IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 1 August 2024, accepted 28 August 2024, date of publication 10 September 2024, date of current version 29 November 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3456893

RESEARCH ARTICLE

Deep Learning Frontiers in 3D Object Detection: A Comprehensive Review for Autonomous Driving

AMBATI PRAVALLIKA[1,2], MOHAMMAD FARUKH HASHMI[1], (Senior Member, IEEE), AND ADITYA GUPTA[3]

[1]Department of Electronics and Communication Engineering, National Institute of Technology Warangal, Warangal 506004, India
[2]Department of Electronics and Communication Engineering, VNR Vignana Jyothi Institute of Engineering & Technology, Bachupally, Hyderabad 500090, India
[3]Department of Information and Communication Technology, University of Agder, 4886 Grimstad, Norway

Corresponding author: Aditya Gupta (aditya.gupta@uia.no)

ABSTRACT Self-driving cars or autonomous vehicles (AVs) represent a transformative technology with the potential to revolutionize transportation. The rise of self-driving cars has driven remarkable progress in 3D object detection technologies, crucial in safe and efficient autonomous driving. This analysis explores the pivotal function of three-dimensional object detection in improving AV safety and performance, underscoring its importance within the larger framework of self-driving vehicle systems. We offer a thorough examination of techniques, including deep learning frameworks such as Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs), to assess their advantages and drawbacks in 3D object detection. The progression of reference datasets, such as KITTI, Waymo, and NuScenes, was examined, emphasizing their crucial role in propelling detection algorithms forward and enabling comparative studies across diverse methodologies. Key performance evaluation metrics, including Average Precision (AP) and Intersection over Union (IoU), are essential for assessing detection accuracy. Furthermore, we investigated the integration of computer vision and deep learning techniques in object recognition, showing their impact on improving the perceptual capabilities of AVs. This paper also addresses significant challenges in 3D object detection, such as occlusion, scale variation, and the need for real-time processing, while proposing future research directions to overcome these obstacles. This work investigates the most recent 3D object detection methods for self-driving cars, emphasizing the importance of advanced deep learning models and multi-sensor fusion methods. In addition, we identify crucial topics for further investigation, such as enhancing sensor fusion algorithms, increasing computational efficiency, and addressing ethical, security, and privacy concerns. The utilization of these technologies in real-world autonomous driving scenarios is examined by specifying their possible advantages and constraints. It provides valuable insights for researchers and practitioners to guide the development of robust 3D object detection systems crucial for the safe deployment of autonomous driving technologies.

INDEX TERMS Autonomous driving systems, CNNs, 3D object detection.

I. INTRODUCTION

A. OVERVIEW OF SELF-DRIVING CARS

One of the most discussed technologies today is self-driving cars, often called autonomous vehicles (AVs). These groundbreaking vehicles can transform the way individuals travel. An autonomous vehicle is defined by the University of Michigan Center for Sustainable Systems as one that uses ''technology to partially or entirely replace the human driver in navigating a vehicle from an origin to a destination while avoiding road hazards and responding to traffic conditions.'' According to the SAE [1], the six levels of ''automated'' driving, previously known as the Society of Automotive Engineers, range from the first three levels (starting from zero).

The associate editor coordinating the review of this manuscript and approving it for publication was Rajeeb Dey.

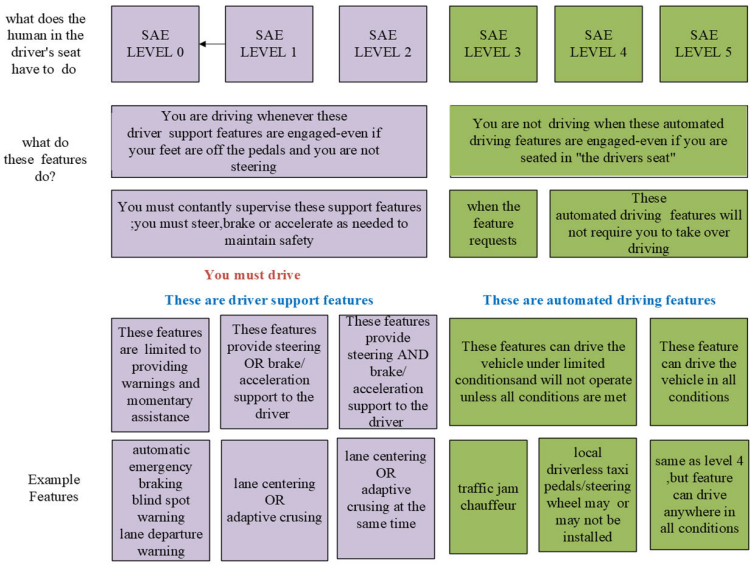
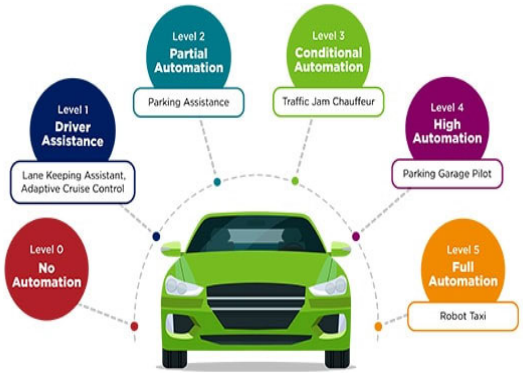


FIGURE 1. The SAE defines six levels of driving automation ranging from 0 (fully manual) to 5 (fully autonomous).

In these levels, the driver controls the vehicle but receives assistance from automated warnings and safety features like obscured area warnings and automatic emergency braking. Levels 3 and 4 denote technological advancements where the vehicle can be autonomous driving in specific situations, although it may still necessitate human intervention for control. Level 5 represents a completely autonomous vehicle operating without human intervention in controlling its functions. This is the sole level at which a vehicle is deemed completely autonomous.



Many companies and universities will put much effort into improving self-driving technology from Level 3 to Level 4 between 2020 and 2030.

The development of fully autonomous vehicles is accelerating at a rate never seen before. With the help of cutting-edge technologies, these cars can detect their environment and drive themselves safely, requiring little to no human input whatsoever. These vehicles rely on cameras, lasers, and detailed maps to perceive their surroundings, identifying objects like cars, pedestrians, and traffic lights. This accurate perception is crucial for safe navigation, like good vision is essential for human drivers [2].

Figure 2. Shows how data flows from sensors to the perception module, then through planning, and finally to control, which directs the vehicle's actions. The Perception System uses sensor fusion [3] to collect real-time data, determine location, update maps, and detect objects like vehicles and pedestrians. The Decision-Making System [4] processes this information to plan routes, predict behaviors, interpret traffic signs, and execute maneuvers such as steering, acceleration, and braking using algorithms like reinforcement learning [5].

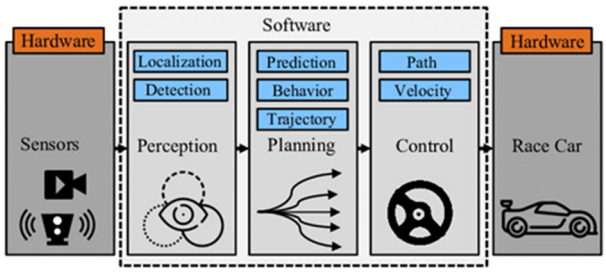


FIGURE 2. Modular perception-planning-action.

The potential advantages of autonomous vehicles are many, and they include less traffic, fewer accidents (because of fewer human errors), more accessible access for individuals with disabilities, and better fuel economy. Broad use of AVs can save 90% of accidents caused by human mistakes and cut traffic congestion by 60%, according to studies [6]. Moreover, AVs can significantly improve mobility and quality of life for individuals with disabilities.

Self-driving cars still face challenges such as occlusion and scale variation, adversarial attacks, cybersecurity, ethical issues, and evolving legal requirements. Regulatory concerns include data privacy, safety standards, and liability. Cybersecurity threats like hacking or sensor spoofing complicate the safe operation of AVs. Safety dilemmas also arise,

as AVs must balance passenger safety with that of other road users, raising questions about decision-making, transparency, and accountability. Despite these challenges, self-driving cars promise a future of safer, more efficient transportation. Addressing these challenges through advanced techniques and robust model architectures is crucial for improving the effectiveness of 2D object detection in applications like autonomous driving.

There has been a growing interest in object detection research in recent years. According to Google Researcher, over 37,000 references to this topic are in the literature. These studies concentrate on developing, enhancing, and applying object detection models, underscoring this research area's significance. Table 1 shows the survey of related works in existing review articles.

B. IMPORTANCE OF 3-D OBJECT DETECTION FOR SELF-DRIVING CARS

One of the primary goals of computer vision is object detection, which is to identify and categorize objects in digital pictures. While 2D object detection continues to garner much attention, 3D object detection has expanded the range of detection techniques. It is now a hot topic, particularly in autonomous driving. Unlike 2D object detection, which focuses solely on predicting object boundaries in two dimensions, 3D object detection leverages sensor data to provide detailed insights into an object's dimensions, position coordinates, speed, and orientation.

This capability is pivotal for autonomous driving and robotics applications, where real-time perception of traffic conditions and efficient route planning are essential. Figure 3. Illustrates the process of 2D and 3D object detection in autonomous vehicles using image and point cloud data.

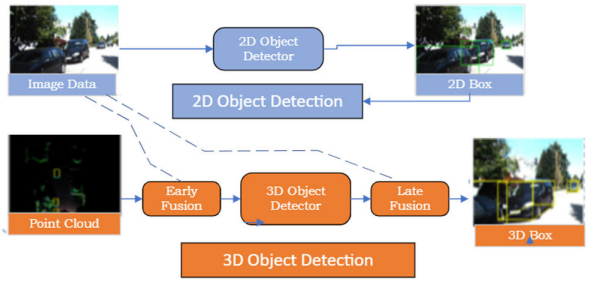


FIGURE 3. The comparison between 2D detection and 3D detection.

3-D Object Detection is a crucial technology for autonomous vehicles, as it allows them to identify and locate objects in three-dimensional space accurately.

2D object detection focuses solely on predicting the 2D boundary box of an object, which doesn't fully meet the needs of real-world scenarios in 3D space. In contrast, 3D object detection aims to use sensor data to provide detailed information such as an object's 3D size, position coordinates, speed, and heading angle. This capability is crucial for applications like autonomous vehicles and intelligent robots, enabling them to perceive real-time traffic conditions and plan optimal routes.

3D object detection creates a 3D bounding box around each object of interest in an image and assigns a class label. A collection of seven parameters can be used to encode a 3D bounding box: $(x, y, z, h, w, l, \theta)$, which include the object's size (height, width, and length), the heading angle (θ), and the coordinates of the object's center (x, y, z). Hardware components such as mono and stereo cameras, infrared or visible light cameras, RADAR (radio detection and ranging), LiDAR (light detection and range), and gated cameras are some of the hardware components used in object detection [17].

The depth information provided by 3D object detection not only aids in identifying and locating objects but also enhances our understanding of their size, orientation, and precise spatial positioning. This depth perception is crucial for robust detection and tracking of vehicles, pedestrians, and obstacles, enabling autonomous vehicles to make informed decisions and navigate safely [18]. Furthermore, 3D object detection systems demonstrate adaptability to diverse environmental conditions, including variations in weather, lighting, and obstructed views, enhancing their reliability and performance in real-world scenarios [19].

Figure 4. Illustrates an example of 3D object detection shown in the image (upper one) and LiDAR point cloud (lower one) scene from a self-driving car's perspective, showcasing the application of 3D object detection technology. The green and red bounding boxes highlight the detected objects (such as cars) on the road, indicating their positions and dimensions in 3D space. This visual representation demonstrates how the autonomous vehicle's perception system identifies and localizes various objects within its environment, which is crucial for safe navigation and decision-making. The detection and classification of objects allow the vehicle to understand its surroundings and plan appropriate actions accordingly.

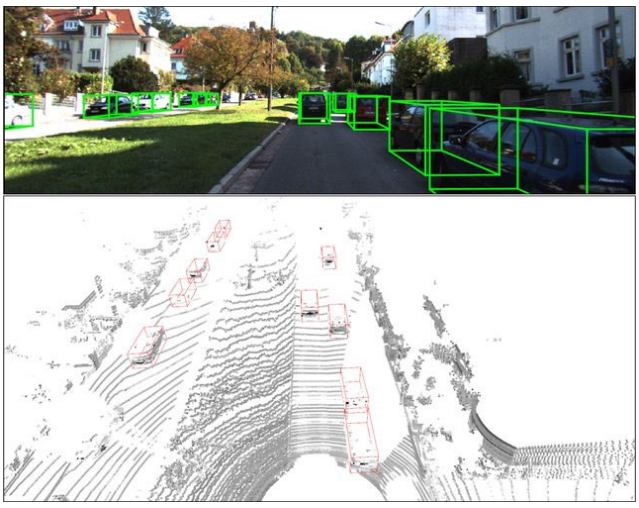


FIGURE 4. Example of 3D object detection in a self-driving car: The top image shows 3D bounding boxes applied to objects in the camera view, while the bottom image displays corresponding 3D bounding boxes in the point cloud data [16].

Accurate 3D object detection, like human vision for computers, is crucial for meeting regulatory standards and

TABLE 1. Related works in existing surveys.

Reference	3D Detection Methods	Dataset Discuss-ion	Metrics	3D Object Detection Pipeline	Backbone Architecture	Software Libraries	Applications	Aim of Study /Survey
[7]	✓	✓	✓	✓	✓	✗	✓	Robustness-aware 3D object detection in autonomous driving
[8]	✓	✓	✓	✓	✓	✗	✗	Survey and systematization of 3D object detection models and methods
[9]	✓	✓	✗	✗	✗	✓	✗	Survey of LIDAR-based 3D object detection methods with deep learning for autonomous driving
[10]	✓	✓	✗	✗	✗	✗	✗	Survey of deep learning-based methods and datasets for monocular 3D object detection
[11]	✓	✗	✗	✗	✗	✓	✗	Survey of deep learning multi-sensor fusion-based 3D object detection for autonomous driving
[12]	✓	✗	✗	✗	✗	✓	✗	Emerging trends in multimodal fusion for 3D object detection in autonomous vehicles
[13]	✓	✗	✗	✗	✗	✗	✗	3D reconstruction, single RGB image, deep learning
[14]	✓	✗	✗	✗	✗	✗	✗	3D object detection, dark scenes, multimodal sensors
[15]	✓	✗	✓	✗	✗	✓	✓	Real-time 3D object detection, autonomous driving
Ours	✓	✓	✓	✓	✓	✓	✓	3D object detection methods including backbone architecture.

ensuring trust in self-driving technology. It enables machines to perceive and understand the three-dimensional world using sensors like cameras or LiDAR, which is vital for applications in autonomous driving, robotics, and augmented reality [20]. This capability allows vehicles to navigate complex environments safely and effectively, contributing significantly to their operational reliability and public acceptance.

C. COMPUTER VISION AND DEEP LEARNING IN OBJECT RECOGNITION

Object detection relies on deep learning and computer vision to accurately identify and locate things in photos and videos. Recognizing objects in images, determining their distances from one another, and following their exact positions are all part of object detection. Autonomous driving systems (ADS) rely heavily on 3D object detection, and advances have greatly improved in computer vision and deep learning technology. These innovations have revolutionized how cars see and understand their environments, paving the way for more precise and dependable detection of things like people, other vehicles, traffic signs, and other roadblocks.

Several deep learning methods, such as You Only Look Once (YOLO) and Convolutional Neural Networks (CNN), have been suggested to improve object recognition accuracy. Various industries can benefit from these technologies, including healthcare, security monitoring, and intelligent driving assistance. Researchers are utilizing deep learning algorithms and Computer Vision techniques to create effective object detection systems that can reliably and accurately identify various items. Figure 5. Illustrates the relationship and hierarchy between Artificial Intelligence (AI), Machine Learning (ML), Deep Learning, and Computer Vision.

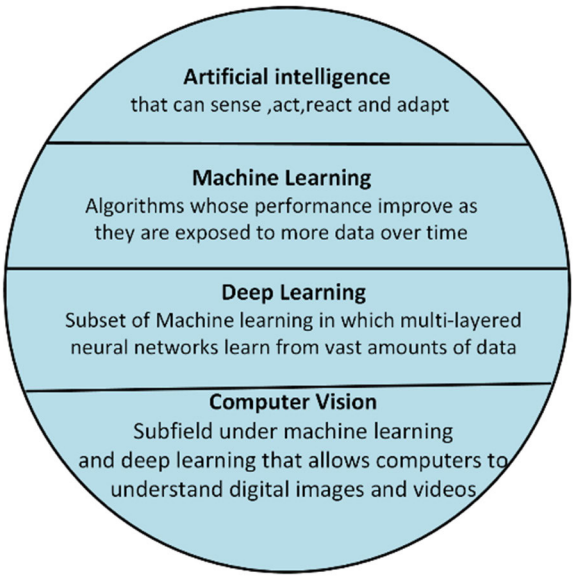


FIGURE 5. Deep learning vs computer vision results.

1) EXPLORING THE POTENTIAL OF DEEP LEARNING IN 3D OBJECT RECOGNITION

In traditional ML, feature extraction is a separate and often manual step requiring domain expertise. In deep learning,

feature extraction is integrated into the model and occurs automatically. Figure 6. Illustrates the streamlined process of deep learning compared to traditional machine learning, highlighting the advantages of automation and integration in handling complex data tasks. Images captured by cameras are a projection of three-dimensional space into two-dimensional vision, which results in the loss of spatial information and fails to meet people's expectations. It is necessary to consider additional 3D spatial data. Many 3D methods are advancing quickly, so 3D sensors like LiDARs, 3D scanners, and RGB-D cameras are getting cheaper and more widely available [21].

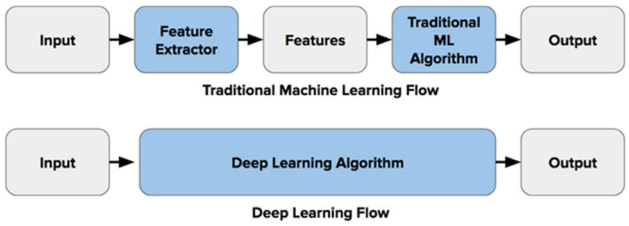


FIGURE 6. Workflow of traditional machine learning and deep learning approaches.

Point clouds acquired with LiDARs are the primary focus of this review. In a standard format, we have geometric location information from each point, and some of those points may even have RGB data. Researchers have shown that accurate environmental perception and precise position are crucial for an autonomous driving system to reliably navigate, make information decisions, and drive safely in complex and dynamic situations. Point clouds are more resistant to changes in lighting than photos, whose data quality is affected by it [22]. Since then, point cloud data has seen extensive application in this domain. Object detection goals and inquiries have evolved in recent years due to breakthroughs in object detection technology. Due to their irregularity and sparsity, 2D object detection algorithms cannot be applied to 3D point clouds. So, to make such methods applicable to 3D scenarios, 2D object detection algorithms must be updated [23].

2) 3D OBJECT DETECTION USING COMPUTER VISION METHODS

Computer vision techniques have evolved significantly in 3D object detection. Traditional computer vision methods for 2D object detection focus on analyzing images to identify and locate objects within a scene. These methods typically rely on geometric and statistical techniques to extract features and classify objects. These methods involved region proposal using sliding windows, hand-designed feature extraction (like HOG and SIFT), and classification/regression stages [24]. However, these approaches were limited by slow speeds, low accuracy, and high computational costs. Deep convolutional neural networks (CNNs) have largely replaced these methods, offering improved performance. Nevertheless, traditional edge detection and image segmentation techniques remain crucial, often complementing deep learning algorithms. These combined approaches have found applications in robotics, autonomous vehicles, augmented reality, and industrial inspection, where understanding 3D environments is essential.

3) INTEGRATION AND APPLICATIONS IN AUTONOMOUS DRIVING

Integrating computer vision and deep learning has propelled 3D object detection to new heights, enabling more accurate and efficient perception systems in autonomous driving. By leveraging advanced sensor technologies, deep learning architectures, and innovative techniques, researchers and practitioners can overcome existing challenges and pave the way for safer and more reliable autonomous vehicles. Future advancements will improve computational efficiency, robustness, and interpretability, ensuring that 3D object detection continues evolving and meeting real-world application demands.

Much improvement in ADS's capabilities has resulted from combining deep learning with computer vision techniques. Using these technologies, businesses such as Waymo and Tesla have developed perception systems that can handle various driving situations [25]. Additionally, the limits of achievable goals are being pushed by ongoing research and development in this discipline. 3D object detection systems are anticipated to continue to perform better because of advancements in neural network topologies, such as using transformers in vision tasks [26]. Intricate urban settings have recently seen the effective use of deep learning models for object identification and categorization. This technology finds wide-ranging applications, from autonomous driving and surveillance to healthcare and industrial automation, driving innovation and efficiency across multiple domains.

Deep learning and computer vision have revolutionized object recognition, impacting diverse fields such as healthcare and autonomous vehicles. To accurately detect objects in photos, deep learning models and convolutional neural

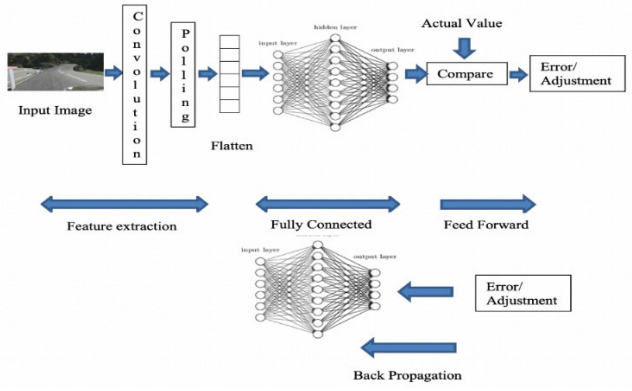


FIGURE 7. Deep Convolutional Neural Network (CNN) used for image recognition and classification [27].

networks (CNNs) succeed in identifying patterns and characteristics [28].

Models of this kind are trained rigorously on labeled datasets to identify visual elements and their interrelationships. By analyzing visual input, extracting relevant features, and evaluating contexts, computer vision algorithms enhance their ability to detect objects. Figure 7. Illustrates the process of object detection using a Convolutional Neural Network (CNN), where convolution, pooling, and backpropagation enable the CNN to learn from the input image and accurately detect and classify objects within it.

Surveillance, industrial automation, augmented reality, and other fields use this technology, boosting productivity and innovation [29].

In addition, new specialized architectures and the integration of many sensor modalities have propelled 3D object detection forward recently. Some deep learning models developed for processing 3D point clouds from LiDAR sensors are PointNet [30] and PointPillars [31]. 3D object detection has also demonstrated encouraging results when using sensor fusion approaches integrating radar, LiDAR, and camera data [32].

II. RESEARCH METHODOLOGY

2D and 3D object detection are covered in section IV. The possible uses of 3D object detection in autonomous driving scenarios and the software libraries used to implement these object detection algorithms are discussed in Section V. In addition, the difficulties and potential avenues for further study in this area are discussed in section VII.

A. SYSTEMATIC REVIEW PROCESS

Our study employed the Preferred Reporting Items for Systematic Reviews and Meta-Analyses (PRISMA) framework [27] to systematically review recent literature on 3D object detection for autonomous vehicles. Our initial search across Google Scholar, Web of Science, IEEE Xplore, PubMed, and Dimensions yielded 1,200 potentially relevant articles using keywords such as 3D object detection, point clouds, LiDAR, and autonomous vehicles.

After addressing duplicate entries, we excluded 250 duplicate documents. An additional 308 records were removed for being unrelated to our study, leaving 642 papers. We then screened the titles and abstracts for relevance, further narrowing our selection. Papers that did not involve deep learning techniques or were not directly related to 3D object detection were excluded, reducing the list by 358 papers. We rigorously conducted a detailed evaluation of the remaining 284 papers.

I was applying inclusion criteria. This led to the exclusion of 82 papers that lacked relevant data, showed non-significant results, or were in a language other than English. Ultimately, two hundred two papers were included in our final

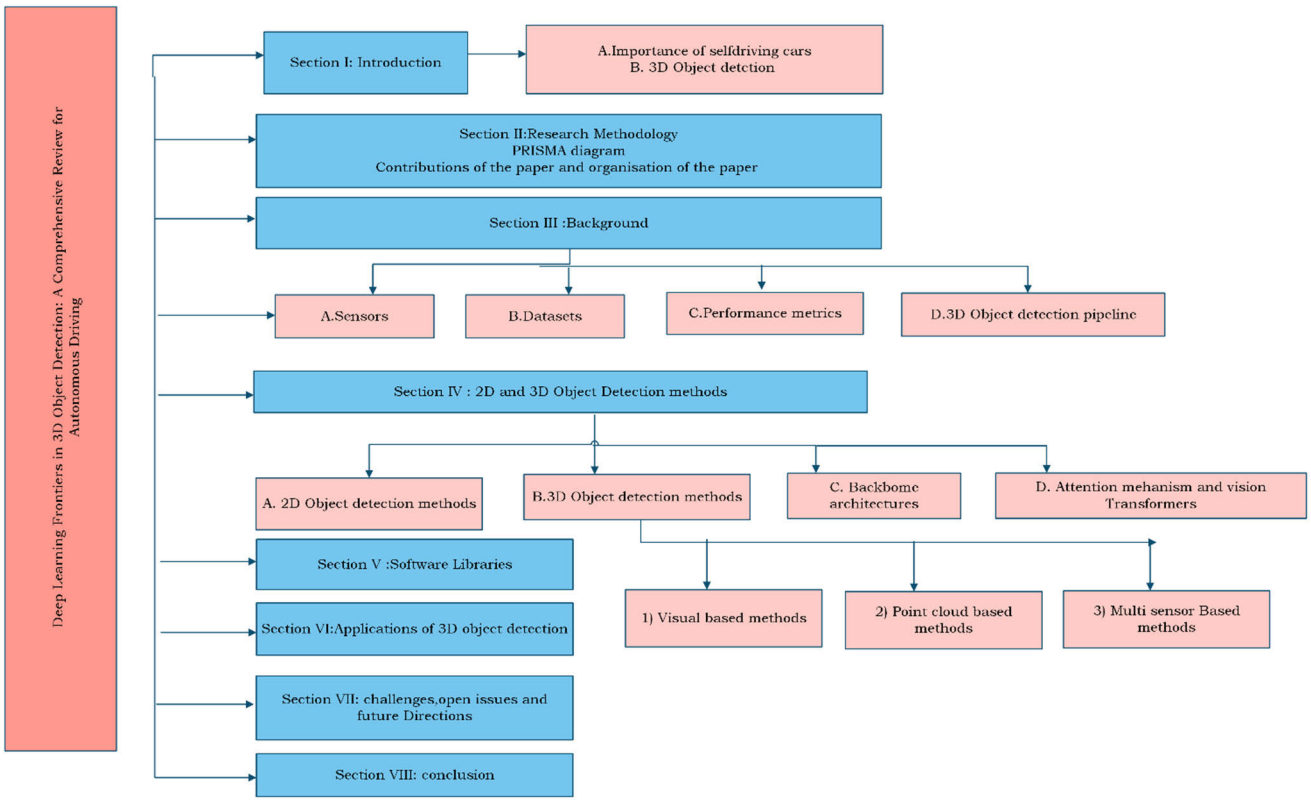


FIGURE 8. Organization of the review paper.

review. These papers were carefully selected to align with our research objective of detecting vehicles in three-dimensional space using deep learning algorithms. Figure 9. Illustrates the systematic process guided by the PRISMA framework, ensuring a robust foundation for our study.

