












Survey Article

A Survey on Terrain Traversability Analysis for Autonomous Ground Vehicles: Methods, Sensors, and Challenges

Paulo V. K. Borges¹, Thierry Peynot², Sisi Liang¹, Bilal Arain²,
Matthew Wildie¹, Melih G. Minareci², Serge Lichman¹, Garima Samvedi²,
Inkyu Sa¹, Nicolas Hudson³, Michael Milford², Peyman Moghadam¹ and
Peter Corke²

¹CSIRO, Brisbane, Australia

²Queensland University of Technology, Brisbane, Australia

³Amazon, Seattle, Washington

Abstract: Understanding the terrain in the upcoming path of a ground robot is one of the most challenging problems in field robotics. Terrain and traversability analysis is a multidisciplinary field combining robotics with image and signal processing, feature extraction, machine learning, three-dimensional (3D) mapping, and 3D geometry. Application scenarios range from autonomous vehicles on urban networks to agriculture, defence, exploration, mining, and search and rescue. Given the broad set of techniques available and the fast progress in this area, in this paper we organize and survey the corresponding literature, define unambiguous key terms, and discuss links among fundamental building blocks ranging from terrain classification to traversability regression. The advantages and the drawbacks of the methods are critically discussed, providing a comprehensive coverage of key aspects, including open code, available datasets for experimentation and comparisons, and important open research issues.

Keywords: perception, obstacle avoidance, terrestrial robotics, navigation

1. Introduction

In recent years, the robotics community has seen a major increase in the number of outdoor applications for autonomous vehicles. Undoubtedly the chief commercial push for outdoor robotics has been that of autonomous cars on urban networks. However, there are numerous other fields that can benefit from ground mobile robots. They include agriculture, search and rescue, mining, defence, environmental monitoring, planetary exploration, and oil and gas, among others. All of those examples share one common requirement: the ability to navigate on off-road terrain. In our

Received: 13 August 2021; revised: 10 February 2022; accepted: 9 May 2022; published: 14 July 2022.

Correspondence: Paulo V. K. Borges, CSIRO, Brisbane, Australia, Email: paulo.borges@csiro.au

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

Copyright © 2022 Borges, Peynot, Liang, Arain, Wildie, Minareci, Lichman, Samvedi, Sa, Hudson, Milford, Moghadam and Corke

DOI: <https://doi.org/10.55417/fr.2022049>

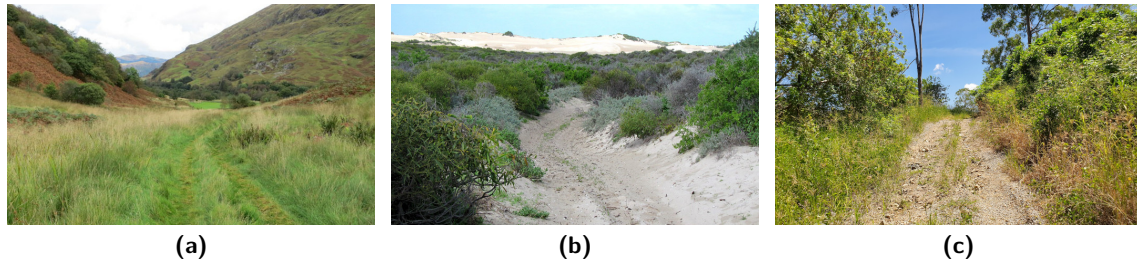


Figure 1. Examples of “off-road” regions as used in this paper, with (a) grass, (b) sand, and (c) gravel surfaces.

discussion, we define off-road as any terrain that is not a part of a paved road network. Some examples are shown in Figure 1, where various types of terrain (grass, gravel, sand) are illustrated.

In off-road navigation, in many cases the robot will also encounter *variable* terrain types (e.g., concrete, grass, mud, sand, puddles, pebbles) in its “mission,” adding the ability to adapt online to the specifications.

Hence, the capability of automatically understanding terrain types and understanding the upcoming path characteristics is a key functionality of intelligent outdoors uncrewed ground vehicles (UGVs).

A number of processing steps are necessary to analyze the scene at different levels of abstraction, ranging from interpreting the geometry to interpreting the appearance of the ground of interest, in addition to proprioceptive sensing. An arguably crucial step consists in organizing range and appearance information coherently, such that a meaningful representation of the terrain is generated.

Within an off-road environment, UGVs need to solve several common challenges, which include an estimation of terrain traversability and determining suitable and efficient paths according to a given criterion (e.g., distance, energy consumption, time) while respecting the kinematics and dynamics limits of the physical platform. In short, identifying whether a region in the upcoming potential path of the robot is traversable (dependent on the type of platform) remains one of the greatest challenges in robotics. At a high level, the problem may combine image and signal processing, feature extraction, machine learning, and three-dimensional (3D) geometry. Given the large number of methods used in terrain understanding and the continuously evolving landscape, in this paper we propose to organize and survey the corresponding literature. We define unambiguously key terms that are often used interchangeably in the literature, and we discuss links among key building blocks ranging from semantic classification to traversability for UGVs. The advantages and the drawbacks of the methods are critically discussed, providing comprehensive coverage of the main aspects of existing methods.

Excellent prior publications have reviewed the area of terrain traversability analysis in the past (Papadakis, 2013; Sancho-Pradel and Gao, 2010), where the authors presented the literature and suggested a taxonomy for the field (Papadakis, 2013). For planetary terrains, a comprehensive review of methods is provided in (Chhaniyara et al., 2012), which finds application in dry and rocky surfaces on which planetary rovers operate. Advancements focused on learning methods for perception and navigation in unstructured environments have also been recently compiled (Guastella and Muscato, 2021), with urban “counterparts” presented in the surveys in (Grigorescu et al., 2020) and (Ni et al., 2020), which focus on deep-learning for autonomous cars in road networks. A survey on sensor fusion methods for obstacle detection that finds application in off-road navigation has also been published (Hu et al., 2020). Specifically, the work in (Guastella and Muscato, 2021) is divided between regression/classification and end-to-end methods that directly map environmental perception into control actions, which is an important and more recent paradigm.

Complementing the studies mentioned above, in this paper we link many of the recent advancements presented in the literature, particularly in machine learning and semantics, with classical statistical methods presented previously. Although partial overlap with previous surveys inevitably

exists, we bring different sensor modalities and fusion options from structured or semistructured environments, and we highlight their relevance for nonurban navigation and traversability. Apart from (Papadakis, 2013), which is broader, many of the previous review papers touch on specific scenarios or elements (e.g., planetary, urban, learning) and have a more focused discussion on definitions for the field. We have added a more complete section on definitions and taxonomy that includes many elements that have not been unified in a single discussion and are all relevant for modern terrain analysis in robotics (which is often a point of confusion in our experience). We also draw attention to, and characterize major challenges (from vegetation to the presence of dust, mud, fog, and negative obstacles) frequently seen, in off-road navigation. We review a number of recent datasets and open code (including some that are urban-focused and can be useful for off-road) that can be beneficial for researchers working in the domain of traversability analysis for ground robots. Throughout the paper, we sometimes use the terms “robot” and “vehicle” interchangeably, with the choice of words depending on the context of the original publication. The reader will notice that some of the methods presented are described as off-line or below real time in their original publications. Despite this limitation, they are included as we believe there is merit in the concept raised by some of those algorithms, and it brings completeness to this paper. We have seen examples of elegant methods that were not real-time a decade ago that today can work in real time, thanks to increased processing power or better implementations (as opposed to “research code,” for example).

Arguably the two main exteroceptive sensing modalities that have been most extensively used for terrain analysis are visual camera and LIDAR (Light Detection and Ranging) sensors. Obviously, LIDAR and vision (including stereo) have their unique properties. LIDAR can provide relatively reliable, consistent, and precise range measurements compared to stereo vision, but it still has limitations regarding sparse point density, active nature, and (generally) higher cost. Stereo vision is cheaper and provides information-rich structured data (i.e., images). On the other hand, vision is sensitive to lighting conditions and calibration, especially for stereo pairs. Due to the aforementioned sensor-dependent characteristics, it is recommended to carefully select the most suitable sensor or fusion approach depending on applications. A number of off-the-shelf sensors are available, and cheaper and better performing sensors (e.g., less noise, better range) are being constantly released in the market.

Hybrid sensors (e.g., RGB-D) are also one of the promising and emerging sensing technologies, and they may be able to moderate LIDAR and stereo vision sensors’ downsides, but they often suffer in outdoor operations due to natural light interference in the active vision.

In many cases, data from those two modalities have been fused for improved performance. Analogously, detection and classification of terrain has used model-based and learning-based approaches. Based on these circumstances, the remainder of the paper is organized according to the structure shown in Figure 2. In this figure, the Roman numerals indicate the section number of each topic. In Section 2 we pinpoint the key components of terrain classification methods and suggest a top-down view of the field through a high-level taxonomy. In Section 3 we review vision-based methods, followed by LIDAR-based methods in Section 4. Section 5 is dedicated to other exteroceptive sensors, while the use of proprioceptive sensing is discussed in Section 6. Sensor-fusion-based methods are discussed in Section 7. In Section 8 we examine major practical challenges that can still limit the application of some of the methods available in the literature. Section 9 reviews available datasets and open software related to terrain analysis. Finally, in Section 10, we provide relevant conclusions and new research directions.

2. Definitions and Taxonomy

In this section, we define common terminology used in a number of aspects of terrain analysis for UGVs. The goal is to provide a consistent set of definitions for terms that are sometimes ambiguously employed in the literature. This discussion should assist the reader in the understanding of this paper, and hopefully serve as a future reference.

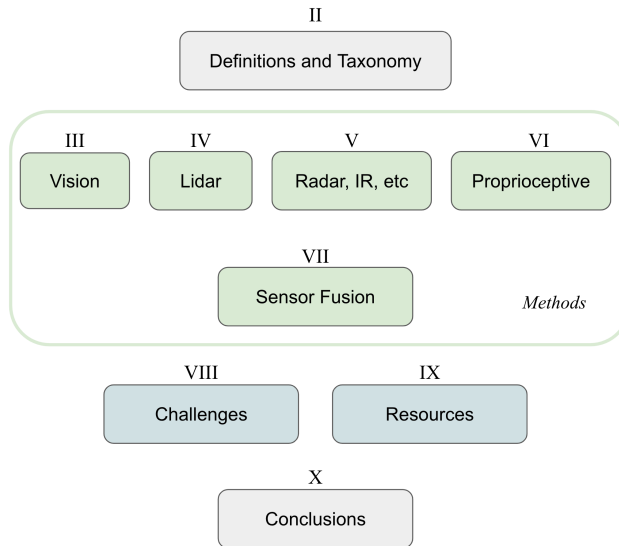


Figure 2. Organizational structure of this paper around the topic of terrain traversability analysis.

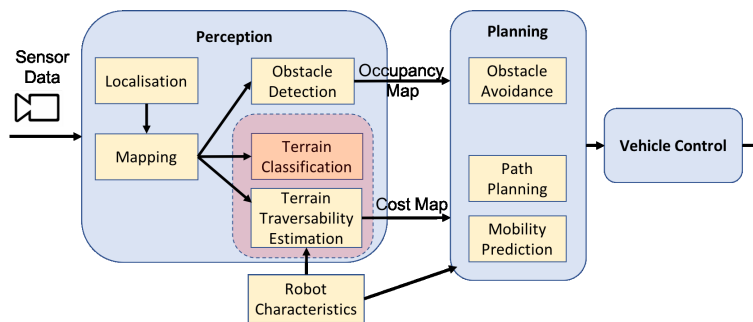


Figure 3. Typical components of a robotic system, with different configurations considered. This paper discusses methods to interpret sensor data, often accumulated in a common spatial representation (map), for the purpose of planning, ensuring safe navigation. Three main strategies are typically used: 1) obstacle detection, 2) terrain classification, or 3) terrain traversability estimation. Among the many different elements that form a robot, in this paper we mostly focus on terrain classification and terrain traversability, as highlighted by the “dashed” pink box.

2.1. Robot Navigation

In the context of mobile robotics, *terrain classification* sits side-by-side with *obstacle detection*, which we can combine into the topic of *traversability analysis* if robot characteristics (e.g., kinematic and dynamic models) are considered. At a very high level, the functionality flow for a robot typically follows the order: (i) localization, (ii-a) obstacle detection, (ii-b) terrain classification, (iii) path planning, and (iv) vehicle control.

Technology has advanced tremendously for wheeled and tracked UGVs for most of those functions (particularly localization and control), but effective terrain traversability analysis still remains partly unsolved for very challenging environments.

Figure 3 illustrates a typical example of architecture used for off-road robot navigation. Sensor data (left) are gathered and combined as needed in a common spatial representation. An example is the generation of digital elevation or terrain maps (DEM/DTM) (Goldberg et al., 2002).

One of three different types of operation is then usually conducted. 1) obstacle detection: given the map and the type of vehicle (e.g., its spatial dimensions), it produces a binary obstacle map, or *occupancy map*, which is used by a planner for simple obstacle avoidance. 2) Terrain classification

produces segments associated with particular classes of terrain, often arbitrarily predefined, e.g., rock, grass, water. 3) Terrain traversability takes into account the map content and the capabilities of the robot to produce a cost/difficulty map, representing how difficult it may be for the robot to traverse particular areas of the terrain. This cost map is then passed on to the planner to determine the best compromise between its objectives and the associated costs.

More specifically, we use the following definitions for the key terms in the context of navigation:

1. *Obstacle detection*: the task of identifying the presence (or otherwise absence) of an obstacle in a map. An obstacle is a location or area in the environment that is considered impossible or unsafe for a robot to traverse through. This is a perception-focused task. A special class of obstacles, referred to as *negative obstacles* in the literature, represents holes and depressions in the environment (Matthies and Rankin, 2003). This term was introduced to distinguish those obstacles from most common obstacles that are “positive,” in the sense that they typically lie above the ground level (e.g., large rocks or trees).
2. *Obstacle avoidance*: the task of, once an obstacle has been detected, determining a feasible path around the obstacle, if any. The task combines perception and local path planning.
3. *Terrain classification*: the task of determining which type of terrain (e.g., grass, asphalt, rocks, water) is present in the environment, generally in a semantic sense. The task usually combines perception and machine learning.
4. *Traversability analysis*: this task usually interprets the terrain at a more refined level than obstacle detection, contrasting its characteristics against the dynamics and kinematics capabilities of the robot, and generating a “difficulty” or cost map of the environment. Such a cost map is usually a 2D representation that indicates a cost value for each location (x, y) (optionally with an orientation). The analysis can include various elements, such as terrain geometry, rugosity, expected friction/traction, and kinematics of the vehicle. A simple traversability map with only two cost/difficulty values possible (traversable, not traversable) is essentially an obstacle map.

Figure 4 graphically illustrates the concepts above for a robot traveling from Point A to Point B in scenarios I–V. The leftmost column containing A–B shows a side view of the environment, showing free space, a generic obstacle, tall grass, a tall tree, and a deep pond, for scenarios I–V, respectively.

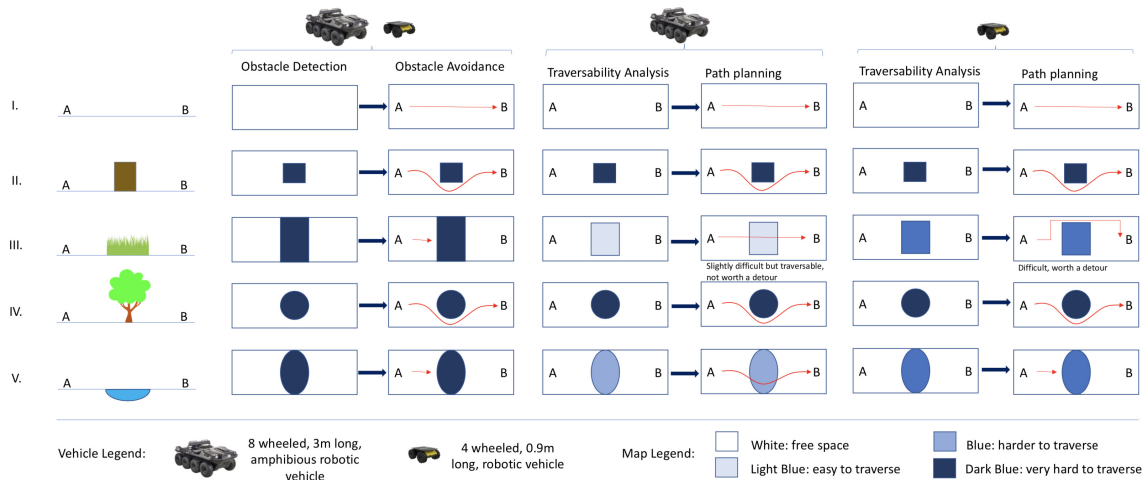


Figure 4. Illustration of the different paths planned (in red) for different robots, different environments, and different tasks (i.e., obstacle detection vs obstacle avoidance, vs. terrain analysis, vs. path planning). Rows III and V show how the same detected area in a map can potentially be traversed by some platforms and not by others; therefore, the planned path will be different for different robots.

The illustration inside the rectangles represent the corresponding abstract top-view occupancy maps, where the blue components represent the obstacles, and the different shades of blue represent how they are interpreted in different modules. For example, light blue means “easy to traverse” while dark blue means “very hard to traverse.” On top of the figure, we illustrate two different types of vehicles: the larger black robot¹ has amphibious capability and has a high clearance that allows it to go over challenging terrain, including tall grass. The small yellow robot² is a small standard exploration robot with low clearance. For all the hypothetical cases described, the achieved paths are shown in red in the top-view plots for the different tasks (obstacle detection vs. obstacle avoidance vs. terrain analysis) and robots (large amphibious robot vs. small regular robot). We can see that the success in planning a path is different depending on the platform and the environment. We can also see the obstacle detection and traversability analysis task focus on understanding the terrain, but not on planning the path. Although it is purely abstract, the goal of this figure is to present an illustrated exercise that assists us in coherently defining the terminology.

2.2. Terrain Analysis, Classification, and Traversability

Before traversability can be discussed, the observed terrain needs to be interpreted. This can involve one or more of the following tasks, depending on the focus of the research/application:

1. *Detection* refers to finding an occurrence of something, which can be of different levels of abstraction or specificity. As an example, “obstacle detection” is generic, while “water detection” is more specific.
2. *Segmentation* separates and clusters different components of a scene into their section or class, without specifying what type of class.
3. *Classification* labels detection or segmentation outputs into one or more potential classes. Binary classification only determines which segments belong to a single class or not.

When performing any of the tasks mentioned above (e.g., classification, detection) there exist some common strategies regarding the design and architecture of the system. In this work, we use the following definitions, but note that terminology is not uniform across fields; for example, the word “model” can mean either something that is at least partially learnt in robotic learning research or a non-learning-based method in other fields:

1. *Model-based method*: an approach that is based on a hand-crafted, and usually explicit, model. Examples include (i) a model of the appearance of water in an image based on the laws of physics, (ii) a “yellow” appearance characterization for sand, or (iii) an explicit and arbitrary criterion that a robot cannot drive over a slope of more than 20 degrees of inclination. Note that specific aspects (e.g., parameters) of the model are often learned from experimental data.
2. *Data-driven or learning method*: an approach that is learning directly from data. Although such an approach would often be using an underlying model as well, this model is learned from data.
3. *Deep learning vs. traditional learning*: in this paper we distinguish *traditional learning* (e.g., Support Vector Machines) from more recent *deep learning* methods that exploit deep neural networks.
4. *Supervised (learning) method*: a data-driven method that learns from provided examples, i.e., using training data that include ground truth labels, usually provided by experts, prior to training.
5. *Semisupervised (learning) method*: an approach that combines a limited amount of labeled data with a large amount of unlabeled data during the training phase. This method is useful in applications where labeling is complex and time-intensive.

¹ Figure and vehicle characteristics extracted from <https://argoatv.com.au/>

² Figure and vehicle characteristics extracted from <https://clearpathrobotics.com/>

6. *Weakly supervised (learning) method*: an approach in which a model is trained on noisy, partially annotated data at a coarse level. This method is useful when there are multiple sources of weak supervision often generated using simple heuristic rules or relying on a more reliable modality to create pseudolabels.
7. *Unsupervised (learning) method*: an approach in which the learning model is given a dataset without specific instructions on what the output should be. The method then aims at automatically extracting patterns or structure in the data by finding meaningful features.
8. *Self-supervised (learning) method*: a more recent paradigm, where a model is trained on unlabeled data by defining a pretext task to learn robust representations and then fine-tune the model to perform specific tasks such as classification, segmentation, or object detection.

The above definitions apply across a number of domains, but they are frequently found in the terrain analysis literature.

2.3. Sensors

Two main categories of sensing are typically used for terrain analysis: *exteroceptive* and *proprioceptive*. Although in this paper we focus mostly on the use of exteroceptive sensing to *predict* what type of terrain the robot is going to face ahead, we also consider the important, and often complementary, role that proprioception has played in the literature.

As most robotic perception research work is based on exteroception, most terrain traversability analysis approaches are based on the interpretation of data from visual cameras and/or LIDARs, as reflected in Sections 3 and 4. Therefore, in this paper, we refer to other sensors as *alternative sensors*, as defined in (Peynot et al., 2015) (see Section 5). In the context of this survey, this concerns mostly sensors such as RADARs, infrared (IR), or hyperspectral cameras.

In terrain analysis, proprioception is mainly used in two ways. The first is to characterize the terrain found under the robot’s body at the *present* moment, for example identifying slippery terrain based on wheel odometry observations, or rugged terrain based on the vibration signature observed with an inertial measurement unit (IMU). The second is to teach a system to predict the behavior of the robot when it *will* be over a certain patch of terrain observed with exteroception. This concept is sometimes referred to as *near-to-far learning* (Stella et al., 2021; Howard et al., 2006; Krebs et al., 2010).

2.4. Traversability Metrics

A variety of metrics have been used in the literature to define the *traversability* of an area of terrain, which is the cost of traveling over that area. In general, traversability for a mobile robot is a function of the terrain shape and the terramechanic characteristics (e.g., soil properties and their interaction with the robot wheels/tracks) (Ishigami et al., 2007; Ishigami et al., 2006). It can also include the state of the robot (Martin, 2018) since the robot’s speed (and orientation) can also play a role.

The terrain shape can be simplified by the slope and roughness of the terrain, as identified by early research on terrain analysis that extracted basic terrain statistics (e.g., variance and slope of patches in front of the robot) to determine a cost (Langer et al., 1994; Gennery, 1999; Hamner et al., 2008). Roughness is defined as a measure of the small-scale variations in the height of a surface. It is well understood to be related to traversability (Bekker, 1969). Mathematically (and considering 3D or depth data), roughness is related to how scattered or linear/planar is the distribution of the points in the area of interest. This can be done using a number of statistical methods, such as least-squares plane fitting computing the residuals (Krüsi et al., 2017; Gennery, 1999), Gaussian mixture models, and principal component analysis over the terrain points (Lalonde et al., 2006). Figure 5 illustrates the concept for 2D LIDAR returns for a vehicle-sized patch (in this example, we used a Baraja LIDAR³ for the scan). On the top image, the terrain is considered “smoother”

³ <https://www.baraja.com/>

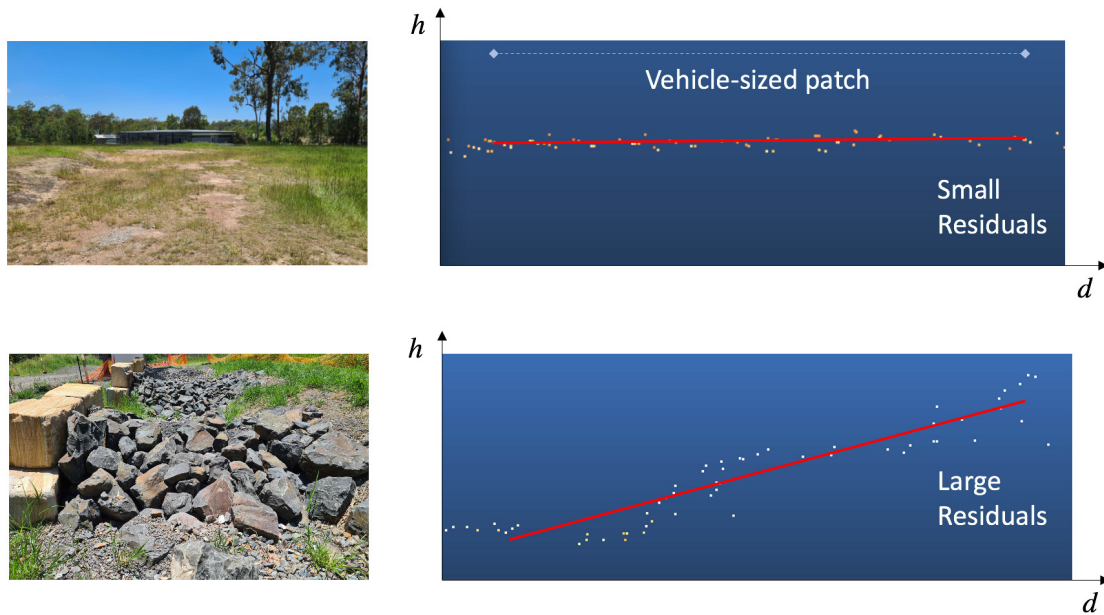


Figure 5. Illustration of the side view of LIDAR returns from a Baraja LIDAR for two different terrains. Please note that on the scans on the right, the points projected to the side 2D view are not from a sliced plane, but rather from a very narrow volume, hence some points appear behind others (which would not be possible for the LIDAR to sense if all points were on exactly the same plane). d corresponds to the distance from the sensor, and h is the height of the points with respect to the sensor frame. Here, d and h are not to scale and serve only as an abstract representation. The top scan shows a “smoother” terrain, with little roughness (small residuals for the red line fitting) on flat ground, corresponding to the grassy image on the top left. The bottom scan represents a rougher terrain (larger residuals for the red line fitting) with more inclination, corresponding to the rocky image on the bottom left. Arguably the terrain in the top image is easier to traverse than that in the bottom image.

than in the bottom image, based on the residuals and slope of the LIDAR returns. Arguably the distribution of points in the top figure represents a more easily traversable terrain than the points in the bottom figure. Regarding slope, the concept should obviously consider both pitch and roll, as both can interfere with traversability. A vehicle’s roll and pitch is a function of the slope of both the local environment and the configuration of the robot’s suspension, with some robots able to actively articulate their suspension to accommodate terrain slopes.

An interesting study in (Molino et al., 2007) further refines costs into cost of coverability and cost of crossability. The coverability cost of some region is a measure of how difficult it is for a robot to explore *all* sections of that region. Crossability of some region is related to the cost of going from Point A to Point B in that region. This is equivalent to our definition of traversability (and consistent with most of the literature), as discussed in Figure 4.

It is always important to distinguish between traversability metrics that are just dependent on the terrain (like the slope and roughness above) and those that are platform-dependent/specific, such as the kinematics of the rover chassis when placed on the terrain location [e.g., (Bonnafous et al., 2001)]. The traversability can also be represented probabilistically, where sensor uncertainty and multiple risk factors (e.g., tip-over, collision, and nearby steps and objects) are fused into a final metric (Fan et al., 2021).

2.5. Performance Evaluation

Although the metrics described in Section 2.4 provide the general approach used in quantifying terrain characteristics, in the literature the performance of terrain traversability/classification studies is often reported using a variety of criteria and metrics that are rarely comparable.

