Journal of Field Robotics

WILEY

RESEARCH ARTICLE OPEN ACCESS

Real-Time Terrain Traversability Analysis and Mapping for Autonomous Robotics in Dynamic Environments: Fusing Appearance- and Geometry-Based Approaches

Semih Beycimen[1] | Dmitry Ignatyev[2] | Argyrios Zolotas[2]

[1]Faculty of Electrical and Electronics, Control and Automation Engineering, Istanbul Technical University, Resitpasa, Sariyer/Istanbul, Turkiye | [2]Centre for Assured and Connected Autonomy, Faculty of Engineering and Applied Sciences, Cranfield University, Bedford, UK

Correspondence: Semih Beycimen (semihbeycimen@gmail.com; semih.beycimen@cranfield.ac.uk)

Received: 29 November 2024 | **Revised:** 9 September 2025 | **Accepted:** 27 September 2025

Funding: The first author acknowledges the Republic of Turkey, Ministry of National Education (YLYS), for supporting the studies under PhD scholarship ref. U9BYTAB2LDGA7LK.

Keywords: artificial intelligence | autonomous vehicles | data fusion | data processing | machine learning | off-road | sensors | terrain traversability | unstructured environment

ABSTRACT

This paper presents advanced methodologies for real-time terrain analysis and mapping in autonomous robotic systems. The focus is on appearance-based terrain traversability analysis and geometric-based terrain traceability analysis. In the appearance-based approach, an enhanced segmentation model using pixel-based augmentation and 13 unique classes is proposed for reliable terrain classification. Semantic images are projected onto a 2.5D map by transforming two-dimensional image data into a three-dimensional coordinate system. The geometric-based approach involves depth estimation from stereo cameras, employing three Zed-2 cameras and the Depth Sensing application programming interface. The research contributes to improved perception and decision-making capabilities of autonomous robots operating in complex and dynamic environments and also provides a new comprehensive data set named CranfieldTerra. Experimental results validate the effectiveness of the proposed methodologies, demonstrating their potential in various applications, such as search and rescue, agriculture, and exploration. This study establishes a foundation for further advancements in autonomous robotics, enhancing their ability to navigate safely and efficiently in challenging terrains.

1 | Introduction

Terrain traversability analysis plays a vital role in facilitating autonomous robot navigation across diverse environments. The accurate assessment of terrain traversability enables informed decision-making, avoidance of hazardous areas, and efficient path planning for robots. Traditionally, terrain traversability analysis has been approached from either a geometric or an appearance-based perspective. Geometric-based methods focus on extracting terrain features, such as slope, step, and roughness, from elevation maps or point-cloud data. Conversely, appearance-based methods rely on visual information captured by cameras or other optical sensors to classify terrain types based on their visual characteristics.

While both geometric- and appearance-based approaches have demonstrated their individual merits, each method also exhibits limitations. Geometric-based methods may encounter challenges in capturing nuanced terrain properties, particularly in complex and heterogeneous environments. The accuracy of geometric-based analysis heavily relies on the quality and resolution of elevation data, which can be demanding to acquire in

https://doi.org/10.1002/rob.70097

real-time or in certain scenarios. Furthermore, geometric features alone may not provide a comprehensive understanding of terrain traversability, as they do not account for certain visual cues influencing robot locomotion, such as texture and vegetation coverage. Conversely, appearance-based methods excel in capturing such visual cues, enabling robots to identify different terrain types based on textures, colors, and other optical properties. However, appearance-based methods may face limitations due to environmental variations, lighting conditions, or occlusions, which can affect the reliability and generalization of the analysis.

To overcome the limitations of individual methods and enhance the accuracy and robustness of terrain traversability analysis, a fusion approach has emerged, combining geometric- and appearance-based information. This fusion aims to leverage the complementary strengths of each method and provide a more comprehensive and reliable assessment of terrain traversability. By integrating the geometric- and appearance-based analyses, the fused traversability map can capture both the fine-grained geometric details influencing robot locomotion and the visual cues influencing terrain classification.

In recent years, significant research efforts have been devoted to terrain traversability analysis, fusion, and mapping using geometric- and appearance-based approaches. Several studies have explored the individual merits of these methods, as well as their integration to enhance terrain understanding and improve robotic navigation.

A classification method with an elevation grid map (Fankhauser and Hutter 2016) created with height features of terrain, such as maximum, minimum, mean heights, and height difference (min–max difference) from the transformed point-cloud data proposed in Meng et al. (2018) can be classified as a geometry-based hybrid method. Then, the terrain has been classified using features of height, roughness and slope angle. The classified type of terrain has been determined according to these terrain features with a comparison approach. The study has been conducted using a three-dimensional (3D) light detection and ranging (LiDAR) and concluded that the method was reliable for classifying the aforementioned terrain type.

A more recent study by Thakker et al. (2020) demonstrated a navigation method for developing a point-cloud-based traversability model for challenging environments. The multifidelity mapping has been generated with depth measurement from LiDAR and pose estimation from different sensors. Traversability assessment has been predicted with this map, and the point cloud was segmented into two classes, that is, ground and obstacle. Also, the platform was included with this point-cloud map according to pose estimation. Then, the traversability matrix was calculated from the settled pose, the point cloud of the surface and its interaction. Furthermore, the cost maps were generated from this point-cloud traversability map for planning algorithms. The proposed method has been tested on four different platforms, that is, Clearpath Husky A200 (skid-steer platform), Telemax Pro (tracked vehicle), X-Maxx (Ackermann suspension), and Spot (Quadruped robot). Although the model is providing useful outcomes in general, it has not provided reliable results for detecting small, narrow and negative (such as holes) obstacles and some unexpected hazards, such as puddle areas or mud. The architecture of the proposed method can be seen in Figure 1.

In a further study, RGB images have been converted to semantic segmentation images based on supervised learning, and these images were fused with a point cloud from the LiDAR sensor to create the semantic mapping. Then, path-planning algorithms have been conducted from this 2.5D semantic map based on point clouds and segmented images and from traversability cost scores that present semantic classes. For semantic segmentation, DeepScene labeled six classes, and Yamaha-CMU data set, Maturana et al. (2018) labeled eight classes for the training of the segmentation algorithm with the fully convolutional network (used to label all pixels of data). Also, the researchers proposed a new learning segmentation algorithm to decrease training speed. It is worth noting that the data set was labeled, and the geometric- and appearance-based traversability model considered both height difference and terrain class.

From a space-related application, Rothrock et al. (2016) proposed a method for Mars rover mission, orbital, and ground-based terrain classification, based on Soil Property and Object Classification that predicts terrain types and features. The study

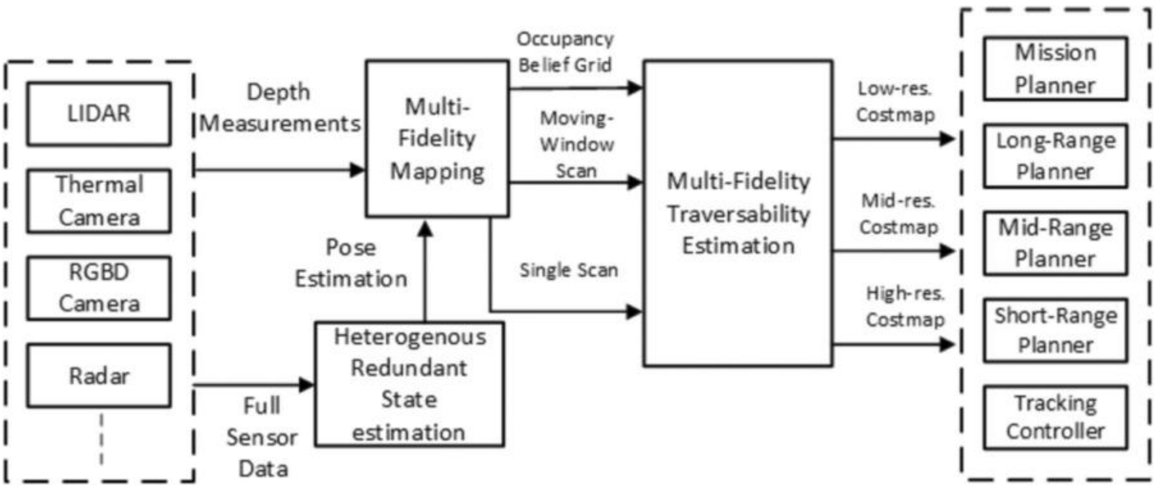


FIGURE 1 | Overall layout of resilient multifidelity method (Thakker et al. 2020). LiDAR, light detection and ranging.

has included two significant steps/missions. The first is the landing site traversability analysis method to categorize the terrain. The method has been applied to 17 terrain types using the semantic segmentation method based on supervised learning. Then, a traversability class has been created with identified terrain classes. Data labeling progress has been completed via web labeling tools manually. Also, some preprocessing algorithms, such as normalizing, darkening, and converting images to bird-eye images, cropping, and masking using RANSAC, have been used to create a realistic Mars environment. The study has been verified in Columbia Hills and was deemed suitable by the researchers for the landing study of the rover. The method has predicted the terrain types, such as smooth regolith, dense ridges, rock fields, scarps, and deep sand with 90.2% accuracy overall. But, there was some misclassification, especially for small rocks, due to issues, such as data distribution. The second step is the slip prediction method, and the terrain classes predicted in the previous step have been fused with slope and wheel slip. Terrain slope has been predicted from the rover's tilt and the slip and rover's position have been calculated with visual odometry. The result shows that the prediction of slip for rocks is acceptable. However, the results are not satisfactory for sand due to some geometric and appearance features, such as terrain geometry and sand depth. Of course, this paper presented a more bespoke demanding environment, that of an extraterrestrial planet.

A method of fusing 3D cost maps and terrain classification has been proposed by Roncancio et al. (2014). Three main steps were followed, that is, creating a 3D traversability cost map, terrain segmentation/classification and obtaining a segmented traversability map. The traversability map has been generated from the 3D point cloud, and the cost score included a combination of terrain slope, cell elevation, and obstacles. The cell-based terrain classification is created by a feature vector included 14 properties of the image, such as values of RGB channels, skewness, and entropy. Also, inverse perspective mapping and tuned support vector machine algorithms have been used to eliminate the distribution of image views for better classification and to try to find the best training parameters, respectively. The accuracy of the method has been calculated according to the combination of traversable and untraversable predictions. The last main step is that the 3D traversability cost map has been converted to a grayscale image and then to a segmented traversability map. In this study, the cost value in every cell has been associated with the terrain class. Furthermore, the proposed approach has enabled the terrain to be classified successfully, and the method has been updated simultaneously using online learning. But it has classified the terrain using two classes, that is, traversable and untraversable (not in a more continuous form).