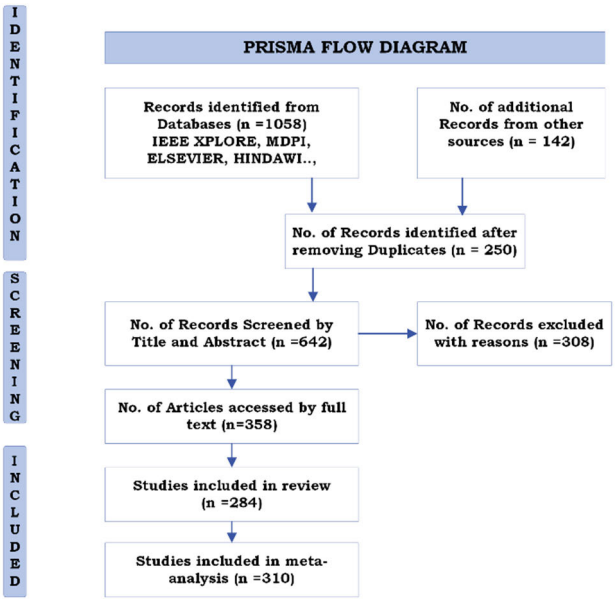


FIGURE 9. PRISMA approach for literature review.

B. PURPOSE OF THE REVIEW PAPER

This review paper aims to synthesize and assess the state of the art in 3D Object Detection (or Detection of Objects in Three Dimensions) by surveying and analyzing relevant research. In doing so, various approaches, tools, algorithms, and databases will be examined for 3D object detection. This review aims to summarize the existing literature, highlight recent developments, and suggest avenues for further study. The planned review has several features, such as a systematic literature review, methodology classification according to usefulness and efficacy, technique comparison, and critical assessment of performance measures, including speed, accuracy, and robustness. Problems like occlusion, scale variation, and real-time processing may also be addressed in the review, along with novel approaches to these issues.

C. ORGANIZATION OF THE REVIEW PAPER

Figure 8. Outlines the logical structure that the paper follows. The introduction presents the central problem and sets the stage for the rest of the paper. Next, the datasets used, and the procedures for preparing the input data are described in detail. The paper then examines feature selection, feature extraction, and classification methods. Following this, the literature review is organized using the "PRISMA flow" method to categorize various types of relevant research. The review discusses past, present, and future research goals, possible applications, obstacles, and gaps in the literature.

The significance of self-driving cars and 3D object detection is discussed in Section I, which introduces the study approach and outlines the individual contributions. The following are the contributions linked to 3D object detection for autonomous driving systems:

I. Examine various methodologies, tools, algorithms, and databases for 3D object detection.

II. Assess the impact of advancements in 3D object detection on enhancing the safety, efficiency, and overall performance of autonomous driving systems.

III. Discuss problems such as occlusion, scale variation, and real-time processing and propose innovative solutions.

Section III discusses the fundamentals of 3D object detection, including sensors, datasets, pipelines, and performance metrics. Significant content is introduced in Section IV, which offers an in-depth examination of various methodologies for both 2D and 3D object detection. Some examples of these methods are visual-based approaches, point cloud-based methods, multi-sensor fusion strategies, and backbone architectures for these models.

Figure 10. Below is the number of articles published in various top journals from 2020 to 2024.

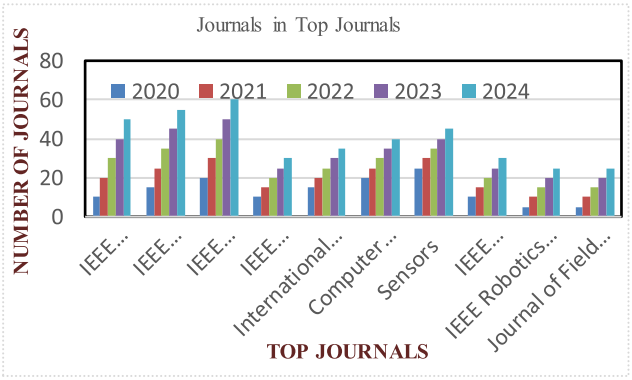


FIGURE 10. Number of journals published in various top journals from 2020 to 2024.

III. BACKGROUND

A. SENSORS

1) LiDAR (LIGHT DETECTION AND RANGING) SENSORS

LiDAR sensors emit laser pulses and measure the time it takes for these pulses to return, allowing them to determine distances to objects accurately. This data is represented as a point cloud involving a set of 3D coordinates that provide in-depth information and intensity levels of the surroundings. Point clouds are sparse and lack structure, unlike images, but they offer robustness in diverse lighting conditions and adverse weather. LiDAR can detect objects at distances exceeding 200 meters, though the density of points on similarly sized objects decreases with distance. However, LiDAR point clouds excel in capturing precise details such as object shape, size, and spatial location, making them essential for tasks like 3D object detection and localization in autonomous driving applications [33].

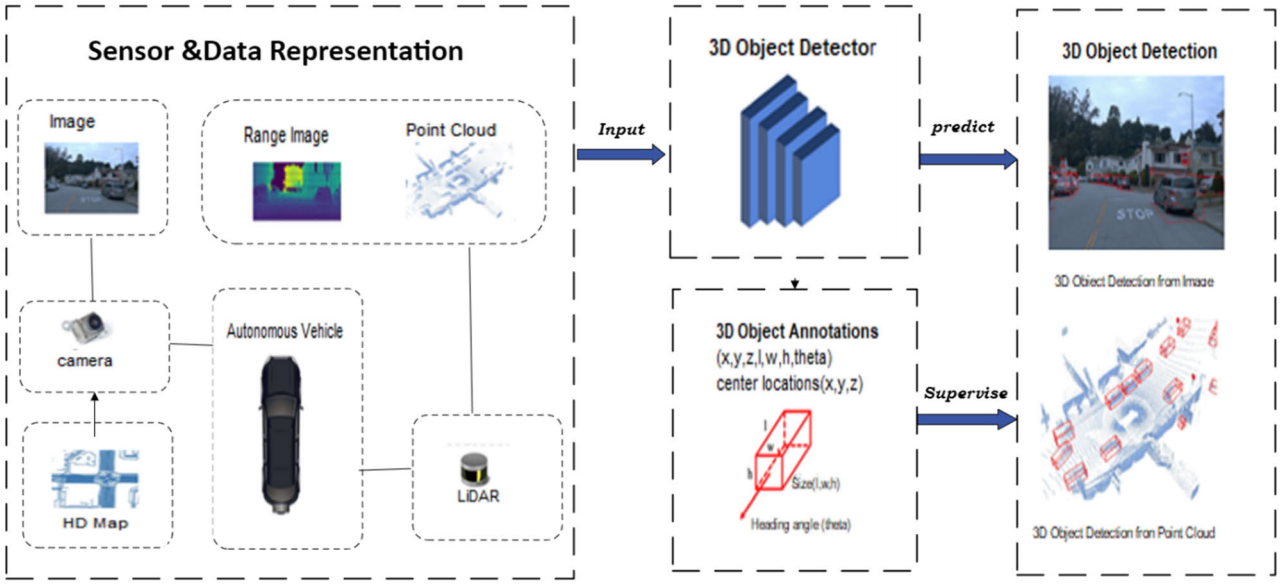


FIGURE 11. An illustration of 3D object detection for autonomous vehicles.

3D object detection methods often employ fusion techniques to combine data from cameras and LiDAR sensors, leveraging their strengths. However, before fusion can occur, the camera calibration and sensor registration are necessary to establish a unified spatial frame of reference.

Figure 11. Effectively shows the integration of different sensor data for comprehensive 3D object detection, highlighting the steps from data acquisition to object annotation and visualization of detection results. The KITTI dataset [34] is widely utilized in autonomous driving research due to its synchronized LiDAR point clouds, camera images, and detailed annotations for various object classes like vehicles, pedestrians, and cyclists.

State-of-the-art algorithms such as PointNet [30], PointRCNN [35], and PointPillars [31] have demonstrated impressive performance on benchmark datasets using LiDAR point clouds.

Despite their advantages in providing accurate, in-depth information and capturing detailed object characteristics, LiDAR systems have limitations. They are costly compared to other sensors and typically offer a limited field of view, necessitating multiple units for comprehensive coverage. Moreover, their performance can be hindered by adverse weather conditions such as fog, heavy rain, or snow, which can affect laser beam transmission.

2) MONOCULAR CAMERAS

Monocular cameras use a single lens to capture images and are critical in 3D object detection for autonomous driving systems (ADS). These cameras are cost-effective, lightweight, and versatile, making them attractive for many autonomous vehicle applications. Despite the inherent challenge of lacking direct depth information, monocular cameras have demonstrated significant potential in accurately detecting and localizing objects in 3D space through advanced computer vision and deep learning techniques. Monocular-based 3D object detection adapts the established framework of 2D object detection to predict 3D properties of objects from single images. Unlike sensors like LiDAR, monocular cameras capture images that lack direct depth information, presenting challenges for accurately estimating 3D object dimensions and positions.

To overcome these challenges, researchers have developed various techniques. These include leveraging geometric and contextual cues within the image [36], incorporating motion information to infer depth dynamics, utilizing separate networks for depth estimation [37], and employing synthetic datasets with annotated 3D information [38]. These approaches aim to enhance the accuracy of 3D object detection using monocular cameras. Despite its limitations, monocular 3D object detection remains attractive due to its cost-effectiveness and simplicity, especially in scenarios where additional sensors such as LiDAR or stereo cameras are not feasible or available. Figure 12. Depicts the context of object detection that demonstrates the transformation and relationship between 2D and 3D bounding boxes in object detection.

3) STEREO CAMERAS

Stereo cameras estimate depth through triangulation by analyzing the disparity between images captured from different viewpoints. They offer a cost-effective depth perception solution compared to LiDAR, making them appealing for autonomous driving applications [39]. Deep learning models like Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs) process stereo data to estimate

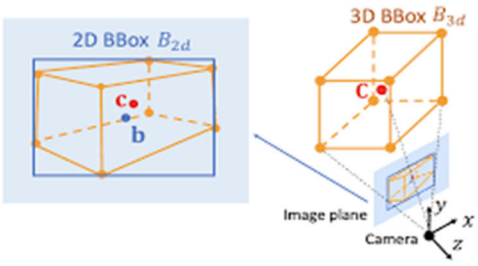


FIGURE 12. Notation for 3D bounding box localization [44] (a) 2D Projection (b) Camera coordinate frame.

depth maps, which can be combined with 2D detections or used directly for 3D object detection [40]. However, stereo depth estimation has limited accuracy and range compared to Li-DAR, especially in low-texture environments or at more considerable distances. Stereo matching can also be computationally expensive and struggle with occlusions or rapid motion [42].

Compared to monocular methods, stereo-based 3D detection leverages depth cues to resolve scale ambiguity and occlusion issues, enabling more accurate 3D localization and size estimation [43]. For instance, Stereo R-CNN [41] extends Faster R-CNN [45] with a 3D box estimation module using stereo image features and disparity maps, improving monocular baselines on the KITTI dataset. PseudoLiDAR [46] converts stereo depth maps to pseudo-LiDAR point clouds, allowing established LiDAR-based detectors to achieve state-of-the-art performance on KITTI.

Despite their advantages, stereo-based methods are limited by factors such as camera baseline, resolution, texture-less regions, low-light conditions, and reflective surfaces, which can impact disparity estimation accuracy [47]. Furthermore, while stereo systems can mitigate some limitations of monocular systems, they still face challenges in dynamic environments with rapid motion and varying lighting conditions.

Recent advancements in deep learning have introduced techniques to improve stereo depth estimation. For example, end-to-end learning frameworks have been developed to directly predict disparity maps from stereo image pairs, enhancing robustness and accuracy. Additionally, integrating temporal information from video sequences can help address occlusions and rapid motion issues by providing additional context for depth estimation. Moreover, developing hybrid systems that combine stereo and other sensor modalities, such as LiDAR and RADAR, is an emerging area of research. These multi-sensor fusion approaches can leverage the strengths of each sensor type, potentially overcoming individual limitations and achieving more robust and accurate perception for autonomous driving systems.

4) DEPTH SENSORS (ToF, STRUCTURED LIGHT)

Depth sensors, such as Time-of-Flight (ToF) and structured light sensors, directly measure the depth of objects in the scene, providing accurate depth information without the need for complex algorithms or multi-view setups [48]. ToF sensors measure the time it takes for a light signal to travel to an object and back, while structured light sensors project a known pattern onto the scene and analyze the deformation of the pattern to estimate depth [49]. Depth sensors have been employed in various robotics and computer vision applications, including 3D object detection and recognition [50]. In autonomous driving, depth sensors can provide valuable information for short-range applications, such as detecting obstacles or pedestrians near the vehicle [51].

One advantage of depth sensors is their robustness to varying lighting conditions, as they rely on active illumination rather than ambient light. However, their range and field of view are typically limited compared to LiDAR, and their performance can be affected by surface properties, such as highly reflective or transparent materials. Additionally, integrating depth sensors like ToF and structured light sensors with stereo camera systems can provide complementary depth information, enhancing the accuracy and reliability of 3D object detection and recognition. This hybrid approach can help mitigate the limitations of each sensor type, leading to improved performance in a broader range of scenarios and environmental conditions.

5) THERMAL CAMERAS

Thermal cameras detect infrared radiation emitted by objects, capturing heat signatures, which makes them particularly useful for detecting pedestrians, animals, and other warm-blooded objects in autonomous driving scenarios [52]. These sensors are invaluable in low-light or challenging lighting conditions, where visible-light cameras may fail. Thermal imaging data has been effectively used alongside visible-light images and other sensor modalities for various perception tasks, including pedestrian detection, object tracking, and semantic segmentation. Deep learning architectures have been developed to fuse thermal and visible-light data, capitalizing on the complementary strengths of each modality [53].

However, thermal cameras also present certain limitations. Their resolution is generally lower than traditional cameras, making it challenging to identify smaller objects or specific details in the imagery. Additionally, like monocular cameras, thermal cameras lack inherent depth information, complicating the accurate perception of three-dimensional objects.

Several research efforts have addressed these limitations by exploring multi-sensor fusion techniques. Choi and Kim [54] investigated thermal infrared imaging in autonomous driving and suggested combining thermal cameras with other sensors, such as LiDAR or radar, could enhance perception capabilities. Portmann et al. [55] proposed a method for people detection and tracking from aerial thermal views, utilizing thermal cameras in conjunction with visible-light cameras.

6) RADAR

Radar sensors emit radio waves and measure the reflected signals to detect objects and estimate their velocity and

relative position. One of the critical advantages of radar is its robustness to various weather conditions, such as rain, fog, and snow, which can significantly impact the performance of other sensors. Additionally, radar can provide accurate range and velocity measurements, making it valuable for tracking moving objects and predicting their trajectories. However, radar lacks visual and contextual information, and its resolution and angular precision are generally lower than LiDAR or camera-based systems [56]. Radar can also struggle with detecting stationary objects and may require additional processing or sensor fusion to achieve reliable 3D object detection and localization.

In autonomous driving, radar is often used with other sensors like cameras and LiDAR, providing complementary information about the dynamics and motion of objects in the environment [57]. Radar data has been used for various perception tasks, including object detection, tracking, and velocity estimation [58]. Deep learning architectures have been developed to fuse radar data with other sensor modalities, such as cameras and LiDAR, for improved 3D object detection and tracking performance [59].

7) ULTRASONIC SENSORS

Ultrasonic sensors use high-frequency sound waves to detect nearby objects and measure their distance. In autonomous driving, ultrasonic sensors are often used for short-range applications, such as parking assistance or detecting obstacles near the vehicle [60]. While ultrasonic sensors are inexpensive and robust in various environmental conditions, their limited range and field of view and their susceptibility to interference from other ultrasonic sources restrict their use in autonomous driving to specific scenarios [61]. Additionally, ultrasonic sensors lack visual and contextual information, making them less suitable for complex perception tasks like 3D object detection and classification.

8) INERTIAL MEASUREMENT UNIT (IMU)

Inertial Measurement Units (IMUs) measure acceleration and orientation, providing valuable information for localization and motion tracking. While IMUs cannot directly detect objects, they complement other sensors by accurately measuring the vehicle's movement and orientation, enabling robust sensor fusion and localization [62]. IMU data is often used with GPS, LiDAR, and cameras to estimate the vehicle's position, velocity, and attitude [63]. This information is crucial for tasks like simultaneous localization and mapping (SLAM), motion planning, and object tracking in dynamic environments [64]. However, IMUs are susceptible to drift over time, and their measurements can be affected by temperature, vibrations, and magnetic interference. Therefore, IMU data is typically integrated with other sensor modalities to mitigate these issues and achieve accurate and reliable localization and motion tracking.

9) GLOBAL POSITIONING SYSTEM (GPS)

Global Positioning System (GPS) receivers provide global positioning and navigation information, enabling localization and mapping for autonomous vehicles. GPS data is often combined with other sensors, such as IMUs and cameras, to achieve robust and accurate localization and mapping [65]. While GPS provides global positioning information, its accuracy can be limited in urban environments due to signal obstructions from buildings and other structures.

10) INTEGRATION AND CONNECTIVITY OF VARIOUS SENSOR MODALITIES

In autonomous driving, achieving robust perception, localization, and mapping requires the integration of multiple sensor modalities. Each sensor type—stereo cameras, depth sensors, thermal cameras, radar, ultrasonic sensors, IMUs, and GPS—has its strengths and limitations. Combining these sensors allows autonomous vehicles to leverage the complementary information from each sensor type, overcoming individual limitations and enhancing overall system performance.

Stereo cameras and depth sensors like Time-of-Flight (ToF) and structured light sensors provide detailed depth information crucial for 3D object detection and scene understanding. Thermal cameras offer valuable data in low-light conditions, enhancing the detection of pedestrians and animals. Radar sensors contribute reliable velocity and range measurements, especially in adverse weather conditions, while ultrasonic sensors offer short-range detection capabilities for parking and obstacle avoidance.

IMUs and GPS provide essential localization and navigation information, with IMUs offering precise movement of data and GPS enabling global positioning. Fusing these sensor modalities ensures robust and accurate perception, localization, and mapping, which is crucial for autonomous vehicles' safe and efficient operation. Table 2 discusses the range of each sensor and its advantages, disadvantages, and applications in autonomous driving.

B. 3D OBJECT DETECTION DATASETS

Most object detection algorithms in deep learning (DL) rely on supervised learning, which necessitates training with labeled and annotated images. Aside from some synthetic datasets derived from gaming engines and simulators, most datasets used in autonomous cars are authentic sensor-generated images. Building and testing autonomous driving 3D object identification systems requires realistic driving scenario datasets of high quality. Ground truth annotations for items like pedestrians, cyclists, and automobiles are included in these datasets, along with sensor data such as images, point clouds, and radar. Developing better 3D object detection algorithms for autonomous vehicles relies heavily on benchmark datasets. Here, you can find datasets that are often used for this purpose. A sample frame for 3D object detection is shown in Figure 13.

TABLE 2. Common sensors used in autonomous driving and their comparisons are included.

Sensor	Working Principle	Range	Advantages	Disadvantages	Applications
LiDAR [33]	It uses laser pulses to measure distances and create 3D maps	Up to 200 m	High accuracy, long-range, detailed 3D mapping	High cost, affected by weather conditions	Autonomous vehicles, surveying, mapping
Monocular Camera [36]	Captures 2D images using a single camera	Depends on lens	Low cost, compact, provides visual information	Cannot measure depth directly, requires processing	Object detection, tracking, SLAM
Stereo Camera [39]	Utilizes two cameras to capture 3D information based on parallax	Up to 50 m	Provides in-depth information at a lower cost than LiDAR	Limited range and accuracy depend on baseline distance	3D mapping, obstacle avoidance
Depth Sensor [48]	Techniques like Time-of-Flight (ToF) or structured light measure depth	Up to 10 m	Direct depth measurement, compact	Limited range, affected by ambient light	Gesture recognition, 3D scanning, robot navigation
Thermal Camera [52]	Detects infrared radiation to create thermal images	Varies	Useful for night vision, detects heat signatures	Limited resolution, affected by environmental factors	Surveillance, search and rescue
Radar [56]	Uses radio waves to detect and locate objects	Up to 200 m	Long range, unaffected by weather	Lower resolution compared to LiDAR	Automotive safety, weather monitoring
Ultrasonic Sensor [60]	Uses high-frequency sound waves to measure distances	Up to 10 m	Simple, low-cost, compact	Limited range, affected by environmental noise	Proximity detection, object avoidance
IMU [62]	Measures acceleration and angular rate	N/A	Provides orientation and motion data, high update rate	Accumulates drift over time, requires calibration	Motion tracking, navigation, stabilization
GPS [65]	Uses satellite signals for location determination	Global coverage	Provides absolute positioning	Requires clear sky view, affected by multipath	Navigation, tracking, surveying

1) KITTI DATASET

A popular dataset for autonomous driving research, especially for 3D object detection tasks, is the KITTI (Karlsruhe Institute of Technology and Toyota Technological Institute) dataset [34]. The data is compiled from a vehicle's synchronized sensors, including stereo cameras, LiDAR, and GPS/IMU, and it was gathered while the car was driving in both urban and rural areas of Karlsruhe, Germany. The dataset includes information from a 360° laser scanner Velodyne HDL-64E, four varifocal lenses, two high-resolution video cameras (color and grayscale), and one inertia navigation system (GPS/IMU).

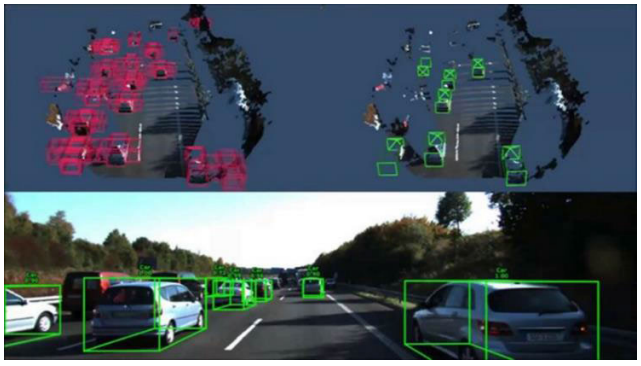


FIGURE 13. A sample frame for 3D object detection from the KITTI dataset.

Annotations of vehicles, people, and bicycles in the KITTI dataset give 3D bounding boxes with exact coordinates and orientations, making them ideal for 3D object detection. The dataset has 7,481 training photos and 7,518 test images, split into multiple portions. Intersection over Union (IoU) and Average Precision (AP) are two measures utilized for evaluation. The "Car" class achieved an AP score of 90.25% using the state-of-the-art method PointRCNN [35] and an AP score of 88.88% using PV-RCNN [66]. KITTI helps build robust 3D object detection algorithms since it accurately represents occlusions and truncations.

2) NuScenes DATASET

For studies involving autonomous vehicles in urban settings, researchers have created the large-scale nuScenes dataset [67]. It is a compilation of information gathered from many sensor modalities, including cameras, LiDAR, radar, and IMU, while a vehicle navigates complicated urban environments in Singapore and Boston. A wide variety of object classes, including cars, pedestrians, bicycles, and traffic cones, are covered in the collection, along with detailed annotations for 3D bounding boxes, characteristics, and tracking information. To detect three-dimensional objects in point clouds, Zhu et al. [68] suggested a network that uses cylindrical and perspective perception and trains and evaluates on the nuScenes dataset. By combining radar and camera data, Nabati and Qi [69] developed CenterFusion,

a center-based 3D object recognition method successfully tested on the nuScenes dataset.

3) A*3D DATASET

Along with RGB pictures and LiDAR data, the A3D Dataset [70] includes 230K 3D object annotations, seven classes, and 39K frames. It is a demanding and representative dataset for autonomous driving applications because it is collected from different locations in Singapore and covers distinct scenarios, hours, and weather conditions. To overcome obstacles like sparse point clouds and occlusion, Ren et al. [71] suggested a 3D object detection method that makes use of the A3D dataset.

4) H3D DATASET

Several cameras, LiDAR, and GPS/IMU sensors were used to compile the Honda Research Institute 3D Dataset (H3D) [72], a 3D object detection and tracking dataset in the San Francisco Bay Area. It features tough settings such as highly interactive, complex, and occluded traffic scenarios in its eight classes and 160 traffic scenes with one million bounding box labels in 27,721 frames. To detect and track 3D objects, Battra et al. [73] suggested a multi-modal fusion method and tested it on the H3D dataset.

5) LIBRE DATASET

The LIBRE dataset [74] was compiled using ten separate LiDAR sensors in three distinct settings: static targets, bad weather, and dynamic traffic near Nagoya University in Japan. The sensors covered a range of models and laser configurations. The file also contains data from auxiliary sensors, such as RGB and infrared cameras, event cameras, IMUs, GNSS, and CAN networks. By analyzing the LIBRE dataset, Zhu et al. [75] investigated improving 3D object detection by integrating several LiDAR sensors.

6) WAYMO OPEN DATASET

Waymo, formerly the Google self-driving car project, has released a large-scale dataset called the Waymo Open Dataset [76]. This dataset comprises approximately twelve million scenes, each comprising camera images and LiDAR data. The dataset includes around twelve million 2D annotated ground truth bounding boxes for the camera images and twelve million 3D annotated ground truth bounding boxes for the LiDAR data. Each scene lasts twenty seconds and is recorded in various urban and suburban settings. Using the Waymo Open Dataset for training and evaluation, PointRCNN, an end-to-end 3D object detection algorithm introduced by Shi et al. [35], achieved state-of-the-art performance. Lang et al. [31] introduced a fast and efficient 3D object detection method called PointPillars. This approach utilizes the Waymo Open Dataset for benchmarking and comparison with other techniques.

7) ASTYX RADAR DATASET

Recorded in Germany utilizing a Velodyne VLP-16 LiDAR, a Point Grey Blackfly camera, and the Astyx 6455 HiRes radar, the Astyx Radar dataset [77] is a radar-centric automobile dataset. With annotations for 3D location, rotation, dimensions, class information, occlusion indicator, and position and dimension uncertainty, the collection contains ground truth data for seven classes: Bus, Car, Cyclist, Motorcyclist, Person, Trailer, and Truck.

8) LYFT L5 DATASET

The Lyft L5 dataset [78] is created using 64-wire radars and other cameras and formatted according to the nuScenes dataset standard. More than 55,000 3D annotated frames, each tagged by a human, are part of the dataset. These include surface maps, underlying HD spatial semantic maps, and nine classes. To recognize 3D objects, Alaba et al. [12] suggested a multi-modal fusion method that trains and evaluates the Lyft L5 dataset.

9) PANDASET

One public dataset Hesai & Scale has made available for autonomous driving is the PandaSet dataset [80]. Information gathered from six cameras, an inbuilt GPS/IMU, and one mechanical and one solid-state LiDAR are all part of it. The dataset includes 48,000 camera images, 16,000 LiDAR sweeps, more than 100 scenes, 28 object classification annotation classes, and 37 semantic segmentation labels. A method for detecting three-dimensional objects using point clouds was suggested by Zhang et al. [81]. They trained and tested their model on the PandaSet dataset. To help with the contextual understanding needed for motion prediction and scene reconstruction, the dataset contains high-definition semantic maps that specify traffic features, road rules, and lane geometry [79].

10) CITYSCAPES DATASET

Stereo video sequences captured from streets in several cities, mostly in Germany and adjacent countries, comprise the Cityscapes collection [82]. With data collected throughout the year and in various climates, this dataset features 30 classes with pixel-level annotations, including parking, sidewalk, pole, road, and person. Using the Cityscapes dataset for both training and evaluation, Luo et al. [83] presented a deep learning method for stereo matching that is both efficient and effective. The Cityscapes dataset is commonly applied in tasks like localization, drivable region detection, depth estimation, and colorization. To detect vehicles using deep learning, Hwang et al. [84] suggested fusing thermal and visual sensors and using the KAIST dataset.

