

Review

Not peer-reviewed version

---

# Challenges and Solutions for Autonomous Ground Robot Scene Understanding and Navigation in Unstructured Outdoor Environments: A Review

---

[Liyana Wijayathunga](#) , [Alexander Rassau](#) <sup>\*</sup> , [Douglas Chai](#)

Posted Date: 17 April 2023

doi: 10.20944/preprints202304.0373.v1

Keywords: unstructured environments; mobile robots; robot navigation; perception; scene understanding; path planning; autonomous robots; ground robots





Preprints.org is a free multidiscipline platform providing preprint service that is dedicated to making early versions of research outputs permanently available and citable. Preprints posted at Preprints.org appear in Web of Science, Crossref, Google Scholar, Scilit, Europe PMC.

Copyright: This is an open access article distributed under the Creative Commons Attribution License which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

## Article

# Challenges and Solutions for Autonomous Ground Robot Scene Understanding and Navigation in Unstructured Outdoor Environments: A Review

Liyana Wijayathunga <sup>1</sup>, Alexander Rassau <sup>1\*</sup> and Douglas Chai <sup>1</sup>

<sup>1</sup> School of Engineering, Edith Cowan University, Joondalup WA 6027, Perth, Australia

\* Correspondence: a.rassau@ecu.edu.au

**Abstract:** The capabilities of autonomous mobile robotic systems have been steadily improving due to recent advancements in computer science, engineering, and related disciplines such as cognitive science. In controlled environments, autonomous robots have been able to achieve relatively high levels of autonomy. In more unstructured environments, however, the realisation of autonomous mobile robots remains challenging due to limitations in the robots' external environment understanding. Many autonomous mobile robots use classical, learning-based or hybrid approaches for navigation. The classical navigation approach typically includes robot perception, localisation, environmental mapping, path planning and motion control stages. More recent learning-based methods may replace the complete navigation pipeline or selected stages of the classical approach. For effective deployment, autonomous robots need to be able to understand their external environments at a sophisticated level according to their intended applications. Therefore, in addition to robot perception, scene analysis and higher-level scene understanding (e.g., traversable/non-traversable, and rough or smooth terrain) are required for autonomous robot navigation in unstructured outdoor environments. A wide number of alternative approaches have been proposed in recent years to attempt to address these scene understanding requirements. This paper provides a comprehensive review and critical analysis of these methods in the context of their applications to the problems of robot perception and scene understanding in unstructured environments, and the related problems of localisation, environment mapping and path planning. State-of-the-art sensor fusion methods and multimodal scene understanding approaches are also discussed and evaluated within this context. The paper concludes with an in-depth discussion regarding the current state of the autonomous ground robot navigation challenge in unstructured outdoor environments and the most promising future research directions to overcome these challenges.

**Keywords:** unstructured environments; mobile robots; robot navigation; perception; scene understanding; path planning

## 1. Introduction

The use of autonomous mobile robotic systems is rapidly expanding in many scientific and commercial fields, and the capabilities of these robots have been growing due to continuous research and industrial efforts. The range of different research application areas in autonomous mobile robotics is wide, including areas such as surveillance, space exploration, defence, petrochemical, industrial automation, disaster management, construction, marine, personal assistance, extreme environments, sports entertainment, agriculture, transportation logistics, and many other industrial and non-industrial applications [1]. Many different types of robot platforms have been and continue to be developed for these applications and autonomous mobile robotics is a global and continuously evolving scientific field. Robots that move in contact with ground surfaces are commonly referred to as mobile ground robots, and these robots may be deployed in different working environments, including indoor or outdoor, structured or unstructured, and in proximity to static or dynamic actors. Each of these environments creates various challenges for robot applications. The overall system

configuration of a robot mostly depends on the relevant operating environment, for example, a robot designed for a static environment may not effectively adapt to dynamic situations that arise within the environment [2].

The development of ground robotic systems is particularly challenging when the intended application area of autonomous vehicles/robots falls into the unstructured off-road category, making this a very active area of research. The difficulties mainly arise due to the weak scene understanding of robots in these unstructured environment scenarios. The agriculture industry is one example of a relevant application area to deploy scene understanding based off-road autonomous robotic systems. In recent times, the agriculture industry has shown a growing adoption of robotics technologies, but in general, the involvement of novel technologies depends on economic sustainability. Ground mobile robots and manipulators are already used in precision farming to pick fruits, harvest vegetables and for weeding, but their application areas are generally fairly narrow due to limited scene understanding capability. In the past few years, self-driving vehicles have been gradually growing in technological capability and market size within the automobile industry [3,4]. Despite these developments, autonomous driving is still a very challenging problem, and in complex scenarios the performance level remains below that of an average human operator. This is particularly true when the intended application area of autonomous vehicles/robots includes off-road areas. This low performance mainly arises due to the weak external environment understanding of robots. In general, application areas such as disaster management, environment exploration, defence, mining and transportation are associated with complex and unstructured environments. Therefore, these fields are also driving further research on mobile robot scene understanding, as it relates to the broader topic of autonomous robot navigation.

Several challenges of the classical autonomous robot navigation pipeline remain for current robotic systems under the topics of

- perception,
- localisation and mapping, and
- scene understanding.

In robot perception, robots sense environments using different sensors and extract actionable information via the sensor data. Perception plays an important role in the realisation of autonomous robots. For robots to perform effectively in unstructured outdoor environments in real-time, it is essential that they possess accurate, reliable, and robust perception capabilities. To achieve these characteristics, in general, autonomous robots in complex scenarios are equipped with several sensor modalities (that can be exteroceptive or proprioceptive) [5,6]. Different modalities such as sound, pressure, temperature, light, and contact have been used in robot environmental perception applications [7]. Sensor fusion (combining different sensor modalities) has been applied in many recent autonomous mobile robot/self-driving applications [8]. Multimodal sensor fusion brings the complementary properties of different sensors together to achieve better environment perception across a range of conditions [9]. Many recent deep learning-based sensor fusion methods have shown higher robustness in perception than conventional mono-sensor methodologies [6,10]. Camera, Light Detection and Ranging (LiDAR), radar, ultrasonic, Global Navigation Satellite System (GNSS), inertial measurement unit (IMU) and odometry sensors are used in many mobile ground robot perception applications. The most frequently used robot vision-based sensor fusion methods combine camera images with LiDAR point clouds. Research into sensor fusion techniques remains an important component of achieving better sensing capabilities for unstructured outdoor environments.

In the second stage of the autonomous robot navigation pipeline, the information retrieved from perception sensors is used for robot localisation and to map their external environments. Autonomous mobile robots require accurate and reliable environmental mappings and localisation at a sophisticated level based on the application context. The Simultaneous Localisation and Mapping (SLAM) approach is a commonly used technique in autonomous mobile robot systems to represent the robot positions and the map of their external environments when both the robot pose and environmental map are previously unknown. Many SLAM systems use LiDAR sensors and vision-based sensors such as

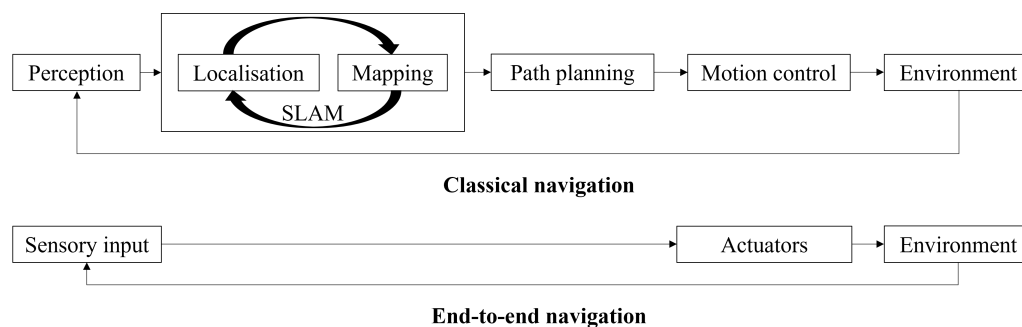
Red-Green-Blue (RGB)/RGB-Depth (RGB-D) cameras that can retrieve visual or proximity information, and the fusion of these sensors has achieved better robustness than using either camera or LiDAR sensors alone [11].

A fundamental aim of robotic vision is to interpret the semantic information present in a scene to provide scene understanding. Scene understanding goes beyond object detection and requires analysis and elaboration of the data retrieved by the sensors [12]. This concept is used in many practical applications such as self-driving vehicles, surveillance, transportation, mobile robot navigation, and activity recognition. Understanding scenes using images or videos is a complex problem, however, and requires more steps than just recording and extracting features [13]. Scene understanding can be aided by taking advantage of multiple sensor modalities, and this is usually termed multimodal scene understanding [14]. One of the requirements for autonomous robots operating in unstructured environments is a capacity to understand the surrounding environment. Scene understanding comprises subtasks such as depth estimation, scene categorisation, object detection, object tracking, and event categorisation [15]. These scene understanding sub-tasks can describe different aspects of a scene acquired by perception sensors. In scene understanding, a representation is given to a scene by carrying out some of the above tasks jointly to get a holistic understanding of the retrieved scene. To generate this overall representation, the information observed from the above-mentioned scene understanding subtasks must be combined meaningfully. Some of the early methods used to obtain scene understanding include using a block world assumption [16], or bottom-up top-down inference [17], and many of these early works have depended on heuristics rather than learning-based methods and, thus, were not suitable for generalisation to unstructured real-world scenarios [15]. The most used recent methods try to acquire information from deep learning-based (i.e., convolutional neural networks (CNNs), graph neural networks, vision transformers) approaches. Scene understanding in unstructured outdoor environments using multimodal scene understanding concepts is a challenging task. Many modern scene understanding methods use feature-based high-level representations of environments. In unstructured environments, however, detection of useful object features is challenging. Therefore, it is difficult to reliably interpret visual information from unstructured or dynamic environments [18]. However, to attain real-world effectiveness, robots should understand their operating environments up to a level that is accurate enough to execute real-time and goal-oriented decisions.

One of the key capabilities required for autonomous robot operation is autonomous path planning for robot navigation. Path planning is generally separated into global and local planning. For global planning, previous knowledge of the operating environment is necessary, and this planning method is also identified as an off-line mode for robot path planning. Robot local path planning, also known as online robot path planning, allows for real-time decisions to be made by the robot in response to perception of the local environment. Autonomous robot local path planning for optimal terrain traversal in unstructured outdoor environments is an important challenge to solve for robots operating in off-road conditions. This is due to the limitations of standard path planning algorithms, which are incapable of performing the desired tasks in dynamic or unstructured environments where the system lacks prior knowledge and/or already existing maps [19]. Artificial potential fields, simulated annealing, fuzzy logic, artificial neural networks, and dynamic window [20] approaches are some of the algorithms that have been used in robot local path planning [21]. An optimal local path planning approach should enable robots to adaptably deal with their environments, such as assisting in avoiding dynamic obstacles or identifying traversable routes through varying terrain conditions. In unstructured environments, classical path planning concepts such as Rapidly exploring Random Tree (RRT) variants, Batch Informed Tree (BIT), D\* algorithm variants, artificial potential field methods, A\* algorithm variants and learning-based path planning methods have been used [22]. Over the past years, deep learning has been used to enhance the performance of sensor fusion [23], multimodal scene understanding [24] and robot local path planning [25] techniques.

For autonomous navigation, a robot must have the ability to understand both its pose and enough about the external environment to determine an optimal and traversable path to safely reach a goal position without human assistance in the robot control loop. However, fully autonomous navigation in unstructured environments has not yet been achieved despite significant advancements in computing and engineering technology. Autonomous vehicle navigation on urban roads has been of great interest due to its emerging commercial applications around the world. As a result, autonomous perception technologies are developing continuously due to the competitiveness of this industry [26]. Compared to the level of development of methods for autonomous navigation in urban settings [27], however, autonomous navigation in off-road scenarios has not been studied to the same extent, meaning there are significant opportunities for new research in this area. A large number of autonomous navigation techniques have been explored and these can be broadly divided into two subsections according to the approach used to execute control commands following the processing of the input sensory data.

As shown in Figure 1, autonomous navigation methods can be categorised as the classical modular pipeline or end-to-end learning-based approaches. The modular pipeline architecture includes intermediate representations that humans can interpret and that can provide information related to the failure modes of the approach. Furthermore, the system development can be parallelised among several expert teams because of its modular nature. Perception, localisation and mapping, path planning and robot control modules may consist of classical, learning-based or hybrid methods [8, 28]. Due to algorithmic and modelling limitations, however, the modular approach may not be optimal for general autonomous navigation applications. Also, machine learning-based modules must be separately trained and validated using auxiliary loss functions, which can create suboptimality across the system as a whole. In contrast, end-to-end learning-based systems learn policies from observations of the outcomes that result from the actions of autonomous systems [16,29]. In these systems, Deep Reinforcement Learning (DRL) is used to refine the control algorithms used to determine autonomous actions. However, commonly employed methods like imitation learning can suffer from overfitting and cause problems with the poor generalisation of the system behaviour when deploying in different environment scenarios [30]. Also, when errors occur in an end-to-end system, it can be complex to investigate because the origin of these errors is hidden in the holistic neural network-based architectures [8,31]. It is clear, however, that future development of autonomous robot navigation will rely on further advancements in robot environmental perception and machine learning.



**Figure 1.** Classical and end-to-end autonomous navigation approaches.

The remainder of this paper is structured in the following manner. A discussion of robot vision and the types of active ranging sensors that are used for robot environment perception is presented in Section 2. This section also discusses deep learning-based camera and LiDAR sensor fusion methods for depth estimation, object detection, and semantic and instance segmentation. Section 3 describes modern mobile ground robot scene understanding techniques and scene representations that are utilised in robot navigation. Robot path planning algorithms at the global and local levels are discussed in Section 4. In Section 5, the details of robot vision and ranging sensors, fusion methods, scene understanding concepts, and local navigation approaches are summarised. Section 6 provides



an overview of the research challenges related to the topic. Additionally, it offers some potential future research directions. Finally, conclusion of the review can be found in Section 7.

## 2. Robot Environment Perception for Navigation

Robot external and internal environment sensing by extraction of raw sensor data and their interpretation is the basic principle of robot perception. In the modular or end-to-end robot navigation approach, sensors play a critical role in capturing the environment or internal robot attributes for robot perception. Therefore, to achieve better perception, a wide range of different sensor modalities have been investigated by researchers. A sensor modality represents a sensor that inputs a particular form of energy and processes the signal using similar methods. Modalities include raw input types for sensors like sound, pressure, light (infrared, visible), or magnetic fields. Robot perception sensor modalities commonly include cameras (infrared, RGB or depth), LiDAR, radar, sonar, GNSS, IMU, and odometry sensors.

### 2.1. Vision and Ranging Sensors

Camera sensors are usually incorporated in vision applications by researchers to retrieve environmental information for the use of mobile robots. However, LiDAR sensors have shown more reliability in low-light environmental conditions than cameras and produce highly accurate depth measurements. LiDAR sensors come with 2D or 3D mapping capability [32], and these sensors can generate high-fidelity point clouds of outdoor environments, although these unstructured point clouds tend to become increasingly sparse as the sensor range is increased.