Terrain classification methods tend to report standard metrics of classification performance, such as precision and recall, F1 scores, confusion matrices, or Intersection over Union (IoU) for image-based applications (Jiang et al., 2020). Although these metrics and tools are standard and well understood, any comparison between methods requires that the classes and their definition be exactly the same, which is rarely the case in practice. Even the common use of semantic labels such as “rock” or “vegetation” falls short of guaranteeing reliable and consistent definitions. The characteristics of the hardware used in the experimental validation (e.g., sensor resolution) and the experimental conditions can also introduce some discrepancy in the meaning of the results obtained.

Due to the variety of definitions of metrics (or cost functions) used for terrain traversability (Molino et al., 2007), researchers are often not able to directly compare the performance of different traversability estimation methods. In addition, although some metrics are directly observable (e.g., terrain slope) or indirectly observable (e.g., rover configuration on the terrain), others are simply unobservable. In the former case, the performance of corresponding methods can be partly evaluated by measuring the performance of the predictions made by the approach (e.g., comparing the *predicted* difficulty or cost with the *actual* configuration of the robot when it traversed a given terrain patch). However, in the latter case this is not possible; the evaluation may be limited to common sense observations (e.g., clear obstacles on the map are given a very high difficulty or cost) and to the subsequent behavior of the path planner when it uses the computed traversability map.

3. Vision-Based Approaches

Vision-based heuristic approaches for terrain classifications have been in use since the early 1970s. In this paper, we have divided the vision-based literature into different subsections, which include non-learning methods (using stereovision), traditional learning-based method, and deep learning-based methods. Terrain classification using traditional learning-based methods relies on hand-designed features classifying pixels of image patches. On the other hand, deep learning-based methods use a set of possible models to automatically identify underlying features that are used in the classification. Before addressing learning methods, we start this section discussing nonlearning stereo vision-based methods in Section 3.1. In Sections 3.2 and 3.3, learning-based methods for terrain classification are discussed.

3.1. Nonlearning Methods for Classification and Traversability using Stereovision

In an outdoor environment, terrain classification using vision-based approaches requires robust range estimation for the traversability analysis. Outdoor natural environment terrain perception could be achieved by constructing a dynamic representation of the scene in terms of occupancy grids or digital elevation maps. In this section, we review nonlearning vision-based methods and technologies that have been used to classify terrain in outdoor environments. The predominant approaches for measuring terrain traversability are based on geometric processing. Indeed, the Mars Exploration Rover (MER) vision and navigation system research contributed significantly in developing the nonlearning stereo vision approaches over the years (Goldberg et al., 2002). Stereovision provides the geometric aspects of the scene for perception in real time for robot navigation. Interested readers may refer to (Matthies et al., 2007) to review the stereo vision approaches used in the MER program for visual odometry and rover navigation. These methodologies are relevant to off-road robot navigation, which does not assume a flat surface in front of a vehicle and positive obstacles with respect to the ground plane. In addition, some of the more specific open challenges for off-road autonomy and navigation are discussed in Section 8.

A terrain classification algorithm was developed in (Manduchi et al., 2005) using stereo range measurements for safe off-road navigation in a highly vegetated environment. Terrain classes such as soil, rock, green (photosynthetic) vegetation, and dry (nonphotosynthetic) vegetation were considered in this work. Two approaches were presented to perceive the terrain; the first was based on the surface reflectivity captured using a stereo color camera, and the second was based on analyzing

the range data from LIDAR. A color calibration dataset was captured in an outdoor environment to perform classification using maximum likelihood. The variability in each class due to color was adequately predicted by the classifier to perceive the terrain classes of interest. Note that the color shift and the illumination change due to weather conditions are the two major challenges in using color-based classification approaches. These challenges could be overcome by having a large enough dataset to minimize aleatoric and epistemic uncertainties. Therefore, it is of the utmost importance to consider a probabilistic framework for terrain classification and traversability analysis for outdoor unstructured environments.

Disparity maps have often been used to find the traversable areas in the grid map. The u-disparity occupancy grid was used in (Kuthirummal et al., 2011) to map the stereo point cloud into the cells. Then, the elevation histogram was computed from the point cloud that maps to the individual cells. The elevation histogram of all the cells was used to remove the outliers in detecting obstacles based on different heights within the cell. The compatibility function was used to determine the adjacent traversable cells, which was then used to construct a graph. The connected components in the graph are used to find the traversable, obstacle, and unknown cells of the map. The methodology was tested on Black-I Landshark UGV, equipped with a LIDAR and two pairs of stereo cameras, in a front-back arrangement, mounted on a pan-tilt unit. The experimental results were shown in a static outdoor scenario, where the disparity maps were obtained at 15 Hz to find obstacles at 28 ms per frame. Note that the performance of this methodology is dependent on the accuracy of the disparity maps obtained in the outdoor environment.

In (Dubbelman et al., 2007), an hysteresis threshold was used to detect positive and negative obstacles during daytime and nighttime. A stereo camera was used together with infrared light to detect obstacles during the night. First, a disparity image was estimated using the multiresolution technique through the sum of absolute differences of multiple windows (Van Der Mark and Gavrilu, 2006). The quality of the estimated disparity was validated using the normalized difference between different matches and the signal-to-noise ratio of local regions in the intensity image. A fine-to-coarse selection scheme was used in the stereo image pyramid for robust disparity estimate in the night condition. Positive obstacles were estimated using the hysteresis threshold around the patches of obstacle pixels and using the morphological operation. Negative obstacles were detected using the ratio between expected and estimated depth of an image, and by applying the hysteresis threshold. Test images were captured in an off-road scenario using a stereo camera during daytime and nighttime. Positive obstacles were detected up to 25 m during the day and night. Negative obstacles were detected up to 10 m in the daytime. The method was unable to detect negative obstacles from a reasonable range at nighttime.

A terrain classification and traversability analysis was used for Legged Squad Support System (LS3) quadruped vehicle foot placement in natural outdoor environments (Bajracharya et al., 2013). A near-field terrain map, including classification of vegetation and negative obstacles, was constructed with a resolution of 5 cm for path planning and controlling the robot. Stereo visual odometry was used to accumulate voxel maps using dense stereo range data, which inherit the terrain classification labels and the geometry statistics. Vehicle movement to approach natural outdoor terrain was controlled using the resultant ground elevation map, which includes the statistics and classification, for obstacle avoidance and navigation. During nighttime operations, a custom NIR illumination system was used without modifying or adjusting the parameters of the proposed algorithm. Over 245 distinct experiments were conducted in different weather conditions and natural outdoor environments. Negative obstacles were detected during the nighttime in most of the terrain condition using a custom NIR illumination system.

In outdoor environments, it is not always possible to obtain the disparity maps with sharp boundaries and without any flattening effects. Predefined camera uncertainty models may not be sufficient to accurately detect obstacles when using geometric methods. In particular, negative obstacle detection during day and night can pose a challenge using heuristic methods and passive measurements only. In contrast, the nonlearning methods are predictable in terms of computation performances and results. Their dependence on weather conditions could affect the performance in

Table 1. Traditional learning-based terrain classification methods.

Reference	Architecture	Terrain Type	Data
(Angelova et al., 2007)	Color, texture + Decision tree/Nearest neighbor	Sand, soil, grass, gravel, asphalt, woodchip	RGB
(Bajracharya et al., 2008)	Color + SVM	Traversable, nontraversable	RGB, Stereo
(Moghadam and Wijesoma, 2009)	Color, texture + SVM/ARTMAP	Traversable, nontraversable	RGB, Stereo
(Zou et al., 2014)	Color, texture, SURF, SIFT + SVM/ELM/Nearest neighbor	Gravel, hard soil, pothole, grass	RGB
(Hang et al., 2017)	Codebook + SVM	Asphalt, dirt, grass, gravel, rock, sand	RGB
(Filitchkin and Byl, 2012)	SURF + SVM	Asphalt, grass, gravel, mud, soil, woodchips	RGB
(Lee and Kwak, 2011)	SURF + MLP	Sky, grass, tree, soil, gravel, outlier	RGB

some situations. However, the stereo vision-based methods were demonstrated and proven to work in many research programs including the DARPA Learning Applied to Ground Vehicles (LAGR) program, the DARPA Legged Squad Support System (LS3) program, and the Mars Exploration Rover (MER) program.

3.2. Traditional Learning-Based Methods for Terrain Classification

Terrain classification can help ground robots to perceive the surrounding off-road environments and perform traversability analysis based on scene appearance. Significant work based on image processing and machine-learning methods has been devoted to the problem of terrain classification. Traditional learning-based methods for terrain classification, shown in Table 1, extract visual features (e.g., color, texture) (Angelova et al., 2007; Zou et al., 2014) or distinctive features [e.g., Speeded-up Robust Features (SURF)] (Filitchkin and Byl, 2012; Lee and Kwak, 2011) for training a classifier. For example, a hierarchical classifier is proposed in (Angelova et al., 2007) to lower the computational cost for differentiating between different terrains (e.g., soil vs grass, sand vs gravel). The classification was performed in a top-down fashion, starting from simple classifiers and advancing into more complex classification tasks. Feature representations of varying complexity including average color, color histogram, and texture-based features were used at different levels of the hierarchy. A decision tree classifier was employed at each level, and a nearest neighbor was used only in the last stage. The hierarchical classifier was evaluated on small patches from six terrain classes (sand, soil, grass, gravel, asphalt, woodchip) in comparison with baseline (flat, nonhierarchical) classifiers. The results showed that the hierarchical classifiers achieved competitive accuracy and are generally faster than the baseline classifiers.

The long-range vision-based terrain classification method was proposed in (Bajracharya et al., 2008) using a stereo camera. A histogram-based naive-Bayes classifier was used for terrain classification using the stereo range data. The classification results were back-projected to the original image as labeled image windows. To train a Support Vector Machine (SVM), 2D normalized color histograms in labeled image windows were calculated for terrain classification. To verify the learning mechanism used, which is learning long-range terrain classification from short-range terrain classification, the data labeled by the local field classifier are split into a set of pixels at the near field and another set at the middle field. The long-range classifier was trained only on the near set and then evaluated on both the near and middle sets separately. The results showed that SVM performed well at range extension with even higher accuracy at middle-field.

A near-to-far, online learning method using stereo images was proposed in (Moghadam and Wijesoma, 2009). The authors used the near-field stereo information associated with the terrain geometry and appearance (points belong to a ground plane or not) to train a classifier to classify

the far-field terrain well beyond the stereo range for each incoming image. Their proposed method can train incrementally over time on the incoming data stream adapting to unknown environments without using any hand-labeled training data.

A comprehensive analysis of learning-based terrain classification methods by exploring various image descriptor types was provided in (Zou et al., 2014). The model was evaluated on classic composite descriptors (local ternary patterns, scalable color, edge histogram and color structure, and homogeneous texture), novel composite descriptors [color and edge directivity, fuzzy color and texture histogram, and joint composite descriptor (JCD)], bag of visual words (BOVW) sparse vector of occurrence [SURF, scale-invariant feature transforms (SIFT)], and local ternary patterns. Several classifiers were tried, namely extreme learning machines (ELM), SVM, and nearest-neighbor classifiers. Results showed that the approach based on JCD and ELM classifiers performed best in terms of classification effectiveness. JCD allows representing different terrain images with significant interclass discrepancies, while ELM has mild optimization constraints and obtains better performance over various types of terrain.

Consecutive steps in the classification pipeline and different fusion methods for visual terrain classification were described in (Hang et al., 2017), with the focus being on the BOVW framework. The paper presented a comparison of different BOVW frameworks and fusion methods for visual terrain classification. The BOVW framework is based on the idea of using overcomplete basis vectors to encode the local descriptors. These basis vectors are also known as codewords, and a collection of those codewords is referred to as a codebook. The codebook is computed on the training set and used for the descriptors of all images. The codewords are considered to be characteristically representative of the image descriptors. The authors designed an optimum pipeline and developed the hybrid representation to produce an effective and efficient visual terrain classification system, which was robust to diverse noises and illumination alterations.

Similar to (Hang et al., 2017), the work in (Filitchkin and Byl, 2012) used BOVW created from SURF for terrain classification. To improve SURF feature extraction, a gradient descent inspired algorithm was proposed to adjust the SURF Hessian threshold. A vocabulary was then generated from SURF descriptors by applying k-means clustering. There are six different terrain types: asphalt, grass, gravel, mud, oil, and woodchips. Once the vocabulary has been created, each type of terrain image can be described by a word frequency vector. The linear SVM was trained using these frequency vectors afterwards.

A hybrid approach combining SURF features and a deep neural network for classifying terrain types, such as sky, tree, grass, and soil, was presented in (Lee and Kwak, 2011). SURF feature vectors were extracted from small regions in a grayscale image which was converted from an original RGB image, and they were then used to train a multilayer perceptron (MLP) network (Mitchell et al., 1997). Their MLP network was set as one input layer, two hidden layers, and one output layer. Although the experiments for real off-road images showed that the method had a good performance, the resulting classification presented a blocky structure. This is mainly because feature extraction is based on small patches rather than a pixel.

One important aspect is that the appearance of natural terrain such as mud, rock, vegetation, water, and sand can exhibit marked interclass similarity and significant intraclass variation. Therefore, the main challenge for traditional learning-based approaches is to find complex features that can not only accommodate intraclass variation but can also distinguish different terrain classes correctly. Recent advances in computer vision, especially deep semantic segmentation, have shown great success in scene understanding. The main advantage of deep learning methods is that they are able to learn high-level features from data, whereas traditional learning-based approaches require domain expertise and hand-designed feature extraction. The disadvantage, of course, is the usually large amount of data required for training. A comparison between a convolutional neural network (CNN) and an SVM trained on SURF in the context of terrain classification for off-road driving was given in (Shen and Kelly, 2017). The CNN model was built with Keras (Chollet et al., 2015) using Theano backend (Team et al., 2016) and took as input normalized RGB pixel values. Both classifiers were trained on 100×100 pixel images of six terrain categories: pavement, dirt, foliage,

bark, grass, and dry vegetation. The classification was tested on mixed terrain images, and generally, the CNN was more robust to various sources of noise and generated smoother images. The SVM additionally struggled to identify grass, likely due to the color insensitivity of SURF features. This early finding has suggested the superior performance of a deep neural network over traditional classification methods in terrain classification accuracy (Shen and Kelly, 2017).

Summary. Traditional learning-based methods for terrain classification rely on hand-crafted features (color, variance, contrast, and many other statistics). Advantages are that these methods do not need large datasets for training the classifiers. Labeling is a major challenge in the case of terrain data. Because terrain types often have a fuzzy boundary in an image [e.g., where exactly is the boundary between grass and dirt in Figure 1(b)?], labeling data is time-consuming and imprecise, and hand-crafted features work around that issue. Disadvantages of the traditional learning methods include the following: a) The feature extraction methodology is usually designed for specific terrain classes and may not be easily generalized to different terrain types; and b) the terrain classes exhibit similarity and variation among different types. Therefore, it is challenging to extract complex features for a specific terrain type that can be robust to the variation and similarities in different terrain classes. In addition, the concept of “shape” and structure, frequently exploited in many recognition tasks in computer vision, does not apply to terrain (unlike a car or person, where elements can be spatially structured).

3.3. Terrain Classification Using Deep Learning-Based Methods

The recent success of deep learning networks has enabled remarkable progress in image and video semantic segmentation. Semantic segmentation methods using deep learning usually take image data as an input and learn hierarchies of features through training. Afterwards, the trained network is able to produce per-pixel labeled output. In the following paragraphs, we discuss deep-learning image-based and video-based methods for terrain classification. A selection of deep learning-based methodologies is given in Table 2.

3.3.1. Deep Learning-Based Image Semantic Segmentation

The fully convolutional network (FCN) introduced in (Long et al., 2015) was originally proposed for the semantic segmentation task. The insight of this approach is to take advantage of existing CNN classifiers that are able to learn hierarchies of features and transform them by replacing the fully connected layers with convolutional layers to produce coarse output maps. These maps are then upsampled to dense pixel labels by fractionally stride convolution (i.e., deconvolution). The skip connections were used to refine the segmentation by using higher resolution encoder feature maps. Most of the state-of-the-art semantic segmentation methods are based on FCNs. Subsequently, a FCN was used for semantic segmentation in terrain in (Maturana et al., 2018). The FCN is based on the VGG16 architecture (Simonyan and Zisserman, 2014), which is a large network with 13 convolutional layers and 3 fully connected layers. VGG16 was replaced by Darknet, a similar and more efficient architecture, in (Redmon, 2016). The Darknet version was found to be faster than the original one (Maturana et al., 2018).

An encoder-decoder architecture named SegNet was introduced in (Badrinarayanan et al., 2017). The encoder serves to produce hierarchical feature maps with a CNN backbone such as VGG without fully connected layers. The decoder, conversely, maps these low resolution image representations to pixelwise predictions by a set of upsampling and convolution layers. Specifically, the decoder uses the max-pooling indices in the corresponding encoder to perform nonlinear upsampling. SegNet provides a modular design by decoupling the segmentation architecture into encoder and decoder. In the context of terrain analysis, this generic design allows the extension with different encoding and decoding methods and aids practitioners to pick the most suited design choice for terrain classification.

In (Ronneberger et al., 2015) the authors proposed a u-shaped architecture network (U-Net) where feature maps from different encoding layers are concatenated with the upsampled feature

Table 2. Deep learning-based terrain classification methods.

Reference	Architecture	Application	Terrain type	Data
Semisupervised methods				
(Wellhausen et al., 2019) [‡]	ERFNet	Terrain	Asphalt, Grass, Dirt, Sand	RGB
(Hirose et al., 2018)**	GONet	Indoor, Real-time	NA	RGB
Fully-Supervised methods				
(Valada et al., 2019) ^{†‡}	Adapnet++	Terrain, Urban, General	Grass, Sky, Vegetation	RGB, Infrared, Depth
(Iwashita et al., 2019) [‡]	TU-net, TDeepLab	Terrain	Sand, Soil, Rocks	RGB, Infrared
(Rothrock et al., 2016) [‡]	DeepLab-v1	Terrain	Sand, Rocks	Grayscale
(Maturana et al., 2018) [‡]	FCN	Terrain	Sky, Road, Grass, Vegetation	RGB
(Kim et al., 2018) [‡]	ENet, 3D CNN	Terrain	Vegetation, Outdoor terrain	RGB, Point cloud
(Suryamurthy et al., 2019) [‡]	ENet, ERFNet, SegNet	Terrain	Sand, Gravel, Road	RGB
(Long et al., 2015)*	FCN	General	NA	RGB
(Badrinarayanan et al., 2017)*	SegNet	Urban, General	Tree, Vegetation	RGB
(Ronneberger et al., 2015)*	U-net	Medical	NA	RGB
(Chen et al., 2014)*	DeepLab-v1	General	NA	RGB
(Palazzo et al., 2020)*	DeepLab-v2 / ResNet	Terrain	Grass, Road	RGB
(Chen et al., 2018b)*	DeepLab-v2	Urban, General	Urban vegetation	RGB
(Chen et al., 2017b)*	DeepLab-v3	Urban, General	Urban vegetation	RGB
(Chen et al., 2018c)*	DeepLab-v3+	Urban, General	Urban vegetation	RGB
(Paszke et al., 2016)*	ENet	Urban, General, Real-time	Tree, Urban vegetation	RGB
(Romera et al., 2017)*	ERFNet	Urban, Real-time	Urban vegetation	RGB
(Tremel et al., 2016)**	SQ	Urban, Real-time	Urban vegetation	RGB
(Mehta et al., 2018)**	ESPNet	Urban, General, Real-time	Urban vegetation	RGB
(Zhao et al., 2018)**	ICNet	Urban, General, Real-time	Urban vegetation	RGB
(Chiodini et al., 2020) [†]	DeepLab-v3+	Terrain	Mars	RGB

[†]Stereo camera was used in this work.

[‡]These methods are used for terrain classification.

*Instead of original work, modified forms are used for terrain classification.

**These methods are useful for real-time terrain classification.

maps from the corresponding decoding layers. Motivated by the success of U-Net in medical image segmentation, a TU-net architecture that incorporated two U-Nets for terrain classification was developed in (Iwashita et al., 2019). The approach fused thermal and RGB images at different levels (early, middle, and late) to enable the network to be robust to illumination changes. The experiments showed that TU-net achieved higher accuracy than U-net with RGB images only.

Refinements on fully convolutional networks were introduced to improve the segmentation accuracy by incorporating context. (Yu and Koltun, 2015) introduced dilated or atrous convolutions, which expanded the receptive field without losing resolution based on the dilation factor. Thus, it provided a better solution for handling multiple scales. DeepLab-v1 was proposed in (Chen et al., 2014), which proposed using atrous convolution to explicitly control the resolution at which feature responses are computed and the fully connected conditional random fields (CRF) as a postprocessing. CRF increase the segmentation accuracy at the cost of additional computation. The authors then proposed the improved version that uses atrous spatial pyramid pooling (ASPP) for multiscale support (Chen et al., 2018b). The DeepLab method was refined further by augmenting the ASPP

module with image-level features to capture longer range information (Chen et al., 2017b). The latest version (Deeplab-v3+) added the decoder module to improve segmentation along the object boundaries (Chen et al., 2018c). As one of the state-of-the-art semantic segmentation architectures, Deeplab has become a popular network for terrain analysis. A multimodal (e.g., RGB, near-infrared, and depth) semantic segmentation framework that leveraged Resnet-50 (He et al., 2016b) and ASPP module in Deeplab-v3 was proposed in (Valada et al., 2019). The results showed that this multimodal framework achieved state-of-the-art performance on a number of public datasets, including one dataset collected in a forest environment, while demonstrating robustness in adverse perceptual conditions such as rain, snow, and night. Deeplab-v1 was employed for terrain classification on high resolution images taken by the powerful telescopic imager in Mars' orbit as well as for ground images from Curiosity (Rothrock et al., 2016). The frames were manually annotated in regions of “high confidence” in the images. Although the method is not reported to run “live” on Curiosity, the classifier output assisted in landing site traversability analysis for the Mars 2020 Rover mission, and slip prediction for the Mars Science Laboratory mission. The TDeeplab (Two Deeplab) architecture that was built upon Deeplab-v2 was proposed in (Iwashita et al., 2019). In this work, RGB and infrared images were fused at early or late stages to perform terrain classification. The results showed that TDeeplab performed better than Deeplab using RGB images only.

As with terrain analysis, many advanced real-world applications such as autonomous robot navigation and self-driving cars demand real-time processing of data on embedded devices. Accurate deep architectures such as Deeplab-v3+ or PSPNet (Zhao et al., 2017) require enormous resources and are computationally intensive. To address such challenge, alternative methods that focus on reducing the computation complexity of deep CNN architectures while retaining remarkable accuracy have gained more and more attention: e.g., SQ (Trembl et al., 2016), ENet (Paszke et al., 2016), ESPNet (Mehta et al., 2018), ERFNet (Romera et al., 2017), and ICNet (Zhao et al., 2018). Among these methods, ERFNet and ICNet achieved better tradeoffs between frame rate and accuracy. ICNet introduced a cascade feature fusion unit that fused semantic information in low resolution with details from high-resolution images. ERFNet proposed a non-bottleneck-1D layer that utilizes residual connections and factorized convolutions. In an early attempt to adopt efficient semantic segmentation networks for off-road navigation, the deep multimodal network presented in (Kim et al., 2018) consisted of 2D CNN and 3D CNN, which are fused by projecting 3D features to image space to perform terrain classification. In this work, 2D CNN was based on ENet, and 3D CNN was used for the point cloud. The results showed that the multimodal network was more robust to segment terrains under various seasonal conditions compared with unimodality networks.

Another approach using learning techniques was proposed in (Suryamurthy et al., 2019), where classification was performed pixelwise, labeling the terrain as stone, sand, road/sidewalk, wood, grass, metal, in addition to a general roughness estimation. Focusing on real-time applications, the authors compared ENet, ERFNet, and SegNet networks and found that ERFNet gave the best performance with faster inference time. One limitation is that all results are presented in small scale terrain. Although the target experiments are with humanoid robots, expansion to UGVs seems relatively straightforward.

All the learning-based methods discussed above belong in the realm of supervised learning, which requires a large amount of labeled data for training. In some classification tasks, it can be difficult to attain sufficient labeled data due to the high cost of the data-labeling process. Semisupervised learning, as a branch of machine learning, aims to deal with the situation in which a small amount of labeled data is available (Zhu, 2005). A semisupervised deep-learning method for terrain classification was presented in (Wellhausen et al., 2019). Five terrain classes— asphalt, sand, gravel, dirt, and grass—are sparsely labeled in images, and their labels cover between 0.1% and 10.9% of image pixels. The authors selected ERFNet with skip connection and trained the network using a semisupervised learning technique called Mean Teacher (Tarvainen and Valpola, 2017). The experiments showed that the weakly trained network was able to generate a dense prediction of terrain classes in RGB images. A semisupervised deep-learning approach named GONet, which leveraged Generative Adversarial Network (GAN) (Mirza and Osindero, 2014) for traversability

estimation from fisheye images, was proposed in (Hirose et al., 2018). GAN was trained in a semisupervised manner from traversable images of a fisheye camera and thus only learned to generate traversable images. GONet compared the input image with the GAN generated image that was similar to the input and as if it came from the set of traversable images. The difference between an input image and its GAN generated image was then used to determine whether the area seen in the input image was traversable or not. The results showed GONet outperformed supervised baselines using ResNet (He et al., 2016a) in terms of accuracy and computation efficiency. However, GONet has three limitations for terrain analysis: 1) it classified an image as traversable/nontraversable without providing significant image semantic information; 2) it was evaluated on fish-eye images of indoor environment only; 3) it could estimate traversability only at short range. Finally, (Sofman et al., 2006) showed that self-supervision could be achieved for traversal estimation using overhead imagery.

Furthermore, multimodal networks, where multiple sensing modalities can be fused to exploit their complementary properties, have recently been proposed for terrain classification. Although multimodal networks have demonstrated higher accuracy than the network using unimodal information under challenging perceptual conditions (Iwashita et al., 2019; Valada et al., 2019; Kim et al., 2018), they are large and may not be suitable for real-time applications. Future work is expected to develop efficient multimodal networks for off-road navigation.

Summary. State-of-the-art methods for terrain classification are built on deep semantic segmentation networks. Deep-learning-based methods have several advantages: a) They do not require domain expertise and can automatically learn high-level features from data; b) the network design is generic and can be retrained to classify new terrains; and c) the network can be extended to a multimodal network which can fuse multiple sensor data. However, these methods require a large amount of annotated data for training neural networks. While usually the main limitation for labeling data is resources in most cases, the case of terrain labeling comes with the challenge of uncertainty and fuzzy boundaries. As an example, determining what is a horse or a chair when labeling an image is a straightforward task. With terrain, however, it is sometimes hard to visually differentiate between sand, dirt, or mud in an image, as they can look very similar (and yet will have very different traversability characteristics). In addition, the spatial boundaries of each class are not always easy to define, leading to noise in the labeling process. Recent advances in simulated training data are promising for well-defined geometric structures (i.e., classes that have a “shape” such as cars, trees, etc.), but are still challenging for terrain elements that rely more on texture than shape. Results on simulated terrain data have illustrated that artificially generating the realistic texture (not only visually pleasing for a human observer) that can be effectively used for training is an ongoing research topic.