11) ApolloScape DATASET

A massive dataset collected in four areas of China using a combination of inertial measurement units (IMUs), global navigation satellite systems (GNSS), and video cameras is

known as the ApolloScape dataset [86]. Semantic segmentation in two and three dimensions, object detection, and self-localization are some of its intended uses. For 3D object detection and semantic segmentation, Huang et al. [87] suggested a multi-task learning method that trains and evaluates the ApolloScape dataset.

12) DRIVING STEREO DATASET

For autonomous driving in various environments, including cities, suburbs, highways, and rural roads, the Driving-stereo dataset [88] includes more than 180,000 stereo pictures. Data from color and stereo cameras and 3D LiDAR and GPS/IMU systems acquired in various weather situations make up the stereo matching a distance-aware and semantic-aware evaluation method for stereo matching. Using the Driving stereo dataset, Song et al. [89] suggested a distance-aware and semantic-aware evaluation method for stereo matching.

13) RobotCar DATASET

Recorded in central Oxford, UK, using numerous cameras, LiDAR, and GPS/GLONASS sensors, the Oxford RobotCar dataset [90] spans May 2014 to December 2015. This dataset is ideal for testing the resilience of perception algorithms in various settings. It contains over 20 million photos captured in various weather situations, including severe rain, darkness, bright sunshine, and snow. With the RobotCar dataset as their basis, Agarwal et al. [91] suggested a topology based on continuous appearances for persistent localization and mapping.

14) SYNTHIA DATASET

Among the many uses for the SYNTHIA dataset [93] are autonomous driving-related tasks, including semantic segmentation, object recognition, place identification, and change detection. There are thirteen different types of objects annotated semantically in the dataset, ranging from sky and buildings to roads and sidewalks and plants to lane markings and poles, cars and traffic signs, pedestrians and cyclists, and other random things. Ros et al. [92] suggested a domain adaption method that uses the SYNTHIA dataset as a simulated source domain and actual datasets as a target domain to improve semantic segmentation.

15) ONCE DATASET

For 3D object detection in autonomous driving scenarios, there is a large-scale dataset called the One Million sCenEs (ONCE) dataset [94]. It contains 7 million relevant camera photos and 1 million LiDAR scenes. The 144 hours of driving data used to create this dataset makes it far bigger than other 3D autonomous driving datasets, such as nuScenes and Waymo. To overcome the computational difficulties caused by the size of the ONCE dataset, Yu et al. [95] suggested a method for efficient 3D object detection.

To improve 3D object detection in autonomous driving, gathering and using extensive datasets is essential. The rich and varied data these datasets provide makes building and testing resilient algorithms in various environments possible. 3D object detection models can be continuously improved with the help of datasets such as the well-known KITTI dataset and the more recent ONCE dataset, offering distinct properties and scenarios. These datasets improve the performance and generalizability of autonomous driving systems by addressing real-world difficulties such as occlusions, truncations, and different climatic conditions.

Here, we offer some popular datasets for autonomous driving 3D object identification. In Table 3, you can see which 3D public datasets are comparable. This table compares a wide range of frequently used datasets in autonomous driving. These datasets have been crucial in propelling the discipline forward by supplying varied and difficult data for training and testing state-of-the-art algorithms.

C. PERFORMANCE METRICS FOR 3D OBJECT DETECTION

Evaluating the performance of 3D object detection systems is critical for advancing research and development in autonomous driving and other applications. Several essential metrics are commonly used to assess these systems' accuracy, robustness, and efficiency. Figure 14. Shows the essential metrics to evaluate the effectiveness of object detection methods.

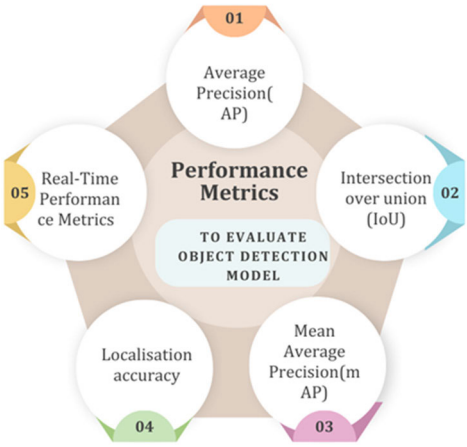


FIGURE 14. Essential metrics for 3D object detection dataset.

1) INTERSECTION OVER UNION (IoU)

Intersection over Union (IoU) is a fundamental metric for evaluating the spatial overlap between predicted and ground-truth 3D bounding boxes. It is defined as the volume of the intersection between the two boxes divided by the volume of their union. IoU values range from 0 to 1, with higher values indicating better overlap. A common threshold of 0.7 is often used to determine whether a predicted bounding box is true or false positive [98]. In 3D object detection, IoU is calculated by considering the volumetric intersection and union of the predicted and ground-truth 3D bounding boxes. Detections with IoU values above a predefined threshold (e.g., 0.5 or 0.7)

TABLE 3. Summarized features of different datasets.

Dataset	Year	Scenes	Classes	Annotated Frames	3D Boxes	Sensors	Variety	Recording Regions
KITTI [34]	2012	22	8	15k	200K	LiDAR & Camera	Daytime	Karlsruhe, Germany
Cityscapes [82]	2016	-	30	-	-	Camera	Spring, summer, fall, day, night, sunrise, different weather	Primarily Germany
RobotCar [90]	2017	-	-	-	-	Camera, LiDAR, & GPS	Heavy rain, night, direct sunlight, snow	UK
nuScenes [67]	2019	1k	23	40k	1.4M	LiDAR, Camera, & Radar	Daytime & nighttime	Boston, USA
H3D [72]	2019	160	8	27k	1.1M	Camera, LiDAR, & GPS/IMU	Daytime	San Francisco, USA
ApolloScape [86]	2019	-	35	140K	70K	Camera, LiDAR, & IMU/GNSS	Daytime & nighttime	Four different regions, China
Lyft Level 5 [78]	2019	366	9	46k	1.3M	Camera & Radar	Daytime	Palo Alto, USA
DrivingStereo [88]	2019	-	6	-	-	Camera, LiDAR, & IMU/GNSS	Sunny, rainy, cloudy, foggy, dusky	China
LIBRE [74]	2020	-	-	-	-	LiDAR	Fog, rain, strong light	Nagoya University, Japan
PandaSet [80]	2020	-	28	48k (images)	16k (sweeps)	Camera & LiDAR	Daytime & nighttime	San Francisco, USA
Waymo [76]	2020	1k	4	200k	12M	Camera & LiDAR	Daytime, nighttime, dawn, dusk	San Francisco, Phoenix, Mountain View, USA
ONCE [94]	2021	-	5	-	1M	Camera & LiDAR	Daytime, nighttime, rain	China

are considered true positives, while those below the threshold are false positives.

IoU is often incorporated into other performance metrics, such as Average Precision (AP), for classification and localization accuracy. While IoU is commonly used for evaluating 2D bounding box predictions, 3D-IoU specifically measures the overlap between predicted and ground-truth 3D bounding boxes. Considering The bounding boxes' height, width, and depth dimensions account for the volumetric intersection and union.

3D-IoU is particularly relevant for autonomous driving applications, where accurate 3D localization of objects is crucial for safe navigation and decision-making [99]. Figure 15. Demonstrates a comprehensive process for 3D object detection, transforming raw point cloud data into structured information that can be used to calculate Generalized Intersection over Union (GIoU) loss, which is a metric used to evaluate the accuracy of the predicted bounding boxes against the ground truths for tasks such as autonomous navigation.

2) AVERAGE PRECISION (AP)

Average Precision (AP) is a widely used metric that summarizes the precision-recall curve into a single value. It is calculated as the average of the precision values at different recall levels [100]. In 3D object detection, AP is computed based on the IoU threshold and the class of objects. This metric is commonly used in benchmarks like KITTI [34] and nuScenes [67]. The most common IoU thresholds are 0.7 for cars and 0.5 for pedestrians and cyclists [101]. AP values range from 0 to 100, with higher values indicating better performance.

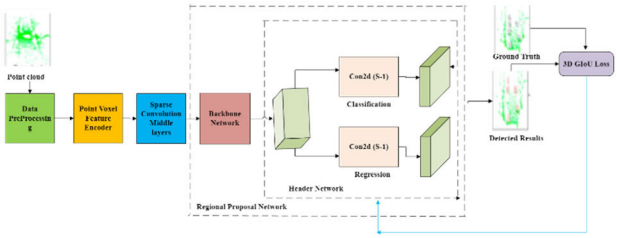


FIGURE 15. 3D Object detection system using point cloud data for 3D-IoU generalization [96].

3) MEAN AVERAGE PRECISION (mAP)

Mean Average Precision (mAP) is an extension of AP that averages the AP values across multiple classes. It is calculated by taking each class's mean of the AP values. It provides an overall measure of the detection performance across different object categories [102]. mAP is commonly reported for benchmarks that involve multiple object classes, such as the KITTI dataset. In the context of autonomous driving, mAP provides a comprehensive evaluation of the model's ability to detect and localize various types of objects, such as cars, pedestrians, cyclists, and other relevant classes.

4) LOCALIZATION ACCURACY

Localization accuracy measures the ability of a 3D object detection method to estimate objects' position and orientation

in 3D space accurately, typically evaluated using the average distance between the predicted and ground-truth 3D bounding box centers [103]. Some studies also report average orientation similarity (AOS), which calculates the cosine similarity between the predicted and ground-truth orientation vectors [104], offering insight into orientation accuracy. Average Localization Precision (ALP) assesses the precision of 3D bounding box localization, computed as the average-age 3D IoU of the localized bounding boxes relative to the ground truth [41].

5) F1 SCORE

The F1 score is an essential metric for evaluating 3D object detection models, providing a balanced measure of precision and recall. Research works such as PointRCNN, PV-RCNN [66], PointPillars, 3DSSD [105], and SECOND [106] have demonstrated the effectiveness of using the F1 score to assess their models. These studies show that achieving a high F1 score is critical for ensuring reliable and accurate 3D object detection in various applications, particularly in autonomous driving.

Additionally, Average Orientation Precision (AOP) measures the precision of predicted orientations by considering the average cosine similarity between the predicted and ground truth orientation vectors for localized bounding boxes. Average Translation Error (ATE) evaluates the average Euclidean distance between the predicted and ground-truth translations of object center positions [32]. At the same time, Average Scale Error (ASE) computes the average discrepancy in size between the predicted 3D bounding boxes and the ground truth. These metrics collectively provide a comprehensive evaluation of localization accuracy in 3D object detection. Complex-YOLO adapts the YOLO framework for 3D object detection by integrating RGB images and LiDAR point clouds. The model reduces localization error by jointly optimizing object detection and localization through a multi-task loss function. Complex-YOLO significantly improves localization accuracy, making it suitable for real-time 3D object detection [97].

6) TRUE POSITIVE RATE (TPR) AND FALSE POSITIVE RATE (FPR)

In 3D object detection, True Positive Rate (TPR) and False Positive Rate (FPR) are critical metrics for evaluating detection models. These metrics indicate how well a model identifies true positives and how often it falsely detects non-existent objects [107]. They are commonly used to plot Receiver Operating Characteristic (ROC) curves, which display TPR against FPR at various thresholds. This graphical representation aids in visualizing the model's performance across different thresholds and helps assess trade-offs between sensitivity and specificity, enabling optimization of detection models [111].

7) REAL-TIME PERFORMANCE METRICS

3D object detection for self-driving cars requires fast performance. This speed is measured by inference time (processing one data frame), FPS (frames processed per second), and latency (overall data-to-result delay). All three need to be minimal. Additionally, models need to be efficient with computer resources and ideally small for faster processing, even if it means a slight trade-off in accuracy. It is important to note that the choice of performance metrics and evaluation protocols may vary across different autonomous driving datasets and benchmarks, such as KITTI, nuScenes, and Waymo Open Dataset. Researchers and practitioners often report multiple metrics to comprehensively evaluate their 3D object detection models, accounting for both accuracy and real-time performance requirements. Table 4. summarizes common performance metrics used in 3D object detection, including the metric name, formula, and description.

These metrics provide a comprehensive evaluation of 3D object detection algorithms, each highlighting different aspects of performance, from accuracy to error measurement.

8) PERFORMANCE OF EXISTING METRICS

Existing research papers on 3D object detection often report a combination of the above metrics to comprehensively evaluate their proposed methods. For example, the seminal work by Zhou and Tuzel [109] on SECOND reports AP values of 81.97%, 65.46%, and 62.85% for cars, pedestrians, and cyclists, respectively, on the KITTI test set using an IoU threshold of 0.7 for cars and 0.5 for pedestrians and cyclists. Additionally, Shi et al. [35], in their PointRCNN method, achieved an AP of 88.88% for the "Car" class on the KITTI 3D object detection benchmark. Lang et al. [31] introduced PointPillars, which leverages the Waymo Open Dataset, reporting significant detection speed and accuracy improvements. These comprehensive evaluations include AP and mAP and metrics such as 3D IoU, ATE, and ASE to ensure that the models are robust in various real-world conditions and scenarios. Furthermore, recent advancements such as the Center Fusion method demonstrate the integration of radar and camera data, showcasing the evolution and increasing complexity of evaluation metrics used in 3D object detection research. This holistic approach ensures that both the localization accuracy and real-time performance metrics are thoroughly assessed, providing a clearer picture of the model's effectiveness in practical applications.

The SECOND method proposed by Yan et al. [106] achieves AP values of 83.34%, 72.55%, and 65.82% for cars, pedestrians, and cyclists, respectively, on the KITTI test set using the same IoU thresholds. Additionally, they report localization accuracy, with average errors of 0.076m, 0.091m, and 0.069m for the three classes. These metrics highlight the method's precision in estimating object positions and orientations.

Shi et al. [66], in their PV-RCNN paper, report mean Average Precision (mAP) values of 83.90%, 57.90%, and 70.47% for cars, pedestrians, and cyclists, respectively, on the KITTI test set. They also provide a detailed analysis of localization accuracy, with average errors of 0.062m, 0.088m, and 0.065m

TABLE 4. Performance metrics used in 3D object detection.

Metric	Formula	Description
Mean Average Precision (mAP)	$mAP = \sum_{i=1}^{N} APi$	The average of the average precision (AP) for each class. AP is the area under the precision-recall curve.
Intersection over Union (IoU)	$IoU=\frac{A \cap B}{AUB}$	Measures the overlap between the predicted and ground truth bounding boxes.
Precision	$Precision = \frac{TP}{TP+FP}$	The ratio of true positive (TP) detections to the total predicted positives (TP + false positives, FP).
Recall	$Recall = \frac{TP}{TP+FN}$	The ratio of true positive (TP) detections to the total actual positives (TP + false negatives, FN).
F1 Score	$F1 = 2\frac{precision.recall}{precision+recall}$	The harmonic mean of precision and recall, provides a balance between the two.
Average Precision (AP)	$AP=\sum_n (R_n - R_{n-1})P_n$	The area under the precision-recall curve, where PnP_nPn and RnR_nRn are precision and recall at the nth threshold.
Root Mean Square Error (RMSE)	$RMSE=\sqrt{\frac{1}{N}\sum_{i=1}^{N}(p_i - g_i)^2}$	Measures the average magnitude of the error between predicted values (p) and ground truth values (g).
Mean Absolute Error (MAE)	$MAE=\sqrt{\frac{1}{N}\sum_{i=1}^{N}(p_i - g_i)}$	MAE calculates the average absolute difference between predicted values (p_i)and actual values (g_i)
True Positive Rate (TPR)	$TPR = \frac{TP}{TP+FN}$	Also known as recall or sensitivity, it measures the proportion of actual positives that are correctly identified.
False Positive Rate (FPR)	$FPR=\frac{FP}{TN+FP}$	The ratio of false positives (FP) to the sum of false positives and true negatives (TN).
Mean Error (ME)	$ME=\sqrt{\frac{1}{N}\sum_{i=1}^{N}(p_i - g_i)}$	The average errors between predicted values (p_i) and ground truth values (g_i).

for the three classes. This detailed breakdown ensures that the models are evaluated on detection accuracy and their ability to localize objects precisely in 3D space.

Furthermore, Bastico [64] explores the integration of simultaneous depth completion and 3D object detection via deep learning for scene reconstruction in autonomous driving scenarios. This study underscores the importance of evaluating 3D object detection models under various conditions to ensure their robustness in practical applications.

These comprehensive evaluations include metrics such as 3D IoU, Average Translation Error (ATE), and Average Scale Error (ASE) to provide a holistic view of the models' performance. Such detailed assessments are essential for advancing 3D object detection technology, ensuring it meets the accuracy and real-time performance requirements crucial for applications like autonomous driving.

D. 3D OBJECT DETECTION PIPELINE

The 3D object detection pipeline involves several key stages that prepare, process, and analyze input data to produce accurate 3D representations and detections. This process is pivotal in applications such as autonomous driving, robotics, and augmented reality, where accurate spatial understanding is essential. The advent of deep learning has significantly advanced this field, enabling data-driven approaches to learn complex 3D features directly from raw sensor inputs.

The 3D object detection pipeline is a multi-step process that transforms raw sensor data into accurate 3D representations and detections, essential for applications such as autonomous driving, robotics, and augmented reality. The process begins with data acquisition, which is collecting information from sensors like LiDAR, cameras, and radar. LiDAR sensors provide detailed point cloud data, while cameras capture high-resolution image data. Following data acquisition, the raw sensor data undergoes preprocessing to remove noise, outliers, and irrelevant information. Techniques such as ground removal, voxelization, and downsampling are commonly used for LiDAR point clouds [64]. Zhou and Tuzel [109] proposed a voxel-based approach that significantly enhances object detection performance by efficiently managing and processing the point cloud data.

Next, the pipeline focuses on feature extraction, where relevant features are drawn from the preprocessed data to represent objects of interest. This stage can involve hand-crafted features like statistical properties or advanced deep-learning models. PointNet++ [108], for example, introduced a hierarchical feature learning network that captures both local and global features from point clouds, significantly improving the feature representation. The subsequent stage is region proposal generation, where regions of interest (ROIs) are identified to indicate potential object locations. Techniques such as PointNet [30], PointPillars [31], and region proposal networks (RPNs) [110] are utilized. VoxelNet employs a voxel-based approach for generating 3D proposals, improving object detection accuracy and efficiency in 3D space.

Once ROIs are generated, the pipeline moves to object classification and localization. In this stage, the proposed ROIs are classified into various object categories (e.g., cars, pedestrians, cyclists) and localized within the 3D environment. Various deep learning architectures, such as

PointRCNN [35], SECOND [106], and PointPainting [112], are utilized for this purpose. AVOD [113] combines image and LiDAR data to enhance the accuracy of 3D object detection. The final stage involves post-processing and tracking. Post-processing techniques like non-maximum suppression (NMS) or confidence thresholding are applied to remove duplicate or low-confidence detections. Furthermore, object tracking algorithms, such as AB3DMOT [115], associate detections across consecutive frames, enabling robust 3D multi-object tracking crucial for autonomous driving scenarios. This comprehensive pipeline ensures accurate detection and localization of objects in 3D space and integrates multiple sensor inputs and advanced deep-learning techniques to improve robustness and efficiency.

The below Figure16. Compares traditional and deep learning approaches for object detection, showcasing the differences in their pipelines from input to detection.

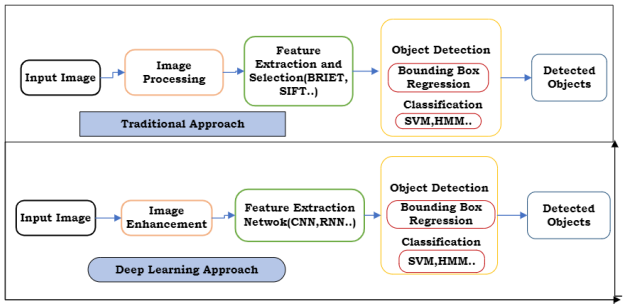


FIGURE 16. Traditional and deep learning approaches for object detection.

3D object detection relies on a comprehensive pipeline that transforms raw sensor data into meaningful 3D object information. By leveraging advanced preprocessing, feature extraction, and detection algorithms, it is possible to achieve highly accurate and reliable 3D object detection, essential for applications in autonomous driving, robotics, and augmented reality.

IV. 2D & 3D OBJECT DETECTION METHOD

Vehicles, pedestrians, and traffic signs are just a few examples of the many roadside objects that autonomous driving systems must be able to identify and locate. Because of their exceptional performance and capacity to acquire intricate features from data, object detectors based on deep learning have been popular in this field. The most effective way to extract usable information from images is through very deep convolutional neural networks (CNNs), which have been made possible by the rise in the computational capacity of contemporary GPUs [120]. The classification of object detection and its methods are shown in Figure 17.

A. 2D OBJECT DETECTION METHODS

Object detection and localization are two sides of the same coin in the object detection process. There are two main types of state-of-the-art models for this task: ones with two detection stages and ones with just one stage. Compared to one-stage detectors, two-stage detectors often achieve better accuracy but incur a greater computational cost.

1) TRADITIONAL DETECTORS

Traditional methods for object detection relied on hand-crafted features and classifiers to identify objects in images. These methods typically consist of two stages: feature extraction and classification. Feature extraction involves computing a set of image features to represent the objects of interest. Features used in traditional methods are encoded by feature descriptors like Scale Invariant Feature Transform (SIFT), Haar, Histogram of Gradients (HOG), or Speeded Up Robust Features (SURF). After feature extraction, a classifier determines whether the extracted features correspond to an object or background. Common classifiers used in traditional methods include Support Vector Machines (SVMs) and AdaBoost classifiers [24]. Traditional object detection methods have limitations, such as heavy reliance on hand-crafted features and a lack of end-to-end learning. These methods are also computationally expensive because they require separate feature extraction and classification steps for each object proposal in an image.

2) DEEP LEARNING APPROACHES TO 2D OBJECT DETECTION

Deep learning-based object detection, facilitated by applying Convolutional Neural Networks (CNNs) in image classification and parallel computing, has surpassed traditional methods. It learns high-level features directly from raw images, eliminating the need for hand-crafted features and effectively handling complex object appearance and variation. Moreover, deep learning-based methods can be trained end-to-end, reducing manual feature engineering and enabling easy adaptation to new domains through transfer learning. Deep learning-based methods mainly have two development routes: two-stage and one-stage detectors.

a: ONE-STAGE DETECTORS

These models accomplish both object localization and classification in a single network pass. You Only Look Once (YOLO) [121] and Single Shot MultiBox Detector (SSD) [122] are two examples. Compared to two-stage detectors, one-stage detectors are often faster and more suited for real-time applications; nevertheless, they may lose some accuracy in the process.

The development of one-stage detectors that do not use pre-defined anchor boxes has lately gained popularity as an alternative method. For example, FCOS attempts to streamline the object recognition process by predicting the four distances comprising the bounding box and utilizing the item's center point to identify positives. Due to its simplicity and adaptability, FCOS can surpass RetinaNet in performance without requiring the meticulous creation of anchor boxes for every situation. Alternative approaches,

such as ExtremeNet and CornerNet, find key points (such as the upper left and bottom right corners) before creating bounding boxes to identify objects. However, the keypoint-based approaches are far slower and necessitate more post-processing involvement.

However, these one-stage models couldn't be accurate due to the photos' severe foreground-background imbalance. By adjusting the SSD architecture's loss function, RetinaNet aimed to address this issue [123]. Similarly to FastPNN, RetinaNet suggests a new focused loss function that prioritizes challenging objects over easy samples. In addition, architectures like MobileNets were designed with specialized lightweight backbone networks that prioritize speed maximization [124].

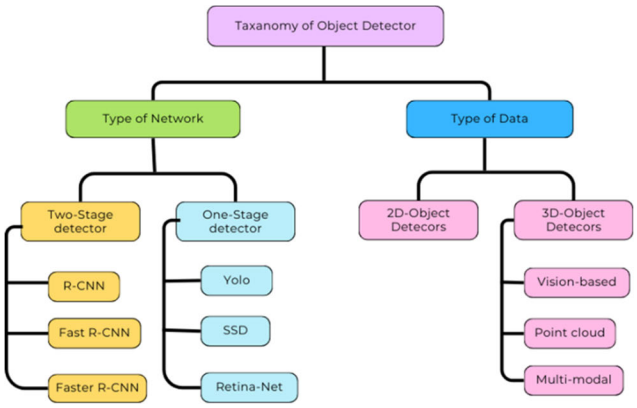


FIGURE 17. Taxonomy of object detection taxonomy of object detection.

Another anchor-based option is YOLO detectors, which partition the picture into sections and forecast probability and bounding boxes for each section. Despite being less precise, YOLO networks have demonstrated faster inference rates than RetinaNet. In contrast, YOLOv3 improved detection accuracy with the help of multi-scale forecasts and DarkNet-53, a superior backbone network that uses residual blocks and skip connections [125].

Ramos et al. [116] evaluated the performance of the YOLO model on the KITTI dataset for self-driving applications. Abdellatif Mtibaa et al. [117] improve object recognition in autonomous cars by utilizing YOLOv4 and Deep SORT for real-time multi-object detection. Jia et al. [118] enhanced the accuracy of YOLOv5 without compromising its speed. At the same time, Li and Zhao [119] created an efficient model based on YOLOv7 to improve detection performance and fulfill real-time safety needs. Various one-stage detection method studies are discussed in Table 5.

Anchor-free one-stage detectors have become more popular due to their independence from pre-defined anchor boxes. FCOS streamlines object detection by utilizing the centroid of objects to determine positive instances and predicting the four distances that constitute the bounding box. It surpasses RetinaNet in terms of simplicity and adaptability. Alternative approaches such as Corner-Net [127] and ExtremeNet [128] utilize keypoint detection, specifically identifying corners, to produce bounding boxes. However, these methods necessitate more intricate post-processing and exhibit slower performance.

The backbone network can function as either a Convolutional Neural Network (CNN) or a transformer-based network. The classification of a backbone network as either a one-stage or two-stage network depends on the design of its head network. Figure 18 (a) demonstrates that the one-stage detector conducts both object localization and classification simultaneously in the head network. In contrast, the two-stage detector conducts localization and classification on regions that have been obtained following the region proposals, as depicted in (b). Compared to ResNet backbones, DarkNet-53 is far more powerful than its predecessor, DarkNet-19.

b: TWO-STAGE DETECTORS

Two-stage frameworks partition the detection process into a stage for proposing regions and a stage for classifying those regions. At first, these models suggest multiple object possibilities, referred to as regions of interest (RoI), by utilizing reference boxes (anchors). In the second phase, the proposals are classified, and their exact locations are refined.

These models first propose a set of regions that might contain objects and then classify these regions. The most well-known example is the Faster R-CNN (Region-based Convolutional Neural Network). Two-stage detectors typically achieve higher accuracy as they refine object localization in the second stage but are usually slower and require more computational resources. Both types of detectors have their strengths and applications, depending on the specific.

Requirements of the autonomous driving system, such as the need for real-time performance versus the need for higher detection accuracy. The selection of the convolutional backbone network and the tuning of hyperparameters are critical steps in optimizing the performance of these models.

In the R-CNN [30] framework, a CNN is used for both classification and bounding box regression, with an external selective search method employed to generate proposals. Faster R-CNN improved efficiency significantly by sharing features between its two stages [45]. It uses a convolutional backbone network, such as ResNet [131] or VGG [132], to produce global feature maps. These maps are utilized by the Region Proposal Network (RPN) [130] and the detection network, reducing the cost of generating proposals from external sources.

Faster R-CNN has inspired many subsequent efforts to improve detection accuracy through various strategies, including developing more effective backbones to capture more detailed representations. For example, Feature Pyramid Networks (FPN) extract Region of Interest (RoI) features from various levels based on their scale, improving convolutional maps' multi-scale characteristics.