#### 2.1.1. Vision-based Sensor Types

Vision is crucial for a robot to navigate in unknown environments. Robots are equipped with vision-based sensors like cameras to understand environmental information. Cameras are generally cheaper than LiDAR sensors and are a type of passive sensor (although some depth cameras use active sensing). The monocular configuration is highly used in standard RGB cameras, for example, the GoPro Hero camera series. They are compact passive sensors that consume low power, approximately 5 watts depending on resolution and mode settings. Monocular SLAM has been explored in the research literature due to its simple hardware design [27,33]. However, the algorithms that are used for it are very complex because depth measurements cannot be directly retrieved from static monocular images. Monocular cameras also suffer from pose estimation problems [34]. The pose of the camera is obtained by referencing previous poses, hence errors in pose estimation propagate through the process, and this phenomenon is called scale drift [23,24,35].

Stereo cameras are inspired by human eyes and use two lenses and separate passive image sensors to obtain two perspectives of the same scenes. They have been used in indoor and outdoor SLAM applications. These camera types use the disparity of the two images to calculate the depth information. Stereo camera vision does not suffer from scale drift. Available popular stereo cameras are Bumblebee 2, Bumblebee XB3, Surveyor stereo vision system, Capella, Minoru 3D Webcam, Ensenso N10, ZED 2 and PCI nDepth vision system. Stereo camera power consumption is generally between 2 to 15 Watts. The maximum range of stereo cameras varies from 5 to 40 m at different depth resolution values. The accuracy of these sensors varies from around a few millimetres to 5cm at the maximum range [36]. The cost of these sensors varies from one hundred to several thousand Australian dollars. RGB-D camera sensors consist of monocular or stereo cameras coupled to infrared transmitters and receivers. Microsoft's Kinect camera is a relatively inexpensive RGB-D sensor that provides colour images and depth information of image pixels. The Kinect sensor is mainly used for indoor robot applications due to the saturation of infrared receivers in outdoor scenarios from sunlight [37]. The Kinect sensor has three major versions, Kinect 1, Kinect 2, and Azure Kinect (the latest version). Kinect 1 uses structured light for depth measurement, and the other models use time-of-flight (ToF) as the depth measuring principle. The newest Kinect model, Azure Kinect also generates substantial noise in outdoor bright

light conditions with a practical measuring range below 1.5 metres [38]. In general, RGB-D sensors utilise three depth measurement principles, structured light, ToF, and active infrared stereo techniques. The structured light RGB-D sensors underperform compared to ToF techniques in measuring the range of distant objects. The structured light technique is vulnerable to multi-device interference. The ToF methods suffer from multi-path interference and motion artefacts. The active infrared stereo principle has drawbacks due to common stereo matching problems such as occluded pixels and flying pixels near contour edges [39]. Active infrared stereo cameras are identified as an extension of passive stereo cameras. They offer more reliable performance in indoor and some outdoor scenarios. However, they require high computation capability. Table 1 shows the main depth measurement methods that are used in RGB-D sensors.

Table 1. Depth sensor modalities.

Technique	Typical Sensors	Advantages	Disadvantages
Structured light	Kinect v1, Xtion PROLive, RealSense SR300 and F200	High accuracy and precision in indoor environments	Limited range, not suitable for outdoor environment due to noise from ambient light, interference from reflections and other light sources
ToF	Kinect v2	Good for indoor outdoor applications, long range, robust to illumination changes	Lower image resolution than structured light cameras, high power consumption, cost varies with resolution, rain fog can affect sensor performance
Active infrared stereo	RealSense R200, RealSense D435, D435i	Compact, lightweight, dense depth images	Stereo matching requires high processing power, struggle at high occlusions and featureless environments, relatively low range especially outdoors

Event cameras are asynchronous sensors that use different visual information acquisition principles than the standard frame-based image sensors. This camera type samples light based on the changes in the scene dynamics (asynchronously measuring brightness per pixel). These cameras currently cost several thousands of dollars. Event cameras have advantages like very high temporal resolution, high dynamic range, low latency (in microseconds), and lower power consumption than standard cameras. However, this camera type is not suitable for the detection of static objects. The main burden is the requirement of new methods (algorithms and hardware, e.g., neuromorphic based approaches or spiking neural networks) to process event camera outputs to acquire data and information because traditional image processing methods are not directly applicable [40].

Omnidirectional cameras are utilised in robotic applications where more information is needed about surrounding environments. These cameras provide a wider-angle view than conventional cameras, which typically have a limited field of view. Due to the lens configuration of omnidirectional cameras, the obtained images have distortions. Therefore, these cameras require different mathematical models, like the unified projection model, to correct image distortions. A summary of these different camera types and their advantages and disadvantages is shown in Table 2.

Table 2. Camera configurations.

Configuration	Advantages	Disadvantages
Monocular	Compactness, low hardware requirements	No direct depth measurements
Stereo	Depth measurements, low occlusions	Fails in featureless environments, CPU intensive, accuracy /range depends on camera quality
RGB-D	Color and depth information per pixel	Limited range, reflection problems on transparent, shiny, or very matte and absorbing objects
Event	High temporal resolution, suitable for changing light intensities, low latency [41]	No direct depth information, costly, not suitable for static scenes, requires non-traditional algorithms
Omni-directional	Wide angle view (alternative to rotating cameras)	Lower resolution, need special methods to compensate for image distortions

2.1.2. Active Ranging Sensors

Active sensors emit energy to the environment and measure the return signal from the environment. There are several types of active-ranging sensors such as reflectivity, ultrasonic, laser rangefinder (e.g., LiDAR), optical triangulation (1D), and structured light (2D). The commonly used LiDAR sensor is an active-ranging sensor that shows improved performance compared to ultrasonic ToF sensors for perception in autonomous navigation systems [42]. LiDAR imaging is one of the most studied topics in the optoelectronics field. The ToF measurement principle is used by LiDAR sensors for obtaining depth measurements in different environments. Rotating LiDAR imagers were the first type to successfully achieve acceptable performance using a rotating-mirror mechanical configuration with multiple stacked laser detectors [43]. New trends in LiDAR technology are in the development of low-cost, compact solid-state commercial LiDAR sensors [44,45]. The three most widely used LiDAR techniques utilise pulsed, amplitude-modulated, and frequency-modulated laser beams. The most common commercially available method is to use a pulsed laser beam, which directly measures the time taken by the pulsed signal to return to the sensor. In such sensors, time measurements require resolutions in picoseconds (high-speed photon detectors). Therefore, the cost of pulsed LiDAR sensors is comparably higher than the other two methods for equivalent range and resolution [46]. The range of a pulsed LiDAR is limited by the Signal-to-Noise Ratio (SNR) of the sensor. Pulsed LiDAR sensors are suitable in indoor and outdoor environments due to their instantaneous higher peak pulse contrast to ambient irradiance noise. However, Amplitude Modulated Continuous Wave (AMCW) LiDAR sensors with the same average signal power have lower continuous wave peaks and hence are vulnerable to solar irradiance. AMCW LiDAR sensors are popular in indoor robot applications due to low SNR. In general, there is no significant depth resolution difference between pulsed LiDAR and AMCW LiDAR, but pulsed LiDAR may outperform AMCW LiDAR accuracy at the same optical power levels because of higher SNR. Increasing the modulation frequency of AMCW LiDAR sensors improves depth resolution but reduces ambiguity distance. Thus, pulsed sensors have higher ranges compared to AMCW sensors. AMCW sensors also have slow frame rates relative to pulsed sensors and usually underperform in dynamic environments. Frequency Modulated Continuous Wave (FMCW) sensors have higher depth resolution in comparison to pulsed and AMCW LiDAR sensors. These sensors can measure the depth and velocity of targets simultaneously, a highly advantageous feature for the autonomous vehicle industry. These sensors can also avoid interference from other LiDAR sensors because of the frequency modulation. FMCW sensors, however, require accurate frequency sweeps to generate emitter signals, which is a challenging task. FMCW sensor technology has been in continuous



development but has not yet established itself at the commercial level. A summary of these three LiDAR technologies is provided in Table 3.

Table 3. LiDAR sensor types.

Configuration	Advantages	Disadvantages
Pulsed	High frame rate	Low depth resolution, higher inference from other LiDAR sensors
AMCW	Not limited by low SNRs, however, not effective at very low SNRs	Low accuracy than FMCW, lower depth resolution than FMCW
FMCW	Velocity and range detection in a single shot, higher accuracy than AMCW, higher depth resolution	Currently at the research and development stage

The concept of robotic perception pertains to various robotics applications that utilise sensory information and modern deep learning methods. These applications include identifying objects, creating representations of the environment, comprehending scenes, detecting humans/pedestrians, recognising activities, categorising locations based on meaning, scene reconstruction, and others. Towards the goal of fully autonomous robot navigation, robust and accurate environmental perception is a necessity. Passive RGB cameras are relatively low cost and can capture rich environment details, however, their perception abilities are vulnerable to environmental illumination changes and occlusions. Therefore, sensor fusion methods are important to gain robust perception.

2.2. LiDAR and Camera Data Fusion

Achieving a reliable real-time understanding of external 3D environments is very important for safe robot navigation. Visual perception using cameras is commonly employed in many mobile robotic systems. Camera images can be efficiently and often effectively processed by CNN-based deep learning architectures. However, relying on one sensor can lead to robustness challenges particularly in applications like self-driving vehicles, autonomous field robots, etc. Therefore, different sensor modalities are often combined to achieve better reliability and robustness for autonomous systems. The fusion of LiDAR and camera data is one of the most investigated sensor fusion areas in the multimodal perception literature [47]. Camera and LiDAR fusion has been applied in various engineering fields and has shown better robustness than camera-only robot navigation approaches [48]. This fusion strategy is more effective and popular than other sensor fusion approaches such as radar-camera, LiDAR-radar, and LiDAR-thermal camera. Still, the technical challenges and cost of sensors and processing power requirements have constrained the application of these methods in more general human activities. Recent deep-learning algorithms have significantly improved the performance of camera-LiDAR fusion methods [49], with monocular and stereo cameras mainly used with LiDAR sensors to fuse images and point cloud data [50].

Deep learning-based LiDAR and camera sensor fusion methods have been applied in depth completion, object detection, and semantic and instance segmentation. Image and point cloud scene representations include volumetric, 2D multi-view projection, graph-based and point-based. In general, most of the early methods fuse LiDAR and camera image semantics using 2D CNNs. Many 2D-based networks project LiDAR points onto respective image planes to process feature maps through 2D convolutions [51–54]. Several works have used point cloud processing techniques such as PointNet [55] to extract features or 3D convolutions [56] to detect objects in volumetric representations [57]. Some other LiDAR and image fusion methods use 2D LiDAR representations for feature fusion and then cluster and segment 3D LiDAR points to generate 3D region proposals [58]. Voxel-based representations and multi-view camera-LiDAR fusion approaches are utilised to generate 3D proposals in object detection. State-of-the-art camera-LiDAR semantic segmentation methods

employ feature fusion methods to obtain 2D and 3D voxel-based segmentation results. Multi-view approaches map RGB camera images onto the LiDAR Bird's-Eye-View (BEV) plane to align respective features from the RGB image plane to the BEV plane [59–62], and several other methods propose to combine LiDAR BEV features with RGB image features directly [47,50]. These direct mapping methods use trained CNNs to align image features with LiDAR BEV features from different viewpoints.

Computer vision has been a rapidly growing field in the past decade, and the development of machine learning methods has only accelerated this. Recently, deep learning strategies have influenced the rapid advancement of various computer vision algorithms. Computer vision includes subtopics like object detection, depth estimation, semantic segmentation, instance segmentation, scene reconstruction, motion estimation, object tracking, scene understanding and end-to-end learning [37]. Computer vision methods have been applied to a great extent in emerging autonomous navigation applications. However, these vision techniques may be less effective in previously unseen or complex environments and highly rely on the trained domain. Therefore, continuous improvements are being made towards the development of fully autonomous systems. Several state-of-the-art benchmarking datasets have been utilised to compare the performance of different autonomous driving vision methods. KITTI [63], Waymo [64], A2D2 [65], nuScenes [66], and Cityscapes [67] are some examples of these state-of-the-art autonomous driving datasets.

### 2.2.1. Dense Depth Prediction

Dense depth completion is a technique that estimates dense depth images from sparse depth measurements. Achieving depth perception is of significant importance in many different engineering industries and research applications such as autonomous robotics, self-driving vehicles, augmented reality, and 3D map construction. LiDAR sensors, monocular cameras, stereo cameras, and RGB-D cameras have been the most utilised in dense depth estimation applications, but these sensors have specific limitations in the estimation process. LiDAR sensors with high accuracies are costly to use in large-scale applications. Three main challenges have been identified in LiDAR depth completion [68]. The first relates to the fact that, in general, even expensive LiDAR sensors produce sparse measurements for distant objects. Also, LiDAR points are irregularly spaced compared to monocular RGB images. Therefore, it is non-trivial to increase the depth prediction accuracy using the corresponding colour image. Secondly, there are difficulties in combining multiple sensor modalities. The third challenge is that depth estimation using deep learning-based approaches has limitations with regards to availability of pixel-level ground truth depth labels for training networks. Another possible approach to depth estimation is by using stereo cameras. Stereo cameras, however, require accurate calibration, demand high computational requirements, and fail in featureless or uniformly patterned environments. RGB-D cameras are capable of depth sensing but have a limited measuring range and poor performance in outdoor environments. A technique called depth inpainting can be used as a depth completion method for structured light sensors like the Microsoft Kinect 1.0, and these sensors produce relatively dense depth measurements, but are generally only usable in indoor environments. Dense depth estimation techniques generally up-sample sparse and irregular LiDAR depth measurements to dense and regular depth predictions. Depth completion methods, however, still have a variety of problems that need to be overcome. These challenges are primarily sensor-dependent, and solutions should overcome respective difficulties at the algorithm development stage.

Many state-of-the-art dense depth prediction networks combine relatively dense depth measurements or sparse depth maps with RGB images to assist the prediction process. In general, retrieving dense depth detail from relatively dense depth measurements is easier than from sparse depth maps. In relatively dense depth images, a higher percentage of pixels (typically over 80%) have observed depth values. Therefore, in similar scenarios, predicting dense depth is relatively less complex. However, in autonomous navigation applications, 3D LiDAR sensors account for only approximately 4% of pixels when depth measurements are mapped onto the camera image space, which creates challenges in generating reliable dense depth images [68].