A method based on images converted to height maps that represent the terrain of 10 m × 10 m by synthesizing data was presented in Omar Chavez-Garcia et al. (2017). Then, features such as terrain steepness and height of the steps were calculated, and convolutional neural network (CNN)-based approaches were used to predict the traversability of the terrain. The proposed method was verified using various data sets and tested with a real platform, that is, Pioneer 3-AT. The test has shown that the CNN-based method is better than the feature-based and baseline classifier methods. The main shortcoming of the study is that the method has been trained with simulation data only.

A semantic map-generating method based on the simultaneous localization and mapping (SLAM) algorithm has been proposed by W. Deng et al. (2020). The semantic segmentation algorithm has been set using the Inception-v3 learning architecture and an RGB data set generated in the RoboCup Rescue-Robot-League environment. Then, an SLAM algorithm has been implemented to create a geometric map using depth and RGB images. Also, labeled images have been converted to refined semantic images with a flood-filling algorithm using semantic images and depth information from a stereo camera. With these semantic images and geometric maps, semantic and filtered semantic maps have been generated to reflect the environment. This method has been provided by using dense information from the environment and more reliable mapping.

Hedrick et al. (2020) proposed a path-planning method for Mars surface missions. In the study, the terrain traversability model has been obtained from the digital elevation model (DEM), cumulative fractional area (CFA) map, and terrain classes. The CFA map has been labeled into low, medium and high classes, and the slope computed from the DEM has been divided into five classes. The traversability map has been created based on these terrain and vehicle features, and the velocity map has been labeled with five different velocity ranges and an untraversable label. The terrain has been segmented into benign, rough, sandy, and untraversable areas. Then, the traversability map was updated with new measurement information. After creating the expected velocity category map, Yen's (1970) k-shortest paths algorithm has been implemented to find the most reliable path-planning options. By this proposed method, the terrain can be learned by the platforms simultaneously.

In another recent study (Leung et al. 2022), RGB images, point cloud from LiDAR, and robot motion data have been used to generate the traversability cost map. The method has three main stages, that is, converting RGB images to the semantic mask with a supervised semantic segmentation algorithm, generating a 2.5D robot-centric elevation map and obtaining the traversability cost map. The Gated-SCNN (Takikawa et al. 2019), ERFNet (Romera et al. 2017) architectures, the RELLIS-3D (Jiang et al. 2021), and ImageNet (J. Deng et al. 2009) data sets, which have 18 different terrain types, were used to train semantic segmentation algorithms. The 2.5D elevation map that includes the height information on the terrain in each grid cell was generated using 3D point-cloud and robot motion data. After that, the robot-centric cost map has been created with several transformations, such as converting coordinates from map to camera and from camera to pixel. Also, the slope, roughness and step height of the terrain have been calculated from height information on the elevation map. Combining maps based on terrain types and geometric information, the traversability cost map was thus generated. In this way, a reliable traversability map is obtained based on both appearance and geometric information for the terrain. However, the proposed method was not tested on a real terrain setup, and the data set used for training and validation was limited.

The research on object detection, perspective, and obstacle recognition in robotics, as presented in Xu et al. (2022), is thoroughly examined. The key findings, methodologies employed, and recommendations from each study are systematically analyzed and summarized in a tabular format.

Moshayedi et al. (2023) explore the integration of Unity and the Robot Operating System (ROS) for simulating and controlling robots, with a particular focus on Automated Guided Vehicles (AGVs). The study emphasizes the benefits of utilizing Unity for realistic 3D visualizations and ROS for advanced robotic control. The researchers successfully integrated TurtleBot2i and Razbot AGV platforms into a 3D Unity environment, demonstrating precise simulation and control. Their findings reveal that Unity significantly enhances graphical interface quality and interactivity. Additionally, the paper underscores the importance of proper workspace configuration to prevent simulation issues. Future research will aim to evaluate various AGV models, comparing their real-world and simulated performances to gain deeper insights into their effectiveness.

In this paper, an investigation is conducted into real-time geometric- and appearance-based terrain traversability analysis, fusion, and mapping. The focus lies on the process of extracting geometric features from elevation maps or point clouds, utilizing techniques, such as slope estimation, step detection, and roughness analysis. Additionally, appearance-based methods are explored, making use of visual information to classify terrain types, leveraging machine learning algorithms, and image processing techniques. By analyzing the visual appearance of terrains, important cues related to texture, color, and other optical properties that impact traversability can be captured.

Various fusion strategies are then examined to integrate the information from these two approaches, aiming to produce a unified and enhanced traversability map. These strategies encompass simple fusion techniques, such as weighted averaging or maximum/minimum selection, as well as more sophisticated approaches, such as machine learning-based fusion models. The fused traversability map provides a holistic representation of terrain traversability, considering both geometric- and appearance-based aspects.

Additionally, the challenges associated with real-time analysis and mapping are explored, as timely and efficient terrain traversability assessment is crucial for dynamic robotic navigation scenarios. Potential algorithmic optimizations, hardware considerations, and parallel computing techniques are investigated to ensure real-time performance without compromising the accuracy of the fused traversability analysis. Furthermore, the integration of the traversability map within a mapping framework is discussed, allowing robots to navigate and update the map in real-time as they explore the environment.

By fusing the geometric- and appearance-based information in real-time, autonomous robots can improve their understanding of surrounding terrains, enabling safer and more efficient navigation in complex and dynamic environments. This paper aims to provide an in-depth exploration of real-time geometric- and appearance-based terrain traversability.

1.1 | Contribution and Novelty

This paper makes a significant contribution to the field of autonomous robotics, specifically in the domain of real-time terrain analysis and mapping. The methodologies introduced represent a fusion of appearance-based and geometric-based approaches to terrain traversability analysis, providing a more comprehensive solution for autonomous navigation in dynamic and complex environments. The key contributions and novelty of this study are outlined as follows:

Advanced Terrain Classification: The development of a robust semantic segmentation model using pixel-based augmentation and 13 distinct terrain classes marks a significant advancement in terrain classification. This approach offers enhanced accuracy compared with existing models, particularly in environments where terrain types are diverse and complex, which are often difficult for conventional systems to analyze.

Introduction of the CranfieldTerra Data Set: This paper introduces the CranfieldTerra data set, which includes both appearance data (RGB, greyscale, labeled images, etc.) and geometric data (LiDAR, point cloud, stereo point cloud, inertial measurement unit [IMU], Global Positioning System, and odometry) and robot-specific data (robot geometric data, calibration files, etc.). This comprehensive data set expands the understanding of terrain traversability by incorporating a wide range of terrain types, providing a valuable tool for future research in autonomous robotics. The CranfieldTerra data set serves as a detailed foundation for machine learning models focused on terrain classification, analysis, and mapping, ultimately enabling more reliable and adaptable autonomous systems.

Fusion of Geometric- and Appearance-Based Methods: The combination of appearance-based terrain classification and geometric-based terrain analysis results in an integrated approach that maximizes the strengths of both methods. The fused traversability map, created by blending semantic information with depth estimation, provides a more dynamic and accurate representation of the terrain. This fusion greatly enhances autonomous robots' decision-making and navigation capabilities, improving their ability to navigate in varied environments.

Real-Time Traversability Mapping: The system's real-time performance showcases its effectiveness in autonomous applications, such as search and rescue, agriculture, and exploration. By integrating multiple sensor inputs, including stereo cameras and LiDAR, the system ensures continuous terrain analysis, enabling autonomous systems to operate safely and efficiently in real-time without interruptions.

Improvement of Traversability Evaluation: The methodologies presented improve the traversability evaluation process by combining both geometric features (such as slope, step height, and roughness) and appearance-based features (terrain class scores). This multisensor fusion approach results in a more robust and reliable traversability map, enabling autonomous robots to navigate safely in a wide range of environments, including those that may have been challenging or previously inaccessible. Importantly, the evaluation of geometric information, such as slope, roughness, and step height, is not limited to LiDAR data alone. This paper also demonstrates the ability to extract and evaluate these features from stereo cameras (front, back, and side), significantly expanding the range of data sources used for traversability analysis and enhancing the reliability of the system.

Introduction of the Real-Time Switch Model: The Real-Time Switch Model introduced in this paper is designed to dynamically adapt the performance of terrain analysis models based on the density rate of the environment. This approach optimizes the balance between accuracy and response time by selecting appropriate models tailored to different environmental complexities. Specifically, it utilizes the FRRN-B model for high-density environments (over 50%), where cluttered scenes require high segmentation accuracy. For moderate-density environments (between 25% and 50%), the DenseASPP and ERFNet models are employed to achieve a balance between computational efficiency and segmentation precision. In sparse environments (below 25% density), the DenseASPP model alone is sufficient, offering quick processing times without significant accuracy trade-offs. This dynamic model-switching strategy ensures efficient computational resource usage while maintaining high segmentation accuracy across diverse environmental conditions, enabling robust adaptability and optimizing real-time performance in autonomous systems.

The novelty of this study lies in its potential to significantly enhance the operational efficiency and safety of autonomous systems across various domains. The methodologies presented in this paper set a new standard for terrain analysis and mapping, laying the groundwork for the development of more advanced, reliable, and adaptable autonomous robotic systems. By addressing real-time performance, multisensor fusion, and terrain variability, this study represents a major step forward in the deployment of autonomous systems in complex, real-world scenarios.

2 | Methodology

2.1 | Appearance-Based Terrain Traversability Analysis

2.1.1 | Semantic Segmentation Algorithm

In our previous work (Beycimen et al. 2025, 2022), we proposed an enhanced segmentation model. This model incorporated a pixel-based, detailed augmentation method utilizing 13 unique classes for reliable terrain classification. For the annotation process, various grid sizes were employed: a wider grid for the clearer, deterministic portions of the classes, and a denser grid for the transition areas between classes, enabling more accurate and detailed annotation. Although this approach was time-consuming, efforts are underway to automate the process.

TABLE 1 | Color coding for semantic segmentation classes.