Subsequent studies have aimed to improve residual networks' internal connections to effectively utilize

TABLE 5. Related studies on one-stage detector methods.

S. No.	Name of Technique	Input Size	Highlights	Pros	Cons
1	SSD (Single Shot MultiBoxDetector) [123]	Image	Efficient, predicts bounding boxes and class probabilities in one pass	- Fast processing speed - Good for real-time applications	- Lower accuracy compared to two-stage detectors
2	YOLO (You Only Look Once) [122]	Image	Fast, single-stage framework	- Extremely fast processing speed - Simple architecture	- Lower accuracy compared to two-stage detectors
3	RetinaNet [123]	Image	Focuses on addressing limitations of SSD	- Improved accuracy over SSD - Good balance between speed and accuracy	- More complex architecture compared to SSD/YOLO
4	DSSD (Dense Single Shot Detector) [129]	Image	Combines strengths of SSD and YOLOv3	- Improved accuracy over SSD - Maintains fast processing speed	- Slightly more complex architecture compared to SSD
5	FCOS (Fully Convolutional One-Stage Object Detection) [126]	Image	Anchor-box free approach	- Achieves high accuracy - Efficient for resource-constrained devices	- May struggle with small objects

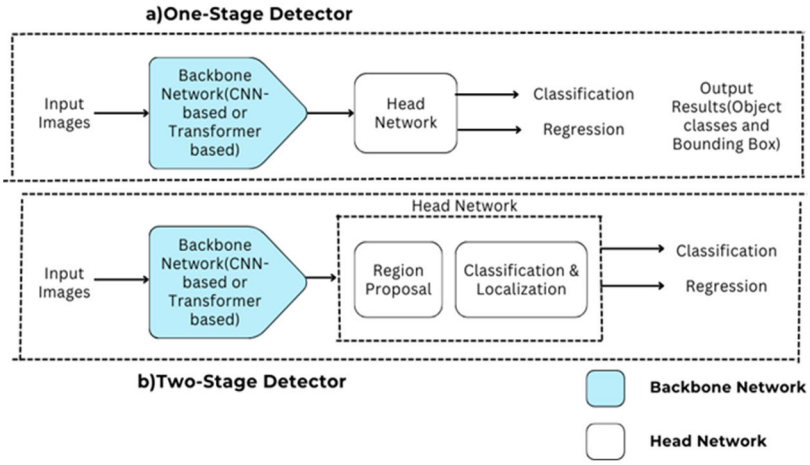


FIGURE 18. Basic deep learning-based one-stage vs. two-stage object detection model architectures.

convolutional maps' multi-scale characteristics. Examples of such studies include ResNeXt, which employs grouped convolutions [133], and Res2Net [134].

Additional research has improved detection accuracy by suggesting alterations to the current Faster R-CNN framework. For example, Cascade R-CNN proposes using multiple detectors trained with progressively higher intersection-over-union (IoU) thresholds [130]. This architectural design can enhance the accuracy of detections, but it also results in a higher computational load, which makes it less suitable for real-time applications.

Recent enhancements in two-stage detectors focus on architectural improvements and novel methods to boost detection accuracy and speed. DetectoRS introduces a recursive feature pyramid and switchable atrous convolutions to enhance feature extraction. Libra R-CNN integrates balanced feature pyramids and IoU-balanced sampling to address class imbalance and improve detection accuracy. Dynamic R-CNN adjusts the IoU threshold dynamically during training, improving the robustness and accuracy of the model.

Sonam Rinchen et al. also propose a scalable Multi-task Learning (MTL) system for object detection in autonomous vehicles. This system leverages computer vision techniques and extends Mask R-CNN [136] to manage multi-label scenarios. Evaluations on the BDD100K dataset show superior performance over Mask R-CNN in mean average precision at 50 (mAP50) [137].

Both types of detectors have their strengths and limitations, depending on the specific requirements of the autonomous driving system, such as the need for real-time performance versus higher detection accuracy. This structured approach to object detection ensures that autonomous driving systems can accurately and efficiently identify and localize objects, contributing to safer and more reliable vehicle operation. Prominent two-stage detection methods are compared in Table 6.

B. 3D OBJECT DETECTION METHODS

2D object detectors provide bounding boxes with four degrees of freedom (DOF): [x, y, height, width] or [x_{min},

TABLE 6. Two-stage detector method comparisons.

S.No.	Name of Technique	Input Size	Highlights	Pros	Cons
1	R-CNN(Regions with CNN features) [30]	Image	Pioneering two-stage detector	- High accuracy - Can be adapted to various tasks	- Relatively slow processing speed
2	Fast R-CNN [135]	Image	Faster version of R-CNN	- Improved processing speed over R-CNN - Maintains good accuracy	- Still slower than one-stage detectors
3	Faster R-CNN [45]	Image	Region Proposal Network (RPN) for efficient proposal generation	- Significantly faster processing speed compared to Fast R-CNN - High accuracy	- More complex architecture compared to R-CNN/Fast R-CNN
4	Mask R-CNN [136]	Image	Performs instance segmentation in addition to object detection	- Enables pixel-level segmentation of objects - Maintains good detection accuracy	- Slightly slower than Faster R-CNN for detection only
5	CascadeR-CNN [130]	Image	Multi-stage object detection	Improves accuracy with multi-stage refinement	Increased computational cost, more complex architecture

y_{min}, x_{max}, y_{max}]. They only determine the position of an object on a 2D plane without depth information, which is essential for tasks like path planning and collision avoidance. In contrast, 3D object detectors use cameras, lidar, or radar data to generate 3D bounding boxes with (x, y, z) coordinates, (height, width, length) dimensions, and yaw information.

These detectors employ techniques like point clouds and frustum point nets for real-time object prediction [16]. 3D object detection is crucial in various applications, including guiding autonomous vehicles, robotics, and augmented reality. It involves identifying and locating objects within a 3D environment. Here, we explore three prominent categories of 3D object detection methods.

1) VISION-BASED METHODS

Vision-based techniques for 3D object detection play a crucial role in enabling safe and reliable autonomous driving systems. These techniques leverage camera sensors and computer vision algorithms to perceive the surrounding environment, identify objects of interest, and estimate their 3D positions and orientations. This section reviews various vision-based approaches for 3D object detection, focusing on their applications in autonomous driving scenarios. We highlight the strengths and limitations of these methods and their contributions to improving the safety and efficiency of autonomous vehicles.

a: MONOCULAR 3D OBJECT DETECTION

Monocular 3D object detection aims to infer the 3D positions and shapes of objects from single RGB images. This task is particularly challenging due to the lack of explicit in-depth information and the inherent ambiguities in perspective projection. Various approaches have been developed to address these challenges, leveraging deep learning techniques and geometric reasoning.

MonoFusion [37] addresses these issues by leveraging the geometric relationship between 2D and 3D bounding boxes, combining the outputs of a 2D object detector and a depth estimation network to predict 3D bounding boxes, demonstrating impressive results on the KITTI dataset. Another notable method is MonoPSR [38], which uses perspective key points and their associated depth values to recover 3D bounding boxes without explicit depth estimation, achieving state-of-the-art performance on the KITTI dataset. RTM3D [139] is a real-time monocular 3D detection framework focusing on speed and efficiency while maintaining high detection accuracy, leveraging a fast inference pipeline suitable for real-time applications. Despite advancements, a significant performance gap remains between monocular and LiDAR-based 3D object detection due to monocular imagery's limitations, such as the lack of depth information and difficulties in handling occlusions and complex scenes.

Monocular 3D object detection has led to the development of several innovative methods that significantly enhance detection accuracy and robustness. Figure 19. Depicts a timeline of significant methods and advancements in monocular 3D object detection for autonomous driving from 2017 to 2023. Each method is associated with its authors and highlights vital contributions. Here's a brief overview of the processes depicted in the following section. Chen et al. introduced Deep3DBox [140], a method that uses deep learning to predict 3D bounding boxes from a single RGB image by leveraging geometric constraints to improve accuracy. Following this, Chen et al. developed Mono3D [141], which focuses on detecting 3D objects using monocular images by generating 3D object proposals and optimizing them with a deep neural network, combining 2D and 3D information to enhance detection performance. Li et al. proposed DeepMANTA [142], a multi-task network that integrates vehicle detection, part localization, and visibility prediction

TABLE 7. Monocular based methods.

Method	Methodology	Advantages	Limitations	Future Research Directions
Direct Regression	Directly regresses 3D bounding boxes from the image. (e.g., MonoPS [38])	Simple and efficient.	Prone to errors due to lack of depth information. Limited accuracy.	Explore incorporating geometric constraints or leveraging attention mechanisms for improved localization.
Depth Prediction with Detection	Predicts a depth map from the image and then performs 3D detection on the predicted depth and image features. (e.g., DispNet [152])	Handles occlusions better than direct regression.	Depth prediction accuracy can significantly impact overall performance.	Develop more robust depth estimation methods that are less sensitive to lighting variations.
Pseudo-LiDAR	Generates a pseudo-LiDAR point cloud from the image and performs 3D detection on the point cloud. (e.g., PointPillars [31])	Leverages established point cloud detection methods. Achieves higher accuracy compared to simpler approaches.	Requires complex pipelines for point cloud generation. High computational cost for real-time applications.	Explore lightweight point cloud generation methods and efficient 3D detection algorithms for real-time scenarios.
Learning-based Shape Priors	Leverages a shape prior (e.g., 3D object models) to guide the detection process. (e.g., FCOS3D [153])	Improves performance for objects with known shapes.	Requires a large database of 3D object models. Limited generalizability to unseen object categories.	Develop methods that can learn shape priors dynamically from data or adapt to new object categories efficiently.

to improve detection accuracy and robustness under varying conditions. Brazil and Liu presented M3D-RPN [143], a region proposal network designed explicitly for monocular 3D object detection that generates high-quality 3D region proposals, significantly improving detection accuracy. Dubey et al. introduced MonoGRNet [144], which combines geometric reasoning with deep learning to predict 3D bounding boxes and refine them using a geometric reasoning module for better accuracy in complex scenes.

Liu et al. developed SMOKE [145], a single-stage monocular 3D object detection framework designed for real-time applications, which directly regresses 3D bounding boxes and object classes from a single image with high efficiency and competitive accuracy. Chen et al. introduced MonoPair [146], a method that handles occlusions by pairing objects based on their spatial relationships, enhancing robustness in cluttered scenes. Li et al. presented RTM3D [139], a real-time monocular 3D detection framework emphasizing speed and efficiency while maintaining high detection accuracy through a fast inference pipeline. Liu et al. introduced MonoFlex [147], which incorporates flexible 3D representations to improve the adaptability and accuracy of monocular 3D object detection, effectively handling various object shapes and sizes. Hu et al. developed Kinematic3D [148], leveraging temporal information from consecutive monocular images to refine 3D bounding boxes using kinematic constraints, achieving higher precision. Zhou et al. introduced MonoCon [149], which incorporates contextual information from the surrounding environment to enhance detection accuracy and robustness in complex scenes through a context-aware network. Finally, Meng et al. presented MonoDLE [150], a depth-aware method that leverages depth estimation to refine 3D bounding boxes, integrating depth cues directly into the detection pipeline to improve accuracy significantly. Ranasinghe et al. present an innovative approach to 3D object detection and pose estimation using monocular images, addressing challenges like the lack of depth information and the need for precise object localization and orientation estimation.

Recent advancements have improved camera-based 3D object detection for autonomous vehicles. Zhang et al. [154] developed a multi-camera system using Vision Transformers (ViTs) to handle missing depth information better than traditional CNNs. ViTs use attention mechanisms to process spatial relationships and depth cues from multiple cameras, making the system more robust. Song et al. [155] introduced MDS Net. This one-stage monocular 3D detection network uses a unique depth-based structure and a new angle loss function to enhance depth and angle prediction accuracy. This method performs better than existing ones on the KITTI dataset and works well in real-time, making it valuable for autonomous vehicles. Qu et al. [156] proposed MonoDCN, which combines dynamic convolution with depth maps to improve monocular 3D detection in Table 7. Monocular-based methods are tabulated with their methodology, advantages, limitations, and future directions.

This approach learns from depth and semantic information, greatly enhancing the accuracy and efficiency of the KITTI dataset. MonoDCN shows the benefits of Combining depth and semantic cues for better 3D detection in autonomous vehicles.

b: STEREO CAMERA BASED

Stereo vision is a well-established technique in computer vision that mimics human binocular perception to estimate depth from two camera viewpoints. By capturing a scene from slightly different positions, stereo cameras provide two perspectives that can be used to triangulate the 3D positions of points in the scene [157]. The key idea behind stereo vision is to find corresponding points in the left and right camera images and measure their disparity, the horizontal shift between the matching points. This disparity is inversely

TABLE 8. Comparison of vision-based techniques for 3D object detection in autonomous driving.

Feature	Monocular Camera [36]	Stereo Camera [39]	Multi-View Fusion [261]
Sensor Setup	Single camera	Two cameras	Multiple cameras
Depth Information	No direct depth	A disparity map provides depth cues	Requires additional processing
Strengths	Low-cost, widely available	Improved depth estimation compared to monocular	Potentially richer information for object recognition
Weaknesses	Inaccurate 3D localization, sensitive to lighting	Limited depth range, challenges with repetitive textures	Increased complexity requires synchronization
Common Approaches	- Geometric constraints (camera projection, object size priors) - Learning from paired image-depth data - Advanced deep learning architectures (3D CNNs, Transformers)	- Exploiting disparity maps for depth estimation - 3D reconstruction techniques	- Early fusion (combining raw sensor data) - Late fusion (combining extracted features) - Deep learning-based fusion
Accuracy	Lower compared to stereo and multi-view	Better than monocular, but limitations remain	Potentially highest, but it depends on the fusion strategy
Computational Cost	Lower	Moderate	Higher
Suitability	Cost-effective solution for less demanding applications	The balance between accuracy and cost	High-performance applications requiring robust object detection

proportional to the point's depth, allowing the system to estimate the 3D structure of the scene [158]. Compared to monocular methods, stereo-based 3D object detection offers several advantages. The explicit depth information provided by stereo vision helps to resolve the scale ambiguity and occlusion issues common in monocular imagery. By incorporating depth cues, stereo-based methods can more accurately estimate 3D positions and sizes of objects, leading to improved detection accuracy [159]. Table 13 presents a detailed overview of various vision-based methods utilized in 3D object detection.

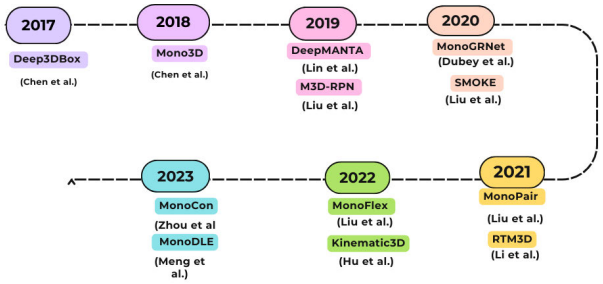


FIGURE 19. Timeline diagram of significant methods and advancements in monocular 3D object detection for autonomous driving.

3DOP [160] proposed a dense 3D mapping approach that combines stereo depth estimation with region-based object detection for accurate 3D localization of vehicles. Stereo Block Matching [161] introduced a stereo image-based 3D object detection system that leverages geometric constraints and depth information for precise 3D bounding box estimation. Stereo R-CNN [41] integrated stereo geometry into a region-based convolutional neural network (R-CNN) framework for accurate 3D object detection and localization.

Stereo R-CNN integrates stereo geometry into the R-CNN framework for enhanced 3D object detection accuracy and localization. Pseudo-LiDAR [162], introduced by Wang et al., converts stereo depth maps into 3D point clouds, mimicking LiDAR data, and applies point cloud-based detection algorithms. Pseudo-LiDAR++ [43] proposed an end-to-end deep learning approach that converts stereo depth maps into 3D pseudo-point clouds and applies PointNet++ for 3D object detection. DSGN [163], introduced by Chen et al., proposes a deep learning architecture that combines stereo depth estimation with 3D object detection. Pseudo-Mono [164] addresses inaccurate depth estimation using stereo images as input in monocular 3D object detection. A lightweight depth predictor generates depth maps, and the framework uses left input images from the stereo camera to create enhanced visual and multi-scale depth features through depth indexing and feature matching probabilities.

While stereo camera-based methods have shown promising results, they can be affected by challenging scenarios like low-texture regions, occlusions, and depth discontinuities, leading to incorrect depth estimations and errors in 3D object detection.

c: MULTI-VIEW FUSION

Multi-view fusion methods leverage multiple RGB images captured from different viewpoints to reason about the 3D scene jointly. By combining information from various perspectives, these methods aim to overcome the limitations of monocular or stereo approaches and provide more accurate 3D localization and detection.

Li et al. [41] introduced MVX-Net, which enhances 3D detection by projecting 3D proposals onto multiple views and fusing the resulting features. Liang et al. [4] developed M3D-RPN, a model that fuses multi-view features through a 3D

region proposal network. Vora et al. [112] presented PointPainting, a method that sequentially combines point cloud and multi-view RGB features. Recently, Wang et al. [165] proposed a transformer-based multi-view fusion approach using attention mechanisms to integrate information from various viewpoints. Similarly, Zhu et al. [166] introduced a dynamic multi-view fusion framework that adjusts feature weighting and fusion to improve robustness against occlusions and viewpoint variations.

2) POINT CLOUD-BASED

Point cloud-based methods have emerged as a promising approach for 3D object detection, particularly in autonomous driving and robotics. Point clouds are three-dimensional representations of the environment, created by capturing the spatial coordinates of objects using sensors such as LiDAR (Light Detection and Ranging) or depth cameras. These methods aim to accurately detect and localize 3D objects from point cloud data, which is essential for obstacle avoidance, path planning, and scene understanding.

Here are some commonly used point cloud-based methods for 3D object detection.

3) PointNet AND ITS VARIANTS

PointNet [30] and its successor PointNet++ [108] introduced a groundbreaking approach to process unordered point clouds using deep learning architectures directly. These techniques have been widely adopted and extended in numerous studies for 3D object detection in autonomous driving scenarios. VoxelNet [109], proposed by Zhou and Tuzel, utilizes PointNet-based networks to learn features from voxelized point cloud data, enabling efficient 3D object detection. This method has been extensively applied and evaluated in various benchmarks and datasets, such as the KITTI and NuScenes datasets. Qi et al. introduced PointNet [30], a pioneering architecture for directly processing raw point clouds using deep learning. This method effectively handles unordered point sets, setting a foundation for future point cloud-based methods. Zhou and Tuzel proposed VoxelNet [109], which converts point clouds into voxels and then applies 3D convolutional neural networks (CNNs) to learn features, significantly improving 3D object detection performance. In the same year, Qi et al. extended their earlier work with PointNet++ [108], incorporating hierarchical feature learning to more effectively capture local structures in point clouds. Yan et al. developed SECOND [106], which introduced sparse 3D convolutions further to enhance the efficiency of processing voxelized point clouds. Isele et al. introduced PointPillars [5], which encodes point clouds into a pseudo-image for efficient processing using 2D CNNs. Figure 20. Presents a timeline diagram of all the methods implemented from 2018 to 2024.

Shi et al. presented PV-RCNN [66], combining point-based and voxel-based networks to achieve high detection accuracy. They also introduced Point-GNN [167], leveraging

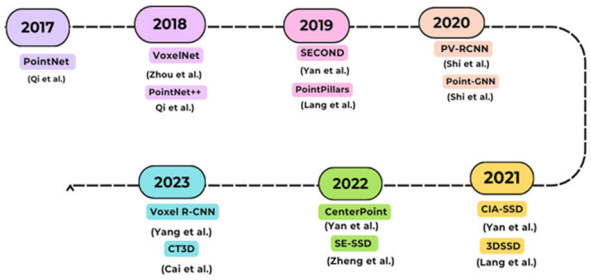


FIGURE 20. Timeline diagram of point cloud-based architectures.

graph neural networks to capture the relationships between points for improved detection performance. Yan et al. introduced CIA-SSD [168], a single-stage detector that balances speed and accuracy. Lang et al. presented 3DSSD [105], focusing on efficient and accurate single-stage 3D object detection. Yan et al. developed CenterPoint [169], which predicts the centers of objects and regresses to their 3D bounding boxes. Zheng et al. proposed SE-SSD [170], which enhances single-stage detectors with self-ensembling techniques to improve detection robustness. Yang et al. introduced Voxel R-CNN [171], combining the benefits of voxel-based and region-based convolutional networks for higher accuracy. Cai et al. presented CT3D [172], a novel method that leverages contextual information for improved 3D object detection.

Frustum PointNets [16], introduced by Qi et al., combines 2D object detection from RGB images with 3D point cloud segmentation using PointNet-based networks. This fusion-based approach has been widely adopted and extended in 'subsequent research works for autonomous driving applications.

LiDAR sensors excel in 3D object detection for autonomous vehicles, but GNN-based methods like Point-GNN are slow for real-time applications. Trilaksono et al. [173] propose optimizations for Point-GNN, achieving up to 2.83x faster inference with minimal accuracy loss. They also explore incorporating new features, leading to a further boost in detection performance. This research paves the way for the practical use of GNNs in real-time autonomous driving. Tong et al. [174] propose MDRNet, a new point cloud backbone network for 3D object detection. It addresses information loss in existing methods using Spatial-aware Dimensionality Reduction (SDR) and Multi-level Spatial Residuals (MSR). MDRNet integrates with grid-based detectors and significantly improves benchmarks like nuScenes, KITTI, and DAIR-V. This method is valuable for improving accuracy in BEV-based 3D object detection tasks. Ibrahim et al. [175] propose a real-time 3D object detection model for roadside LiDAR systems in autonomous vehicles. This method improves upon existing detectors and achieves high accuracy (45 Hz inference speed) on various datasets. It also demonstrates successful transfer learning between different countries' datasets. The LiDAR-based detector improves perception for connected and automated vehicles, enhancing

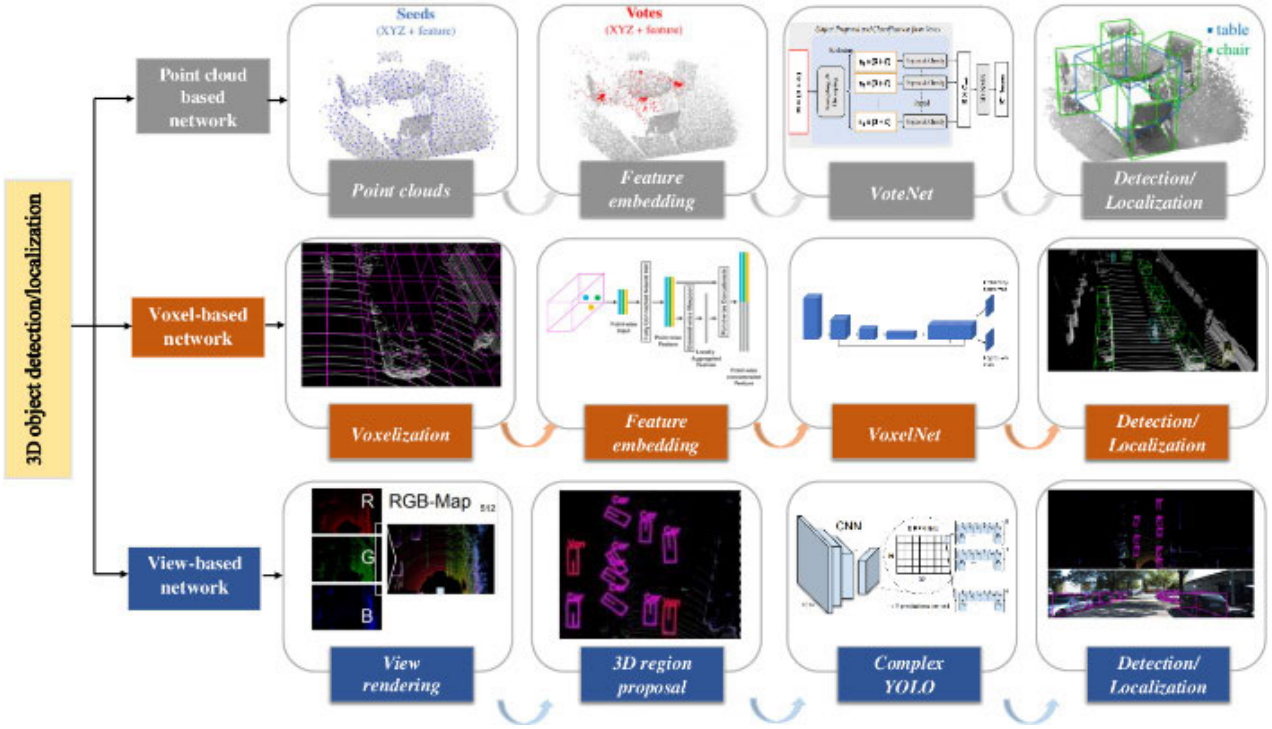


FIGURE 21. Illustrations of network architectures for 3D object detection and localization [176].

safety by providing information on surrounding vehicles, even those hidden from view. Figure 21. Highlights the diversity of approaches and network architectures for 3D object detection, ranging from direct point cloud processing to voxelized representations and view-based methods that leverage 2D image processing techniques.

C. VOXEL-BASED METHODS

Voxel-based methods, which convert point cloud data into a 3D voxel grid representation, have been widely utilized for 3D object detection in autonomous driving scenarios. This approach leverages the strengths of 3D CNNs while preserving the spatial information in the point cloud. Dániel Kozma et al. [177] introduced a novel approach to training neural networks for 3D object detection in autonomous driving. They tackle the challenge of acquiring diverse and sufficient annotated data by combining semi-pseudo-labeling with innovative 3D augmentations. This method enriches existing datasets with pseudo-labels derived from semi-supervised learning techniques and introduces novel 3D augmentations to enhance dataset diversity and quantity. Their specialized convolutional neural network for 3D object detection demonstrates improved performance in practical settings, contributing to the advancement of safer and more effective transportation systems. Yang et al. [178] address the challenges of disorder and sparsity in LiDAR point cloud data for 3D object detection. They propose an approach based on PV-RCNN that improves accuracy, especially for small objects, by incorporating multiple outputs from the voxel CNN and introducing a Multi-Scale Region Proposal Network. Evaluations on the KITTI dataset show a 5% improvement in detecting small objects, demonstrating the method's robustness for autonomous vehicles.

Li et al. [179] propose TED, an efficient 3D object detection method for autonomous vehicles. Traditional methods struggle with object rotations and require high computational power. TED addresses this using a novel architecture with a sparse convolution backbone and efficient feature processing. This approach achieves state-of-the-art performance on the KITTI benchmark, making it a valuable tool for real-time object detection in autonomous driving.

Zhang et al. [180] propose PLOT, a lightweight 3D object detection network for autonomous vehicles. It tackles the computation time and power consumption limitations in embedded systems for roadside and vehicle-side object detection. PLOT utilizes a pillar-based point cloud representation and a cross-stage partial network backbone to achieve real-time performance while maintaining accuracy. This method was validated on the Waymo Open Dataset, demonstrating its promise for lightweight and real-time 3D object detection in autonomous driving. Table 9 compares basic point-based techniques for 3D object detection.

D. MULTI-SENSOR FUSION BASED

Multi-sensor fusion is a rapidly evolving field, and new developments and techniques are constantly emerging. As autonomous vehicles continue to advance, the fusion of

TABLE 9. Comparison of point-based techniques for 3D object detection in autonomous driving.