### 2.2.2. Dense Depth from Monocular Camera and LiDAR Fusion

Depth estimation based solely on monocular images is not reliable or robust. Therefore, to address these monocular camera limitations, the LiDAR-monocular camera fusion-based depth estimation process has been proposed by researchers. Using monocular RGB images and sparse LiDAR depth maps, a residual network learning-based autoencoder decoder network was introduced by [69] to estimate dense maps. However, this method needs a ground truth depth image when retrieving sparse depth images during the network training process. In practice, obtaining such ground truth images is not simple or easily scalable [68]. To mitigate the requirement of a ground truth depth image, [68] presented a self-supervised model-based network that only requires a monocular RGB image sequence and LiDAR sparse depth images in the network training step. This network consists of a deep regression model to identify a one-to-one transformation from a sparse LiDAR depth map to a dense map. This method achieved state-of-the-art performance on the KITTI dataset and considers the pixel-level depth estimation as a deep regression problem in machine learning. LiDAR sparse depth maps use per-pixel depth, and pixels without measured depth are set to zero. The proposed network follows an encoder-decoder architecture. The encoder has a sequence of convolutions, and the decoder has a set of transposed convolutions to up sample feature spatial resolutions. Convolved sparse depth data and colour images are concatenated into a single tensor and input to residual blocks of ResNet-34 [70]. The self-supervised training framework requires only colour/intensity images from monocular cameras and sparse depth images from LiDAR sensors. In the network training step, a separate RGB supervision signal is used (a nearby frame). LiDAR sparse depth can be used as a supervision signal. However, this framework requires a static environment to be able to warp the second RGB frame to the first one.

With this implementation, the root-mean-square of the depth prediction error reduces as a power function with the increment of the resolution of the LiDAR sensor (i.e., the number of scan lines). One of the limitations of this approach is that the observed environment needs to be stationary. If not, the network will not generate accurate results. Large moving objects and surfaces that have specular reflectance can cause the process of training the network to fail. These reasons reduce the applicability of this method in dynamic situations that are often present in outdoor environments. Also, this network training process may get stuck in the local minimums of the photometric loss function due to improper network weight initialisation. This effect may result in output depth images which are not close enough to the ground truth because of the erroneous training process.

In [71], a real-time sparse-to-dense map is constructed by using a convolutional spatial propagation network (CSPN). The propagation process preserves LiDAR sparse depth input values in the final depth map. This network aims to extract the affinity matrix for respective images. The introduced method learns affinity matrix by using a deep convolution neural network. The training process of the network model is achieved by incorporating a stochastic gradient descent optimiser. This network implementation showed memory paging cost as a dominant factor when larger images were fed into the PyTorch-based network. The CSPN network has shown good performance in real-time and thus is well-suited for applications such as robotics and autonomous driving. CSPN++ [72] is an improved version of the CSPN network with adaptively learning convolutional kernel sizes and numbers of iterations for propagation. The network training experiments were carried out using four NVIDIA Tesla P40 Graphic Processing Unit (GPU)s on the KITTI dataset. This research shows that hyper-parameter learning from weighted assembling can lead to significant accuracy improvements, and weighted selection could reduce the computational resource with the same or better accuracy compared to the CSPN network.

### 2.2.3. Dense Depth from Stereo Camera and LiDAR Fusion

Estimating depth using stereo cameras provides more reliable results compared to monocular cameras. LiDAR sensors can produce depth measurements with improved accuracy over increased ranges and varying lighting conditions. The fusion of LiDAR and stereo camera sensors produces more

accurate 3D mappings of environments than LiDAR-monocular camera depth completion. However, stereo cameras commonly have shorter detection ranges, and depth estimation becomes challenging in texture-less environments and high occlusion scenarios. One of the significant works in LiDAR-stereo camera fusion is presented in [73]. This paper presents the first unsupervised LiDAR-stereo fusion network. The network does not require dense ground truth maps, and training is done in an end-to-end manner that shows a broad generalisation capability in various real-world scenarios. The sparsity of LiDAR depth measurements can vary in real-world applications. One advantage of the network proposed in this work is that it handles a high range of sparsity up to the point where the LiDAR sensor has no depth measurements. A feedback loop has been incorporated into the network to connect outputs with inputs to compensate for noisy LiDAR depth measurements and misalignments between the LiDAR and stereo sensors. This network is currently regarded as one of the state-of-the-art methods for LiDAR-stereo camera fusion.

#### 2.2.4. Multimodal Object Detection

Reliable object detection is a vital part of the autonomous navigation of robots/vehicles. Object detection in autonomous navigation is described as identifying and locating various objects in an environment scene in the form of bounding boxes, including dynamic objects and static objects. Object detection may become difficult due to sensor accuracy, lighting conditions, occlusions, shadows, reflections, etc. One major challenge in object detection is occlusion, which consists of different types. The main occlusion types are self-occlusion, inter-object occlusion, and object-to-background occlusion [74]. Early image-based object detection algorithms commonly include two steps. The first stage is dividing the image into multiple smaller sections. Then, these sections are conveyed into an image classifier to identify whether the image section contains an object or not. If any object is detected in an image section, the respective portion of the original image is marked with the relevant object label. The sliding window approach is one way of achieving the above-mentioned first step [75].

A different set of algorithms uses a technique in contrast to the sliding window approach, by grouping similar pixels of an image to form a region. These regions are then fed to a classifier to identify semantic classes (with grouping is done using image segmentation methods). Further improved image segmentation can be achieved by using the selective search algorithm [76]. The selective search algorithm emphasises a hierarchical grouping-based segmentation algorithm. In this method, initially detected image regions are merged in a stepwise manner by selecting the most similar segments until the whole image represents a single region. These regions resulting from each step are added to the region proposals and fed to a classifier. The classifier performance depends on the used region proposal method. This object detection approach does not produce real-time performance suitable for autonomous navigation applications. However, advances such as SSPnet [77], Fast Regional-based Convolutional Neural Networks (R-CNNs) [78], and Faster R-CNN [79] were introduced to address this issue. The Faster R-CNN network generates a feature map by utilising the CNN layer output, and the region proposal generation is achieved by sliding a window (comprising three different aspect ratios and sizes) over the feature map. Each sliding window is mapped to a vector and fed to two parallel classification and regression networks. The classification network calculates the probability of region proposals containing objects, and the regression network indicates the coordinates of each of the proposals.

Object detection research has been mainly employed in the autonomous vehicle industry (for vehicle and pedestrian detection [80]) and mobile robotics. In contrast to camera-only object detection, sensor fusion has been implemented in different real-world applications to obtain more accurate and robust detection results. As previously discussed, LiDAR and camera sensor fusion are one of the most used and highest performing sensor fusion methods. LiDAR and camera sensor fusion object detection approaches consist of two main techniques. These are the sequential and one-step models [50]. Sequential models use 2D proposal-based methods or direct 3D proposals to detect objects. In the sequential approach, 2D/3D regions are proposed in the first stage then 3D bounding box regression

is done in the second stage. The 2D/3D region proposal stage incorporates fused image-point cloud regions that may contain objects. In the bounding box regression stage, feature extraction from region proposals and bounding box prediction is done. One-step models generate region proposals and achieve bounding box regression in parallel in a single step. The 2D proposal-based sequential approach uses 2D image semantics to generate a 2D proposal and point cloud processing methods to detect dynamic objects. This approach utilises already developed image processing models to identify 2D object proposals and then project these proposals to LiDAR 3D point cloud space for object detection.

Two approaches are mainly used to manipulate image-based 2D object proposals and irregular 3D LiDAR data. In the first method, image-based 2D bounding boxes are projected to the LiDAR 3D point cloud to implement 3D point cloud object detection algorithms. The second approach utilises point cloud projections on the 2D images and applies 2D semantic segmentation techniques to achieve point-wise semantic labels of the points within the semantic regions [50]. LiDAR-camera 2D sequential object detection methods include result, feature, and multi-level fusion strategies. 2D proposal-based result level fusion methods incorporate image object detection algorithms to retrieve 2D region proposals. These retrieved 2D object bounding boxes are then mapped onto 3D LiDAR point clouds. The enclosing points in frustum proposals are transferred into a point cloud-based 3D object detection algorithm [53]. The overall performance of this object detection approach depends on the modular behaviour of the 2D detection architecture. The sequential fusion may lose complementary data in LiDAR point clouds due to initial 2D image object proposal detection. 2D proposal-based feature-level fusion uses a mapping from 3D LiDAR point clouds onto the respective 2D images and employs image-based techniques for image feature detection and object detection. One of these approaches appends per-point 2D semantic details as additional channels of LiDAR 3D point clouds and uses an existing LiDAR-based object detection method [81]. However, this approach is not optimal for identifying objects in a three-dimensional world because 3D details in the point clouds may be lost due to the projection.

Multi-level fusion combines 2D result-level fusion with feature-level fusion. This approach uses already available 2D object detectors and generates 2D bounding boxes. Then, points within these bounding boxes are detected. Subsequently, image and point cloud features are combined within the bounding boxes to estimate 3D objects. LiDAR and camera object detection using 3D proposal-based sequential models avoid 2D to 3D proposal transformations and directly generate 3D proposals from 2D or 3D data. This technique consists of two approaches, namely multi-view and voxel-based. MV3D [62] is a multi-view object detection network that uses LiDAR and camera data to predict the full 3D envelope of objects in the 3D environment. A deep fusion scheme was proposed to fuse features from multiple views in respective regions. The detection network comprises two networks, the 3D proposal network and the region-based fusion network. As the inputs, the LiDAR BEV, LiDAR front view and the RGB camera image are fed to the network. The LiDAR BEV is fed to the 3D proposal network to retrieve 3D box proposals. These proposals are used to extract features from the LiDAR front-view and camera RGB image inputs. Then, using these extracted features, the deep fusion network predicts object size, orientation, and location in the 3D space. The network was built on the 16-layer VGGnet [82] and the KITTI dataset was used for the training process.

One of the drawbacks of the multi-level fusion method is the loss of small objects in the detection stage due to feature map down-sampling. Combining image and point cloud feature maps by RoI (Regions of Interest)-pooling decreases the fine-grained spatial details. MVX-net [61] introduces a method to fuse point cloud and image data voxel-wise or point-wise. 2D CNNs are used for the image feature extraction process, and a VoxelNet [83] based network detects 3D objects in the voxel representation. The input 3D LiDAR point cloud is mapped to the 2D image for image feature extraction in the point-wise fusion method, and then voxelisation and processing are done using VoxelNet. In voxel-wise fusion, the point cloud is firstly voxelised and then projected onto the image-based 2D feature representation to extract features. This sequential approach achieved



state-of-the-art performance for 3D object detection at the time of its publication. Object detection utilising one-stage models performs object proposal retrieval and bounding box prediction in a single process. These detection models are suitable for real-time autonomous robot decision-making scenarios. State-of-the-art single-stage object detection methods like [84] simultaneously process depth images and RGB images to fuse points with image features, and then the generated feature map is used for bounding box prediction. In [84], two CNN-based networks process point cloud and RGB front-view images in parallel. One CNN identifies LiDAR features, and the other CNN extracts RGB image features. Then these RGB image features are mapped into the LiDAR range view. Finally, mapped RGB and LiDAR image features are concatenated and fed into LaserNet [85] for object detection and semantic segmentation. This network has been trained in an end-to-end manner. The network training was done for 300k iterations with a batch size of 128, distributed over 32 GPUs. The image fusion, object detection and semantic segmentation process took 38 milliseconds on an Nvidia Titan Xp GPU.

### 2.2.5. Multimodal Semantic Segmentation

Scene semantic segmentation assigns a semantic category label to each pixel in a scene image, and it can be regarded as a refinement of object detection [86]. A scene can incorporate obstacles, free space, and living creatures (not necessarily limited to these categories). The complete semantic segmentation of images applies these semantic categories to all pixels across an image. Many recent computer vision methods rely on CNN architectures. These networks favour dense depth image data over sparse data. In general, LiDAR sensors produce irregular sparse depth measurements. Reference [61] introduced a technique to utilise LiDAR sparse data and RGB images to achieve depth completion and semantic segmentation in a 2D view. This network can work with sparse depth measurement densities as low as 0.8%, and at the time of its publication, this method showed state-of-the-art performance on the KITTI benchmark dataset. The base network of this prior work was adopted from NASNet [87], which has an encoder and decoder architecture. Using LiDAR and RGB image feature fusion, [88] proposed a novel method to achieve 2D semantic segmentation in 2D images. This method introduced a self-supervised network that suits different object categories, geometric locations, and environmental contexts. This self-supervised network uses two sensor modality-specific encoder streams, which concatenate to a single intermediate encoder and then connect into a decoder to fuse the complementary features. The segmentation part is achieved with a network termed AdapNet++ [88] that consists of an encoder-decoder architecture. All these network models were implemented using the deep learning TensorFlow library. Another high-performing deep learning-based LiDAR-Camera 2D semantic segmentation method was presented in [89]. In this method, the generated 3D LiDAR point data is mapped to the 2D image and up-sampled to retrieve a 2D image set that consists of spatial information. Then, fully convolutional networks are used to segment the image using three approaches, signal level, feature, and cross-fusion. In the cross-fusion method, the network was designed to learn directly from the input data.

The techniques discussed up to now have been 2D semantic segmentation methods. In contrast to 2D methods, 3D semantic segmentation approaches provide a realistic 3D inference of environments. An early approach for a 3D scene semantic segmentation network is presented in [90] termed 3DMV. This method requires relatively dense depth scans along with RGB images, and it was developed to map indoor scenarios. Voxelised 3D geometries are built by using LiDAR depth scans. Two-dimensional feature maps are extracted from the RGB images using CNNs, and these image feature maps are mapped in a voxel-wise manner with the 3D grid geometry. This fused 3D geometry is then fed into 3D CNNs to obtain a per-voxel semantic prediction. The overall performance of the approach depends on the voxel resolution, and real-time processing is challenging for higher voxel resolutions. Therefore, this dense volumetric grid becomes impractical for high resolutions. The system was implemented using PyTorch and utilised 2D and 3D convolution layers already provided by the application programming interface. Semantic segmentation of point clouds is challenging for structureless and featureless regions [91]. A point-based 3D semantic segmentation framework has been introduced by [91]. This

approach effectively optimises geometric construction and pixel-level features of outdoor scenes. The network projects features of detected RGB images into LiDAR space and learns 2D surface texture and 3D geometric attributes. These multi-viewpoint features are extracted by implementing a semantic segmentation network and then fused point-wise in the point cloud. Then, this point data is passed to a PointNet++ [92] based network to identify per-point semantic label predictions. A similar approach was followed by Multi-view Point-Net [93] to fuse RGB semantics with 3D geometry to get per LiDAR point semantic labels.

Instead of localised or point cloud representations, [94] have used a high dimensional lattice representation for LiDAR and camera data processing. This representation reduces memory usage and computational cost by utilising bilateral convolutional layers. Then, these layers employ convolutions for unoccupied sections in the generated lattice representation. Firstly, the identified features of point clouds are mapped to a high-dimensional lattice, and then convolutions are used. Following this, CNNs are applied to detect image features from multi-view images, and these features are projected to the lattice representation to combine with three-dimensional lattice features. The generated lattice feature map was assessed by using 3D CNNs to get point-wise labels. A spatial-aware and hierarchical learning strategy has been incorporated to learn 3D features. The introduced network was capable of training in an end-to-end manner.