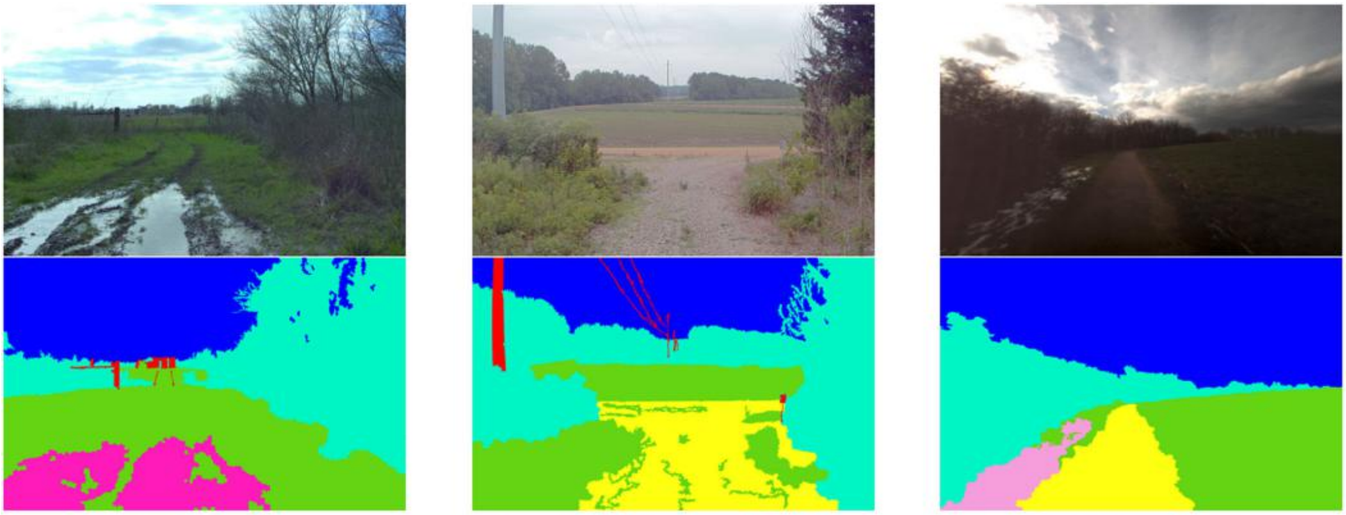
Class	Red	Green	Blue	Color
Sky	0	0	255	🟦
Vegetation	0	245	196	🟦
Objects	255	0	0	🟥
Person	126	47	142	🟪
Pavement	162	20	47	🟫
Vehicle	255	0	255	🟪
Mud	255	26	185	🟪
House	128	128	128	⬜
Asphalt road	217	83	25	🟧
Smooth road	255	255	17	🟨
Rough road	246	158	220	🟪
Very rough road	246	18	97	🟥
Low traversable grass	100	212	19	🟩

TABLE 2 | Accuracy rate and response time of FRRN-B and DenseASPP NN models (Mini ITX computer with Nvidia GTX1060 GPU).

Network models	Average test accuracy	Response time (approximately) (s)
FRRN-B	90.7	0.75
BiSeNet	88.1	0.58
ERFNet	87.9	0.49
DenseASPP	87.1	0.25

The color coding and image snapshots from both existing data sets and our data set are shown in Table 1 and Figures 2 and 3. Several feature extraction methods and neural networks were evaluated and compared to establish more robust models. The following models were employed for feature extraction: *ResNet*50, *ResNet*101, *ResNet*152, *MobileNetV*2, and *InceptionV*4. These networks utilized pretrained versions trained on over a million images from the ImageNet database. The primary classification algorithms used the following networks: *FC-DenseNet*56, *FC-DenseNet*67, *FC-DenseNet*103, *Encoder-Decoder*, *Encoder-Decoder-Skip*, *RefineNet*10, *FRRN-A*, *FRRN-B*, *MobileUNet*, *MobileUNet-Skip*, *PSPNet*, *GCN*, *DeepLabV*3, *DeepLabV*3_*plus*, *AdapNet*, *DenseASPP*, *BiSeNet*, *ERFNet*, and *Modified-BiSeNet*. Some of these neural network models used in real-time implementation are explained below.

The augmentation method demonstrated superior performance compared with literature studies, yielding high accuracy rates. Additionally, a comprehensive data set was used in that study. However, the results for the *mud* class were not satisfactory, and the results for the *person* and *objects* classes were slightly lower. These issues were due to the limited number of classes in the data set. We collected a new data set from a dense environment in Aspley Woods, Bedford, England, to address this. We relabeled the data using the same augmentation method and trained the models with the same algorithms and parameters on the same computer. Some image snapshots from this data set are shown in Figure 4.



(a) Rellis-3D (b) CAT: CAVS Traversability Dataset (c) Freiburg Forest

(d) SEMFIRE forest (e) MAGNUM Off-Road (f) YAMAHA-CMU

FIGURE 2 | Example image snapshots and data labeling from existing data sets in the literature of terrain mapping. (a) Rellis-3D, (b) CAT:CAVS traversability data set, (c) Freiburg Forest, (d) SEMFIRE forest, (e) MAGNUM off-road, and (f) Yamaha-CMU. [Color figure can be viewed at wileyonlinelibrary.com]

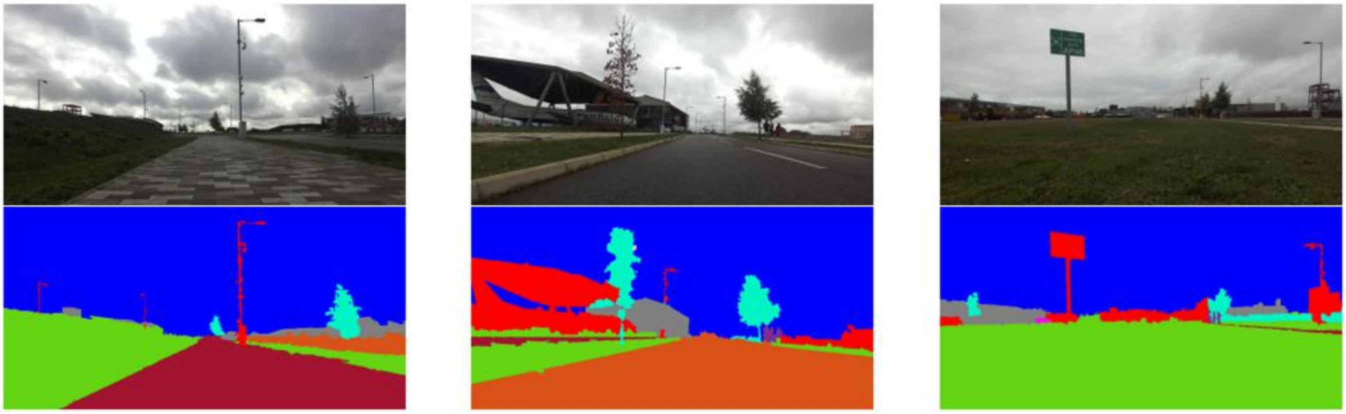


FIGURE 3 | Data labeling of CranfieldTerra data set. [Color figure can be viewed at wileyonlinelibrary.com]

FIGURE 4 | Expansion of CranfieldTerra data set: Data labeling for new terrain collection. [Color figure can be viewed at wileyonlinelibrary.com]

2.1.2 | Reflecting Semantic Images on a 2.5D Map

After semantic segmentation, images are projected onto a 2.5D map by transforming two-dimensional (2D) image data into a 3D coordinate system using camera calibration parameters and depth information from the robot-centric elevation map. This process requires intrinsic parameters (focal length, optical center, and distortion coefficients) and extrinsic parameters (rotation and translation), which were obtained from Zed-2 camera files ($image_{\text{info}}$).

The Pinhole Camera Model (PCM) (Forsyth and Ponce 2002) was used to map 2D images to world coordinates. PCM assumes a perspective projection where a 3D point $P(X, Y, Z, 1)$ is mapped to an image point p via

$$p = K[R|t]P, \tag{1}$$

where K represents the intrinsic parameters, and $[R|t]$ defines the camera's pose in the world frame.

After the transformation of images, semantic mapping was created according to terrain classes obtained from semantic segmentation algorithms. The semantic images include a score that represents the terrain types, such as 1:sky, 2:house, 3:asphalt road, 4:rough road, and so forth. According to each class, a traversability score, which is between 0 and 1, was assigned. Here, 0 means untraversable, and 1 means fully traversable. The detailed progress can be seen in Algorithm 1.

Algorithm 1. Algorithm for appearance-based terrain classification scores.

```
Input: $class \in \{class_{\text{Sky}}, class_{\text{Vegetation}}, class_{\text{Objects}}, class_{\ldots}\}$
Output: $T_a \in [0, 1]$ (Terrain Classification Score)
$i \leftarrow 13$
if $class == class_{\text{Sky}}$ then
    $T_a = 0$
```

(Continues)

```
else if $class == class_{\text{Vegetation}}$ then
    $T_a = 0.05$
else if $class == class_{\text{Objects}}$ then
    $T_a = 0$
else if $class == class_{\text{Person}}$ then
    $T_a = 0.05$
else if $class == class_{\text{Pavement}}$ then
    $T_a = 0$
else if $class == class_{\text{Vehicle}}$ then
    $T_a = 0.05$
else if $class == class_{\text{Mud}}$ then
    $T_a = 0.2$
else if $class == class_{\text{AsphaltRoad}}$ then
    $T_a = 1$
else if $class == class_{\text{SmoothRoad}}$ then
    $T_a = 0.9$
else if $class == class_{\text{RoughRoad}}$ then
    $T_a = 0.7$
else if $class == class_{\text{VeryRoughRoad}}$ then
    $T_a = 0.4$
else if $class == class_{\text{LowTraversableGrass}}$ then
    $T_a = 0.5$
end if
```

2.1.3 | Real-Time Switch Model

In this study, we developed a *Real-Time Switch Model* to dynamically adjust the performance of semantic segmentation models based on the density rate of the environment. The goal of this method is to optimize the balance between accuracy and response time by selecting appropriate models tailored to varying levels of environmental complexity.

The *environment density rate* is a critical factor in model selection. It is calculated by dividing the total pixel count of each image by the aggregated pixel count of key environmental features, which include "Vegetation," "Objects," "Person," "Vehicle," "Mud," and "House" classes. This metric provides a quantifiable measure of environmental density, guiding the model selection process.

To identify the most suitable models for different terrain types, four neural network models were evaluated: *FRRN-B*, *DenseASPP*, *BiSeNet*, and *ERFNet*. These models were selected based on their performance metrics, specifically accuracy and response time, across various terrains within the Cranfield Campus, which included both off-road and semi-off-road areas.

During the evaluation, the models were tested for their ability to classify and segment environmental features accurately and efficiently. On the basis of the density rate calculation, the models were assigned as follows:

- *FRRN-B:* This model was designated for environments with a density rate exceeding 50%, where the terrain is highly complex and cluttered. Its robust architecture, which focuses on feature refinement and residual connections, makes it effective in handling dense scenes with numerous classifiable elements.
- *DenseASPP and ERFNet:* These models were selected for environments with a density rate between 25% and 50%. In these moderately cluttered terrains, the goal was to strike a balance between computational efficiency and segmentation precision. DenseASPP, with its densely connected atrous spatial pyramid pooling, captures multiscale context while maintaining computational efficiency, while ERFNet provides faster processing times with acceptable accuracy.
- *DenseASPP:* In environments with a density rate below 25%, where the terrain is relatively sparse, the DenseASPP model alone was deemed sufficient. This model is particularly suited for scenarios with fewer objects of interest, providing quick processing times with minimal trade-off in accuracy.

