Received 27 January 2024; revised 30 May 2024 and 28 August 2024; accepted 5 September 2024.
Date of publication 19 September 2024; date of current version 18 October 2024.
This article was recommended by Executive Editor Sanjiv Singh.

Digital Object Identifier 10.1109/TFR.2024.3464369

RoadRunner—Learning Traversability Estimation for Autonomous Off-Road Driving

JONAS FREY[1,2] (Student Member, IEEE), MANTHAN PATEL[1,2] (Student Member, IEEE), DEEGAN ATHA[1], JULIAN NUBERT[1,2] (Student Member, IEEE), DAVID FAN[1] (Member, IEEE), ALI AGHA[1] (Member, IEEE), CURTIS PADGETT[1], PATRICK SPIELER[1] (Member, IEEE), MARCO HUTTER[2] (Member, IEEE), AND SHEHRYAR KHATTAK[1] (Member, IEEE)

[1]Jet Propulsion Laboratory (JPL), California Institute of Technology (Caltech), Pasadena, CA 91011 USA
[2]Robotic Systems Laboratory, Swiss Federal Institute of Technology (ETH Zürich), 8092 Zürich, Switzerland

CORRESPONDING AUTHOR: JONAS FREY (jonfrey@ethz.ch)

This work was supported in part by the Jet Propulsion Laboratory, California Institute of Technology, through the National Aeronautics and Space Administration under Grant 80NM0018D0004; in part by the Defense Advanced Research Projects Agency (DARPA); and in part by the Swiss Federal Institute of Technology (ETH Zürich). The work of Jonas Frey and Julian Nubert was supported by the Max Planck ETH Center for Learning Systems.

(Regular Article)

ABSTRACT Autonomous navigation at high speeds in off-road environments necessitates robots to comprehensively understand their surroundings using onboard sensing only. The extreme conditions posed by the off-road setting can cause degraded image quality as well as limited sparse geometric information available from light detection and ranging (LiDAR) sensing when driving at high speeds. In this work, we present RoadRunner, a novel framework capable of predicting terrain traversability and elevation directly from camera and LiDAR sensor inputs. RoadRunner enables reliable autonomous navigation by fusing sensory information and generates contextually informed predictions about the geometry and traversability of the terrain while operating at low latency. In contrast to existing methods, which rely on classifying handcrafted semantic classes and using heuristics to predict traversability costs, our method directly predicts traversability. It is trained on labels that can be automatically generated in hindsight in a self-supervised fashion. The RoadRunner network architecture builds upon advances from the autonomous driving domain, which allow us to embed LiDAR and camera information into a common bird's eye-view perspective. Training is enabled by utilizing an existing traversability estimation stack to generate training data in hindsight in a scalable manner from real-world off-road driving datasets. Furthermore, RoadRunner improves the system latency by a factor of ∼4, from 500 to 140 ms, while improving the accuracy for traversability costs and elevation map predictions. We demonstrate the effectiveness of RoadRunner in enabling safe and reliable off-road navigation at high speeds in multiple real-world driving scenarios through unstructured desert environments.

INDEX TERMS Deep learning for visual perception, field robots.

I. INTRODUCTION

ANIMALS, such as the greater roadrunner, are capable of traversing complex off-road terrains at an impressive running speed of 24–32 km/h [1]. This capability makes the roadrunner a role model in terms of perception capabilities required for autonomous ground vehicles to advance toward enabling high-speed operations in unstructured off-road environments. One of the key components to facilitate safety while driving at high speeds is the assessment of where a robot can navigate safely. This assessment is known as *traversability estimation*, where the objective is to estimate the affordance or risk the robot must take when navigating the perceived terrain. The affordance may depend on the robot's hardware, control system, terrain geometry, physical terrain properties, and other factors.

Complex and unstructured off-road environments pose a variety of obstacles and other potential risks, which need to be correctly assessed by an autonomous ground vehicle.

FIGURE 1. Example deployment environments showcasing high-speed off-road navigation.

In addition, short reaction times are required at high speeds and potential hazards must be identified at long distances to guarantee safety. Moreover, only partial observations of the environment are available, given that onboard sensors such as LiDARs and cameras can only provide information at a limited update rate and only perceive a restricted field of view with increasing intrinsic sparsity at longer distances. In addition, when considering driving off-road, robots cannot rely on high-definition map information, given that the robot may be deployed for the first time in the environment (space exploration) or the environment might have changed significantly due to seasonal variations. Furthermore, extreme conditions, such as dust, dirt, fog, and rain, are common in an off-road setting and can lead to degraded perception modalities, as shown in the case of LiDAR sensors [2]. As a result, the traversability estimation system has to cope with all of these challenges to allow for navigation in an off-road setting at high speeds.

In comparison, self-driving cars operate at higher maximum speeds and in more dynamic environments. Despite this, the terrain traversability estimation remains simple as cars primarily drive on paved roads. Therefore, the classification of scene semantics is mostly sufficient to conclude the traversability. As a result, most works focus on identifying semantic classes such as lanes, cars, and traffic signs from images [3] or LiDAR data [4]. Despite the difference in the use case, self-driving cars are equipped with a similar sensing setting, and our work is inspired by the recent advances in multimodal network architectures that fuse different sensory information into a common bird's eye view (BEV) perspective [3], [5].

Historically, the first progress toward autonomous navigation in unstructured off-road environments was achieved throughout the DEMO III [6], PerceptOR [7], and learning applied to ground vehicles (LAGR) [8], [9] programs, as well as the DARPA Grand Challenge. As part of these programs, multiple works pioneered traversability estimation from camera and LiDAR data and applied learning-based approaches, as well as, nonlearning-based approaches, which rely on heuristics to assess the terrain traversability. More recently, using simulation-based approaches [10], [11], which can evaluate the terrain with respect to the specific robot's capabilities, and self-supervised approaches, which make use of real-world deployment data [12] have gained significant popularity, for estimating traversability in an off-road setting.

To further enhance the off-road traversability estimations, we present RoadRunner: a multicamera multi-LiDAR learning architecture able to predict terrain traversability and elevation with low latency directly from sensor inputs. Our architecture leverages point cloud and pretrained image segmentation backbones and fuses information into a unified BEV representation. A convolutional neural network (CNN) predicts a robot-centric traversability and elevation map from the unified feature embedding. The training data are generated in a self-supervised manner using hindsight to determine the traversability and the geometry of the terrain without any human labeling or explicit identification of semantic classes during run time. The concept of hindsight refers to the fact that offline processing with future and past sensor measurements can be used to create more reliable estimates compared to the online deployment of the same algorithm. To generate the training data for RoadRunner, we use the NASA Jet Propulsion Laboratory offroad autonomy research stack X-Racer, (see Section III-C), a sophisticated off-road driving stack that includes mapping and traversability estimation, which leverages geometry and semantics with carefully crafted and field-proven heuristics. These heuristics encode the vehicle-specific capabilities. While X-Racer is not the main contribution of this article, we first provide an overview of all relevant components to this work in Section III-C. RoadRunner provides a framework for using hindsight and end-to-end learning to overcome the limitations inherent to a nonlearned traversability software stack. In addition, by predicting traversability and elevation simultaneously, RoadRunner functions as a single perception and mapping module, providing all perceptual information necessary for downstream path planning. Throughout this work, all experiments are conducted using our customized Polaris RZR S4 1000 Turbo. This side-by-side off-road vehicle functions as a testbed to enhance off-road autonomy and can be seen in Fig. 1.

The key contributions of our work are given as follows.

1) A novel RoadRunner network architecture that can simultaneously predict traversability costs and elevation map information directly from multi-LiDAR and multicamera data at low latency.

2) A general framework to generate pseudo ground truth elevation maps and traversability costs using temporal data aggregation in hindsight to train the network in a self-supervised manner.
3) We present an overview of NASA Jet Propulsion Laboratory's off-road autonomy research stack X-Racer, one of the most advanced off-road autonomy software stacks.
4) An exhaustive evaluation of the proposed RoadRunner architecture on real-world field test data and comparisons to existing network architectures as well as an ablation study.

RoadRunner outperforms the elevation mapping and traversability estimation of X-Racer, which was used to generate training data, by leveraging visual and geometric data. In addition, it is capable of predicting missing elevation and traversability information based on the learned context. The same holds for traversability detection, where obstacles can be detected at a longer range. We demonstrated that our proposed RoadRunner architecture, compared to the X-Racer stack, improves traversability cost estimation by 52.3% in mean squared error (mse) and 36.0% in mean absolute error (MAE) for elevation map estimation while reducing the perception-to-traversability latency during inference by a factor of ∼4 compared to the existing software stack.

II. RELATED WORK

In this section, we provide an overview of traversability estimation methods, focusing on off-road driving, traversability from semantics, self-supervision, and other approaches. In addition, we provide an overview of semantic segmentation methods operating in a BEV representation.

A. TRAVERSABILITY ESTIMATION FOR OFF-ROAD DRIVING

Terrain traversability is dependent upon both the terrain's geometry and its physical properties such as stiffness, friction, or granularity. Several other factors influence the traversability, which include the specific robotic system in use, the applied control strategy [12], and the robot's state during its interaction with the terrain [13]. For example, different robots (*hardware and control dependence*) can surmount distinct obstacles when driving at varying speeds (*state dependence*). In this work, we simplify this multivariant view of traversability following [13] and model the traversability as the affordance/risk required to overcome a given terrain as a probability distribution. We marginalize the robot-state dependency and target a specific robotic system and control strategy. To make the overall traversability more interpretable, we employ the conditional value at risk (CVaR) metric. The CVaR metric, which is a *coherent risk metric* being monotonic, subadditive, homogeneous, and translational invariant, allows us to emphasize different parts of the risk distribution associated with a given terrain [14].

Pomerleau [15] pioneered learning for autonomous driving by training a multilayer perceptron (MLP) to predict online the steering command from a low-resolution camera image by learning from demonstration for on-road driving end-to-end. The first steps toward off-road autonomy were achieved in the DEMO III project [6] followed by the Perceptor program [7], in which off-road capable unmanned ground vehicles (UGVs) were retrofitted with a variety of sensors, including cameras and LiDARs for cross-country navigation. The developed perception system at the time was primarily based on heuristics and, therefore, failed to generalize to the complex off-road domain. As part of this program, Manduchi et al. [16] classified terrain into a discrete set of classes from color images and used stereo depth to detect obstacles, highlighting the importance of using different sensor modalities.