3.3.2. Deep Learning-Based Video Semantic Segmentation

The state of the art discussed in the previous section focuses on still-image semantic segmentation. In off-road navigation, mobile robots receive video sequences from a camera sensor and can then use them to perceive their surroundings. Naturally, it is possible to apply image segmentation algorithms on each video frame, however such an approach completely ignores temporal continuity and coherence cues that might help achieve higher segmentation accuracy and faster execution speed. A number of recent approaches that utilize temporal information have emerged for video semantic segmentation.

Optical flow has been exploited by deep video semantic segmentation pipelines to improve accuracy and temporal consistency. Features wrapped from the previous frame by optical flow were combined with those of the current frame to perform video segmentation in (Gadde et al., 2017). Optical flow learned in a flow network (Dosovitskiy et al., 2015) was used to propagate the features from the key frame to the current frame in (Zhu et al., 2017b). A spatiotemporal transformer gated recurrent layer introduced in (Nilsson and Sminchisescu, 2018) combined optical flow-based feature warping with a gated recurrent unit. An efficient video semantic segmentation pipeline that used the optical flow method to exploit temporal information and ICNet for the main semantic segmentation

architecture was proposed in (Paul et al., 2020). Alternatively, some methods have focused on reducing inference times. The Long Short-Term Memory (LSTM) network was used in (Mahasseni et al., 2017) to optimally select a subset of frames for pixel-level labeling by CNN and interpolated the segmentation results to unselected frames. In (Shelhamer et al., 2016) the authors observed that high-level representations within a network evolved slowly in a video, and thus proposed clockwork networks that scheduled the computation of feature maps for key frames and shared feature maps in between. Subsequent studies in (Li et al., 2018) and (Xu et al., 2018) further optimized scheduling and propagation on video via adaptive feature propagation and adaptive selection of key frames as schemes. To our knowledge, most video segmentation methods have been evaluated on datasets, such as Cityscapes (Cordts et al., 2016) or CamVid (Brostow et al., 2009), which were set up for urban scene understanding and autonomous driving. Although some terrain types (e.g., trees, grass, or vegetation) are labeled in these datasets, there is no direct implementation of video semantic segmentation methods for terrain analysis in the literature. Nevertheless, these specialized methods for videos are worth exploring for future work.

A key bottleneck in the implementation of video semantic segmentation networks is that they require the pixel level labeling in many frames of the training videos. Acquiring pixel level annotations for image semantic segmentation is already costly. To ensure dataset diversity, many videos, each of which consists of hundreds of frames even for a very short movie, require the annotation of a very large number of frames.

3.4. Considerations

The traditional vision-based approaches developed in the early 1990's have evolved over the years and recently have been incorporated into deep learning methodologies. Traditional learning-based and stereo-vision methodologies for terrain classification rely on geometry and do not need a large dataset for training the classifiers. Researchers worldwide contributed towards the DARPA Learning Applied to Ground Vehicles (LAGR) program that led to the evolution of vision-based terrain classification methodologies during the last 20 years. In particular, the Jet Propulsion Laboratory (JPL) championed the near-field real-time stereo geometry approach and self-supervised long distance RGB feature learning. Only a few studies investigated video semantic segmentation methods for terrain classification using temporal information between frames. A major advantage of vision-based solutions for classification is that they provide appearance information. While LIDAR is suitable for determining the geometry and roughness of the terrain independently of illumination conditions, it cannot differentiate between two different types of flat terrain with the same geometrical features (e.g., sand and mud). This is an important advantage of vision. While stereo does combine the strengths of appearance and geometry, it must overcome the well-known calibration and illumination challenges. Recent deep learning methods have led to very high semantic segmentation performance for classes that are present in large public datasets, however those have focused mostly on urban environments, and the demonstration of terrain classification in complex, off-road environments is still limited in the literature. In addition, a key challenge in modern visual terrain classification is the data-labeling process. This is because terrain does not contain well-defined geometrical shape. In addition, terrain classes can be easily mislabeled by a human expert (e.g., sand can be confused with dirt, or dry soil with wet soil, i.e., mud). This characteristic makes the use of weakly supervised learning methods a promising avenue, where noisy data are taken into account.

4. LIDAR-Based Approaches

A LIDAR sensor primarily provides a geometric representation of scenes by emitting multilight and detecting their returns. The 3D points of [x,y,z float] data type that are typically output by the sensor encode the metric distance along each beam of light with respect to the origin of the sensor. The reflectivity at that point can also be used to provide additional information. Returns not only encapsulate the depth of the scenes but also offer additional physical knowledge in relation to the elements of the scene reached by the laser beams, which can be utilized for terrain analysis. In

this section, we will discuss LIDAR sensors and their uses with traditional model-based and deep learning-based approaches for terrain analysis.

One of the most powerful aspects of a LIDAR sensor is its accuracy and consistency in measurements compared to other sensors. As exemplified in (Brubaker et al., 2013), simple processing such as a region growing with an outline filtering technique (e.g., random sample consensus) can produce useful results. This shows how simple geometric structures can give strong features for segmentation.

Note that the use of LIDAR for terrain segmentation or classification has been extensively established in the remote sensing community. In this paper, we focus on terrain analysis with higher density point intensity (e.g., more than 200 pts/m²) and finer analysis scale (e.g., immediate surroundings of a robot rather than city scale).

In addition to the well-established image-based techniques discussed in the previous section, there has been a recent significant increase in LIDAR scene segmentation methods. These are sometimes used in conjunction with image-based techniques to generate features that incorporate both RGB and geometry data [e.g., (Qi et al., 2017; Choy et al., 2019) among many others, and we will cover these in more detail in Section 7]. These provide advantages such as 3D object finding (for example in the case of all-terrain autonomous vehicle driving, trees, lakes, or boulders) and high resolution scene understanding (Tchapmi et al., 2017). While there are many model-based approaches that can give strong scene information from LIDAR point clouds, the majority of recent work has been focused on applying deep learning to the point cloud data format. Much progress on LIDAR point clouds using deep learning has been made specifically targeting the autonomous driving (Li et al., 2020), but a direct translation of those methods to terrain analysis is not always possible. The rest of this section will discuss in more detail the aforementioned LIDAR-based terrain analysis approaches: 1) traditional learning-based and 2) deep-learning-based approaches.

4.1. Traditional Learning-based Methods for Terrain Classification

Prior to the rise of deep-learning or Convolutional Neural Networks (CNNs), conventional model-based approaches were often exploited to process LIDAR data. The work in (Thomas, 2015) exemplifies the use of Maximum-Likelihood (ML) for processing multispectral LIDAR data. Three multiwavelength data fed into a commercial classification framework which classifies the data into 4 classes using Maximum Likelihood. Although the LIDAR sensor provided high-quality measurements which led to accurate classification results, one of the major concerns of this study is that the cost and scale of this multispectral LIDAR setup may be unsuitable for some uncrewed ground robots (e.g., 1 m³ and 100 kg). Tree-trunk detection using SVMs and nodding 2D planar LIDAR is demonstrated in (McDaniel et al., 2012). Even though this system requires a stationary ground station for data gathering, the reported results were impressive and tested in various environments where the trees were obstacles in the terrain.

In remote sensing, LIDAR is one of the most important sensors to derive Digital Terrain Model (DTM) or Digital Elevation Model (DEM) (Mallet and David, 2016), which are widely populated for many remote sensing applications such as characterizing surface roughness and micro-topography (Brubaker et al., 2013). Airborne LIDAR data are usually used for these tasks and processed with noise removal and feature extraction (Mallet et al., 2016) for terrain characterization. As mentioned earlier, this research area is very broad and beyond the scope of our paper, so that we only highlight some fundamental studies (Mallet and David, 2016; Brubaker et al., 2013; Mallet et al., 2016).

Bayesian generalized kernel inference is used to determine a traversability map from LIDAR data in (Shan et al., 2018). Incoming LIDAR data are incorporated as training data, and terrain elevation is estimated. Therefore, they use a regression model on the LIDAR scans for all cells within a distance threshold. Traversability is directly computed for the cells intersected by LIDAR points. This is used as the training data for traversability inference, applied to all grid cells that are within the same distance threshold used in the previous inference step. Even though the research focuses on simulated data, they show promising results for the off-road with real-time processing.

Even though the strength of deep-learning approaches is dominant in terrain classification, traditional learning-based approaches are still popular and demonstrate promising results. Recently, a purely model-based terrain classification based on the power spectral density (PSD) of the surface 3D profile obtained from a point cloud was presented in (Reina et al., 2020). It provides insight not only on the magnitude of irregularities but also on how these irregularities are distributed at various wavelengths. Waviness (indicates how quickly the energy of the surface decreases as the wave-number increases) and the overall energy of the surface at the reference wave-number are chosen as characteristic properties of a certain elevation profile (collectively referred to as roughness parameters). Natural terrains show a larger energy decrease for increasing wave number than paved surfaces.

As discussed in Section 2.4, roughness can be calculated by the distribution of the points in the area of interest. In (Hamner et al., 2008), for example, the data are collected with a 2D LIDAR pointing down at the terrain, in a push-broom fashion. After the data are buffered, least-squares plane fitting can compute the residuals, where more residuals represent harder to traverse terrain. A similar approach can be extended to 3D sensors [see (Krüsi et al., 2017)].

LIDAR is an effective sensor to detect negative obstacles such as ditches, ruts, or depressions that are often occluded by the surrounding terrain. One approach is to voxelize the LIDAR data, and classify each observed voxel into vegetation and solid surfaces using the statistics of each cell. Voxels are a representation of LIDAR data obtained by dividing the scene volume into a collection of 3D regular cubes (called voxels), and the LIDAR points are allocated to 3D voxels, with voxel values being assigned according to the statistics of the LIDAR point(s) within the corresponding voxels. After voxel generation, ray tracing can then be used to determine areas of occlusion between voxels, enabling classification of these areas as negative obstacles using the context and class of the surrounding voxels (Heckman et al., 2007). Similar methods exist using heuristics and SVM to identify negative obstacles along a ray of LIDAR returns (Larson and Trivedi, 2011). Detection of negative obstacles can also be computed with 2D height maps propagating information from observed map cells to infer the unobserved terrain in the 2D map (Morton and Olson, 2011).

According to the survey conducted in this section, the following strategies can be incorporated in learning-based methods:

1. *DEM, DSM*: Precise digital elevation model and digital surface model play a pivotal role in terrain classification. This is mainly due to the fact that local/global feature extraction is performed using these data.
2. *Pre- or postprocessing*: Although LIDAR provides relatively consistent measurements, it is also important to get rid of obvious outliers and trends (e.g., biased mean). Voxelization is one of the popular methods in point cloud postprocessing, but there is a tradeoff in resolution and processing time.
3. *Feature selection*: Most of the traditional approaches made use of hand-crafted features, and the performance mostly relies on how much features are distinguishable (e.g., features extracted from vegetation vs. rock or mud).
4. *Learning algorithms*: SVM and Random Forest are often used for multi-/binary terrain classification tasks. Given discriminative features, stochastic framework options are also considered such as Bayesian or Gaussian Mixture Models.

4.2. Deep-learning-based Methods for Terrain Classification

There has been a recent increase in research into using CNNs to semantically segment LIDAR scenes. These scene segmentation techniques incorporate the use of LIDAR point-clouds to infer geometry or a combination of geometry *and* color based features. A semantically segmented LIDAR scene could be used for obstacle detection, for segmentation of point regions (for example as rough or manmade terrain vs. smooth or natural terrain), and could supplement conclusions drawn from RGB data. In the remainder of this section, we focus on purely LIDAR-based methods, discussing their role

in analyzing terrain. Later, in Section 7, we will discuss how some of the LIDAR approaches are combined with vision-based approaches.

When semantically segmenting point clouds, there are performance benefits by considering the 3D scene as a voxel-like structure. This can, however, cause finer resolution in the scene to be lost. To try to get the best of both methods, SegCloud (Tchapmi et al., 2017) uses a voxelized grid structure to perform 3D convolution. It avoids the coarse granularity present in most voxelization methods by applying trilinear interpolation (TI). The core contributions of this paper are the conjoining of the power of 3D convolution on regular voxel-like surfaces with the fine-grain resolution afforded by Fully Connected Conditional Random Fields at the point level with TI. Although it has not been shown in off-road terrain analysis directly, it is shown to successfully segment urban scenes into classes such as high-vegetation, low-vegetation, man-made terrain, natural terrain, and buildings on the Semantic3D dataset (Hackel et al., 2017). One limitation of this paper is that there are no real-time inference time results reported.

Processing points directly from the point cloud is an interesting problem that was first proposed by PointNet (Qi et al., 2016) and improved by its successor PointNet++ (Qi et al., 2017). Both of these papers were pivotal works in applying CNN techniques directly to point cloud data. They have been used in both object-centric and scene-centric contexts to great effect. PointNet applies a small transformation network [a 3D extension of the type used by (Jaderberg et al., 2015) for 2D image alignment] to predict an affine transform matrix that aligns the point cloud to a canonical space. It relies on the inherent symmetry of a max-pooling layer to backpropagate the signal to train the network. Pointnet++ extends on this by considering small “neighborhoods” of points, thus allowing features to be learned at different scales (Qi et al., 2017). More recently, networks have built on the ideas conceived through PointNet, and have achieved great success in semantic scene segmentation benchmarks such as Semantic3D segmentation-8 (Hackel et al., 2017) and the ScanNet benchmark (Dai et al., 2017). Both of these datasets involve measuring the success of a model in semantically segmenting a large scene of points. Generating a similar cloud from a LIDAR attached to a vehicle and semantically segmenting the scene into classes would allow for terrain regions to be classified and integrated with information about the robot in question to determine traversability.

One way of identifying successful networks in the task of semantically segmenting scenes is to check performance on dataset benchmarks. The work in (Choy et al., 2019) proposes the use of high dimensional convolutions to directly process 3D videos. They create a point-cloud data ML engine dubbed the “Minkowski Engine.” The paper implements a U-Net structure (Ronneberger et al., 2015) to preserve scene geometry at different abstraction levels, and it achieved excellent performance on the ScanNet indoor scene segmentation benchmark. There is also a focus on 3D videos, which is potentially relevant to the on-vehicle terrain analysis task. The network itself implements sparse convolutions using their proposed engine and sparse tensors for efficiency and speed.

LIDAR data may or may not hold RGB features depending on the sensor setup available for fusion. Inherently, a portion of the information generated by LIDAR data is purely geometric. The system in (Boulch, 2020) operates on both spatial and color features. A geometrical weighting function is applied to the input points, with each allocated a weight based on the points similarity to the points in a kernel patch. This enables training of the network on point cloud data without using RGB values. They found experimentally that training without color generated different geometry-based features to those in the colorised case. Indeed, both tests generated strong results. This research paper has excelled at the Semantic3d Semantic8 benchmark (Hackel et al., 2017) (an outdoor large-scale point-cloud segmentation challenge into the categories 1: man-made terrain, 2: natural terrain, 3: high vegetation, 4: low vegetation, 5: buildings, 6: hard scape, 7: scanning artefacts, 8: cars).

It should be noted that not all of the above models have been used in real time, and thus some work and consideration would be needed as to inference time and resource usage in the context of terrain analysis with moving UGVs. The problem of achieving multi-modal terrain segmentation (RGB and LIDAR) is one that would be highly valuable if achieved as it would allow leveraging of the strengths of both sensor types. Research conducted in the area of driverless cars and their

Table 3. LIDAR-based approaches.

Reference	Architecture	Application	Terrain type	Data	Year
(Thomas, 2015)	Model-based (Maximum Likelihood)	Terrain classification	Vegetation, building, road etc.	3 multispectral LIDAR	2015
(McDaniel et al., 2012)	Model-based (SVM with height filter)	Traversability analysis	Off-road, forest	Nodding LIDAR with IMU	2012
(Krüsi et al., 2017)	Model-based (Optimisation)	Traversability analysis	Urban, off-road	LIDAR and IMU	2017
(Shan et al., 2018)	Model-based (Bayesian Inference)	Traversability analysis	Urban, off-road	LIDAR	2018
(Reina et al., 2020)	Model-based (Power spectral density)	Terrain classification	Urban, off-road	LIDAR	2020
(Langer et al., 1994), (Gennery, 1999), (Hamner et al., 2008)	Model-based (Regression)	Terrain roughness analysis	Rough terrain (Martial)	LIDAR	1994-2008
(Larson and Trivedi, 2011)	Model-based (SVM)	Traversability analysis	Negative obstacles (ditches, rut)	LIDAR	2011
(Tchapmi et al., 2017)	SegCloud	Semantic Segmentation (Autonomous driving)	Urban	RGB, Point cloud	2017
(Lang et al., 2019)	Point Pillars	3D Object detection (62 Hz)	Urban, off-road	RGB, Point cloud	2019
(Qi et al., 2016)	PointNet	3D Object classification and segmentation	N/A	RGB, Point cloud	2016
(Qi et al., 2017)	PointNet++	3D Object classification and scene segmentation	Urban	RGB, Point cloud	2017
(Martínez et al., 2020)	Model-based (Multiple Classifiers)	Traversability cost	Urban	LIDAR	2020
(Goodin et al., 2021)	Model-based (Regression)	Traversability cost	Off-road	LIDAR	2021

†indicates airborne LIDAR

‡indicates mobile LIDAR

datasets, which while perhaps not explicitly terrain or scene segmentation, may provide insights to real-time functioning LIDAR based networks. Of note is the system in (Milioto et al., 2019) which has performed well in the semantic scene segmentation problem based on a labeled KITTI dataset (Geiger et al., 2012) and is able to run on an Nvidia Jetson Xavier at between 5 and 13 frames per second. This paper performs a spherical projection of the point cloud to create a 2D image. They are then able to use a CNN backbone and GPU-based computing to perform rapid calculations that allow for high inference speeds.

The summary of the investigated LIDAR-based approaches is presented in Table 3 based on their application, terrain type, architecture (method for model-based approaches), and input data. Model-based approaches have been widely utilized for traversability analysis, whereas deep learning-based show strength in scene segmentation and 3D object detection using visual and depth information. This is mainly due to increasing popularity in autonomous driving and many open urban datasets available.

4.3. Traversability Analysis

Important geometric ground information for terrain traversability can be derived from point clouds obtained from a LIDAR sensor. This was demonstrated in (Krüsi et al., 2017) by fitting a robot's

footprint to the map and analyzing the local density of map points to estimate terrain assessment. Some level of field navigation (e.g., rough outdoor terrain and dynamic urban environments) performance is presented. It addresses challenging problems efficiently, and it is a promising strategy for traversability analysis. The concept can be extended to include meshes and consider a flood-fill approach to the meshes to identify traversable areas (Ruetz et al., 2019).

Without deep-learning, LIDAR is an efficient sensor to be used for regression instead of a classifier. From the regression point of the view, ease of traversability can be estimated. A common strategy is to extract basic terrain statistics from patches around the robot, most commonly in front, on its upcoming path (usually vehicle-sized patches). For the patches, metrics like the variance can be calculated (Langer et al., 1994; Gennerly, 1999), in addition to the inclination (Hamner et al., 2008).

A different paradigm is to apply semisupervised learning to 3D data to differentiate between traversable and nontraversable terrain (Suger et al., 2015). Initially, the robot learns its traversability capabilities based on human operation across a given environment. From this partially and only positive labeled training data, the proposed approach infers a model for the traversability analysis of that particular platform.

An earlier model-based approach for point cloud-based terrain classification is presented in (Vandapel et al., 2004). It uses local 3D point statistics to compute saliency features that capture the spatial distribution of points in a local neighborhood. The authors create a parametric model of the saliencies distribution by fitting a GMM using the Expectation-Maximization (EM) algorithm. The Bayesian method is used to classify data into three classes: 1) clutter to capture grass and tree canopy, 2) linear to capture thin objects like wires or tree branches, and 3) surface to capture solid objects like ground terrain surface, rocks, or tree trunks. Their method could produce a visually accurate classification, although some data were misclassified as linear at the surface edges and scan lines of the laser. Reducing the number of classes to 2 (surface and canopy) produced better results and real-time performance even on the computer platform available at the time of publication (2004). Despite these facts, generally speaking, these deterministic model-based approaches often struggle with data samples that are not taken into account over the training phase (e.g., fitting GMM using EM). This then increases the likelihood of poor system performance in unseen environments.

4.4. Considerations

The use of LIDAR data for scene semantic segmentation and terrain analysis is one of the most popular and powerful strategies. This is mainly due to the fact that it is capable of capturing metric information, which is very useful especially for robot tasks (e.g., autonomous navigation) and providing high-quality measurement consistency and high-fidelity information (e.g., a variety of returning waveforms depending on terrain property) which also can be used for terrain analysis.

With regard to processing LIDAR data (e.g., point clouds), 3D convolutional neural networks are gaining popularity among the computer vision, machine learning, and robotics communities in either supervised, semi- or weakly supervised, self-supervised manners. Supervised approaches may be the most popular due to their simplicity and already established work in 2D convolution. However, one of bottlenecks is analogous to those of deep 2D CNNs (i.e., difficulty in obtaining high-quality annotation data).

LIDAR-based approaches can be powerful to distinguish some categories of terrain, especially obstacles with distinct geometry such as tree trunks or vegetation, however, terrain sections with no clear geometric features remain difficult to distinguish without using appearance information. For example, a section of flat terrain can be critically different in terms of terrain traversability depending on whether it is composed of sand, dry soil, or mud, however the state of the art has not demonstrated the ability to reliably estimate this difference.

5. Alternative Exteroceptive Sensing

Alternative exteroceptive sensing methods are particularly useful to classify types of terrain such as mud, water, or vegetation. Most terrain categories can be disambiguated using vision and

LIDAR-based approaches for terrain traversability analysis. However, mud and vegetation pose different challenges due to their nature and dependency on weather conditions. In addition to LIDARs, active sensors such as radars are used in self-driving vehicles for improved detection, tracking, and classification in low visibility conditions. Alternative passive sensors such as infrared (IR) sensors have been used to classify mud and vegetation in low light conditions. In the following paragraphs, we discuss alternative passive and active sensing applications to mud and vegetation detection using IR, hyperspectral cameras, and radars, respectively.

5.1. IR and Nonvisible Spectrum

Alternative exteroceptive sensors are used in the literature for the classification of mud, vegetation, and soil. For example, in the remote sensing literature, multispectral sensing is used to identify these elements. The bare soil observed by the multispectral data exhibits a linear relationship between the near-IR (NIR) and the red band, known as the soil line. This soil line could be used to classify the wet and dry soil using the bare soil pixels placements. The NIR and red band of the scene are used to find the Normalized Difference Vegetation Index (NDVI) (Wang et al., 2014)(Rankin and Matthies, 2010). The soil line is obtained by using the least-squares fit of the NIR and red reflectance data captured in two different scenarios using the multispectral camera. In the first experiment, the area contained vegetation and bare soil, in and out of shadows. In the second experiment, the area contained wet soil along with the vegetation. The difference between the NIR and red reflectance was used as a measure to classify between vegetation, dry soil, and mud in both scenarios. Given the soil line slope and y-intercept, a normal distance to the soil line image and a distance along the soil line image were generated. The former image together with the NDVI is used to classify the vegetation from the soil. The later image along with NDVI is used to segment mud from the dry soil. It was observed that the multispectral data could be used to classify vegetation and soil in shadows and dry soil out of shadows. However, it was not possible to detect whether the soil in the shadow is dry or wet using this method.

Intuitively, wet soil absorbs more light than the dry soil. Therefore, shortwave infrared could be used to measure the soil moisture contents. It was found that the intensity difference of shortwave infrared for different types of soils is difficult to detect and, therefore, the sensor is not very useful to identify the mud (Rankin and Matthies, 2010). However, shortwave infrared could be used to identify water-saturated fields such as mud puddles. The wettest portions of the field will appear darker in the image and therefore they could be classified easily. Furthermore, additional clues such as water reflections from the sky could help to identify the mud patches in the terrain. Using the shortwave infrared image, the mud could be classified based on the darker regions in the image. However, other classes such as snow, ice, water, and vegetation also appear similar in contrast, and therefore it is not possible to distinguish mud from these classes only using the shortwave infrared.

Midwave infrared and long-wave infrared sensors are within the thermal infrared spectrum. The main reason behind using this spectrum is to find the temperature difference between cooled surfaces (e.g., mud, water) and the uncooled surfaces (e.g., ground, dry soil) for passive terrain perception (Owens and Matthies, 1999). The authors in (Rankin and Matthies, 2010) also find that thermal imagery is useful to detect mud in nominal weather conditions. Furthermore, thermal infrared sensors are useful to detect mud from the thin occluded vegetation such as pine needles and leaves. Still, the temperature difference between different classes may not always be sufficient to resolve the segmentation of other classes from mud, in particular in nighttime operations. Mud and water classification, using the stereo pair of thermal infrared cameras, is demonstrated in (Rankin et al., 2011) along with pedestrian and vehicle detection, tree trunk detection, and negative obstacle detection. Long-wave infrared stereo cameras are used in off-road natural environments during day and nighttime to classify the terrain for the military applications. The temperature was recorded for dry soil, mud, and air during an overcast day. The lowest temperature difference of 1.5 °C was recorded between the classes, which provides thermal contrast in the long-wave infrared imagery to distinguish between the dry soil and the mud in the field. Due to the thermal contrast, the

dark regions appeared in the thermal imagery as candidates of mud. However, shadows, snow, ice, vegetation, and water can also appear in dark regions within the long-wave thermal infrared imagery, and therefore additional information from different sensors is required to better perceive the terrain.

Beyond-visible-spectrum sensors have also been combined with other exteroceptive (Milella et al., 2017) and proprioceptive sensors (Milella et al., 2019), where ground mapping and characterization are proposed. Multimodal ground maps were generated with a robot moving on various ground types (stone-paved, ploughed ground, and grass), with results showing good classification accuracy, with particular application in farming.

Thermal images have also been used in combination with RGB in order to avoid expensive annotation of nighttime images by leveraging an existing daytime RGB-dataset and using a teacher-student training method to transfer the dataset's knowledge to the nighttime domain (Vertens et al., 2020). The authors use a domain adaptation method to align the learned feature spaces across the domains through a novel two-stage training scheme. This work also introduces a dataset consisting of over 20 000 time-synchronized and aligned RGB-thermal image pairs. Although the method is not focused exclusively on ground analysis, it finds direct applicability to terrain datasets and has shown interesting results in autonomous driving.

5.2. Hyperspectral

Hyperspectral cameras measure reflected light in hundreds of narrow bands at each pixel across the electromagnetic spectrum to produce a hypercube. In contrast, multispectral cameras typically cover from 2 to 10 carefully selected bands along electromagnetic spectrum for specific applications. The high spectral resolution of hyperspectral cameras increases the possibility of accurately classifying a broad range of materials and terrain surfaces. The recent advances in optics, sensor technologies, and mobile computing have transformed hyperspectral imaging from a satellite-based sensing method to a mobile, in-situ, and small payload compatible for ground vehicles and UAVs for scene analysis. Hyperspectral data can be collected by either point spectrometers, line imagers, or full frame imagers. Due to the sensor characteristics, they can generally be grouped as VIS-NIR (visible through near infrared, or roughly 400–1000 nm), the near-infrared range (NIR 1000–1700 nm), short-wave infrared (SWIR 1000–2500 nm), and medium-wave infrared (MWIR 2500–5000 nm).

