-M [165]	V0.075-R101	LC	70.2	65.4
MFusion [76]	V0.075-R101	LC	71.3	67.7
TransFusion [164]	V0.075-DLA34	LC	71.7	68.9
BEVFusion [167]	V0.075-Swin-T	LC	72.9	70.2

5. Challenges, Open Issues, and Future Directions

The challenges of 3D object detection include varying sensing conditions, unstructured data formats, adverse weather, limited training sets, and imbalanced datasets. Unlike image data, point cloud data is unstructured, unordered, and sparse, characterized by numerous empty points, making its processing more complex and time-consuming. Various sensor fusion techniques have been developed for enhanced perception systems to address these issues. However, a universally accepted method for fusing different sensors to optimize results still needs to be discovered. Depending on the application, early fusion may yield superior results, while in other cases, deep fusion techniques may outperform others. Consequently, identifying the optimal point cloud data representation and fusion techniques and improving point cloud data processing efficiency remain unresolved. This section discusses these challenges, ongoing issues, and potential future developments in 3D object detection for autonomous vehicles.

1. Point Cloud Data Encoding: Point cloud data, characterized by its unstructured, unordered, and sparse nature, presents significant challenges in processing. Various methods have been developed to process point cloud data, such as voxels, PointNet, and projection view. Yet, a universally accepted representation of point cloud data still needs to be improved. The projection method addresses the high computational demands associated with 3D convolution in raw point cloud and voxel processing, but it leads to data loss during domain projection. Conversely, processing data in voxel format resolves the structuring issue but does not mitigate the computational intensity of 3D convolution. Furthermore, Directly processing raw point cloud data is inefficient due to its sparsity and lack of order. Identifying the most effective point cloud data representation to achieve high performance and efficient memory storage still needs to be solved.
2. Fusion Techniques: Various fusion techniques, including early/data fusion, middle/feature fusion, and late/decision fusion, have been developed, with middle fusion further categorized into deep fusion and single-layer fusion. The effectiveness of these techniques varies across different tasks, leading to no consensus on a superior method. A technique yielding excellent results in one application might underperform in another, and searching for a universally effective fusion method for broad applications is an ongoing challenge. Currently, most fusion research focuses on combining LiDAR and RGB images. However, integrating a wider range of sensors, such as thermal, LiDAR, RGB images, and radar, could enhance detection reliability and

increase the safety of autonomous systems, especially in scenarios of sensor failure or compromised performance due to various conditions.

3. Vision Transformer Bottlenecks: Vision transformers have shown impressive performance in various vision applications, owing to their capability to handle long-range dependencies and their generalization ability across different datasets. However, their large number of parameters, with some models having up to a trillion and substantial computational demands, poses challenges in resource-constrained environments, such as edge devices. To address this issue, developing a lightweight vision transformer model that maintains accuracy is crucial. The self-attention head, the most computationally intensive part of vision transformers, is a key focus for optimization. Integrating vision transformers with invertible downsampling techniques, such as wavelets [176,177], could reduce computational load without sacrificing important information.
4. Multitask Learning: Many tasks share a typical architecture in the lower part of their DL models. Utilizing a unified architecture and design to fuse these tasks while differentiating their decision-making processes in the upper part of the model can enhance efficiency in terms of time and computation. Studies such as [103,167] demonstrate the feasibility of multitasking jobs in this context. Developing and implementing an efficient multitask model can save time and memory. For instance, by isolating only the decision-making level, it's possible to construct a network capable of performing 3D object detection, 3D semantic segmentation, and tracking within a single framework.
5. Improve Generalization Ability: Annotating and labeling datasets, a fundamental aspect of supervised learning, requires considerable time and resources. Additionally, models may underperform with data not encountered during training. While unsupervised learning-based models offer a potential solution, their accuracy often needs to be improved compared to supervised learning. To address these challenges, developing semi-supervised learning-based models is crucial for better generalization abilities of detection systems. These models use a small number of labeled datasets and a larger volume of unlabeled images, leveraging the abundance of freely available images while maintaining relatively high accuracy. Semi-supervised 3D object detection is less common compared to 2D object detection. Recent advancements, such as [178], have introduced semi-supervised 3D object detection for autonomous driving using a teacher-student network. In this approach, the teacher model generates pseudo-labels, which are then used by the student model for training alongside the labeled dataset. Exploring one-shot learning, a less common approach in 3D object detection, could be another direction for semi-supervised 3D object detection.
6. More Representative Datasets: Many datasets are unbalanced or fail to robustly represent real-world conditions, such as adverse weather or nighttime driving. Constructing a more representative dataset that includes class balance, adverse weather conditions, and nighttime driving scenarios is as crucial as developing a robust model for 3D object detection. Despite the significant time and financial investment required to prepare such a dataset, developing a robust object detection model is essential.
7. Domain Adaptive Models: DL models' performance often declines when tested in a domain different from the one in which they were trained. Domain adaptation, either homogeneous or heterogeneous, is crucial for ensuring that a model maintains effectiveness across varying domains [179]. This is especially important in autonomous driving, where the model must perform well despite changes in a domain, such as country-specific variations in traffic signs and other factors. Robust 3D perception systems that can learn from and adapt to environmental changes are essential. Such systems improve autonomous driving and enhance the efficiency of critical components like path planning, localization, and control. These latter systems rely on inputs about the environment from the perception systems. Therefore, a highly accurate

and efficient perception system is vital for optimal autonomous vehicle planning, localization, and control.

8. Robust and Lightweight Models: 2D object detection has achieved significant performance improvements in real-time applications, as evidenced by the YOLO series networks [180]. However, 3D object detection has yet to reach a similar level of readiness for real-time implementation. Most 3D object detection models are too resource-intensive for deployment on embedded hardware devices in real-time settings. This limitation is partly due to the nature of the datasets used for training, which often feature skewed class distributions or are predominantly collected during daytime and sunny conditions. Robust, lightweight models capable of functioning under various environmental conditions, including snow and nighttime driving, should be developed to advance toward fully autonomous driving. A representative dataset for training these models should encompass a balanced class distribution and be compiled under diverse environmental conditions, including sunny, foggy, snowy, and nighttime scenarios. Applying network optimization techniques, such as pruning, quantization, quantization-aware training, and knowledge distillation, may reduce the computational complexity without losing much performance.
9. Explainable AI: While DL models have significantly improved performance in various areas, their large size and complexity often make them resemble "black boxes". This obscurity complicates understanding their internal workings and how they arrive at decisions. Explainability builds trust and acceptance, especially in crucial fields such as autonomous driving, where comprehending the basis of decisions is just as vital as the decisions themselves. Moreover, easier-to-interpret models enable developers and researchers to identify and fix potential biases or flaws, leading to more ethical and equitable results. Therefore, developing DL models that balance high performance and clear, transparent operation is an essential focus for ongoing research and innovation in the machine learning domain.
10. Robust Simulators: Using simulators to create artificial data can complement existing datasets effectively. Hence, the need for simulators that can realistically replicate data from LiDAR, radar, and cameras is paramount. These simulators should be able to mimic various weather conditions and provide training and testing for edge technologies and typical driving scenarios. Examples of such simulators include CARLA [181] and the Mississippi State University Autonomous Vehicle Simulator (MAVS) [182]. MAVS stands out with features like automated off-road terrain generation, diverse weather data generation, and automatic data labeling for cameras and LIDAR sensors.