This *Real-Time Switch Method* dynamically selects the appropriate model based on the evaluated environmental density, ensuring that the system adapts in real-time to varying terrain complexities. By incorporating this dynamic switching strategy, the system maintains optimal performance across diverse environments, ensuring both high segmentation accuracy and efficient use of computational resources. For a more comprehensive explanation, please refer to Beycimen et al. (2025).

2.2 | Geometric-Based Terrain Traceability Analysis

2.2.1 | Depth Estimation from Stereo Cameras

Stereo cameras are a type of camera system comprised of two or more cameras with intersecting fields of view. Computing the disparity between corresponding points in two images and using this disparity to estimate depth is required for stereo camera depth estimation (Scharstein and Szeliski 2003). We used three Zed-2 cameras in this study, and depth estimation was calculated using the Depth Sensing application programming interface by Stereolabs.

Here is our general depth estimation process flow:

- *Calibration:* The initial phase involves calibrating the stereo cameras. This requires determining the relationship between the physical configuration of the cameras and their image output. It requires knowing the focal length, optical center, and coefficients of lens distortion. In this investigation, three Zed-2 cameras were employed, and the ZED SDK self-calibration tool was used to recalculate and refine calibration parameters.
- *Rectification:* After calibration, images need to be rectified. This procedure entails aligning the two images so that points that correspond lie on the same horizontal line. It is a crucial stage because it facilitates the problem of locating points that correspond between the left and right images.
- *Disparity computation:* When images have been corrected, the disparity can be calculated. Disparity refers to the positional difference of the same object as seen by two cameras. This can be accomplished through the use of block matching, semiglobal block matching, graph cuts, and so forth. Essentially, these algorithms compare small windows or blocks between two images in an effort to identify the closest match for each block.
- *Depth estimation and Postprocessing:* After calculating the disparity, the profundity can be estimated. This is founded on the triangulation principle. Given the disparity d, focal length f, and baseline B (the distance between the cameras), the depth Z can be calculated as $Z = fB/d$. Keep in mind that disparity maps and depth maps are fundamentally the same form of data (they both display the distance of objects from the camera), but they use different units. Although disparity is inversely proportional to depth, it is not the same as depth. The depth map required postprocessing to yield better results. To display the depth map, its values are scaled to the range $[0, 255]$, where 255 (white) represents the closest possible depth value and 0 (black) represents the farthest possible depth value as a depth normalization step.
- *Three-dimensional Reconstruction/Point-Cloud Model:* The depth map can be used to generate a point cloud, and then a surface reconstruction algorithm can be applied to reconstruct a 3D model of the scene. Here, the X and Y coordinates of each point represent the pixel coordinates, while the Z coordinate represents the depth. Each of the four channels used to store the point cloud's data corresponds to a 32-bit floating-point value. The channels include R, G, B, and alpha (4×8-bit), which are combined into a single 32-bit floating-point value to store color information.

2.2.2 | Elevation Mapping Model

A robot-centric elevation map is a spatial data representation that encodes the environment's 3D structure relative to the robot's position and orientation. This map captures the adjacent terrain's elevation information while preserving a compact and efficient representation for real-time processing and adaptation. The robot-centric elevation map provides a more precise and dynamic comprehension of the environment, enabling autonomous robotic systems with enhanced localization and navigation capabilities.

In contrast to traditional mapping techniques such as occupancy grid mapping and SLAM, which frequently prioritize 2D spatial information, robot-centric elevation maps emphasize elevation data to provide a more complete representation of the environment. This emphasis on elevation enables the capture of crucial terrain features, such as gradients, irregular surfaces, and obstacles, which are essential for a variety of robotic applications, including navigation in unstructured and challenging environments.

The process of creating a robot-centric elevation map involves combining sensory data—typically obtained from LiDAR, stereo cameras, or other range sensors—into a grid-like structure with the robot's location as the origin. Each unit in the grid represents a localized area in the environment and contains elevation and confidence values associated with that area. By revising the map in real-time as the robot moves and collects new sensor data, the robot-centric elevation map maintains a continually evolving representation of the environment, enabling precise and efficient localization and navigation.

Given its unique advantages in representing complex environments and its potential to improve robot localization and navigation capabilities, the robot-centric elevation map is a subject of considerable interest among robotics researchers. This mapping technique is expected to yield significant advancements in the development of autonomous robotic systems capable of navigating and interacting with a vast array of environments as more research is conducted on it.

In this study, the elevation mapping model was obtained as in the study of Fankhauser et al. (2018). Creating a robot-centric elevation map involves the following stages, which can be broadly categorized as follows:

- *Data Acquisition:* Sensory data from A Velodyne VLP-16 LiDAR and three Zed-2 stereo cameras provided information about the 3D structure of the surrounding environment. These data typically consist of point clouds, depth images, or other representations of environmental distance measurements.
- *Sensor Calibration and Transformation:* These sensors were calibrated to ensure accurate measurements and account for potential discrepancies caused by misalignment, noise, or other variables. Under contemplation of the robot's position, orientation, and sensor mounting configuration, the acquired sensor data was transformed into the robot's coordinate system. This phase enables the integration of sensory data into a robot-centric spatial representation that is coherent.
- *Map Initialization:* The robot-centric elevation map is initialized by defining its properties, such as the grid size, cell resolution, and the initial robot's position within the map. The map is designed to provide sufficient coverage of the environment while maintaining a balance between computational efficiency and spatial detail.
- *Elevation Estimation and Confidence Update:* For each transformed data point, the corresponding elevation value was determined, and the corresponding elevation map cell was updated. Each cell was also assigned confidence values, which represent the accuracy of the elevation data.
- *Map Updating and Maintenance:* The robot-centric elevation map needs to be perpetually updated as the robot moves through the environment and accumulates new sensor data to maintain an accurate representation of the surroundings. This entails updating elevation values, confidence levels, or other map properties based on updated sensor data. Additionally, the map requires repositioning or resizing to accommodate the robot's shifting position and ensure continuous coverage of the surrounding environment.
- *Map Fusion and Filtering:* The range measurement data for cells with new measurements was fused with robot motion data for all cells. In this way, the enhanced elevation map included height information from the terrain obtained.
- *Map Utilization:* The robot-centric elevation map is now usable for a variety of robotic applications, including localization, path planning, obstacle detection and avoidance, and terrain analysis. Elevation and confidence data from the map can be incorporated into these algorithms to improve the robot's perception and navigation abilities. These maps were used to obtain geometric features of terrain, such as slope, step height, and roughness. These progressions were detailed in Section 2.3.

2.3 | Extracting Geometric Features of Terrain

Assume that we have a robot-centric elevation map M with dimensions $(m \times n)$, in which each cell (i, j) in the map contains an elevation value $z(i, j)$.

The terms means and formulas used to compute the slope, step, and roughness maps are as follows:

1. *Slope:* The slope is a measure of steepness or the degree of inclination of a surface. The slope at a point (i, j) can be calculated using the gradient of the elevation map at that point. The gradient vector is given by the partial derivatives of the elevation map in the x- and y-directions, denoted as $\frac{\Delta z}{\Delta x}$ and $\frac{\Delta z}{\Delta y}$, respectively. The magnitude of the gradient vector is the slope, and it can be calculated as follows:

$$slope(i, j) = \sqrt{\left(\frac{\Delta z}{\Delta x}\right)^2 + \left(\frac{\Delta z}{\Delta y}\right)^2}.$$

To compute the slope angle in degrees, you can use the arctangent function:

$$slope\_angle(i, j) = atan(slope(i, j)) \times \frac{180}{\pi}.$$

2. *Step:* This represents the difference in elevation between two adjacent points. It is related to the concept of "gradient" in mathematics. The step at a point can be approximated by the maximum difference in elevation between the point and its neighbors:

$$\begin{aligned} step(i,j) = \max(&|z(i,j) - z(i+1,j)|, |z(i,j) \\ &- z(i-1,j)|, |z(i,j) - z(i,j+1)|, |z(i,j) \\ &- z(i,j-1)|). \end{aligned}$$

3. *Roughness:* This is a measure of the variability in the elevation of a surface. The roughness at a point can be defined in various ways depending on the specific application. A common measure is the standard deviation of the elevations in a neighborhood around the point:

$$\begin{aligned} roughness(i,j) = \ &std\_dev(z(i-1,j-1), z(i-1,j), \\ &z(i-1,j+1), z(i,j-1), z(i,j), z(i,j+1), \\ &z(i+1,j-1), z(i+1,j), z(i+1,j+1)). \end{aligned}$$

2.3.1 | Geometric-Based Terrain Traversability Mapping

After calculating step height, slope, and roughness features, they are reflected on a 2.5D traversability map. The critical values of $roughness_{\text{critical}}$, $steo_{\text{critical}}$, and $slope_{\text{critical}}$ are the maximum values that the platform can go without falling or rolling. If the values of step height, slope, or roughness are higher than critical values, the traversability score is indicated as 0. If not, the traversability scores are calculated as below:

$$\begin{aligned} rougness_{\text{traversability}} &= 1 - rougness/rougness_{\text{criticalValue}}, \\ slope_{\text{traversability}} &= 1 - slope/slope_{\text{criticalValue}}, \\ step_{\text{traversability}} &= 1 - step/step_{\text{criticalValue.}} \end{aligned}$$

These steps can be progressed as in Algorithm 2.

Algorithm 2. Algorithm for geometric-based terrain traversability scores.

```
Input: $rougness$, $step$, $slope$ (Geometric features of the terrain)
    $rougness_{critical}$, $step_{critical}$, $slope_{critical}$ (Critical threshold values for each feature)
Output: $rougness_{traversability}$, $step_{traversability}$, $slope_{traversability}$ (Traversability scores for each feature)
$i \leftarrow 10$
if $rougness \leq rougness_{criticalValue}$ then
  $rougness_{traversability} \leftarrow 1 - rougness/rougness_{critical}$
else
  $rougness_{traversability} \leftarrow 0$
```

(Continues)

```
end if
if $step \leq step_{criticalValue}$ then
  $step_{traversability} \leftarrow 1 - step/step_{critical}$
else
  $step_{traversability} \leftarrow 0$
end if
if $slope \leq slope_{criticalValue}$ then
  $slope_{traversability} \leftarrow 1 - slope/slope_{critical}$
else
  $slope_{traversability} \leftarrow 0$
end if
```

Then these three traversability features were fused as follows:

$$\begin{aligned} T_{\text{g}} = \omega_1 * rougness_{\text{traversability}} + \omega_2 * step_{\text{traversability}} \\ + \omega_3 * slope_{\text{traversability}}). \end{aligned}$$

Here, ω_1, ω_2, and ω_3 are the weights, and the sum of these values is equal to 1. These traversability scores were calculated and saved for all grids. In this way, geometric-based traversability was obtained.