#### 2.2.6. Multimodal Instance Segmentation

Instance segmentation identifies individual instances within a semantic category. It is considered a more advanced semantic segmentation method. This method not only provides per-pixel semantic categories but also distinguishes object instances, which is more advantageous for robot scene understanding. However, instance segmentation in autonomous navigation introduces more challenges than semantic segmentation. Instance segmentation approaches based on fused LiDAR-camera sensor data show proposal-based and proposal-free architectures. A voxel-wise 3D instance segmentation approach was introduced by [95] that consists of two-stage 3D CNN networks. A feature map was extracted from the low-resolution voxel grid by implementing 3D CNNs. Another feature map was obtained from the RGB multi-view images using 2D CNNs and projected onto the associated voxels in the volumetric grid to append with respective 3D geometry features. Then, object classes and 3D bounding boxes are predicted by feeding these fused features to a 3D CNN architecture. In the second phase, another 3D CNN estimates the per-pixel object instances using already identified features, object classes and bounding boxes.

These voxel-based segmentation methods are constrained by the voxel resolution and require increased computation capabilities with higher grid resolutions. The application of instance segmentation in LiDAR-camera fusion for real-time systems is challenging. Some research studies had limitations, such as the system developed in [96], which does not support dynamic environments. A proposal-free deep learning framework that jointly realises 3D semantic and instance segmentation is presented in [97]. This method performs 3D instance segmentation in the BEV of point clouds. However, this approach is less effective in identifying vertically oriented objects because of the BEV segmentation approach. This method first extracts a 2D semantic and instance map from a 2D BEV representation of the observed point cloud. Then, using the mean shift algorithm [98] and semantic features of the 2D BEV, instance segmentation is achieved by propagating 3D features onto the point cloud. It should be noted that the instance segmentation approaches discussed have been developed to segment 3D point clouds from static indoor environments, and these methods have not shown any segmentation capabilities in dynamic environments.

Overall, while significant progress has been made in the perception capabilities of autonomous robots, particularly in regards to object detection and scene segmentation, many of the existing approaches have only been tested in relatively structured indoor environments, and substantial additional work may be required to adapt these techniques to be used in unstructured outdoor

environments. Nonetheless, some very useful research directions have been identified that have the potential to significantly advance this field.

### 3. Robot Scene Understanding for Navigation Planning

For effective navigation at local and global scales, robots build maps to represent their external environments and to assist with making safe and accurate decisions at local levels. SLAM refers to the real-time combination of robot localisation and external environment mapping. SLAM is highly focused on mapping techniques. Early mobile ground robot map-building approaches comprised feature-based and photometric error-based methods. Robots could efficiently localise their positions in sparse or dense maps by implementing these mapping techniques. However, autonomous robots require more than basic 2D or 3D maps for effective task planning. Therefore, semantic SLAM techniques have been developed and investigated. Topological maps represent the environment as abstract graphs in contrast to classical metric maps. Classical metric maps are used in robotics to visualise the environment as a two-dimensional grid of cells, and each of these cells represents a specific location or area. The occupancy of a cell is typically represented by a binary value. However, these maps have limitations, such as their assumption of a flat and static environment and their inability to illustrate sensor data uncertainty.

To overcome these limitations, researchers have developed probabilistic mapping techniques, such as occupancy grid mapping, which use probabilities to represent occupancy and can handle more complex and dynamic environments. The metric maps include details such as visited locations by a robot, proximity to features and general arrangement of external environments. One of the challenges in classical 3D geometric mapping is the accumulation of position errors over time at the global map level. Topological maps are not directly affected by these error-driven shifts in the geometrical maps, which means that topological mapping is more adaptable to different environmental contexts than geometric mapping. Classical and topological mapping methods have their strengths and weaknesses. Hybrid methods combine metric and topological mapping techniques to maximise each method's benefits. Many hybrid representations have been utilised by indoor robots to generate environmental mappings. Hybrid mapping representations from recent research works generally consist of two or more layers in a hierarchy (e.g., places, categories, ontology). The research work presented in [99] uses depth measurements, semantic segmentation and scene flow to map a 3D environment. This concept can detect the dynamic behaviour of objects. Neural implicit scalable encoding for SLAM [100] introduces a SLAM framework that utilises a hierarchical feature representation. It uses a hierarchical feature grid with encoder-decoder networks to predict occupancies. The geometric maps of indoor environments are developed by using extracted coarse, mid, and fine-level feature grids. The middle and fine-level feature grids represent observed scene geometry. The feature grid in the coarse level detects indoor geometries such as the floor and walls. The coarse level occupancy is used in the scene reconstruction of unobserved environment regions. In the reconstruction process, coarse-level occupancy is optimised in the mid-level feature grid and refined at the fine level.

Kimera [101] presents a dynamic scene graph that consults a five-layer hierarchical representation. The method combines a metric semantic mesh with a topological spatial representation. It is capable of segmenting indoor structures and is robust in overcrowded environments. The five layers of the graph include metric-semantic mesh, objects and agents, places and structures, rooms, and buildings. To effectively implement these hybrid hierarchical maps in real-world applications, the representations need to be updated as the environment changes and should accurately map the interconnections between environment attributes. These hierarchical mapping techniques were implemented in indoor environments where semantic information extraction is less complex, however, and have not yet been applied to unstructured outdoor environments. Kimera-multi [102] is a state-of-the-art metric semantic SLAM system for multi-robot systems. The method is an extension of Kimera [101], and it was tested in computer simulated environments based on SLAM datasets, and outdoor datasets collected by ground robots. This concept creates a 3D mesh that labels outdoor environments using

semantic labels to gain high-level scene understanding. Kimera-multi outperforms the accuracy of Kimera [101] visual-inertial odometry SLAM. Overall, Robot semantic understanding requires the classification of both objects and places. To extract these predefined object models, conditional random field models, CNN-based scene and place classification algorithms, and scene graphs (e.g., robot scene graphs [103], dynamic scene graphs [104]) are utilised.

SLAM techniques can produce sparse or dense geometric or semantic maps of outdoor environments. However, these maps may not have adequate detail for safe local robot navigation where terrains consist of abrupt deviations and vegetation. Therefore, deep learning and image-based scene understanding are necessary directions to identify terrain properties and suitable environment regions for robot navigation in challenging outdoor unstructured environments. Recent robot navigation trends have emerged to segment camera egocentric images using computer vision approaches such as visual attention mechanisms [105]. [106] presents a camera-only approach for segmenting egocentric images to assist robot navigation. The segmentation procedure consists of soft labelling the images according to three levels of driveability. For every pixel, soft labels are mapped by assessing original semantic classes. The SegNet [107] based deep convolutional encoder-decoder architecture was used for the pixel classification. Another paper [108] introduces an RGB-based method to classify terrains into navigability levels by extracting multi-scale features. This work achieved state-of-the-art performance on the RUGD [109] and RELLIS-3D [110] datasets at the time of publication. A multi-head self-attention network architecture was used to classify the terrain into smooth, rough, bumpy, forbidden, obstacle, and background regions. This network requires further improvements to identify drastic terrain geometry changes (i.e., slopes) to gain higher accuracy in navigability segmentation.

In general, 2D scene understanding can lose essential details from the mapping process of 2D data to the 3D real world. Therefore, 3D scene understanding-based methods are used with the most recent robot navigation applications. [111] presented a 3D semantic segmentation method using a CNN encoder-decoder architecture. This approach inputs RGB images and depth images for the fusion of feature maps, with the FuseNet [112] architecture taken as the base for the CNN. Visual dynamic object-aware SLAM [113] combines SLAM with dynamic modelling of scenes, enabling robots to operate effectively in dynamic environments. This work introduces instance segmentation of dynamic objects with the camera trajectory. The model can identify and track stationary structures and dynamic objects, and integrate them into the traditional SLAM framework. This framework does not need prior knowledge of objects and their shapes or models. The method uses monocular depth estimation to achieve instance-level segmentation and optical flow. Then, static and dynamic object features are tracked to trace the trajectories. The local map is updated in each consecutive frame. Environmental scene understanding using only monocular cameras can be challenging in mapping applications. As a result, semantic segmentation using LiDAR, radar, and stereo cameras has been investigated. The development of deep learning techniques has increased the usage of point cloud data in robot 3D scene understanding applications. VoxNet [114] is a 3D CNN architecture that was designed to obtain voxel grid representations of the environment using point cloud data. Although this network is capable of detecting objects, object detection alone is not sufficient for scene understanding. [115] used a LeNet [116] based architecture for semantic segmentation of point clouds. This approach could segment outdoor scenes into seven object categories by voxelisation of input point cloud data. [117] proposed a method for semantic segmentation of indoor scenes using only depth images. This network jointly predicts the 3D occupancy grid and semantic categories. However, some of the objects were missing during the inference due to a lack of depth details for those objects. In general, point cloud data is irregular, therefore, only representing point cloud data using uniform voxel grids might not accurately interpret real-world information. To address this adverse effect, [118] proposed the OctNet architecture to represent point cloud data in a hierarchical partitioning of space using voxel representations. One of the problems of using voxel grid representations is the requirement of high computational power for varying sparsity of irregular point cloud data. Multi-view representations of 3D point clouds can be employed to reduce the adverse effects of volumetric grid representations. [119]



attempts to generate 2D views of point cloud data, which are then utilised to analyse large urban point cloud data in 3D scenes using deep CNNs. Then, "tangent convolutions" are applied to the point cloud data to obtain an image by projecting local geometry to a 2D plane tangent to a point.

In general, 2D projective methods are more efficient and scalable than 3D volumetric representations. However, there may be a degree of information loss in the final 3D structure due to inaccuracies in those projections. Instead of converting point cloud data into uniform volumetric representations, processing these point clouds as unstructured data is also a feasible approach [120]. [55] proposed PointNet, one of the significant point cloud semantic segmentation works. This work shows a degree of permutation invariance to the input point cloud data. It can classify objects by allocating them to categories (e.g., tv, table) and carry out semantic segmentation in scenes. To try to improve on these point-level processing techniques, several works have been carried out that use deep learning methods [121–125]. Computer memory requirements for 3D voxel representations tend to vary according to the level of voxel resolution. In the presence of sensor noise, the estimation of the occupancy of voxels might be challenging. Instead of using 3D volumetric representations, graph and tree-based representations have been proposed to recover 3D structures from point cloud data. [126] introduces a 3D graph neural network to semantically segment the environment using 2D RGB image and point cloud data. This method uses both CNNs and graph neural networks to predict semantic classes, with the CNN used to extract features from 2D RGB images. The Superpoints graph method was introduced by [127] to learn scene contextual information and geometry. Superpoints are interest points in a point cloud that are detected and described using a neural network. They are densely sampled and designed to be efficient and robust for use in SLAM applications. First, the input point cloud is separated into geometric elements (named Superpoints), and then these Superpoints are combined considering their mutual features. Finally, graph convolutions are implemented to generate contextual information and semantic labels. This method was shown to be appropriate for the segmentation of large point clouds. Another significant work that processes order less point cloud data using a permutation invariant deep learning architecture is So-Net [128]. This network extracts features from point clouds hierarchically. The receptive field of the CNN is controlled systematically by performing a point-to-node k-nearest neighbour search to retrieve the point cloud spatial information. This network conserves the topology attributes of the point representations.

Overall, a number of approaches show promise for improving the scene understanding capabilities of autonomous robots, although questions remain regarding how effectively they can be applied in outdoor unstructured environments, particularly with dense vegetation or other complex terrain features. Improvements in both sensor capabilities and algorithms are likely to be necessary to achieve significant progress in this area, but some promising research directions have been highlighted.

#### 4. Mobile Robot Local Path Planning

Approaches for path planning of robots in local environments can be broadly classified into either classical methods or learning-based methods that attempt to modify the robots' path-planning based on environmental conditions [129,130]. The classical methods follow a modular architecture with environmental perception, planning of paths relative to generated global maps, and trajectory following. Many classical techniques are applicable for robot navigation in static environments however, their suitability can substantially diminish in unstructured or dynamic environments [131]. In general, these methods can be used effectively in indoor mobile robot applications but may not be suitable for outdoor off-road navigation conditions (e.g., terrain with grass might be traversable, despite appearing blocked, and ground with mud or sand may not be suitable for pathing, despite appearing clear), hence the requirement to investigate machine learning-based navigation approaches. Several global and local motion planning methods for ground robot navigation are compared in Tables 4 and 5.

The A\* and Dijkstra algorithms have been well-researched in the past decades, and these methods have shown their potential by extensively being applied with the Robot Operating System (ROS) in



many real-world robotics applications. Combining these two path-planning methods with heuristic searching is effective in relatively low complexity 2D environments. However, these methods require heavy computational capabilities in high-dimensional environments or may struggle in unstructured and dynamic working environments.

**Table 4.** Global path planning algorithms.

Algorithms	Advantages	Disadvantages
Dijkstra	The calculation strategy is not complex and give the shortest path	The increment of traversal nodes complicates the calculations
A*	In static environments the algorithm search efficiency is high	Not appropriate for dynamic environments
D*	Good for dynamic environment path planning and more efficient than A*	Planning longer paths via D* creates challenges
RRT	Fast convergence, high search capability	Algorithm efficiency is low in unstructured environments
Genetic	Appropriate for complex environments, good for finding optimal paths	Low algorithm convergence speed, low search ability in local paths
Ant colony	Appropriate for complex environments, can be combined with other heuristic-based path planners	Slow convergence rate, easily trapped in local minima
Particle swarm optimisation	High convergence rate, good robustness	Frequently, solutions converge into local optimal solutions

**Table 5.** Local path planning algorithms.

Algorithms	Advantages	Disadvantages
Artificial potential field	Can be implemented for 3D path planning, and can solve the local minimum problem	Cannot guarantee the optimal solution
Simulated annealing	Flexible and easy implementation, can deal with noisy data and non-linear models	Can produce unstable results, the trade-off between accuracy and speed
Fuzzy logic	Strong robustness, decrease the dependencies between environmental data	Needs accurate prior knowledge, poor learning capabilities
Neural network	Strong robustness, and learning ability from experiences	Low path planning efficiency
Dynamic window	Good self-adaptation to environments	Not appropriate for unstructured complex environments

Random sampling-based path planning algorithms generally consist of BITs and RRTs. Regionally Accelerated Batch Informed Trees (RABIT) are more widely used and perform better in high dimensional and dynamic environments [132] in comparison to graph search-based path planning algorithms. Bionic-based intelligent robot path planning methods simulate the behaviours of insects to generate evolving paths. These evolutionary methods include the ant colony, particle swarm

optimisation, genetic and artificial bee colony algorithms. Many other optimised versions of these algorithms have been proposed to improve calculation efficiency and avoid local minimum problems. A welding robot system developed in [133] had a combination of genetic and particle swarm optimisation algorithms to solve for the shortest route, while avoiding obstacles. The artificial potential field method, ant colony optimisation, and geometry optimisation path methods were combined in another research study [134] to search the globally optimal path in 2D scenarios using computer simulations. This method showed fast solutions and reduced the risk of trapping robots in local minimum points. A multi-objective hierarchical particle swarm optimisation method has been proposed by [135] to plan global optimal path trajectories in cluttered environments. This method utilises three layers to generate the robot navigation trajectories. The triangular decomposition method [136] is applied in the first layer, the Dijkstra algorithm is used in the second, and a modified particle swarm optimisation algorithm is applied in the last layer.