S.No.	Name of the Technique	Input Size	Highlights	Pros
1	PointNet [30]	Point Cloud	Learns features directly from point clouds	- Flexible for various point cloud tasks - Can handle irregular point cloud shapes
2	PointNet++ [108]	Point Cloud	Hierarchical architecture for feature extraction	- Captures local and global features of point clouds - Improves upon PointNet for complex scenes
3	VoxelNet [109]	Point Cloud (converted to Voxel Grid)	Efficient processing for large point clouds	- Handles large datasets efficiently - Good accuracy for 3D object detection
4	PointPillars [31]	Point Cloud	Efficient 3D detection using point cloud pillars	- Handles large point clouds efficiently - Good accuracy for 3D object detection

data from various sensors will play a crucial role in enabling safe and reliable operation in complex environments.

1) CAMERA-LiDAR FUSION

Researchers have explored fusing camera and LiDAR data for object detection, semantic segmentation, and depth estimation tasks. Techniques like projecting LiDAR point clouds onto camera images or using camera images to enhance LiDAR point cloud features have been investigated. LiDAR offers accurate and comprehensive data that is essential for understanding and perceiving three-dimensional space. In their publication, Fan et al. [237] provide the Snow-CLOCs procedure. It offers a fresh method for merging detection outcomes from camera and LiDAR sensors into a cohesive detection set using a sparse tensor representation. Extracting features using a 2D convolutional neural network improves object recognition and localization. The algorithm has an 86.61% detection accuracy for identifying vehicles in snowy situations, highlighting its superior performance compared to conventional approaches.

Li et al. [181] introduced a novel one-stage convolutional neural network (CNN) designed to perform 3D object detection by fusing multi-sensor data.

Integrating multi-sensor data addresses the limitations inherent in using a single sensor type, such as missing depth information in cameras or the sparse nature of LiDAR data. Figure 22. Shows the architecture of a complex retina network that depicts a multi-modal 3D object detection architecture that combines point cloud data from LiDAR and image data from cameras.

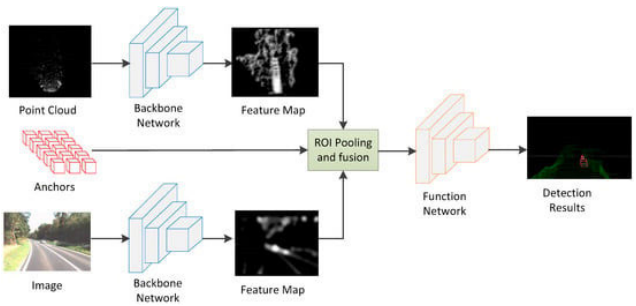


FIGURE 22. The architecture of our proposed Complex-Retina network.

Li et al. [41] proposed a multi-view 3D object detection network that fuses camera and LiDAR data. Meanwhile, Liang et al. [4] developed a multi-task, multi-sensor fusion approach for 3D object detection. Ku et al. [110] introduced a method for joint 3D proposal generation and object detection from view aggregation using camera and LiDAR data. Frustum Point-Nets, introduced by Qi et al. [16], bridges the gap between 2D image data and 3D point clouds.

Building on complementary sensor strengths, Alaba and Ball [182] propose a fusion model that combines camera imagery (rich detail) with LiDAR (accurate depth) for 3D object detection. This approach tackles the limitations of each sensor and achieves competitive results on the Nuscenes dataset. Chen et al. [183] address challenges in crowded scenes by incorporating relationships between objects. Their method fuses camera data with LiDAR, building a bird's-eye view map that captures object connections. This approach significantly improves detection accuracy in dense environments compared to existing methods.

Researchers are continuously improving multi-sensor fusion techniques for 3D object detection in autonomous vehicles. Liu et al. [184] proposed PMPF, a framework that enriches camera data with contextual information from LiDAR, achieving significant performance gains. Zhao et al. [185] introduced a two-stage network that leverages pre-trained models and augmented data to enhance robustness, particularly in pedestrian detection. Hou et al. [186] developed a DNN-based method with a unique fusion scheme for LiDAR and camera data, demonstrating superior performance in detecting distant and occluded objects. Adekallu et al. [187] proposed OD-C3DL, a system that combines camera and LiDAR data for real-time object detection with high accuracy, making it suitable for practical applications in autonomous vehicles. These advancements highlight ongoing progress in multi-sensor fusion for robust and accurate 3D object detection, a critical component for autonomous vehicle perception systems.

Bhardwaj et al. [188] propose a method to improve object detection in autonomous vehicles. LiDAR and cameras have limitations: LiDAR struggles with distant objects, and cameras struggle in low-visibility conditions. Their approach combines upsampled LiDAR data with camera data to create

higher-quality point clouds, achieving better object detection than LiDAR alone on the KITTI benchmark. This method offers a promising way to overcome the limitations of individual sensors for more robust object detection in autonomous vehicles.

Li et al. [189] propose MVMM, a novel 3D object detection method for autonomous vehicles that combines camera images and LiDAR point clouds. Unlike traditional methods using only point clouds, MVMM extracts color and texture information from images to improve detection accuracy. This method effectively fuses sensor data and achieves competitive performance on the KITTI dataset, even in challenging scenarios. MVMM highlights the potential of multi-modal approaches for robust 3D object detection in autonomous driving

Dai et al. [190] propose SRDL, a 3D object detection method for autonomous vehicles that overcomes the limitations of past approaches. Existing methods struggle to utilize spatial and semantic information from LiDAR and camera data. SRDL addresses this by combining stereo image information with LiDAR data through techniques like Deep Feature Fusion. This method significantly improves the KITTI benchmark, demonstrating its effectiveness for enhanced perception in autonomous driving systems.

2) RADAR-CAMERA FUSION

Combining radar and camera data has been explored for object detection and tracking tasks. Radar provides robust distance and velocity information, while cameras provide visual details. Fusion methods include using radar data to guide attention in cameras or using camera data to enhance radar detections.

3) CAMERA, LiDAR, AND RADAR FUSION

To further enhance the robustness and reliability of 3D object detection systems, researchers have explored the fusion of multiple sensor modalities, including cameras, LiDAR, and radar. Figure 23. Shows that the multi-sensor fusion approach aims to leverage the strengths of each modality, such as the rich semantic information from cameras, accurate depth perception from LiDAR, and robust long-range detection capabilities of radar.

The PointPaintingNet [191], proposed by Vora et al., is a notable work in this category. It presents a sequential fusion approach that combines camera images, LiDAR point clouds, and radar data for 3D object detection.

The authors demonstrated improved performance compared to single-sensor and dual-sensor fusion methods on the NuScenes dataset.

Another interesting work is the Camera-LiDAR-Radar Object Detection (CLRNet) [192], introduced by Li et al. This method proposes a multi-modal fusion network integrating features from cameras, LiDAR, and radar data. The authors evaluated their approach on the NuScenes dataset

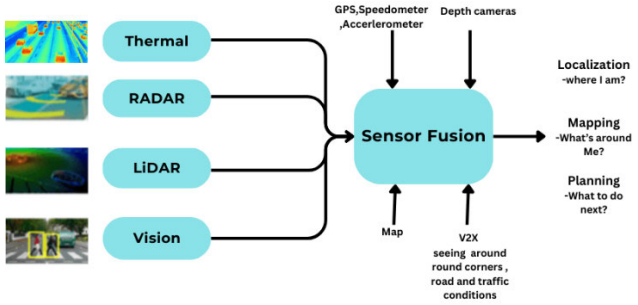


FIGURE 23. Multi-sensor fusion approaches.

and showed improved performance compared to other fusion-based methods.

With the increasing complexity of autonomous driving scenarios, researchers have recognized the limitations of relying on a single sensor modality for robust 3D object detection. This has led to exploring multi-sensor fusion techniques that leverage the complementary strengths of different sensor modalities, such as cameras, LiDAR, and radar.

Deep learning architectures, such as specialized fusion layers, attention mechanisms, and end-to-end trainable networks, have been developed specifically for multi-sensor fusion. Liang et al. and Vora et al. [193] proposed deep fusion networks for multi-sensor fusion tasks in autonomous vehicles. Zhao et al. [194] present a multi-sensor 3D detection method for small objects that integrates LiDAR and camera data using advanced fusion modules to improve detection accuracy, achieving notable results on the KITTI dataset.

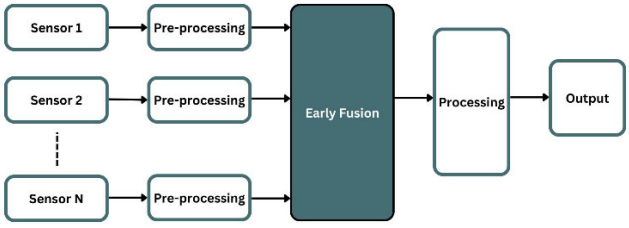


FIGURE 24. Early fusion architecture: raw data from multiple sensors are pre-processed, calibrated, and registered before fusing into a unified dataset. These fused data are then processed for object detection, classification, and tracking.

MV3D [199] is a multi-view 3D object detection network that integrates LiDAR, camera, and radar sensor inputs to generate 3D bounding boxes. The approach projects 3D proposals into a bird's-eye view and front view, enhancing object detection accuracy by leveraging complementary data from different sensors. AVOD (Aggregate View Object Detection) [110] combines features from both LiDAR point clouds and RGB images. It proposes an efficient fusion architecture that aggregates features at multiple scales, significantly improving detection performance compared to single-modality detectors. PointFusion [4] uses a two-stage approach that generates 2D object proposals from images and then fuses these with corresponding 3D point cloud

TABLE 10. Comparison of multi-sensor fusion-based approaches for real-time 3D object detection.

Fusion Approach	Features	Benefits	Drawbacks
Early Fusion[85]	Combines unprocessed data from various sensors before performing further analysis.	Utilizes comprehensive data to capture all possible interactions, hence improving feature extraction	The task involves a significant amount of processing resources and necessitates the coordination of sensor data.
Feature-Level Fusion[85]	It refers to combining or integrating features from several sources or modalities to create a unified representation. Integrates characteristics derived from sensor data following initial processing.	Minimizes computational burden, harnesses resilient characteristics from each sensor	The complexity of feature compatibility and extraction design.
Decision-Level Fusion[85]	Combining many decisions or outputs from different sources or classifiers to make a final decision or output. Consolidates conclusive determinations from the individual study of each sensor.	Requires less computational resources, flexible in decision-making	There is a risk of losing specific information when using raw and feature-level data, which may reduce accuracy if particular decisions are not vital.

TABLE 11. Comparison of multi-sensor fusion-based techniques for 3D object detection.

Name of the Technique	Input Size (Multiple Sensors)	Highlights	Pros	Cons
Camera + LiDAR Fusion [194] [114]	Image (Camera) + Point Cloud (LiDAR)	Combines strengths of cameras (visual information) and LiDAR (depth information)	- Improved accuracy and robustness in varying lighting conditions - More comprehensive understanding of the environment	- Requires careful calibration and synchronization of sensors - Increased computational complexity
Radar + Camera Fusion [181] [114]	Image (Camera) + Radar Range-Doppler data	Complements camera data with long-range object detection from radar	- Improved detection of distant objects and in low-visibility scenarios - Provides additional information like object velocity	- Radar data interpretation can be challenging - May require specialized processing techniques
LiDAR + IMU Fusion [195] [196]	Point Cloud (LiDAR) + Inertial Measurement Unit (IMU) data	Incorporates motion information from IMU for dynamic object tracking	- Improved tracking accuracy for moving objects - Compensates for LiDAR limitations in capturing motion	- Requires accurate sensor calibration and alignment - Increased complexity of data fusion algorithms
Multi-modal Transformer Fusion [197] [198]	Various sensor data (e.g., Camera, LiDAR, Radar)	Employs transformers to learn relationships between different sensor modalities	- Enables flexible fusion of diverse sensor data types - Potential for improved overall understanding of the environment	- Requires large amounts of training data for effective learning - High computational cost for complex transformer architectures

data. Table 10 shows a comparison of various fusion-based methods for 3D object detection.

This method improves detection accuracy by leveraging high-resolution information from images and spatial depth information from point clouds. MMF (Multi-Modal Fusion) [16] integrates features from multiple sensors (e.g., LiDAR, camera) using a deep neural network. It enhances object detection accuracy by fusing complementary information from different modalities, addressing the limitations of relying on a single sensor. PI-RCNN (Part-Informed RCNN) [35] incorporates part information into the Region-based Convolutional Neural Network (RCNN) framework. PI-RCNN improves detection accuracy and robustness using part-aware features, particularly for small and partially occluded objects. PointPainting [31] augments LiDAR point clouds with semantic segmentation outputs from an image-based neural network. This fusion of semantic and geometric information significantly improves the performance of 3D object detection models. Figure 25. Depicts a timeline of significant methods and advancements in multisensor-fusion-based 3D object detection for autonomous driving from 2017 to 2023.

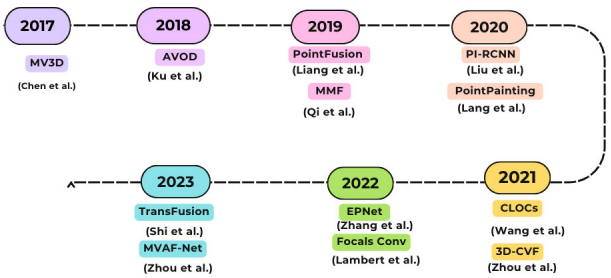


FIGURE 25. A timeline of significant methods in multisensor fusion methods.

CLOCs (Cross-Modality and LiDAR-Camera Object Candidates) [200] leverage both camera and LiDAR data by creating candidate boxes in each modality and then refining these proposals through cross-modal feature fusion, achieving superior detection accuracy.3D-CVF (Cross-View Fusion) [109] introduces a novel fusion strategy that combines bird's-eye view and perspective view features from LiDAR and camera data, respectively. This approach enhances detection robustness by integrating different perspectives. EPNet (Edge-enhanced PointNet) [201] integrates edge features into the PointNet architecture to better capture object boundaries and enhance detection accuracy, particularly for small and partially occluded objects. Focals Conv [202] incorporates focal loss and convolutional operations tailored for sparse data. This method improves detection precision by emphasizing difficult-to-detect objects during training.

Transfusion [203] utilizes transformer-based architectures to fuse multi-modal sensor data. The self-attention mechanism in trans-formers allows the model to effectively combine and learn from different sensor inputs, improving detection performance. MVAF-Net (Multi-View Attention Fusion Network) [204] uses attention mechanisms to fuse multi-view data from cameras and LiDAR. This approach enhances the network's ability to learn from diverse sensor inputs, improving detection accuracy and robustness.

Key Challenges of Sensor Fusion Methods:

Sensor Calibration and Synchronization: Accurate calibration and synchronization of different sensors is crucial for effective fusion, as data from different sensors may have varying resolutions, frame rates, and viewpoints. Misalignments and temporal offsets between sensor data can lead to inaccurate fusion results.

Computational Complexity: Fusing data from multiple sensors can be computationally intensive, especially when using deep learning models. Efficient architectures and parallelization techniques are required for real-time performance in autonomous vehicles.

Data Representation and Alignment: Different sensors provide data in different formats (e.g., images, point clouds, radar data), and aligning these diverse data representations for fusion can be challenging. Techniques for projecting and transforming data from different modalities into a common representation are essential.

Robustness and Generalization: Multi-sensor fusion systems must be robust to varying environmental conditions, sensor failures, and scenarios while generalizing well to new environments and situations. Achieving robust and generalizable fusion models is a significant challenge.

Interpretability and Uncertainty: Deep learning models for multi-sensor fusion can be complex and opaque, making it difficult to interpret their decisions and understand their uncertainties. Improving the interpretability and quantifying the uncertainty of fusion models is crucial for safety-critical applications like autonomous driving.

E. BACKBONE NETWORKS FOR FEATURE EXTRACTION

Deep learning has revolutionized object detection for autonomous vehicles. A crucial component of these systems is the backbone architecture, which extracts features from sensor data (camera images, LiDAR point clouds) for object identification and localization. The following section presents an exploration of existing work on backbone architectures for both 2D (camera) and 3D (LiDAR) object detection:

2D Object Detection: Convolutional Neural Networks (CNNs) are essential for 2D object detection, with popular architectures like VGG [132], ResNet [131], and Inception [205] excelling at learning hierarchical features from images. These networks start by extracting low-level features, such as edges and textures in the shallow layers, and progress to high-level features, like object parts and shapes in the deeper layers. Their strength lies in capturing spatial relationships within images.

MobileNet [124], designed for mobile and embedded vision applications, uses depth-wise separable convolutions to reduce the number of parameters and computations. It is commonly used in SSD (Single Shot MultiBox Detector) for real-time object detection. MobileNetV2 and MobileNetV3 further improve efficiency and performance, making them suitable for edge computing in autonomous vehicles. In Figure 26, a bar graph illustrates the commonly used backbone architectures from 2016 to 2024.

Efficient Net [206], a more recent CNN family, focuses on balancing accuracy and computational efficiency, making it ideal for real-time applications in autonomous vehicles. It employs a compound scaling method that proportionally increases depth, width, and resolution to maintain accuracy while reducing model size.

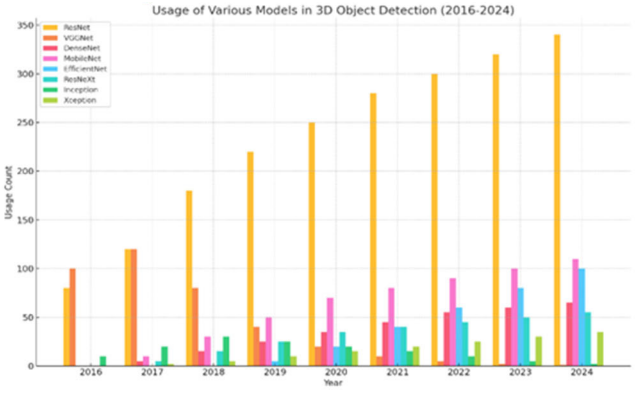


FIGURE 26. Commonly used backbone architecture during 2016–24.

Doing so will protect your figures from common font and arrow stroke issues when working on the files across multiple platforms. Point cloud architectures 3D convolutional neural networks (CNNs) have been pivotal in advancing 3D object detection for autonomous driving. Among point-based networks, PointNet, introduced by Qi et al. [30], was a pioneering architecture that directly processed raw point

clouds using shared multi-layer perceptions (MLPs) to extract features. Its successor, PointNet++ [108], incorporated hierarchical features learning to capture local structures in point clouds through a multi-scale approach. PointCNN, developed by Li et al., utilized X-convolution to exploit local geometric structures of point clouds, effectively learning feature representations from unordered point sets [207].

Voxel-based networks have also shown significant advancements. VoxelNet, proposed by Zhou and Tuzel, converted point clouds into a voxel grid and applied 3D convolutional layers to extract features, combining point-wise features within each voxel for robust detection [109]. SECOND, introduced by Yan et al., was an efficient implementation of VoxelNet that used sparse 3D convolutions, significantly reducing computational complexity and improving processing speed [106]. PointPillars, developed by Lang et al., encoded point cloud data into pseudo-images by dividing the space into vertical columns (pillars) and applied 2D convolutions for feature extraction, optimizing for real-time performance [31].

Graph-based architectures also excel in point cloud processing by representing data as graphs to capture geometric relationships. Dynamic Graph CNN (DGCNN) by Wang et al. constructed dynamic graphs to capture local geometric structures in point clouds and used edge convolution operations to learn features from graph representations [208]. Point-GNN, introduced by Shi et al., utilized graph neural networks to process point clouds, leveraging graph structures to capture spatial relationships and improve detection accuracy [167].

Hybrid networks combine various methods to enhance 3D object detection. PV-RCNN, proposed by Shi et al., combined voxel-based and point-based methods by first applying 3D sparse convolutions on voxel grids and then refining features using PointNet-based operations on raw points [66]. MVX-Net, developed by Sindagi et al., integrated multi-view data by combining 2D image features with 3D point cloud features, thereby improving detection performance through complementary sensor information [209]. Hybrid architectures blend different models to leverage their strengths. Point-BERT by Yu et al. merges transformer and graph-based modules to learn global context and local geometric features [210].

Transformer-based architectures, originally designed for natural language processing, have been adapted to computer vision for their ability to model long-range dependencies and capture global context. PointTransformer by Zhao et al. applied transformer architectures to point clouds, using self-attention mechanisms to capture long-range dependencies and enhance feature learning [211]. Voxel Transformer, introduced by Mao et al., incorporated transform-er modules within a voxel-based framework, leveraging attention mechanisms to process and fuse features from voxel grids [212]. Another noteworthy model, 3D-MAN by Fan et al., employs multi-head attention to capture local and global contexts, improving 3D object detection and point cloud segmentation [213].

Advanced convolutional networks like SE-SSD, developed by Zheng et al., integrated squeeze-and-excitation modules within a single-stage 3D detection framework, enhancing feature learning by recalibrating channel-wise feature responses [170]. CenterPoint, introduced by Yin et al., detected 3D objects by predicting the centers of objects and regressing to other properties using a combination of point-based and voxel-based feature extraction techniques [169]. These backbone architectures are essential for efficiently and accurately extracting features from complex 3D data, enabling robust 3D object detection in autonomous driving applications. The choice of backbone depends on the application's specific requirements, such as real-time processing capabilities, accuracy, and the type of sensor data available. These diverse architectural innovations in transformer-based, graph-based, and hybrid models push the boundaries of point cloud processing and 3D vision applications.

F. ATTENTION MECHANISMS AND VISION TRANSFORMER

1) ATTENTION MECHANISM IN CNNs

An attention mechanism is a component used in convolutional neural networks (CNNs) to focus on specific parts of an input.

Attention mechanisms enable neural networks to selectively concentrate on the input data's most pertinent aspects, augmenting their capacity to comprehend and model relationships and dependencies. Convolutional neural networks (CNNs) process input characteristics uniformly, but certain features may possess greater importance in determining the network's prediction. Attention mechanisms employ dynamic weighting of features according to their significance, allowing for selective concentration on the most critical aspects of the input. This process enhances the accuracy of tasks such as picture classification, object detection, and segmentation.

There are five sorts of attention methods: channel attention, spatial attention, temporal attention, branch attention, and hybrid attention [214]. Hybrid attention methods integrate core techniques like channel and spatial or temporal attention. Channel attention refers to the concentration on parts of the input (what to focus on), spatial attention indicates the direction of attention (where to focus), temporal attention determines when to prioritize important features (when to focus), and branch attention determines which segment of the input should be given attention (which to focus on). These concepts are primarily used in multi-branch architectures such as highway networks [215].

Among the several techniques for channeling attention, the Squeeze-and-Excitation Network (SENet) [216] is notable for being a groundbreaking attention module. Channel attention can be viewed as a process for selecting objects, as each channel corresponds to a unique item [217]. Spatial attention can be conceptualized as a system that chooses specific areas

that require attention. GENet [218] and PSANet [219] are spatial attention modules that utilize depth-wise convolution and subnetworks for feature aggregation [220]. The hybrid channel and spatial attention module is a flexible system that combines the benefits of both channel and spatial attention modules to choose items and locations. Hybrid attention modules, like the Residual Attention Network [221] and SCNet [222], simultaneously forecast channel and spatial attention networks.

2) VISION TRANSFORMERS (ViTs)

Vision Transformers (ViTs) is a neural network architecture that uses self-attention processes to analyze picture data, enabling more efficient capture of global context compared to conventional convolutional neural networks (CNNs) [211]. Visual Transformers (ViTs) have been investigated for 3D object recognition in autonomous driving and other domains. Visual Transformers (ViTs) segment a picture into patches, analogous to how words are processed in natural language processing. The patches are incorporated into a space with fewer dimensions and then inputted into a transformer model. The self-attention mechanism in Vision Transformers (ViTs) assesses the significance of several patches concerning one another, facilitating a thorough image comprehension [223]. Specific networks like LiDAR point clouds are explicitly designed to process sparse data. Some examples of transformers are the Point Transformer, Fast Point Transformer [212], and Voxel Transformer.

The need for attention mechanisms and Vision Transformers arises from their ability to enhance the performance of neural networks in capturing relevant features, understanding global context, and handling complex and large-scale tasks. These advancements are crucial for improving the accuracy and efficiency of models in various applications, including image classification, object detection, segmentation, and processing of 3D data in autonomous driving and other fields. Segregation of paper concerning datasets, classification of datasets, methods, and software.

V. SOFTWARE LIBRARIES AND FRAMEWORKS AND HARDWARE OPTIMIZATIONS FOR 3D OBJECT DETECTION

A. TOOLS AND LIBRARIES FOR IMPLEMENTING REAL-TIME DEEP LEARNING APPLICATIONS

1) THEANO

Theano [224] is a Python-based numerical library for fast mathematical computations that can run on both CPUs and GPUs. It is tightly integrated with NumPy [225]. Theano makes writing deep learning models easy and allows training models on GPUs for better performance. Neural networks and data are represented as matrices, and operations on data are defined as matrix calculations. Vectorized code runs quickly due to parallel execution. Theano was designed to handle large neural network computations used in deep learning and was one of the pioneering industry-standard libraries for deep learning research and development [32].

2) DeepLearning4j

DeepLearning4j [226] is an open-source deep learning framework developed by a machine learning group and supported by the Skymind team. It is written in Java and is compatible with any JVM-based language, such as Scala, Kotlin, or Clojure. The primary computations are written in C, C++, and CUDA. DeepLearning4j leverages various distributed computing frameworks like Apache Spark and Hadoop to speed up training. Its performance on multiple GPUs is equivalent to Caffe. It supports various deep neural networks, including neural tensor networks, deep autoencoders, deep be-lief nets, and restricted Boltzmann machines.

3) CAFFE

Caffe [227] is an open-source deep learning toolkit introduced by Berkeley AI Research (BAIR) and community contributors. It is written in C++ with a Python interface. Caffe is designed with expression, speed, and modularity in mind [228]. It supports models and optimization by configuration without hard coding. Caffe supports NVIDIA cuDNN and Intel MKL, GPU, and CPU-based acceleration computational kernel libraries. It supports various deep learning architectures for image classification and segmentation, including CNNs, LSTMs, RNNs, and deep neural networks. Caffe provides very limited abstraction, making it easy to perform unconventional, hardcore modifications.

4) TORCH

Torch [229] is an open-source scientific framework that extensively supports machine learning algorithms. It is a scripting language based on the Lua programming language and provides an interface to C. The core Torch package supports flexible N-dimensional arrays or Tensors and provides routines for indexing, transposing, slicing, resizing, and cloning. Torch provides optimized libraries to simplify the use of popular neural networks. Users can develop arbitrary graphs of neural networks and run them in parallel on CPUs and GPUs effectively [230].

5) TensorFlow

TensorFlow [231] is an open-source software library for high-performance mathematical computations. TensorFlow's flexible architecture allows easy deployment of computations on various platforms like CPUs, GPUs, and TPUs, ranging from single machines to clusters of servers to hand-held and edge devices. TensorFlow was invented to express numeric computations as a graph G (e, n), where n represents nodes and e represents edges. Mathematical operations of the graph are represented as nodes, while edges represent the tensors (multidimensional data) transferred between them.

A rich open-source software ecosystem significantly propels the rapid advancement in 3D object detection. These tools, rooted in deep learning and computer vision principles, provide researchers with high-level APIs, efficient implementations, and pre-trained models. This section

TABLE 12. Software libraries and tools used in object detection.