6. Conclusions

This survey has presented the latest advancements in CNNs and vision transformer-based multimodal fusion for 3D object detection methods in autonomous driving. It also reviewed the sensors and datasets commonly used in this field. However, a notable gap in most datasets is their need for more representation of inclement weather and nighttime driving conditions, which are crucial for mimicking real-world driving scenarios. There is an anticipation for more comprehensive datasets that better represent real-world driving conditions and provide a balanced class distribution. Datasets encompassing a variety of weather conditions and driving scenarios are vital for developing a robust driving system. In this context, a sophisticated simulator capable of replicating diverse weather and driving scenarios could be instrumental in building more resilient models. Compared to their 2D counterparts, DL-based 3D object detection methods are less streamlined for deployment in actual systems. Furthermore, there needs to be a consensus on the optimal approach to selecting fusion techniques. Sensor fusion emerges as a key solution for enabling driving under any conditions — at any time and in any weather. While 3D object detection methods have significantly advanced autonomous driving, several challenges remain in enhancing the speed and accuracy of fusion-based 3D object detection models for real-time processing. The area of sensor failure and its potential solutions, essential for maintaining system

integrity, have received limited attention from researchers. Future research in 3D object detection is also anticipated to focus on open-set conditions with domain adaptation, either integrated or as a separate consideration.

Author Contributions: Conceptualization and methodology, S.Y.A.; formal analysis, S.Y.A.; writing—original draft preparation, S.Y.A.; writing—review and editing, J.E.B. and A.C.G.; supervision, J.E.B. and A.C.G. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: Data are contained within the article.

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

Abbreviations

The following abbreviations are used in this manuscript:

AVs	Autonomous Vehicles
BEV	Bird's-Eye View
CNN	Convolutional Neural Network
CV	Computer Vision
DL	Deep Learning
FPN	Feature Pyramid Network
IOU	Intersection Over Union
LiDAR	Light Detection and Ranging
mAP	Mean Average Precision
MHA	Multi-Head Attention
Radar	Radio Detection and Ranging
RPN	Region Proposal Network
SOTA	State-of-the-art
VFE	Voxel Feature Encoding
ViT	Vision Transformer

References

1. National Highway Traffic Safety Administration. *Early Estimates of Motor Vehicle Traffic Fatalities and Fatality Rate by Sub-Categories in 2021*; Technical Report; US National Safety Council: Washington, DC, USA, 2022.
2. Alaba, S.Y.; Ball, J.E. A survey on deep-learning-based lidar 3D object detection for autonomous driving. *Sensors* **2022**, *22*, 9577. [CrossRef] [PubMed]
3. Feng, D.; Haase-Schuetz, C.; Rosenbaum, L.; Hertlein, H.; Glaeser, C.; Timm, F.; Wiesbeck, W.; Dietmayer, K. Deep multi-modal object detection and semantic segmentation for autonomous driving: Datasets, methods, and challenges. *IEEE Trans. Intell. Transp. Syst.* **2020**, *22*, 1341–1360. [CrossRef]
4. Arnold, E.; Al-Jarrah, O.Y.; Dianati, M.; Fallah, S.; Oxtoby, D.; Mouzakitis, A. A survey on 3D object detection methods for autonomous driving applications. *IEEE Trans. Intell. Transp. Syst.* **2019**, *20*, 3782–3795. [CrossRef]
5. Chen, X.; Ma, H.; Wan, J.; Li, B.; Xia, T. Multi-view 3D object detection network for autonomous driving. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 1907–1915.
6. Ku, J.; Mozifian, M.; Lee, J.; Harakeh, A.; Waslander, S.L. Joint 3D proposal generation and object detection from view aggregation. In Proceedings of the 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Madrid, Spain, 1–5 October 2018; pp. 1–8.
7. Rahman, M.M.; Tan, Y.; Xue, J.; Lu, K. Recent advances in 3D object detection in the era of deep neural networks: A survey. *IEEE Trans. Image Process.* **2019**, *29*, 2947–2962. [CrossRef] [PubMed]
8. Li, Y.; Ma, L.; Zhong, Z.; Liu, F.; Chapman, M.A.; Cao, D.; Li, J. Deep learning for LiDAR point clouds in autonomous driving: A review. *IEEE Trans. Neural Netw. Learn. Syst.* **2020**, *32*, 3412–3432. [CrossRef] [PubMed]
9. Guo, Y.; Wang, H.; Hu, Q.; Liu, H.; Liu, L.; Bennamoun, M. Deep learning for 3D point clouds: A survey. *IEEE Trans. Pattern Anal. Mach. Intell.* **2020**, *43*, 4338–4364. [CrossRef] [PubMed]
10. Fernandes, D.; Silva, A.; Névoa, R.; Simões, C.; Gonzalez, D.; Guevara, M.; Novais, P.; Monteiro, J.; Melo-Pinto, P. Point-cloud based 3D object detection and classification methods for self-driving applications: A survey and taxonomy. *Inf. Fusion* **2021**, *68*, 161–191. [CrossRef]
11. Li, B.; Shi, Y.; Qi, Z.; Chen, Z. A Survey on Semantic Segmentation. In Proceedings of the 2018 IEEE International Conference on Data Mining Workshops (ICDMW), Singapore, 17–20 November 2018; pp. 1233–1240. [CrossRef]