2.3.2 | Fusing Appearance and Geometric-Based Traversability Maps

The geometric-based traversability map was obtained using slope, step height and roughness traversability values and the appearance-based traversability map was created using semantic images where all classes have a traversability score. These two maps were fused with a basic method in which geometric and semantic traversability scores were summed for each grid.

Assume we have the traversability of a cell (i,j) according to the geometric-based method $T_{\text{g}}(i,j)$, and the appearance-based method $T_{\text{a}}(i,j)$.

A simple way to fuse these two traversability maps is to take the weighted average of the two traversability values for each cell:

$$T_{\text{f}}(i,j) = w_{\text{g}} T_{\text{g}}(i,j) + w_{\text{a}} T_{\text{a}}(i,j).$$

Here, $T_{\text{f}}(i,j)$ is the fused traversability of cell (i,j), and w_{g} and w_{a} are the weights for the geometric-based method and appearance-based method, respectively (where $w_{\text{g}} + w_{\text{a}} = 1$).

The weights can be adjusted based on the reliability of each method. For instance, if the geometric-based method is more reliable, then it w_{g} might be larger than w_{a}.

If both methods are considered equally reliable, the weights might be the same, leading to the simple average:

$$T_{\text{f}}(i,j) = \frac{T_{\text{g}}(i,j) + T_{\text{a}}(i,j)}{2}.$$

In some situations, taking the weighted minimum or maximum traversability could be appropriate.

- *Minimum:* This would be a conservative approach, where you only consider a cell traversable if both methods agree that it is traversable:

  $$T_f(i,j) = \min(w_g T_g(i,j), w_a T_a(i,j)).$$

- *Maximum:* This would be an optimistic approach, where you consider a cell traversable if either method considers it traversable:

  $$T_f(i,j) = \max(w_g T_g(i,j), w_a T_a(i,j)).$$

3 | Result and Discussion

3.1 | Semantic-Segmentation-Based Terrain Classification

We trained the data set in 200 epochs under the aforementioned conditions using the 18 NN models. The ResNet-152 model was employed as a feature extractor in this study due to its high accuracy, efficient training time, and moderate computational demand. The data set was partitioned into 70% for training, 15% for validation, and 15% for testing.

The results for the initial data set (consisting of six pre-existing data sets and our CranfieldTerra data set) and the expanded data set (which includes the six pre-existing data sets, CranfieldTerra, and an additional, newly collected data set) are presented in Tables 3–6.

As demonstrated in the tables, the accuracy rates for *Objects*, *Persons*, and *Mud* areas have significantly increased. This improvement is attributed to the increased number of classes in the data set. The accuracy rates and response times for the FRRN-B model (which had the highest accuracy rate) and the DenseASPP model (which had the best response time) during real-time testing are detailed in Table 2. We selected the FRRN-B model for dense environments and the DenseASPP model for sparse environments during real-time testing. This approach provided higher accuracy rates for safe driving in dense environments, such as narrow forest paths, and faster response times in sparse environments, such as large grassy areas.

3.2 | Elevation Mapping

The Elevation mapping module is generated from LiDAR and stereo cameras. Figure 5 demonstrates real-time elevation mapping for Velodyne VLP-16 LiDAR, and Figure 6 is for stereo cameras in different positions. The real-time elevation mapping from LiDAR and stereo cameras from different scenes yielded promising outcomes. The system effectively processed to generate high-resolution elevation maps with accurate terrain information. The following key results were observed:

- *Accuracy:* The elevation mapping algorithm proved a high level of accuracy, capturing the terrain height information with minimal errors. The LiDAR-based maps were optimal between 0.3 and 20 m. While stereo cameras were effective between 0.1 and 2 m. The generated elevation maps closely matched the real data, as confirmed by comparison with the elevation measurements collected by observers.
- *Real-time Performance and Visualization:* The system accomplished real-time performance by efficiently processing the captured LiDAR and camera data and producing real-time elevation maps. This allowed geometric-based analysis of the terrain. However, the rate of updates was restricted by computational power. Also, the generated elevation maps were attractive and offered important insights into the terrain.
- *Maps from Different Sensors:* Combining LiDAR and camera-based maps enabled more precise mapping and the detection of a diverse spectrum of terrain.
- *Variability of Scenes:* The map worked effectively in different scenarios, including various terrains, vegetation types, and environmental conditions.
- *Adaptability:* The system demonstrated adaptability to multiple sensor installations and configurations, allowing integration with various range sensors, including LiDAR, Radar, and stereo cameras. This adaptability made it appropriate for a vast array of applications and platform types, including robots with legs and wheels.

3.3 | Generating Geometric Features

The terrain's roughness, height difference, and slope were independently derived from LiDAR and camera data, as previously described. While these three features were generated from the same elevation maps, their effectiveness varied depending on the terrain perspective. The slope feature performed better on highly rough terrains, while the height difference feature performed better in a forest environment. Therefore, the roughness, slope, and step height features were combined to produce a more precise map of traversability. The system processed data from multiple sensors to generate precise maps of terrain traversability. The following significant outcomes were observed:

- *Extensive Terrain Traversability Mapping:* The combination of roughness, slope, and step height features produced highly accurate maps of terrain traversability. The system successfully collected and merged data from multiple sources, allowing for an exhaustive understanding of the terrain.
- *Real-Time Mapping:* Real-time performance was demonstrated by the system's processing of sensor data and

TABLE 3 | Performance comparison of semantic segmentation models across initial and extended data sets.

	Train time (100 epochs)		Average accuracy		Precision		Recall		*F*1 score		IoU score	
Network models	**Initial**	**Extended**	**Initial**	**Extended**	**Initial**	**Extended**	**Initial**	**Extended**	**Initial**	**Extended**	**Initial**	**Extended**
FRRN-B	21:06:11	24:45:04	90.1	90.81	91.6	91.77	90.9	91.42	90.8	91.33	68.8	71.10
FRRN-A	20:33:44	23:55:20	89.9	90.33	90.3	90.69	89.9	90.29	89.5	89.88	68.7	70.44
BiSeNet	15:51:44	19:27:44	89.7	90.60	90.1	91.00	89.7	90.60	89.2	90.09	68.5	71.24
ERFNet	13:46:19	17:05:02	88.6	93.92	89.5	94.87	89.1	94.45	88.4	93.70	67.9	74.01
Encoder-Decoder with skip	15:04:25	18:33:13	90.0	91.80	90.0	91.80	90.0	91.80	89.7	91.49	66.6	69.93
MobileUNet-Skip	29:40:48	33:02:24	87.8	88.68	87.6	88.48	87.8	88.68	87.1	87.97	64.2	67.41
Encoder-Decoder	14:58:23	18:27:11	89.2	92.77	89.6	93.18	89.2	92.77	88.8	92.35	64.1	67.95
PSPNet	13:39:55	17:10:09	88.2	90.85	89.5	92.19	88.2	90.85	88.1	90.74	63.4	67.84
AdapNet	12:45:16	16:19:50	87.7	89.45	88.8	90.58	87.7	89.45	87.5	89.25	62.1	64.58
GCN	16:32:23	20:04:04	87.1	90.58	87.2	90.69	87.2	90.69	86.3	89.75	61.8	66.74
DenseASPP	7:54:34	11:30:34	87.24	87.30	87.92	88.10	86.94	87.30	85.9	86.70	60.70	61.20
FC-DenseNet56	20:21:55	23:56:29	88.3	91.83	89.3	92.87	88.3	91.83	88.1	91.62	61.4	65.08
Mobile-Unet	29:53:01	33:16:03	87.1	92.33	88.5	93.81	87.3	92.54	86.9	92.11	60.5	65.95
DeepLaB-v3+	12:28:15	16:05:41	86.7	91.04	87.4	91.77	86.7	91.04	85.9	90.20	60.5	64.74
Refinet	18:32:20	21:59:42	86.1	88.68	86.9	89.51	86.1	88.68	85.4	87.96	59.1	60.87
FC-DenseNet103	26:36:08	30:10:42	86.9	89.51	88.1	90.74	86.9	89.51	86.4	88.99	58.7	62.22
FC-DenseNet67	23:31:42	26:53:18	87.1	88.84	88.3	90.07	87.1	88.84	86.9	88.64	58.6	61.53
DeepLaB-v3	12:21:11	15:39:54	81.4	84.66	83.8	87.15	81.4	84.66	80.1	83.30	53.0	57.24

Abbreviation: IoU, Intersection over Union.

TABLE 4 | Detailed comparison of class-specific metrics in semantic segmentation models across initial and extended data sets (for objects, person, and mud).

Network models	Objects (initial)	Objects (extended)	Person (initial)	Person (extended)	Mud (initial)	Mud (extended)
FRRN-B	81.3	85.37	82.3	87.24	66.9	78.60
FRRN-A	81.3	84.55	83.6	86.11	67.9	78.09
BiSeNet	82.8	86.11	85.6	89.88	68.8	79.12
ERFNet	79.8	84.59	83.4	86.74	67.9	79.44
Encoder-Decoder with skip	77.6	79.15	82.1	84.56	63.9	74.12
MobileUNet-Skip	79.7	81.29	82.9	84.56	62.5	71.88
Encoder-Decoder	73.9	76.86	79.3	82.47	66.8	76.15
PSPNet	74.6	78.33	79.3	80.89	61.8	69.22
AdapNet	71.4	73.54	79.3	81.68	56.2	62.38
GCN	76.3	79.35	80.1	83.30	65.1	74.87
DenseASPP	62.3	64.79	80.8	84.03	62.0	71.92
FC-DenseNet56	69.9	73.40	80.5	82.92	69.2	79.58
Mobile-Unet	73.7	78.12	79.4	84.16	62.7	72.11
DeepLaB-v3+	69.8	73.99	79.8	81.40	59.8	58.01
Refinet	74.4	78.86	79.3	84.06	58.4	67.74
FC-DenseNet103	68.9	72.35	74.3	77.27	59.8	69.97
FC-DenseNet67	74.1	76.32	79.4	80.19	64.1	75.00
DeepLaB-v3	64.4	66.98	79.5	81.09	61.4	72.45

generation of traversability maps. An overall geometric-based traversability map was created using data from four sensors (one LiDAR and three cameras). Thus, a comprehensive mapping of traversability based on geometry was obtained.

- *Multisensor Fusion:* By integrating data from multiple sensors such as cameras and LiDAR, the system was able to generate a comprehensive representation of the terrain's characteristics. The combination of roughness, slope, and step height features from both cameras and LiDAR allowed for a more robust and precise evaluation of terrain traversability.