S.No	Framework /Library	Features	Interface	CNN	RNN	DBN/ RBM	Developer	Platform Support	GPU Support	Programming Languages	Model Support	Task
1	**Keras [10, 232]**	Fast prototyping, modular, minimalistic modules, extensible, arbitrary connection schemes	Python	√	√	√	F. Chollet	Linux, macOS, Windows	Yes	Python, R	CNN, RNN, DBN, RBM	Commonly used as a high-level interface for building and experimenting with deep learning models, including object detection and segmentation tasks.
2	**Deeplearning4j [11, 233, 234, 132]**	Distributed deep learning framework, microservice architecture	Python, Java, Scala	√	√	√	Skymind engineering team	Linux, macOS, Windows, Android	Yes	Java, Scala, Python (Keras), Kotlin	CNN, RNN, DBN, RBM	Distributed deep learning and large-scale training.
3	**Apache Singa [12, 235, 236]**	Scalable distributed training platform	Python, Java, C++	√	√	√	Apache Incubator	Linux, macOS, Windows	Yes	Python, C++, Java	CNN, RNN, DBN, RBM	Used in research papers exploring distributed deep learning frameworks and applications.
4	**MXNet [13, 233,238]**	A blend of symbolic programming and imperative programming, portability, auto-differentiation	Python, R, C++, Julia	√	√	Yes	Distributed (Deep) Machine Learning Community	Linux, macOS, Windows, Android, AWS, iOS, JavaScript	Yes	C++, Python, Julia, MATLAB, JavaScript, Go, R, Perl	CNN, RNN	Utilized in research papers involving large-scale distributed training, especially for computer vision tasks like object detection and segmentation.
5	**Neon [14, 239]**	Fastest performance on various hardware & deep networks (GoogLeNet, VGG, AlexNet, Generative Adversarial Networks)	Python	√	√	√	Intel	-	Yes	-	CNN, RNN, DBN, RBM	Utilized in research papers focusing on deep learning models and techniques, particularly in the computer vision domain.
6	**PyTorch [240]**	-	Python, C++	√	√	-	-	Linux, macOS, Windows	Yes	Python, C++	CNN, RNN	Gained widespread adoption in recent years for research in computer vision, natural language processing, and other domains.

TABLE 12. *(Continued.)* **Software libraries and tools used in object detection.**

7	**Caffe [4, 137]**	Modularity, speed, plaintext schema for modeling and optimization, data storage, and blob communication	C, C++, command line interface, Python, MATLAB	√	√	Yes	Berkeley Vision & Learning Center	Linux, macOS, Windows	Yes	Python, C++, Java	CNN, RNN	Widely adopted in computer vision research, particularly for image classification, object detection, and segmentation tasks.
8	**Microsoft Cognitive Toolkit CNTK [5]**	Automatic hyperparameter adjustment, batch normalization, multidimensional dense data processing	Python, C++, C#, command line interface	√	√	Yes	Microsoft Research	Windows, Linux, Command Line	Yes	Python, C++, Command Line	CNN, RNN	Used in research papers involving large-scale deep learning models and applications, including object detection and recognition tasks.
9	**TensorFlow [6]**	Math calculations employing data flow graph, inception, image classification, auto-differentiation, portability	C++, Python, Java, Go	√	√	√	Google Brain team	Linux, macOS, Windows, Android	Yes	Python (Keras), C/C++, Java, Go, JavaScript, R, Julia	CNN, RNN, DBN, RBM	Widely used in various research areas, including computer vision, natural language processing, and recommendation systems.
10	**Theano [7]**	Multi-dimensional array math expression compilation	Python	√	√	√	Université de Montréal	Cross-platform	Yes	Python (Keras)	CNN, RNN, DBN, RBM	Developing and experimenting with new deep learning models and architectures.
11	**Torch [8]**	Support for N-dimensional arrays, automated gradient differentiation, neural models, and energy models	C, C++, Lua	√	√	√	R. Collobert, K. Kavukcuoglu, C. Farabet	Linux, macOS, Windows, Android	Yes	Lua, C, OpenCL/C++ utilities	CNN, RNN, DBN, RBM	Widely used for image classification and object detection tasks.
12	**Chainer [9]**	Define-by-Run network definition, multi-GPU parallelization Forward/Backward	Python	√	√	Yes	Preferred Networks Inc.	Linux, macOS	Yes	Python	CNN, RNN	Used in exploring new deep learning architectures and techniques, particularly in the computer vision domain.

surveys key libraries that are shaping the 3D perception landscape.

Table 12 provides a detailed discussion of various libraries and their features.

B. HARDWARE AND OPTIMIZATIONS FOR REAL-TIME 3D OBJECT DETECTION IN AUTONOMOUS DRIVING

Specific hardware optimizations significantly improve real-time 3D object detection in autonomous driving. Using GPUs with tools like CUDA and cuDNN can dramatically reduce computation time by optimizing GPU performance [241] [242]. Techniques like sparse convolutions, seen in networks such as SECOND, improve efficiency by processing only the essential parts of 3D data [106]. Quantization, available in tools like TensorRT, lowers memory usage and speeds up computation by converting models to lower precision [243]. Pruning unnecessary weights in neural networks reduces model size and speeds up processing [244]. Edge computing with devices like NVIDIA Jetson AGX Xavier combines multiple processing units on a single chip for efficient real-time processing [245]. These hardware optimizations create highly efficient and robust 3D object detection systems for autonomous vehicles.

VI. APPLICATIONS

3D object detection has widespread applications across various domains, leveraging the ability to perceive and understand the three-dimensional environment. This section explores various applications of 3D object detection techniques, highlighting their significance and impact.

A. AUTONOMOUS VEHICLES AND ROBOTICS

Autonomous Driving: 3D object detection enables autonomous vehicles to perceive and understand their surrounding environment, allowing them to detect pedestrians, vehicles, and other objects for safe navigation and collision avoidance.

Robotics: In robotics, 3D object detection plays a crucial role in robotic navigation, object manipulation, and human-robot interaction. Robots with 3D object detection capabilities can better perceive and interact with their surroundings.

B. AUGMENTED REALITY AND VIRTUAL REALITY

Augmented Reality (AR): AR applications use 3D object detection to overlay virtual objects or information onto the real world. For example, AR navigation systems can use 3D object detection to recognize landmarks and guide users with augmented directions.

Virtual Reality (VR): In VR environments, 3D object detection enables the creation of immersive experiences by accurately detecting and interacting with virtual objects in three-dimensional space.

C. HEALTHCARE AND MEDICAL IMAGING

Medical Imaging: In medical imaging, 3D object detection is used for tasks such as tumor detection, organ segmentation, and anatomical landmark localization. Accurate detection of anatomical structures is essential for diagnosis, treatment planning, and surgical guidance.

Healthcare Robotics: Robotics-assisted surgeries rely on 3D object detection to provide real-time feedback to surgeons and assist in precise instrument positioning, enhancing surgical accuracy and patient outcomes.

D. URBAN PLANNING AND SMART CITIES

Urban Planning: 3D object detection aids urban planners in analyzing urban landscapes, detecting infrastructure elements such as buildings, roads, and utilities, and assessing the environmental impact of urban development projects.

Smart Transportation: In smart cities, 3D object detection is used for traffic monitoring, pedestrian detection, and intelligent transportation systems to improve traffic flow, enhance safety, and optimize urban mobility.

Video Surveillance: Surveillance systems leverage 3D object detection to detect and track suspicious activities, identify intruders, and monitor crowd behavior in public spaces, airports, and critical infrastructure facilities.

Border Control: Border security agencies use 3D object detection to detect smuggling activities, monitor border crossings, and identify threats in border regions, enhancing national security and border control efforts.

Industrial Automation and Manufacturing

Quality Inspection: In manufacturing, 3D object detection is used for quality inspection of products, detecting defects, and ensuring compliance with manufacturing standards.

Warehouse Automation: Automated warehouses utilize 3D object detection for inventory management, palletizing, and robotic picking, improving efficiency and productivity in logistics operations.

E. ENTERTAINMENT AND GAMING

Virtual Gaming: Gaming applications incorporate 3D object detection for player tracking, gesture recognition, and interactive gameplay experiences, enhancing immersion and realism in virtual environments.

Film and Animation: In film production and animation, 3D object detection is used for motion capture, character animation, and special effects, facilitating the creation of visually stunning and lifelike digital content. Table 12 provides a comprehensive discussion of various application areas, along with the corresponding datasets and techniques used for 3D object detection.

F. ENVIRONMENTAL MONITORING AND AGRICULTURE

Environmental Monitoring: 3D object detection aids in ecological monitoring by detecting and analyzing natural phenomena such as landslides, forest fires, and changes in terrain morphology.

TABLE 13. Applications of 3D object detection in various sectors.

Application Area	Year	Focus	Datasets	Techniques Used	Description	Results/Performance
Indoor Scene Understanding [246]	2022	3D object detection and instance segmentation	ScanNet, SUN RGB-D	Deep Learning, Point Cloud Processing	Novel 3D object detection and instance segmentation approach	Improved performance on benchmark datasets
Robotics and Automation [248][249]	2022	6-DoF Grasping	YCB, NIST	Reinforcement Learning, Deep Learning	Discovery of intra-grasp and in-hand manipulation affordance cues	Improved grasping success rate
Healthcare [256] [257]	2022	Medical Image Analysis	Public Medical Datasets	Deep Learning, 3D Object Detection	3D object detection and segmentation for medical image analysis	Improved accuracy in medical image analysis tasks
Augmented and Virtual Reality [260] [261]	2022	Augmented Reality Applications	Custom Dataset	Deep Learning, Object Tracking	3D object detection and tracking for augmented reality applications	Accurate object detection and tracking for AR experiences
Retail and Logistics [258] [259]	2021	Warehouse Automation	Custom Dataset	Deep Learning, Pose Estimation	3D object detection and pose estimation for warehouse automation	Improved efficiency in warehouse operations
Augmented and Virtual Reality [260] [261]	2021	Construction Safety (VR)	Custom Dataset	Deep Learning, Object Detection	Virtual reality for construction safety using 3D object detection and tracking	Improved safety training and hazard awareness
Autonomous Driving [18]	2021	Pedestrian and Cyclist Detection	KITTI, nuScenes	Deep Learning, 3D Object Detection	3D object detection and tracking for pedestrian and cyclist safety in autonomous driving	Accurate detection and tracking of pedestrians and cyclists
Augmented Reality [247]	2020	3D object detection and pose estimation	Custom RGB-D Dataset	Deep Learning, Pose Estimation	3D object detection and pose estimation for AR applications	Real-time AR experience
Agriculture [252] [253]	2020	Fruit Picking Robots	Custom Dataset	Deep Learning, Instance Segmentation	3D object detection and instance segmentation for fruit-picking robots	Improved picking success rate
Construction and Infrastructure [254][255]	2020	Infrastructure Monitoring	Custom Dataset	Deep Learning, Instance Segmentation	3D object detection and instance segmentation for infrastructure monitoring	Accurate detection and segmentation of infrastructure components
Cultural Heritage [262] [263]	2020	Virtual Museum Applications	Custom Dataset	Deep Learning, 3D Reconstruction	3D object detection and reconstruction for cultural heritage applications using deep learning	Immersive virtual museum experiences
Robotics and Automation [248][249]	2019	Ambidextrous Robot Grasping	Dexterity Network (Dex-Net)	Deep Learning, Imitation Learning	Learning ambidextrous robot grasping policies	Successful grasping in cluttered environments
Surveillance and Security [250]	2019	Indoor Object Detection	SUN RGB-D	Deep Learning, Point Cloud Processing	3D object detection and instance segmentation for indoor spaces	Accurate object detection and segmentation

TABLE 13. *(Continued.)* **Applications of 3D object detection in various sectors.**

Cultural Heritage [262] [263]	2019	Cultural Heritage Preservation	Custom Dataset	Deep Learning, 3D Reconstruction	3D object detection and reconstruction for cultural heritage applications	Accurate 3D reconstruction of cultural heritage artifacts
Autonomous Driving [251]	2018	Multi-view 3D object detection	KITTI, nuScenes	Deep Learning, Multi-view Fusion	Multi-view 3D object detection for autonomous driving	Improved performance on benchmark datasets
Construction and Infrastructure [254][255]	2018	Construction Safety	Custom Dataset	Deep Learning, Object Detection	3D object detection and 3D modelling for construction safety	Improved safety monitoring and hazard detection
Healthcare [256] [257]	2018	Surgical Robot Guidance	Custom Dataset	Deep Learning, Instrument Segmentation	Automatic instrument segmentation in robot-assisted surgery	Accurate instrument segmentation for surgical robot guidance
Retail and Logistics [258] [259]	2018	Amazon Picking Challenge	Custom Dataset	Deep Learning, Multi-view Fusion	Multi-view self-supervised deep learning for 6D pose estimation	Successful object picking and pose estimation
Agriculture [252] [253]	2017	Fruit Detection	Custom Dataset	Deep Learning, Object Detection	Real-time deep neural network for fruit detection	Accurate fruit detection and localization

Precision Agriculture: In agriculture, 3D object detection is used for crop monitoring, yield estimation, and precision farming applications, optimizing resource allocation and improving crop yields.

These applications demonstrate the versatility and importance of 3D object detection across various domains, ranging from automotive and healthcare to entertainment and agriculture, contributing to technological advancements and enhancing our daily lives. This broader scope also provides a more complete picture of the current state of the art, making the review more valuable to researchers and practitioners.

VII. CHALLENGES, OPEN ISSUES, AND FUTURE DIRECTIONS

The challenges in 3D object detection include varying sensing conditions, unstructured data formats, inclement weather, insufficient training sets, and unbalanced datasets. Unlike image data, point cloud data is unstructured, unordered, and sparse, making its processing more complex and time-consuming due to many empty points. Although various sensor fusion techniques have been developed for perception systems, there is no consensus on the optimal method for sensor fusion. In some applications, early fusion yields better results, while in others, deep fusion techniques outperform other methods. Consequently, key questions remain unresolved, including the optimal point cloud data representation, the most effective fusion techniques, and the efficient processing of point cloud data. 3D object detection for autonomous driving is a rapidly evolving field driven by advancements in deep learning and multisensory fusion technologies. However, several challenges and open issues remain, shaping future research directions.

A. CURRENT ISSUES IN 3D OBJECT DETECTION

1) SENSOR LIMITATIONS

Cameras, LiDAR, and radar are some of the sensors that have their characteristics with pros and cons. Indeed, LiDAR can provide accurate, in-depth information and perform well in different weather conditions, though sometimes it may struggle with close-range detection and is expensive. Cameras are fine for recognizing textures and colors but do not provide depth data well enough to work effectively in 3D without assistance. Although resistant to adverse weather, radars have low resolution and suffer from signal interference.

2) ENVIRONMENTAL VARIATIONS

The performance of 3D object detection systems is greatly influenced by environmental variations that include but are not limited to changes in weather conditions such as rain, fog, and snow, as well as lighting conditions. These factors may curtail the sensor from optimal object detection and classification; hence, a multisensory approach is needed to enhance resilience.

3) OCCLUSIONS AND SCALE VARIATIONS

The objects present in the driving environment may occur due to some other objects or obstacles. This makes detection

highly complicated. Variations in object scales due to their distances may also result in detection inaccuracies.

4) DATA SPARSITY AND ANNOTATION

LiDAR data provides valuable spatial information but is often sparse and unstructured, complicating processing. Additionally, training deep learning models requires large, labeled datasets, which can be resource-intensive and time-consuming to acquire.

5) PRIVACY CONCERNS IN AUTONOMOUS VEHICLE

Data Collection and Usage: Autonomous vehicles collect vast amounts of data, including location, behavior, and biometric data, which pose significant privacy risks. The potential misuse of this data can lead to surveillance, tracking, and unauthorized profiling of drivers and passengers.

Cybersecurity Risks: AVs are vulnerable to cyber threats, including hacking, unauthorized access, and control of vehicle systems, which can endanger occupants and others on the road. Security protocols like encryption, access controls, and anomaly detection are essential to mitigate these risks.

Privacy Enhancing Technologies (PETS): PETs, such as anonymization and data minimization, help protect user privacy while allowing for valuable data processing. Differential privacy and multi-party computation enable secure data sharing without compromising privacy.

6) ECONOMIC AND REGULATORY CHALLENGES

Implementing 3D object detection in autonomous vehicles faces several economic and regulatory challenges. Economically, high sensor costs, significant computational requirements, and the lack of economies of scale hinder widespread adoption. Additionally, the labor-intensive process of annotating large datasets adds to development expenses. Moreover, the homologation process for regulatory approval is time-consuming and costly. Addressing these challenges requires collaboration among industry stakeholders, government, and academia to develop effective solutions and standards for safe deployment.

7) ETHICAL AND SAFETY CONCERNS

Decision-Making in Inevitable Accidents: Self-driving cars need moral guidelines that can be adjusted to make ethical decisions during inevitable accidents, weighing the safety of pedestrians and occupants.

The Protection of Data and Openness: Ethical concerns regarding ownership, consent, and transparency are raised by using data in autonomous vehicles. Maintaining public trust by ensuring users have control over their data and that it is used ethically is essential.

Trust and Approval From Society: The ability of machines to make decisions that can determine life or death poses concerns about society's confidence in self-driving cars. Creating strong ethical guidelines and legal frameworks is crucial for ensuring these entities' safe and reliable functioning.

Safety Issues: Safety is of utmost importance in autonomous vehicle networks, as any security breach can result in severe consequences like accidents or traffic disturbances.

Adherence to Safety Regulations: Low adherence to international standards such as ISO 21434 and UNECE WP.29 emphasizes the importance of increased standardization among regions to support the worldwide adoption of AVs. Adhering to different regional regulations and having strong security measures in place can increase production expenses and restrict economies of scale in a business. Manufacturers must balance being innovative and dealing with regulatory compliance expenses.

Global Standards Alignment: Differences in geographical regulations make it challenging to implement AV technologies. Global standards alignment is crucial for the widespread integration of AVs.

8) OTHER COMMON CHALLENGES

3D object detection in autonomous systems faces several challenges, including the sensitivity of sensor performance to environmental factors like lighting and weather, leading to inconsistent data quality.

Often used in these systems, point cloud data is inherently unstructured and sparse, complicating processing and interpretation. Adverse weather conditions further degrade sensor data, reducing detection reliability. The limited availability and uneven distribution of training datasets result in biased models that struggle to generalize across different environments. Moreover, the complexity of combining data from multiple sensors remains unresolved, with no consensus on optimal fusion strategies.

While vision transformers offer promising results, their high computational demands make them challenging to deploy in resource-constrained environments, highlighting the need for lightweight and efficient models suitable for real-time applications. Enhancing model generalization with limited labeled data through semi-supervised learning, improving domain adaptation, and ensuring transparency in decision-making are critical areas where research gaps exist.

Addressing these challenges and gaps is essential for advancing the reliability and effectiveness of 3D object detection systems in autonomous driving.

B. RESEARCH GAPS AND FUTURE DIRECTIONS

Future directions for 3D object detection must focus on several key objectives to enhance performance and applicability.

Accurate Detection: The goal of accurate detection involves developing methods to reliably detect and classify objects in 3D space, even in complex scenes with occlusions and varying object scales. Current 3D object detection algorithms struggle with accurately detecting objects in cluttered environments and at long distances. Improving

detection accuracy in these scenarios is crucial for reliable autonomous driving. Techniques such as advanced deep learning models like PointNet++ and VoxelNet, hierarchical structures, data augmentation, and transfer learning are critical.

Unsupervised Domain Adaptation: In many cases, labeled data in the new domain (e.g., rural setting) might be scarce or unavailable. Unsupervised domain adaptation techniques enable the model to learn from the new domain's unlabeled data, leveraging the latest data's structure and features to align it with the labeled data from the original domain.

Efficient Processing: Efficient processing aims to handle large amounts of 3D data and complex scenes while maintaining real-time capabilities. Processing high-resolution point clouds in real-time requires significant computational resources. Optimizing computational efficiency is necessary for practical deployment in autonomous vehicles. This can also be achieved by optimizing processing pipelines with octree structures and graph-based methods.

Robustness to Variations: Different environments can vary significantly in data distributions, such as visual features, traffic patterns, and sensor readings. For example, an urban environment differs considerably from a rural one. 3D object detection algorithms often fail in adverse weather conditions like rain, fog, or snow. Increasing robustness under these conditions is essential for reliable autonomous operation. Domain adaptation techniques are crucial for helping models adjust to these variations, ensuring accurate predictions even when the data encountered differs from the training data. Integrating multimodal sensor data (e.g., radar, cameras) with lidar is needed to improve detection robustness.

Transfer Learning: Transfer learning is a common approach in domain adaptation, where a model trained in one domain (e.g., urban) is fine-tuned with data from another domain (e.g., rural). This allows the model to retain its learned knowledge while adapting to the new environment, improving its performance without extensive retraining from scratch.

Real-Time Processing: Achieving real-time processing capabilities is necessary for applications requiring timely decision-making, such as autonomous driving and robotics. This involves leveraging advanced hardware acceleration and optimizing processing pipelines. Achieving real-time processing is challenging due to the high data rate of lidar sensors. Enhancing real-time capabilities is critical for timely decision-making in autonomous driving. A real-time 3D object detection pipeline that utilizes hardware acceleration (e.g., GPUs, FPGAs) and optimized software frameworks is required. There is a need for practical implementation of domain adaptation techniques in real-world autonomous vehicle (AV) systems, particularly when deploying vehicles across different geographic regions or transitioning between urban and rural environments. Research should focus on testing and refining domain adaptation techniques in real-world scenarios to ensure they enhance the robustness and reliability of AV systems.

Validation and Testing Protocols: Validating and testing 3D object detection systems for autonomous vehicles ensures their reliability and safety before deployment. This process involves using comprehensive datasets, such as KITTI and nuScenes, to evaluate performance across various scenarios. Simulation environments like CARLA and LGSVL enable extensive testing in controlled and repeatable conditions. In contrast, real-world testing on closed courses and public roads helps identify potential issues in actual driving conditions. Safety and reliability metrics, including false positive rate, false negative rate, mean average precision, localization error, and orientation error, are used to assess the systems' performance.

Vision transformers, although highly effective, demand significant processing resources and necessitate optimization for contexts with limited resources. It is necessary to develop models that are lightweight, efficient, and capable of multitasking. Additionally, improving generalization abilities through semi-supervised learning and producing balanced datasets that accurately represent real-world conditions are also important. Furthermore, incorporating domain adaptation and explainable AI is crucial to upholding the efficacy of models in various settings and establishing confidence in autonomous systems.

In summary, implementing advanced models like PointNet++ and VoxelNet, optimizing processing techniques, enhancing robustness through various methods, improving generalization, and achieving real-time processing are critical steps toward advancing 3D object detection technology.

VIII. CONCLUSION

Self-driving cars, also known as autonomous vehicles (AVs), are a groundbreaking technology that has the potential to reshape transportation completely. This review has examined the crucial significance of 3D object detection in improving the safety and effectiveness of autonomous vehicles (AVs). We have emphasized the importance of incorporating 3D object identification techniques into self-driving technologies, underscoring their indispensability for precise perception and decision-making.

Our investigation was extensive and encompassed a range of elements, such as sensors, databases, and the pipeline for 3D object detection. Crucial sensors and standard datasets like KITTI, Waymo, and NuScenes have been essential in improving detection algorithms and enabling comparison assessments. Performance measures like Average Precision (AP) and Intersection over Union (IoU) are crucial for assessing detection accuracy. The review analyzed various methodologies and backbone structures for 2D and 3D object detection. We examined techniques that rely on visual data, point cloud data, and many sensors, emphasizing their benefits and drawbacks. In addition, we examined

software libraries that facilitate the execution of real-time deep-learning applications.

When considering the uses of 3D object detection, we observed its importance in other fields, specifically in improving autonomous driving systems. Our analysis revealed persistent difficulties, such as managing obstructions, variations in size, and guaranteeing immediate processing. The survey also discussed potential future focus areas, such as enhancing the ability to handle variances, optimizing processing efficiency, and strengthening generalization capabilities.

Advancements in 3D object detection research are underway, with vision transformers paving the way for new opportunities in innovation. State-of-the-art real-time 3D object detection models have been effectively utilized in autonomous cars, unmanned aerial vehicles, and intelligent inspection robots. Subsequent investigations should prioritize utilizing attention processes, knowledge distillation, transfer learning, and domain adaptability to augment detection performance, efficiency, and generalization in intricate scenarios.

This comprehensive survey highlights the significant challenges in 3D object detection, including occlusion, scale variation, and real-time processing demands. Handling unstructured, unordered, and sparse point cloud data in 3D object detection and representation presents several significant challenges. Existing methodologies, such as voxel processing, PointNet, and projection views, often incur high computational costs and can lead to data loss during projection, impacting accuracy. Moreover, no universally agreed-upon method for representing point cloud data complicates research and application in this field. Fusion techniques, whether early, middle, or late, show variable effectiveness in integrating data from various sensors (e.g., LiDAR, RGB cameras, radar) and do not provide a one-size-fits-all solution. While multitask learning can enhance efficiency by allowing simultaneous processing of tasks like 3D object detection and semantic segmentation, designing effective models remains a significant challenge.

Additionally, semi-supervised learning can potentially reduce the high costs associated with annotating large datasets, but its application in 3D object detection is still limited. Current datasets often fail to adequately represent real-world conditions, such as adverse weather or nighttime scenarios, necessitating the creation of more representative datasets, which can be costly.

Domain adaptation is another critical issue, as deep learning models frequently struggle to maintain performance when applied to domains other than those they were trained in, underscoring the need for adaptive models. The complexity of deep learning models often renders them "black boxes," complicating the interpretability of their decision-making processes. Furthermore, there is a pressing demand for robust and lightweight 3D object detection models that can operate effectively on embedded devices in real-time scenarios [11].

TABLE 14. List of acronyms.

LiDAR	Light Detection and Ranging
3D	Three-Dimensional
PRISMA	Preferred Reporting Items for Systematic Reviews and Meta-Analyses
KITTI	Karlsruhe Institute of Technology and Toyota Technological Institute
AP	Average Precision
IoU	Intersection over Union
GPS	Global Positioning System
ToF	Time-of-Flight
CNN	Convolutional Neural Networks
RNN	Recurrent Neural Networks
SLAM	Simultaneous Localization and Mapping
HD	High Definition
GIoU	Generalized Intersection over Union
Map	Mean Average Precision
ALP	Average Localization Precision
AOP	Average Orientation Precision
ATE	Average Translation Error
ASE	Average Scale Error
TPR	True Positive Rate
FPR	False Positive Rate
FPS	Frames Per Second
ROC	Receiver Operating Characteristic
NMS	Non-Maximum Suppression
AB3DMOT	Associating and Tracking 3D Objects in Multiple Frames with Multi-Object Tracking
YOLO	You Only Look Once
SSD	Single Shot MultiBox Detector
FCOS	Fully Convolutional One-Stage Object Detection
FPN	Feature Pyramid Networks
SORT	Simple Online and real-time tracking
Deep SORT	Deep Simple Online and Real-time Tracking
RPN	Region Proposal Network
FPN	Feature Pyramid Network
LiDAR	Light Detection and Ranging
BEV	Bird's Eye View
GNN	Graph Neural Network
SSD	Single Shot MultiBox Detector

TABLE 14. *(Continued.)* **List of acronyms.**

RCNN	Region-based Convolutional Neural Network
ViT	Vision Transformer
DCN	Dynamic Convolution Network
3DOP	3D Object Proposals
DSGN	Depth-guided Stereo 3D Object Detection
MVX-Net	Multi-View Extremal Regions Network
M3D-RPN	Multi-task, Multi-sensor, Multi-scale Region Proposal Network
PointNet	Point-based Neural Network
VoxelNet	Voxel-based Neural Network
SECOND	Sparsely Embedded Convolutional Detection
PV-RCNN	Point-Voxel RCNN
CIA-SSD	Categorical Information Aggregation SSD
3DSSD	3D Single-Stage Speed and Accuracy
SE-SSD	Self-Ensembling SSD
CT3D	Contextual 3D
MDRNet	Multi-Dimensional Residual Network
EPNet	Edge-enhanced PointNet
CLOCs	Cross-modality and LiDAR-Camera Object Candidates

Traditional pooling and convolution layers can lead to information loss, necessitating exploring alternative operations that preserve data integrity. Lastly, integrating architectures like transformers with convolutional neural networks (CNNs) holds promise for enhancing detection performance but requires substantial training data and presents unique design challenges.