12. Lateef, F.; Ruichek, Y. Survey on semantic segmentation using deep learning techniques. *Neurocomputing* **2019**, *338*, 321–348. [CrossRef]
13. Yu, H.; Yang, Z.; Tan, L.; Wang, Y.; Sun, W.; Sun, M.; Tang, Y. Methods and datasets on semantic segmentation: A review. *Neurocomputing* **2018**, *304*, 82–103. [CrossRef]
14. Qian, R.; Lai, X.; Li, X. 3D Object Detection for Autonomous Driving: A Survey. *arXiv* **2021**, arXiv:2106.10823.
15. Cui, Y.; Chen, R.; Chu, W.; Chen, L.; Tian, D.; Li, Y.; Cao, D. Deep learning for image and point cloud fusion in autonomous driving: A review. *IEEE Trans. Intell. Transp. Syst.* **2021**, *23*, 722–739. [CrossRef]
16. Wang, Y.; Mao, Q.; Zhu, H.; Deng, J.; Zhang, Y.; Ji, J.; Li, H.; Zhang, Y. Multi-modal 3D object detection in autonomous driving: A survey. *Int. J. Comput. Vis.* **2023**, *131*, 2122–2152. [CrossRef]
17. Alaba, S.Y.; Ball, J.E. Deep Learning-Based Image 3-D Object Detection for Autonomous Driving. *IEEE Sens. J.* **2023**, *23*, 3378–3394. [CrossRef]
18. Chen, X.; Kundu, K.; Zhu, Y.; Ma, H.; Fidler, S.; Urtasun, R. 3D object proposals using stereo imagery for accurate object class detection. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *40*, 1259–1272. [CrossRef] [PubMed]
19. Pham, C.C.; Jeon, J.W. Robust object proposals re-ranking for object detection in autonomous driving using convolutional neural networks. *Signal Process. Image Commun.* **2017**, *53*, 110–122. [CrossRef]
20. Khader, M.; Cherian, S. An Introduction to Automotive LIDAR. *Tex. Instrum.* **2018**.
21. Lambert, J.; Carballo, A.; Cano, A.M.; Narksri, P.; Wong, D.; Takeuchi, E.; Takeda, K. Performance analysis of 10 models of 3D LiDARs for automated driving. *IEEE Access* **2020**, *8*, 131699–131722. [CrossRef]
22. Pusztai, Z.; Hajder, L. Accurate calibration of LiDAR-camera systems using ordinary boxes. In Proceedings of the IEEE International Conference on Computer Vision Workshops, Venice, Italy, 22–29 October 2017; pp. 394–402.
23. Beltrán, J.; Guindel, C.; García, F. Automatic Extrinsic Calibration Method for LiDAR and Camera Sensor Setups. *arXiv* **2021**, arXiv:2101.04431.
24. Bai, Z.; Jiang, G.; Xu, A. LiDAR-Camera Calibration Using Line Correspondences. *Sensors* **2020**, *20*, 6319. [CrossRef]
25. An, P.; Ma, T.; Yu, K.; Fang, B.; Zhang, J.; Fu, W.; Ma, J. Geometric calibration for LiDAR-camera system fusing 3D-2D and 3D-3D point correspondences. *Opt. Express* **2020**, *28*, 2122–2141. [CrossRef]
26. Zhou, L.; Li, Z.; Kaess, M. Automatic extrinsic calibration of a camera and a 3D lidar using line and plane correspondences. In Proceedings of the 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Madrid, Spain, 1–5 October 2018; pp. 5562–5569.
27. Pusztai, Z.; Eichhardt, I.; Hajder, L. Accurate calibration of multi-lidar-multi-camera systems. *Sensors* **2018**, *18*, 2139. [CrossRef] [PubMed]
28. Zhu, Y.; Li, C.; Zhang, Y. Online camera-lidar calibration with sensor semantic information. In Proceedings of the 2020 IEEE International Conference on Robotics and Automation (ICRA), Paris, France, 31 May–31 August 2020; pp. 4970–4976.
29. Patole, S.M.; Torlak, M.; Wang, D.; Ali, M. Automotive radars: A review of signal processing techniques. *IEEE Signal Process. Mag.* **2017**, *34*, 22–35. [CrossRef]
30. Rudy Burger, T.S.; Sumida, S. *Beyond The Headlights: ADAS and Autonomous Sensing*; Technical Report; Woodside Capital Partners: Palo Alto, CA, USA, 2016.
31. Geiger, A.; Lenz, P.; Urtasun, R. Are we ready for autonomous driving? The kitti vision benchmark suite. In Proceedings of the 2012 IEEE Conference on Computer Vision and Pattern Recognition, Providence, RI, USA, 16–21 June 2012; pp. 3354–3361.
32. Simony, M.; Milzy, S.; Amendey, K.; Gross, H.M. Complex-yolo: An euler-region-proposal for real-time 3D object detection on point clouds. In Proceedings of the European Conference on Computer Vision (ECCV) Workshops, Munich, Germany, 8–14 September 2018; pp. 197–209.
33. Caesar, H.; Bankiti, V.; Lang, A.H.; Vora, S.; Liong, V.E.; Xu, Q.; Krishnan, A.; Pan, Y.; Baldan, G.; Beijbom, O. nuScenes: A multimodal dataset for autonomous driving. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 11621–11631.
34. Pham, Q.H.; Sevestre, P.; Pahwa, R.S.; Zhan, H.; Pang, C.H.; Chen, Y.; Mustafa, A.; Chandrasekhar, V.; Lin, J. A* 3D dataset: Towards autonomous driving in challenging environments. In Proceedings of the 2020 IEEE International Conference on Robotics and Automation (ICRA), Paris, France, 31 May–31 August 2020; pp. 2267–2273.
35. Patil, A.; Malla, S.; Gang, H.; Chen, Y.T. The H3D dataset for full-surround 3D multi-object detection and tracking in crowded urban scenes. In Proceedings of the 2019 International Conference on Robotics and Automation (ICRA), Montreal, QC, Canada, 20–24 May 2019; pp. 9552–9557.
36. Carballo, A.; Lambert, J.; Monrroy, A.; Wong, D.; Narksri, P.; Kitsukawa, Y.; Takeuchi, E.; Kato, S.; Takeda, K. LIBRE: The multiple 3D LiDAR dataset. In Proceedings of the 2020 IEEE Intelligent Vehicles Symposium (IV), Las Vegas, NV, USA, 19 October–13 November 2020; pp. 1094–1101.
37. Quigley, M.; Conley, K.; Gerkey, B.; Faust, J.; Foote, T.; Leibs, J.; Wheeler, R.; Ng, A.Y. ROS: An open-source Robot Operating System. In Proceedings of the ICRA Workshop on Open Source Software, Kobe, Japan, 12–17 May 2009; Volume 3, p. 5.
38. Sun, P.; Kretzschmar, H.; Dotiwalla, X.; Chouard, A.; Patnaik, V.; Tsui, P.; Guo, J.; Zhou, Y.; Chai, Y.; Caine, B.; et al. Scalability in perception for autonomous driving: Waymo open dataset. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 2446–2454.