These results demonstrate the efficacy of a real-time, geometric-based approach to terrain traversability mapping that combines roughness, slope, and step height features from multiple sensors. The accurate and expeditious evaluation of terrain traversability contributes to safer and more efficient navigation for vehicles and machines operating on demanding terrains. The fused maps from LiDAR and cameras can be seen in Figure 7.

3.4 | Combination of Appearance and Geometric-Based Maps

The semantic segmentation-based terrain classification and geometric-based mapping were explained in Sections 2.1 and 2.3.1. Also, fusing these two maps was demonstrated in Section 2.3.2. The combination of appearance and geometric-based traversability maps provided significant results for evaluating the suitability of the terrain for navigation. The system provided a comprehensive comprehension of traversable terrain by combining information from both appearance-based and geometric-based approaches. The following significant outcomes were observed:

- *More Robust Traversability Map:* Combining the appearance and geometric features of terrains provided an effective representation of the terrain. Each grid in the overall TTA map included a score from 0 to 1 delivered from features of roughness, height difference, slope, and terrain classes. Here, 0 means untraversable and 1 is fully traversable. With our highly accurate semantic-based terrain classification model and geometric-based map using both LiDAR and stereo cameras, a very comprehensive and accurate model was obtained.
- *Multifunctional:* With this method, even in the event of a delay or absence of LiDAR or camera data, the generated map consistently provides reliable and safe results. For instance, if there is a lack of data from the LiDAR sensor, the geometric-based map seamlessly incorporates information solely from the stereo camera. Similarly, if there is a failure to receive data from the camera, the geometry-based map functions independently to ensure continuous operation.

TABLE 5 | Detailed comparison of class-specific metrics in semantic segmentation models across initial and extended data sets (for sky, vegetation, pavement, vehicle, and house).

Network models	Sky (initial)	Sky (extended)	Vegetation (initial)	Vegetation (extended)	Pavement (initial)	Pavement (extended)	Vehicle (initial)	Vehicle (extended)	House (initial)	House (extended)
FRRN-B	90.8	91.71	88.7	89.59	93.6	94.54	83.6	82.76	88.5	89.39
FRRN-A	91.2	90.29	84.2	83.36	93.9	92.96	86.5	83.91	89.6	88.70
BiSeNet	89.8	90.70	85.3	86.15	93.7	94.64	89.2	89.20	87.9	89.66
ERFNet	90.1	88.30	84.2	82.52	92.4	90.55	84.5	83.66	87.1	84.49
Encoder-Decoder with skip	90.6	92.41	84.2	85.88	93.1	94.96	86.6	85.73	73.8	75.28
MobileUNet-Skip	90.4	92.21	81.1	81.10	93.1	94.96	83.1	81.60	85.5	86.36
Encoder-Decoder	89.9	90.80	77.7	78.48	92.6	90.75	79.6	74.82	85.9	87.62
PSPNet	91.1	91.10	83.0	83.00	88.2	88.20	79.8	79.00	81.1	81.91
AdapNet	92.7	89.92	77.6	75.27	90.5	87.79	84.5	81.97	84.4	81.87
GCN	89.7	87.91	81.9	80.26	90.1	88.30	78.5	75.36	78.8	77.22
DenseASPP	86.5	88.23	90.1	91.90	89.9	91.70	79.5	78.71	78.4	79.97
FC-DenseNet56	92.3	95.07	79.9	82.30	91.4	92.31	73.5	72.03	78.8	79.59
Mobile-Unet	87.4	83.90	80.8	77.57	89.9	86.30	83.6	80.26	86.9	83.42
DeepLaB-v3+	92.3	89.53	74.9	72.65	92.0	89.24	78.1	76.54	87.6	86.72
Refinet	91.2	89.38	75.9	74.38	89.7	87.01	82.9	81.24	75.6	74.09
FC-DenseNet103	87.6	88.48	74.5	75.25	91.2	92.11	74.1	73.36	81.1	84.34
FC-DenseNet67	89.6	89.60	77.1	77.18	90.2	91.10	71.7	70.27	72.5	73.95
DeepLaB-v3	82.7	83.53	88.4	89.28	90.7	92.51	72.1	71.38	68.5	69.19

TABLE 6 | Detailed comparison of class-specific metrics in semantic segmentation models across initial and extended data sets (for asphalt road, smooth road, rough road, very rough road, and low traversable grass).

Network models	Asphalt road (initial)	Asphalt road (extended)	Smooth road (initial)	Smooth road (extended)	Rough road (initial)	Rough road (extended)	Very rough road (initial)	Very rough road (extended)	Low traversable grass (initial)	Low traversable grass (extended)
FRRN-B	95.4	93.49	93.9	94.84	86.7	87.57	93.4	94.33	85.4	87.11
FRRN-A	95.3	92.44	92.1	91.18	86.5	85.64	93.2	92.27	86.1	86.10
BiSeNet	95.5	94.55	82.2	83.02	85.1	85.95	89.1	89.99	87.9	88.78
ERFNet	95.1	93.20	84.6	82.91	84.6	82.91	89.0	87.22	88.9	88.90
Encoder-Decoder with skip	94.4	89.68	97.3	99.25	86.9	86.90	87.8	89.56	82.3	85.59
MobileUNet-Skip	94.3	93.36	92.9	94.76	82.5	84.15	87.2	88.07	85.7	87.41
Encoder-Decoder	94.5	92.61	93.6	94.54	85.4	86.25	90.6	91.51	83.9	84.74
PSPNet	94.9	93.95	91.7	91.70	83.8	83.80	91.1	92.92	84.1	83.26
AdapNet	91.3	88.56	82.4	79.93	84.8	82.26	91.2	88.46	81.7	79.25
GCN	94.2	92.32	91.5	89.67	81.5	79.87	89.0	87.22	80.5	78.89
DenseASPP	94.8	89.11	94.4	96.29	83.3	84.97	90.3	92.11	80.8	82.42
FC-DenseNet56	94.3	90.53	91.7	95.37	81.9	84.36	89.9	92.60	80.2	82.61
Mobile-Unet	94.9	91.10	84.8	81.41	86.7	83.23	88.6	85.06	79.6	76.42
DeepLaB-v3+	91.1	88.37	93.7	90.89	78.8	76.44	86.2	83.61	88.9	86.23
Refinet	94.1	91.28	89.0	87.22	81.8	80.16	86.5	84.77	83.5	82.67
FC-DenseNet103	92.9	92.90	92.7	93.63	76.9	77.67	89.1	89.99	89.2	90.09
FC-DenseNet67	93.2	87.61	88.5	86.73	83.1	81.44	94.7	91.86	87.6	91.98
DeepLaB-v3	90.5	85.98	84.6	85.45	85.1	85.95	85.8	86.66	81.9	82.72

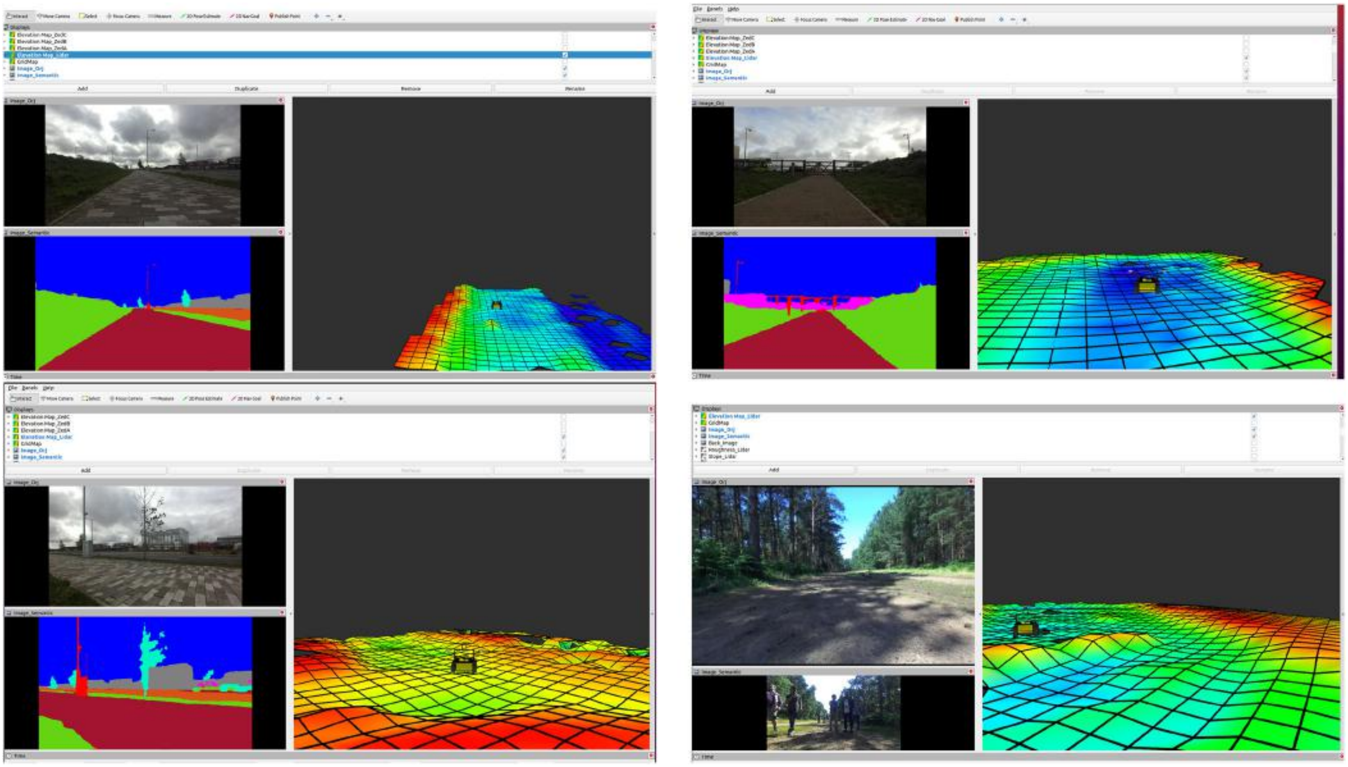
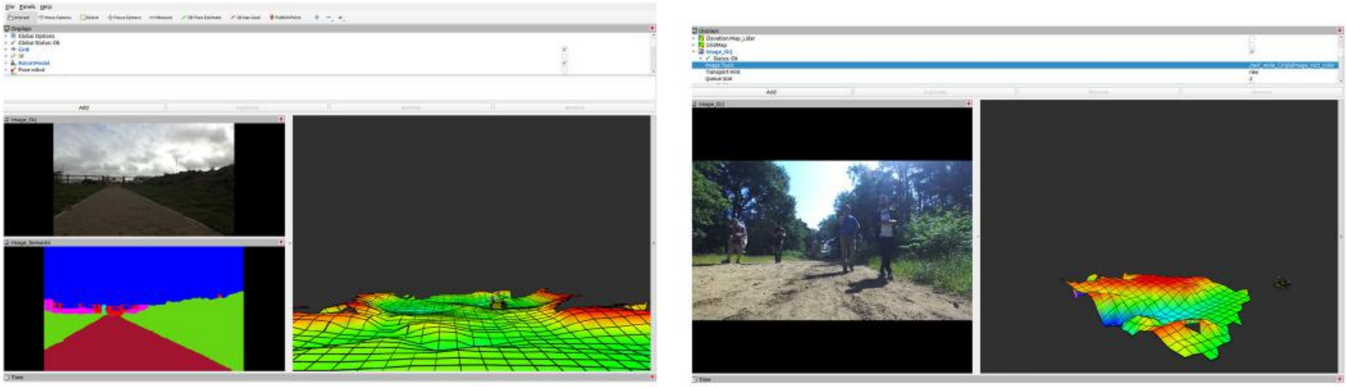
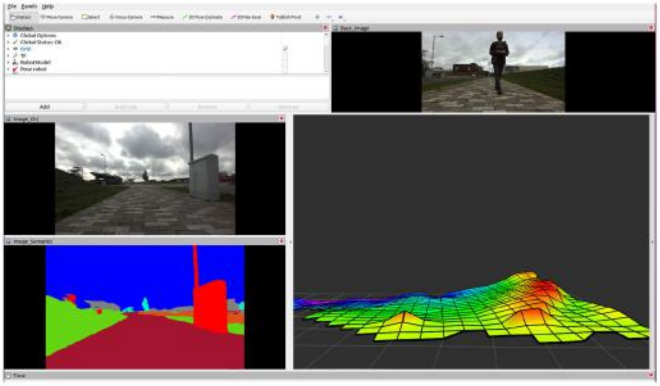