By addressing these critical issues, we aim to guide future research directions that can enhance the effectiveness and reliability of 3D object detection systems. The insights provided in this paper are intended to support researchers and practitioners in developing robust and efficient solutions for autonomous driving, ultimately contributing to the safe and successful deployment of autonomous vehicles in diverse environments.

To summarize, this review offers helpful perspectives and recommendations for scholars and practitioners involved in autonomous driving. By tackling the specified difficulties and utilizing the described approaches, the progress of strong, effective, and dependable 3D object recognition systems can be significantly enhanced, leading to safer and more efficient autonomous cars. Deep learning improves traffic flow and system performance by enabling cars to sense and respond to their surroundings accurately. This survey provides a thorough basis for future progress and breakthroughs in autonomous driving technologies.

In conclusion, this paper highlights the need for ongoing innovation and interdisciplinary collaboration to address current challenges. The aim is to enhance detection systems so they can reliably operate in real-world conditions, ultimately advancing safer and more autonomous vehicle technologies that have the potential to transform transportation.

REFERENCES

[1] S. Pendleton, H. Andersen, X. Du, X. Shen, M. Meghjani, Y. Eng, D. Rus, and M. Ang, "Perception, planning, control, and coordination for autonomous vehicles," *Machines*, vol. 5, no. 1, p. 6, Feb. 2017.

[2] Badue, R. Guidolini, R. V. Carneiro, P. Azevedo, V. B. Cardoso, A. Forechi, and A. F. De Souza, "Self-driving cars: A survey," *Expert Syst. Appl.*, vol. 165, Jul. 2021, Art. no. 113816.

[3] A. Gupta, A. Anpalagan, L. Guan, and A. S. Khwaja, "Deep learning for object detection and scene perception in self-driving cars: Survey, challenges, and open issues," *Array*, vol. 10, Jul. 2021, Art. no. 100057.

[4] M. Liang, B. Yang, Y. Chen, R. Hu, and R. Urtasun, "Multi-task multi-sensor fusion for 3D object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 7337–7345.

[5] D. Isele, R. Rahimi, A. Cosgun, K. Subramanian, and K. Fujimura, "Navigating occluded intersections with autonomous vehicles using deep reinforcement learning," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2018, pp. 2034–2039.

[6] B. Paden, M. Cáp, S. Z. Yong, D. Yershov, and E. Frazzoli, "A survey of motion planning and control techniques for self-driving urban vehicles," *IEEE Trans. Intell. Vehicles*, vol. 1, no. 1, pp. 33–55, Mar. 2016.

[7] Z. Song, L. Liu, F. Jia, Y. Luo, G. Zhang, L. Yang, L. Wang, and C. Jia, "Robustness-aware 3D object detection in autonomous driving: A review and outlook," 2024, *arXiv:2401.06542*.

[8] M. Drobnitzky, J. Friederich, B. Egger, and P. Zschech, "Survey and systematization of 3D object detection models and methods," *Vis. Comput.*, vol. 40, no. 3, pp. 1867–1913, Mar. 2024.

[9] G. Zamanakos, L. Tsochatzidis, A. Amanatiadis, and I. Pratikakis, "A comprehensive survey of LiDAR-based 3D object detection methods with deep learning for autonomous driving," *Comput. Graph.*, vol. 99, pp. 153–181, Oct. 2021.

[10] S.-H. Kim and Y. Hwang, "A survey on deep learning based methods and datasets for monocular 3D object detection," *Electronics*, vol. 10, no. 4, p. 517, Feb. 2021.

[11] S. Alaba, A. Gurbuz, and J. Ball, "A comprehensive survey of deep learning multisensor fusion-based 3D object detection for autonomous driving: Methods, challenges, open issues, and future directions," *Authorea Preprints*, vol. 2023, pp. 1–17, Aug. 2023.

[12] S. Y. Alaba, A. C. Gurbuz, and J. E. Ball, "Emerging trends in autonomous vehicle perception: Multimodal fusion for 3D object detection," *World Electr. Vehicle J.*, vol. 15, no. 1, p. 20, Jan. 2024.

[13] M. S. U. Khan, A. Pagani, M. Liwicki, D. Stricker, and M. Z. Afzal, "Three-dimensional reconstruction from a single RGB image using deep learning: A review," *J. Imag.*, vol. 8, no. 9, p. 225, Aug. 2022.

[14] Hussain and S. R. Mehdi, "A comprehensive review: 3D object detection based on visible light camera, infrared camera, and LiDAR in dark scene," SSRN, Apr. 2024, doi: 10.2139/ssrn.4781073.

[15] M. Contreras, A. Jain, N. P. Bhatt, A. Banerjee, and E. Hashemi, "A survey on 3D object detection in real time for autonomous driving," *Frontiers Robot. AI*, vol. 11, pp. 45–67, Mar. 2024.

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

[17] J. Mao, S. Shi, X. Wang, and H. Li, "3D object detection for autonomous driving: A comprehensive survey," *Int. J. Comput. Vis.*, vol. 131, no. 8, pp. 1909–1963, Aug. 2023.

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

[19] Z. Yin, T. Darrell, and F. Yu, "Hierarchical discrete distribution decomposition for match density estimation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 6037–6046.

[20] P. Koopman and M. Wagner, "Autonomous vehicle safety: An interdisciplinary challenge," *IEEE Intell. Transp. Syst. Mag.*, vol. 9, no. 1, pp. 90–96, Spring 2017.

[21] H. Xu, G. Lan, S. Wu, and Q. Hao, "Online intelligent calibration of cameras and LiDARs for autonomous driving systems," in *Proc. IEEE Intell. Transp. Syst. Conf. (ITSC)*, Oct. 2019, pp. 3913–3920.

[22] K. A. Tychola, E. Vrochidou, and G. A. Papakostas, "Deep learning based computer vision under the prism of 3D point clouds: A systematic review," *Vis. Comput.*, vol. 40, no. 11, pp. 8287–8329, Nov. 2024.

[23] Peng, "Deep learning for 3D object detection and tracking in autonomous driving: A brief survey," arXiv:2311.06043, 2023.

[24] A. F. M. S. Saif and Z. R. Mahayuddin, "Vision based 3D object detection using deep learning: Methods with challenges and applications towards future directions," *Int. J. Adv. Comput. Sci. Appl.*, vol. 13, no. 11, pp. 203–214, 2022.

[25] *Waymo's Self-Driving Cars Are Being Put to Work in a Completely Driverless Ride-hailing Service in Phoenix*, Hawkins, Verge, Mumbai, India, Apr. 22, 2020.

[26] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby, "An image is worth 16×16 words: Transformers for image recognition at scale," 2020, *arXiv:2010.11929*.

[27] D. Moher, "Preferred reporting items for systematic reviews and meta-analyses: The PRISMA statement," *Ann. Internal Med.*, vol. 151, no. 4, p. 264, Aug. 2009.

[28] Y. Guo, H. Wang, Q. Hu, H. Liu, L. Liu, and M. Bennamoun, "Deep learning for 3D point clouds: A survey," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 43, no. 12, pp. 4338–4364, Dec. 2021.

[29] S. Zhou, H. Xu, G. Zhang, T. Ma, and Y. Yang, "Leveraging deep convolutional neural networks pre-trained on autonomous driving data for vehicle detection from roadside LiDAR data," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 11, pp. 22367–22377, Nov. 2022.

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

[31] A. H. Lang, S. Vora, H. Caesar, L. Zhou, J. Yang, and O. Beijbom, "PointPillars: Fast encoders for object detection from point clouds," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 12689–12697.

[32] F. Nobis, M. Geisslinger, M. Weber, J. Betz, and M. Lienkamp, "A deep learning-based radar and camera sensor fusion architecture for object detection," in *Proc. IEEE Sensor Data Fusion Trends Solutions Appl. (SDF)*, Nov. 2019, pp. 1–7.

[33] T. Yin, X. Zhou, and P. Krähenbühl, "Center-based 3D object detection and tracking," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 11779–11788.

[34] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? The KITTI vision benchmark suite," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2012, pp. 3354–3361.

[35] S. Shi, X. Wang, and H. Li, "PointRCNN: 3D object proposal generation and detection from point cloud," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 770–779.

[36] B. Xu and Z. Chen, "Multi-level fusion based 3D object detection from monocular images," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 2345–2353.

[37] M. Bijelic, T. Gruber, and W. Ritter, "A benchmarking of LiDAR sensors for outdoor environmental mapping," *Sensors*, vol. 18, no. 10, p. 3235, 2018.

[38] J. Ku, A. D. Pon, and S. L. Waslander, "Monocular 3D object detection leveraging accurate proposals and shape reconstruction," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 11859–11868.

[39] H. Laga, L. Josifovski, and J. Kehren, "Convolutional neural networks for depth estimation from stereo 360° Panoramas," *Computer. Vis. Image Understand.*, vol. 197, Apr. 2020, Art. no. 103010.

[40] J. Choi, "Range sensors: Ultrasonic sensors, Kinect, and LiDAR," in *Humanoid Robotics: A Reference*, A. Goswami and P. Vadakkepat, Eds. Dordrecht, The Netherlands: Springer, 2019, pp. 2521–2538.

[41] P. Li, X. Chen, and S. Shen, "Stereo R-CNN based 3D object detection for autonomous driving," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 7636–7644.

[42] J. Zbontar and Y. LeCun, "Stereo matching by training a convolutional neural network to compare image patches," *J. Mach. Learn. Res.*, vol. 17, pp. 2287–2318, Apr. 2016.

[43] Y. You, Y. Wang, W.-L. Chao, D. Garg, G. Pleiss, B. Hariharan, M. Campbell, and K. Q. Weinberger, "Pseudo-LiDAR++: Accurate depth for 3D object detection in autonomous driving," in *Proc. ICLR*, 2020, pp. 1–22.

[44] Z. Qin, J. Wang, and Y. Lu, "MonoGRNet: A geometric reasoning network for monocular 3D object localization," in *Proc. AAAI Conf. Artif. Intell.*, vol. 33, Jan. 2019, pp. 8851–8858.

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

[46] Y. You, "Pseudo-LiDAR++: Accurate depth for 3D object detection in autonomous driving," in *Proc. IEEE Int. Conf. Robot. Autom.*, Oct. 2020, pp. 9906–9912.

[47] Y. Zhu, Y. Tian, D. Metaxas, and P. Dollár, "Robust stereo matching with surface normal prediction," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, May 2017, pp. 2540–2548.

[48] M. Hansard, S. Lee, O. Choi, and R. P. Horaud, *Time-of-Flight Cameras: Principles, Methods and Applications*. Cham, Switzerland: Springer, 2012.

[49] S. Foix, G. Alenya, and C. Torras, "Lock-in time-of-flight (ToF) cameras: A survey," *IEEE Sensors J.*, vol. 11, no. 9, pp. 1917–1926, Sep. 2011.

[50] F. Darabi and S. Jalalzadeh, "One-loop quantum cosmological correction to the gravitational constant using the kink solution in de sitter universe," 2010, *arXiv:1011.0557*.

[51] S. Rathore, H. Akolekar, P. Jain, S. Sarkar, and V. Kanagaraj, "Depth-assisted sensor fusion for autonomous driving: A review," *IEEE Sensors J.*, vol. 22, no. 8, pp. 6283–6301, Aug. 2022.

[52] R. Gade and T. B. Moeslund, "Thermal cameras and applications: A survey," *Mach. Vis. Appl.*, vol. 25, no. 1, pp. 245–262, Jan. 2014.

[53] B. Miethig, A. Liu, S. Habibi, and M. V. Mohrenschildt, "Leveraging thermal imaging for autonomous driving," in *Proc. IEEE Transp. Electrific. Conf. Expo (ITEC)*, Jun. 2019, pp. 1–5.

[54] J. D. Choi and M. Y. Kim, "A sensor fusion system with thermal infrared camera and LiDAR for autonomous vehicles and deep learning based object detection," *ICT Exp.*, vol. 9, no. 2, pp. 222–227, Apr. 2023.

[55] J. Portmann, S. Lynen, M. Chli, and R. Siegwart, "People detection and tracking from aerial thermal views," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2014, pp. 1794–1800.

[56] B. Major, D. Fontijne, A. Ansari, R. T. Sukhavasi, R. Gowaikar, M. Hamilton, S. Lee, S. Grzechnik, and S. Subramanian, "Vehicle detection with automotive radar using deep learning on range-azimuth-Doppler tensors," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. Workshop (ICCVW)*, Oct. 2019, pp. 924–932.

[57] M. Meyer and G. Kuschk, "Deep learning based 3D object detection for automotive radar and camera," in *Proc. 16th Eur. Radar Conf. (EuRAD)*, Oct. 2019, pp. 133–136.

[58] Y. Zhou, L. Liu, H. Zhao, M. López-Benítez, L. Yu, and Y. Yue, "Towards deep radar perception for autonomous driving: Datasets, methods, and challenges," *Sensors*, vol. 22, no. 11, p. 4208, May 2022.

[59] S. M. Patole, M. Torlak, D. Wang, and M. Ali, "Automotive radars: A review of signal processing techniques," *IEEE Signal Process. Mag.*, vol. 34, no. 2, pp. 22–35, Mar. 2017.

[60] T. K. Chan and C. S. Chin, "Review of autonomous intelligent vehicles for urban driving and parking," *Electronics*, vol. 10, no. 9, p. 1021, Apr. 2021.

[61] T. Nesti, S. Boddana, and B. Yaman, "Ultrasonic sensor-based object detection for autonomous vehicles," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jul. 2023, pp. 210–218.

[62] D. J. Yeong, G. Velasco-Hernandez, J. Barry, and J. Walsh, "Sensor and sensor fusion technology in autonomous vehicles: A review," *Sensors*, vol. 21, no. 6, p. 2140, Mar. 2021.

[63] H. Durrant-Whyte and T. Bailey, "Simultaneous localization and mapping: Part I," *IEEE Robot. Autom. Mag.*, vol. 13, no. 2, pp. 99–110, Jun. 2006.

[64] M. Bastico, "Simultaneous depth completion and 3D object detection via deep learning for scene reconstruction in autonomous driving scenarios," Doctoral dissertation, Telecommun. Polytech. Univ. Madrid Spain 2021.

[65] Y. Gao, S. Liu, M. Atia, and A. Noureldin, "INS/GPS/LiDAR integrated navigation system for urban and indoor environments using hybrid scan matching algorithm," *Sensors*, vol. 15, no. 9, pp. 23286–23302, Sep. 2015.

[66] S. Shi, C. Guo, L. Jiang, Z. Wang, J. Shi, X. Wang, and H. Li, "PV-RCNN: Point-voxel feature set abstraction for 3D object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 10526–10535.

[67] H. Caesar, V. Bankiti, A. H. Lang, S. Vora, V. E. Liong, Q. Xu, A. Krishnan, Y. Pan, G. Baldan, and O. Beijbom, "NuScenes: A multimodal dataset for autonomous driving," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 11618–11628.

[68] X. Zhu, H. Zhou, T. Wang, F. Hong, Y. Ma, W. Li, H. Li, and D. Lin, "Cylindrical and asymmetrical 3D convolution networks for LiDAR segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 9934–9943.

[69] R. Nabati and H. Qi, "CenterFusion: Center-based radar and camera fusion for 3D object detection," in *Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, Jan. 2021, pp. 1526–1535.

[70] Q.-H. Pham, P. Sevestre, R. S. Pahwa, H. Zhan, C. H. Pang, Y. Chen, A. Mustafa, V. Chandrasekhar, and J. Lin, "A*3D dataset: Towards autonomous driving in challenging environments," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2020, pp. 2267–2273.

[71] Z. Ren, "Range adaptation for 3D object detection in LiDAR," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, May 2022, pp. 4005–4014.

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

[73] Battra, "Multi-modal fusion for 3D object detection and tracking," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, Jul. 2021, pp. 10473–10479.

[74] A. Carballo, J. Lambert, A. Monrroy-Cano, D. R. Wong, P. Narksri, Y. Kitsukawa, E. Takeuchi, S. Kato, and K. Takeda, "LIBRE: The multiple 3D LiDAR dataset," 2020, *arXiv:2003.06129*.

[75] M. Zhu, C. Ma, P. Ji, and X. Yang, "Cross-modality 3D object detection," in *Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, Jan. 2021, pp. 3771–3780.

[76] P. Sun, "Scalability in perception for autonomous driving: Waymo open dataset," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 2443–2451.

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

[78] S. Mandal, S. Biswas, V. E. Balas, R. N. Shaw, and A. Ghosh, "Lyft 3D object detection for autonomous vehicles," in *Artificial Intelligence for Future Generation Robotics*. Amsterdam, The Netherlands: Elsevier, 2021, pp. 119–136.

[79] J. Houston, G. Zuidhof, L. Bergamini, Y. Ye, L. Chen, A. Jain, S. Omari, V. Iglovikov, and P. Ondruska, "One thousand and one hours: Self-driving motion prediction dataset," 2020, *arXiv:2006.14480*.

[80] P. Xiao, Z. Shao, S. Hao, Z. Zhang, X. Chai, J. Jiao, Z. Li, J. Wu, K. Sun, K. Jiang, Y. Wang, and D. Yang, "PandaSet: Advanced sensor suite dataset for autonomous driving," in *Proc. IEEE Int. Intell. Transp. Syst. Conf. (ITSC)*, Sep. 2021, pp. 3095–3101.

[81] J. Tang, S. Zou, J. Ju, X. Li, D. Chen, and Y. Xu, "Research on 3D point cloud object detection method based on pointnet model," in *Proc. 5th Int. Conf. Frontiers Technol. Inf. Comput. (ICFTIC)*, Qiangdao, China, 2023, pp. 315–318, doi: 10.1109/ICFTIC59930.2023.10456265.

[82] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele, "The cityscapes dataset for semantic urban scene understanding," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 3213–3223.

[83] W. Luo, A. G. Schwing, and R. Urtasun, "Efficient deep learning for stereo matching," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 5695–5703.

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

[85] X. Wang, K. Li, and A. Chehri, "Multi-sensor fusion technology for 3D object detection in autonomous driving: A review," *IEEE Trans. Intell. Transp. Syst.*, vol. 25, no. 2, pp. 1148–1165, Feb. 2024.

[86] X. Huang, X. Cheng, Q. Geng, B. Cao, D. Zhou, P. Wang, Y. Lin, and R. Yang, "The ApolloScape dataset for autonomous driving," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2018, pp. 1067–10676.

[87] Huang, "Multi-task learning for dense prediction tasks: A survey," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 7, pp. 4047–4062, Jul. 2020.

[88] G. Yang, X. Song, C. Huang, Z. Deng, J. Shi, and B. Zhou, "DrivingStereo: A large-scale dataset for stereo matching in autonomous driving scenarios," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 899–908.

[89] X. Song, G. Yang, X. Zhu, H. Zhou, Y. Ma, Z. Wang, and J. Shi, "AdaStereo: An efficient domain-adaptive stereo matching approach," *Int. J. Comput. Vis.*, vol. 130, no. 1, pp. 1–20, 2022.

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

[91] P. Agarwal, S. Kumar, J. Ryde, J. Corso, V. Krovi, and N. Ahmed, *Towards Persistent Localization and Mapping With a Continuous Appearance-Based Topology*. Cambridge, MA, USA: MIT Press, 2013.

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

[93] Y. Zhang, P. David, and B. Gong, "Curriculum domain adaptation for semantic segmentation of urban scenes," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 2039–2049.

[94] Q. Hu, "One million scenes for autonomous driving: ONCE dataset," 2022, *arXiv:2205.03099*.

[95] X. Ye, M. Shu, H. Li, Y. Shi, Y. Li, G. Wang, and E. Ding, "Rope3d: The roadside perception dataset for autonomous driving and monocular 3d object detection task," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2022.

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

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

[98] R. D. Singh, A. Mittal, and R. K. Bhatia, "3D convolutional neural network for object recognition: A review," *Multimedia Tools Appl.*, vol. 78, no. 12, pp. 15951–15995, Jun. 2019.

[99] Y. Wang and J. Ye, "An overview of 3D object detection," 2020, *arXiv:2010.15614*.

[100] R. Padilla, W. L. Passos, T. L. B. Dias, S. L. Netto, and E. A. B. da Silva, "A comparative analysis of object detection metrics with a companion open-source toolkit," *Electronics*, vol. 10, no. 3, p. 279, Jan. 2021.

[101] Mousavian, D. Anguelov, J. Flynn, and J. Košecká, "3D bounding box estimation using deep learning and geometry," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jul. 2017, pp. 7074–7082.

[102] T. Lin, "Microsoft COCO: Common objects in context," in *Proc. Eur. Conf. Comput. Vis.*, 2014, pp. 740–755.

[103] F. Manhardt, W. Kehl, and A. Gaidon, "ROI-10D: Monocular lifting of 2D detection to 6D pose and metric shape," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 2064–2073.

[104] Y. Xiang, W. Choi, Y. Lin, and S. Savarese, "Data-driven 3D voxel patterns for object category recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 1903–1911.

[105] Z. Yang, Y. Sun, S. Liu, and J. Jia, "3DSSD: Point-based 3D single stage object detector," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 11037–11045.

[106] Y. Yan, Y. Mao, and B. Li, "SECOND: Sparsely embedded convolutional detection," *Sensors*, vol. 18, no. 10, p. 3337, Oct. 2018, doi: 10.3390/s18103337.

[107] Y. Shen, Z. Feng, L. Liu, and H. Zhao, "Performance evaluation of object detection based on ROC curve," *J. Phys., Conf. Ser.*, vol. 887, Jul. 2017, Art. no. 012007.

[108] R. Qi, L. Yi, H. Su, and L. J. Guibas, "PointNet++: Deep hierarchical feature learning on point sets in a metric space," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 30, 2017, pp. 1–10.

[109] Y. Zhou and O. Tuzel, "VoxelNet: End-to-end learning for point cloud based 3D object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 4490–4499.

[110] J. Ku, M. Mozifian, J. Lee, A. Harakeh, and S. L. Waslander, "Joint 3D proposal generation and object detection from view aggregation," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Jan. 2019, pp. 1–8.

[111] A. P. Bradley, "The use of the area under the ROC curve in the evaluation of machine learning algorithms," *Pattern Recognit.*, vol. 30, no. 7, pp. 1145–1159, Jul. 1997.

[112] S. Vora, A. H. Lang, B. Helou, and O. Beijbom, "PointPainting: Sequential fusion for 3D object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 4603–4611.

[113] Z. Bai, D. Bi, J. Wu, M. Wang, Q. Zheng, and L. Chen, "Intelligent driving vehicle object detection based on improved AVOD algorithm for the fusion of LiDAR and visual information," *Actuators*, vol. 11, no. 10, p. 272, Sep. 2022.

[114] S. Scheidegger, J. Benjaminsson, E. Rosenberg, A. Krishnan, and K. Granström, "Mono-camera 3D multi-object tracking using deep learning detections and PMBM filtering," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Jun. 2018, pp. 433–440.

[115] X. Weng, J. Wang, D. Held, and K. Kitani, "3D multi-object tracking: A baseline and new evaluation metrics," 2019, *arXiv:1907.03961*.

[116] Ramos, A. Correia, and R. J. Rossetti, "Assessing the YOLO series through empirical analysis on the KITTI dataset for autonomous driving," in *Proc. Int. Conf. Intell. Transport Syst.* Cham, Switzerland: Springer, 2019, pp. 203–218.

[117] D. Meimetis, I. Daramouskas, I. Perikos, and I. Hatzilygeroudis, "Real-time multiple object tracking using deep learning methods," *Neural Comput. Appl.*, vol. 35, no. 1, pp. 89–118, 2023.

[118] X. Jia, Y. Tong, H. Qiao, M. Li, J. Tong, and B. Liang, "Fast and accurate object detector for autonomous driving based on improved YOLOv5," *Sci. Rep.*, vol. 13, no. 1, p. 9711, Jun. 2023.

[119] T. Li and Q. Zhao, "AE-YOLO: An improved YOLOv7 based on attention enhancement for rail flaw detection," *IFAC-PapersOnLine*, vol. 56, no. 2, pp. 7638–7643, 2023.

[120] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.

[121] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, "SSD: Single shoat multibox detector," in *Proc. ECCV*, in Lecture Notes in Computer Science, vol. 2016, 2016, pp. 21–37.

[122] J. Redmon and A. Farhadi, "YOLO9000: Better, faster, stronger," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Honolulu, HI, USA, Jul. 2017, pp. 6517–6525.

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

[124] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, "MobileNets: Efficient convolutional neural networks for mobile vision applications," 2017, *arXiv:1704.04861*.

[125] J. Redmon and A. Farhadi, "YOLOv3: An incremental improvement," 2018, *arXiv:1804.02767*.

[126] Z. Tian, C. Shen, H. Chen, and T. He, "FCOS: Fully convolutional one-stage object detection," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Seoul, South Korea, Oct. 2019, pp. 9626–9635.

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

[128] X. Zhou, J. Zhuo, and P. Krähenbuhl, "Bottom-up object detection by grouping extreme and center points," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Long Beach, CA, USA, Jun. 2019, pp. 850–859.

[129] Fu, W. Liu, D. Huang, and J. Deng, "DSSD: Dense single shot detector," 2017, *arXiv:1704.04854*.

[130] Z. Cai and N. Vasconcelos, "Cascade R-CNN: Delving into high quality object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Salt Lake City, UT, USA, Jun. 2018, pp. 6154–6162.

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

[132] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, San Diego, CA, USA, 2015, pp. 1–14.

[133] S. Xie, R. Girshick, P. Dollár, Z. Tu, and K. He, "Aggregated residual transformations for deep neural networks," 2016, *arXiv:1611.05431*.

[134] S.-H. Gao, M.-M. Cheng, K. Zhao, X.-Y. Zhang, M.-H. Yang, and P. Torr, "Res2Net: A new multi-scale backbone architecture," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 43, no. 2, pp. 652–662, Feb. 2021.

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

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

[137] S. Rinchen and Y. Sun, "Multi-task learning for object detection in autonomous driving," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Sep. 2022, pp. 1292–1301.

[138] Karypidis, "Self-supervised learning for 3D object detection using point contrastive learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2024, pp. 5678–5687.

[139] P. Li, X. Chen, and S. Shen, "RTM3D: Real-time monocular 3D detection from object keypoints for autonomous driving," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2020, pp. 1–11.

[140] Chen, H. Ma, J. Wan, B. Li, and T. Xia, "3D object proposals for accurate object class detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 5, pp. 1161–1174, May 2017.

[141] X. Chen, K. Kundu, Y. Zhu, H. Ma, S. Fidler, and R. Urtasun, "3D object proposals for accurate object class detection," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, vol. 28, 2018, pp. 1–9.

[142] P. Li, X. Chen, S. Shen, and C. Xu, "DeepMANTA: A multi-task network for monocular 3D object detection and tracking," *IEEE Trans. Image Process.*, vol. 28, no. 3, pp. 1041–1055, Mar. 2019.

[143] G. Brazil and X. Liu, "M3D-RPN: Monocular 3D region proposal network for object detection," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 9286–9295.

[144] S. R. Dubey, S. K. Singh, B. B. Chaudhuri, and S. Rani, "MonoGRNet: Monocular geometric reasoning network for 3D object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 10320–10329.