39. Meyer, M.; Kuschk, G. Automotive radar dataset for deep learning based 3D object detection. In Proceedings of the 2019 16th European Radar Conference (EuRAD), Paris, France, 2–4 October 2019; pp. 129–132.
40. Kesten, R.; Usman, M.; Houston, J.; Pandya, T.; Nadhamuni, K.; Ferreira, A.; Yuan, M.; Low, B.; Jain, A.; Ondruska, P.; et al. Lyft Level 5 Perception Dataset. *arXiv* **2020**, arXiv:2006.14480. Available online: https://github.com/wenkaip-personal/pyromid_l5prediction (accessed on 5 January 2024).
41. Xiao, P.; Shao, Z.; Hao, S.; Zhang, Z.; Chai, X.; Jiao, J.; Li, Z.; Wu, J.; Sun, K.; Jiang, K.; et al. PandaSet: Advanced Sensor Suite Dataset for Autonomous Driving. In Proceedings of the 2021 IEEE International Intelligent Transportation Systems Conference (ITSC), Indianapolis, IN, USA, 19–22 September 2021; pp. 3095–3101.
42. Cordts, M.; Omran, M.; Ramos, S.; Rehfeld, T.; Enzweiler, M.; Benenson, R.; Franke, U.; Roth, S.; Schiele, B. The cityscapes dataset for semantic urban scene understanding. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 3213–3223.
43. Choi, Y.; Kim, N.; Hwang, S.; Park, K.; Yoon, J.S.; An, K.; Kweon, I.S. KAIST multi-spectral day/night data set for autonomous and assisted driving. *IEEE Trans. Intell. Transp. Syst.* **2018**, *19*, 934–948. [CrossRef]
44. Huang, X.; Wang, P.; Cheng, X.; Zhou, D.; Geng, Q.; Yang, R. The apolloscape open dataset for autonomous driving and its application. *IEEE Trans. Pattern Anal. Mach. Intell.* **2019**, *42*, 2702–2719. [CrossRef] [PubMed]
45. Yang, G.; Song, X.; Huang, C.; Deng, Z.; Shi, J.; Zhou, B. Drivingstereo: A large-scale dataset for stereo matching in autonomous driving scenarios. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 899–908.
46. Maddern, W.; Pascoe, G.; Linegar, C.; Newman, P. 1 year, 1000 km: The Oxford RobotCar dataset. *Int. J. Robot. Res.* **2017**, *36*, 3–15. [CrossRef]
47. Ros, G.; Sellart, L.; Materzynska, J.; Vazquez, D.; Lopez, A.M. The synthia dataset: A large collection of synthetic images for semantic segmentation of urban scenes. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 3234–3243.
48. Mao, J.; Niu, M.; Jiang, C.; Liang, H.; Chen, J.; Liang, X.; Li, Y.; Ye, C.; Zhang, W.; Li, Z.; et al. One million scenes for autonomous driving: Once dataset. *arXiv* **2021**, arXiv:2106.11037.
49. Bijelic, M.; Gruber, T.; Mannan, F.; Kraus, F.; Ritter, W.; Dietmayer, K.; Heide, F. Seeing through fog without seeing fog: Deep multimodal sensor fusion in unseen adverse weather. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 11682–11692.
50. Weng, X.; Man, Y.; Park, J.; Yuan, Y.; O'Toole, M.; Kitani, K.M. All-In-One Drive: A Comprehensive Perception Dataset with High-Density Long-Range Point Clouds. Available online: http://www.aiodrive.org/ (accessed on 5 January 2024).
51. Palffy, A.; Pool, E.; Baratam, S.; Kooij, J.F.; Gavrila, D.M. Multi-class road user detection with 3+ 1D radar in the View-of-Delft dataset. *IEEE Robot. Autom. Lett.* **2022**, *7*, 4961–4968. [CrossRef]
52. Zheng, L.; Ma, Z.; Zhu, X.; Tan, B.; Li, S.; Long, K.; Sun, W.; Chen, S.; Zhang, L.; Wan, M.; et al. Tj4dradset: A 4d radar dataset for autonomous driving. In Proceedings of the 2022 IEEE 25th International Conference on Intelligent Transportation Systems (ITSC), Macau, China, 8–12 October 2022; pp. 493–498.
53. Paek, D.H.; Kong, S.H.; Wijaya, K.T. K-radar: 4d radar object detection dataset and benchmark for autonomous driving in various weather conditions. *arXiv* **2022**, arXiv:2206.08171.
54. Matuszka, T.; Barton, I.; Butykai, Á.; Hajas, P.; Kiss, D.; Kovács, D.; Kunsági-Máté, S.; Lengyel, P.; Németh, G.; Pető, L.; et al. aiMotive Dataset: A Multimodal Dataset for Robust Autonomous Driving with Long-Range Perception. *arXiv* **2022**, arXiv:2211.09445.
55. Guo, M.H.; Xu, T.X.; Liu, J.J.; Liu, Z.N.; Jiang, P.T.; Mu, T.J.; Zhang, S.H.; Martin, R.R.; Cheng, M.M.; Hu, S.M. Attention mechanisms in computer vision: A survey. *Comput. Vis. Media* **2022**, *8*, 331–368. [CrossRef]
56. Srivastava, R.K.; Greff, K.; Schmidhuber, J. Training very deep networks. *Adv. Neural Inf. Process. Syst.* **2015**, *28*, 2377–2385.
57. Hu, J.; Shen, L.; Sun, G. Squeeze-and-excitation networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 7132–7141.
58. Chen, L.; Zhang, H.; Xiao, J.; Nie, L.; Shao, J.; Liu, W.; Chua, T.S. Sca-cnn: Spatial and channel-wise attention in convolutional networks for image captioning. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 5659–5667.
59. Gao, Z.; Xie, J.; Wang, Q.; Li, P. Global second-order pooling convolutional networks. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 3024–3033.
60. Wang, Q.; Wu, B.; Zhu, P.; Li, P.; Zuo, W.; Hu, Q. ECA-Net: Efficient channel attention for deep convolutional neural networks. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 11534–11542.
61. Qin, Z.; Zhang, P.; Wu, F.; Li, X. Fcanet: Frequency channel attention networks. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Nashville, TN, USA, 20–25 June 2021; pp. 783–792.
62. Yang, Z.; Zhu, L.; Wu, Y.; Yang, Y. Gated channel transformation for visual recognition. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 11794–11803.
63. Hu, J.; Shen, L.; Albanie, S.; Sun, G.; Vedaldi, A. Gather-excite: Exploiting feature context in convolutional neural networks. *Adv. Neural Inf. Process. Syst.* **2018**, *31*.