In general, the local path planning strategies that have been discussed use the available sensor data of robots regarding their surroundings to map, understand, and generate local paths while avoiding obstacles. These local path planning methods are effective in mobile robotic applications because the data captured by sensors varies in real time in response to the dynamics of environments. In comparison to global path planning strategies, local path planning is more critical for practical robot operation and usually serves as a bridge between global planning and the direct control of robots. However, local path planners have one notable drawback in that they often lead robots to local minimum points. Many classical local path planning algorithms can generate optimal paths in the local environments while avoiding local minimum problems. These methods include techniques such as the Fuzzy Logic algorithm, artificial potential field method, and simulated annealing algorithm. In general, these methods do not evaluate the relative velocities between robots and dynamic environment objects, however, which can lead to difficulties. In many worst-case scenarios, even the velocity profiles of these obstacles can be hard to acquire for the robots. Visual-inertial odometry methods show success in outdoor environment navigation scenarios, but are not suitable for navigating in off-road conditions without pre-built maps or GPS assistance.

Machine learning methods have been applied for mobile robot navigation to learn semantic information [137–139] and statistical patterns [140,141] of environments. Several other research works [142–145] have used machine learning to achieve robustness in path following. In recent years, many Reinforcement Learning (RL) and imitation learning methods and approaches based on self-supervised learning [146–149] have been applied in mobile robot navigation policy design and for training supervision. Many classical modular and deep learning-based approaches have been utilised in the navigation modules of outdoor mobile robots [150]. RL uses strategies to learn optimal robot decisions from experiences. The interconnection between environments and robots is modelled as a Markov Decision Process (MDP). Robots receive rewards as feedback signals for training while traversing different environments. The basic RL process is illustrated in Figure 2. RL methods can be separated into model-free and model-based learning scenarios. In model-free RL, the robots are not required to evaluate the MDP model rewards or policies directly and can obtain these directly through what the robot experiences. Model-free RL approaches have several subcategories such as value-based, policy-based, and Actor-Critic (improved versions of the policy-based algorithms) [151,152].

In value-based methods, optimal policies are obtained by iteratively updating the value functions. Policy gradient-based methods directly approximate a policy network and update the policy parameters to get an optimal policy that maximises the reward value. Deep Q network (DQN) and Double DQN are the two main value-based DRL methods. A DQN-based end-to-end navigation method has been introduced in [153]. In this work, a feature-extracting network used an edge segmentation method to improve the efficiency of the network training process. The simulated models were transferred to the real world without significant performance loss. Discrete robot actions were implemented based on a grid map. The robot exploration framework is divided into decision, planning, and mapping modules. The learning-based decision module has shown good performance, efficiency,

and adaptability in novel environments. A graph-based technique has been implemented for mapping module. The applied path planning module includes the A\* algorithm as the global planner along with a timed elastic band [154] local planner.

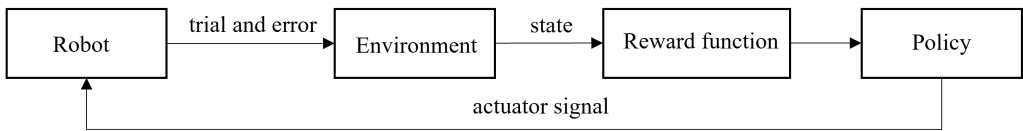


Figure 2. Robot RL approach.

Value-based DRL methods produce discrete actions meaning they are not appropriate for the continuous robot action space. The policy-based DRL methods provide continuous motion commands for robots. Combining the policy gradient with the value function creates the Actor-Critic RL method. In general, Actor-Critic algorithms are well-suited for the continuous motion space of autonomous mobile robots. The Actor approximates the policy used to generate actions in the environment. The Critic architecture is accountable for using a reward function to evaluate robot actions iteratively and guides the Actor in the successive iteration. The Critic uses deep learning value functions such as DQN, and Double DQN to evaluate each iteration step. The Actor-Critic approach includes learning methods like Depth Deterministic Policy Gradient algorithm (DDPG), Trust Region Policy Optimisation (TRPO), Proximal Policy Optimisation (PPO), Asynchronous Advantage Actor-Critic (A3C), and Soft Actor-Critic (SAC) [15]. An extended Actor-Critic algorithm was proposed in [155]. The visual navigation module of the deep neural network consisted of depth map prediction and semantic segmentation auxiliary tasks. The proposed learning network requires an image of the target and an observed image as inputs. This network architecture was proposed to obtain a visual navigation policy for indoor environments. A summary of DRL motion planning methods is included in Table 6.

Table 6. DRL motion planning methods.

Algorithms	Advantages	Disadvantages
DQN	Updates are done offline, are not complex, and are reliable	Only discrete motions
DDPG	High sample efficiency, less data correlation and faster convergence compared to DQN	The poor generalisation of novel environments
TRPO	Ensure stable convergence	Too many assumptions, may create large errors
PPO	Simplified solution process, good performance and easier to implement compared to TRPO	Low sampling efficiency
A3C	Asynchronous parallel network training, fast convergence, suitable for multi-robot systems	Require large training data, Difficult to migrate model to real world
SAC	Better robustness and sample efficiency compared to the above methods	Bulky model size

Model-based robot RL methods generate models of external environments using supervised training and implement value functions to learn actions that maximise the return. Model-based RL has faster convergence and high sample efficiency. [156] presents a biped robot that learns on a rotating platform by combining model-based and model-free machine learning methods. This paper has addressed the overfitting problem of model-based robots. The research has simulated the

robot in a 2D scenario and has shown a reduction in learning time than model-free RL algorithms. Imitation learning is another related approach which uses demonstrations by experts operating robots for specific tasks to obtain policy functions. Mobile robots learn mappings between observations of these demonstrations and appropriate robot actions. Path trajectories can be rapidly generated by manipulating learned policies from expert demonstrations. However, the robot's capacity to practically record these demonstration results from real-world experiments can be challenging. [157] has introduced a method to acquire a mapping from actions to states using ego-centric videos collected by a human demonstrator using a mobile phone camera. The policy learning step was executed within a simulation platform, then the developed navigation policy was tested on a Clearpath Jackal wheeled robot in an indoor environment. The robot and the trained videos had view-point mismatches, but the model was robust to those changes. The system was successfully able to map camera sensor inputs to actuator commands using the developed imitation learning policy.

Inverse RL is another approach capable of learning reward functions from expert demonstrations. However, inverse RL is more appropriate for exploring novel environments because imitation learning tries to directly follow a demonstrator rather than improve beyond that knowledge level. The creation of an inverse RL network by employing a vision-based imitation learning method was presented by [157]. This method approximates a value function from one middle layer of a policy trained by imitation learning. The related experiments were carried out on a real-world setup and using the ROS Gazebo simulation platform. This proposed method has shown the capability of generalising the system for unseen environments by producing usable cost maps. Traditional geometric-based SLAM procedures lack the capacity to capture dynamic objects in external environments and are thus only suitable for planning robot actions in static environments. RL-based techniques, however, can learn policies by considering dynamic obstacles in outdoor environments, although the sample efficiency is lower compared to imitation learning and model-based learning. The implementation of learning-based maps and traditional path planners is one way to improve sample efficiency. A mobile robot affordance map generation using RL policy-based method was introduced by [158]. This research used metric cost maps (attaching semantics and geometry) for robot navigation. The A\* classical path planner was implemented for path generation. This learning-based SLAM approach was implemented using a simulation software. Semantic, dynamic, and behavioural attributes of unseen environments have been learnt by the model using the simulation scheme. Local path planning remains a highly challenging task, particularly in environments with dynamic actors or challenging terrain features such as unpredictably varying topography or varying ground conditions. Modern machine learning approaches, coupled with advanced SLAM techniques show promise for enabled effective navigation planning, but methods that can effectively handle all of these challenges do not yet exist and considerable further work will be required to enable fully autonomous navigation in all conditions.

## 5. Summary of the Current State-of-the-Art Techniques

Autonomous navigation can be broadly categorised into the classical modular pipeline methods and end-to-end learning approaches. The modular pipeline approach has limitations due to the requirement for high levels of human intervention in designing the modules, the loss of information through each module, and the overall lack of robustness when conditions vary beyond anticipated limits. The end-to-end learning approach suffers from problems such as over-fitting and poor generalisation. Both modular and end-to-end robot navigation approaches rely on sensors to capture information about the environment or internal robot attributes. Researchers have investigated different sensor modalities to improve robot perception, including raw input types like sound, pressure, light, and magnetic fields, as well as common robot perception sensor modalities like cameras, LiDAR, radar, sonar, GNSS, IMU, and odometry sensors. It is crucial to have a reliable real-time understanding of external 3D environments to ensure safe robot navigation. While cameras are commonly used in mobile robotics, relying on a single sensor is not always robust. Different sensor modalities are

often combined in robotic applications to improve perception reliability and robustness. Camera and LiDAR fusion has been a popular approach in the robotics community. This fusion method has shown better performance than other vision-based dual sensor fusion approaches, and recent advances in deep learning algorithms have further improved its performance. Monocular and stereo cameras are commonly used with LiDAR sensors to fuse images and point cloud data. However, the technical challenges and cost of sensors and processing power requirements still limit the widespread application of these methods for regular use. Computer vision has experienced rapid growth in the past decade, with deep learning methods accelerating its development. Object detection, depth estimation, semantic segmentation, instance segmentation, scene reconstruction, motion estimation, object tracking, scene understanding, and end-to-end learning are among the subtopics of computer vision. These methods have been widely applied in autonomous navigation, but their accuracy and reliability can be limited, prompting ongoing efforts to improve their quality. Benchmarking datasets, such as KITTI, Waymo, A2D2, nuScenes, Cityscapes, RELLIS-3D, RUGD, Freiburg [24], and WildDash [159], have been used to compare the performance of different vision methods in autonomous urban driving and off-road driving applications.

Semantic SLAM techniques provide geometric and semantic details of external environments. However, existing semantic SLAM techniques may not be adequate for the safe navigation of robots in outdoor unstructured environments due to challenges in feature detection, uneven terrain conditions (which can lead to robot localisation errors, detection errors, etc.) and effects of wind on vegetation, which can lead to sensor noise and ambiguous results. Therefore, terrain traversability analysis, and improved scene understanding using deep learning methods have been developed that are showing promise with assisting robot navigation in outdoor unstructured environments.

Many classical robot path-planning approaches are used for navigation in local environments. These classical approaches rely on a modular architecture to perceive the environment, plan paths relative to generated maps, and follow trajectories. However, these classical methods are often challenged in dynamic or deformable environments and are not well-suited for off-road navigation conditions where terrains may be unpredictable. Learning-based methods have been developed to improve robot path-planning abilities under different environmental conditions and have shown promise in addressing these challenges. In particular, learning-based navigation methods have been shown to be better suited for off-road navigation conditions, where terrains can be highly variable and unpredictable.

## **6. Research Challenges and Future Directions in Unstructured Outdoor Environment Navigation**

Despite the significant progress that has been made in improving robots' abilities to perceive their environment, localise themselves within it, and plan paths to navigate through it, significant challenges remain with the application of these methods to unstructured outdoor environments. Many limitations exist with respect to perception capability, particularly when sensor and computation costs are a factor, and effective methods for robust scene understanding within such environments have not yet been demonstrated, substantially hindering the ability to deploy fully autonomous robots into such environments. This section of the paper explores the current state of these significant research challenges and provides suggestions for the most promising directions for future research efforts to overcome them.

### *6.1. Research Challenges*

SLAM methods have been extensively used for robot navigation applications in indoor environments. Urban autonomous navigation systems also implement SLAM algorithms successfully for sophisticated motion planning. Only a few applications have demonstrated the use of SLAM for dirt road and offroad uneven terrain navigation. Many visual SLAM techniques are unstable in offroad environments with trees, bushes, uneven terrain and cluttered objects. Application of classical SLAM approaches in offroad environments with vegetation is challenging because of the irregular



and dynamically varying structure of the natural environment. LiDAR-based methods may detect traversable regions as geometric obstacles. Therefore, existing SLAM research on robot navigation in off-road unstructured environments is limited. Several feature-based SLAM algorithms fail due to reasons such as the absence of specific features or the loss of detected features caused by the dynamic movement of vegetation due to wind. In general, mobile robots in offroad environments in the presence of vegetation use GNSS with IMUs to estimate the real-time position of robots. In relatively open tree canopies, GNSS sensors can establish a consistent communication link with satellites. The GNSS positioning accuracy in sparse tree canopy environments can be decimeter level [160]. However, under dense tree canopies, GNSS may lose location data due to potential signal interruptions.

Visual SLAM (VSLAM) uses features or pixel intensity in images to generate information about environments. Many visual SLAM methods employ feature-based (e.g., points, lines, corners, planes) algorithms for detecting objects or developing a higher-level understanding of environments. In general, monocular, stereo and RGB-D cameras are used in VSLAM. RGB-D camera SLAM is restricted to indoor applications because of its sensitivity to ambient light conditions. SLAM approaches that use event cameras are still not well established. Many point-feature-based SLAM methods are not robust in low-textured scenes like plane surfaces and are vulnerable to illumination changes. Line features are less sensitive to changes in lighting conditions than point features. Plane features are often found in the artificial environment but are not regularly present in natural environments.

Using pixel intensities instead of features in SLAM is more robust to changes in lighting conditions and results in more scene information. Modern semantic VSLAM methods are mostly restricted to specific environments such as indoor or urban driving. Many VSLAM approaches focus on feature-based techniques to generate maps and localise robots. In an offroad environment with vegetation, feature extraction is challenging. The number of distinguishable features in these environments might be low due to the unstructured nature of natural environments. The presence of occlusions, shadows and illumination variation in offroad unstructured environments can obstruct useful feature detection in VSLAM. Therefore, obtaining a higher level of semantic understanding is more complex than understanding urban driving scenarios. In order to improve VSLAM performance in vegetated, outdoor, unstructured environments, one can train deep learning models to understand features using a large dataset. However, there are difficulties in adapting these networks to unseen environments [161].

LiDAR SLAM is more robust than VSLAM in varying illumination conditions and under shadows. Point cloud registration and feature-based methods are the most common LiDAR SLAM approaches [162]. The LiDAR SLAM point cloud registration approach is more robust in vegetated natural environment mapping than the feature-based method. An adverse effect of LiDAR SLAM is the loss of semantic details in the environment. Researchers have introduced camera and LiDAR sensor fusion-based SLAM concepts to compensate for these issues. There are two separate LiDAR-camera fused SLAM directions, loosely coupled and tightly coupled. In loosely coupled SLAM systems, the modalities use two independent approaches for extracting features and high-level information to combine the two information streams. Computations in loosely-coupled systems are less complex and generate faster results than in tightly coupled systems. However, these systems may produce low accuracies due to complementary information losses in independent sensor data processing steps. Tightly-coupled models fuse the two forms of sensor data into one framework to generate feature detection or higher-level semantic understanding. Therefore, this approach is more robust in challenging environments. However, these methods are more complex and need high computer memory and GPU requirements.