The LAGR program [17] aimed to improve the perception system developed within the Perceptor program using machine learning techniques. For this, Muller et al. [18] expanded on the idea of predicting the steering command end-to-end from images [15], but instead of an MLP used a CNN in an off-road setting. For the network to learn correct driving behavior, it implicitly learned obstacle avoidance and, thereby, implicitly traversability. Hadsell et al. [19] used reliable short-range geometric measurements to supervise the training of a simple classifier to predict the traversability from image data in a self-supervised manner and proposed the concept of spatial label propagation. Kim et al. [8] instead clustered the environment and assigned positive and negative labels through interaction to determine the traversable areas. Konolige et al. [20] explicitly identified the ground plane from stereo depth and classified paths based on image segmentation and geometric information by fitting a model from a single image before each deployment.

As part of the DARPA Grand Challenge [21], Thrun et al. [22] proposed a probabilistic terrain analysis for high-speed desert driving taking into account LiDAR measurements that are used to assess the obstacles based on height difference in combination with an adaptive vision approach. In contrast, Urmson et al. [23] fully relied on geometry captured by LiDAR and radio detection and ranging (RADAR). Both teams showcased long-term autonomy in a desert scenario, within a narrow scope and perfectly pre-planned routes under GPS guidance. Our overall system does not require precise GPS guidance and can drive in more complex open-field environments at high speeds.

1) TRAVERSABILITY FROM SEMANTICS

Modern deep learning methods excel in semantic scene understanding from image [24], [25], [26], [27] and point cloud data [28]. Commonly, identified semantic classes are fused into a map representation [26], [29] and then associated with a traversability score [26] using heuristics about the traversability properties of the given semantic class. Notably, datasets, such as RUGD [30], Rellis-3D [31], or Freiburg

Forest [32], provide high-quality annotated off-road semantics and, however, are limited in size and diversity, hindering generalization to diverse off-road scenarios. Moreover, the reliance on manual labeling and heuristic-based mapping from semantics to traversability inherently limits the performance of the traversability estimation.

Within an off-road setting, Bradley et al. [33] perform a per-voxel terrain density classification from LiDAR and camera images using a random forest (RF) classifier on a diverse legged robot dataset. Maturana et al. [26] perform semantic segmentation of images and project the labels into a 2.5-D semantic elevation map for a UGV. Schilling et al. [34] utilize learned semantics segmentation features and heuristic features from LiDAR data to predict safe, risky, and obstacle regions. Viswanath et al. [25] predict semantics in image space using a CNN, while Shaban et al. [28] predict semantics in the BEV perspective using LiDAR. Shaban et al. [28] employ a sparse CNN and allow to feedforward latent features from previous predictions in combination with memory units.

Rovers used in space exploration face the same perception challenges as off-road driving, where the accurate identification of terrain hazards is pivotal to mission success [35]. As a result, the identification of semantic terrain classes and physical interaction has been exhaustively studied. Rothrock et al. [36] employed a CNN to classify Martian terrain and predict the wheel slip of the Mars Perseverance rover. Swan et al. [37] curated a comprehensive Mars terrain semantic segmentation dataset. Other works focus on improving semantic segmentation by reducing the number of training samples, given the challenges associated with collecting training data in a space exploration scenario [38], [39]. Endo et al. [40] assess risk-aware traversability costs by fusing semantic terrain classification and a slip model in a probabilistic manner.

Most relevant to our work is *TerrainNet* [41], which follows [42] by modeling the environment by a ground and ceiling layer, where each layer contains its associated semantics. The geometry and semantics are predicted by a neural network taking multiple RGB and depth camera images as input. Features from the camera are accumulated in a BEV representation by predicting the corresponding location of the feature in image space from stereo depth.

2) TRAVERSABILITY FROM SELF-SUPERVISION

Traversability estimation approaches based on scene semantics typically require expensive annotated data. Methods operating in a self-supervised manner aim to overcome this limitation by generating a training signal without relying on manual annotation. Instead, they exploit information from other sensor modalities [41], [43], [44], [45], [46], [47], [48], [49] or the interaction of the robot with the environment [12], [49], [50], [51], [52], [53], [54], [55], [56], [57]. The generated supervision signal allows training a model that predicts a look-ahead estimate of the terrain, all without requiring the robot to be near to or interact with the terrain.

Brooks and Iagnemma [43] train an image classifier to predict terrain classes of the Mars-analog environment identified by proprioception. Otsu et al. [44] use co-training and self-training to improve two classifiers to predict terrain classes from images and proprioception for space exploration. Ahtiainen et al. [52] learn traversability from LiDAR measurements by accumulating information in a map and training a support vector machine (SVM) to classify traversability supervised by positive labels the robot visited and negative labels based on heuristics. Higa et al. [47] directly estimate driving energy for rovers from images using the measured energy consumption. Castro et al. [45] adapt the proprioception-based pseudo labels by measuring vibration data, which is used as the network input to predict a traversability grid map from a colorized elevation map.

Richter and Roy [50] proposed to use anomaly detection to predict safe image regions for indoor navigation with a wheeled robot. Multiple other works integrated the concept of anomaly detection and tools available in evidential deep learning to learn from real-world data without manual labeling [12], [51], [56], [58], [59]. Contrastive learning has shown promising results in learning expressive representations that can be used for traversability estimation in a self-supervised manner [46], [55], [57].

Chen et al. [60] generate pseudo labels from heuristic-based LiDAR analysis, while Frey et al. [12] use a velocity-tracking criterion to assess terrain traversability to train a vision model. Both methods allow for training the traversability prediction network online during deployment. In WayFAST [53], the terrain traversability is approximated by the tracking error from a model predictive controller and used to predict traversability from image data. Similarly, Cai et al. [54] predict worst case expected cost and traction from a semantic elevation map in addition to predicting confidence values using density estimation. The *TerrainNet* framework [41] proposes using co-labeling of semantics by a human and a pretrained semantic network, while the geometry is supervised by LiDAR measurements.

Our method uses self-supervision, but instead of predicting semantics or relying on heuristic-based analysis, we use the sophisticated field-tested traversability estimation software stack X-Racer, which allows us to account for a variety of risks in a probabilistic manner. This allows RoadRunner to obtain a nearly dense supervision signal and account for uncertainty in the signal itself during learning, therefore overcoming problems associated with sparse labels when only considering the terrain vehicle interaction. Creating the supervision signal using hindsight imposes few constraints on specific motions to follow. This enables us to train on data collected by a human driver as well as on data collected during an autonomous mission. However, where and which training data to collect remain crucial design choices.

3) OTHER TRAVERSABILITY ESTIMATION APPROACHES

Model-based approaches for assessing geometric traversability have attracted continuous research, particularly in the

context of wheeled robots, where the wheel-to-terrain interactions can be more easily defined compared to more complex legged systems [61]. These approaches demonstrate good generalization, specifically when the underlying assumptions such as the rigidity of the environment hold. For this, recent works rely on analysis of point cloud data [13], [62], [63], mesh data [64], or elevation maps [13], [61], [65]. On the other hand, data-driven approaches use simulation to collect data in a trial-and-error fashion to estimate traversability for wheeled [10] or legged robots [11].

Complementary, learned dynamics models from real-world data are capable of accounting for unmodeled effects in simulation and are used for planning. Kahn et al. [66] and Kim et al. [67] learn a forward dynamics model, where they predict future events and states of the robot from real-world data and use a simulation to collect data, respectively. Xiao et al. [68] learn the inverse kinodynamics model of a wheeled robot from proprioception and Karnan et al. [69] expand this concept by conditioning the model on visual data of future terrain patches to anticipate terrain interaction. Therefore, these approaches implicitly learn traversability information about the environment for the deployed robotic system.

Inverse reinforcement learning (IRL) aims to learn the underlying reward structure guiding an agent's behavior [70]. By using demonstration data and the IRL formulation, Wulfmeier et al. [70] train a neural network to predict cost maps for simple game-like environments. Triest et al. [71] translated this concept to off-road autonomous driving and predicted vehicle-centric traversability grid maps.

Bouman et al. [65] assess geometric traversability based on multifidelity terrain maps and superposition of individual traversability maps. This work was expanded in [13] and [72], where traversability risk is modeled as a distribution allowing to account for different geometric risks while taking into account uncertainty. Dixit et al. [73] added further improvements for subterranean exploration and incorporated semantic risks assessed solely based on processing geometry. Our X-Racer software stack, used to generate the pseudo ground truth, adapts this perspective of traversability estimation and improves multiple components of the traversability assessment tailored to off-road driving. We explain the in-depth adaptation made to facilitate high-speed off-road navigation in Section III-C.

B. LEARNING SEMANTICS IN BEV REPRESENTATION

The mobility of ground robots is typically constrained by the terrain geometry. Under this reduced mobility assumption, it is sufficient to model the environment from a BEV perspective, in which the geometry, semantics, or traversability costs can be identified. One challenge is to associate features from a camera image into the BEV perspective. If no depth measurements are available, the inverse perspective mapping (IPM) algorithm under the flat ground assumption can project image features to the BEV perspective. Roddick et al. [74] showed that projecting multiscale image features into 3-D using an orthographic projection improves 3-D object detection performance for autonomous driving. This method projects features uniformly along a camera ray in 3-D and consecutively pools the features along the z-dimension. Similarly, *Lift Splat Shoot* [3] extends this concept to multiple cameras and shows promising results for the task of semantic segmentation in the BEV perspective. Instead of densely distributing features along the ray, they predict a categorical distribution over discrete depth bins for each pixel indicating the contribution of each feature in the intermediate 3-D representation. Pseudo-LiDAR approaches, such as *BEV-Seg* [75], predict explicit dense depth from the monocular image, which is used to establish a one-to-one correspondence between image pixels and grid cells. *Simple-BEV* [76] and *M2BEV* [77] show that while the feature lifting problem is well researched, a uniform lifting approach [74] can result in competitive performance and more gain can be achieved using RADAR sensor data [76] or a larger image backbone [77]. *BEVFusion* [5] considers the problem of fusing LiDAR data and camera images into a unified representation. Camera features are lifted into 3-D following [3], with a significantly more efficient implementation that addresses concerns in [76] and [77]. Depth features are concatenated in the BEV representation with image features, while no LiDAR measurements are leveraged for the feature lifting. *LAPTNet-FPN* [78] uses the LiDAR measurements to lift camera features into 3-D, by directly associating the sparse LiDAR measurements to the downsampled image features. Alternative attention-based lifting approaches have been investigated in [79].