64. Zhao, H.; Zhang, Y.; Liu, S.; Shi, J.; Loy, C.C.; Lin, D.; Jia, J. Psanet: Point-wise spatial attention network for scene parsing. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 267–283.
65. Chollet, F. Xception: Deep learning with depthwise separable convolutions. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 1251–1258.
66. Wang, X.; Girshick, R.; Gupta, A.; He, K. Non-local neural networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 7794–7803.
67. Cao, Y.; Xu, J.; Lin, S.; Wei, F.; Hu, H. Gcnet: Non-local networks meet squeeze-excitation networks and beyond. In Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops, Long Beach, CA, USA, 16–17 June 2019; pp. 1971–1980.
68. Chen, Y.; Kalantidis, Y.; Li, J.; Yan, S.; Feng, J. A^2-nets: Double attention networks. *Adv. Neural Inf. Process. Syst.* **2018**, *31*, 350–359.
69. Wang, F.; Jiang, M.; Qian, C.; Yang, S.; Li, C.; Zhang, H.; Wang, X.; Tang, X. Residual attention network for image classification. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 3156–3164.
70. Liu, J.J.; Hou, Q.; Cheng, M.M.; Wang, C.; Feng, J. Improving convolutional networks with self-calibrated convolutions. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 10096–10105.
71. Misra, D.; Nalamada, T.; Arasanipalai, A.U.; Hou, Q. Rotate to attend: Convolutional triplet attention module. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, Virtual, 5–9 January 2021; pp. 3139–3148.
72. Fu, J.; Liu, J.; Tian, H.; Li, Y.; Bao, Y.; Fang, Z.; Lu, H. Dual attention network for scene segmentation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 3146–3154.
73. Woo, S.; Park, J.; Lee, J.Y.; Kweon, I.S. Cbam: Convolutional block attention module. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 3–19.
74. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, L.; Polosukhin, I. Attention Is All You Need. *arXiv* **2017**, arXiv:1706.03762.
75. Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S.; et al. An image is worth 16 $\times$16 words: Transformers for image recognition at scale. *arXiv* **2020**, arXiv:2010.11929.
76. Alaba, S.Y.; Ball, J.E. Multi-sensor fusion 3D object detection for autonomous driving. In Proceedings of the Autonomous Systems: Sensors, Processing and Security for Ground, Air, Sea, and Space Vehicles and Infrastructure 2023, SPIE, Orlando, FL, USA, 30 April–5 May 2023; Volume 12540, pp. 36–43.
77. Fan, L.; Pang, Z.; Zhang, T.; Wang, Y.X.; Zhao, H.; Wang, F.; Wang, N.; Zhang, Z. Embracing single stride 3D object detector with sparse transformer. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 18–24 June 2022; pp. 8458–8468.
78. Yu, H.; Qin, Z.; Hou, J.; Saleh, M.; Li, D.; Busam, B.; Ilic, S. Rotation-invariant transformer for point cloud matching. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Vancouver, BC, Canada, 18–22 June 2023; pp. 5384–5393.
79. Liu, Z.; Yang, X.; Tang, H.; Yang, S.; Han, S. FlatFormer: Flattened Window Attention for Efficient Point Cloud Transformer. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Vancouver, BC, Canada, 18–22 June 2023; pp. 1200–1211.
80. Guo, M.H.; Cai, J.X.; Liu, Z.N.; Mu, T.J.; Martin, R.R.; Hu, S.M. Pct: Point cloud transformer. *Comput. Vis. Media* **2021**, *7*, 187–199. [CrossRef]
81. Yan, X.; Zheng, C.; Li, Z.; Wang, S.; Cui, S. Pointasnl: Robust point clouds processing using nonlocal neural networks with adaptive sampling. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 5589–5598.
82. Zhao, H.; Jiang, L.; Jia, J.; Torr, P.H.; Koltun, V. Point transformer. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Nashville, TN, USA, 20–25 June 2021; pp. 16259–16268.
83. Park, C.; Jeong, Y.; Cho, M.; Park, J. Fast point transformer. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 18–24 June 2022; pp. 16949–16958.
84. Mao, J.; Xue, Y.; Niu, M.; Bai, H.; Feng, J.; Liang, X.; Xu, H.; Xu, C. Voxel transformer for 3D object detection. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Nashville, TN, USA, 20–25 June 2021; pp. 3164–3173.
85. He, C.; Li, R.; Li, S.; Zhang, L. Voxel set transformer: A set-to-set approach to 3D object detection from point clouds. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 18–24 June 2022; pp. 8417–8427.
86. Wu, X.; Lao, Y.; Jiang, L.; Liu, X.; Zhao, H. Point transformer v2: Grouped vector attention and partition-based pooling. *Adv. Neural Inf. Process. Syst.* **2022**, *35*, 33330–33342.
87. Sun, P.; Tan, M.; Wang, W.; Liu, C.; Xia, F.; Leng, Z.; Anguelov, D. Swformer: Sparse window transformer for 3D object detection in point clouds. In Proceedings of the European Conference on Computer Vision, Tel Aviv, Israel, 23–27 October 2022; Springer: Tel Aviv, Israel, 2022; pp. 426–442.
88. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.