A full-frame hyperspectral camera mounted on a ground vehicle for terrain classification in rough terrain and dynamic scenes is used in (Winkens et al., 2017; Winkens and Paulus, 2018). A fully supervised method (Random Forest) is trained on the normalized spectral reflectance (VIS-NIR) to get an initial per-pixel classification. Then, a fully connected conditional random field is utilized to enhance and smooth the segmentation results using neighborhood information. The authors demonstrate that vegetation class with high chlorophyll content gives the highest accuracy while humans and painted materials have the lowest reported accuracy. They also show that the classifier is able to separate the road from rough ground and obstacles (Winkens et al., 2017).

To overcome the lack of labeled hyperspectral datasets, the work in (Ma et al., 2018) formulated the problem as an anomaly detection, with background pixels versus foreground (target pixels). An AutoEncoder (AE) is used to learn high-level features of the hyperspectral data (126 bands from 400–2500 nm) in an unsupervised way. Then, subpixel segments are determined according to local adaptive weights to their neighboring pixels. It is assumed that anomalies have a lower occurrence probability than the background pixels, therefore the reconstruction errors are directly used as an anomaly score to segment backgrounds from anomaly targets.

In (Liyanage et al., 2020) the authors used a hyperspectral camera in VNIR range for weakly supervising RGB images for off-road UGVs in unstructured terrains. A manual dataset is collected using a VNIR hyperspectral camera covering different terrain classes such as muddy, grass, gravel, and various natural object types. First, a Min-Max pooling method is used for band selection to reduce the dimension of a hyperspectral data cube from 204 spectral bands to 25 spectral bands (a cube is a representation composed of stacked images of the same scene seen at adjacent wavelengths so that for any image point a complete spectral reflectance curve is provided). Then, a shallow

neural network classifier is trained and tested. For RGB images, the false-colored RGB images generated from the HSI data cube are used to provide one-to-one correspondence between RGB and hyperspectral images. The results suggest that classification based on hyperspectral images gives overall good pixelwise accuracy, but weak labeling of a limited number of false-colored RGB images did not boost the classification performance.

While the application of a hyperspectral camera for land cover classification on aerial and remote platforms has been widely investigated, the impact of hyperspectral data for in-situ, ground-based robots has yet to be determined. Most notably, the majority of the current methods rely heavily on dimensionality reduction and band selection approaches to deal with the high dimensionality and the nonlinear properties of hyperspectral data. However, this additional spectral information, often statistically dependent in a local spectral window, could be utilized as a data regularization term in deep-learning-based solutions.

5.3. RADARs

RADARs have been used in some robotics applications to take advantage of their larger wavelengths (lower frequency) compared to LIDARs. This section focuses on two types of radar: Ultrawide band (UWB) radars and mm-wave scanning radars.

5.3.1. UWB RADAR

The ultra-wideband (UWB) radar operates at low frequency, which results in a large beamwidth. Therefore, a UWB radar can penetrate through some elements like vegetation. In (Ahtiainen et al., 2013; Ahtiainen et al., 2015), the authors have used UWB radar to reduce the number of obstacles seen by the LIDAR measurements. First, a grid-based traversability map was built using the LIDAR data. Then, an adaptive threshold for the obstacles was computed using the UWB radar. The fused traversability map was obtained by using the radar measurements cells in the map, initialized with the higher probability, in the untraversable areas due to the LIDAR measurements. The minimum and maximum range for the obstacle detection using the radar was set to 3.5 and 10 m, respectively. The minimum and maximum thresholds were also set to avoid clutter measurements. The map cells were considered occupied if the measured intensity exceeded the detection threshold. The map cells were considered free (no obstacles) when the measured intensity was below the detection threshold. The free cells were also classified into two different regions: a) the region before the first detection; b) region after the first detection. The conditional probability of two different regions was computed to formulate the probabilistic sensor model. In (Ahtiainen et al., 2015), the UWB radar sensor model was learned using a SVM to create a probabilistic occupancy grid. The features were extracted using the radar data, where the traversable and untraversable cells on the grid map were manually labeled. The extracted features from the radar data were a) the angle from the radar to the labeled cell, b) range bin index, and c) measured intensity. The learning was done using the C-support vector classification and using the radial basis function kernel. The resultant sensor model was used to create the probabilistic occupancy map, which results in reducing the obstacles clutter due to the LIDAR measurements. The methodology was tested in a flat lawn and rural environment using LIDAR and UWB radar on two different vehicles. A true negative rate (pass rate to distinguish between vegetation and obstacle) and a false positive rate (miss rate) were used to evaluate the performance of the algorithm in the different field trials and experiments. It was observed that the true negative rate was 92% of 10 cm depth of foliage and drops to 83% with 40 cm of foliage depth. The false negative rate increases from 9% to 15% for the same experiment. It is evident that, while LIDAR can provide close- and long-range measurements, sensor fusion using UWB radar can improve the overall obstacle detection in outdoor environments.

5.3.2. mm-wave Scanning RADAR

High-quality mm-wave RADARs used for imaging usually operate at a higher EM frequency than UWB and offer better resolution and lower noise. Although limited research has made use of mm-wave RADARs for terrain analysis, these sensors have an advantage of extra reliability in adverse

environmental conditions due to a better penetration of its signal through airborne dust, smoke, and fog, for example, even though this penetration is not sufficient to see through as much vegetation as their UWB counterparts. Once processed appropriately, the returns from a mm-wave imaging RADAR can be used in a similar way to LIDAR returns, albeit with higher noise and usually lower resolution and accuracy. In (Reina et al., 2011) a frequency-modulated continuous wave (FMCW) mm-wave RADAR was used to classify the terrain as obstacle or traversable on an off-road vehicle using a traditional supervised learning method. Then the same authors used a camera to help train a classifier run on the RADAR data (Reina et al., 2012).

5.4. Considerations

Multispectral sensors have limited ability to disambiguate mud from other terrain classes in shadows robustly. Reports indicate that thermal imagery can be a reasonable passive sensing option at nighttime for mud detection. In the literature, a combination of passive sensors such as thermal, stereo, and polarization cameras was recommended to detect mud during day and nighttime. UWB radar, used in combination with LIDAR and/or stereo camera, has the potential to detect actual obstacles through foliage and vegetation during day and nighttime, in close proximity to the robot, thereby allowing a robot to drive through some level of vegetation with some more confidence. However, this requires extensive scanning or multiple UWB units on the robot. mm-wave RADARs can see through environmental obscurants such as airborne dust and smoke very effectively but with lower resolution and accuracy than LIDAR. They are a good complementary sensor in challenging environmental conditions.

6. Proprioceptive Sensing

The last two decades showed growing interest in using self-sensing (proprioceptive) methods for characterizing traversability of the terrain. There are several reasons for this attention. First, the information obtained from the exteroceptive sensors may not correctly identify the traversability properties of the observed terrain, even if the terrain class itself has been categorized correctly. The differences in appearances (e.g., wet sand vs. dry sand) could be too subtle but nevertheless require significant adjustments to driving control in order to achieve the best dynamic performance. An excellent example of such ambiguity is provided by (Martin, 2018) and shown in Figure 6, where the left image shows loose terrain and the right image shows hard aggregate concrete. Another example of potential ambiguity for different types of soil is shown in Figure 7. Also, sometimes identification may fail due to insufficient training set, current environmental conditions, or prior events like fallen leaves on the ground. In these circumstances, more reliable or alternative information about the current terrain is in the feedback from the proprioceptive sensors. Second, considering that labeling of data for training is usually very time-consuming, it may be possible to use traversability classification obtained from the proprioceptive sensors to provide ground truth for visual- or geometry-based

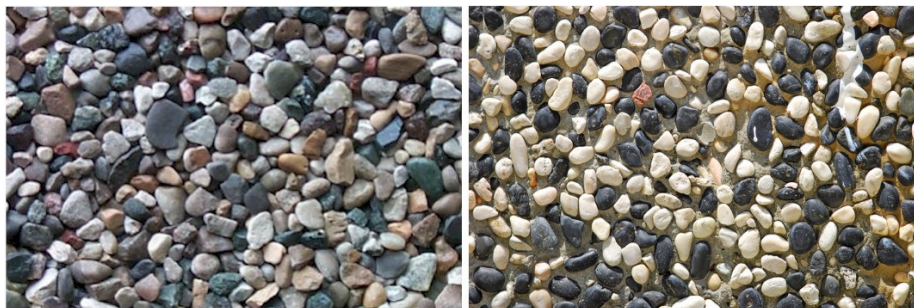


Figure 6. An example of terrain appearance ambiguity for loose gravel and exposed aggregate concrete. Figure extracted from (Martin, 2018).

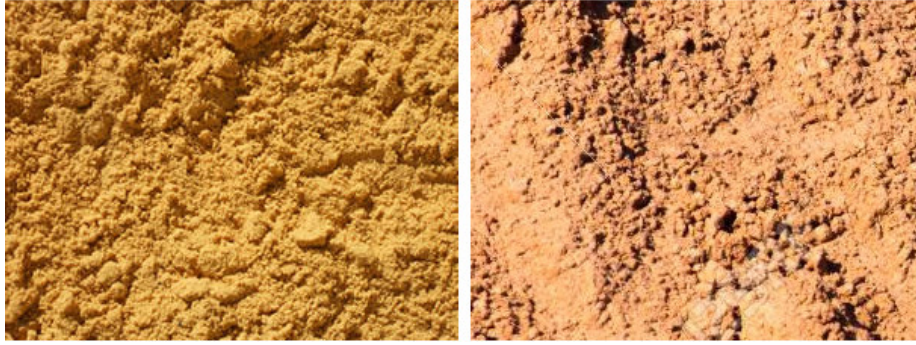


Figure 7. Another example of potential visual ambiguity between wet sand on the left and red clay on the right.

exteroceptive data. Third, proprioceptive sensors usually provide measurements in a single time domain dimension and as such require much less processing power and fewer storage volumes compared to 2D or 3D information from the exteroceptive sensors. Together with very small power consumption, this can be an attractive property for autonomous vehicles, especially of small size (Best et al., 2013). Also, compared to LIDARs or RADARs they are not intrusive and do not emit signals into the environment, which may be beneficial in certain applications (e.g., defence).

6.1. Vibration

It is important to note that vibration-based measurements depend heavily on the vehicle body dynamics (Coyle, 2010). Any terrain classification based on vibration data should take the vehicle mass distribution model into account. This is in addition to nonlinear dependency on the current velocity and acceleration of the vehicle. As such, the classifier cannot be immediately transferred to other types of vehicles without defining the model for the new vehicle. This issue is further complicated if the vehicle should carry some nonconsistent useful load for practical applications. Unfortunately, not many authors address these issues and usually make assumptions on the rigid body mass distribution (same vehicle was used for training and testing) as well as consistent vehicle dynamics.

Vehicle vibration analysis can nevertheless be successfully used for terrain classification taking into account the limitations described above. Spectral analysis of the inertial sensor data at different vehicle speeds were used to extract features for training a probabilistic neural network in (Sadhukhan, 2004). Performance of the classifier was tested on four types of terrain (grass, gravel, packed dirt, and sand) and showed reasonable results at higher vehicle speeds. The work in (Weiss et al., 2006) evaluated the performance of an SVN-based classifier on the data collected from the accelerometer mounted on the vehicle body. The paper proposed a novel feature extraction method which showed better performance than those based on power spectral density of Fourier transforms. Both of the above papers reported that classification rates decrease at lower speeds.

A comprehensive discussion on the fundamentals of terrain classifiers based on inertial sensors, including physics behind vibration sensing and application of frequency-domain techniques, has been presented by Coyle in his thesis (Coyle, 2010). The author details the benefits of several different pattern recognition classifiers, which are compared based on accuracy and computational speed. It was argued that vehicle speed and load dependency are the most difficult problems to address, generally requiring taking these parameters into account while training the classifier. As a result, a large amount of empirical data may need to be collected to ensure good accuracy of the algorithm. A theoretical background for vibration-based terrain classification was presented in (DuPont et al., 2008a; DuPont et al., 2008b). The papers demonstrated that the signature of a particular terrain is given by the magnitude of the spatial frequency response of the system. They have also shown that the speed of the vehicle and the vibration transfer function of the system define a map from the spatial frequency response to the frequency responses of the vibration sensors. As a result, the

magnitudes of the latter frequency responses can serve as speed-dependent terrain signatures. A related approach combining deep-learning with vibration signatures has also been proposed, where the authors present high accuracy discrimination between sand, brick, cement, and soil (Bai et al., 2019).

6.2. Multisensor Approaches

Early work in (Ojeda et al., 2006) explored terrain classification based on a range of onboard proprioceptive sensors such as gyros, accelerometers, wheel encoders, as well as motor current and voltage sensors. In addition, the paper describes some less commonly used sensors (microphones as well as ultrasonic and infrared sensors) and their effectiveness for terrain classification. A multilayer feedforward neural network was trained to recognise five different terrains: gravel, grass, sand, pavement, and dirt. The work used frequency domain response for classification and discussed the benefits of using inertial sensors over other types of modalities. The paper presented theoretical analysis of a strong correlation between motor currents and rates of turn (MCR) and soil parameters, arguing that MCR curves can be used to predict driving parameters for safe handling on the specific terrain.

The doctorate work by Martin (Martin, 2018) proposes methods for a robot to sense terrain, estimate terramechanical properties, build traversability maps, and plan optimal energy paths. The work investigates proprioceptive sensors to estimate the terramechanical characteristics of the terrain over which the robot is driving. By sensing terramechanical traversability online, it is possible to build spatial maps of traversability using the robot's past experience, which can be used for planning minimal-energy paths. To extrapolate the method to areas beyond the path driven, Gaussian Process (GP) regression is used to interpolate traversability estimates. As part of the study in terramechanics, various classification algorithms based on wheel slippage and IMU sensors were tested in (González and Iagnemma, 2018). Deep neural networks and CNNs were compared to more traditional SVMs and multilayer perceptron-based classifiers. One of the advantages of the former methods was that there was no need for any filtering of the input data while maintaining good performance. Interestingly, SVMs outperformed CNNs while detecting high-slip samples, but CNNs worked better at detecting moderate-slip samples. Adding to CNNs, recurrent neural networks have also been used for terrain classification using proprioceptive sensing (IMU and wheel odometry) (Vulpi et al., 2021). In that work, sensor signals are classified as time series directly using both a recurrent neural network and CNN having as input higher-level features or spectrograms from the temporal signals. For both networks, results show comparable performance when contrasted with SVMs. The authors have made an open source package available online.⁴

6.3. Audio Analysis

Iagnemma *et al.* were among the first to propose using auditory sensors for terrain classification (Iagnemma and Dubowsky, 2002). They also described a mathematical framework for wheel-terrain interaction analysis based on simplified forms of classical terramechanics equations with the emphasis on real-time calculations. This work was further developed for the estimation of terrain cohesion and internal friction angle for planetary rovers (Iagnemma et al., 2004). DuMond *et al.* observed a variability in the measurement of cohesion and friction angle when driving on nonhomogeneous terrains, and they developed a stochastic model for estimating these parameters (Dumond et al., 2009). The work in (Libby and Stentz, 2012) specifically concentrated on using sound to distinguish different types of terrain based on acoustic data alone. They applied this method on two types of vehicle-terrain interactions: benign (driving over grass, pavement, or gravel) and hazardous (splashing in water, hitting an object, and losing traction), and they achieved good classification results especially for the latter type of interaction. An investigation of sounds from vehicle-terrain interaction was also performed in (Valada and Burgard, 2017). The authors used a new CNN

⁴ https://github.com/Ph0bi0/T_DEEP

architecture for learning deep spatial features, complemented with LSTM units that learn complex temporal dynamics. The results demonstrated that learning temporal dynamics can improve classification compared to learning only in the spatial domain. In addition, they evaluated the robustness of the model to various types of acoustic noise, from pure white noise to domestic and street noise. One of the conclusions was that without noise-aware training, the accuracy of terrain classification can significantly drop if the signal-to-noise ratio (SNR) is less than 20 dB. An updated version of this technique used unsupervised acoustic feature learning for self-supervised visual terrain classification (Zürn et al., 2021). The results illustrate that the proprioceptive terrain classifier exceeds the state-of-the-art among unsupervised methods and that the self-supervised exteroceptive semantic segmentation model has a performance comparable to supervised learning with manually labeled data.

6.4. Special Cases

A special case of a vibration sensor mounted on a vehicle with shock absorbers and elastic tyres was investigated in (Mei et al., 2019). Any vibration is significantly dampened in this case, which complicates discrimination of different terrains. The authors analyzed seven different classifiers used on three types of features and concluded that a one-dimensional LSTM network provides the best accuracy in these conditions, though it may not be the fastest method for real-time applications. The modeling of the kinematic and dynamic behavior of a skid-steer vehicle allowed development of a robust terrain classification algorithm based on the slippage experienced by the vehicle during turning motion (Reina and Galati, 2016). Using common onboard sensors (wheel encoders, electrical current sensors, and yaw rate gyroscope) the proposed system could characterize four types of terrain (asphalt, dirt road, ploughed ground, and sand) in real time during normal vehicle operations. Proprioceptive sensing for robotic terrain classification was also used for legged robots in (Szadkowski et al., 2018). Although the paper focuses on a hexapod robot, it investigates using an LSTM model to classify terrain into several categories; asphalt, bricks, dirt, office, stairs, and grass based on an angle error from the front two legs of the robot. It achieves strong results in some of these categories (grass, office, stairs) and with lower accuracy in the other categories. It is possible that conclusions drawn from this paper could be adapted to other proprioceptive sensors (e.g., IMU) on wheeled vehicles. Torque and state information from joints in legged platforms have also been used for terrain classification (Ahmadi et al., 2021; Tennakoon et al., 2018; Wu et al., 2016; Best et al., 2013). In general, legged platforms have the advantage of having a higher flexibility in probing the terrain and extracting information about its characteristics.

6.5. Fusion Strategies

In general, using only proprioceptive sensors for any long-term off-road driving would not be sufficient as they only react to the current vehicle dynamics and cannot be used for identifying the surrounding terrain, which would prohibit any route planning as a result. But such inherently reactive nature gives a unique ability to reflect the current driving conditions at the current vehicle speed with the current mass distribution. As such they can play an important part for robust terrain traversability analysis. For example, a joint visual (using color and geometry) and proprioceptive (motion resistance, vehicle slippage, and vibration) data classification algorithm was proposed in (Reina et al., 2018; Reina et al., 2017) to support autonomous operations by an agricultural vehicle.

As discussed earlier, proprioceptive sensors can provide ground truth information for training classifiers based on exteroceptive modalities. A framework for self-supervised training of vision-based classifier was proposed in (Brooks and Iagnemma, 2012), which allowed a robotic system to learn to predict mechanical properties of distant terrain, based on measurements of mechanical properties of similar terrain that has been traversed previously.

The authors developed two classifiers for proprioceptive sensing, one for vibration and another for wheel traction, and a terrain classifier based on visual features such as color, texture, and

geometry. Initial evaluation showed that self-supervised training of the latter classifier by employing classification results from the former showed similar or better results compared to a fully supervised approach. Another study (Bajracharya et al., 2009) describes a fully integrated real-time system for autonomous off-road navigation, which uses end-to-end learning from onboard proprioceptive sensors, operator input, and stereo cameras to adapt to the current terrain. The system is using its proprioceptive sensors as the source of supervision, so that it can learn the mapping of terrain geometry and appearance to traversability online and fully autonomously. At the same time, the image-based terrain classifier is capable of classifying terrain in the far field. As a result, the system can adapt to terrain that it has never seen before and be robust to a changing environment.

A recent study in (Kahn et al., 2021) expands traditional geometric-based traversability analysis using a method that learns about physical navigation affordances from experience. The authors developed a novel navigation system called *BADGR*, which was trained with self-supervised off-policy data collected in real-world environments without any simulation or human supervision. The system was trained on three different events (collision, bumpiness, and position) to generate an image-based, action-controlled predictive deep neural network model. The experiments showed that the navigation system cannot just outperform pure LIDAR based policy in complex real-world environments but can improve its performance as it gathers more data.

6.6. Considerations

Proprioceptive sensors can play an important role in terrain traversability analysis despite their inability to sense characteristics of the “upcoming” terrain. Sensors like IMU and wheel slippage detectors provide valuable information about the current vehicle dynamics and as such can be employed both for vehicle control and terrain traversability estimation. Although their usefulness can be limited if they are used individually, the data from these sensors can be complementary to other sensor modalities for the terrain classification training (various sensor fusion approaches will be discussed in the next section). It is possible to train a classifier purely on the data from various proprioceptive sensors to detect the type of terrain the vehicle is driving on (e.g., sand, rock, or gravel), but it would be more practical to directly use dynamic vehicle parameters, such as traction, slippage, skidding, etc., to assess terrain driveability. An interesting research direction is in developing self-supervised machine learning algorithms for training exteroceptive data classifiers using these immediate parameters as ground truth.

7. Sensor Fusion Approaches

In this section, we present terrain analysis approaches using sensor fusion. To achieve robust and accurate scene understanding, mobile robots and autonomous vehicles are usually equipped with various sensors and multiple sensing modalities that can be fused to exploit their complementary properties. There exist various ways of fusing these diverse data (e.g., from stochastic or simple concatenation). The choice of the appropriate fusion strategy is important and depends on the input data available and the objectives and priorities of the application.

7.1. Fusion Strategies

Figure 8 shows two common strategies for sensor fusion that apply to several methods that can be used for terrain analysis. The first strategy, shown in the top figure, initially processes the data in a single modality (or device). Subsequently, the results are merged with simple voting or more sophisticated approaches that consider the amount of noise in each sensor or sensing modality. The bottom figure illustrates a more advanced strategy, where the data from the sensors are initially merged and then the resulting *combined* data are processed.

Regarding the types of sensors, in this section we focus on exteroceptive and proprioceptive sensors and their fusion. Probably the most common example is the combination of LIDAR and cameras, although many more exist.

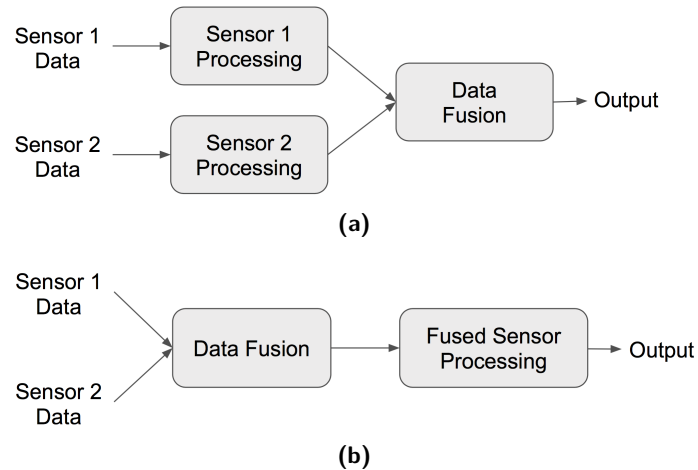


Figure 8. Illustration of the different sensor fusion strategies that are commonly seen in methods that can be applied terrain analysis. The top figure illustrates the “process & fuse” strategy while the bottom figure shows the “fuse & process” approach. Both strategies rely on time synchronization and spatial calibration of the sensors to work correctly.

A strategy to estimate traversability using a LIDAR-camera is to check for statistical coherency between the data extracted from both modalities (Aeschmann and Borges, 2015). Using a 2D LIDAR and a stereo-camera, the stereo pair can provide 3D ground shape information, however its depth observations are generally noisier than that from a LIDAR. This can cause false-positives/negative obstacles to occur. If the 3D data are checked with a more precise 2D range sensor in points of interest, the likelihood of errors is decreased. Because visual stereo data are often heavy to process, fusion can be done after the 3D data are reduced to a DEM representation (Oniga and Nedeveschi, 2009).

Alternatively, the 3D LIDAR and single camera setup also presents advantages. In the method presented in (Shinzato et al., 2014), terrain analysis is based on fusion of sparse and unstructured 3D point clouds and images. As with most fusion methods, it requires not only accurate time synchronization but also extrinsic calibration between sensors. In this case, the calibration should make it possible to transform a 3D point from real world in the 2D image coordinate. The core idea is to use spatial relationship in image perspective view (birds-eye) combined with 3D range values to determine if a point corresponds to an obstacle or not. Subsequently, polar histograms are used to generate a confidence map that represents the ground area in the camera view.

Feature representations from different sensing modalities can be fused at early, middle, or late stages (Figure 9).

- Early fusion combines different sources of data as early as possible, before the interpretation of the sensor data. For example, two exteroceptive inputs such as color information and depth are combined to produce a colored point cloud, or depth is added to an RGB image to produce an RGB-D image. Redundant information then needs to be fused in a nonconflicting manner, which can be challenging as different sensors might perceive the world in different ways. This colored point cloud or RGB-D image is then used as input to a single classifier, which benefits from more input data, usually allowing for more discrimination power than using a single source of data. However, if one of the sources of sensory data is corrupted, this can negatively affect the fused output.
- Late fusion usually combines output at the latest stage. An example is the fusion of two or more classification outputs that were performed independently. For example, one classifier might use vision only, and the second classifier uses LIDAR data only. The combination of those outputs may use some form of confidence in each classification. Another common strategy is to use three different classifiers and adopt the outcome that at least two of them produce (Korthals

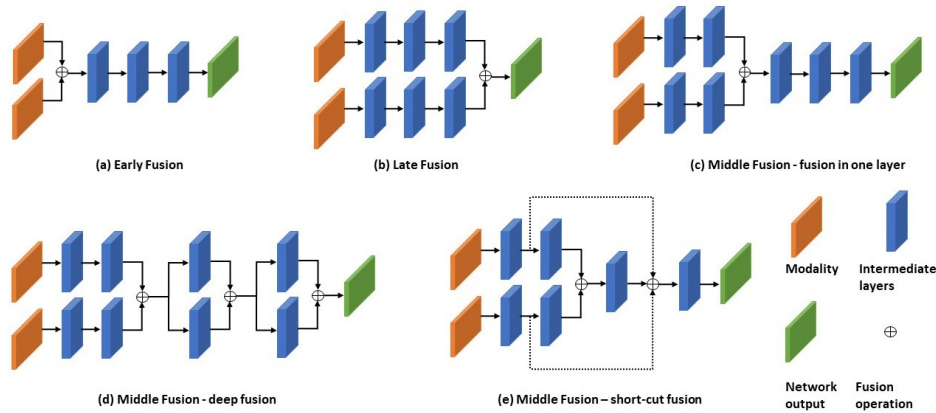


Figure 9. Illustration of early fusion, late fusion, and middle fusion methods used by multimodal fusion networks.

et al., 2018). The main advantage of late fusion is to be more reliable if one of the sources of data is corrupted; even though one of the classifiers may make errors, the other classifiers can still produce valid outputs.

7.2. Multimodal deep learning fusion network

In the context of multimodal fusion for terrain segmentation using learning, three main fusion operations can be summarized as follows (Feng et al., 2020):

- Addition or Average Mean: The feature maps of multiple modalities are either added element-wise, or averaged.
- Concatenation: The feature maps from different modalities are usually stacked along their depth before they move to a next layer.
- Mixture of experts: The feature map of each modality is processed by its domain-specific network called “expert.” Afterwards, the outputs of multiple expert networks are averaged with the weights modeled by the gating network, which takes the combined outputs from all the expert networks as input.