III. METHOD

A. PROBLEM STATEMENT

Our RoadRunner network predicts vehicle-centric elevation and traversability cost maps using multicamera and multi-LiDAR inputs. In the following, we describe the notation (Section III-B) used throughout this article and an overview of X-Racer stack (Section III-C) used for generating training data in hindsight (Section III-D). Next, we explain the network architecture (Section III-E), followed by the implementation details (Section III-F). The overview of the RoadRunner architecture is presented in Fig. 2.

B. NOTATION

We use the following coordinate frames: world frame $\mathtt{W}$, base frame $\mathtt{B}$, gravity-compensated base frame $\mathtt{B}_g$, camera frames ${}^{k}\mathtt{C}$, and LiDAR frames ${}^{l}\mathtt{L}$, with $k \in \{1, \ldots, C\}$ and $l \in \{1, \ldots, L\}$. Here, C and L denote the number of cameras and LiDAR sensors, respectively. The world frame $\mathtt{W}$ in the following coincides with the typical odometry frame definition. Its relative transformation to the base frame $\mathtt{B}_g$ is assumed to be locally consistent and smooth. The frames are visualized in Fig. 3. Tensors are denoted by bold capital letters $\mathbf{F}_{A\times B\times C} \in \mathbb{R}^{A\times B\times C}$, where the lower right subscript indicates the respective dimensions. Camera images of height H

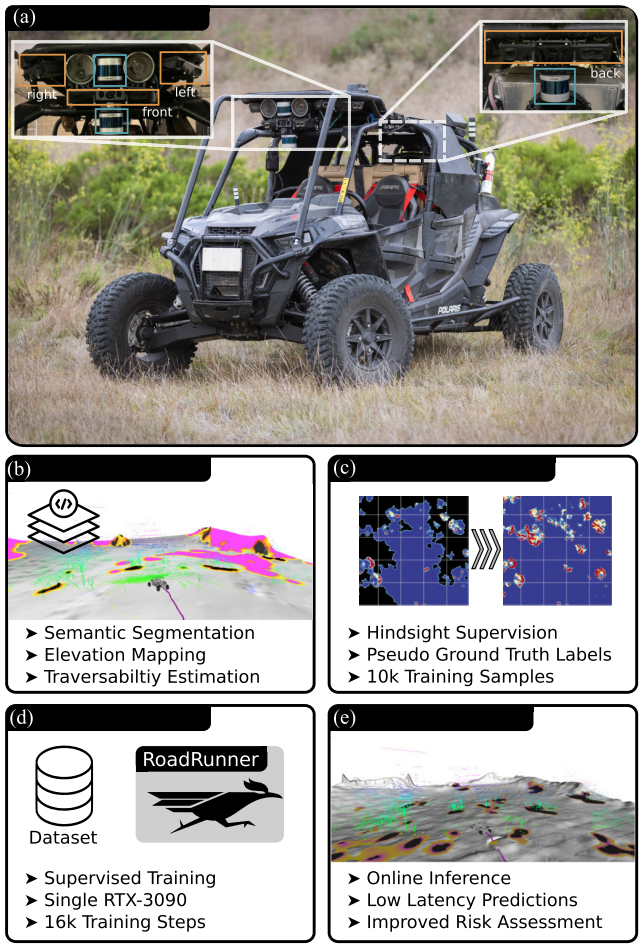

FIGURE 2. Overview of the RoadRunner architecture. RoadRunner network is trained on (a) real-world driving data, which is first processed by (b) X-Racer to generate an elevation and traversability assessment based on the currently available sensory data. (c) Pseudo ground truth labels are generated by fusing information from past and future measurements to obtain reliable traversability and elevation estimates (hindsight labeling). (d) and (e) RoadRunner network is trained offline based on the large dataset and can be deployed online for improved performance and reduced latency, respectively.

and width W are denoted as ${}^{k}\mathbf{I}^{t}_{H\times W\times 3}$, where k indicates the respective camera index and t is the timestamp. The intrinsic calibration of the camera k is given by ${}^{k}\mathbf{K}_{3\times 3}$. Similarly, the point cloud measured by the LiDAR l is denoted as ${}^{l}\mathbf{P}^{t}_{N_l\times 3}$, where N_l is the number of points and t is the respective timestamp.

We assume that the point cloud scans are motion-compensated and merged for all L LiDAR sensors, resulting in a single big point cloud $\mathbf{P}^{t}_{N\times 3}$, with $N=\sum_{l=1}^{L} N_l$. A grid map with dimension $H_g \times W_g$ is denoted as $\mathbf{G}_{H_g\times W_g}(x,y)$ and is expressed within the gravity-aligned base frame $\mathtt{B}_g$. The frame $\mathtt{B}_g$ defines the yaw orientation and center of the grid map, located at $(H_g/2, W_g/2)$. Therefore, we refer to the grid maps being position and orientation vehicle-centric.

C. X-RACER ARCHITECTURE

1) OVERVIEW

RoadRunner leverages the highly capable mapping and traversability assessment X-Racer stack for learning. X-Racer allows the Polaris RZR S4 1000 Turbo to autonomously drive off-road at up to 15 m/s within a wide range of complex off-road environments. The task at hand, which guided the development of X-Racer, is to reach a set of goal positions using onboard sensing without reliance on precise prior global mapping in a GPS-denied environment. In the following, we will summarize the most important parts of X-Racer stack relevant to the traversability assessment used for local planning. In addition, we will elaborate on the inherent limitations that are addressed by RoadRunner with respect to the existing X-Racer stack. A simplified overview of all components can be seen in Fig. 4.

2) ODOMETRY AND SEMANTICS

We assume that precise vehicle poses are provided by our LiDAR-inertial odometry estimation and localization module [86] in combination with a graph-based filtering and fusion algorithm [81]. Four RGB images captured by the Carnegie Robotics MultiSense S27 RGB-D cameras are individually processed by a segmenter vision transformer model [82] adapted and fine-tuned using [87], to predict multiclass terrain semantics, which include classes such as trail, ground, vegetation, rock, and sky. The segmentation network is trained in a supervised fashion using manually annotated images from real-world deployments and photo-realistic renderings using the Duality Robotics Falcon Pro simulator [88].

3) MAPPING

The point clouds sensed by three Velodyne VLP-32C LiDAR sensors are filtered for vehicle self-correspondences using a predefined mask, and dust points are removed using an intensity-based filtering method. The filtered point cloud is associated with predicted semantics in the image domain based on proximity using the camera intrinsics and camera-to-LiDAR extrinsic while correctly accounting for the ego motion. The resulting semantic point cloud is probabilistically fused into a 3-D vehicle-centric occupancy map covering a volume of 100 × 100 × 30 m, which is defined as the *micro range*, at an isotropic voxel resolution of 20 cm. The 3-D voxel mapper is a significantly enhanced version of [83], which has been extended to allow for low-latency robot-centric mapping, voxel decay, fusion of semantic segmentation, and a wide range of further features while being able to run in real time on a single GPU. From the 3-D occupancy map, an elevation map is extracted, using the height of the lowest occupied voxel along the z-direction, followed by a sequence of geometric and semantics-based heuristics, to estimate the ground plane beneath vegetation and hazards. A heuristic-based approach is used to fuse the semantic probabilities along the z-direction into the elevation

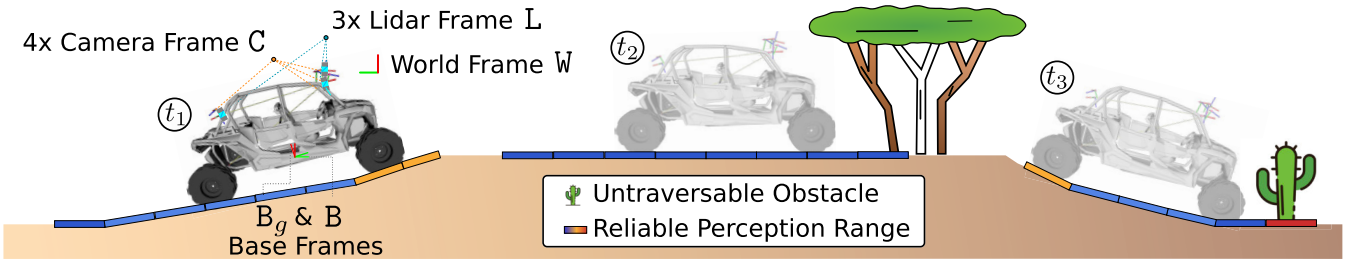

FIGURE 3. Definition of frames and illustration of hindsight self-supervision. At the first timestep, t_1, the reference frames are defined. The gravity-aligned base frame B_g is fixed to the vehicle (position and yaw), with roll and pitch being gravity-aligned. The tiled region below the vehicle on the ground illustrates the reliable perception range per timestamp, where sufficient sensory information is available such that X-Racer can correctly predict the elevation and traversability (color of each tile). When the vehicle approaches the tree at timestamp t_2, X-Racer can correctly predict that the area underneath the canopy is traversable. Similarly, in timestamp t_3, the cactus can be identified as untraversable. While X-Racer requires exhaustive geometric information, which is only available in the proximity of the vehicle, more precise traversability and elevation maps, the so-called pseudo ground truth, can be generated as a learning objective for RoadRunner when taking into account future and past sensory information. For example, in timestamp t_2, RoadRunner can learn to correctly identify the cactus as a hazard from the image data, even with insufficient geometric information available.

map, which takes into account the distance of the voxel to the elevation map surface and its semantic class probabilities. For example, all voxels up to a height of 2.5 m above the elevation map labeled as a tree with a likelihood of over 50% are fused into the elevation map cell. On the other hand, voxels above 2.5 m labeled as *tree* can be excluded to allow driving underneath the canopy.