[145] Z. Liu, Z. Wu, and R. Tóth, "SMOKE: Single-stage monocular 3D object detection via keypoint estimation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2020, pp. 4289–4298.

[146] Y. Chen, L. Tai, K. Sun, and M. Li, "MonoPair: Monocular 3D object detection using pairwise spatial relationships," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 12090–12099.

[147] L. Zhenyu, Z. Li, Z. Chen, A. Li, L. Fang, Q. Jiang, Xi. Liu, and J. Jiang, "Towards model generalization for monocular 3d object detection,"

[148] T. Wang, J. Pang, and D. Lin, "Monocular 3d object detection with depth from motion," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2022.

[149] F. Sattar, F. Karray, M. Kamel, L. Nassar, and K. Golestan, "Recent advances on context-awareness and data/information fusion in ITS," *Int. J. Intell. Transp. Syst. Res.*, vol. 14, pp. 1–19, 2016.

[150] X. Jiang, S. Jin, X. Zhang, L. Shao, and S. Lu, "MonoMAE: Enhancing monocular 3D detection through depth-aware masked autoencoders," 2024, *arXiv:2405.07696*.

[151] H. Ranasinghe, A. Hegde, and V. Patel, "3D object detection and pose estimation using monocular images," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2024, pp. 1–4.

[152] N. Mayer, E. Ilg, P. Häusser, P. Fischer, D. Cremers, A. Dosovitskiy, and T. Brox, "A large dataset to train convolutional networks for disparity, optical flow, and scene flow estimation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 4040–4048.

[153] T. Wang, X. Zhu, J. Pang, and D. Lin, "FCOS3D: Fully convolutional one-stage monocular 3D object detection," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. Workshops (ICCVW)*, Oct. 2021, pp. 913–922.

[154] J. Zhang, G. Lin, X. Song, and Z. Liu, "Vision transformers (ViTs) for multi-camera 3D object detection in autonomous vehicles," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Aug. 2021, pp. 1–13.

[155] Y. Song, J. Kim, H. Choi, and S. Lee, "MDS Net: A one-stage monocular 3D object detection network for autonomous driving," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 1–10.

[156] S. Qu, X. Yang, Y. Gao, and S. Liang, "MonoDCN: Monocular 3D object detection based on dynamic convolution," *PLoS ONE*, vol. 17, no. 10, Oct. 2022, Art. no. e0275438.

[157] R. Szeliski, *Computer Vision: Algorithms and Applications*. Cham, Switzerland: Springer, 2010.

[158] A. Fusiello, E. Trucco, and A. Verri, "A compact algorithm for rectification of stereo pairs," *Mach. Vis. Appl.*, vol. 12, no. 1, pp. 16–22, Jul. 2000.

[159] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*. Cambridge, U.K.: Cambridge Univ. Press, 2003.

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

[161] Xu and Z. Chen, "STEREOBlackboxee: A stereo image-based 3D object detection system for autonomous vehicles," 2020, *arXiv:2004.08900*.

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

[163] Y. Chen, S. Liu, X. Shen, and J. Jia, "DSGN: Deep stereo geometry network for 3D object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Seattle, WA, USA, Jun. 2020, pp. 12533–12542.

[164] C. Tao, J. Cao, C. Wang, Z. Zhang, and Z. Gao, "Pseudo-mono for monocular 3D object detection in autonomous driving," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 33, no. 8, pp. 3962–3975, Jan. 2023.

[165] C. Wang, Y. Dai, W. He, Y. Gu, and Q. Tian, "TransFusion: Robust LiDAR-camera 3D object detection with transformer," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Aug. 2022, pp. 1–4.

[166] X. Zhu, H. Hu, S. Lin, and J. Dai, "Deformable ConvNets V2: More deformable, better results," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 9300–9308.

[167] W. Shi and R. Rajkumar, "Point-GNN: Graph neural network for 3D object detection in a point cloud," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 1708–1716, doi: 10.1109/CVPR42600.2020.00178.

[168] Y. Yan, Y. Zhang, Y. Mao, and B. Li, "CIA-SSD: Confident IoU-aware single-stage object detector from point cloud," in *Proc. AAAI Conf. Artif. Intell.*, 2021, vol. 35, no. 3, pp. 3169–3176, doi: 10.1609/aaai.v35i3.16359.

[169] Y. Yan, Y. Mao, and B. Li, "CenterPoint: Center-based 3D object detection and tracking," *IEEE Trans. Intell. Vehicles*, vol. 7, no. 1, pp. 1–10, Jan. 2022, doi: 10.1109/TIV.2021.3074584.

[170] W. Zheng, J. Tang, and B. Li, "SE-SSD: Self-ensembling single-stage object detector from point cloud," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 14494–14503, doi: 10.1109/CVPR42600.2020.01007.

[171] Yang, A. H. Lang, and S. Vora, "Voxel R-CNN: Efficient point cloud voxelization for 3D object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Aug. 2023, pp. 10629–10638, doi: 10.1109/CVPR42600.2020.01125.

[172] Y. Cai, X. Wang, and B. Li, "CT3D: Contextual 3D object detection using multi-modal sensor fusion," *IEEE Trans. Intell. Transp. Syst.*, vol. 24, no. 2, pp. 1–10, Feb. 2023, doi: 10.1109/TITS.2022.3141775.

[173] R. Trilaksono, T. Bonse, J. Peissig, and D. Haase, "Efficient point-GNN for real-time LiDAR 3D object detection," in *Proc. IEEE/CVF Winter Conf. Appl. Comput. Vis. (WACV)*, Aug. 2022, pp. 2286–2295.

[174] Q. Tong, D. Meng, W. Yuan, G. Xie, A. Yuille, T. Hospedales, and C. Gong, "MDRNet: Multi-level dimensionality reduction network for 3D object detection from point clouds," 2204, *arXiv:2204.04996*.

[175] S. Ibrahim, V. Zouhar, P. Stohl, M. Čadík, and J. Šochman, "VoxSter: A novel real-time and extended field-of-view LiDAR 3D object detection system," 2021, *arXiv:2110.07712*.

[176] Y. Li, L. Ma, Z. Zhong, F. Liu, M. A. Chapman, D. Cao, and J. Li, "Deep learning for LiDAR point clouds in autonomous driving: A review," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 32, no. 8, pp. 3412–3432, Aug. 2021.

[177] Kozma, J. Vaisanen, S. Koskinen, E. Rahtu, and J. Kangas, "3D object detection from pseudo-labels and novel augmentations for autonomous driving," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jul. 2021, pp. 472–473.

[178] Z. Yang, Y. Sun, S. Liu, X. Shen, and J. Jia, "STD: Sparse-to-dense 3D object detector for point cloud," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 1951–1960.

[179] J. Li, X. Dai, Z. Zhang, Y. Chen, Y. Zhang, and J. Yu, "TED: Tiny and efficient 3D object detection from point clouds," 2022, *arXiv:2203.08781*.

[180] Z. Zhang, J. Sun, H. Ma, Y. Zhou, W. Wan, J. Zhang, and D. Liang, "PLOT: Pillar-based lightweight object detection network for autonomous vehicles," 2022, *arXiv:2205.08049*.

[181] M. Li, Y. Hu, N. Zhao, and Q. Qian, "One-stage multi-sensor data fusion convolutional neural network for 3D object detection," *Sensors*, vol. 19, no. 6, p. 1434, Mar. 2019, doi: 10.3390/s19061434.

[182] S. Y. Alaba and J. Ball, "Multi-sensor fusion 3D object detection for autonomous driving," *Proc. SPIE*, vol. 12540, pp. 36–43, Jun. 2023, doi: 10.1117/12.2663424.

[183] L. Chen, Z. Wang, H. Wang, and P. Zhou, "3D object detection based on neighborhood graph search in dense scenes," in *Proc. 3rd Int. Conf. Robot. Control Eng.*, May 2023, pp. 184–190, doi: 10.1145/3598151.3598182.

[184] K. Liu, Y. Li, X. Chen, and B. Luo, "PMPF: Point-cloud multiple-pixel fusion for 3D object detection," 2022, *arXiv:2209.09483*.

[185] M. Zhao, S. He, Y. Jiang, W. Wan, H. Ma, and Z. Zhang, "Cross-modal panoramic driving perception network for automotive multi-task learning," 2022, *arXiv:2209.08639*.

[186] Y. Hou, J. Ren, Q. Mazumder, Z. Yuan, Y. Shen, and W. W. Cohen, "Multi-sensor 3D object detection for autonomous vehicles: Leveraging uncertainty propagation from sensors to predictions," 2022, *arXiv:2204.00826*.

[187] T. R. Adekallu, A. Singh, A. Wang, and S. Gupta, "OD-C3DL: Object detection using camera and 3D LiDAR for autonomous vehicles," 2022, *arXiv:2205.14957*.

[188] M. Bhardwaj, S. Gupta, and B. N. Vellambi, "DOPS: Fusing LiDAR and camera data for improving 3D object detection in autonomous vehicles," 2021, *arXiv:2110.03109*.

[189] Y. Li, D. Wang, W. Zhang, H. Ma, and H. Fu, "MVMM: Multi-view mixture of modalities for 3D object detection," 2021, *arXiv:2112.06924*.

[190] Z. Dai, Z. Guan, Q. Chen, Y. Xu, and F. Sun, "Enhanced object detection in autonomous vehicles through LiDAR—Camera sensor fusion," *World Electr. Vehicle J.*, vol. 15, no. 7, p. 297, Jul. 2024.

[191] S. Vora, A. H. Lang, B. Helou, and O. Beijbom, "PointPaintingNet: Sequential fusion for 3D object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Aug. 2020, pp. 4604–4612.

[192] Z. Li, L. Zhang, L. Zhu, H. Tan, H. Cao, and J. Huang, "CLRNet: Camera-LiDAR-radar multi-modal object detection for autonomous driving," 2021, *arXiv:2111.07792*.

[193] M. Liang, B. Yang, S. Wang, and R. Urtasun, "Deep continuous fusion for multi-sensor 3D object detection," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2019, pp. 641–656.

[194] Y. Zhao, S. Luo, X. Huang, and D. Wei, "A multi-sensor 3D detection method for small objects," *World Electr. Vehicle J.*, vol. 15, no. 5, p. 210, May 2024.

[195] Y. Cheng, X. Lu, J. Zhou, Y. Zheng, and H. Bao, "Improving 3D object detection by fusion of LiDAR and IMU data," *IEEE Trans. Intell. Transp. Syst.*, vol. 22, no. 8, pp. 5301–5311, Aug. 2020.

[196] W. Wan, K. Xu, X. Wang, G. Xie, and S. Pu, "Robust LiDAR-IMU integration for autonomous driving: A GNSS-aided LiDAR-IMU calibration approach," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Mar. 2020, pp. 1–3.

[197] Y. Wang, V. C. Guizilini, T. Zhang, Y. Wang, H. Zhao, and J. Solomon, "Detr3D: Delving into the efficient reading of 3D Siamese representations via industrial grade 3D detection networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Oct. 2022, pp. 1–11.

[198] Zhu, X. Ma, Y. Jiang, Z. Cai, and R. Urtasun, "Cylindrical-attentional multi-view fusion for 3D object detection," 2022, *arXiv:2205.07807*.

[199] Chen, H. Ma, J. Wan, B. Li, and T. Xia, "Multi-view 3D object detection network for autonomous driving," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 1907–1915, doi: 10.1109/CVPR.2017.205.

[200] S. Pang, D. Morris, and H. Radha, "CLOCs: Camera-LiDAR object candidates fusion for 3D object detection," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Oct. 2020, pp. 10386–10393, doi: 10.1109/IROS45743.2020.9341791.

[201] Y. Zhang, Q. Lu, D. Zhou, and L. Zhang, "EPNet: Enhancing point features with edge representation for 3D object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Sep. 2022, pp. 4981–4990, doi: 10.1109/CVPR.2022.00498.

[202] J. Lambert, Z. Liu, J. Hays, and S. Narasimhan, "Focal loss for dense object detection," *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Apr. 2020, pp. 2980–2988.

[203] S. Shi, Z. Wang, and J. Shi, "TransFusion: Robust LiDAR-camera fusion for 3D object detection with transformers," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2023, pp. 12041–12050, doi: 10.1109/CVPR.2023.01239.

[204] Y. Zhou, S. Shi, Z. Wang, and C. R. Qi, "Multi-view attention fusion network for 3D object detection," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Jun. 2023, pp. 10231–10240, doi: 10.1109/ICCV.2023.01123.

[205] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 1–9.

[206] M. Tan and Q. Le, "EfficientNet: Rethinking model scaling for convolutional neural networks," in *Proc. Int. Conf. Mach. Learn.*, 2019, pp. 6105–6114, doi: 10.1109/ICML.2019.6105.

[207] Y. Li, R. Bu, M. Sun, W. Wu, X. Di, and B. Chen, "PointCNN: Convolution on X-transformed points," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 31, 2018, pp. 1–11.

[208] Y. Wang, Y. Sun, Z. Liu, S. E. Sarma, M. M. Bronstein, and J. M. Solomon, "Dynamic graph CNN for learning on point clouds," *ACM Trans. Graph.*, vol. 38, no. 5, pp. 1–12, Oct. 2019.

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

[210] W. Yu, C. Luo, P. Zhou, X. Yang, J. Liu, and J. Lu, "Point-BERT: Pre-trained point cloud transformer for 3D vision," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Aug. 2022, pp. 1–14.

[211] H. Zhao, L. Jiang, J. Jia, P. Torr, and V. Koltun, "Point transformer," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 16239–16248.

[212] J. Mao, Y. Xue, M. Niu, H. Bai, J. Feng, X. Liang, H. Xu, and C. Xu, "Voxel transformer for 3D object detection," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Nashville, TN, USA, Oct. 2021, pp. 3144–3153.

[213] L. Fan, Z. Pang, T. Zhang, Y.-X. Wang, H. Zhao, F. Wang, N. Wang, and Z. Zhang, "Embracing single stride 3D object detector with sparse transformer," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 8448–8458.

[214] M.-H. Guo, T.-X. Xu, J.-J. Liu, Z.-N. Liu, P.-T. Jiang, T.-J. Mu, S.-H. Zhang, R. R. Martin, M.-M. Cheng, and S.-M. Hu, "Attention mechanisms in computer vision: A survey," *Comput. Vis. Media*, vol. 8, no. 3, pp. 331–368, 2022.

[215] R. K. Srivastava, K. Greff, and J. Schmidhuber, "Training very deep networks," in *Proc. 28th Int. Conf. Adv. Neural Inf. Process. Syst.*, 2015, pp. 2377–2385.

[216] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Salt Lake City, UT, USA, Jun. 2018, pp. 7132–7141.

[217] L. Chen, H. Zhang, J. Xiao, L. Nie, J. Shao, W. Liu, and T.-S. Chua, "SCA-CNN: Spatial and channel-wise attention in convolutional networks for image captioning," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Honolulu, HI, USA, Jul. 2017, pp. 5659–5667.

[218] J. Hu, L. Shen, S. Albanie, G. Sun, and A. Vedaldi, "Gather-excite: Exploiting feature context in convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 31, 2018, pp. 1–13.

[219] H. Zhao, Y. Zhang, S. Liu, J. Shi, C. C. Loy, D. Lin, and J. Jia, "PSANet: Point-wise spatial attention network for scene parsing," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, Munich, Germany, 2018, pp. 267–283.

[220] F. Chollet, "Xception: Deep learning with depthwise separable convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Honolulu, HI, USA, Jul. 2017, pp. 1800–1807.

[221] F. Wang, M. Jiang, C. Qian, S. Yang, C. Li, H. Zhang, X. Wang, and X. Tang, "Residual attention network for image classification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Honolulu, HI, USA, Jul. 2017, pp. 6450–6458.

[222] J.-J. Liu, Q. Hou, M.-M. Cheng, C. Wang, and J. Feng, "Improving convolutional networks with self-calibrated convolutions," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Seattle, WA, USA, Jun. 2020, pp. 10093–10102.

[223] C. Park, Y. Jeong, M. Cho, and J. Park, "Fast point transformer," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, New Orleans, LA, USA, Jun. 2022, pp. 16928–16937.

[224] T. Theano Development Team, "Theano: A Python framework for fast computation of mathematical expressions," 2016, *arXiv:1605.02688*.

[225] C. R. Harris, "Array programming with NumPy," *Nature*, vol. 585, no. 7825, pp. 357–362, Sep. 2020.

[226] *Deeplearning4j: Open Source Distributed Deep Learning for the JVM*, Deeplearning4j Develop. Team, Apache Softw. Found. License, Forest Hill, MA, USA, 2016.

[227] J. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, and T. Darrell, "Caffe: Convolutional architecture for fast feature embedding," in *Proc. 22nd ACM Int. Conf. Multimedia*, 2014, pp. 675–678.

[228] Caffe. *Caffe: Deep Learning Framework*. Accessed: Nov. 14, 2024. [Online]. Available: https://caffe.berkeleyvision.org/

[229] R. Collobert, K. Kavukcuoglu, and C. Farabet, "Torch7: A MATLAB-like environment for machine learning," in *Proc. NIPS Workshop*, 2011, pp. 1–6.

[230] PyTorch. *PyTorch: An Open-Source Deep Learning Framework*. Accessed: Nov. 14, 2024. [Online]. Available: https://pytorch.org/

[231] M. Abadi, "TensorFlow: Large-scale machine learning on heterogeneous distributed systems," 2016, *arXiv:1603.04467*.

[232] A. Gulli and S. Pal, *Deep Learning With Keras*. Birmingham, U.K.: Packt, 2017.

[233] M. Dixit, A. Tiwari, H. Pathak, and R. Astya, "An overview of deep learning architectures, libraries and its applications areas," in *Proc. Int. Conf. Adv. Comput., Commun. Control Netw. (ICACCCN)*, Oct. 2018, pp. 293–297.

[234] X. Meng, "MLlib: Machine learning in Apache spark," *J. Mach. Learn. Res.*, vol. 17, no. 1, pp. 1235–1241, 2016.

[235] G. Chen, A. T. T. Dinh, J. Gao, B. C. Ooi, K.-L. Tan, and S. Wang, "SINGA: Putting deep learning in the hands of multimedia users," in *Proc. 26th ACM Int. Conf. Multimedia*, 2018, pp. 610–618.

[236] A. Paszke, S. Gross, S. Chintala, G. Chanan, E. Yang, Z. DeVito, Z. Lin, A. Desmaison, L. Antiga, and A. Lerer, "Automatic differentiation in pytorch," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 30, 2017, pp. 1–4.

[237] X. Fan, D. Xiao, Q. Li, and R. Gong, "Snow-CLOCs: Camera-LiDAR object candidate fusion for 3D object detection in snowy conditions," *Sensors*, vol. 24, no. 13, p. 4158, Jun. 2024.

[238] T. Chen, M. Li, Y. Li, M. Lin, N. Wang, M. Wang, T. Xiao, B. Xu, C. Zhang, and Z. Zhang, "MXNet: A flexible and efficient machine learning library for heterogeneous distributed systems," 2015, *arXiv:1512.01274*.

[239] NVIDIA. *Neon: Nervana Systems' Deep Learning Framework*. [Online]. Available: https://neon.nervanasys.com/

[240] A. Paszke, S. Gross, S. Chintala, and G. Chanan, "Pytorch: Tensors and dynamic neural networks in Python," *PyTorch, Tensors Dyn. Neural Netw. Python Strong GPU Acceleration*, vol. 6, no. 3, p. 67, 2017.

[241] NVIDIA. *CUDA Toolkit*. Accessed: Nov. 14, 2024. [Online]. Available: https://developer.nvidia.com/cuda-toolkit

[242] NVIDIA. *cuDNN*. Accessed: Nov. 14, 2024. [Online]. Available: https://developer.nvidia.com/cudnn

[243] NVIDIA. *TensorRT*. Accessed: Nov. 14, 2024. [Online]. Available: https://developer.nvidia.com/tensorrt

[244] S. Han, J. Pool, J. Tran, and W. J. Dally, "Learning both weights and connections for efficient neural network," in *Proc. NIPS*, Dec. 2015, pp. 1135–1143.

[245] NVIDIA. *NVIDIA Jetson AGX Xavier*. [Online]. Available: https://developer.nvidia.com/embedded/jetson-agx-xavier

[246] Y. Liu, D. Jiang, C. Xu, Y. Sun, G. Jiang, B. Tao, X. Tong, M. Xu, G. Li, and J. Yun, "Deep learning based 3D target detection for indoor scenes," *Appl. Intell.*, vol. 53, no. 9, pp. 10218–10231, May 2023.

[247] K.-B. Park, S. H. Choi, M. Kim, and J. Y. Lee, "Deep learning-based mobile augmented reality for task assistance using 3D spatial mapping and snapshot-based RGB-D data," *Comput. Ind. Eng.*, vol. 146, Aug. 2020, Art. no. 106585.

[248] X. Yan, J. Hsu, M. Khansari, Y. Bai, A. Pathak, A. Gupta, J. Davidson, and H. Lee, "Learning 6-DOF grasping interaction via deep geometry-aware 3D representations," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2018, pp. 3766–3773.

[249] J. Mahler, M. Matl, V. Satish, M. Danielczuk, B. DeRose, S. McKinley, and K. Goldberg, "Learning ambidextrous robot grasping policies," *Sci. Robot.*, vol. 4, no. 26, Jan. 2019, Art. no. eaau4984.

[250] R. Tesse, "A deep learning approach to instance segmentation of indoor environment," M.S. thesis, Dept. Electron. Telecommun., Politecnico di Torino, Turin, Italy, 2022.

[251] B. Deng, C. R. Qi, M. Najibi, T. Funkhouser, Y. Zhou, and D. Anguelov, "Revisiting 3D object detection from an egocentric perspective," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 34, 2021, pp. 26066–26079.

[252] I. Sa, Z. Ge, F. Dayoub, B. Upcroft, T. Perez, and C. McCool, "DeepFruits: A fruit detection system using deep neural networks," *Sensors*, vol. 16, no. 8, p. 1222, Aug. 2016.

[253] X. Hua, H. Li, J. Zeng, C. Han, T. Chen, L. Tang, and Y. Luo, "A review of target recognition technology for fruit picking robots: From digital image processing to deep learning," *Appl. Sci.*, vol. 13, no. 7, p. 4160, Mar. 2023.

[254] J. Lee and S. Lee, "Construction site safety management: A computer vision and deep learning approach," *Sensors*, vol. 23, no. 2, p. 944, Jan. 2023.
[255] F. Guo, Y. Qian, Y. Wu, Z. Leng, and H. Yu, "Automatic railroad track components inspection using real-time instance segmentation," *Comput.-Aided Civil Infrastruct. Eng.*, vol. 36, no. 3, pp. 362–377, Mar. 2021.
[256] R. Yang and Y. Yu, "Artificial convolutional neural network in object detection and semantic segmentation for medical imaging analysis," *Frontiers Oncol.*, vol. 11, Mar. 2021, Art. no. 638182.
[257] A. A. Shvets, A. Rakhlin, A. A. Kalinin, and V. I. Iglovikov, "Automatic instrument segmentation in robot-assisted surgery using deep learning," in *Proc. 17th IEEE Int. Conf. Mach. Learn. Appl. (ICMLA)*, Dec. 2018, pp. 624–628.
[258] C. Rennie, R. Shome, K. E. Bekris, and A. F. De Souza, "A dataset for improved RGBD-based object detection and pose estimation for warehouse pick-and-place," *IEEE Robot. Autom. Lett.*, vol. 1, no. 2, pp. 1179–1185, Jul. 2016.
[259] A. Zeng, K.-T. Yu, S. Song, D. Suo, E. Walker, A. Rodriguez, and J. Xiao, "Multi-view self-supervised deep learning for 6D pose estimation in the Amazon picking challenge," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2017, pp. 1386.
[260] E. Shreyas, M. H. Sheth, and Mohana, "3D object detection and tracking methods using deep learning for computer vision applications," in *Proc. Int. Conf. Recent Trends Electron., Inf., Commun. Technol. (RTEICT)*, Aug. 2021, pp. 735–738.
[261] V. Getuli, P. Capone, A. Bruttini, and S. Isaac, "BIM-based immersive virtual reality for construction workspace planning: A safety-oriented approach," *Autom. Construct.*, vol. 114, Jun. 2020, Art. no. 103160.
[262] M. Skamantzari and A. Georgopoulos, "3D visualization for virtual museum development," *ISPRS - Int. Arch. Photogramm., Remote Sens. Spatial Inf. Sci.*, vol. 41, pp. 961–968, Jun. 2016.
[263] S. Van Nguyen, S. T. Le, M. K. Tran, and H. M. Tran, "Reconstruction of 3D digital heritage objects for VR and AR applications," *J. Inf. Telecommun.*, vol. 6, no. 3, pp. 254–269, Jul. 2022.

AMBATI PRAVALLIKA received the B.Tech. degree in electronics and communication engineering from the Vaagdevi Institute of Technology and Science, JNTU Anantapur, Peddasettipalli, Proddatur, Andhra Pradesh, in 2009, and the M.Tech. degree in embedded systems from Vignan University, Vadlamudi, Guntur, in 2012. She is currently pursuing the Ph.D. degree with the National Institute of Technology Warangal, under the supervision of Dr. Mohammad Farukh Hashmi. She is an Assistant Professor with the V. N. R. Vignan Jyothi Institute of Engineering and Technology, Hyderabad. She has a teaching experience of nine years. Her primary areas of research interests include object detection, the IoT, machine learning, and deep learning.

MOHAMMAD FARUKH HASHMI (Senior Member, IEEE) received the B.E. degree in electronics and communication engineering from MIT Mandsaur/RGPV Bhopal University, in 2007, the M.E. degree in digital techniques and instrumentation from the Shri Govindram Seksaria Institute of Technology and Science (SGSITS) (Autonomous State Government) Indore/RGPV Bhopal University, in 2010, and the Ph.D. degree from the Visvesvaraya National Institute of Technology (VNIT), Nagpur, in 2015, under the supervision of Dr. Avinash G. Keskar. He is currently an Assistant Professor with the Department of Electronics and Communication Engineering, National Institute of Technology (NIT) Warangal. He has published up to 75 articles, including 25 SCI-indexed research papers in international/national journals/conferences of publishers, such as IEEE, Elsevier, and Springer. He has also published one patent to his credit. He was a Principal Investigator of a research project worth five lakhs funded by the Institute Seed Grant through TEQIP III. He has a teaching and research experience of 13 years. He has supervised two Ph.D. scholars. He is guiding six Ph.D. scholars. His current research interests include computer vision, machine vision, machine learning, deep learning, embedded systems, the Internet of Things, digital signal processing, image processing, and digital IC design. He is a Life Member of IETE, ISTE, and IAENG Societies. He is also serving as an Active and Potential Technical Reviewer for IEEE Access, *IET Image Processing*, *IET Computer Vision*, *Wireless Personal Communications*, *IEEE Systems Journal*, *Sensors* (MDPI), *Electronics* (MDPI), *Diagnostics* (MDPI), *The Visual Computer*, *Applied Soft Computing* (Elsevier), *Color Research & Application*, *The Journal of Supercomputing*, and various other journals, such as Elsevier/Springer/IEEE Transactions publishers of reputed journals.

ADITYA GUPTA received the B.E. degree in electronics and telecommunication engineering from CSVTU University, in 2012, the M.E. degree in signal processing, in 2014, and the Ph.D. degree from the Visvesvaraya National Institute of Technology (VNIT), Nagpur, in 2019, under the supervision of Dr. K. D. Kulat. He is currently a Postdoctoral Fellow with the University of Agder, Norway.