89. Lin, T.Y.; Dollár, P.; Girshick, R.; He, K.; Hariharan, B.; Belongie, S. Feature pyramid networks for object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 2117–2125.
90. Wang, Z.; Zhan, W.; Tomizuka, M. Fusing bird's eye view lidar point cloud and front view camera image for 3D object detection. In Proceedings of the 2018 IEEE Intelligent Vehicles Symposium (IV), Changshu, China, 26–30 June 2018; pp. 1–6.
91. Liu, Y.; Suo, C.; Liu, Z.; Liu, Y.H. A Multi-Sensor Fusion Based 2D-Driven 3D Object Detection Approach for Large Scene Applications. In Proceedings of the 2019 IEEE International Conference on Robotics and Biomimetics (ROBIO), Dali, China, 6–8 December 2019; pp. 2181–2188.
92. Meyer, M.; Kuschk, G. Deep learning based 3D object detection for automotive radar and camera. In Proceedings of the 2019 16th European Radar Conference (EuRAD), Paris, France, 2–4 October 2019; pp. 133–136.
93. Simonyan, K.; Zisserman, A. Very deep convolutional networks for large-scale image recognition. *arXiv* **2014**, arXiv:1409.1556.
94. Nobis, F.; Shafiei, E.; Karle, P.; Betz, J.; Lienkamp, M. Radar Voxel Fusion for 3D Object Detection. *Appl. Sci.* **2021**, *11*, 5598. [CrossRef]
95. Xu, D.; Anguelov, D.; Jain, A. Pointfusion: Deep sensor fusion for 3D bounding box estimation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 244–253.
96. Qi, C.R.; Su, H.; Mo, K.; Guibas, L.J. Pointnet: Deep learning on point sets for 3D classification and segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 652–660.
97. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. In Proceedings of the Neural Information Processing Systems (NIPS), Montreal, QC, Canada, 7–12 December 2015.
98. Huang, L.; Yang, Y.; Deng, Y.; Yu, Y. Densebox: Unifying landmark localization with end to end object detection. *arXiv* **2015**, arXiv:1509.04874.
99. Liang, M.; Yang, B.; Wang, S.; Urtasun, R. Deep continuous fusion for multi-sensor 3D object detection. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 641–656.
100. Wang, S.; Suo, S.; Ma, W.C.; Pokrovsky, A.; Urtasun, R. Deep parametric continuous convolutional neural networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 2589–2597.
101. Yang, B.; Liang, M.; Urtasun, R. Hdnet: Exploiting hd maps for 3D object detection. In Proceedings of the Conference on Robot Learning, Zürich, Switzerland, 29–31 October 2018; pp. 146–155.
102. Yang, B.; Luo, W.; Urtasun, R. Pixor: Real-time 3D object detection from point clouds. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 7652–7660.
103. Liang, M.; Yang, B.; Chen, Y.; Hu, R.; Urtasun, R. Multi-task multi-sensor fusion for 3D object detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 7345–7353.
104. Du, X.; Ang, M.H.; Karaman, S.; Rus, D. A general pipeline for 3D detection of vehicles. In Proceedings of the 2018 IEEE International Conference on Robotics and Automation (ICRA), Brisbane, QLD, Australia, 21–25 May 2018; pp. 3194–3200.
105. Shin, K.; Kwon, Y.P.; Tomizuka, M. Roarnet: A robust 3D object detection based on region approximation refinement. In Proceedings of the 2019 IEEE Intelligent Vehicles Symposium (IV), Paris, France, 9–12 June 2019; pp. 2510–2515.
106. Chen, Y.; Liu, S.; Shen, X.; Jia, J. Fast point r-cnn. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Long Beach, CA, USA, 15–20 June 2019; pp. 9775–9784.
107. Mousavian, A.; Anguelov, D.; Flynn, J.; Kosecka, J. 3D bounding box estimation using deep learning and geometry. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 7074–7082.
108. Meyer, G.P.; Charland, J.; Hegde, D.; Laddha, A.; Vallespi-Gonzalez, C. Sensor fusion for joint 3D object detection and semantic segmentation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops, Long Beach, CA, USA, 16–20 June 2019; pp. 1230–1237.
109. Meyer, G.P.; Laddha, A.; Kee, E.; Vallespi-Gonzalez, C.; Wellington, C.K. Lasernet: An efficient probabilistic 3D object detector for autonomous driving. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 12677–12686.
110. Lim, T.Y.; Ansari, A.; Major, B.; Fontijne, D.; Hamilton, M.; Gowaikar, R.; Subramanian, S. Radar and camera early fusion for vehicle detection in advanced driver assistance systems. In Proceedings of the Machine Learning for Autonomous Driving Workshop at the 33rd Conference on Neural Information Processing Systems, Vancouver, BC, Canada, 8–14 December 2019; Volume 2.
111. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.Y.; Berg, A.C. Ssd: Single shot multibox detector. In Proceedings of the European Conference on Computer Vision, Amsterdam, The Netherlands, 11–14 October 2016; Springer: Amsterdam, The Netherlands, 2016; pp. 21–37.
112. Zhou, Y.; Sun, P.; Zhang, Y.; Anguelov, D.; Gao, J.; Ouyang, T.; Guo, J.; Ngiam, J.; Vasudevan, V. End-to-end multi-view fusion for 3D object detection in lidar point clouds. In Proceedings of the Conference on Robot Learning, Cambridge, MA, USA, 16–18 November 2020; pp. 923–932.
113. Lang, A.H.; Vora, S.; Caesar, H.; Zhou, L.; Yang, J.; Beijbom, O. Pointpillars: Fast encoders for object detection from point clouds. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 12697–12705.

114. Zhou, Y.; Tuzel, O. Voxelnet: End-to-end learning for point cloud based 3D object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 4490–4499.
115. Vora, S.; Lang, A.H.; Helou, B.; Beijbom, O. Pointpainting: Sequential fusion for 3D object detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 4604–4612.
116. Shi, S.; Wang, X.; Li, H. Pointrcnn: 3D object proposal generation and detection from point cloud. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 770–779.
117. Wang, L.; Fan, X.; Chen, J.; Cheng, J.; Tan, J.; Ma, X. 3D object detection based on sparse convolution neural network and feature fusion for autonomous driving in smart cities. *Sustain. Cities Soc.* **2020**, *54*, 102002. [CrossRef]
118. Wang, G.; Tian, B.; Zhang, Y.; Chen, L.; Cao, D.; Wu, J. Multi-View Adaptive Fusion Network for 3D Object Detection. *arXiv* **2020**, arXiv:2011.00652.
119. Krispel, G.; Opitz, M.; Waltner, G.; Possegger, H.; Bischof, H. Fuseseg: Lidar point cloud segmentation fusing multi-modal data. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, Snowmass Village, CO, USA, 1–5 March 2020; pp. 1874–1883.
120. Wu, B.; Wan, A.; Yue, X.; Keutzer, K. Squeezeseg: Convolutional neural nets with recurrent crf for real-time road-object segmentation from 3D lidar point cloud. In Proceedings of the 2018 IEEE International Conference on Robotics and Automation (ICRA), Brisbane, QLD, Australia, 21–25 May 2018; pp. 1887–1893.
121. Hong, D.S.; Chen, H.H.; Hsiao, P.Y.; Fu, L.C.; Siao, S.M. CrossFusion net: Deep 3D object detection based on RGB images and point clouds in autonomous driving. *Image Vis. Comput.* **2020**, *100*, 103955. [CrossRef]
122. Chen, C.; Fragonara, L.Z.; Tsourdos, A. RoIFusion: 3D Object Detection From LiDAR and Vision. *IEEE Access* **2021**, *9*, 51710–51721. [CrossRef]
123. Yoo, J.H.; Kim, Y.; Kim, J.S.; Choi, J.W. 3D-cvf: Generating joint camera and lidar features using cross-view spatial feature fusion for 3D object detection. *arXiv* **2020**, arXiv:2004.12636.
124. Wang, L.; Chen, T.; Anklam, C.; Goldluecke, B. High Dimensional Frustum PointNet for 3D Object Detection from Camera, LiDAR, and Radar. In Proceedings of the 2020 IEEE Intelligent Vehicles Symposium (IV), Las Vegas, NV, USA, 19 October–13 November 2020; pp. 1621–1628.
125. Qi, C.R.; Liu, W.; Wu, C.; Su, H.; Guibas, L.J. Frustum pointnets for 3D object detection from rgb-d data. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 918–927.
126. Qi, C.R.; Yi, L.; Su, H.; Guibas, L.J. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. *arXiv* **2017**, arXiv:1706.02413.
127. Kim, Y.; Choi, J.W.; Kum, D. GRIF Net: Gated Region of Interest Fusion Network for Robust 3D Object Detection from Radar Point Cloud and Monocular Image. In Proceedings of the International Conference on Intelligent Robots and Systems (IROS2020), Las Vegas, NV, USA, 24 October–24 January 2020.
128. Ren, M.; Pokrovsky, A.; Yang, B.; Urtasun, R. Sbnet: Sparse blocks network for fast inference. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 8711–8720.
129. Nabati, R.; Qi, H. Centerfusion: Center-based radar and camera fusion for 3D object detection. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, Virtual, 5–9 January2021; pp. 1527–1536.
130. Gong, Z.; Lin, H.; Zhang, D.; Luo, Z.; Zelek, J.; Chen, Y.; Nurunnabi, A.; Wang, C.; Li, J. A Frustum-based probabilistic framework for 3D object detection by fusion of LiDAR and camera data. *ISPRS J. Photogramm. Remote Sens.* **2020**, *159*, 90–100. [CrossRef]
131. Chen, J.; Bai, T. SAANet: Spatial adaptive alignment network for object detection in automatic driving. *Image Vis. Comput.* **2020**, *94*, 103873. [CrossRef]
132. Huang, T.; Liu, Z.; Chen, X.; Bai, X. Epnet: Enhancing point features with image semantics for 3D object detection. In Proceedings of the European Conference on Computer Vision, Glasgow, UK, 23–28 August 2020; Springer: Glasgow, UK, 2020; pp. 35–52.
133. Gao, X.; Hu, D. MVDCANet: An End-to-End Self-Attention Based MultiView-DualChannel 3D Object Detection. *IEEE Sens. J.* **2021**, *21*, 27789–27800. [CrossRef]
134. Cao, J.; Li, Y.; Sun, M.; Chen, Y.; Lischinski, D.; Cohen-Or, D.; Chen, B.; Tu, C. Do-conv: Depthwise over-parameterized convolutional layer. *arXiv* **2020**, arXiv:2006.12030.
135. Miao, Z.; Chen, J.; Pan, H.; Zhang, R.; Liu, K.; Hao, P.; Zhu, J.; Wang, Y.; Zhan, X. PVGNet: A Bottom-Up One-Stage 3D Object Detector With Integrated Multi-Level Features. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 20–25 June 2021; pp. 3279–3288.
136. He, C.; Zeng, H.; Huang, J.; Hua, X.S.; Zhang, L. Structure aware single-stage 3D object detection from point cloud. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 11873–11882.
137. Xu, S.; Zhou, D.; Fang, J.; Yin, J.; Bin, Z.; Zhang, L. FusionPainting: Multimodal fusion with adaptive attention for 3D object detection. In Proceedings of the 2021 IEEE International Intelligent Transportation Systems Conference (ITSC), Indianapolis, IN, USA, 19–22 September 2021; pp. 3047–3054.
138. Zhang, H.; Yang, D.; Yurtsever, E.; Redmill, K.A.; Özgüner, Ü. Faraway-frustum: Dealing with lidar sparsity for 3D object detection using fusion. In Proceedings of the 2021 IEEE International Intelligent Transportation Systems Conference (ITSC), Indianapolis, IN, USA, 19–22 September 2021; pp. 2646–2652.
139. Dou, J.; Xue, J.; Fang, J. SEG-VoxelNet for 3D vehicle detection from RGB and LiDAR data. In Proceedings of the 2019 International Conference on Robotics and Automation (ICRA), Montreal, QC, Canada, 20–24 May 2019; pp. 4362–4368.