4) TRAVERSABILITY ASSESSMENT

The traversability assessment is based on a previously developed planner for subterranean and unstructured environments [13]. It handles perceptual uncertainty based on the notation of certainty and risk. We provide an overview of the different risks taken into account to obtain the wheel risk, which describes the risk associated with an individual wheel interacting with the terrain. First, a reliability map is computed, which is proportional to the density of the available geometric information. For example, if insufficient geometric observations about a specific area within the elevation map are available, the estimates for these cells are considered to be less reliable. Furthermore, the elevation map is interpolated and smoothed such that the risk for slope, curvature, roughness, and positive and negative obstacles can be extracted. Additional semantic risks for hard and soft obstacles are computed. The final wheel risk, which describes the risk associated with the terrain–wheel interaction, is given by the CVaR of the combined assessed risk distributions. The full X-Racer software stack is tuned in simulation and on real-world data. In the following, we refer to the CVaR of the wheel risk as the traversability, where a value of 0 indicates safe to traverse, whereas 1 is unsafe.

5) TRAJECTORY PLANNING

The resulting CVaR wheel risk traversability map of size 100×100 m at a resolution of 20 cm is provided to the MPPI planner [89] responsible for obstacle avoidance at high speed. The planner operates with a 5-s planning horizon and is evaluated at a frequency of 30 Hz. The planner has access to the dynamics model of the vehicle and minimizes the time to the goal while trading off soft risks and smoothness of the path. The planned trajectories have to obey rollover, collision, and velocity limitations strictly.

6) LIMITATIONS

While the current software stack is capable of safely guiding the vehicle within desert environments and individual components can be tuned, multiple limitations exist, which are addressed by RoadRunner. Geometric perception of the environment is the dominant driver of X-Racer: semantic information is only incorporated if geometric measurements are available. Therefore, a risk identified only in the image domain cannot be taken into account. This problem is specifically pronounced when operating at higher speed, given the limited update rate of the LiDAR (10 Hz) and its sparse nature at long distances. In addition, while the set of considered semantic classes is carefully selected, it inherently limits the flexibility of the downstream traversability assessment to account for semantic risks. Furthermore, while the sequential structure of the stack allows for interpretability and adjustment of individual components, it leads to a high latency between perception and traversability estimation of over 500 ms. This high latency limits the maximum speed, which is determined by the emergency braking distance and hazard detection (HD) range. Finally, the current software stack cannot make use of (past) experience, in contrast to a rally driver who can anticipate the road progression when turning into a narrow corner and thus take more sophisticated actions under uncertainty. X-Racer stack does not include these generative abilities to imagine the environment, which we argue are important for driving at high speeds.

D. RoadRunner HINDSIGHT TRAVERSABILITY GENERATION

Using the X-Racer software stack, we generate training data for RoadRunner in a self-supervised manner using hindsight. In particular, the traversability and elevation are improved

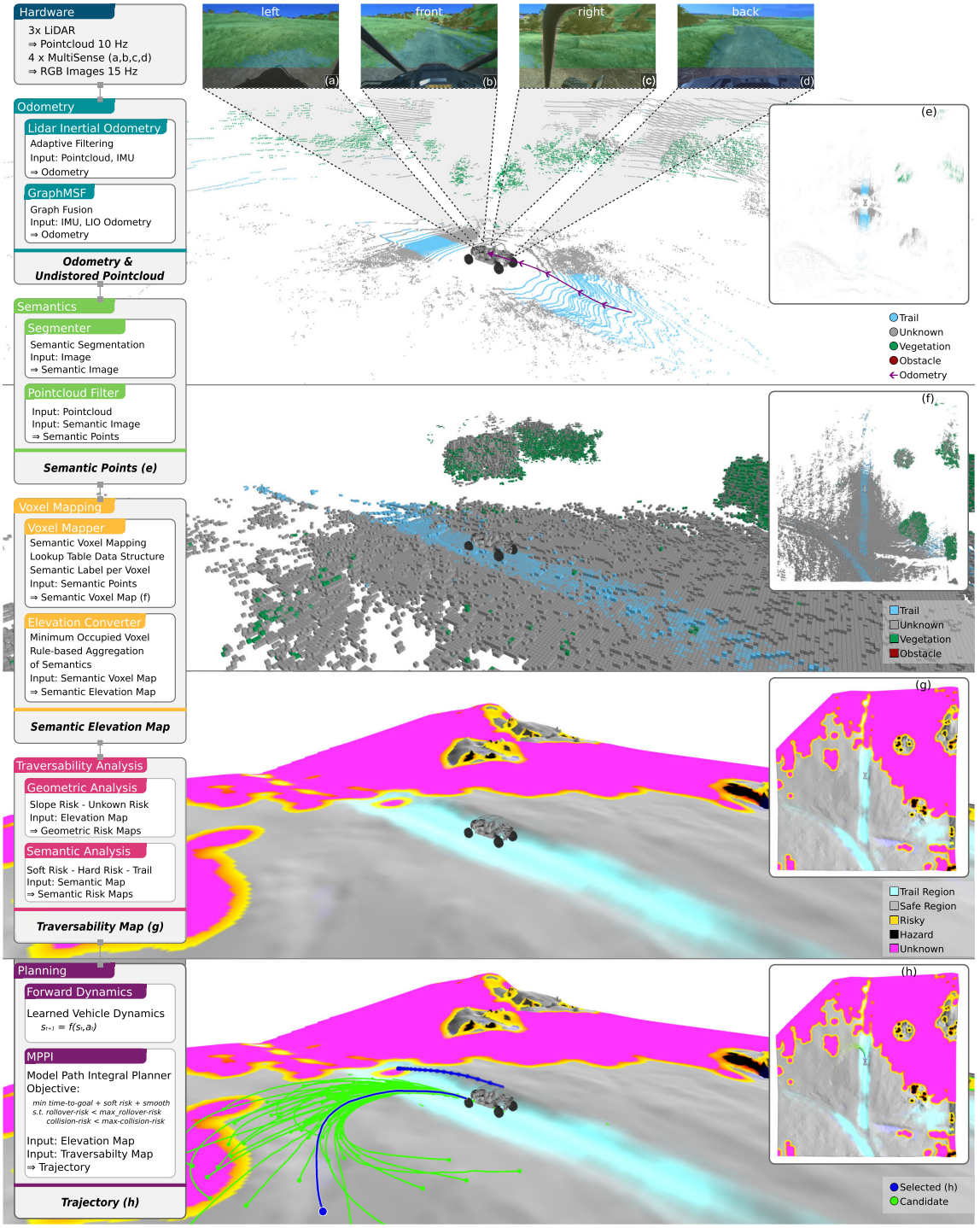

FIGURE 4. Overview of X-Racer. We use our LiDAR inertial odometry system [80] in combination with GraphMSF [81] to obtain smooth, accurate, and high-frequency odometry estimates. (a)–(d) Segmenter [82] is used to predict the semantic classes from each camera image, which are then projected onto the undistorted and filtered point cloud, yielding (e) semantics points. The semantic points are further accumulated in a vehicle-centric voxel map using our voxel mapper based on [83], resulting in a semantic voxel map. (f) Rule-based aggregation method allows converting the semantic voxel map further into a semantic elevation map where the aggregation is tailored to off-road driving and the physical characteristics of our vehicle. (g) Subsequently, both geometric and semantic risks are assessed based on the semantic elevation map, resulting in a traversability map [13]. (h) For downstream trajectory optimization, both the traversability and the elevation are provided to an model predictive path integral (MPPI) planner [84], which employs a learned vehicle dynamics model [85] to compute the final trajectory for a given goal location.

by leveraging privileged future measurements, not available to X-Racer during run time. This reduces the uncertainty about the environment and allows for the extension of the reliable prediction range beyond the current horizon. An illustrative example is presented in Fig. 3. This is implemented by accumulating predictions obtained by X-Racer over time, requiring no modifications of major components such as the mapping backend. To reduce errors due to localization drift, all transformations are performed in the locally consistent odometry frame. Only measurements collected within a time window of 60 s are accumulated for each grid map.

Algorithm 1 Compute Hindsight Ground Truth

```
G_list [ ] ← grid maps within 60s window;
p ← reference SE(2) position;
default ← default value ground truth map;
Fusion ← fusion function;
// Sort with increasing timestamp
Sort(G_list);
// Initalize the return grid map
G_gt ← torch.full(G_list[0].shape, default);
foreach G in G_list do
    // Transform the grid map to G_gt frame
    G_transformed ← Transform(G, p);
    // Fuse information into G_gt
    G_gt ← Fusion(G_gt, G_transformed);
end
return G_gt;
```

Algorithm 1 describes the fusion scheme used for the pseudo ground truth reliability, elevation, and traversability map generation. In the following, we will designate this as the ground truth for simplicity, omitting the term "pseudo." It is important to note, however, that the presented ground truth does not reflect the actual reliability, elevation, or traversability but can be used as a good approximation.

The default value for each map is initialized to be unreliable (value 0), the maximum height ($+\infty$), and traversable (value 0). Each predicted grid map within the given time window is first translated and rotated to the reference position. The fusion function is given by the cell-wise minimum for the elevation, the cell-wise maximum for the reliability, and the temporally latest measurement in addition to a confidence threshold for traversability. The traversability fusion is deliberately not chosen to be a simple maximum function, which one could argue facilitates safety. Experimentally, the maximum function results in an overpessimistic traversability assessment given that under uncertainty areas may be predicted as untraversable by X-Racer. Later, during deployment, with more information available, these areas can often be classified as safe to traverse.

E. RoadRunner NETWORK

RoadRunner is specifically designed to allow for high-speed off-road navigation. This requires RoadRunner to operate at a low latency of 140 ms such that a new prediction of the environment can be provided every 3 m when driving at a speed of up to 20 m/s. Therefore, we choose a prediction range of 100 × 100 m, which is adequate for handling high speeds. We also directly operate on single LiDAR scans rather than a fused map representation. This makes RoadRunner less susceptible to sparse geometric information at high speeds. The number of geometric points per LiDAR point cloud is nearly independent of the vehicle velocity. This is not the case when operating on an accumulated map representation, where the density of geometric information strongly correlates with the vehicle velocity.