In unstructured outdoor environments, robots require perception, scene understanding and identification of terrain regions suitable for robot navigation. If robot working environments are dynamic, achieving these tasks will be more challenging. Robot scene understanding involves a robot's ability to perceive, analyse, and interpret its working environment. Scene understanding relies on subtopics such as object recognition, semantic segmentation, instance segmentation and scene

representations of images or videos. Robot scene understanding is a continuous process, and robots need to be able to learn and adapt to new environments and situations over time. Implementation of machine learning algorithms in scene understanding has increased the possibilities of generating more sophisticated learning-based architectures. However, in general, these methods have high computational requirements. Object recognition in unstructured environments is challenging due to higher frequency of object occlusions, similar features in different objects, indistinct object boundaries and varying lighting conditions. Semantic segmentation of an unstructured environment is another essential part of understanding the environment, and it is more complex than segmentation in urban environment scenarios due to the variation of different semantic classes. It is challenging to extract fine-grained semantic details from unstructured outdoor environments. These environments have wide semantic diversity, and training robots using large, annotated datasets is challenging due to the limited availability and difficulty of creating such datasets.

Researchers use instance segmentation algorithms to acquire a semantically rich understanding of environments, relationships between objects, and contexts. However, limited research has been conducted on robot instance segmentation of urban road environments, and these methods require high computational requirements and are not robust in off-road scenarios. Robots should learn how to robustly interpret environments over time using scene-understanding information to overcome these limitations. Point-based, graph-based, tree-based, multi-view projections, image-based and volumetric representations have been developed to achieve scene understanding beyond basic perception. These techniques can comprehend environments beyond what can be inferred from a single image. Further investigation of the above-mentioned scene representations is necessary to learn and evolve scene interpretations over time. Novel concepts need to be developed and investigated to extract information from scenes at the object and semantic levels.

Robot domain identification and adaptation are valuable for real-world applications to improve performance and achieve more sophisticated navigation capability. In general, robots require the adaptability of learning-based navigation models. Therefore, networks that are trained on one domain (e.g., simulated environment [163], urban city environment) need to identify and adapt to changes in the application domain (e.g., real world, rural off-road environment). These domain gaps can be expressed as one challenge for robot navigation in unstructured environments. The domain gap refers to the significant deviations in environment attributes of the known or trained domains relative to the actual operating domains of the robot. These differences in application domain can occur due to variations in weather, texture, previously unseen terrain conditions, environment context changes (e.g., static to dynamic) or different factors in the robot operating environments. Deep learning architectures such as reinforcement learning can generate robust performance in one target domain but may subsequently fail due to an unforeseen drastic change in that environment.

Robots can be designed to be more adaptable if they are trained on large and diverse real-world data. However, the collection of real-world data to achieve such diversity can be challenging. Thus, there is a requirement for the development of more efficient data collection and data augmentation techniques. One option to address this is to train robots in computer simulators using data collected synthetically instead of training on real-world data. This method has benefits such as a reduction of the number of real-world experiments and a more economical and safer deployment in comparison to real-world systems. However, the trained models might have low adaptation to the real world due to overfitting to the simulation environment conditions. If the physics and the environments of simulators are significantly different from the target domain, the learned policies and loss functions in the simulator domains might not be directly applicable to the real world, thus, leading to low performance. Novel techniques are needed to ensure that learning-based models, policies, and loss functions are transferable to the application domain to bridge this gap. Real-time performance is a critical aspect of robots working effectively in unstructured environments. The models trained in simulators might be unnecessarily complex for application in the real world. Therefore, techniques of domain adaptation must be transferable, economically sustainable, and computationally efficient in

making decisions. In safety-critical applications, domain adaptation must be performed while ensuring safety, which requires careful validation and testing to ensure that the adapted model behaves safely in the real world.

## 6.2. Future Research Directions

Multimodal sensor fusion for robot vision provides more robust and reliable environment perception and scene understanding results than mono-sensor robot vision strategies in unstructured environments. Cameras and LiDARs are usually employed to retrieve RGB and depth data of external environments, respectively. However, high-resolution LiDAR sensors are costly, and their depth data sparsity increases when the observation range is increased. In contrast to mechanical LiDARs, solid-state LiDARS can provide highly accurate and higher resolution depth data. Their costs are also likely to be significantly lower than mechanical spinning LiDARs, but these sensors are still at the research stage. The development and commercialisation of solid-state LiDARs will likely be pivotal for future advancements in robotic unstructured environment perception.

In robot navigation, domain change can involve training a model on simulated data and then fine-tuning it on real-world data to improve its performance in the real world. Robot domain adaptability can be enhanced by using techniques such as transfer learning, adversarial training, or domain adversarial neural networks. Researchers have proposed techniques like data augmentation, feature alignment, or domain adaptation loss functions to address the domain gap. However, domain adaptation remains an active area of research in robotics, and there is still much to be done to improve the performance of machine-learning models in real-world settings. End-to-end learning architectures for robot navigation are appropriate for understanding robot tasks in the local motion planning context. For long-range navigation and in different environment domains, learning-based methods in combination with the classical hierarchical navigation pipeline show promise to produce more robust and safe results.

Robot environment perception, analysis and scene understanding are critical for robots to generate useful maps and make intelligent decisions. The recent research works such as Kimera [101], Kimera-multi [102] are able to generate high-level geometric and hierarchical semantic environment understanding for robots. Robot systems that consist of geometric and hierarchical scene representations combined with incremental scene understanding will be advantageous in future robot outdoor unstructured navigation applications. However, the degree of scalability and adaptability of these approaches in unstructured environments with vegetation remains an open research question. In vegetated unstructured environments, obtaining feature-based scene understanding is challenging. Many SLAM methods provide rich geometric data but may incorrectly identify grass covered terrains as not being traversable and muddy areas as traversable. Therefore, the combination of LiDAR based geometric/semantic maps with the assistance of reinforcement learning of ego-centric images for robot local path planning could be a promising direction for more robust outdoor unstructured environment navigation. This approach will combine robot SLAM with image-based scene understanding to generate higher levels of spatial understanding of local environments for robots to be able to generate more accurate and safe movements. Overall, our comprehensive review of the literature has indicated that further research into multimodal sensor fusion techniques and deep learning-based scene understanding and task planning methods provides the most promise towards achieving the goal of fully autonomous navigation in outdoor unstructured environments.

## 7. Conclusion

This paper has provided a comprehensive review of the current state-of-the-art in autonomous mobile ground robot navigation, identified research gaps and challenges, and suggested promising future research directions for improved autonomous navigation in outdoor unstructured terrains. A broad review of robot sensing, camera-LiDAR sensor fusion, robot scene understanding and local

path planning techniques has been provided to deliver a comprehensive discussion of their essential methodologies and current capabilities and limitations. The use of deep learning, multimodal sensor fusion, incremental scene understanding concepts, scene representations that preserve input data topology and spatial geometry, and learning-based hierarchical path planning concepts are identified as promising research domains to investigate in order to realise fully autonomous navigation in unstructured outdoor environments. Our review has indicated that applying these techniques to outdoor unstructured terrain robot navigation research can likely improve robot domain adaptability, scene understanding and conscious decision-making abilities.

**Author Contributions:** Conceptualisation, A.R., D.C. and L.W.; writing-original, L.W.; writing-review and editing, A.R., D.C.; supervision, A.R., D.C. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research received no external funding.

**Data Availability Statement:** Data sharing not applicable.

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

## References

1. Rubio, F.; Valero, F.; Llopis-Albert, C. A review of mobile robots: Concepts, methods, theoretical framework, and applications. *International Journal of Advanced Robotic Systems* **2019**, *16*, 1–22.
2. Cai, K.; Wang, C.; Cheng, J.; De Silva, C.W.; Meng, M.Q.H. Mobile robot path planning in dynamic environments: A survey. *arXiv preprint arXiv:2006.14195* **2020**.
3. Quarles, N.; Kockelman, K.M.; Lee, J. America's fleet evolution in an automated future. *Research in Transportation Economics* **2021**, *90*, 1–12.
4. Pavel, M.I.; Tan, S.Y.; Abdullah, A. Vision-based autonomous vehicle systems based on deep learning: A systematic literature review. *Applied Sciences* **2022**, *12*.
5. Zhang, S.; Yao, J.; Wang, R.; Liu, Z.; Ma, C.; Wang, Y.; Zhao, Y. Design of intelligent fire-fighting robot based on multi-sensor fusion and experimental study on fire scene patrol. *Robotics and Autonomous Systems* **2022**, *154*, 1–18.
6. Li, Q.; Kroemer, O.; Su, Z.; Veiga, F.F.; Kaboli, M.; Ritter, H.J. A review of tactile information: Perception and action through touch. *IEEE Transactions on Robotics* **2020**, *36*, 1619–1634.
7. Alatise, M.B.; Hancke, G.P. A review on challenges of autonomous mobile robot and sensor fusion methods. *IEEE Access* **2020**, *8*, 39830–39846.
8. Feng, D.; Haase-Schütz, C.; Rosenbaum, L.; Hertlein, H.; Glaeser, C.; Timm, F.; Wiesbeck, W.; Dietmayer, K. Deep multi-modal object detection and semantic segmentation for autonomous driving: Datasets, methods, and challenges. *IEEE Transactions on Intelligent Transportation Systems* **2020**, *22*, 1341–1360.
9. Hu, C.; Yang, C.; Li, K.; Zhang, J. A forest point cloud real-time reconstruction method with single-line LiDAR based on visual-IMU fusion. *Applied Sciences* **2022**, *12*.
10. Jin, X.B.; Su, T.L.; Kong, J.L.; Bai, Y.T.; Miao, B.B.; Dou, C. State-of-the-art mobile intelligence: Enabling robots to move like humans by estimating mobility with artificial intelligence. *Applied Sciences* **2018**, *8*.
11. Yang, T.; Li, Y.; Zhao, C.; Yao, D.; Chen, G.; Sun, L.; Krajník, T.; Yan, Z. 3D ToF LiDAR in mobile robotics: A review. *arXiv preprint arXiv:2202.11025* **2022**.
12. Moon, J.; Lee, B.H. PDDL planning with natural language-based scene understanding for UAV-UGV cooperation. *Applied Sciences* **2019**, *9*.
13. Yang, M.; Rosenhahn, B.; Murino, V. *Multimodal scene understanding: Algorithms, applications and deep learning*; Academic Press: United Kingdom, 2019; pp. 1–7.
14. Zhang, Y.; Sidibé, D.; Morel, O.; Mériaudeau, F. Deep multimodal fusion for semantic image segmentation: A survey. *Image and Vision Computing* **2021**, *105*, 1–17.
15. Sun, H.; Zhang, W.; Yu, R.; Zhang, Y. Motion planning for mobile robots—Focusing on deep reinforcement learning: A systematic review. *IEEE Access* **2021**, *9*, 69061–69081.
16. Janai, J.; Güney, F.; Behl, A.; Geiger, A.; others. Computer vision for autonomous vehicles: Problems, datasets and state of the art. *Foundations and Trends® in Computer Graphics and Vision* **2020**, *12*, 1–308.

17. Gupta, A.; Efros, A.A.; Hebert, M. Blocks world revisited: Image understanding using qualitative geometry and mechanics. 11th European Conference on Computer Vision, Heraklion, Crete, Greece, 5-11 September 2010, pp. 482–496.
18. Kocić, J.; Jović, N.; Drndarević, V. Sensors and sensor fusion in autonomous vehicles. 26th Telecommunications Forum (TELFOR), Serbia, Belgrade, 20-21 November 2018, pp. 420–425.
19. Muñoz-Bañón, M.Á.; Candelas, F.A.; Torres, F. Targetless camera-LiDAR calibration in unstructured environments. *IEEE Access* **2020**, *8*, 143692–143705.
20. Li, A.; Cao, J.; Li, S.; Huang, Z.; Wang, J.; Liu, G. Map construction and path planning method for a mobile robot based on multi-sensor information fusion. *Applied Sciences* **2022**, *12*.
21. Wang, W.; Shen, J.; Cheng, M.M.; Shao, L. An iterative and cooperative top-down and bottom-up inference network for salient object detection. IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 15-20 June 2019, pp. 5968–5977.
22. Santos, L.C.; Santos, F.N.; Pires, E.S.; Valente, A.; Costa, P.; Magalhães, S. Path planning for ground robots in agriculture: A short review. IEEE International Conference on Autonomous Robot Systems and Competitions (ICARSC), Ponta Delgada, Portugal, 15-16 April 2020, pp. 61–66.
23. Fayyad, J.; Jaradat, M.A.; Gruyer, D.; Najjaran, H. Deep learning sensor fusion for autonomous vehicle perception and localization: A review. *Sensors* **2020**, *20*, 1–35.
24. Valada, A.; Oliveira, G.L.; Brox, T.; Burgard, W. Deep multispectral semantic scene understanding of forested environments using multimodal fusion. International Symposium on Experimental Robotics, Tokyo, Japan, 3-6 October 2016, pp. 465–477.
25. Lei, X.; Zhang, Z.; Dong, P. Dynamic path planning of unknown environment based on deep reinforcement learning. *Journal of Robotics* **2018**, *2018*, 1–10.
26. Crespo, J.; Castillo, J.C.; Mozos, O.M.; Barber, R. Semantic information for robot navigation: A survey. *Applied Sciences* **2020**, *10*, 1–28.
27. Galvao, L.G.; Abbod, M.; Kalganova, T.; Palade, V.; Huda, M.N. Pedestrian and vehicle detection in autonomous vehicle perception systems—A review. *Sensors* **2021**, *21*, 1–47.
28. Hewawasam, H.; Ibrahim, M.Y.; Appuhamillage, G.K. Past, present and future of path-planning algorithms for mobile robot navigation in dynamic environments. *IEEE Open Journal of the Industrial Electronics Society* **2022**, *3*, 353–365.
29. Martini, M.; Cerrato, S.; Salvetti, F.; Angarano, S.; Chiaberge, M. Position-Agnostic Autonomous Navigation in Vineyards with Deep Reinforcement Learning. IEEE 18th International Conference on Automation Science and Engineering (CASE), 20-24 August 2022, pp. 477–484.
30. Huang, Z.; Lv, C.; Xing, Y.; Wu, J. Multi-modal sensor fusion-based deep neural network for end-to-end autonomous driving with scene understanding. *IEEE Sensors Journal* **2020**, *21*, 11781–11790.
31. Hamza, A. Deep reinforcement learning for mapless mobile robot navigation. Master's thesis, Luleå University of Technology, Sweden, 2022.
32. Carrasco, P.; Cuesta, F.; Caballero, R.; Perez-Grau, F.J.; Viguria, A. Multi-sensor fusion for aerial robots in industrial GNSS-denied environments. *Applied Sciences* **2021**, *11*.
33. Li, R.; Wang, S.; Gu, D. DeepSLAM: A robust monocular SLAM system with unsupervised deep learning. *IEEE Transactions on Industrial Electronics* **2020**, *68*, 3577–3587.
34. Aguiar, A.; Santos, F.; Sousa, A.J.; Santos, L. FAST-FUSION: An improved accuracy omnidirectional visual odometry system with sensor fusion and GPU optimization for embedded low cost hardware. *Applied Sciences* **2019**, *9*.
35. Li, Y.; Brasch, N.; Wang, Y.; Navab, N.; Tombari, F. Structure-slam: Low-drift monocular slam in indoor environments. *IEEE Robotics and Automation Letters* **2020**, *5*, 6583–6590.
36. Zaffar, M.; Ehsan, S.; Stolkin, R.; Maier, K.M. Sensors, SLAM and long-term autonomy: A review. NASA/ESA Conference on Adaptive Hardware and Systems (AHS), United Kingdom, 6-9 August 2018, pp. 285–290.
37. Sabattini, L.; Levratti, A.; Venturi, F.; Amplo, E.; Fantuzzi, C.; Secchi, C. Experimental comparison of 3D vision sensors for mobile robot localization for industrial application: Stereo-camera and RGB-D sensor. 12th International Conference on Control Automation Robotics & Vision (ICARCV), Guangzhou, China, 5-7 December 2012, pp. 823–828.
38. Tölgyessy, M.; Dekan, M.; Chovanec, L.; Hubinský, P. Evaluation of the azure kinect and its comparison to kinect v1 and kinect v2. *Sensors* **2021**, *21*, 1–23.