140. Howard, A.G.; Zhu, M.; Chen, B.; Kalenichenko, D.; Wang, W.; Weyand, T.; Andreetto, M.; Adam, H. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv* **2017**, arXiv:1704.04861.
141. Roy, A.G.; Navab, N.; Wachinger, C. Concurrent spatial and channel 'squeeze & excitation'in fully convolutional networks. In Proceedings of the International Conference on Medical Image Computing and Computer-Assisted Intervention, Granada, Spain, 16–20 September 2018; Springer: Granada, Spain, 2018; pp. 421–429.
142. Liu, S.; Huang, D. Receptive field block net for accurate and fast object detection. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 385–400.
143. Lu, H.; Chen, X.; Zhang, G.; Zhou, Q.; Ma, Y.; Zhao, Y. SCANet: Spatial-channel attention network for 3D object detection. In Proceedings of the ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Brighton, UK, 12–17 May 2019; pp. 1992–1996.
144. Sindagi, V.A.; Zhou, Y.; Tuzel, O. Mvx-net: Multimodal voxelnet for 3D object detection. In Proceedings of the 2019 International Conference on Robotics and Automation (ICRA), Montreal, QC, Canada, 20–24 May 2019; pp. 7276–7282.
145. Wang, C.H.; Chen, H.W.; Fu, L.C. VPFNet: Voxel-Pixel Fusion Network for Multi-class 3D Object Detection. *arXiv* **2021**, arXiv:2111.00966.
146. Zhang, L.; Li, X.; Tang, K.; Jiang, Y.; Yang, L.; Zhang, Y.; Chen, X. FS-Net: LiDAR-Camera Fusion With Matched Scale for 3D Object Detection in Autonomous Driving. *IEEE Trans. Intell. Transp. Syst.* **2023**, *24*, 12154–12165. [CrossRef]
147. Mahmoud, A.; Hu, J.S.; Waslander, S.L. Dense voxel fusion for 3D object detection. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, Waikoloa, HI, USA, 2–7 January 2023; pp. 663–672.
148. Lin, C.; Tian, D.; Duan, X.; Zhou, J.; Zhao, D.; Cao, D. CL3D: Camera-LiDAR 3D object detection with point feature enhancement and point-guided fusion. *IEEE Trans. Intell. Transp. Syst.* **2022**, *23*, 18040–18050. [CrossRef]
149. Pang, S.; Morris, D.; Radha, H. CLOCs: Camera-LiDAR object candidates fusion for 3D object detection. In Proceedings of the 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Las Vegas, NV, USA, 24 October–24 January 2020; pp. 10386–10393.
150. Ren, J.; Chen, X.; Liu, J.; Sun, W.; Pang, J.; Yan, Q.; Tai, Y.W.; Xu, L. Accurate single stage detector using recurrent rolling convolution. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 5420–5428.
151. Cai, Z.; Fan, Q.; Feris, R.S.; Vasconcelos, N. A unified multi-scale deep convolutional neural network for fast object detection. In Proceedings of the European Conference on Computer Vision, Amsterdam, The Netherlands, 11–14 October 2016; Springer: Amsterdam, The Netherlands, 2016; pp. 354–370.
152. Cai, Z.; Vasconcelos, N. Cascade r-cnn: High quality object detection and instance segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.* **2019**, *43*, 1483–1498. [CrossRef]
153. Yan, Y.; Mao, Y.; Li, B. Second: Sparsely embedded convolutional detection. *Sensors* **2018**, *18*, 3337. [CrossRef]
154. Shi, S.; Guo, C.; Jiang, L.; Wang, Z.; Shi, J.; Wang, X.; Li, H. Pv-rcnn: Point-voxel feature set abstraction for 3D object detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 10529–10538.
155. Zhang, Z.; Liang, Z.; Zhang, M.; Zhao, X.; Li, H.; Yang, M.; Tan, W.; Pu, S. RangeLVDet: Boosting 3D Object Detection in LIDAR with Range Image and RGB Image. *IEEE Sens. J.* **2021**, *22*, 1391–1403. [CrossRef]
156. Redmon, J. Darknet: Open Source Neural Networks in C. 2013–2016. Available online: http://pjreddie.com/darknet/ (accessed on 5 January 2024).
157. Redmon, J.; Farhadi, A. Yolov3: An incremental improvement. *arXiv* **2018**, arXiv:1804.02767.
158. Kuang, H.; Liu, X.; Zhang, J.; Fang, Z. Multi-modality cascaded fusion technology for autonomous driving. In Proceedings of the 2020 4th International Conference on Robotics and Automation Sciences (ICRAS), Wuhan, China, 12–14 June 2020; pp. 44–49.
159. Xie, L.; Xiang, C.; Yu, Z.; Xu, G.; Yang, Z.; Cai, D.; He, X. PI-RCNN: An efficient multi-sensor 3D object detector with point-based attentive cont-conv fusion module. In Proceedings of the AAAI Conference on Artificial Intelligence, New York, NY, USA, 7–12 February 2020; Volume 34, pp. 12460–12467.
160. Jiao, J.; Yun, P.; Tai, L.; Liu, M. MLOD: Awareness of extrinsic perturbation in multi-lidar 3D object detection for autonomous driving. In Proceedings of the 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Las Vegas, NV, USA, 24 October–24 January 2020; pp. 10556–10563.
161. Kim, Y.; Kim, A. On the uncertainty propagation: Why uncertainty on lie groups preserves monotonicity? In Proceedings of the 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Vancouver, BC, Canada, 24–28 September 2017; pp. 3425–3432.
162. Wang, Y.; Chen, X.; Cao, L.; Huang, W.; Sun, F.; Wang, Y. Multimodal token fusion for vision transformers. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 18–24 June 2022; pp. 12186–12195.
163. Wang, Y.; Ye, T.; Cao, L.; Huang, W.; Sun, F.; He, F.; Tao, D. Bridged transformer for vision and point cloud 3D object detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 18–24 June 2022; pp. 12114–12123.
164. Bai, X.; Hu, Z.; Zhu, X.; Huang, Q.; Chen, Y.; Fu, H.; Tai, C.L. Transfusion: Robust lidar-camera fusion for 3D object detection with transformers. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 18–24 June 2022; pp. 1090–1099.