A convoluted mixture of deep experts architecture that fused segmentation masks from different modality networks including RGB, depth, and infrared was proposed in (Valada et al., 2016a). More recently, the authors introduced a fusion module that could dynamically fuse intermediate feature maps from multiple modalities according to the object class, its spatial location, and the scene context (Valada et al., 2019).

Recent work in (Wang, 2019) discusses proprioceptive, visual, and LIDAR sensing modalities and their combinations. Feature sets are vibration frequencies from accelerometer, co-occurrence matrix for visual texture classifier, and power spectral density for LIDAR. The author employs SVM and principal component analysis (PCA) based classifiers for all three modalities. The results show a significant increase in classification accuracy for multimodal classification over single modalities.

Earlier work concentrates on obstacle detection methods, but discusses terrain classification for traversability (Manduchi et al., 2005). The authors use stereo range measurements that do not rely on typical structural assumption on the scene (such as the presence of a visible ground plane). They use a color-based classification system to label the detected obstacles according to a set of terrain classes and an algorithm for the statistical analysis of LIDAR data that allows to discriminate between grass and obstacles (such as tree trunks or rocks), even when such obstacles are partially hidden in the grass. Terrain classes include soil/rock, green (photosynthetic) vegetation, dry (nonphotosynthetic) vegetation (which includes tree bark), and a “none of the above” class. The authors show that photosynthetic vegetation displays distinctive spectral characteristics.

A recent multisensory terrain classification algorithm with a combination of geometric and semantic features is presented in (Schilling et al., 2017). It focuses on urban navigation using road, sidewalk, vegetation, and terrain classes from the CityScapes dataset (Cordts et al., 2016). The authors employ transfer learning to adapt the model to off-road environments (e.g., classifying snow). They use late fusion to combine the visual and geometric (point cloud) features using a random forest algorithm to classify the terrain traversability into three classes: safe, risky, and obstacle. Also focusing on complex terrains (caves, collapsed buildings), a multimodal fusion network has been proposed in (Nguyen et al., 2020), where the authors focus on simulated data to train a network using LIDAR and visual data. They show examples of successful transfer from simulation to operations on a real robot.

It is worth considering using other spectral modalities to assist terrain classification. The work in (Bradley et al., 2007) specifically explores near infrared (NIR) response of green vegetation for chlorophyll detection, and it shows that a simple pixel-by-pixel comparison between red and NIR reflectance, normally referred to as a vegetation index or a band-ratio, provides a powerful and robust way to detect vegetation (often seen as obstacles in off-road terrain).

Over the course of preparing this survey, we found a few papers that attempted to exploit the fusion of active sensors such as LIDAR or RADAR (Guerrero et al., 2015; Peynot et al., 2010b; Milella et al., 2014) but not as many as the camera-LIDAR combination. This is mainly because 1) operating and processing a RADAR sensor is challenging due to low SNR, and 2) the relatively large size of a sensor. Obviously, there are unique characteristics of the use of RADAR such as longer range detection (~ 100 m) and material-specific amplitude returns which can provide useful information for terrain analysis (e.g., varying intensities for vegetation and rocks).

7.3. Considerations

We presented different approaches of multimodal fusion spanning from early to late fusion. Whereas each approach shows promising results but comes with its own challenges, aside from sensor cost and weight aspects, the tradeoff to consider is usually performance and computation efficiency. For an agile robot, the early-fusion approach is arguably recommended due to its simpler complexity and more feasible real-time performance than others. Research shows that choosing the right sensors for the fusion in respect to the terrain that should be detected is crucial to enable a high performing system. The more complementary sensors are, the more object properties can be captured and fused—challenging terrain such as mud and water is likely to need additional sensory input to be detected reliably. For instance, knowing that vegetation reflects on water surfaces, a dedicated method just for these challenging tasks might be appropriate, and it would then possibly overwrite other methods.

8. Major Challenges

In this section, we discuss some common key challenges that significantly affect the performance of terrain traversability analysis systems. Although some of those aspects have been discussed as part of the methods mentioned throughout the paper, we focus on the specific characteristics that create the challenges and methods that aim to overcome them.

8.1. Challenging terrain classes: Water and mud

Some terrain classes that are of particular interest for terrain traversability estimation have proven to be challenging to distinguish; these include water and mud, to which we dedicate this subsection. More so than other terrains, these classes might be traversable in some areas and not in others, while not giving many distinctive features to differentiate. Additionally, traversing through these areas might cause deformations of the terrain, and it is especially risky for UGVs to slip on. In the literature, these classes are usually considered individually, and classification is often binary.

Therefore, in this section we specifically discuss research that has been carried out to detect these classes, show their individual challenges, and the results that have been obtained when doing binary classification within learning- and model-based methods.

8.1.1. *Water Detection*

When trying to detect water with vision-based methods, one of the main challenges is that surrounding terrain may reflect off the water body and its appearance may be visible on the water surface. In addition, some areas might reflect the sunlight or bright areas of the sky. In some methods (Nguyen et al., 2017; Rankin and Matthies, 2010) reflections are leveraged to gain an initial guess of water occurrences in an image. Another challenge is that water bodies change appearance based on their distance and angle with respect to the perceiving sensor. Therefore, most successful methods combine multiple water cues to find all water bodies in an image and rarely rely on one single water property. This suggests that a late fusion approach for water detection is likely to perform best, as shown in (Rankin and Matthies, 2006). With respect to temporal changes, water also tends to not present extreme variations (apart from reflections) from the point of view of a moving observer from frame to frame, making this a distinct feature (Borges et al., 2008). Additionally, some information is processed or assumed in order to increase performance. Knowledge about the ground plane and surrounding terrain or sensor position can be used to determine reflective angles and to reject areas of space where water surfaces would not be expected, or modulate the probability of occurrence of water in parts of an image frame (Borges et al., 2008). For instance, if a different type of terrain is classified (significantly) below ground plane level, it is likely to be a reflection, as shown in (Rankin and Matthies, 2006). Note that none of the aforementioned methods use active sensors for water detection, though some add a LIDAR to gain a better understanding about ground-plane and surroundings with respect to the camera or movement in-between frames.

Another approach that uses temporal information is proposed by (Santana et al., 2012). This algorithm detects water by extracting its dynamic change in texture over time. It is a model-based method and aims to recognize chaotic changes in consecutive frames. The extraction happens based on the chaotic movement of the water body's optical flow. Usually, when rigid objects move across a scene, most optical flow vectors on this object point towards a similar direction, whereas with water the optical flow vectors contain more entropy. Even with camera movements, most neighboring flow vectors follow a similar direction. The authors expand the region to fill an appearance-based segment and expand the boundaries to include regions that do not have the chaotic optical flow but still fit into the same color segment. Evidently, this method needs a moving water body, either disturbed by wind or in a flowing state, such as rivers or creeks.

Other methods [e.g., (Nguyen et al., 2017)] have leveraged input from additional sensors to enhance performance, for example using polarized light sensors and stereo information. In this work, the reflective properties of polarized light were modeled to detect highly reflective areas of a certain wavelength, and then to train a Gaussian mixture model (GMM). The method first calculates a disparity map, then estimates a ground plane and computes reflection and azimuth angles with respect to the ground. Finally, the GMM is trained from pixelwise labels.

The work published in (Rankin and Matthies, 2010) proposed an explicit model considering especially reflections and color variation. The authors found that color works well for closer proximity, whereas reflection works best when parts of the sky are reflected in water bodies that are farther away. First, the method finds regions in an image with low texture but a high intensity compared to the surrounding area. These areas are then passed as possible water candidates, after which they propagate around this area using a flood-fill algorithm, by using the intensity gradient as a threshold to determine the edge of the water body. If intensity decreases rapidly, it is likely to be a boundary of the water body. These regions are then merged into ellipsoids and the overall density is used to further determine their likelihood to be a water body. Lastly, the approach combines both methods to increase accuracy and allow detection of water near and far from the camera.

A method that has a high focus on tractability with a UGV has also been developed by Rankin et al. (Rankin and Matthies, 2006). In this work the authors found that the combination of multiple

methods and sensors might bare the best overall robustness. Here they performed late fusion on multiple cues for water that can be found in a 2D image. The actual water detection was done by using a combination of multiple explicit water models. The broad combination of multiple methods seems to enhance the performance of the algorithm. Multiple cues indicate the initial position of the water bodies. These methods are based on color, texture, stereo range reflections, and zero stereo disparity. Hue, saturation, and brightness levels were tuned to get water cues from its color. Stereo range data were used to find patterns with range reflections, reflections of trees, or other terrain extending below ground level. Additionally, zero disparity areas on the ground have also been found to indicate water. Afterwards, ground detection was used to estimate the water bodies elevation by considering the surrounding ground, and temporal filtering in the world map was used to further increase accuracy. Estimating the ground plane helped to embed the water body into a 3D map with a more accurate depth/elevation. All water bodies were detected from at least a 7 m distance, with only 0.2% false positives. This method shows how multiple cues can be included to boost performance.

Recently, learning-based approaches for water detection have also been proposed (Li et al., 2019)(Wang and Wang, 2019). The two methods use the same dataset for training and testing, and their results should be highly comparable. Even though they do not compare results within their papers, both compete against a similar baseline achieved by using a simple FCN structure, and both succeed in achieving better performance. One proposes a neural network structure to include temporal knowledge (Li et al., 2019), whereas the other optimizes on the network architecture and activation function (Wang and Wang, 2019). Both methods show improvements against simple FCN architectures on the Puddle-1000 dataset (Han et al., 2018), where region-independent filters were trained to abstract texture or color properties to do binary classification on water bodies.

Previously discussed methods exploit a combination of known properties in order to detect water with vision sensors. Overall, using a dedicated method for water detection and excluding its detection from further processing seems to be most effective. More recently, a water-dedicated binary classifier using deep learning proposed in (Wang and Wang, 2019) showed the most promising results when substantial labeled data for training are available. Note that existing applications for UGVs usually classify water as an obstacle, and they do not investigate the potential traversability of the water bodies.

8.1.2. Mud Detection

In the context of terrain classification, there are two broad categories of mud: a) wet soil, which appears darker than the dry soil or sand; and b) water puddles in the soil, which is the combination of water and mud. The latter could be detected using water detection methods. In this section, we review the methodologies for the classification of mud, which appears as isolated wet soil surrounded by dry soil in nominal weather conditions. Imaging sensors such as stereo cameras were used to segment the darker soil from the surrounding region in the daytime (Rankin and Matthies, 2008). Stereo cameras are usually used to identify tall vegetation from mud and soil in the cluttered natural scenario. In (Rankin and Matthies, 2010), a stereo camera was used to detect and remove ground clutter by using terrain elevation measurements, estimated local tilt of the terrain, and local plane-fit residual. The local tilt was estimated using the least-squares plane fit, given the minimum elevation measurement within each grid cell over $1.2 \text{ m} \times 1.2 \text{ m}$ in a 40 cm resolution grid map. The resultant map, without the ground clutter, could further be used to classify mud and other classes. Color could also be used as a feature to distinguish mud from shadow. However, segmenting the shadow in mud or when the entire field is wet (for example, after rain) is difficult using only a color camera. The water reflection cues could be useful during the bright daylight conditions. Using color, mud detection is performed by detecting edges and finding a threshold to isolate darker regions (mud) compared to surrounding regions (dry soil) (Rankin and Matthies, 2010). The authors found that the mud detection based only on color is not useful to distinguish between tree shadows and mud, particularly when the entire field is wet.

Polarization contrast provides a simplified measure of the degree of linear polarization at each pixel value, depending on the filter orientation (Pandian, 2008). It was observed that the polarization

contrast for water detection can be changed based on weather conditions. However, for mud detection, the contrast was higher than the surrounding dry soil and is not affected by the weather conditions. Therefore, the degree of linear polarization can detect the mud robustly, depending on the weather conditions. It is worth mentioning that the polarization contrast for wet and dry soil in the shadows is similar. Therefore, it is difficult to classify mud using only the polarization contrast in the shadows. In (Rankin and Matthies, 2008; Rankin and Matthies, 2010), the polarization contrast was tested to detect mud in the daytime nominal weather conditions along with the stereo camera on the robot. The polarization pixels were projected onto the left stereo camera image to classify the mud within the world map. The mud patches were at two different locations in the test scenario. The first mud patch was detected at a range of 30.5 m along with a second mud patch at a range of 48.1 m. The second mud patch was a cluster of small spaced mud bodies, which appeared to be a single mud area in the polarization imagery. Both mud plots were classified correctly, without false positives, in the test scenario and at a comfortable range from a vehicle for decision-making purposes. In the literature, mud, sand, and vegetation detection methodologies also used alternate exteroceptive sensors and proprioceptive sensors, which are discussed in Sections 5 and 6, respectively.

In all reviewed work on mud detection, multiple sensors were used to classify mud areas in nominal weather conditions. Detecting mud in shadows and in wet weather conditions is a challenging problem and requires a multimodal approach. From the literature, it is evident that mud detection in natural and cluttered scenarios is still an open research problem, in particular using the passive sensors. To our knowledge, traversability through mud using vision approaches is not yet considered in the literature.

8.1.3. Water and mud traversability

Finally, terrain classes like water and mud are challenging specific cases because the traversability (rather than just detection) depends on a number of factors, many of which are not directly observable by any sensors. Like the general case of obscuration already discussed, water can obscure other traversability-affecting terrain types or obstacles which lie beneath the surface. The depth of water or mud is also difficult to detect with normal sensing modalities, although there has been some limited success with polarized RGB-Depth sensors (Yang et al., 2017) and LIDAR (Matthies et al., 2003).

8.2. Vegetation

Vegetation poses a number of challenges for evaluating traversability (Wellington and Stentz, 2004). It depends on the nature of the vegetation—for example, a small bush versus a thick tree trunk—and on the ability of the combined sensing and algorithmic system to successfully detect these relevant characteristics. Traversability of vegetation is also complicated because, unlike predominantly ground-plane-based classes like water or sand, vegetation exists in the volume above the ground plane, and subtle variations can result in large differences in traversability. For example, a tree-trunk anchored to the ground may be untraversable, but that same thickness in a horizontal branch may be traversable through bending the branch, if its base is far enough from the path of the vehicle. Finally, vegetation-based traversability cannot be determined entirely in isolation but must also occur jointly with assessment of the traversability of other elements in the scene. For example, the traversability of a small tree may be irrelevant if that tree is submerged in water that is sufficiently deep to trigger a “nontraversable” flag.

Vegetation can also obscure or interfere with assessment of other terrain traversability characteristics, most notably by partially or fully obscuring other terrain types or even obstacles, such as large rocks. This obstruction is relevant to both visual and most range-based sensors such as LIDAR. Some sensing modalities such as radar offer the potential for limited detection through foliage (MacDonald et al., 1981), offering the ability to detect major physical obstacles—like a large rock—as discussed in Sec. 5) but limited ability to detect and recognise different types of terrain when fully obscured by foliage, such as distinguishing sand from dirt.

8.3. Negative Obstacles

Negative obstacles such as cliffs, ditches, ruts, or other depressions pose a difficult problem for autonomous off-road navigation. Negative obstacles exist on unpaved road surfaces but are especially common when traversing natural terrain with highly capable vehicles, such as Crusher (Stentz et al., 2007), or the Legged Squad Support System (LS3) (Bajracharya et al., 2013). Vehicle mounted sensors cannot easily see negative obstacles as the near-field terrain occludes the trailing drop, slope, or rising edge. The difficulty of seeing negative obstacles from ground level prompted the DARPA PerceptOR program to detect them from the air (Matthies and Rankin, 2003). Compared to a positive obstacle, occlusions and viewing angles result in fewer pixels-on-target (Matthies and Rankin, 2003), which in turn reduces the effective detection range, often to within the stopping distance of ground vehicles moving at any appreciable speed.

Negative obstacles are often defined by occlusion, rather than being a distinct observable class such as a tree or rock. From a distance they can only be seen as a discontinuity, making image-based detection difficult. For example, a ridge in an undulating pathway hides the terrain beyond, even if the pathway continues and can be seen in the distance. In off-road navigation, traversability determination has to consider that this ridge could hide a cliff, washout, or crater, whereas on-road vehicles make use of prior maps and road continuity assumptions. Even as a vehicle approaches the negative obstacle and gains observability into it, the underlying terrain may very well be of the same class as the surroundings, making labeling difficult in image space.

Most existing methods for negative obstacle detection are fundamentally geometrical, measuring a drop in terrain elevation, or identifying unknown or unobserved areas surrounding the vehicle. These methods commonly use either stereo vision (and methods presented in Section 3.1) or LIDAR sensors (and associated methods in Section 4.1). For both LIDAR and stereo, classification of the surrounding terrain (i.e., into vegetation or a solid surface) is critical as the occlusions caused by vegetation can confound purely geometric approaches. Thermal imagery, exploiting the differential cooling of depressions in the ground at night, has been used for negative obstacle detection (Rankin et al., 2007), but there are no prominent methods that depend only on monocular color camera imagery in outdoor terrains.

8.4. Presence of Airborne Dust, Smoke and Fog

Environmental conditions affect different sensors differently. Airborne dust (a common phenomenon for UGVs operating in dry terrain) has been identified as a challenging problem in perception due to the creation of false-positive obstacles in LIDARs (Peynot et al., 2009), hence generating a wrong representation of the terrain. These errors are usually due to the misinterpretation of the perception system, which tends to be considered sections of LIDAR point clouds due to airborne dust particles as obstacles. Many recent LIDAR models can provide multiple echos/returns, which allows for some mitigation of this issue, in cases of light dust. In addition, recent studies showed that it may be possible to detect which LIDAR points are due to airborne particles such as dust and snow (Stanislas et al., 2019).

A similar effect is observed with rain (or snow), where the water particles cause misleading LIDAR returns as well as attenuation. Studies have shown that the humidity in the air acts as a screen for the infrared radiation (Weichel, 1990). Both fog (Ijaz et al., 2013) and rain (Filgueira et al., 2017) reduce the signal intensity by absorption and diffusion phenomena of the beam by the small water particles. Fog and rain act then as a screen on LIDAR sensors that limit their capabilities and detection range. Cameras, in contrast, are generally more robust to a certain amount of dust (Borges et al., 2010; Peynot and Kassir, 2010) or rain in the environment. Although cameras may suffer from a loss of contrast in those circumstances, the effects can be partially mitigated with adequate image processing, making many vision-based terrain analysis methods applicable (evidently, the performance depends on the amount of dust or rain). Care must be taken, however, for dust or water not to be deposited on the sensor itself. Solutions have been engineered to ‘clean’ the sensors, with wipers (Ingram et al., 2020) that can remove water, dust or snow.

Smoke is a problem for both LIDARs and cameras, and once again, it depends on how severe. Smoke can obviously significantly affect the visual information from cameras. Standard LIDARs also tend to fail in smoke, although more modern multi-echo return LIDARs are more robust to the phenomenon. Thermal IR cameras can reliably see through smoke and have been successfully used, often in combination with visual cameras, for robot navigation in such conditions (Brunner et al., 2013).

Other sensors like sonar and radar work much better in penetrating smoke, dust or fog. As discussed in Section 5, however, they often lack the resolution of LIDARs and the richness in information of cameras and suffer from much lower signal-to-noise ratio. This can be a problem for creating detailed maps of the terrain or for classification. However, this can be mitigated by an intelligent combination of sensor data provided by LIDAR and mm-wave RADAR, as in (Gerardo-Castro et al., 2014). Sonar can also suffer severely from multipath effects, making mm-wave RADAR usually better for imaging in fog, airborne dust, and smoke. Recent RADAR developments driven by a rise in demand from the automotive industry, in particular self-driving cars projects, have led to the availability of affordable and compact RADAR units, however they are usually limited to obstacle detection and tracking. To perform terrain classification and advanced traversability analysis, higher-end imaging RADAR are required, such as in (Peynot et al., 2010b), but they remain relatively expensive.

As indicated in Section 7, it is particularly important that UGVs are equipped with multiple sensors with different physical properties, such that redundancy between sensors can be exploited. The specific choice of sensors will depend on the application scenario and the likely elements to be encountered during operations.

8.5. Extreme Illumination Challenges including Night Operations

The day-night cycle is one of the most dominant challenges for all visual perception systems and presents a range of challenges including low light and high dynamic range situations. The first and most obvious is that illumination at nighttime is generally reduced, causing challenges for visual camera-based perception systems that have been primarily developed for well-illuminated conditions. In the past, infrared or multispectral sensors have been used to mitigate these issues, but they are not always a panacea for operations in low light. For example, in natural environments, temperature differentials across much of the environment can reduce to near zero during a night cycle, rendering the environment near-featureless in appearance to a heat-based sensor (Maddern et al., 2014). Advances in visible spectrum camera technology over the past decade have resulted in relatively lower cost camera hardware that can see at least as well as the human eye in low light conditions—and with specialist hardware, in almost pitch black conditions (Mount and Milford, 2016).

One of the attractive properties of these developments is that the image representation produced at night is similar in appearance to that produced during the day, in terms of color and intensity representation, unlike a thermal camera (Maddern et al., 2014). Consequently, a range of techniques developed for daylight conditions is adaptable for nighttime conditions, as opposed to requiring entirely new development, as is the case with alternative visual sensing modalities like thermal cameras or event cameras (dynamic vision sensors) (Kim et al., 2016). With these improvements, visual detection at night becomes more tractable (although issues such as blurring remain problematic). Algorithmic attempts to address perception for vehicles in low light conditions by integrating poor quality information over time to produce improved performance have been developed (Milford and Wyeth, 2012). Nighttime operations are not entirely without advantage either: in environments without external illumination, much of the challenge of dealing with shadows encountered during the daytime is removed (Corke et al., 2013).

For vision-based systems, the second challenge in nighttime operations concerns the range of illumination conditions that are encountered. Factors including artificial lighting from vehicle and personnel can introduce major variations in illumination across the scene that can rapidly vary from one moment to the next. This nighttime challenge is in many ways the more challenging of the two, because it is not simply a question of everything in the scene being darker, but rather one

of dealing with a potentially very wide range of illumination conditions that can vary rapidly. Once again, nonvisual sensors are largely unaffected by such illumination challenges, but incorporating visual sensing is relevant to obtaining improved terrain traversability performance. As new hardware developments in camera technology have enabled performance in low light conditions, they have also led to increased camera dynamic range—the range of variation in intensity in a scene that can be captured simultaneously. This improved hardware performance is still likely insufficient to deal with both direct illumination by artificial lighting and completely unlit areas of a scene. Multiple camera setups can be implemented, such as is employed in some autonomous vehicle systems where a dedicated camera is used to reliably detect traffic signals (Diaz-Cabrera et al., 2015). Extreme lighting variations are not only encountered at night: daytime operation, especially around sunrise and sunset, can result in challenging conditions due to the low position of the sun in the scene.

Naturally, nighttime operations are not a challenge for systems based on active sensors such as LIDARs, however relying purely on those sensors can be a liability in some applications where passive sensing is preferred.

8.6. Deformable and Unstable Terrain

Most terrain traversability studies consider the terrain to be *rigid*, i.e., the terrain shape and characteristics are static and they never change, even over time. However, few studies consider the case of *deformable* terrain, where the terrain (usually its geometry) can change as a result of the interaction with the robot and an attempt to predict this terrain deformation (Ho et al., 2013). For example, an unstable pile of rock may change shape once a robot drives over it, due to the weight applied to it (Ho et al., 2016), mud may be moved as a robot slips on it, or low cohesion soil such as sand could move. In extreme cases, some authors consider terrain *collapsing* when a patch of terrain cannot hold the weight of the robot (Tennakoon et al., 2018; Tennakoon et al., 2020). Examples include a rotten wooden floor or holes covered with leaves or thin ice.

8.7. Soiled Camera Lens due to Mud, Dirt, Water, Foam

Here we address the issue of disruption to the camera lens itself rather than the environment in front of the lens, caused by mud, water, or other environmental conditions that decrease visibility and therefore affect the performance of the learning-based system. The first set of approaches to this problem involve attempting to correct for the disruption while retaining access to the information in the image. In (Uricar et al., 2019), a Generative Adversarial Network (GAN) is used to generate an augmented soiled dataset for training a model. CycleGAN was used to generate the soiled version of the clean images, captured using the fisheye camera (Zhu et al., 2017a). The authors have trained two DeepLabV3 models (Chen et al., 2018b), using the WoodScape dataset (Yogamani et al., 2019), on the clean and the soiled images. ResNet50 and FCN8 are used as an encoder and decoder to develop the binary semantic segmentation network. To evaluate the performance, the model trained on clean images was tested on both clean and soiled images. A drop of 21.8% mean Intersect Over Union (IoU) was observed by the model trained on the clean images, when tested using the soiled images. The decrease in the performance of typical classification tasks, common for autonomous driving, suggests that the epistemic uncertainty in the model could be reduced using the soiled dataset.

Recovery of information from the image may not always be possible; in these cases, reliable detection of the disruption is still useful. This can be achieved through various methods including relatively straightforward learning-based techniques such as in (Zeng et al., 2017) and (Zhang et al., 2014), which learn to predict the utility of the image based on relevant training data.

8.8. Other Challenges

Terrain ambiguity is the general challenge of terrain types with different traversability implications being indistinguishable using one or more of the sensing modalities, even for human observers. For

range-based sensors like LIDAR or range-producing sensors like stereo vision, this ambiguity can result from terrain types that have very similar geometric texture. For vision sensors, some terrains can have highly similar visual appearance in certain conditions, such as loose sand versus compacted wet sand.

Obscuration is another common challenge that has long been known (Schwartz and Sharir, 1987) in the computer vision literature: although a system can detect the terrain types present in the current image, there may be critical terrain types obscured by parts of the scene (such as foliage, or leaf cover on the ground plane). By the time those obscured terrain types become observable (if ever), it may be too late—for example, if the vehicle has pushed through the foliage only to drive directly into a deep pool of water. The complementary problem is also possible: a shallow pool of undisrupted water may not be directly detectable using visual techniques, which would instead only see the underlying terrain type such as dirt.

9. Open Resources: Datasets and Open Code

This section lists some of the relevant publicly available semantically annotated datasets and open code that find application in terrain analysis.

9.1. Datasets

While there are multiple publicly available autonomous driving datasets, only a few focus on off-road⁵ navigation. The vast majority of semantically annotated driving datasets have been captured for urban navigation, and the emphasis is on common objects that appear in an urban scene, such as roads, cars, pedestrians, traffic signs, or buildings. Nonetheless, many of these datasets include relevant off-road terrain classes, such as road-side vegetation, gravel, snow, and water, and they represent a similar viewpoint. An off-road terrain detection system can be initially trained on one of these urban navigation datasets and then fine-tuned on the (much smaller) off-road datasets, if required. For this reason, this section briefly summarizes both urban and off-road datasets.

Most of these annotated datasets contain single sensor modality, in the form of visual information, mostly in the form of images (Neuhold et al., 2017; Procopio, 2007; Zhou et al., 2019) or video-sequences (Yu et al., 2018; Brostow et al., 2009). Some datasets (Cordts et al., 2016; Valada et al., 2016b; Huang et al., 2018) are captured using a stereo-camera setup, which, in addition to visual information, can also provide depth information. A couple of datasets also contain a geometric profile of the terrain captured by LIDARs in the form of a point cloud (Geyer et al., 2020; Sun et al., 2020). The terrain geometry data can be either stand-alone or synchronized with visual sequences.