39. Evangelidis, G.D.; Hansard, M.; Horaud, R. Fusion of range and stereo data for high-resolution scene-modeling. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **2015**, *37*, 2178–2192.
40. Glover, A.; Bartolozzi, C. Robust visual tracking with a freely-moving event camera. IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Vancouver, BC, Canada, 24–28 September 2017, pp. 3769–3776.
41. Gallego, G.; Delbrück, T.; Orchard, G.; Bartolozzi, C.; Taba, B.; Censi, A.; Leutenegger, S.; Davison, A.J.; Conradt, J.; Daniilidis, K.; others. Event-based vision: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **2020**, *44*, 154–180.
42. Yuan, W.; Li, J.; Bhatta, M.; Shi, Y.; Baenziger, P.S.; Ge, Y. Wheat height estimation using LiDAR in comparison to ultrasonic sensor and UAS. *Sensors* **2018**, *18*, 1–20.
43. Moosmann, F.; Stiller, C. Velodyne slam. IEEE Intelligent Vehicles Symposium, Baden-Baden, Germany, 5–9 June 2011, pp. 393–398.
44. Li, K.; Li, M.; Hanebeck, U.D. Towards high-performance solid-state-lidar-inertial odometry and mapping. *IEEE Robotics and Automation Letters* **2021**, *6*, 5167–5174.
45. Poulton, C.V.; Yaacobi, A.; Cole, D.B.; Byrd, M.J.; Raval, M.; Vermeulen, D.; Watts, M.R. Coherent solid-state LIDAR with silicon photonic optical phased arrays. *Optics letters* **2017**, *42*, 4091–4094.
46. Behroozpour, B.; Sandborn, P.A.; Wu, M.C.; Boser, B.E. Lidar system architectures and circuits. *IEEE Communications Magazine* **2017**, *55*, 135–142.
47. Xu, X.; Zhang, L.; Yang, J.; Cao, C.; Wang, W.; Ran, Y.; Tan, Z.; Luo, M. A review of multi-sensor fusion slam systems based on 3D LIDAR. *Remote Sensing* **2022**, *14*, 1–27.
48. Li, Y.; Yu, A.W.; Meng, T.; Caine, B.; Ngiam, J.; Peng, D.; Shen, J.; Lu, Y.; Zhou, D.; Le, Q.V.; others. Deepfusion: LiDAR-camera deep fusion for multi-modal 3D object detection. IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), New Orleans, Louisiana, USA, 18–24 June 2022, pp. 17182–17191.
49. Zheng, W.; Xie, H.; Chen, Y.; Roh, J.; Shin, H. PIFNet: 3D object detection using joint image and point cloud features for autonomous driving. *Applied Sciences* **2022**, *12*.
50. Cui, Y.; Chen, R.; Chu, W.; Chen, L.; Tian, D.; Li, Y.; Cao, D. Deep learning for image and point cloud fusion in autonomous driving: A review. *IEEE Transactions on Intelligent Transportation Systems* **2021**, *23*, 722–739.
51. Du, X.; Ang, M.H.; Karaman, S.; Rus, D. A general pipeline for 3D detection of vehicles. 2018 IEEE International Conference on Robotics and Automation (ICRA), Brisbane, Australia, 21–25 May 2018, pp. 3194–3200.
52. Yang, Z.; Sun, Y.; Liu, S.; Shen, X.; Jia, J. Ipod: Intensive point-based object detector for point cloud. *arXiv preprint arXiv:1812.05276* **2018**.
53. Qi, C.R.; Liu, W.; Wu, C.; Su, H.; Guibas, L.J. Frustum pointnets for 3D object detection from RGB-D data. IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018, pp. 918–927.
54. Shin, K.; Kwon, Y.P.; Tomizuka, M. Roarnet: A robust 3D object detection based on region approximation refinement. IEEE intelligent vehicles symposium (IV), Paris, France, 9–12 June 2019, pp. 2510–2515.
55. Qi, C.R.; Su, H.; Mo, K.; Guibas, L.J. Pointnet: Deep learning on point sets for 3D classification and segmentation. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 21–26 July 2017, pp. 652–660.
56. Tran, D.; Bourdev, L.; Fergus, R.; Torresani, L.; Paluri, M. Learning spatiotemporal features with 3D convolutional networks. IEEE International Conference on Computer Vision (ICCV), Santiago, Chile, 2015, pp. 4489–4497.
57. Maturana, D.; Scherer, S. Voxnet: A 3D convolutional neural network for real-time object recognition. IEEE/RSJ international conference on intelligent robots and systems (IROS), Hamburg, Germany, 28 Sept–2 Oct 2015, pp. 922–928.
58. Xu, D.; Anguelov, D.; Jain, A. Pointfusion: Deep sensor fusion for 3D bounding box estimation. IEEE/CVF Conference on Computer Vision and Pattern Recognition, 18–23 June 2018, pp. 244–253.
59. Ku, J.; Mozifian, M.; Lee, J.; Harakeh, A.; Waslander, S.L. Joint 3D proposal generation and object detection from view aggregation. IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Madrid, Spain, 1–5 October 2018, pp. 1–8.
60. Liang, M.; Yang, B.; Wang, S.; Urtasun, R. Deep continuous fusion for multi-sensor 3D object detection. 15th European conference on computer vision (ECCV), Munich, Germany, 8–14 September 2018, pp. 641–656.

61. Sindagi, V.A.; Zhou, Y.; Tuzel, O. Mvx-net: Multimodal voxelnet for 3D object detection. *International Conference on Robotics and Automation (ICRA)*, Montreal, Canada, 20-24 May 2019, pp. 7276–7282.
62. Chen, X.; Ma, H.; Wan, J.; Li, B.; Xia, T. Multi-view 3D object detection network for autonomous driving. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 21-26 July 2017, pp. 1907–1915.
63. Geiger, A.; Lenz, P.; Stiller, C.; Urtasun, R. Vision meets robotics: The kitti dataset. *The International Journal of Robotics Research* **2013**, *32*, 1231–1237.
64. Sun, P.; Kretschmar, H.; Dotiwalla, X.; Chouard, A.; Patnaik, V.; Tsui, P.; Guo, J.; Zhou, Y.; Chai, Y.; Caine, B.; others. Scalability in perception for autonomous driving: Waymo open dataset. *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Seattle, WA, USA, 13-19 June 2020, pp. 2446–2454.
65. Geyer, J.; Kassahun, Y.; Mahmudi, M.; Ricou, X.; Durgesh, R.; Chung, A.S.; Hauswald, L.; Pham, V.H.; Mühlegg, M.; Dorn, S.; others. A2d2: Audi autonomous driving dataset. *arXiv preprint arXiv:2004.06320* **2020**.
66. Caesar, H.; Bankiti, V.; Lang, A.H.; Vora, S.; Liong, V.E.; Xu, Q.; Krishnan, A.; Pan, Y.; Baldan, G.; Beijbom, O. nuscenes: A multimodal dataset for autonomous driving. *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Seattle, WA, USA, 13-19 June 2020, pp. 11621–11631.
67. Cordts, M.; Omran, M.; Ramos, S.; Rehfeld, T.; Enzweiler, M.; Benenson, R.; Franke, U.; Roth, S.; Schiele, B. The cityscapes dataset for semantic urban scene understanding. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, NV, USA, 27-30 June 2016, pp. 3213–3223.
68. Ma, F.; Cavalheiro, G.V.; Karaman, S. Self-supervised sparse-to-dense: Self-supervised depth completion from lidar and monocular camera. *International Conference on Robotics and Automation (ICRA)*, Montreal, Canada, 20-24 May 2019, pp. 3288–3295.
69. Ma, F.; Karaman, S. Sparse-to-dense: Depth prediction from sparse depth samples and a single image. *IEEE international conference on robotics and automation (ICRA)*, Brisbane, Australia, 21-25 May 2018, pp. 4796–4803.
70. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, NV, USA, 27-30 June 2016, pp. 770–778.
71. Cheng, X.; Wang, P.; Yang, R. Depth estimation via affinity learned with convolutional spatial propagation network. *European Conference on Computer Vision (ECCV)*, Munich, Germany, 8-14 September 2018, pp. 103–119.
72. Cheng, X.; Wang, P.; Guan, C.; Yang, R. CSPN++: Learning context and resource aware convolutional spatial propagation networks for depth completion. *34th AAAI Conference on Artificial Intelligence*, New York, USA, 7-12 February 2020, pp. 10615–10622.
73. Cheng, X.; Zhong, Y.; Dai, Y.; Ji, P.; Li, H. Noise-aware unsupervised deep LiDAR-stereo fusion. *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Long Beach, CA, USA, 15-20 June 2019, pp. 6339–6348.
74. Jalal, A.S.; Singh, V. The state-of-the-art in visual object tracking. *Informatica* **2012**, *36*, 1–22.
75. Tang, P.; Wang, X.; Wang, A.; Yan, Y.; Liu, W.; Huang, J.; Yuille, A. Weakly supervised region proposal network and object detection. *15th European conference on computer vision (ECCV)*, Munich, Germany, 8-14 September 2018, pp. 352–368.
76. Uijlings, J.R.; Van De Sande, K.E.; Gevers, T.; Smeulders, A.W. Selective search for object recognition. *International Journal of Computer Vision* **2013**, *104*, 154–171.
77. Hong, M.; Li, S.; Yang, Y.; Zhu, F.; Zhao, Q.; Lu, L. SSPNet: Scale selection pyramid network for tiny person detection from UAV images. *IEEE Geoscience and Remote Sensing Letters* **2021**, *19*, 1–5.
78. Girshick, R. Fast R-CNN. *IEEE International Conference on Computer Vision (ICCV)*, Santiago, Chile, 7-13 December 2015, pp. 1440–1448.
79. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster R-CNN: Towards real-time object detection with region proposal networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **2015**, *39*, 1–9.
80. Kim, J.; Cho, J. Exploring a multimodal mixture-of-YOLOs framework for advanced real-time object detection. *Applied Sciences* **2020**, *10*.
81. Gupta, S.; Girshick, R.; Arbeláez, P.; Malik, J. Learning rich features from RGB-D images for object detection and segmentation. *13th European conference on computer vision (ECCV)*, Zurich, Switzerland, 6-12 September 2014, pp. 345–360.

82. Simonyan, K.; Zisserman, A. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556* **2014**.
83. Zhou, Y.; Tuzel, O. Voxnet: End-to-end learning for point cloud based 3D object detection. *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Salt Lake City, UT, USA,, 18-23 June 2018, pp. 4490–4499.
84. Meyer, G.P.; Charland, J.; Hegde, D.; Laddha, A.; Vallespi-Gonzalez, C. Sensor fusion for joint 3D object detection and semantic segmentation. *IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 16-17 June 2019, pp. 1–8.
85. Meyer, G.P.; Laddha, A.; Kee, E.; Vallespi-Gonzalez, C.; Wellington, C.K. Lasernet: An efficient probabilistic 3D object detector for autonomous driving. *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Long Beach, California, 15-20 June 2019, pp. 12677–12686.
86. Guo, Z.; Huang, Y.; Hu, X.; Wei, H.; Zhao, B. A survey on deep learning based approaches for scene understanding in autonomous driving. *Electronics* **2021**, *10*, 1–29.
87. Zoph, B.; Vasudevan, V.; Shlens, J.; Le, Q.V. Learning transferable architectures for scalable image recognition. *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Salt Lake City, UT, USA, 18-23 June 2018, pp. 8697–8710.
88. Valada, A.; Mohan, R.; Burgard, W. Self-supervised model adaptation for multimodal semantic segmentation. *International Journal of Computer Vision* **2020**, *128*, 1239–1285.
89. Caltagirone, L.; Bellone, M.; Svensson, L.; Wahde, M. LIDAR–camera fusion for road detection using fully convolutional neural networks. *Robotics and Autonomous Systems* **2019**, *111*, 125–131.
90. Dai, A.; Nießner, M. 3DMV: Joint 3D-multi-view prediction for 3D semantic scene segmentation. *European Conference on Computer Vision (ECCV)*, Munich, Germany, 8-14 September 2018, pp. 452–468.
91. Chiang, H.Y.; Lin, Y.L.; Liu, Y.C.; Hsu, W.H. A unified point-based framework for 3D segmentation. *International Conference on 3D Vision (3DV)*, Québec, Canada,, 16-19 September 2019, pp. 155–163.
92. Qi, C.R.; Yi, L.; Su, H.; Guibas, L.J. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. *Advances in Neural Information Processing Systems* **2017**, *30*, 1–10.
93. Jaritz, M.; Gu, J.; Su, H. Multi-view pointnet for 3D scene understanding. *IEEE/CVF International Conference on Computer Vision Workshop (ICCVW)*, Seoul, Korea (South), 27-28 October 2019, pp. 1–9.
94. Su, H.; Jampani, V.; Sun, D.; Maji, S.; Kalogerakis, E.; Yang, M.H.; Kautz, J. Splatnet: Sparse lattice networks for point cloud processing. *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Salt Lake City, UT, USA, 18-23 June 2018, pp. 2530–2539.
95. Hou, J.; Dai, A.; Nießner, M. 3D-SIS: 3D semantic instance segmentation of RGB-D scans. *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Long Beach, CA, USA, 15-20 June 2019, pp. 4421–4430.
96. Narita, G.; Seno, T.; Ishikawa, T.; Kaji, Y. Panopticfusion: Online volumetric semantic mapping at the level of stuff and things. *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Macau, China, 4-8 November 2019, pp. 4205–4212.
97. Elich, C.; Engelmann, F.; Kontogianni, T.; Leibe, B. 3D bird’s-eye-view instance segmentation. *41st DAGM German Conference on Pattern Recognition*, Dortmund, Germany, 10-13 September 2019, pp. 48–61.
98. Comaniciu, D.; Meer, P. Mean shift: A robust approach toward feature space analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **2002**, *24*, 603–619.
99. Kochanov, D.; Ošep, A.; Stückler, J.; Leibe, B. Scene flow propagation for semantic mapping and object discovery in dynamic street scenes. *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Daejeon, Korea, 9-14 October 2016, pp. 1785–1792.
100. Yue, Y.; Zhao, C.; Li, R.; Yang, C.; Zhang, J.; Wen, M.; Wang, Y.; Wang, D. A hierarchical framework for collaborative probabilistic semantic mapping. *IEEE international conference on robotics and automation (ICRA)*, Paris, France, 31 May - 31 August 2020, pp. 9659–9665.
101. Rosinol, A.; Violette, A.; Abate, M.; Hughes, N.; Chang, Y.; Shi, J.; Gupta, A.; Carlone, L. Kimera: From SLAM to spatial perception with 3D dynamic scene graphs. *The International Journal of Robotics Research* **2021**, *40*, 1510–1546.
102. Tian, Y.; Chang, Y.; Arias, F.H.; Nieto-Granda, C.; How, J.P.; Carlone, L. Kimera-multi: Robust, distributed, dense metric-semantic slam for multi-robot systems. *IEEE Transactions on Robotics* **2022**, *38*, 2022–2038.