1) PREPROCESSING AND FEATURE EXTRACTION

The RoadRunner network builds upon the open-source available implementation of *Lift Splat Shoot* [3], *PointPillars* [4], and *BEVFusion* [5], which combines and improves the prior two methods. The information flow and RoadRunner network architecture are shown in Fig. 5. The images obtained from the hardware-synchronized cameras are first downsampled to a resolution of 512 × 384 and then normalized. The image timestamp t determines the reference frame of the vehicle-centric grid map. The latest measurement of each LiDAR is motion compensated, transformed to the base reference frame $\mathtt{B}_g$, and merged to a single merged point cloud $\mathbf{P}^t_{N\times 3}$. Features are extracted from each camera image using an EfficientNet-B0 [24], which downsamples the input size by a factor of 16, yielding a corresponding feature map $\mathbf{F}^k_{32\times 24\times 328}$ for each image k. The merged point cloud is passed through a *PointPillars* backbone with the same grid configuration as the target grid map of 100 × 100 m at a resolution of 20 cm and results in a point cloud feature map $\mathbf{F}^{\text{pcd}}_{H_G/2\times W_G/2\times 96}$. The previous elevation estimate $\mathbf{G}^{\text{ele},t-1}_{H_G\times H_W\times 1}$ is translated and rotated to align with the new vehicle position. In addition, the elevation is normalized by first scaling the elevation by a factor of 0.05 and then clipping to values between -1 and 1, resulting in $\tilde{\mathbf{G}}^{\text{ele},t}_{H_G\times H_W\times 1}$. This scaling limits the range of elevation to ± 20 m, which is adequate for our off-road scenario (refer to the normalized elevation histogram in Fig. 6).

2) LIFTING

The lifting of a feature from the camera image plane into the 3-D space is achieved by using the pinhole camera model, following [3]. Along the respective ray of the pixel, a set of equally spaced discrete points in 3-D are distributed, resulting in a set of feature points. The weight of a feature point is given by a categorical distribution predicted along the ray based on the image feature embedding of the respective pixel using a 1 × 1 convolutional layer. The feature points are rasterized into grid map cells with the same dimensions and resolution as the target grid map. The weighted feature points coinciding with a grid cell are accumulated by a channel-wise summation using the improved *splat* implementation proposed in [5], resulting in the image feature map $\mathbf{F}^{\text{img}}_{H_G\times W_G\times 328}$.

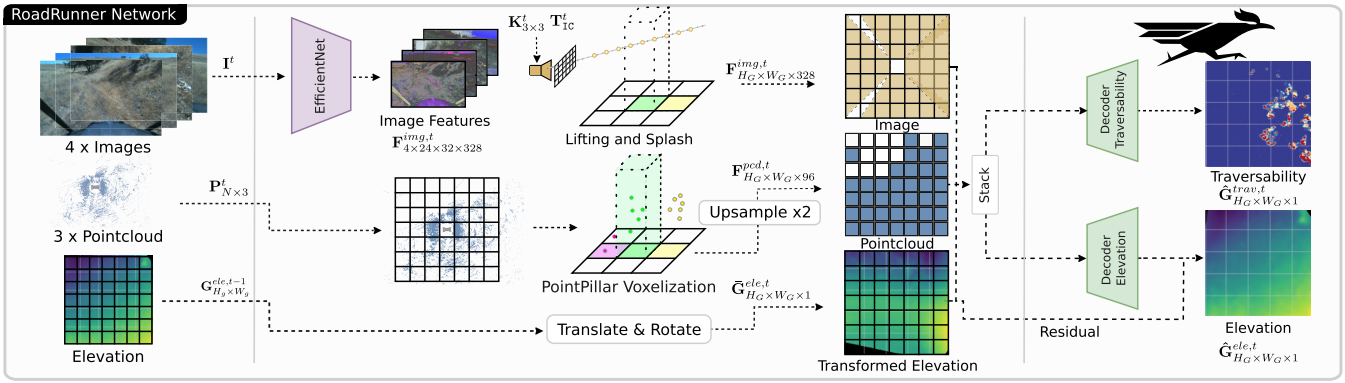

FIGURE 5. Overview of the RoadRunner network. The input to the network consists of four RGB images and a filtered and merged point cloud from three LiDAR sensors, in addition to the past elevation prediction of the previous timestamp $t-1$. The network uses the *Lift Splat Shoot* [3] and the PointPillar [4] architecture to encode the visual and geometric information, respectively. The elevation information is normalized and transformed to the current position, which is then used to predict the traversability and elevation based on separate decoder networks.

Next, we emphasize why we express the grid map in the gravity-aligned base frame $\mathtt{B}_g$. This choice offers a primary advantage of accumulating information along the gravity-aligned z-direction when performing the *splat* operation. In particular, when considering driving up a steep incline, the information about vertical obstacles such as tree trunks or vertical poles is accumulated within a single grid cell, instead of washed out over multiple cells. If no feature point is associated with a grid cell, it is set to 0.

3) MULTIMODAL FUSION

First, the point cloud feature map is upsampled to a resolution of $H_G \times W_G \times 96$. The three feature maps, namely, $\mathbf{F}^{\text{img}}_{H_G \times W_G \times 328}$, $\mathbf{F}^{\text{pcd}}_{H_G \times W_G \times 96}$, and $\bar{\mathbf{G}}^{\text{ele},t}_{H_G \times H_W \times 1}$, are concatenated resulting in the multimodal feature map of size $\mathbf{F}^{\text{fused}}_{H_G \times W_G \times 425}$. Supplying the nearest neighbor interpolated elevation map to the network facilitates the preservation of elevation information for regions that are unobservable within the currently processed merged LiDAR point cloud. Two separate decoder networks are trained for elevation and traversability estimation, predicting $\hat{\mathbf{G}}^{\text{ele}}_{H_G \times W_G}$ and $\hat{\mathbf{G}}^{\text{trav}}_{H_G \times W_G}$, respectively. Each decoder consists of a set of convolutional, batch norm, and ReLU activation layers following a residual architecture. The elevation decoder predicts the residual elevation to the motion-compensated elevation. To enforce a smooth traversability output, a median filter of window size 5×5 is applied to the predicted elevation.

4) RoadRunner LOSS

To train RoadRunner, we use the weighted mean squared error (WMSE) for the traversability and elevation. The loss is calculated only for the valid grid cells containing a supervision signal

$$\mathcal{L}^{\text{layer}}_{\text{WMSE}} = \frac{1}{|\mathbf{G}|} \sum_{x,y \in \mathbf{G}} w^{\text{layer}} \left(\mathbf{G}(x,y) - \hat{\mathbf{G}}(x,y) \right)^2. \quad (1)$$

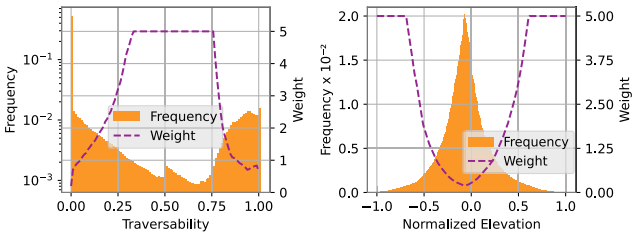

FIGURE 6. Normalization weights for the traversability and normalized elevation. The left axis provides the frequency per bin. The right axis (magenta dotted line) provides the weight. The wheel risk is strongly imbalanced, with most areas being fully traversable (value of 0). The elevation follows a mirrored exponential distribution, with its peak being slightly negative, given that the base frame $\mathtt{B}_g$ is above the ground.

Here, the weighting factor w^{layer} is calculated based on the value of the target grid cell. For this, we compute the normalized frequency $\mathcal{F}_{N_{\text{bins}}}$ of elevation and traversability scores over the training dataset in N_{bins} bins. The clipped inverse normalized frequency gives the final weight

$$w^{\text{layer}} = \text{clip}\left(\left(\bar{\mathcal{F}}^{\text{layer}} * N_{\text{bins}} \right)^{-1}, 0.2, 5 \right). \quad (2)$$

Multiplying by the number of bins results in a weight of 1 for all bins with a uniform probability. The relation between normalized frequency and resulting weight is visualized in Fig. 6. The final optimization objective is given by

$$\mathcal{L}_{\text{FINAL}} = \mathcal{L}^{\text{trav}}_{\text{WMSE}} + \mathcal{L}^{\text{elev}}_{\text{WMSE}}. \quad (3)$$

F. IMPLEMENTATION DETAILS

We use the AdamW optimizer [90] with a learning rate of 10^{-3}, OneCycleLearningRate-schedule [91], and optimize for a total of 16 000 steps. The network is trained on an Nvidia RTX3090 GPU with a batch size of 6. For the EfficientNet-B0, we use pretrained weights from ImageNet and freeze the respective layers. For the lifting operation, we generated a

TABLE 1. Dataset overview—statistics recorded in paso robles.

Split	Duration	Distance	# Traj	# Samples
Train	3389 s	8618 m	4^{+}	10626
Val	644 s	2236 m	4^{-}	2658
Test	1700 s	5727 m	4	7874

$^{+/-}$ Indicates the first 80% ($^{+}$) or last 20% ($^{-}$) of the same trajectory respectively.

total of 353 280 feature points given by the downsampled height of 24, downsampled width of 32, four cameras, and $N_D = 230$ feature points, where we selected the closest distance being 4 m and the furthest distance being 50 m with a spacing of 0.2 m analog to the grid resolution. In total, RoadRunner accounts for 24.0 M parameters, with 5.3 M EfficientNet-B0, 4.4 M *PointPillars*, and 4.6 M *Lift Splat Shoot*, and the remaining 9.8 M parameters correspond to the decoder networks.

IV. EXPERIMENTS

A. DATASET—SCENARIOS

To train our model, we collect a total of 16.5 km of off-road driving data at Paso Robles, CA, USA. The dataset consists of eight individual trajectories resulting in a total of 21 158 samples. We apply our hindsight traversability generation (Section III-D) on all collected trajectories. The trajectories are split into training, validation, and testing data (see Fig. 7). The test data are collected in a different geographic location compared to the training and validation data. However, the test data are from the same ecoregion with similar topography and vegetation. For training and validation, four trajectories are collected, and roughly, the first 80% of them are used for training, while the last 20% of them are used for validation. Table 1 summarizes the key statistics about the dataset.

Each recorded trajectory is converted into a set of samples by selecting camera images based on a minimal traveled distance criterion between samples of 20 cm. In practice, this maintains the diversity of the data while reducing storage and computing requirements during training. Each sample consists of the merged LiDAR scan obtained by all LiDARs, four camera images, elevation map, and the ground truth elevation and ground truth traversability map.