9.1.1. Urban Navigation Datasets

One of the most popular autonomous navigation datasets is Cityscapes (Cordts et al., 2016), which has been used by many researchers for training and benchmarking of self-driving cars. Although its primary purpose is urban navigation, it contains several useful terrain classes: road, sidewalk, rail tracks, and vegetation. The Mapillary Vistas dataset (Neuhold et al., 2017) is another large and diverse urban dataset with 152 object categories with several useful terrain classes such as water, snow, sand, vegetation, road, and terrain. BDD 100 K (Yu et al., 2018) is the largest (120 000 000 images) and most diverse publicly available urban dataset. It has similar class specifications to Cityscapes while also including datasets for driveable surface labeling, semantic instance segmentation, etc. Another popular dataset was built by LabelMe (Russell et al., 2008), a community-based online annotation tool for general computer vision research. Being an open tool, it has a significant number of object classes, of which some classes (such as road, field, grass, river, plant, sand, rock, desert) are relevant to terrain classification.

⁵ “Off-road” as defined in Section 1.

9.1.2. Off-road Navigation Datasets

Two of the most relevant off-road datasets are the Freiburg Forest dataset (Valada et al., 2016b) and the RELLIS-3D dataset (Jiang et al., 2020). The Freiburg dataset is a semantically labeled dataset of unstructured forest environments with six classes: obstacle, trail, sky, grass, vegetation, and void. It was collected by a robot driven on a narrow forest road in a German province over 3 days in various lighting conditions. It includes the following sensor modalities: the robot's odometry, a Velodyne HDL 64, four Bumblebee stereo cameras, and an Applanix navigation system. The benefit of this dataset is that with such a rich set of modalities, it is possible to evaluate various types of joint classifiers. The RELLIS-3D dataset is a multimodal dataset collected in an off-road environment containing annotations for 13 556 LiDAR scans and 6235 images. The data were collected on a university campus and present challenges in terms of class imbalance and environmental topography. The dataset contains RGB camera images, LiDAR point clouds, a pair of stereo images, high-precision GPS measurement, and IMU data, all in an ROS format. This is the most comprehensive dataset available for off-road data.

Another notable dataset captured in nonurban environments is DARPA LAGR (Procopio, 2007). It consists of three off-road scenarios for two lighting conditions, resulting in six image sequences. The data are stored in a Matlab-6 compatible format and represent various terrains (mulch, dirt, grass) as well as natural obstacles (trees, dense shrubs, hay bales). Unfortunately, the dataset has only three labeled classes: Obstacle, Ground-plane, and Unknown. It would require further labeling if this dataset is to be used for terrain classification purposes. A related dataset focusing on images has been published recently (Dabbiru et al., 2021), where the authors include the type of vehicle and consider this variable for traversability evaluation.

The Marulan dataset (Peynot et al., 2010b) is collected in variable environments using synchronized sensors, which include four 2D laser scanners, a radar scanner, a color camera, and an infrared camera. This dataset includes the presence of airborne dust, smoke, and rain.

9.1.3. Vision-LIDAR synchronized Urban Navigation Datasets

Just like vision-based public datasets, most Vision-LIDAR synchronized datasets such as the Audi dataset (Geyer et al., 2020) and Waymo open datasets (Sun et al., 2020) are of urban cityscapes. The Audi dataset includes about 40 000 frames of synchronized and semantically segmented images and point cloud labels, and another 12 000 frames of 3D bounding boxes. The semantically segmented vision and corresponding synchronized point cloud data are categorized into 38 classes, such as pedestrian, car, vegetation, etc. This dataset was captured using six cameras and five LIDARs and has a 360 degree view. The Waymo dataset contains about 1950 segments of independently labeled 3D 7-DoF bounding box labels for LIDAR data, and 2D bounding box labels for camera data. Each sensor's data are synchronized and annotated into five classes: vehicle, pedestrian, cyclists, signs, and no-label. This dataset was captured using five LIDARs and five cameras pointing front, front left, front right, side left, and side right.

Tables 4 and 5 provide a further summary of various relevant semantically annotated datasets that can be used for training and testing of terrain classification, along with the useful classes, and information on where to find them. It should be noted that not all datasets can be used directly for training an autonomous vehicle's perception system. Some of the reported datasets in Tables 4 and 5 have a limited amount of data, while some of the publicly available off-road sequences have a limited number of terrain classes, as discussed previously. Image augmentation and/or expanding the class range by additional labeling (e.g., a drivable surface could be further classified into asphalt, gravel, or a dirt road) should be considered.

9.1.4. Synthetic Datasets

It is worth mentioning that off-road synthetic datasets can be generated using simulation frameworks to produce purely virtual environments. Although there is always a danger that a classifier trained on computer-generated data may not perform as well in natural environments, synthetic datasets can still be very useful as they can capture a range of situations including edge-case

Table 4. Terrain analysis datasets (Part 1/2) (sensor types: P - proprioceptive, V - monocular camera, S - stereo camera, L - LIDAR).

Name	Dataset Type	Terrain Classes	Sensing Type	Link and Comments
Freiburg Forest (Valada et al., 2016b)	Off-road dataset	trail, grass, vegetation, obstacle, sky, void	PS	http://lifonav.informatik.uni-freiburg.de/datasets.html Contains the following multimodal/spectral images with ground-truth annotations: RGB, Depth, NIR, NRG, NDVI, EVI, and their variants.
RELLIS-3D (Jiang et al., 2020)	Off-road dataset	asphalt, dirt, grass, floor, tree, pole, water, sky, vehicle, object, build, log, person, fence, bush, concrete, barrier, puddle, mud, rubble	PSL	https://unmannedlab.github.io/research/RELLIS-3D a multimodal dataset collected in an off-road environment containing annotations for 13 556 LiDAR scans and 6235 images. It includes RGB camera images, LiDAR point clouds, a pair of stereo images, high-precision GPS measurement, and IMU data.
CityScapes (Cordts et al., 2016)	Urban cityscape	road, sidewalk, parking, rail track, vegetation, terrain	S	https://www.cityscapes-dataset.com/ Stereo video sequences recorded in street scenes
Mapillary Vistas (Neuhold et al., 2017)	Urban cityscape	road, sidewalk, vegetation, snow, sand, water, building, wall, fence, pole, bridge, tunnel	V	https://www.mapillary.com/datasets 25 000 high-resolution images, 152 object categories, variety of weather, season, time of day, camera, and viewpoint. Free version available for noncommercial research.
LabelMe (Russell et al., 2008)	Mixed	road, field, grass, river, plant, sand, rock, desert	V	http://labelme.csail.mit.edu Dataset and online annotation tool to build image databases for computer vision research
BDD 100K (Yu et al., 2018)	Urban cityscapes	road, ground, vegetation, sidewalk, sky, car, street light, rider, building, wall, fence, pole, bridge, tunnel	VP	https://bair.berkeley.edu/blog/2018/05/30/bdd/ Total 120 000 K images (10 K instance segmentation), 40 instance segmentation object classes, variety of weather, season, time of day, and viewpoint. Free version available for noncommercial research.
ApolloScape-Scene (Huang et al., 2018)	Urban cityscapes	road, sidewalk, vegetation, pole, building, wall, fence, bridge, tunnel, overpass	VS	http://apolloscape.auto/scene.html Includes RGB videos with high resolution images and per pixel annotation, survey-grade dense 3D points with semantic segmentation, stereoscopic video, and panoramic images
Marulan (Peynot et al., 2010b)	Special classes	grass	VL+IR	http://sdi.acfr.usyd.edu.au/ The main contribution is the inclusion of multiple elements such as airborne dust, smoke, and rain.

scenarios, generated via simulation, that might be hard to observe in real-world data. Besides, modern computer-generated images can achieve a very high level of realism, in many cases barely distinguishable from the real data. Secondly, the user can adjust various parameters of the model to get different driving scenarios (object arrangement and behavior, weather, lighting, etc.) as well as different (sometimes with subtle difference) types of terrain, which would be much harder to find and capture in real life. Thirdly, and most importantly, considering that the most labor-intensive process in dataset creation is ground truth labeling, there is an undeniable advantage of having immediate and well-segmented ground truth information about different parts of the synthetic world. This

Table 5. Terrain analysis datasets (Part 2/2) (sensor types: P - proprioceptive, V - monocular camera, S - stereo camera, L - LIDAR).

Name	Dataset Type	Terrain Classes	Sensing Type	Link and Comments
Terrain8 (Wu et al., 2019)	Terrain images	asphalt, dirt, grass, floor, gravel, rock, sand, and wood chips	V	http://pan.baidu.com/s/1o7Clk0a each class contains 300 images of the same size (256x256 pixels). Certain images are captured with a camera under different weather conditions, and the others come from Google Image Search. Most of the images in Terrain8 are collected with the camera facing downward to the ground.
ADE20K (Zhou et al., 2019)	Scene dataset	road, pavement, dirt, grass, gravel, sand, water, snow, vegetation, trees	V	https://groups.csail.mit.edu/vision/datasets/ADE20K/ 25 K densely annotated images in different scene categories with corresponding segmentation masks. Object parts are associated with object instances.
CamVid (Brostow et al., 2009)	Urban cityscapes	road, shoulder, lane markings, sidewalk, parking block, tree, vegetation, building, wall, fence, pole, bridge, tunnel, archway	V	http://mi.eng.cam.ac.uk/research/projects/VideoRec/ Over 10 min of high-quality 30 Hz footage is being provided, with corresponding semantically labeled images at 1 Hz and in part, 15 Hz
DARPA LAGR (Procopio, 2007)	Off-road dataset	ground, obstacle, unknown	S	https://mikeprocopio.com/labeledlagrdata.html MATLAB-6 compatible *.mat files with the raw RGB image as well as the disparity information.
Kaggle Vale (Hosseinpour et al., 2019)	Movability based terrain	blue foil, styrofoam, linoleum, cardboard, rubber	P	https://www.kaggle.com/sadhoss/vale-semantic-terrain-segmentation Sensor readouts from quadruped robot: eight potentiometers attached to each joint, Inertial Measurement Unit, a Gyroscope and a Magnetometer.
Audi dataset (Geyer et al., 2020)	Urban cityscapes	pedestrian, car, vegetation	VL	https://www.a2d2.audi/a2d2/en/dataset.html About 40 000 frames of synchronized Vision-LIDAR semantically labeled data categorized into 38 classes with a 360 degree view.
Waymo datasets (Sun et al., 2020)	Urban cityscapes	vehicle, pedestrian, cyclists, signs, and no-label	VL	https://waymo.com/open/data/ About 1950 segments of synchronized 3D 7-DoF bounding boxes of LIDAR and 2D bounding boxes of camera data categorized into 5 classes.
CARLA (Dosovitskiy et al., 2017) & AirSim (Shah et al., 2017)	Simulation engine	simulated terrains	PVSL	http://carla.org/ and https://microsoft.github.io/AirSim The simulation platform supports flexible specification of sensor suites (including LIDARs, multiple cameras, depth sensors, and GPS, among others), environmental conditions, full control of all static and dynamic actors, maps generation, and more. CARLA is provided with integration with ROS via ROS-bridge.

is especially valuable for nonvisual types of data such as point clouds. Such synthetic datasets can give a good indication of the viability of the classifier approach at the early stages of the development.

There are many tools for creating realistic virtual worlds, such as Unity3D ([Unity Technologies, nd](#)), Unreal Engine ([Epic Games, nd](#)), and Blender ([The Blender Foundation, nd](#)). Gazebo ([Open Source Robotics Foundation, nd](#)) is another widely used simulator in robotics development and a major component of the ROS environment. Plugins can expand the capabilities of Gazebo to include dynamic loading of custom models and the use of stereo and infrared cameras, LIDAR, RADAR, GPS, or IMU sensors. Another notable synthetic dataset creation tool is the open source CARLA ([Dosovitskiy et al., 2017](#)) project, based on Unreal Engine, which has the purpose of simplifying dataset creation. Although as with many other datasets CARLA's primary purpose is urban driving, it can be useful in generating weather conditions and driving scenarios on various terrains. AirSim ([Shah et al., 2017](#)), another popular simulation framework also based on Unreal Engine, is a powerful and flexible tool that can use Unreal Marketplace assets to build outdoor environments and thus generate off-road synthetic datasets. Both of these frameworks can generate data from multiple types of sensors including LIDARs, radars, cameras, and IMUs, among others. New sensor types can be added by users if needed.

Despite certain advantages that simulation-only datasets can offer, there is always a tradeoff between the effort to create such datasets and the benefits that they can deliver. Such a tradeoff should be carefully considered before committing to creating a new virtual dataset depending on the project requirements.

9.2. Considerations (Datasets)

As tabulated in Tables 4 and 5, most of the public datasets for autonomous driving focus on urban cityscapes with vision as the primary sensing modality. While they contain some relevant terrain classes for an off-road driving task, the range of these classes is limited. For a robust deep-learning based off-road terrain classification, a greater range of annotated terrain classes is required along with synchronized annotated data from other sensors. However, these datasets provide a good starting point for autonomous navigation modeling, and models trained on these datasets can later be fine-tuned on off-road datasets.

Models trained on datasets such as Cityscapes or Mapillary with additional fine-tuning on RELLIS-3D or Freiburg Forest dataset would provide a good starting point for off-road autonomous navigation. Though such a model will lack the detailed terrain information required for robust navigation in rough terrain, such as identification of mud, gravel, etc., it would be able to identify the common driving hazards and driving surfaces allowing for basic off-road navigation. Then an additional smaller dataset would be required to fine-tune the network. This approach also provides an opportunity for initial model comparison between different networks, in terms of performance and speed, so further training effort can be focused on the most suitable model.

9.3. Open Code

A number of software packages that may potentially be used for terrain classification and analysis are available online, particularly in the learning domain. In this section, we discuss some of the implementations that are available based on the previous investigations within this paper. It is important to note that, in most cases, these techniques cannot be *directly* applied to a specific terrain problem and will likely need specific training and tuning in order to find any potential usefulness in a given terrain analysis domain.

Much of the research in this area is developed in the Python programming language. Currently, some of the most popular backends that are used for learning in Python are Tensorflow ([Abadi et al., 2015](#)) (which also offers some support for other languages) and Pytorch ([Paszke et al., 2019](#)). Most of the learned approaches investigated in this paper use one of these two back-ends to facilitate deep

Table 6. Commonly used platforms for machine-learning algorithms.

Package Name	Comments on Library
Tensorflow (Abadi et al., 2015)	A platform that provides the functionality necessary to deploy Machine Learning systems at every level. It has a wide support base from the community and is widely used in the field. It also has an “ecosystem” of tools that make the production of effective ML models more accessible and more effective. There is also limited support of some other languages, though these language are not yet covered by the Tensorflow stability promises.
Pytorch (Paszke et al., 2019)	PyTorch is another widely used back-end tool for ML solutions. It is implemented in python and provides access to distributed training, an ecosystem of tools to support and facilitate development, and a strong community of users. It is used for a lot of the techniques investigated in this paper.
Keras (Chollet et al., 2015)	Keras is essentially an API that is a wrapper for Tensorflow and Theano. It enables models to be built in more simple, straightforward ways. Keras allows for fast prototyping as well as running seamlessly on CPU and GPU.
Theano (Team et al., 2016)	A python library aimed at allowing users to use GPU-based methods and other optimization techniques to work with large amounts of multidimensional data in an efficient way. It has a focus on tensor expressions and is used in some of the techniques investigated in this paper.
Scikit-Learn (Pedregosa et al., 2011)	Scikit-Learn is a commonly used library that provides tools for data analysis in python. It is available for use under the BSD license and would be used in many of the existing online implementations of the techniques investigated in this paper.

learning, however other libraries and wrappers such as Keras (Chollet et al., 2015) and Theano (Team et al., 2016) are also used in various research. Table 6 provides a brief description of some commonly used software back-ends in tasks that can be applied to terrain analysis.

A list of some relevant vision-based techniques is given in Table 7. Of these, the Deeplab variants (Chen et al., 2018c; Chen et al., 2018b; Chen et al., 2018a) are interesting due to the well maintained code base, their success in semantic image segmentation tasks, and their existing implementation on various datasets such as CityScapes and ADE20K. Although this has not been applied directly to terrain analysis in this repository, Deeplabv3+ may be of interest, as it has achieved good results in semantic segmentation of outdoor or urban scenes, and the existing implementation is well structured, simplifying much of the adaptation process.

Some relevant LIDAR research code bases are also shown in Table 8. While these are all semantic scene segmentation techniques and as such should only be considered potential implementations of future work, they all have excellent online repositories available under various licenses. The ConvPoint repository (Boulch, 2020) has achieved strong results on the large outdoor LIDAR dataset Semantic3D Semantic8 benchmark that segments large outdoor scenes into eight classes, which include vegetation, man-made terrain, buildings, and natural terrain (Hackel et al., 2017). Although this is not directly terrain analysis, semantically segmenting a large outdoor LIDAR scene into these categories may assist with that task.

For exteroceptive sensing, a Matlab package based recurrent neural network classification (Vulpi et al., 2021) is available online ([https://github.com/Ph0bi0/T\\$_\\$_DEEP](https://github.com/Ph0bi0/T$_$_DEEP)).

These tables show some of the existing code-bases online that could be considered to assist research on terrain analysis, although these lists are far from exhaustive, and other considerations such as licensing, technological relevancy, and ease of use would need to be considered.

9.4. Considerations (Open source tools)

As shown in the above tables, many of the techniques investigated in this paper have supporting online code bases that could be considered when beginning research into the terrain analysis task. Most of these methods will likely require some amount of adaptation or rework—specifically those

Table 7. Vision-based methods.

Ref.	Technique	Description
(Shen and Kelly, 2017)	Terrain Classification For Offroad Driving	A repository of this technique is provided, however no licensing information is explicitly attached.
(Valada et al., 2019)	Adaptnet++	An online code base for this technique is available. It claims to need a single GPU with at least 12 GB of memory. Adaptnet++ has been benchmarked against Cityscapes and Freiburg Forest, however it is highly likely that retraining or at least fine-tuning would be required to solve the terrain analysis problem at hand. This implementation uses Tensorflow. It is licensed under the GPL-3.0 License.
(Suryamurthy et al., 2019)	Terrain Segmentation and Roughness Estimation using RGB Data: Path Planning Application on the CENTAURO Robot	There is a code base available for this paper online, however no licensing information is explicitly provided. The code is implemented through a different back end, Caffe (Jia et al., 2014). Adaptation may be required here, as the paper appears to be more focused on small local areas of artificially positioned terrain rather than large scenes outdoors.
(Chen et al., 2018c), (Chen et al., 2018b), (Chen et al., 2018a)	Deeplab Variants	These references are variants on the Deeplab architecture for which there is a model available online under the Tensorflow model garden. There are likely multiple specific implementations available as well, however the main one investigated for this table was the Tensorflow model garden version. It is implemented through Tensorflow and supports a Mobilenet architecture (Sandler et al., 2018),(Howard et al., 2019). This may need to be adapted, retrained, or fine-tuned on a terrain dataset for effective use.
(Romera et al., 2017)	ERFNEt	An implementation of this technique is available online under a noncommercial license. It uses PyTorch and the Cityscapes (Cordts et al., 2016) dataset, meaning some adaptation may be required for off-road terrain analysis.
(Mehta et al., 2018)	ESPNet	The code base for this technique uses PyTorch and is available under the MIT license. It has been shown to run at 9FPS on the Cityscapes Data set on a TX2. Fine-tuning, retraining, or other work would likely be needed to adapt this to off-road terrain analysis.
(Zhao et al., 2018)	ICNet	There is an online implementation of this paper based on the Caffe back-end. It contains pretrained models on Cityscapes. No explicit licensing information is provided.

that show promising results in scene segmentation benchmarks but have not been used explicitly for terrain classification.

Learning-based approaches are fairly consistent in their use of certain standard platforms such as Tensorflow and PyTorch. Many model-based vision approaches also use these or similar packages. These techniques are most promising for terrain analysis as they are performing segmentation. Although a specific implementation should be further investigated before it is selected for use, these tables give an indication of which techniques could be developed and tested more quickly than others, given sufficient fine tuning on prelabeled terrain data.

Table 8. Comments on Code Repositories For LIDAR-based Methods.

Ref.	Technique Name	Comments on code base
(Choy et al., 2019)	Minkowski Engine	A code base was released with the Minkowski Engine Paper. It uses a PyTorch backend and provides an autodifferentiation library for sparse tensors. It was used in the paper to implement a U-Net that functioned on Point Clouds for semantic segmentation. This could be considered for future work, however it is not directly related to terrain analysis. Adaptation necessary may include training on relevant LIDAR data as terrain analysis has not been shown with this model. The repository is online under the MIT license.
(Boulch, 2020)	ConvPoint	Repository that implements the ConvPoint paper. Runs on a PyTorch backend. It is released under a dual license and requires explicit permission for commercial use. Similar to other learned methods, adaptation and alteration may be required to get useful results.
(Liu et al., 2019)	PVCNN (Point Voxel CNN)	Repository implementing PVCNN: Point-Voxel CNN for Efficient 3D Deep Learning. It has been shown to work on an Nvidia Jetson Xavier in a driverless car to perform obstacle detection. While this has not been used explicitly for terrain analysis, it could be considered a promising technique for future work given the hardware used and the demonstrated ability to make it run “live.” The code base is available online under the MIT license.
(Cortinhal et al., 2020)	SalsaNext	A code base was released with the SalsaNext paper, which implements uncertainty-aware semantic segmentation of a full 3D LiDAR point cloud in real time.

10. Synthesis and Conclusions

In this section, we summarize our findings, providing insights and relevant conclusions on the topic of terrain analysis and understanding. First, it is important to point out requirements for success in the task of terrain classification for off-road vehicles and also identify common observations in the literature.

- Methods in the literature often assume that accurate 6-DoF localization ($x, y, z, \text{yaw} + \text{attitude}$) of the vehicle is available at all times, accurately synchronized with all sensor data, unless the method is only classifying each scan/image at a time (e.g., no accumulation of sensory data in a map before classification).
- All multisensor methods require accurate calibration and synchronization between the sensors. This calibration should be both intrinsic and extrinsic. Although this paper does not explore this aspect, it is a crucial requirement and there is a large body of literature with effective calibration methods.
- All learning methods, especially deep learning, require large amounts of (labeled) sensor data, taken in the relevant environments and conditions, and sufficiently capturing the variety of situations and conditions that are significant.
- Most studies in this review conduct the terrain classification *off-line*. When performed onboard the robot, off-road vehicles are usually driving (very) slowly (e.g., at walk pace or similar). Most studies with higher-speed vehicles are on the road in urban environments. They can exploit more structure, making for a “simpler” problem in terms of traversability analysis.

10.1. On Visual-based Methods

More traditional learning-based methods using imaging for terrain classification rely on hand-crafted features and have shown good success in fairly specific applications with a relatively clear bounded domain. These methods generally do not require large datasets for training the classifiers. However,

some of the disadvantages of those methods are (i) the fact that the feature extraction methodology is usually designed for a specific terrain class and may not be easily generalized to a different terrain type; and (ii) the terrain classes exhibit similarity and variations, which are hard to be represented with specific features. Hence, many state-of-the-art methods for terrain classification are built on deep semantic segmentation networks, particularly with the advancements in the last 5 years. The deep networks do not require domain expertise and can automatically learn high-level features from data. Also, the network design can be generic and retrained to classify new terrains. Finally, the network can be extended to a multimodal network that can fuse multiple sensor data. The disadvantage, however, is that a large amount of annotated data is required for training, and the data need to extensively cover the variety that can be encountered in the real world, as those methods are usually not very effective at extrapolating far beyond what is included in the training data. Ensuring that coverage is sufficient is a common challenge. The extension from still images to video can exploit temporal information between frames, which can improve classification accuracy and/or speed. Once again, a drawback of videos is that they tend to expand the labeling effort, with labeling in many frames of the training video. To our knowledge, there is no explicit implementation of video semantic segmentation methods for terrain classification. Nevertheless, these specialized methods for videos are worth exploring and should be investigated.

We also reviewed “expert systems” to detect specific classes of interest such as mud and water as these are particularly challenging. There is minimal research with multiple-class terrain classification that was shown to be successful in detecting water or mud, compared with the dedicated method that was focusing on one (or both) of those specific classes. In terms of pure vision, a water-dedicated binary classifier using deep learning (Wang and Wang, 2019) shows promising results when sufficient labeled data for training are available. To the best of our knowledge, there is no research focusing on traversability analysis through water, but only classifying water as an obstacle. In all reviewed mud detection works, multiple sensors were used for classification. Identifying mud in shadows and in wet weather conditions is extremely challenging and possibly requires a multimodal approach.

10.2. On LIDAR-based Methods

The use of LIDAR data for scene semantic segmentation and terrain analysis has been one of the most popular and powerful strategies in recent works. This is mainly due to the fact that it is capable of capturing accurate metric information, which is very useful especially for robot tasks (e.g., autonomous navigation) and providing a high quality in measurement consistency and high-fidelity information (e.g., a variety of returning waveforms depending on terrain property). It is important to note that LIDAR’s success is generally seen in the analysis of three-dimensional structures in the terrain (tress, rocks, bush, etc.) and for regression of terrain traversability. For classification, in the case of geometrically similar surfaces (e.g., sand, mud, dirt) the usefulness of LIDAR is very limited.

With regard to processing LIDAR data, (e.g., point cloud), 3D convolutional neural networks are gaining popularity among computer vision, machine learning, and robotics communities in either supervised, semi- or weakly supervised, and self-supervised manners. Supervised approaches may be the most popular due to their simplicity and already established work in 2D convolution. However, one of the bottlenecks is analogous to those of 2D CNNs, i.e., difficulty in high-quality annotation data.

10.3. On Alternative Sensors

IR thermal cameras have been used with some success as complementary sources of information to visual cameras, in particular to detect mud, some instances of water, and negative obstacles. They also allow for vision-based perception at nighttime, albeit with its own limitations, but with the advantage of being a passive sensor. In the literature, a combination of passive sensors such as thermal, stereo, and polarization cameras were recommended to detect mud during day and night. It is important to note that IR thermal cameras also robustly see through smoke. Radars have

been used mostly due to the better penetration of their signal, compared with LIDAR and visual cameras. Two main uses can be highlighted here. First, mm-wave radars are very effective at seeing through dense airborne dust, smoke, and fog. FMCW mm-wave radars can be used in a similar way to LIDARs for terrain classification or traversability, however they usually offer a much lower resolution and signal-to-noise ratio, leading to a significant reduction in accuracy compared with LIDAR-based methods. UWB radar, used in combination with LIDAR and/or stereo cameras, has the potential to detect actual obstacles through foliage and vegetation during day and night, in close proximity to the robot, thereby allowing a robot to drive through some level of vegetation with some level of reliability. However, UWB radars are even lower resolution and have lower signal-to-noise ratio compared with FMCW mm-wave radars, and this approach requires extensive scanning or multiple UWB units on the robot.

10.4. On Sensor Fusion

Selecting the right sensors for the fusion in respect to the environmental conditions and the terrain that should be detected is crucial to enable a high performing system. The more complementary sensors with various physical properties are included, the more object properties can be captured and fused. Based on our investigation, some challenging terrain, such as mud and water, is likely to need additional sensory input to be detected robustly. For a reliable field system, sensor fusion is mandatory and should be a core focus in the design of a robust system.

One of the greatest challenges is to determine where to limit the addition of sensors, as more sensors generally cause more annotation effort and computational cost, apart from the device cost. As discussed, LIDARs and cameras, both monocular and stereo, form the bulk of the literature and show the most relevant results. Combining them effectively (which has only been done to a limited extent in the context of terrain analysis) is essential.