103. Kim, U.H.; Park, J.M.; Song, T.J.; Kim, J.H. 3-D scene graph: A sparse and semantic representation of physical environments for intelligent agents. *IEEE Transactions on Cybernetics* **2019**, *50*, 4921–4933.
104. Rosinol, A.; Gupta, A.; Abate, M.; Shi, J.; Carlone, L. 3D dynamic scene graphs: Actionable spatial perception with places, objects, and humans. *arXiv preprint arXiv:2002.06289* **2020**.
105. Liu, H.; Yao, M.; Xiao, X.; Cui, H. A hybrid attention semantic segmentation network for unstructured terrain on Mars. *Acta Astronautica* **2023**, *204*, 492–499.
106. Humblot-Renaux, G.; Marchegiani, L.; Moeslund, T.B.; Gade, R. Navigation-oriented scene understanding for robotic autonomy: learning to segment driveability in egocentric images. *IEEE Robotics and Automation Letters* **2022**, *7*, 2913–2920.
107. Badrinarayanan, V.; Kendall, A.; Cipolla, R. Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **2017**, *39*, 2481–2495.
108. Guan, T.; Kothandaraman, D.; Chandra, R.; Sathiamoorthy, A.J.; Weerakoon, K.; Manocha, D. GA-Nav: Efficient terrain segmentation for robot navigation in unstructured outdoor environments. *IEEE Robotics and Automation Letters* **2022**, *7*, 8138–8145.
109. Wigness, M.; Eum, S.; Rogers, J.G.; Han, D.; Kwon, H. A RUGD dataset for autonomous navigation and visual perception in unstructured outdoor environments. *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Macau, China, 4–8 November 2019, pp. 5000–5007.
110. Jiang, P.; Osteen, P.; Wigness, M.; Saripalli, S. RELIS-3D dataset: Data, benchmarks and analysis. *IEEE international conference on robotics and automation (ICRA)*, Xi'an, China, May 31 - June 4 2021, pp. 1110–1116.
111. Ma, L.; Stückler, J.; Kerl, C.; Cremers, D. Multi-view deep learning for consistent semantic mapping with RGB-D cameras. *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Vancouver, BC, Canada, 24–28 September 2017, pp. 598–605.
112. Hazirbas, C.; Ma, L.; Domokos, C.; Cremers, D. Fusetnet: Incorporating depth into semantic segmentation via fusion-based cnn architecture. *13th Asian Conference on Computer Vision*, Taipei, Taiwan, 20–24 November 2016, pp. 213–228.
113. Zhang, J.; Henein, M.; Mahony, R.; Ila, V. VDO-SLAM: A visual dynamic object-aware SLAM system. *arXiv preprint arXiv:2005.11052* **2020**.
114. Maturana, D.; Scherer, S. Voxnet: A 3D convolutional neural network for real-time object recognition. *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Hamburg, Germany, Sept 28 - Oct 2 2015, pp. 922–928.
115. Huang, J.; You, S. Point cloud labeling using 3D convolutional neural network. *23rd International Conference on Pattern Recognition (ICPR)*, Cancun, Mexico, 4–8 December 2016, pp. 2670–2675.
116. LeCun, Y.; Bottou, L.; Bengio, Y.; Haffner, P. Gradient-based learning applied to document recognition. *Proceedings of the IEEE* **1998**, *86*, 2278–2324.
117. Song, S.; Yu, F.; Zeng, A.; Chang, A.X.; Savva, M.; Funkhouser, T. Semantic scene completion from a single depth image. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Honolulu, Hawaii, 21–26 July 2017, pp. 1746–1754.
118. Riegler, G.; Osman Ulusoy, A.; Geiger, A. OctNet: Learning deep 3D representations at high resolutions. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Honolulu, Hawaii, 21–26 July 2017, pp. 3577–3586.
119. Tatarchenko, M.; Park, J.; Koltun, V.; Zhou, Q.Y. Tangent convolutions for dense prediction in 3D. *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Salt Lake City, Utah, USA, 18–23 June 2018, pp. 3887–3896.
120. Wang, F.; Yang, Y.; Wu, Z.; Zhou, J.; Zhang, W. Real-time semantic segmentation of point clouds based on an attention mechanism and a sparse tensor. *Applied Sciences* **2023**, *13*.
121. Wu, W.; Qi, Z.; Fuxin, L. Pointconv: Deep convolutional networks on 3D point clouds. *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Long Beach, CA, USA, 15–20 June 2019, pp. 9621–9630.
122. Hua, B.S.; Tran, M.K.; Yeung, S.K. Pointwise convolutional neural networks. *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Salt Lake City, UT, USA, 18–23 June 2018, pp. 984–993.
123. Zamorski, M.; Zięba, M.; Klukowski, P.; Nowak, R.; Kurach, K.; Stokowiec, W.; Trzciński, T. Adversarial autoencoders for compact representations of 3D point clouds. *Computer Vision and Image Understanding* **2020**, *193*, 1–8.



124. Ye, X.; Li, J.; Huang, H.; Du, L.; Zhang, X. 3D recurrent neural networks with context fusion for point cloud semantic segmentation. 15th European Conference on Computer Vision (ECCV), Munich, Germany, 8-14 September 2018, pp. 403–417.
125. Wang, Y.; Sun, Y.; Liu, Z.; Sarma, S.E.; Bronstein, M.M.; Solomon, J.M. Dynamic graph cnn for learning on point clouds. *ACM Transactions on Graphics* **2019**, *38*, 1–12.
126. Qi, X.; Liao, R.; Jia, J.; Fidler, S.; Urtasun, R. 3D graph neural networks for RGB-D semantic segmentation. IEEE International Conference on Computer Vision (ICCV), Venice, Italy, 22-29 October 2017, pp. 5199–5208.
127. Landrieu, L.; Simonovsky, M. Large-scale point cloud semantic segmentation with superpoint graphs. IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18-23 June 2018, pp. 4558–4567.
128. Li, J.; Chen, B.M.; Lee, G.H. SO-Net: Self-organizing network for point cloud analysis. IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18-23 June 2018, pp. 9397–9406.
129. Thrun, S. Probabilistic robotics. *Communications of the ACM* **2002**, *45*, 52–57.
130. Siegwart, R.; Nourbakhsh, I.R.; Scaramuzza, D. *Introduction to autonomous mobile robots*, 2nd ed.; MIT press, 2011.
131. Sivakumar, A.N.; Modi, S.; Gasparino, M.V.; Ellis, C.; Velasquez, A.E.B.; Chowdhary, G.; Gupta, S. Learned visual navigation for under-canopy agricultural robots. 17th Robotics: Science and Systems, 12-16 July 2021.
132. Atas, F.; Grimstad, L.; Cielniak, G. Evaluation of sampling-based optimizing planners for outdoor robot navigation. *arXiv preprint arXiv:2103.13666* **2021**.
133. Wang, X.; Shi, Y.; Ding, D.; Gu, X. Double global optimum genetic algorithm–particle swarm optimization-based welding robot path planning. *Engineering Optimization* **2016**, *48*, 299–316.
134. Zhu, S.; Zhu, W.; Zhang, X.; Cao, T. Path planning of lunar robot based on dynamic adaptive ant colony algorithm and obstacle avoidance. *International Journal of Advanced Robotic Systems* **2020**, *17*, 1–14.
135. Mac, T.T.; Copot, C.; Tran, D.T.; De Keyser, R. A hierarchical global path planning approach for mobile robots based on multi-objective particle swarm optimization. *Applied Soft Computing* **2017**, *59*, 68–76.
136. Ghita, N.; Kloetzer, M. Trajectory planning for a car-like robot by environment abstraction. *Robotics and Autonomous Systems* **2012**, *60*, 609–619.
137. Zhu, Y.; Mottaghi, R.; Kolve, E.; Lim, J.J.; Gupta, A.; Fei-Fei, L.; Farhadi, A. Target-driven visual navigation in indoor scenes using deep reinforcement learning. IEEE international conference on robotics and automation (ICRA), Singapore, 29 May - 3 June 2017, pp. 3357–3364.
138. Wijmans, E.; Kadian, A.; Morcos, A.; Lee, S.; Essa, I.; Parikh, D.; Savva, M.; Batra, D. Dd-ppo: Learning near-perfect pointgoal navigators from 2.5 billion frames. *arXiv preprint arXiv:1911.00357* **2019**.
139. Gupta, S.; Davidson, J.; Levine, S.; Sukthankar, R.; Malik, J. Cognitive mapping and planning for visual navigation. *International Journal of Computer Vision* **2020**, *128*, 1311–1330.
140. Datta, S.; Maksymets, O.; Hoffman, J.; Lee, S.; Batra, D.; Parikh, D. Integrating egocentric localization for more realistic point-goal navigation agents. 4th Conference on Robot Learning (CoRL), 16 - 18 November 2020, pp. 313–328.
141. Kumar, A.; Gupta, S.; Fouhey, D.; Levine, S.; Malik, J. Visual memory for robust path following. *Advances in Neural Information Processing Systems* **2018**, *31*, 1–10.
142. Pan, Y.; Cheng, C.A.; Saigol, K.; Lee, K.; Yan, X.; Theodorou, E.; Boots, B. Agile autonomous driving using end-to-end deep imitation learning. *arXiv preprint arXiv:1709.07174* **2017**.
143. Sadeghi, F.; Levine, S. CAD2RL: Real single-image flight without a single real image. Robotics: Science and Systems, Cambridge, Massachusetts, USA, 12-16 July 2017.
144. Ross, S.; Melik-Barkhudarov, N.; Shankar, K.S.; Wendel, A.; Dey, D.; Bagnell, J.A.; Hebert, M. Learning monocular reactive uav control in cluttered natural environments. IEEE International Conference on Robotics and Automation (ICRA), Karlsruhe, Germany, 6-10 May 2013, pp. 1765–1772.
145. Gandhi, D.; Pinto, L.; Gupta, A. Learning to fly by crashing. IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Vancouver, BC, Canada, 24-28 September 2017, pp. 3948–3955.
146. Gasparino, M.V.; Sivakumar, A.N.; Liu, Y.; Velasquez, A.E.; Higuti, V.A.; Rogers, J.; Tran, H.; Chowdhary, G. Wayfast: Navigation with predictive traversability in the field. *IEEE Robotics and Automation Letters* **2022**, *7*, 10651–10658.



147. Sathyamoorthy, A.J.; Weerakoon, K.; Guan, T.; Liang, J.; Manocha, D. TerraPN: Unstructured terrain navigation using online self-supervised learning. *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Kyoto, Japan, 23-27 October 2022, pp. 7197–7204.
148. Hirose, N.; Shah, D.; Sridhar, A.; Levine, S. ExAug: Robot-conditioned navigation policies via geometric experience augmentation. *arXiv preprint arXiv:2210.07450* **2022**.
149. Heimann, D.; Hohenfeld, H.; Wiebe, F.; Kirchner, F. Quantum deep reinforcement learning for robot navigation tasks. *arXiv preprint arXiv:2202.12180* **2022**.
150. Gyaganda, N.; Hatilima, J.V.; Roth, H.; Zhmud, V. A review of GNSS-independent UAV navigation techniques. *Robotics and Autonomous Systems* **2022**, *152*, 1–17.
151. Zhu, K.; Zhang, T. Deep reinforcement learning based mobile robot navigation: A review. *Tsinghua Science and Technology* **2021**, *26*, 674–691.
152. Li, Y. Deep reinforcement learning: An overview. *arXiv preprint arXiv:1701.07274* **2017**.
153. Li, H.; Zhang, Q.; Zhao, D. Deep reinforcement learning-based automatic exploration for navigation in unknown environment. *IEEE transactions on Neural Networks and Learning Systems* **2019**, *31*, 2064–2076.
154. Wu, J.; Ma, X.; Peng, T.; Wang, H. An improved timed elastic band (TEB) algorithm of autonomous ground vehicle (AGV) in complex environment. *Sensors* **2021**, *21*, 1–12.
155. Kulhánek, J.; Derner, E.; De Bruin, T.; Babuška, R. Vision-based navigation using deep reinforcement learning. *European Conference on Mobile Robots (ECMR)*, Prague, Czech Republic, 4-6 September 2019, pp. 1–8.
156. Xi, A.; Mudiyansele, T.W.; Tao, D.; Chen, C. Balance control of a biped robot on a rotating platform based on efficient reinforcement learning. *IEEE/CAA Journal of Automatica Sinica* **2019**, *6*, 938–951.
157. Lee, K.; Vlahov, B.; Gibson, J.; Rehg, J.M.; Theodorou, E.A. Approximate inverse reinforcement learning from vision-based imitation learning. *IEEE International Conference on Robotics and Automation (ICRA)*, Xi'an, China, 31 May - 4 June 2021, pp. 10793–10799.
158. Qi, W.; Mullapudi, R.T.; Gupta, S.; Ramanan, D. Learning to move with affordance maps. *arXiv preprint arXiv:2001.02364* **2020**.
159. Zende, O.; Honauer, K.; Murschitz, M.; Steininger, D.; Dominguez, G.F. Wilddash-creating hazard-aware benchmarks. *15th European Conference on Computer Vision (ECCV)*, Munich, Germany, 8-14 September 2018, pp. 402–416.
160. Tang, J.; Chen, Y.; Kukko, A.; Kaartinen, H.; Jaakkola, A.; Khoramshahi, E.; Hakala, T.; Hyypä, J.; Holopainen, M.; Hyypä, H. SLAM-aided stem mapping for forest inventory with small-footprint mobile LiDAR. *Forests* **2015**, *6*, 4588–4606.
161. Chen, W.; Shang, G.; Ji, A.; Zhou, C.; Wang, X.; Xu, C.; Li, Z.; Hu, K. An overview on visual SLAM: From tradition to semantic. *Remote Sensing* **2022**, *14*, 1–47.
162. Chghaf, M.; Rodriguez, S.; Ouardi, A.E. Camera, LiDAR and multi-modal SLAM systems for autonomous ground vehicles: A survey. *Journal of Intelligent & Robotic Systems* **2022**, *105*, 1–35.
163. Xue, H.; Hein, B.; Bakr, M.; Schildbach, G.; Abel, B.; Rueckert, E. Using deep reinforcement learning with automatic curriculum learning for mapless navigation in intralogistics. *Applied Sciences* **2022**, *12*.

**Disclaimer/Publisher's Note:** The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.