B. METRICS

We assess the *elevation mapping performance* by reporting the MAE, following [41]:

$$\mathcal{L}_{\text{MAE}} = \frac{1}{|\mathbf{G}|} \sum_{x,y \in \mathbf{G}} |\mathbf{G}(x,y) - \hat{\mathbf{G}}(x,y)| \tag{4}$$

where $x, y \in \mathbf{G}$ consists of all valid indices of target grid map layer $\mathbf{G}$, with $\hat{\mathbf{G}}$ denoting the estimated grid map layer. For the *traversability estimation performance*, we follow [12] and evaluate the mse:

$$\mathcal{L}_{\text{mse}} = \frac{1}{|\mathbf{G}|} \sum_{x,y \in \mathbf{G}} \left(\mathbf{G}(x,y) - \hat{\mathbf{G}}(x,y)\right)^2. \tag{5}$$

In addition, we introduce the reliability-weighted performance measures. The weighted mean absolute error (WMAE) and WMSE are given as

$$\mathcal{L}_{\text{WMAE}} = \frac{1}{|\mathbf{G}|} \sum_{x,y \in \mathbf{G}} \mathbf{C}(x,y) \left|\mathbf{G}(x,y) - \hat{\mathbf{G}}(x,y)\right| \tag{6}$$

$$\mathcal{L}_{\text{WMSE}} = \frac{1}{|\mathbf{G}|} \sum_{x,y \in \mathbf{G}} \mathbf{C}(x,y) \left(\mathbf{G}(x,y) - \hat{\mathbf{G}}(x,y)\right)^2 \tag{7}$$

where $\mathbf{C}(x, y)$ denotes the reliability per grid cell and is directly estimated by X-Racer (Section III-C4).

While the previously introduced metrics (mse and WMSE) evaluate traversability as a continuous-valued function, it is also crucial to classify hazardous regions. This is necessary as downstream planning modules explicitly account for those regions. The distance at which hazard regions can be identified reliably bounds the maximum safe velocity of the vehicle, considering both the emergency braking distance and latency. Similar to X-Racer, we introduce a *fatal risk value* threshold. This threshold enables the conversion of the continuous predicted traversability risk map to a binary classification output. We use the same threshold as applied to X-Racer and utilize it for both the ground truth traversability map and the predicted traversability by RoadRunner. This approach allows us to evaluate precision, recall, and F1-Score in terms of HD.

C. TRAVERSABILITY ESTIMATION PERFORMANCE

The primary focus is to investigate the performance of RoadRunner in comparison to X-Racer. We use the nearest neighbor interpolation to fill the missing traversability estimates of X-Racer. RoadRunner outperforms X-Racer in terms of traversability prediction in mse (0.0113–0.0216), precision (0.412–0.217) while slightly underperforming in recall for the selected *fatal risk value* (cf. Table 2). The numerical value of the traversability prediction is strongly dependent on the dataset (compare the difference between the validation and test datasets). This is due to the different environments and vehicle motion. While, e.g., driving down a hill, sufficient information about the environment is available for RoadRunner to output reliable predictions when driving up a hill and approaching the summit, no information about the summit may be available due to occlusions rendering these data inherently difficult, both for RoadRunner and for X-Racer. Despite the numerical difference between the training and test data, RoadRunner's relative performance is consistent compared to X-Racer on the validation and test data.

The mse per grid cell in the vehicle-centric frame averaged across all samples within the test dataset is depicted in Fig. 8. As the vehicle is moving in a straight line, it corresponds to an upward movement in the grid map. Given that the

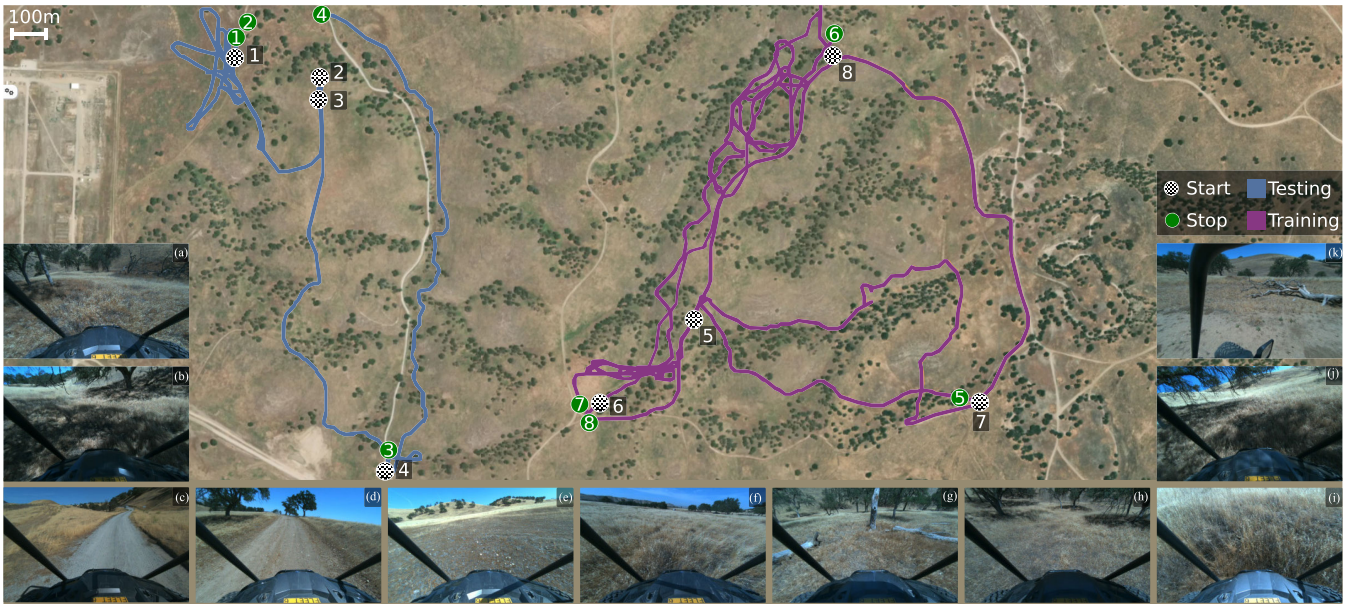

FIGURE 7. Dataset Overview. Magenta trajectories are used for training and validation. Blue trajectories are used for testing. The dataset comprises various driving scenarios illustrated in Fig. 7, including (e) open field, (d) dirt road, and (c) gravel road traversal, as well as operation within (a) confined canyon/ditch settings and (a), (b), and (h) forested environments. The forest environment (b) and (j) necessitated driving beneath the canopy environment posed unique challenges, necessitating driving beneath the canopy. Varying lighting conditions, including shadows from trees [see (a), (b), and (j)], substantially impeded vision-based obstacle detection. In addition, (e) and (f) varying levels of vegetation and (g) and (j) presence of fallen trees within tall grass poses additional challenges for obstacle detection.

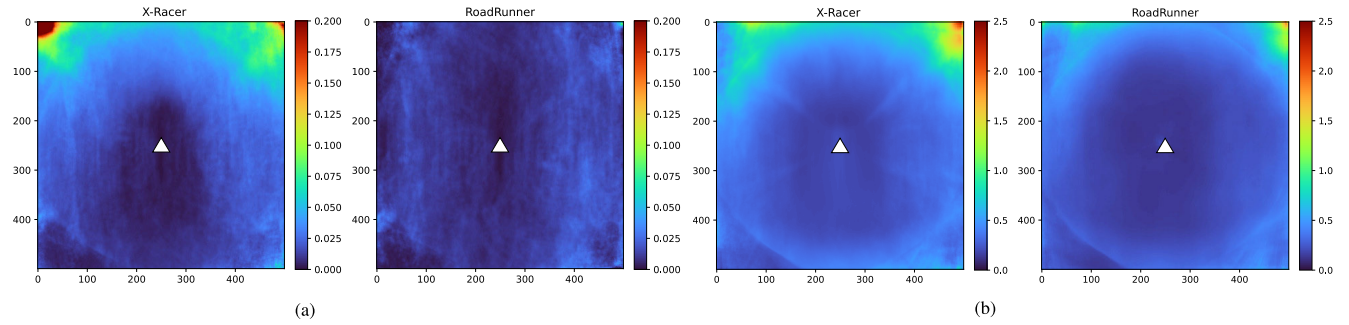

FIGURE 8. Qualitative comparison of X-Racer and RoadRunner per grid cell for (a) traversability prediction in mse and (b) elevation prediction in MAE on average across all samples within the test dataset. The vehicle is located at the center of the map indicated by the Δ, and the front of the vehicle faces toward the top of the grid map. RoadRunner achieves an overall lower elevation and traversability prediction error. X-Racer performs specifically worse in front of the vehicle, where insufficient geometric data have been observed.

TABLE 2. Evaluation of traversability estimation performance.

	MSE↓		WMSE↓		HD-Recall↑		HD-Precision↑	
Method	X-Racer	(ours)	X-Racer	(ours)	X-Racer	(ours)	X-Racer	(ours)
Train	0.0339	**0.0066**	0.0339	**0.0066**	0.292	**0.896**	0.344	**0.777**
Val	0.0413	**0.0295**	0.0413	**0.0295**	0.305	**0.364**	0.399	**0.512**
Test	0.0216	**0.0113**	0.0216	**0.0113**	**0.320**	0.302	0.217	**0.412**

test data exhibit a bias toward driving straight, we expect a better performance in the lower part of the grid map, given that on average, more information is accumulated behind the vehicle. The mse distribution of X-Racer directly confirms and motivates the rationale behind our proposed hindsight ground truth generation. This stems from the observation

TABLE 3. Evaluation of elevation mapping performance.

	MAE [m] ↓		WMAE [m] ↓	
	X-Racer	(ours)	X-Racer	(ours)
Train	0.460	**0.131**	0.460	**0.131**
Val	0.475	**0.323**	0.475	**0.324**
Test	0.329	**0.242**	0.329	**0.242**

that a lower mse is achieved in the lower part of the grid map, which has already been passed by the vehicle. Additional information can be gathered and accumulated for those regions within the semantic voxel map, resulting in an improved traversability estimate. On the other hand, in front of the vehicle, insufficient geometric information is available and the nearest neighbor interpolation leads to a higher error. These results prompt our hypothesis that RoadRunner can more effectively make use of the information provided within the three front-facing cameras and, therefore, estimate traversability better than X-Racer.

The HD performance plotted over the distance is depicted in Fig. 9. Fig. 9 depicts the HD performance plotted over the distance. We highlight that at close range, X-Racer approximately corresponds to the ground truth; therefore, we do not expect RoadRunner to outperform X-Racer. As expected in the vehicle's vicinity, X-Racer detects all hazards correctly (a precision of 1.0); however, at longer ranges, RoadRunner outperforms X-Racer. Above a distance of 25 m, RoadRunner achieves a higher F1-Score than X-Racer while consistently outperforming X-Racer in terms of mse.