10.5. On Open Datasets

Section 9 has shown that the vast majority of the public datasets for autonomous driving contain urban scenes with vision as the primary sensing modality. There are only very limited off-road data, with only a limited number of terrain classes. If robust off-road terrain classification is required, a greater number of annotated terrain classes are necessary, as what is currently openly available is not sufficient. It is also paramount to consider datasets that contain synchronized annotated data from other sensors, particularly LIDAR and imaging. Nonetheless, these online datasets are a relevant starting point in the absence of one's own labeled data, as models trained on these datasets can later be fine-tuned on off-road datasets, or might be considered as a complementary source of data either for additional training or for testing in a different environment.

As examples, models trained on datasets such as Cityscapes or Mapillary with further tuning on the Freiburg Forest or RELIS-3D dataset could potentially provide an initial base for off-road autonomous navigation. Such a model would lack the fine terrain information necessary for robust navigation in rough terrain (e.g., detection of mud, rocks) but it would be able to identify usual driving hazards and driving surfaces, allowing for some level of off-road autonomy. Then, complementary smaller datasets would be necessary to fine-tune the network. This strategy also allows for initial comparisons between networks regarding their performance and speed.

As indicated in Tables 4 and 5, there are no public datasets focusing on off-road navigation on a similar scale to urban cityscapes. RELIS-3D is still the largest one available, so the creation of an even larger scale annotated off-road dataset would certainly assist the community in developing more sophisticated and reliable methods for off-road navigation.

10.6. On Online Software Resources

We have seen in the tables of Section 9 that a number of the techniques discussed in this paper have supporting online code-bases that could be used as a starting point for the terrain analysis

task. Preliminary investigation has shown that most of these methods will require some amount of adaptation or rework—specifically those that show promising results in scene segmentation benchmarks but have not been used explicitly for terrain classification. The fact that they work well in other benchmarks makes them good candidates for further testing.

This review has indicated that learning-based approaches using vision are fairly consistent in their use of certain standard platforms such as Tensorflow and PyTorch. Several model-based vision approaches also use these or similar packages. Although a specific implementation or package needs to be further tuned before it is used, the tables indicate which techniques could be developed and tested more quickly than others, given sufficient fine tuning on one's data. In particular, ESPNet and DeepLabv3+ both have been shown to run semantic image segmentation effectively. The Deeplab code base is excellent and is a part of the Tensorflow model-garden, so it is well structured. It comes with existing pretrained checkpoints that can be fine-tuned, and a Colab notebook, which is useful for speeding up development. Although the default backbone of Deeplab is an Xception network (which can be computationally heavy for onboard usage in some systems), it can be swapped by a Mobilenet backbone, for example, which uses fewer resources.

Acknowledgments

The authors gratefully acknowledge the commitment, support, and funding from our industry partner for this work. T. Peynot, B. Arain, M.G. Minareci, G. Samvedi, M. Milford, and P. Corke acknowledge continued support from the Queensland University of Technology (QUT) through the Centre for Robotics, the support of the Research Engineering Facility (REF) team at QUT for the provision of expertise and research infrastructure in enabling this project, and Ray Johnson and Michael Evans for their support of the project.


ORCID

Paulo V. K. Borges  <https://orcid.org/0000-0001-8137-7245>

Thierry Peynot  <https://orcid.org/0000-0001-8275-6538>

Sisi Liang  <https://orcid.org/0000-0002-6423-2324>

Bilal Arain  <https://orcid.org/0000-0002-2198-2870>

Melih G. Minareci  <https://orcid.org/0000-0001-5135-1220>

Serge Lichman  <https://orcid.org/0000-0001-6200-3360>

Inkyu Sa  <https://orcid.org/0000-0001-5429-0515>

Nicolas Hudson  <https://orcid.org/0000-0002-4931-9659>

Michael Milford  <https://orcid.org/0000-0002-5162-1793>

Peyman Moghadam  <https://orcid.org/0000-0002-8169-3560>

Peter Corke  <https://orcid.org/0000-0001-6650-367X>

References

- Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G. S., Davis, A., Dean, J., Devin, M., Ghemawat, S., Goodfellow, I., Harp, A., Irving, G., Isard, M., Jia, Y., Jozefowicz, R., Kaiser, L., Kudlur, M., Levenberg, J., Mané, D., Monga, R., Moore, S., Murray, D., Olah, C., Schuster, M., Shlens, J., Steiner, B., Sutskever, I., Talwar, K., Tucker, P., Vanhoucke, V., Vasudevan, V., Viégas, F., Vinyals, O., Warden, P., Wattenberg, M., Wicke, M., Yu, Y., & Zheng, X. (2015). TensorFlow: Large-scale machine learning on heterogeneous systems. Software available from tensorflow.org.
- Aeschimann, R. & Borges, P. V. K. (2015). Ground or obstacles? detecting clear paths in vehicle navigation. In *2015 IEEE International Conference on Robotics and Automation (ICRA)*, pages 3927–3934. IEEE.
- Ahmadi, A., Nygaard, T., Kottege, N., Howard, D., & Hudson, N. (2021). Semi-supervised gated recurrent neural networks for robotic terrain classification. *IEEE Robotics and Automation Letters*, 6(2):1848–1855.

- Ahtiainen, J., Peynot, T., Saarinen, J., & Scheduling, S. (2013). Augmenting traversability maps with ultra-wideband radar to enhance obstacle detection in vegetated environments. In *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 5148–5155. IEEE.
- Ahtiainen, J., Peynot, T., Saarinen, J., Scheduling, S., & Visala, A. (2015). Learned ultra-wideband radar sensor model for augmented lidar-based traversability mapping in vegetated environments. In *2015 18th International Conference on Information Fusion (Fusion)*, pages 953–960. IEEE.
- Angelova, A., Helmick, D., & Perona, P. (2007). Fast terrain classification using variable-length representation for autonomous navigation. In *2007 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8. IEEE.
- Badrinarayanan, V., Kendall, A., & Cipolla, R. (2017). Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(12):2481–2495.
- Bai, C., Guo, J., Guo, L., & Song, J. (2019). Deep multi-layer perception based terrain classification for planetary exploration rovers. *Sensors*, 19(14).
- Bajracharya, M., Howard, A., Matthies, L. H., Tang, B., & Turmon, M. (2009). Autonomous off-road navigation with end-to-end learning for the lagr program. *Journal of Field Robotics*, 26(1):3–25.
- Bajracharya, M., Ma, J., Malchano, M., Perkins, A., Rizzi, A. A., & Matthies, L. (2013). High fidelity day/night stereo mapping with vegetation and negative obstacle detection for vision-in-the-loop walking. In *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 3663–3670. IEEE.
- Bajracharya, M., Tang, B., Howard, A., Turmon, M., & Matthies, L. (2008). Learning long-range terrain classification for autonomous navigation. In *2008 IEEE International Conference on Robotics and Automation*, pages 4018–4024. IEEE.
- Bekker, M. G. (1969). Introduction to terrain-vehicle systems. part i: The terrain. part ii: The vehicle. Technical report, MICHIGAN UNIV ANN ARBOR.
- Best, G., Moghadam, P., Kottege, N., & Kleeman, L. (2013). Terrain classification using a hexapod robot. In *Australasian Conference on Robotics and Automation 2013*, pages 1–8. Australian Robotics and Automation Association.
- Bonafous, D., Lacroix, S., & Simeon, T. (2001). Motion generation for a rover on rough terrains. In *Proceedings 2001 IEEE/RSJ International Conference on Intelligent Robots and Systems*, volume 2, pages 784–789. IEEE.
- Borges, P., Zlot, R., Bosse, M., Nuske, S., & Tews, A. (2010). Vision-based localization using an edge map extracted from 3d laser range data. In *2010 IEEE International Conference on Robotics and Automation*, pages 4902–4909. IEEE.
- Borges, P. V. K., Mayer, J., & Izquierdo, E. (2008). A probabilistic model for flood detection in video sequences. In *2008 15th IEEE International Conference on Image Processing*, pages 13–16. IEEE.
- Boulch, A. (2020). Convpoint: Continuous convolutions for point cloud processing. *Computers & Graphics*, 88:24–34.
- Bradley, D. M., Unnikrishnan, R., & Bagnell, J. (2007). Vegetation detection for driving in complex environments. In *IEEE International Conference on Robotics and Automation*. IEEE.
- Brooks, C. A. & Iagnemma, K. (2012). Self-supervised terrain classification for planetary surface exploration rovers. *Journal of Field Robotics*, 29(3):445–468.
- Brostow, G. J., Fauqueur, J., & Cipolla, R. (2009). Semantic object classes in video: A high-definition ground truth database. *Pattern Recognition Letters*, 30(2):88–97.
- Brubaker, K. M., Myers, W. L., Drohan, P. J., Miller, D. A., & Boyer, E. W. (2013). The use of LiDAR terrain data in characterizing surface roughness and microtopography. *Applied and Environmental Soil Science*, 2013, [891534]. <https://doi.org/10.1155/2013/891534>
- Brunner, C., Peynot, T., Vidal-Calleja, T., & Underwood, J. (2013). Selective combination of visual and thermal imaging for resilient localisation in adverse conditions: Day and night, smoke and fire. *Journal of Field Robotics*, 30(4):641–666.
- Chen, L.-C., Collins, M. D., Zhu, Y., Papandreou, G., Zoph, B., Schroff, F., Adam, H., & Shlens, J. (2018a). Searching for efficient multi-scale architectures for dense image prediction. In Bengio, S., Wallach, H. M., Larochelle, H., Grauman, K., Cesa-Bianchi, N., and Garnett, R., editors, *NeurIPS*, pages 8713–8724.
- Chen, L.-C., Papandreou, G., Kokkinos, I., Murphy, K., & Yuille, A. L. (2014). Semantic image segmentation with deep convolutional nets and fully connected crfs. *arXiv:1412.7062*.

- Chen, L.-C., Papandreou, G., Kokkinos, I., Murphy, K., & Yuille, A. L. (2018b). DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs. In *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(4):834–848.
- Chen, L.-C., Papandreou, G., Schroff, F., & Adam, H. (2017b). Rethinking atrous convolution for semantic image segmentation. *arXiv:1706.05587*.
- Chen, L.-C., Zhu, Y., Papandreou, G., Schroff, F., & Adam, H. (2018c). Encoder-decoder with atrous separable convolution for semantic image segmentation. In *2018 Proceedings of the European Conference on Computer Vision (ECCV)*, pages 801–818.
- Chhaniyara, S., Brunskill, C., Yeomans, B., Matthews, M., Saaj, C., Ransom, S., & Richter, L. (2012). Terrain trafficability analysis and soil mechanical property identification for planetary rovers: A survey. *Journal of Terramechanics*, 49(2):115–128.
- Chiodini, S., Torresin, L., Pertile, M., & Debei, S. (2020). Evaluation of 3d cnn semantic mapping for rover navigation. In *2020 IEEE 7th International Workshop on Metrology for AeroSpace (MetroAeroSpace)*, pages 32–36. IEEE.
- Chollet, F. et al. (2015). Keras. <https://keras.io>.
- Choy, C. B., Gwak, J., & Savarese, S. (2019). 4d spatio-temporal convnets: Minkowski convolutional neural networks. *CoRR*, abs/1904.08755.
- Cordts, M., Omran, M., Ramos, S., Rehfeld, T., Enzweiler, M., Benenson, R., Franke, U., Roth, S., & Schiele, B. (2016). The cityscapes dataset for semantic urban scene understanding. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3213–3223. IEEE.
- Corke, P., Paul, R., Churchill, W., & Newman, P. (2013). Dealing with shadows: Capturing intrinsic scene appearance for image-based outdoor localisation. In *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 2085–2092. IEEE.
- Cortinhal, T., Tzelepis, G., & Aksoy, E. E. (2020). Salsanext: Fast, uncertainty-aware semantic segmentation of lidar point clouds. In *International Symposium on Visual Computing*, pages 207–222. Springer.
- Coyle, E. (2010). *Fundamentals and Methods of Terrain Classification Using Proprioceptive Sensors*. Ph.D. thesis, Florida State University.
- Dabbiru, L., Sharma, S., Goodin, C., Ozier, S., Hudson, C., Carruth, D., Doude, M., Mason, G., & Ball, J. (2021). Traversability mapping in off-road environment using semantic segmentation. In *Autonomous Systems: Sensors, Processing, and Security for Vehicles and Infrastructure 2021*, volume 11748, page 117480C. International Society for Optics and Photonics.
- Dai, A., Chang, A. X., Savva, M., Halber, M., Funkhouser, T., & Nießner, M. (2017). Scannet: Richly-annotated 3d reconstructions of indoor scenes. In *Proceedings of Computer Vision and Pattern Recognition (CVPR)*. IEEE.
- Diaz-Cabrera, M., Cerri, P., & Medici, P. (2015). Robust real-time traffic light detection and distance estimation using a single camera. *Expert Systems with Applications*, 42(8):3911–3923.
- Dosovitskiy, A., Fischer, P., Ilg, E., Hausser, P., Hazirbas, C., Golkov, V., Van Der Smagt, P., Cremers, D., & Brox, T. (2015). FlowNet: Learning optical flow with convolutional networks. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2758–2766. IEEE.
- Dosovitskiy, A., Ros, G., Codevilla, F., Lopez, A., and Koltun, V. (2017). CARLA: An open urban driving simulator. In *Proceedings of the 1st Annual Conference on Robot Learning*, pages 1–16.
- Dubbelman, G., van der Mark, W., van den Heuvel, J. C., & Groen, F. C. (2007). Obstacle detection during day and night conditions using stereo vision. In *2007 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 109–116. IEEE.
- Dumond, D. A., Ray, L. E., & Trautmann, E. (2009). Evaluation of terrain parameter estimation using a stochastic terrain model. In Gerhart, G. R., Gage, D. W., & Shoemaker, C. M., editors, *Unmanned Systems Technology XI*, volume 7332, pages 396–405. International Society for Optics and Photonics, SPIE.
- DuPont, E. M., Collins, E., Coyle, E. J., & Roberts, R. G. (2008a). Terrain classification using vibration sensors: theory and methods. *New Research on Mobile Robotics*, pages 1–41.
- DuPont, E. M., Moore, C. A., Collins, E. G., & Coyle, E. (2008b). Frequency response method for terrain classification in autonomous ground vehicles. *Autonomous Robots*, 24:337–347.
- Epic Games (n.d.). Unreal engine, cross-platform game engine.
- Fan, D. D., Otsu, K., Kubo, Y., Dixit, A., Burdick, J., & Agha-Mohammadi, A.-A. (2021). Step: Stochastic traversability evaluation and planning for safe off-road navigation. In *Robotics: Science and Systems*. RSS Foundation, pp. 1–21.

- Feng, D., Haase-Schütz, C., Rosenbaum, L., Hertlein, H., Glaeser, C., Timm, F., Wiesbeck, W., and Dietmayer, K. (2020). Deep multi-modal object detection and semantic segmentation for autonomous driving: Datasets, methods, and challenges. *IEEE Transactions on Intelligent Transportation Systems*, 22(3):1341–1360.
- Filgueira, A., González-Jorge, H., Lagüela, S., Díaz-Vilariño, L., & Arias, P. (2017). Quantifying the influence of rain in lidar performance. *Measurement*, 95:143–148.
- Filitchkin, P. & Byl, K. (2012). Feature-based terrain classification for littledog. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1387–1392. IEEE.
- Gadde, R., Jampani, V., & Gehler, P. V. (2017). Semantic video cnns through representation warping. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 4453–4462. IEEE.
- Geiger, A., Lenz, P., & Urtasun, R. (2012). Are we ready for autonomous driving? the kitti vision benchmark suite. In *Conference on Computer Vision and Pattern Recognition (CVPR)*.
- Gennery, D. B. (1999). Traversability analysis and path planning for a planetary rover. *Autonomous Robots*, 6(2):131–146.
- Gerardo-Castro, M. P., Peynot, T., Ramos, F., & Fitch, R. (2014). Robust multiple-sensing-modality data fusion using gaussian process implicit surfaces. In *IEEE International Conference on Information Fusion*, Salamanca, Spain. IEEE.
- Geyer, J., Kassahun, Y., Mahmudi, M., Ricou, X., Durgesh, R., Chung, A. S., Hauswald, L., Pham, V. H., Mühlegg, M., Dorn, S., Fernandez, T., Jänicke, M., Mirashi, S., Savani, C., Sturm, M., Vorobiov, O., Oelker, M., Garreis, S., & Schubert, P. (2020). A2D2: Audi Autonomous Driving Dataset.
- Goldberg, S. B., Maimone, M. W., & Matthies, L. (2002). Stereo vision and rover navigation software for planetary exploration. In *Proceedings, IEEE Aerospace Conference*, volume 5, page 5. IEEE. doi: 10.1109/AERO.2002.1035370.
- González, R. & Iagnemma, K. (2018). Deepterrmechanics: Terrain classification and slip estimation for ground robots via deep learning. *ArXiv*, abs/1806.07379.
- Goodin, C., Dabbiru, L., Hudson, C., Mason, G., Carruth, D., & Doude, M. (2021). Fast terrain traversability estimation with terrestrial lidar in off-road autonomous navigation. In *Unmanned Systems Technology XXIII*, volume 11758, page 1175800. International Society for Optics and Photonics.
- Grigorescu, S., Trasnea, B., Cocias, T., & Macesanu, G. (2020). A survey of deep learning techniques for autonomous driving. *Journal of Field Robotics*, 37(3):362–386.
- Guastella, D. C., & Muscato, G. (2021). Learning-based methods of perception and navigation for ground vehicles in unstructured environments: a review. *Sensors*, 21(1):73.
- Guerrero, J. A., Jaud, M., Lenain, R., Rouveure, R., & Faure, P. (2015). Towards LIDAR-RADAR based terrain mapping for traversability analysis. In *2015 IEEE International Workshop on Advanced Robotics and its Social Impacts (ARSO)*.
- Hackel, T., Savinov, N., Ladicky, L., Wegner, J. D., Schindler, K., & Pollefeys, M. (2017). SE-MANTIC3D.NET: A new large-scale point cloud classification benchmark. In *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, volume IV-1-W1, pages 91–98.
- Hamner, B., Singh, S., Roth, S., & Takahashi, T. (2008). An efficient system for combined route traversal and collision avoidance. *Autonomous Robots*, 24(4):365–385.
- Han, X., Nguyen, C., You, S., & Lu, J. (2018). Single image water hazard detection using fcn with reflection attention units. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 105–120.
- Hang, W., Baozhen, L., Weihua, S., Zihao, C., Wenchang, Z., Xudong, R., & Jinggong, S. (2017). Optimum pipeline for visual terrain classification using improved bag of visual words and fusion methods. *Journal of Sensors*, volume 2017, pages 1–25. <https://doi.org/10.1155/2017/851394>.
- He, K., Zhang, X., Ren, S., & Sun, J. (2016a). Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 770–778. IEEE.
- He, K., Zhang, X., Ren, S., & Sun, J. (2016b). Identity mappings in deep residual networks. In *European Conference on Computer Vision*, pages 630–645. Springer.
- Heckman, N., Lalonde, J.-F., Vandapel, N., & Hebert, M. (2007). Potential negative obstacle detection by occlusion labeling. In *IEEE International Conference on Intelligent Robots and Systems*, pages 2168–2173.
- Hirose, N., Sadeghian, A., Vázquez, M., Goebel, P., & Savarese, S. (2018). Gonet: A semi-supervised deep learning approach for traversability estimation. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 3044–3051. IEEE.

- Ho, K., Peynot, T., & Sukkarieh, S. (2013). A near-to-far non-parametric learning approach for estimating traversability in deformable terrain. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*.
- Ho, K., Peynot, T., & Sukkarieh, S. (2016). Non-parametric traversability estimation in partially occluded and deformable terrain. *Journal of Field Robotics*, 33(8).
- Hosseinpoor, S., Mantelli, M., & Pittol, D. K. (2019). Vale - semantic terrain segmentation.
- Howard, A., Sandler, M., Chu, G., Chen, L.-C., Chen, B., Tan, M., Wang, W., Zhu, Y., Pang, R., Vasudevan, V., Le, Q. V., & Adam, H. (2019). Searching for mobilenetv3. In *ICCV*.
- Howard, A., Turmon, M., Matthies, L., Tang, B., Angelova, A., & Mjolsness, E. (2006). Towards learned traversability for robot navigation: From underfoot to the far field. *Journal of Field Robotics*, 23(11-12):1005–1017.
- Hu, J.-w., Zheng, B.-y., Wang, C., Zhao, C.-h., Hou, X.-l., Pan, Q., & Xu, Z. (2020). A survey on multi-sensor fusion based obstacle detection for intelligent ground vehicles in off-road environments. *Frontiers of Information Technology & Electronic Engineering*, 21:675–692.
- Huang, X., Cheng, X., Geng, Q., Cao, B., Zhou, D., Wang, P., Lin, Y., & Yang, R. (2018). The apollo-scape dataset for autonomous driving. *arXiv:1803.06184*.
- Iagnemma, K., Kang, S., Shibly, H., & Dubowsky, S. (2004). Online terrain parameter estimation for wheeled mobile robots with application to planetary rovers. *Transactions on Robotics*, 20(5):921–927.
- Iagnemma, K. D., & Dubowsky, S. (2002). Terrain estimation for high-speed rough-terrain autonomous vehicle navigation. In Gerhart, G. R., Shoemaker, C. M., & Gage, D. W., editors, *Unmanned Ground Vehicle Technology IV*, volume 4715, pages 256–266. International Society for Optics and Photonics, SPIE.
- Ijaz, M., Ghassemlooy, Z., Pesek, J., Fiser, O., Le Minh, H., & Bentley, E. (2013). Modeling of fog and smoke attenuation in free space optical communications link under controlled laboratory conditions. *Journal of Lightwave Technology*, 31(11):1720–1726.
- Ingram, B., Schmidt, E., Gilbertson, S., & Wiles, N. (2020). Wiper timing and geometry to minimize sensor occlusion. US Patent 10,589,726.
- Ishigami, G., Miwa, A., Nagatani, K., & Yoshida, K. (2006). Terramechanics-based analysis on slope traversability for a planetary exploration rover. In *Proceedings of the International Symposium on Space Technology and Science*, volume 25, page 1025.
- Ishigami, G., Miwa, A., Nagatani, K., & Yoshida, K. (2007). Terramechanics-based model for steering maneuver of planetary exploration rovers on loose soil. *Journal of Field Robotics*, 24(3):233–250.
- Iwashita, Y., Nakashima, K., Stoica, A., & Kurazume, R. (2019). TU-Net and TDeepLab: Deep Learning-Based Terrain Classification Robust to Illumination Changes, Combining Visible and Thermal Imagery. In *2019 IEEE Conference on Multimedia Information Processing and Retrieval (MIPR)*, pages 280–285.
- Jaderberg, M., Simonyan, K., Zisserman, A., & Kavukcuoglu, K. (2015). Spatial transformer networks. *CoRR*, abs/1506.02025.
- Jia, Y., Shelhamer, E., Donahue, J., Karayev, S., Long, J., Girshick, R., Guadarrama, S., & Darrell, T. (2014). Caffe: Convolutional architecture for fast feature embedding. *arXiv:1408.5093*.
- Jiang, P., Osteen, P., Wigness, M., & Saripalli, S. (2020). Rellis-3d dataset: Data, benchmarks and analysis. *arXiv:2011.12954*.
- Kahn, G., Abbeel, P., & Levine, S. (2021). Badgr: An autonomous self-supervised learning-based navigation system. *IEEE Robotics and Automation Letters*, 6(2):1312–1319.
- Kim, D.-K., Maturana, D., Uenoyama, M., & Scherer, S. (2018). Season-invariant semantic segmentation with a deep multimodal network. In *Field and Service Robotics*, pages 255–270. Springer.
- Kim, H., Leutenegger, S., & Davison, A. J. (2016). Real-time 3d reconstruction and 6-dof tracking with an event camera. In *European Conference on Computer Vision*, pages 349–364. Springer.
- Korthals, T., Kragh, M., Christiansen, P., Karstoft, H., Jørgensen, R. N., & Rückert, U. (2018). Multi-modal detection and mapping of static and dynamic obstacles in agriculture for process evaluation. *Frontiers in Robotics and AI*, 5:28.
- Krebs, A., Pradalier, C., & Siegwart, R. (2010). Adaptive rover behaviour based on online empirical evaluation: Rover-terrain interaction and near-to-far learning. *Journal of Field Robotics*, 27(2):158–18023.
- Krüsi, P., Furgale, P., Bosse, M., & Siegwart, R. (2017). Driving on point clouds: Motion planning, trajectory optimization, and terrain assessment in generic nonplanar environments. *Journal of Field Robotics*, 34(5):940–984.

- Krüsi, P., Furgale, P., Bosse, M., *et al.* (2017). Driving on point clouds: Motion planning, trajectory optimization, and terrain assessment in generic nonplanar environments. *Journal of Field Robotics*, 34(5):940–984
- Kuthirummal, S., Das, A., & Samarasekera, S. (2011). A graph traversal based algorithm for obstacle detection using lidar or stereo. In *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 3874–3880. IEEE.
- Lalonde, J.-F., Vandapel, N., Huber, D. F., & Hebert, M. (2006). Natural terrain classification using three-dimensional lidar data for ground robot mobility. *Journal of Field Robotics*, 23(10): 839–861.
- Lang, A. H., Vora, S., Caesar, H., Zhou, L., Yang, J., & Beijbom, O. (2019). Pointpillars: Fast encoders for object detection from point clouds. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE.
- Langer, D., Rosenblatt, J., & Hebert, M. (1994). A behavior-based system for off-road navigation. *IEEE Transactions on Robotics and Automation*, 10(6):776–783.
- Larson, J., & Trivedi, M. (2011). Lidar based off-road negative obstacle detection and analysis. *Conference Record—IEEE Conference on Intelligent Transportation Systems*. IEEE.
- Lee, S.-Y., & Kwak, D.-M. (2011). A terrain classification method for ugv autonomous navigation based on surf. In *2011 8th International Conference on Ubiquitous Robots and Ambient Intelligence (URAI)*, pages 303–306. IEEE.
- Li, J., Nguyen, C., & You, S. (2019). Temporal 3D fully connected network for water-hazard detection. *2019 Digital Image Computing: Techniques and Applications, DICTA 2019*, pages 1–5.
- Li, Y., Ma, L., Zhong, Z., Liu, F., Chapman, M. A., Cao, D., & Li, J. (2020). Deep learning for lidar point clouds in autonomous driving: a review. *IEEE Transactions on Neural Networks and Learning Systems*, volume 32, pages 3412–3432. doi: 10.1109/TNNLS.2020.3015992.
- Li, Y., Shi, J., & Lin, D. (2018). Low-latency video semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5997–6005.
- Libby, J., & Stentz, A. (2012). Using sound to classify vehicle-terrain interactions in outdoor environments. *2012 IEEE International Conference on Robotics and Automation*, pages 3559–3566.
- Liu, Z., Tang, H., Lin, Y., & Han, S. (2019). Point-voxel CNN for efficient 3d deep learning. *CoRR*, abs/1907.03739.
- Liyanage, D. C., Hudjakov, R., & Tamre, M. (2020). Hyperspectral imaging methods improve rgb image semantic segmentation of unstructured terrains. In *2020 International Conference Mechatronic Systems and Materials (MSM)*, pages 1–5. IEEE.
- Long, J., Shelhamer, E., & Darrell, T. (2015). Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3431–3440. IEEE.
- Ma, N., Peng, Y., Wang, S., & Leong, P. H. (2018). An unsupervised deep hyperspectral anomaly detector. *Sensors*, 18(3):693.
- MacDonald, H., Waite, W., & Demarcke, J. (1981). Use of seasat satellite radar imagery for the detection of standing water beneath forest vegetation. In *Rainbow 80; Fall Technical Meeting*.
- Maddern, W., Stewart, A., McManus, C., Upcroft, B., Churchill, W., & Newman, P. (2014). Illumination invariant imaging: Applications in robust vision-based localisation, mapping and classification for autonomous vehicles. In *Proceedings of the Visual Place Recognition in Changing Environments Workshop, IEEE International Conference on Robotics and Automation (ICRA), Hong Kong, China*, volume 2, page 3. IEEE.
- Mahasseni, B., Todorovic, S., & Fern, A. (2017). Budget-aware deep semantic video segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1029–1038. IEEE.
- Mallet, C., Chehata, N., & Bailly, J.-S. (2016). Airborne LiDAR Data Processing. *Optical Remote Sensing of Land Surface: Techniques and Methods*, pages 249–297. <https://doi.org/10.1016/B978-1-78548-102-4.50006-5>.
- Mallet, C., & David, N. (2016). Digital Terrain Models Derived from Airborne LiDAR Data. *Optical Remote Sensing of Land Surface: Techniques and Methods*, pages 299–319. <https://doi.org/10.1016/B978-1-78548-102-4.50007-7>.
- Manduchi, R., Castano, A., Talukder, A., & Matthies, L. (2005). Obstacle detection and terrain classification for autonomous off-road navigation. *Autonomous Robots*, 18(1):81–102.