FIGURE 5 | Real-time elevation mapping from LiDAR captured different scenes. LiDAR, light detection and ranging. [Color figure can be viewed at wileyonlinelibrary.com]



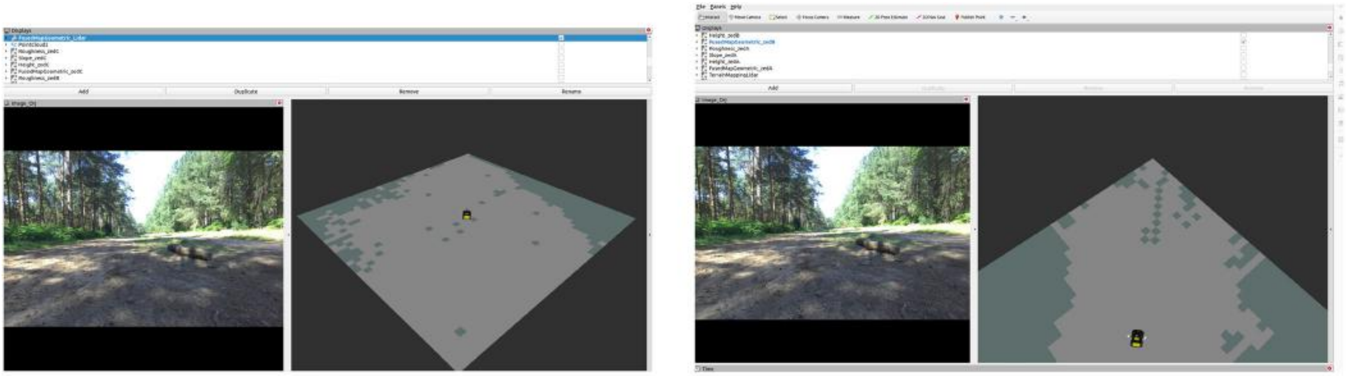
(a) Elevation Map from Front Stereo Camera

(b) Elevation Map from Front Right Camera

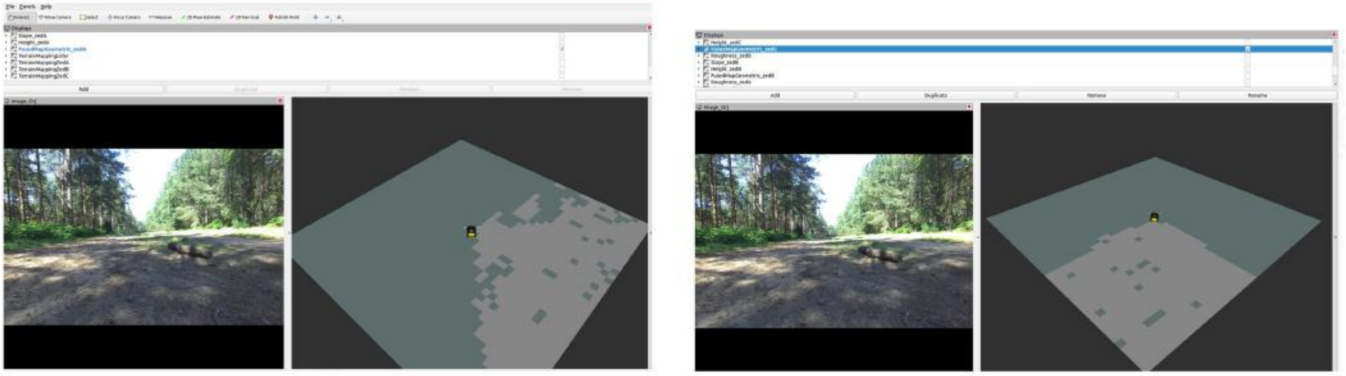


(c) Elevation Map from Rear Stereo Camera

FIGURE 6 | Real-time elevation mapping from Zed-2 stereo cameras captured different scenes. (a) Elevation map from front stereo camera, (b) elevation map from front right camera, and (c) elevation map from rear stereo camera. [Color figure can be viewed at wileyonlinelibrary.com]

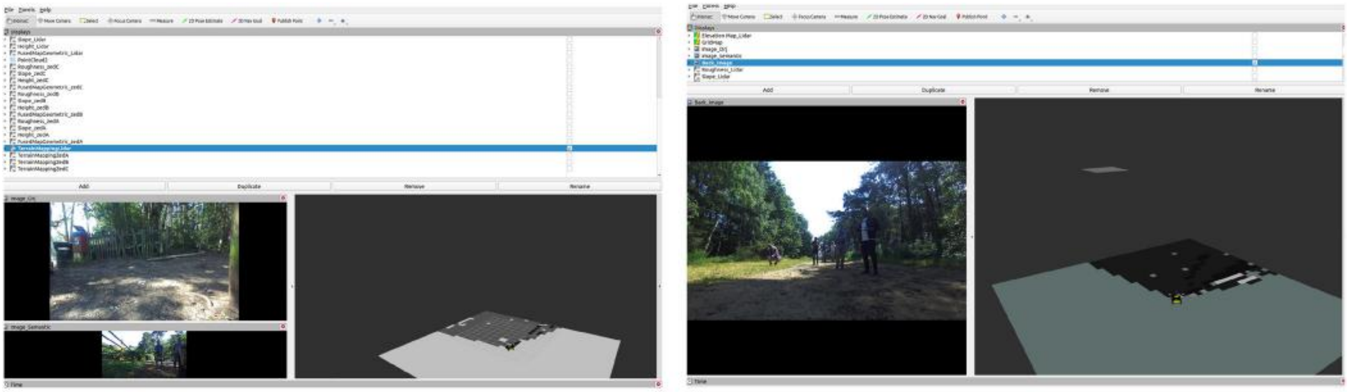


(a) Geometric-Based Map from Lidar (b) Geometric-Based Map from Front Stereo Camera



(c) Geometric-Based Map from Right Stereo Camera (d) Geometric-Based Map from Rear Stereo Camera

FIGURE 7 | Real-time geometric-based terrain traversability mapping fused roughness, slope, and step height features from different sensors. (a) Geometric-based map from LiDAR, (c) geometric-based map from right stereo camera, (b) geometric-based map from front stereo camera, and (d) geometric-based map from rear stereo camera. LiDAR, light detection and ranging. [Color figure can be viewed at wileyonlinelibrary.com]



(a) Overall Traversability Map From Lidar (b) Overall Traversability Map From Back Camera

FIGURE 8 | Real-time overall terrain traversability mapping fused geometric and appearance features of terrain. (a) Overall traversability map from LiDAR and (b) overall traversability map from back camera. LiDAR, light detection and ranging. [Color figure can be viewed at wileyonlinelibrary.com]

Figure 9 shows the traversability distance points of the semantic classes to represent a 2.5D map from the semantic segmentation model, and Figure 8a,b show some overall TTA mapping examples during different scenarios.

Table 7 reports the mean absolute error (MAE) of step height, roughness, and slope with respect to ground-truth measurements, stratified by range from the robot. Results show that in the near-field (0–0.5 m), the fused elevation map consistently reduces error compared with LiDAR-only mapping: step height MAE decreases from 1.20 to 0.62 cm (−48%), roughness MAE from 0.80 to 0.34 cm (−58%), and slope MAE from 1.00° to 0.50° (−50%). Beyond 0.5 m, both methods show nearly identical performance, with differences

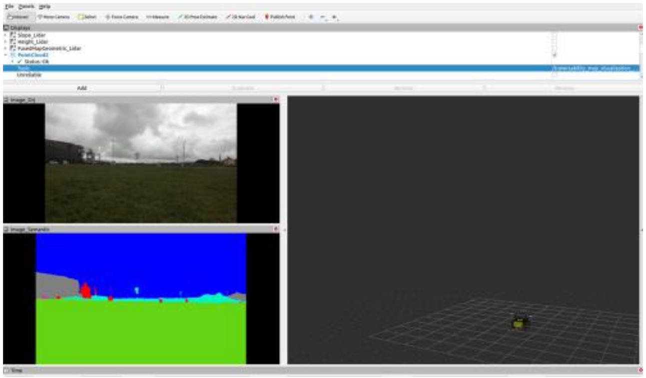


FIGURE 9 | Real-time semantic segmentation-based terrain classification and traversability cloud for representing the semantic segmentation model to 2.5D map. [Color figure can be viewed at wileyonlinelibrary.com]

below 0.1 across all metrics. This confirms that the fusion mainly improves close-range perception, where the combined use of stereo geometry and LiDAR is most effective, due to the LiDAR's field of view and its mounting position on the platform.

Table 8 provides the raw metric values for representative terrain patches (flat asphalt, dirt with ruts, gravel, vegetation, and discrete steps) in both near- and far-field regions. In the near-field, fused maps are closer to ground-truth values, particularly for small discontinuities (e.g., rut depths, vegetation returns, and step edges), which LiDAR-only maps tend to overestimate due to noise and occlusion. In the far-field, LiDAR and fused maps converge to nearly identical values, as expected given stereo disparity limitations and LiDAR dominance at longer distances.

TABLE 7 | Range-binned mean absolute error (MAE) of traversability metrics.