D. ELEVATION MAPPING PERFORMANCE

We report the elevation map performance between RoadRunner and X-Racer in Table 3. Before interpreting the results, we would like to recapitulate the generation of the elevation map procedure by X-Racer. X-Racer fuses geometric information in a structured approach into a volumetric map, which is subsequently used in Section III-D to compute the ground truth elevation. Consequently, when comparing the X-Racer against the ground truth, only a substantial performance increase can be expected for unobserved and therefore interpolated regions where no geometric measurements are available for the elevation mapping performance.

Overall RoadRunner outperforms X-Racer. Following the traversability analysis, we present the cell-wise error maps in MAE [see Fig. 8(b)]. Within both elevation error maps, small artifacts in the form of a unit circle are visible, where a jump in performance decrease can be observed. This artifact can be attributed to the fact that while RoadRunner predicts in the vehicle-centric frame, the ground truth is generated using a fixed heading. This artifact is further discussed in Section IV-G.

To further understand when RoadRunner improves the elevation mapping performance, we split cells into *observed* and *unobserved*. We defined the *observability* of a cell based on whether geometric LiDAR measurements have been registered for the specific cell or not. As shown in Fig. 10, RoadRunner strictly outperforms X-Racer across all datasets and for *observed* and *unobserved* areas. In areas that are *unobserved*, which tend to be either occluded or far away from the vehicle, our method generally has a higher MAE. For *observed* regions, the interpolation of X-Racer fails to accurately reconstruct the elevation, while RoadRunner shows better performance. However, our method encounters challenges due to domain shift when dealing with the *unobserved* regions in the validation and test environments, given that the MAE is significantly higher for validation and testing than for the training regions (compare, see Fig. 10, center plot performance difference between X-Racer and RoadRunner). On the contrary, X-Racer performs consistently across datasets. We hypothesize that this is due to the relatively small dataset providing evidence that by increasing the data augmentation or the dataset size even better performance can be achieved.

In summary, we hypothesize that using visual and geometric cues end-to-end can benefit traversability prediction, while accurate prediction of a continuous elevation is significantly harder, specifically when no geometric information is available.

TABLE 4. Comparison and ablation.

	Elevation		Risk MSE			
	MAE↓	WMAE↓	MSE↓	Recall↑	Precision↑	F1↑
(ours)	**0.242**	**0.241**	**0.0113**	**0.302**	**0.412**	**0.349**
PointPillar	0.645	0.643	0.0118	0.274	0.367	0.314
LSS	0.643	0.640	0.0118	0.281	0.370	0.319
No Weighting	**0.241**	0.241	0.0112	0.297	0.413	0.345
Not Frozen	0.247	0.246	0.0116	0.263	0.406	0.319
Ele-Classification	0.247	0.246	0.0113	0.302	**0.415**	0.349
Common Decoder	0.249	0.248	**0.0108**	**0.348**	0.410	**0.376**

E. COMPARISON AND ABLATION STUDY

To understand the importance of individual modalities, we compare our work to PointPillar [4] (geometry-only) and *Lift Splat Shoot* [3] (vision-only). While a comparison to [41] would be desirable, the different choice of sensor modalities (stereo depth compared to ours using LiDAR) and the closed source code renders the comparison infeasible.

In Table 4, we report the the mse and WMSE for the traversability estimation and the MAE and WMAE for the elevation mapping performance. The superior performance of RoadRunner aligns with the findings of [5] and [41] that multimodal sensor fusion results in superior performance.

In addition, we ablate the individual design choices of RoadRunner. *No Weighting* indicates removing the balancing for the traversability and elevation loss and using the mse

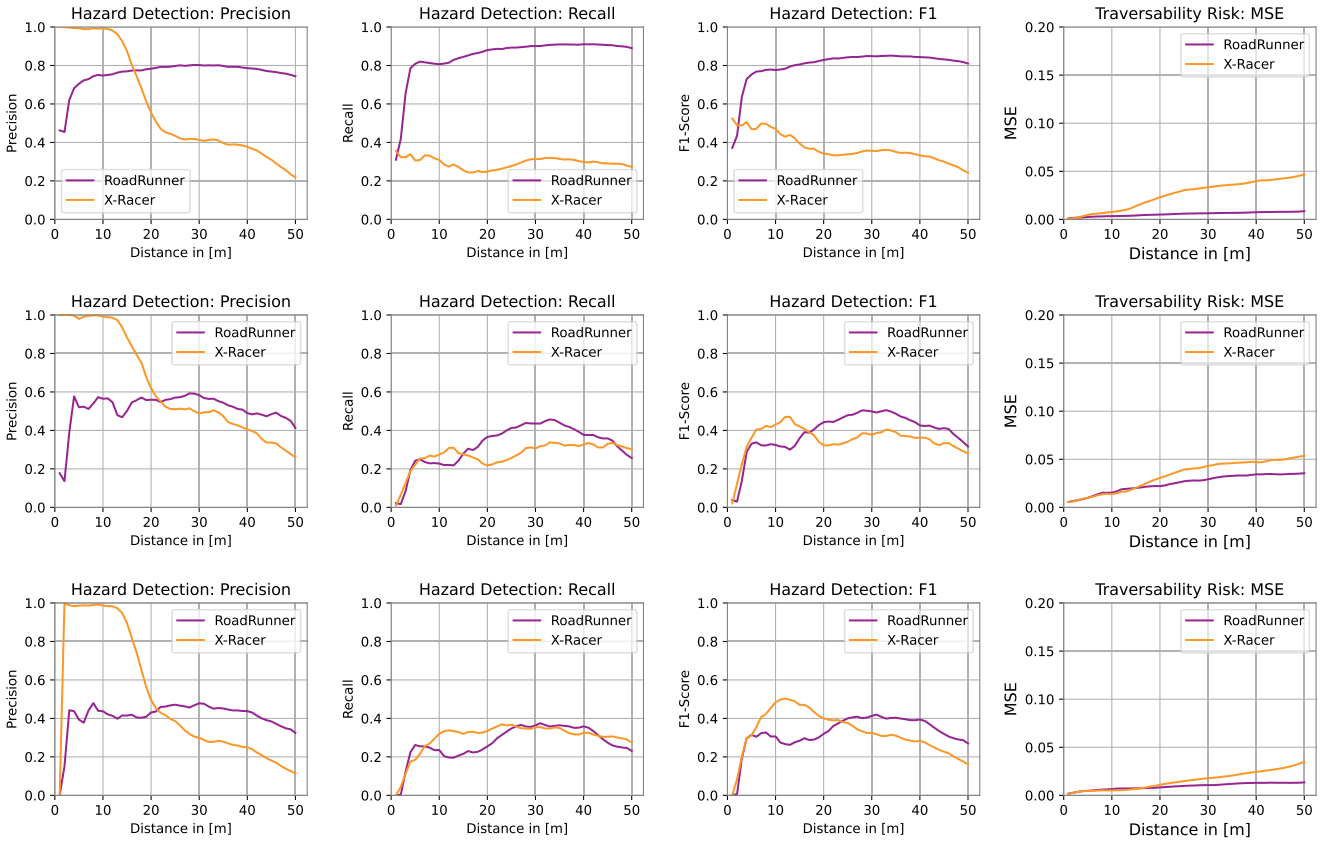

FIGURE 9. HD error metrics of RoadRunner and X-Racer on the training (1st row), validation (2nd row), and test (3th row) dataset. We report the precision (1st column), recall (2nd column), F1-Score (3rd column), and the mse (4th column), plotted over the distance. We report the error evaluated for each 1-m bin independently as opposed to the cumulative error up to the distance. RoadRunner outperforms X-Racer across all dataset above a range of 20 m in terms of mse. In the proximity to the vehicle, X-Racer, as expected, corresponds nearly perfectly to the ground truth resulting in perfect predictions.

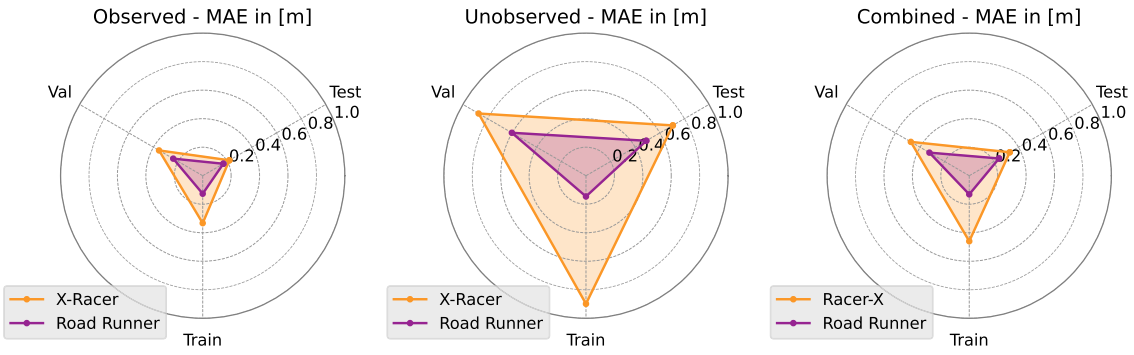

FIGURE 10. Comparison of elevation mapping performance across training, validation, and test datasets. *Observed* indicates that LiDAR measurement is available for the predicted cell, while *Unobserved* denotes that no LiDAR measurement is available. *Combined* is the evaluation of both *Unobserved* and *Observed* regions. Note that there are significantly more observed than unobserved regions, and therefore, *Combined* does not correspond to the arithmetic mean.

and MAE, respectively, without weighting. Removing the balancing slightly increases the traversability prediction performance in terms of mse (0.0113 → 0.0112) but leads to an overall worse F1-Score (0.349 → 0.345). By the increased weighting of high-risk areas, the RoadRunner network predicts a more conservative estimate, which is desirable. *Not Frozen* allows to update the weights of the EfficientNet-B0 backbone. Updating all weights during training leads to better performance on the training dataset but overall hinders generalization to new environments leading to a worse test

score, across all metrics. This validates our design choice of freezing the EfficientNet-B0 backbone. *Common-Decoder* changes the dual decoder network architecture to a shared decoder network for the traversability and elevation mapping task. Given that the performance for the elevation mapping task is degraded ($0.242 \rightarrow 0.249$), we decided on a separate decoder architecture. We hypothesize that the features learned to accurately predict traversability and elevation are not complementary.