- Martin, S. C. (2018). Proprioceptive sensing of traversability for long-term navigation of robots. Ph.D. thesis, Queensland University of Technology.
- Martínez, J. L., Morán, M., Morales, J., Robles, A., & Sánchez, M. (2020). Supervised learning of natural-terrain traversability with synthetic 3d laser scans. *Applied Sciences*, 10(3):1140.
- Matthies, L., Maimone, M., Johnson, A., Cheng, Y., Willson, R., Villalpando, C., Goldberg, S., Huertas, A., Stein, A., & Angelova, A. (2007). Computer vision on mars. *International Journal of Computer Vision*, 75(1):67–92.
- Matthies, L., & Rankin, A. (2003). Negative obstacle detection by thermal signature. In *Proceedings 2003 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2003) (Cat. No. 03CH37453)*, volume 1, pages 906–913. IEEE.
- Matthies, L. H., Bellutta, P., & McHenry, M. (2003). Detecting water hazards for autonomous off-road navigation. In *Unmanned Ground Vehicle Technology V*, volume 5083, pages 231–242. International Society for Optics and Photonics.
- Maturana, D., Chou, P.-W., Uenoyama, M., & Scherer, S. (2018). Real-time semantic mapping for autonomous off-road navigation. In *Field and Service Robotics*, pages 335–350. Springer.
- McDaniel, M. W., Nishihata, T., Brooks, C. A., Salesses, P., & Iagnemma, K. (2012). Terrain classification and identification of tree stems using ground-based LiDAR. *Journal of Field Robotics*, 29(6):891–910.
- Mehta, S., Rastegari, M., Caspi, A., Shapiro, L., & Hajishirzi, H. (2018). Espnet: Efficient spatial pyramid of dilated convolutions for semantic segmentation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 552–568.
- Mei, M., Chang, J., Li, Y., Li, Z., Li, X., & Lv, W. (2019). Comparative study of different methods in vibration-based terrain classification for wheeled robots with shock absorbers. *Sensors*, 19(5).
- Milella, A., Nielsen, M., & Reina, G. (2017). Sensing in the visible spectrum and beyond for terrain estimation in precision agriculture. *Advances in Animal Biosciences*, 8(2):423–429.
- Milella, A., Reina, G., & Nielsen, M. (2019). A multi-sensor robotic platform for ground mapping and estimation beyond the visible spectrum. *Precision Agriculture*, 20(2):423–444.
- Milella, A., Reina, G., Underwood, J., & Douillard, B. (2014). Visual ground segmentation by radar supervision. *Robotics and Autonomous Systems*, 62(5):696–706.
- Milford, M. J., & Wyeth, G. F. (2012). Seqslam: Visual route-based navigation for sunny summer days and stormy winter nights. In *2012 IEEE International Conference on Robotics and Automation*, pages 1643–1649. IEEE.
- Milioto, A., Vizzo, I., Behley, J., & Stachniss, C. (2019). Rangenet++: Fast and accurate lidar semantic segmentation. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE.
- Mirza, M., & Osindero, S. (2014). Conditional generative adversarial nets. *arXiv:1411.1784*.
- Mitchell, T. M. *et al.* (1997). *Machine learning*. McGraw-Hill, New York.
- Moghadam, P., & Wijesoma, W. S. (2009). Online, self-supervised vision-based terrain classification in unstructured environments. In *2009 IEEE International Conference on Systems, Man and Cybernetics*, pages 3100–3105. IEEE.
- Molino, V., Madhavan, R., Messina, E., Downs, A., Balakirsky, S., & Jacoff, A. (2007). Traversability metrics for rough terrain applied to repeatable test methods. In *2007 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1787–1794. IEEE.
- Morton, R. D., & Olson, E. (2011). Positive and negative obstacle detection using the hld classifier. In *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1579–1584. IEEE.
- Mount, J., & Milford, M. (2016). 2d visual place recognition for domestic service robots at night. In *2016 IEEE International Conference on Robotics and Automation (ICRA)*, pages 4822–4829. IEEE.
- Neuhold, G., Ollmann, T., Rota Bulò, S., & Kotschieder, P. (2017). The mapillary vistas dataset for semantic understanding of street scenes. In *International Conference on Computer Vision (ICCV)*.
- Nguyen, A., Nguyen, N., Tran, K., Tjiputra, E., & Tran, Q. D. (2020). Autonomous navigation in complex environments with deep multimodal fusion network. In *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 5824–5830. IEEE.
- Nguyen, C. V., Milford, M., & Mahony, R. (2017). 3D tracking of water hazards with polarized stereo cameras. *Proceedings—IEEE International Conference on Robotics and Automation*, pages 5251–5257. IEEE.
- Ni, J., Chen, Y., Chen, Y., Zhu, J., Ali, D., & Cao, W. (2020). A survey on theories and applications for self-driving cars based on deep learning methods. *Applied Sciences*, 10(8):2749.

- Nilsson, D., & Sminchisescu, C. (2018). Semantic video segmentation by gated recurrent flow propagation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6819–6828. IEEE.
- Ojeda, L., Borenstein, J., Witus, G., & Karlson, R. (2006). Terrain characterization and classification with a mobile robot. *Journal of Field Robotics*, 23:103–122.
- Oniga, F., & Nedeveschi, S. (2009). Processing dense stereo data using elevation maps: Road surface, traffic isle, and obstacle detection. *IEEE Transactions on Vehicular Technology*, 59(3):1172–1182.
- Open Source Robotics Foundation (n.d.). Gazebo, robot simulation made easy.
- Owens, K., & Matthies, L. (1999). Passive night vision sensor comparison for unmanned ground vehicle stereo vision navigation. In *Proceedings IEEE Workshop on Computer Vision Beyond the Visible Spectrum: Methods and Applications (CVBVS'99)*, pages 59–68. IEEE.
- Palazzo, S., Guastella, D. C., Cantelli, L., Spadaro, P., Rundo, F., Muscato, G., Giordano, D., & Spampinato, C. (2020). Domain adaptation for outdoor robot traversability estimation from rgb data with safety-preserving loss. In *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 10014–10021. IEEE.
- Pandian, A. (2008). Robot navigation using stereo vision and polarization imaging. *Master's thesis, Institut Universitaire de Technologie IUT Le Creusot, Universite de Bourgogne*.
- Papadakis, P. (2013). Terrain traversability analysis methods for unmanned ground vehicles: A survey. *Engineering Applications of Artificial Intelligence*, 26(4):1373–1385.
- Paszke, A., Chaurasia, A., Kim, S., & Culurciello, E. (2016). Enet: A deep neural network architecture for real-time semantic segmentation. *arXiv:1606.02147*.
- Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Kopf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., Bai, J., & Chintala, S. (2019). Pytorch: An imperative style, high-performance deep learning library. In Wallach, H., Larochelle, H., Beygelzimer, A., d'Alché-Buc, F., Fox, E., & Garnett, R., editors, *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc.
- Paul, M., Mayer, C., Gool, L. V., & Timofte, R. (2020). Efficient video semantic segmentation with labels propagation and refinement. In *The IEEE Winter Conference on Applications of Computer Vision*, pages 2873–2882. IEEE.
- Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., & Duchesnay, E. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.
- Peynot, T., & Kassir, A. (2010). Laser-camera data discrepancies and reliable perception in outdoor robotics. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Taipei, Taiwan. IEEE.
- Peynot, T., Monteiro, S., Kelly, A., & Devy, M. (2015). Editorial: Special issue on alternative sensing techniques for robot perception. *Journal of Field Robotics*, 32(1).
- Peynot, T., Scheduling, S., & Terho, S. (2010b). The Marulan Data Sets: Multi-Sensor Perception in Natural Environment with Challenging Conditions. *The International Journal of Robotics Research (IJRR)*, 29(13):1602–1607.
- Peynot, T., Underwood, J., & Scheduling, S. (2009). Towards reliable perception for unmanned ground vehicles in challenging conditions. In *2009 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1170–1176. IEEE.
- Procopio, M. J. (2007). Hand-labeled DARPA LAGR datasets. Available at <http://www.mikeprocopio.com/labeledlagrdata.html>.
- Qi, C. R., Su, H., Mo, K., & Guibas, L. J. (2016). Pointnet: Deep learning on point sets for 3d classification and segmentation. *CoRR*, abs/1612.00593.
- Qi, C. R., Yi, L., Su, H., & Guibas, L. J. (2017). Pointnet++: Deep hierarchical feature learning on point sets in a metric space. In Guyon, I., Luxburg, U. V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., & Garnett, R., editors, *Advances in Neural Information Processing Systems 30*, pages 5099–5108. Curran Associates, Inc.
- Rankin, A., Huertas, A., Matthies, L., Bajracharya, M., Assad, C., Brennan, S., Bellutta, P., & Sherwin, G. W. (2011). Unmanned ground vehicle perception using thermal infrared cameras. In *Unmanned Systems Technology XIII*, volume 8045, page 804503. International Society for Optics and Photonics.

- Rankin, A. L., Huertas, A., & Matthies, L. H. (2007). *Night-time negative obstacle detection for off-road autonomous navigation*, volume 6561 of *Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series*, page 656103. SPIE.
- Rankin, A. L., & Matthies, L. H. (2006). Daytime water detection and localization for unmanned ground vehicle autonomous navigation. *Proceedings of the 25th Army Science Conference*.
- Rankin, A. L., & Matthies, L. H. (2008). Daytime mud detection for unmanned ground vehicle autonomous navigation. Technical report, California Institute of Technology Pasadena Jet Propulsion Laboratory.
- Rankin, A. L., & Matthies, L. H. (2010). Passive sensor evaluation for unmanned ground vehicle mud detection. *Journal of Field Robotics*, 27(4):473–490.
- Redmon, J. (2013–2016). Darknet: Open source neural networks in c. <http://pjreddie.com/darknet/>.
- Reina, G., & Galati, R. (2016). Slip-based terrain estimation with a skid-steer vehicle. *Vehicle System Dynamics*, 54(10):1384–1404.
- Reina, G., Galati, R., & Milella, A. (2018). All-terrain estimation for mobile robots in precision agriculture. *2018 IEEE International Conference on Industrial Technology (ICIT)*, pages 63–68. IEEE.
- Reina, G., Leanza, A., Milella, A., & Arcangelo, M. (2020). Mind the ground: A power spectral density-based estimator for all-terrain rovers. *Measurement*, 151.
- Reina, G., Milella, A., & Galati, R. (2017). Terrain assessment for precision agriculture using vehicle dynamic modelling. *Biosystems Engineering*, 162:124–139.
- Reina, G., Milella, A., & Underwood, J. (2012). Self-learning classification of radar features for scene understanding. *Robotics and Autonomous Systems*, 60(11):1377–1388. Towards Autonomous Robotic Systems 2011.
- Reina, G., Underwood, J., Brooker, G., & Durrant-Whyte, H. (2011). Radar-based perception for autonomous outdoor vehicles. *Journal of Field Robotics*, 28(6):894–913.
- Romera, E., Alvarez, J. M., Bergasa, L. M., & Arroyo, R. (2017). Erfnet: Efficient residual factorized convnet for real-time semantic segmentation. *IEEE Transactions on Intelligent Transportation Systems*, 19(1):263–272. IEEE.
- Ronneberger, O., Fischer, P., & Brox, T. (2015). U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical Image Computing and Computer-assisted Intervention*, pages 234–241. Springer.
- Rothrock, B., Kennedy, R., Cunningham, C., Papon, J., Heverly, M., & Ono, M. (2016). SPOC: Deep learning-based terrain classification for Mars Rover missions. In *AIAA SPACE 2016*, AIAA SPACE Forum. American Institute of Aeronautics and Astronautics.
- Ruetz, F., Hernández, E., Pfeiffer, M., Oleynikova, H., Cox, M., Lowe, T., & Borges, P. (2019). Ovp mesh: 3d free-space representation for local ground vehicle navigation. In *2019 International Conference on Robotics and Automation (ICRA)*, pages 8648–8654. IEEE.
- Russell, B., Torralba, A., Murphy, K., & Freeman, W. (2008). Labelme: A database and web-based tool for image annotation. *International Journal of Computer Vision*, 77.
- Sadhukhan, D. (2004). Autonomous ground vehicle terrain classification using internal sensors. In *Master's thesis, Dept. Mech. Eng., Florida State University, Tallahassee, FL, USA*.
- Sancho-Pradel, D. L., & Gao, Y. (2010). A survey on terrain assessment techniques for autonomous operation of planetary robots. *JBIS-Journal of the British Interplanetary Society*, 63(5-6):206–217.
- Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., & Chen, L.-C. (2018). Mobilenetv2: Inverted residuals and linear bottlenecks. In *CVPR*.
- Santana, P., Mendonça, R., & Barata, J. (2012). Water detection with segmentation guided dynamic texture recognition. *2012 IEEE International Conference on Robotics and Biomimetics, ROBIO 2012 - Conference Digest*, volume 1 (December), pages 1836–1841. IEEE.
- Schilling, F., Chen, X., Folkesson, J., & Jensfelt, P. (2017). Geometric and visual terrain classification for autonomous mobile navigation. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE-RSJ.
- Schwartz, J. T., & Sharir, M. (1987). Identification of partially obscured objects in two and three dimensions by matching noisy characteristic curves. *The International Journal of Robotics Research*, 6(2): 29–44.
- Shah, S., Dey, D., Lovett, C., & Kapoor, A. (2017). Airsim: High-fidelity visual and physical simulation for autonomous vehicles. In *Field and Service Robotics*. https://doi.org/10.1007/978-3-319-67361-5_40.
- Shan, T., Wang, J., Englot, B., & Doherty, K. (2018). Bayesian generalized kernel inference for terrain traversability mapping. In *Conference on Robot Learning*, pages 829–838.

- Shelhamer, E., Rakelly, K., Hoffman, J., & Darrell, T. (2016). Clockwork convnets for video semantic segmentation. In *European Conference on Computer Vision*, pages 852–868. Springer.
- Shen, K., & Kelly, M. (2017). Terrain classification for off-road driving cs-229 final report. Oxford University.
- Shinzato, P. Y., Wolf, D. F., & Stiller, C. (2014). Road terrain detection: Avoiding common obstacle detection assumptions using sensor fusion. In *2014 IEEE Intelligent Vehicles Symposium Proceedings*, pages 687–692. IEEE.
- Simonyan, K., & Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. *arXiv:1409.1556*.
- Sofman, B., Lin, E., Bagnell, J. A., Cole, J., Vandapel, N., & Stentz, A. (2006). Improving robot navigation through self-supervised online learning. *Journal of Field Robotics*, 23(11-12):1059–1075.
- Stanislas, L., Nubert, J., Dugas, D., Nitsch, J., Suenderhauf, N., Siegwart, R., Cadena, C., & Peynot, T. (2019). Airborne particle classification in lidar point clouds using deep learning. In *12th Conference on Field and Service Robotics (FSR)*, Tokyo, Japan.
- Stella, E., Negahdaripour, S., Ceglarek, D., & Möller, C. (2021). Multimodal sensing and artificial intelligence: Technologies and applications ii. In *Proceedings of SPIE*, volume 11785, pages 1178501–1. SPIE.
- Stentz, A., Bares, J., Pilarski, T., & Stager, D. (2007). The crusher system for autonomous navigation. *AUVSs Unmanned Systems North America*, 3.
- Suger, B., Steder, B., & Burgard, W. (2015). Traversability analysis for mobile robots in outdoor environments: A semi-supervised learning approach based on 3d-lidar data. In *2015 IEEE International Conference on Robotics and Automation (ICRA)*, pages 3941–3946. IEEE.
- Sun, P., Kretschmar, H., Dotiwala, X., Chouard, A., Patnaik, V., Tsui, P., Guo, J., Zhou, Y., Chai, Y., Caine, B., *et al.* (2020). Scalability in perception for autonomous driving: Waymo open dataset. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2446–2454. IEEE.
- Suryamurthy, V., Raghavan, V. S., Laurenzi, A., Tsagarakis, N. G., & Kanoulas, D. (2019). Terrain segmentation and roughness estimation using rgb data: Path planning application on the centauro robot. In *IEEE-RAS International Conference on Humanoid Robots*. IEEE-RAS.
- Szadkowski, R. J., Drchal, J., & Faigl, J. (2018). Terrain classification with crawling robot using long short-term memory network. In Kůrková, V., Manolopoulos, Y., Hammer, B., Iliadis, L., & Maglogiannis, I., editors, *Artificial Neural Networks and Machine Learning–ICANN 2018*, pages 771–780, Cham. Springer International Publishing.
- Tarvainen, A., & Valpola, H. (2017). Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. In *Advances in Neural Information Processing Systems*, pages 1195–1204.
- Tchapmi, L., Choy, C., Armeni, I., Gwak, J., & Savarese, S. (2017). Segcloud: Semantic segmentation of 3d point clouds. In *2017 International Conference on 3D Vision (3DV)*, pages 537–547.
- Team, T. T. D., Al-Rfou, R., Alain, G., Almahairi, A., Angermueller, C., Bahdanau, D., Ballas, N., Bastien, F., Bayer, J., Belikov, A., *et al.* (2016). Theano: A python framework for fast computation of mathematical expressions. *arXiv:1605.02688*.
- Tennakoon, E., Kottege, N., Peynot, T., & Roberts, J. M. (2018). Safe terrain probing method for multi-legged robots operating on brittle surfaces. In *International Symposium on Experimental Robotics (ISER)*.
- Tennakoon, E., Peynot, T., Roberts, J., & Kottege, N. (2020). Probe-before-step walking strategy for multi-legged robots on terrain with risk of collapse. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pages 5530–5536. IEEE.
- The Blender Foundation (n.d.). Blender, free and open source 3d creation suite.
- Thomas, J. J. (2015). Terrain classification using multi-wavelength LiDAR data. Technical report, Naval Postgraduate School, Monterey, United States.
- Treml, M., Arjona-Medina, J., Unterthiner, T., Durgesh, R., Friedmann, F., Schuberth, P., Mayr, A., Heusel, M., Hofmarcher, M., Widrich, M., *et al.* (2016). Speeding up semantic segmentation for autonomous driving. In *MLITS, NIPS Workshop*, volume 2, page 7.
- Unity Technologies (n.d.). Unity, cross-platform game engine.
- Uricar, M., Sistu, G., Rashed, H., Vobecky, A., Krizek, P., Burger, F., & Yogamani, S. (2019). Let’s get dirty: Gan based data augmentation for soiling and adverse weather classification in autonomous driving. *arXiv:1912.02249*.

- Valada, A., & Burgard, W. (2017). Deep spatiotemporal models for robust proprioceptive terrain classification. *International Journal on Robotics Research*, 36(13–14):1521–1539.
- Valada, A., Dhall, A., & Burgard, W. (2016a). Convolved mixture of deep experts for robust semantic segmentation. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) Workshop, State Estimation and Terrain Perception for All Terrain Mobile Robots*, page 23. IEEE.
- Valada, A., Mohan, R., & Burgard, W. (2019). Self-supervised model adaptation for multimodal semantic segmentation. *International Journal of Computer Vision*, pages 1–47.
- Valada, A., Oliveira, G., Brox, T., & Burgard, W. (2016b). Deep multispectral semantic scene understanding of forested environments using multimodal fusion. In *International Symposium on Experimental Robotics (ISER)*.
- Van Der Mark, W., & Gavrila, D. M. (2006). Real-time dense stereo for intelligent vehicles. *IEEE Transactions on Intelligent Transportation Systems*, 7(1):38–50.
- Vandapel, N., Huber, D. F., Kapuria, A., & Hebert, M. (2004). Natural terrain classification using 3-d lidar data. *IEEE International Conference on Robotics and Automation, 2004. Proceedings. ICRA '04. 2004*, 5:5117–5122. IEEE.
- Vertens, J., Zürn, J., & Burgard, W. (2020). Heatnet: Bridging the day-night domain gap in semantic segmentation with thermal images. In *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 8461–8468. IEEE.
- Vulpi, F., Milella, A., Marani, R., & Reina, G. (2021). Recurrent and convolutional neural networks for deep terrain classification by autonomous robots. *Journal of Terramechanics*, volume 96, pages 119–131. <https://doi.org/10.1016/j.jterra.2020.12.002>.
- Wang, C., Liu, H.-Y., Zhang, Y., & Li, Y.-f. (2014). Classification of land-cover types in muddy tidal flat wetlands using remote sensing data. *Journal of Applied Remote Sensing*, 7(1):073457.
- Wang, L., & Wang, H. (2019). Water hazard detection using conditional generative adversarial network with mixture reflection attention units. *IEEE Access*, 7:167497–167506.
- Wang, S. (2019). *Road Terrain Classification Technology for Autonomous Vehicle*. Springer.
- Weichel, H. (1990). *Laser Beam Propagation in the Atmosphere*, volume 3. SPIE Press.
- Weiss, C., Fröhlich, H., & Zell, A. (2006). Vibration-based terrain classification using support vector machines. *2006 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 4429–4434. IEEE.
- Wellhausen, L., Dosovitskiy, A., Ranftl, R., Walas, K., Cadena, C., & Hutter, M. (2019). Where should I walk? predicting terrain properties from images via self-supervised learning. *IEEE Robotics and Automation Letters*, 4(2):1509–1516.
- Wellington, C., & Stentz, A. (2004). Online adaptive rough-terrain navigation vegetation. In *IEEE International Conference on Robotics and Automation, 2004. Proceedings. ICRA '04. 2004*, volume 1, pages 96–101. IEEE.
- Winkens, C., & Paulus, D. (2018). Context aware hyperspectral scene analysis. *Electronic Imaging*, 2018(17):346–1.
- Winkens, C., Sattler, F., & Paulus, D. (2017). Hyperspectral terrain classification for ground vehicles. In *VISIGRAPP (5: VISAPP)*, pages 417–424.
- Wu, H., Zhang, W., Li, B., Sun, Y., Duan, D., & Chen, P. (2019). Visual terrain classification methods for mobile robots using hybrid coding architecture. In *2019 IEEE 4th International Conference on Image, Vision and Computing (ICIVC)*, pages 17–22. IEEE.
- Wu, X. A., Huh, T. M., Mukherjee, R., & Cutkosky, M. (2016). Integrated ground reaction force sensing and terrain classification for small legged robots. *IEEE Robotics and Automation Letters*, 1(2):1125–1132.
- Xu, Y.-S., Fu, T.-J., Yang, H.-K., & Lee, C.-Y. (2018). Dynamic video segmentation network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6556–6565. IEEE.
- Yang, K., Wang, K., Cheng, R., Hu, W., Huang, X., & Bai, J. (2017). Detecting traversable area and water hazards for the visually impaired with a prgb-d sensor. *Sensors*, 17(8):1890.
- Yogamani, S., Hughes, C., Horgan, J., Sistu, G., Varley, P., O’Dea, D., Uricár, M., Milz, S., Simon, M., Amende, K., et al. (2019). Woodscape: A multi-task, multi-camera fisheye dataset for autonomous driving. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 9308–9318. IEEE.
- Yu, F., & Koltun, V. (2015). Multi-scale context aggregation by dilated convolutions. *arXiv:1511.07122*.
- Yu, F., Xian, W., Chen, Y., Liu, F., Liao, M., Madhavan, V., & Darrell, T. (2018). Bdd100k: A diverse driving video database with scalable annotation tooling.

- Zeng, F., Jacobson, A., Smith, D., Boswell, N., Peynot, T., & Milford, M. (2017). Enhancing underground visual place recognition with shannon entropy saliency. In *Proceedings of the Australasian Conference on Robotics and Automation 2017*, pages 1–10. Australian Robotics and Automation Association.
- Zhang, P., Wang, J., Farhadi, A., Hebert, M., & Parikh, D. (2014). Predicting failures of vision systems. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3566–3573. IEEE.
- Zhao, H., Qi, X., Shen, X., Shi, J., & Jia, J. (2018). Icnnet for real-time semantic segmentation on high-resolution images. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 405–420.
- Zhao, H., Shi, J., Qi, X., Wang, X., & Jia, J. (2017). Pyramid scene parsing network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2881–2890. IEEE.
- Zhou, B., Zhao, H., Puig, X., Xiao, T., Fidler, S., Barriuso, A., & Torralba, A. (2019). Semantic understanding of scenes through the ade20k dataset. *International Journal on Computer Vision*, 127(3):302–321.
- Zhu, J.-Y., Park, T., Isola, P., & Efros, A. A. (2017a). Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2223–2232. IEEE.
- Zhu, X., Xiong, Y., Dai, J., Yuan, L., & Wei, Y. (2017b). Deep feature flow for video recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2349–2358. IEEE.
- Zhu, X. J. (2005). Semi-supervised learning literature survey. Technical report, University of Wisconsin-Madison Department of Computer Sciences.
- Zou, Y., Chen, W., Xie, L., & Wu, X. (2014). Comparison of different approaches to visual terrain classification for outdoor mobile robots. *Pattern Recognition Letters*, 38:54–62.
- Zürn, J., Burgard, W., & Valada, A. (2021). Self-supervised visual terrain classification from unsupervised acoustic feature learning. *IEEE Transactions on Robotics*, 37(2):466–481.

How to cite this article: Borges, P. V. K., Peynot, T., Liang, S., Arain, B., Wildie, M., Minareci, M. G., Lichman, S., Samvedi, G., Sa, I., Hudson, N., Milford, M., Moghadam, P., & Corke, P. (2022). A survey on terrain traversability analysis for autonomous ground vehicles: Methods, sensors and challenges. *Field Robotics*, 2, 1567–1627.

Publisher’s Note: Field Robotics does not accept any legal responsibility for errors, omissions or claims and does not provide any warranty, express or implied, with respect to information published in this article.