Range (m)	Metric	LiDAR MAE	Fused MAE	Δ	Reliable change (%)
0–0.5	Step height (cm)	1.20	0.62	−0.58	−48.3
	Roughness (cm RMS)	0.80	0.34	−0.46	−57.5
	Slope (deg)	1.00	0.50	−0.50	−50.0
0.5–1	Step height (cm)	0.80	0.70	−0.10	−12.5
	Roughness (cm RMS)	0.55	0.50	−0.05	−9.1
	Slope (deg)	0.70	0.65	−0.05	−7.1
1–2	Step height (cm)	0.70	0.68	−0.02	−2.9
	Roughness (cm RMS)	0.50	0.48	−0.02	−4.0
	Slope (deg)	0.60	0.58	−0.02	−3.3
2–5	Step height (cm)	0.65	0.64	−0.01	−1.5
	Roughness (cm RMS)	0.45	0.44	−0.01	−2.2
	Slope (deg)	0.55	0.54	−0.01	−1.8

Abbreviations: LiDAR, light detection and ranging; RMS, root mean square.

TABLE 8 | Ground truth versus LiDAR versus fused values for near-field (0–0.5 m) and far-field (≥ 0.5 m).

	Step (cm)			Roughness (cm)			Slope (deg)		
Patch/site	Ground	LiDAR	Fused	Ground	LiDAR	Fused	Ground	LiDAR	Fused
Near-field (0–0.5 m)									
Flat asphalt	0.0	0.6	0.2	0.3	0.9	0.4	0.5	1.4	0.7
Packed dirt (ruts)	1.0	1.8	1.2	1.2	1.9	1.4	2.0	2.9	2.3
Gravel	2.0	3.1	2.4	2.5	3.6	2.8	4.0	4.9	4.4
Tufted grass	3.0	4.2	3.1	3.0	4.4	3.2	3.0	3.9	3.3
Ramp edge (10 cm)	10.0	11.2	10.5	0.5	0.9	0.6	5.0	6.0	5.4
Stair step (15 cm)	15.0	16.5	15.6	0.6	1.0	0.7	0.0	0.8	0.3
Far-field (≥ 0.5 m)									
Flat asphalt	0.0	0.3	0.2	0.3	0.6	0.5	0.5	0.9	0.8
Packed dirt (ruts)	1.0	1.3	1.2	1.2	1.5	1.4	2.0	2.5	2.4
Gravel	2.0	2.3	2.2	2.5	2.9	2.8	4.0	4.5	4.4
Tufted grass	3.0	3.2	3.1	3.0	3.4	3.3	3.0	3.5	3.4
Ramp edge (10 cm)	10.0	10.6	10.5	0.5	0.7	0.6	5.0	5.6	5.5
Stair step (15 cm)	15.0	15.3	15.2	0.6	0.8	0.7	0.0	0.4	0.3

Abbreviation: LiDAR, light detection and ranging.

As a result, these tables quantitatively demonstrate the main technical advantage of our method: near-field accuracy is substantially improved by LiDAR-stereo fusion, while far-field accuracy is preserved, directly supporting traversability estimation modules that rely on step height, roughness, and slope.

4 | Conclusion

In this paper, we proposed a novel approach to terrain traversability analysis using a combination of appearance-based and geometric-based methods. Our system integrates multiple sensor inputs, including stereo cameras, LiDAR, and IMU, to provide real-time terrain classification and traversability evaluation. This approach offers significant advancements in autonomous navigation, particularly in environments with varying terrain types and complexity.

Compared with traditional methods that rely solely on either appearance-based or geometric-based analysis, our hybrid approach leverages the strengths of both techniques, enabling more accurate and robust terrain classification. While appearance-based methods are often effective in distinguishing terrain types, they can struggle in environments with low contrast or obscured features. On the other hand, geometric-based methods provide valuable depth information but may not be sufficient on their own for classifying complex terrain types. Our fusion of both methods ensures a comprehensive understanding of the terrain, leading to more accurate and reliable traversability maps.

Furthermore, our method includes the introduction of the *Real-Time Switch Model*, which dynamically adjusts the terrain classification model based on environmental density. This dynamic switching ensures that the system maintains high accuracy while optimizing computational efficiency, providing a balance that is essential for real-time applications. The evaluation results demonstrate that our method outperforms traditional models in terms of both accuracy and response time, particularly in complex and dynamic environments.

Additionally, the CranfieldTerra data set, introduced in this paper, provides a valuable resource for training and validating terrain classification models. With its diverse range of terrain types and real-world sensor data, it serves as a comprehensive benchmark for future research in autonomous navigation.

In conclusion, the proposed method offers a superior solution to terrain traversability analysis by combining appearance and geometric information, incorporating real-time dynamic model selection, and providing a reliable framework for real-world autonomous navigation applications. These innovations represent a significant step forward in improving the efficiency, accuracy, and adaptability of autonomous systems operating in diverse and challenging environments.

Author Contributions

Semih Beycimen: conceptualization, methodology, writing – original draft. **Dmitry Ignatyev:** conceptualization, manuscript revision, supervision. **Argyrios Zolotas:** conceptualization, methodology, manuscript revision, supervision.

Acknowledgments

The first author acknowledges the Republic of Turkey, Ministry of National Education (YLYS), for supporting the studies under PhD scholarship ref. U9BYTAB2LDGA7LK.

Data Availability Statement

The data that support the findings of this study are available from the corresponding author upon reasonable request.

References

Beycimen, S., A. Art, M. M. Unal, D. Ignatyev, and A. Zolotas. 2025. *On Real-Time Semantic Segmentation With Comprehensive Off-Road Datasets for Enhanced Terrain Classification*. Engineering Applications of Artificial Intelligence.

Beycimen, S., D. Ignatyev, and A. Zolotas. 2022. “A Comparison Study of Neural Network-Based Semantic Segmentation for Off-Road Traversability.” In *UKRAS22 Conference*, 72–73.

Deng, J., W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. 2009. “ImageNet: A Large-Scale Hierarchical Image Database.” In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, 248–255. IEEE.

Deng, W., K. Huang, and X. Chen, et al. 2020. “RGB-D Based Semantic SLAM Framework for Rescue Robot.” In *Proceedings of the 2020 Chinese Automation Congress, CAC 2020*, 6023–6028. https://doi.org/10.1109/CAC51589.2020.9326866.

Fankhauser, P., M. Bloesch, and M. Hutter. 2018. “Probabilistic Terrain Mapping for Mobile Robots With Uncertain Localization.” *IEEE Robotics and Automation Letters* 3, no. 4: 3019–3026.

Fankhauser, P., and M. Hutter. 2016. “A Universal Grid Map Library: Implementation and Use Case for Rough Terrain Navigation.” In *Robot Operating System (ROS)—The Complete Reference*, 1, 99–120.

Forsyth, D. A., and J. Ponce. 2002. *Computer Vision: A Modern Approach*. Prentice Hall Professional Technical Reference.

Hedrick, G., N. Ohi, and Y. Gu. 2020. “Terrain-Aware Path Planning and Map Update for Mars Sample Return Mission.” *IEEE Robotics and Automation Letters* 5, no. 4: 5181–5188. https://doi.org/10.1109/LRA.2020.3005123.

Jiang, P., P. Osteen, M. Wigness, and S. Saripalli. 2021. “Rellis-3D Dataset: Data, Benchmarks and Analysis.” In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, 1110–1116. IEEE.

Leung, T. H. Y., D. Ignatyev, and A. Zolotas. 2022. “Hybrid Terrain Traversability Analysis in Off-Road Environments.” In *2022 8th International Conference on Automation, Robotics and Applications (ICARA)*, 50–56. IEEE.

Maturana, D., P.-W. Chou, M. Uenoyama, and S. Scherer. 2018. “Real-Time Semantic Mapping for Autonomous Off-Road Navigation.” In *Field and Service Robotics*, 335–350. Springer.

Meng, X., Z. Cao, S. Liang, L. Pang, S. Wang, and C. Zhou. 2018. “A Terrain Description Method for Traversability Analysis Based on Elevation Grid Map.” *International Journal of Advanced Robotic Systems* 15, no. 1: 1–12. https://doi.org/10.1177/1729881417751530.

Moshayedi, A. J., K. Reza, A. S. Khan, and A. Nawaz. 2023. “Integrating Virtual Reality and Robotic Operation System (ROS) for AGV Navigation.” *EAI Endorsed Transactions on AI and Robotics* 2, no. 1: e3.

Omar Chavez-Garcia, R., J. Guzzi, L. M. Gambardella, and A. Giusti. 2017. “Learning Ground Traversability From Simulations.” *IEEE Robotics and Automation Letters* 3, no. 3: 1695–1702. https://arxiv.org/abs/1709.05368.

Romera, E., J. M. Alvarez, L. M. Bergasa, and R. Arroyo. 2017. "ERFNet: Efficient Residual Factorized ConvNet for Real-Time Semantic Segmentation." *IEEE Transactions on Intelligent Transportation Systems* 19, no. 1: 263–272.

Roncancio, H., M. Becker, A. Broggi, and S. Cattani. 2014. "Traversability Analysis Using Terrain Mapping and Online-Trained Terrain Type Classifier." In *IEEE Intelligent Vehicles Symposium Proceedings*, Vol. Iv, 1239–1244. https://doi.org/10.1109/IVS.2014.6856427.

Rothrock, B., J. Papon, R. Kennedy, M. Ono, M. Heverly, and C. Cunningham. 2016. "SPOC: Deep Learning-Based Terrain Classification for Mars Rover Missions." In *AIAA Space and Astronautics Forum and Exposition, SPACE 2016 (September)*. https://doi.org/10.2514/6.2016-5539.

Scharstein, D., and R. Szeliski. 2003. "High-Accuracy Stereo Depth Maps Using Structured Light." In *Proceedings of the 2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2003*, Vol. 1, I. IEEE.

Takikawa, T., D. Acuna, V. Jampani, and S. Fidler. 2019. "Gated-SCNN: Gated Shape CNNs for Semantic Segmentation." In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 5229–5238.

Thakker, R., N. Alatur, D. D. Fan, et al. 2020. "Autonomous Off-Road Navigation Over Extreme Terrains With Perceptually-Challenging Conditions." In *International Symposium on Experimental Robotics*, 161–173. Springer.

Xu, G., A. S. Khan, A. J. Moshayedi, X. Zhang, and Y. Shuxin. 2022. "The Object Detection, Perspective and Obstacles in Robotic: A Review." *EAI Endorsed Transactions on AI and Robotics* 1, no. 1: e13.

Yen, J. Y. 1970. "An Algorithm for Finding Shortest Routes From all Source Nodes to a Given Destination in General Networks." *Quarterly of Applied Mathematics* 27, no. 4: 526–530.