165. Li, Y.; Chen, Y.; Qi, X.; Li, Z.; Sun, J.; Jia, J. Unifying voxel-based representation with transformer for 3D object detection. *Adv. Neural Inf. Process. Syst.* **2022**, *35*, 18442–18455.
166. Li, Y.; Yu, A.W.; Meng, T.; Caine, B.; Ngiam, J.; Peng, D.; Shen, J.; Lu, Y.; Zhou, D.; Le, Q.V.; et al. Deepfusion: Lidar-camera deep fusion for multi-modal 3D object detection. In Proceedings of the EEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 18–24 June 2022; pp. 17182–17191.
167. Liu, Z.; Tang, H.; Amini, A.; Yang, X.; Mao, H.; Rus, D.L.; Han, S. Bevfusion: Multi-task multi-sensor fusion with unified bird's-eye view representation. In Proceedings of the 2023 IEEE International Conference on Robotics and Automation (ICRA), London, UK, 29 May–2 June 2023; pp. 2774–2781.
168. Liu, Z.; Lin, Y.; Cao, Y.; Hu, H.; Wei, Y.; Zhang, Z.; Lin, S.; Guo, B. Swin transformer: Hierarchical vision transformer using shifted windows. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Nashville, TN, USA, 20–25 June 2021; pp. 10012–10022.
169. Yu, F.; Wang, D.; Shelhamer, E.; Darrell, T. Deep layer aggregation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 2403–2412.
170. Lee, Y.; Hwang, J.w.; Lee, S.; Bae, Y.; Park, J. An energy and GPU-computation efficient backbone network for real-time object detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops, Long Beach, CA, USA, 16–20 June 2019; pp. 752–760.
171. Chen, X.; Zhang, T.; Wang, Y.; Wang, Y.; Zhao, H. Futr3D: A unified sensor fusion framework for 3D detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Waikoloa, HI, USA, 2–7 January 2023; pp. 172–181.
172. Yin, T.; Zhou, X.; Krähenbühl, P. Multimodal virtual point 3D detection. *Adv. Neural Inf. Process. Syst.* **2021**, *34*, 16494–16507.
173. Wang, C.; Ma, C.; Zhu, M.; Yang, X. Pointaugmenting: Cross-modal augmentation for 3D object detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 20–25 June 2021; pp. 11794–11803.
174. Chen, Z.; Li, Z.; Zhang, S.; Fang, L.; Jiang, Q.; Zhao, F.; Zhou, B.; Zhao, H. AutoAlign: Pixel-instance feature aggregation for multi-modal 3D object detection. *arXiv* **2022**, arXiv:2201.06493.
175. Chen, Z.; Li, Z.; Zhang, S.; Fang, L.; Jiang, Q.; Zhao, F. Autoalignv2: Deformable feature aggregation for dynamic multi-modal 3D object detection. *arXiv* **2022**, arXiv:2207.10316.
176. Yao, T.; Pan, Y.; Li, Y.; Ngo, C.W.; Mei, T. Wave-vit: Unifying wavelet and transformers for visual representation learning. In Proceedings of the European Conference on Computer Vision, Tel Aviv, Israel, 23–27 October 2022; Springer: Tel Aviv, Israel, 2022; pp. 328–345.
177. Alaba, S.Y.; Ball, J.E. Wcnn3D: Wavelet convolutional neural network-based 3D object detection for autonomous driving. *Sensors* **2022**, *22*, 7010. [CrossRef]
178. Zhang, J.; Liu, H.; Lu, J. A semi-supervised 3D object detection method for autonomous driving. *Displays* **2022**, *71*, 102117. [CrossRef]
179. Wang, M.; Deng, W. Deep visual domain adaptation: A survey. *Neurocomputing* **2018**, *312*, 135–153. [CrossRef]
180. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You only look once: Unified, real-time object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 779–788.
181. Dosovitskiy, A.; Ros, G.; Codevilla, F.; Lopez, A.; Koltun, V. CARLA: An open urban driving simulator. In Proceedings of the Conference on Robot Learning, Mountain View, CA, USA, 13–15 November 2017; pp. 1–16.
182. Hudson, C.; Goodin, C.; Miller, Z.; Wheeler, W.; Carruth, D. Mississippi State University autonomous vehicle simulation library. In Proceedings of the Ground Vehicle Systems Engineering and Technology Symposium, Novi , MI, USA, 11–13 August 2020; pp. 11–13.