F. DEPLOYMENT

Our Polaris RZR S4 1000 Turbo is equipped with a Threadripper 3990x CPU (64 Core 2.9/4.3 GHz), 256-GB RAM, and 4xGeForce RTX 3080 GPUs. We integrate RoadRunner using ROS and use a single GPU for inference. To achieve low latency and overcome the slow data serialization of the Python ROS wrapper, we write a C++ node using Python bindings to the network implemented in PyTorch.

This allows RoadRunner to run with a latency of 131.85 ± 2.5 ms compared with the modular X-Racer with an average latency of over 500 ms. The timings are reported over 700 samples with an average of 36 766 points per sample.

Figs. 11 and 12 provide the example outputs of RoadRunner illustrating successful predictions as well as the limitations. Each output includes visualizations of the traversability risk predicted by RoadRunner, X-Racer, side-by-side to the (hindsight)-generated ground truth risk. In addition, we present visualizations for our predicted elevation, the feedforwarded elevation (nearest neighbor interpolated elevation map predicted by X-Racer), and the ground truth elevation map. The currently merged point cloud is displayed, along with an assessment of the reliability of the ground truth. In the top row, the predicted traversability and elevation by RoadRunner are projected onto the camera images up to a range of roughly 30 m. This comprehensive presentation offers a detailed insight into the capabilities and performance of our RoadRunner in terms of traversability risk and elevation predictions.

Fig. 12(a) illustrates a driving scenario on top of a hill. X-Racer was unable to accumulate sufficient geometric information to correctly identify the trees ahead (highlighted by the yellow box). In contrast, RoadRunner provides a more accurate traversability estimation, correctly identifying the trees ahead as high-risk regions. In Fig. 12(b), a similar scenario is presented: RoadRunner correctly identifies tree trunks as untraversable compared to X-Racer. Specifically, the two trees to the direct left of the vehicle are identified as high-risk regions (highlighted by the yellow box). In Fig. 12(c), the predicted traversability risk map correctly identifies the most prominent risk in the scene—the two trees in the direct surroundings. On the other hand, the trees behind the vehicle are not identified correctly in the traversability risk map. Concerning elevation, while RoadRunner accurately predicts a smooth increase in elevation to the left of the vehicle, it is unable to estimate the correct numerical values and underestimates the slope (highlighted by the red box). This discrepancy becomes apparent when comparing the elevation predicted by RoadRunner with the ground truth map. We hypothesize that this issue arises due to the limited information available from the images and point cloud, which only covers the bottom part of the valley. As a result, RoadRunner can only provide a ''guess'' of the elevation profile within the unobserved areas, given that no direct image or LiDAR measurements are available.

In Fig. 11(a), we highlighted a limitation of automatic ground truth generation. In this scenario, a second vehicle (visible in the rear-facing camera, highlighted by the yellow box) is following the main vehicle. Consequently, the ground truth risk map is corrupted by the motion of this dynamic obstacle. This can be seen by the small untraversable regions in front of the vehicle within the ground truth traversability risk (highlighted by the yellow box). Despite this, both RoadRunner and X-Racer can accurately predict the correct traversability, where the empty road and field ahead are labeled as risk-free traversable.

In Fig. 11(b), RoadRunner demonstrates a particularly accurate estimation of the scene's elevation. The bottom of the canyon in front of the vehicle, which cannot be sensed by the LiDAR sensors, is nearly perfectly identified.

G. LIMITATIONS

The generalization capability of RoadRunner to novel environments is constrained by the limited size and diversity of the training dataset. Ongoing efforts focus on scaling up the dataset size. Given the constraints imposed by limited training data, it is not anticipated that RoadRunner will seamlessly adapt to entirely new, out-of-distribution environments, and we only showed generalization to data within the same ecoregion.

Moreover, RoadRunner's dependence on self-supervised ground truth generation based on X-Racer imposes limitations on its performance. Although self-supervision is cost-effective and eliminates the need for human annotation, RoadRunner's performance in a sense is upper bounded by X-Racer's capability to identify risks. The integration of sparse human supervision can help to address the existing failure cases of X-Racer, such as metallic wire fences that remain undetected by LiDARs and are easily misclassified in the image domain. In addition, dynamic obstacles as seen in Fig. 11(a) are not correctly handled in the ground truth generation pipeline. Furthermore, concerning the generation of ground truth, it is important to note that while the X-Racer operates within a yaw-fixed frame, our approach (RoadRunner) provides predictions within the robot-centric frame. As a result, aligning the pseudo ground truth to the robot-centric frame leads to artifacts in the form of invalid values depending on the rotation angle between the two frames, as shown in Figs. 11 and 12. Due to the missing supervision during training for those specific regions, the network struggles to learn correct predictions. As part of our future work, we plan to address this implementation detail by expanding

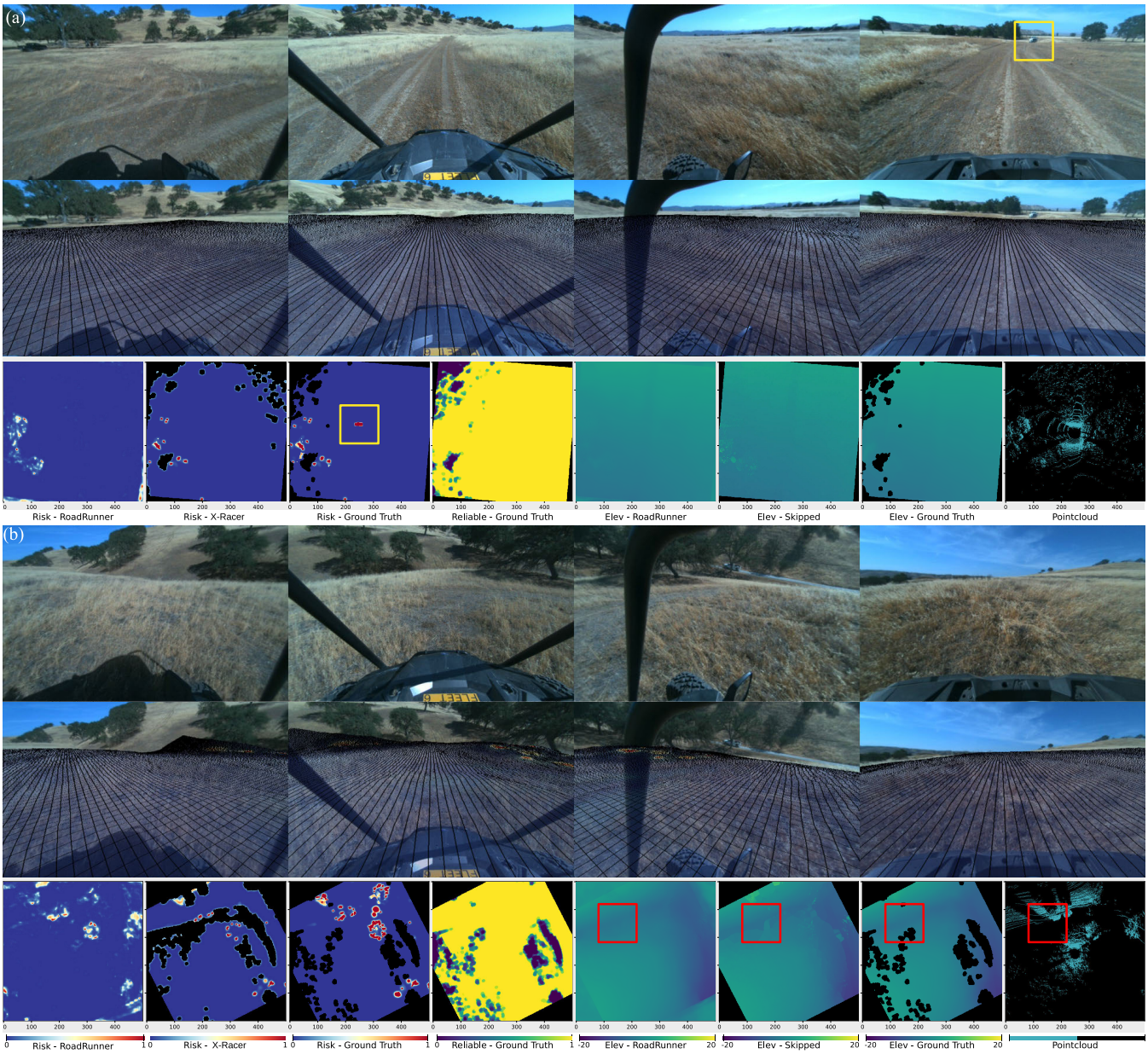

FIGURE 11. Two deployment example predictions. The top two rows of each of the two examples visualize the four onboard camera images (left, front, right, and rear). In the top row, the predicted traversability and elevation are projected onto the images. The bottom row shows the respective traversability, reliability, and elevation grid map, as well as the rasterized merged point cloud. The traversability risk maps are shown for RoadRunner, X-Racer, and the generated ground truth. Blue indicates risk-free traversable and red untraversable. The reliability is shown for the ground truth. The elevation maps are presented for RoadRunner, the nearest neighbor interpolated elevation map predicted by X-Racer, and the ground truth elevation. For all maps, black indicates that no prediction is available. (a) Yellow box in *Risk—Ground Truth* indicates a hazard in front of the robot. No hazard is visible in the front-facing camera. The *Risk—Ground Truth* is corrupted by a car driving behind the vehicle visible in the rear camera image. (b) Red box highlights successful elevation prediction, where RoadRunner learns to correctly predict the small canyon in front of the vehicle, which is not fully observable from the LiDAR point cloud. Further details and interpretation of each example can be found in Section IV-F.

the pseudo ground truth maps by a factor of $\sqrt{2}$, to prohibit those artifacts.

Another limitation is the temporal consistency of the predictions of RoadRunner as well as the missing memory. For instance, when navigating down into a canyon, RoadRunner cannot retain information about obstacles outside the current field of view, as obstacles cannot be captured by cameras nor LiDAR measurements [compare Fig. 12(c)]. This deficit could be addressed by adding memory units or (autoregressive) feedback loops to the RoadRunner network architecture.

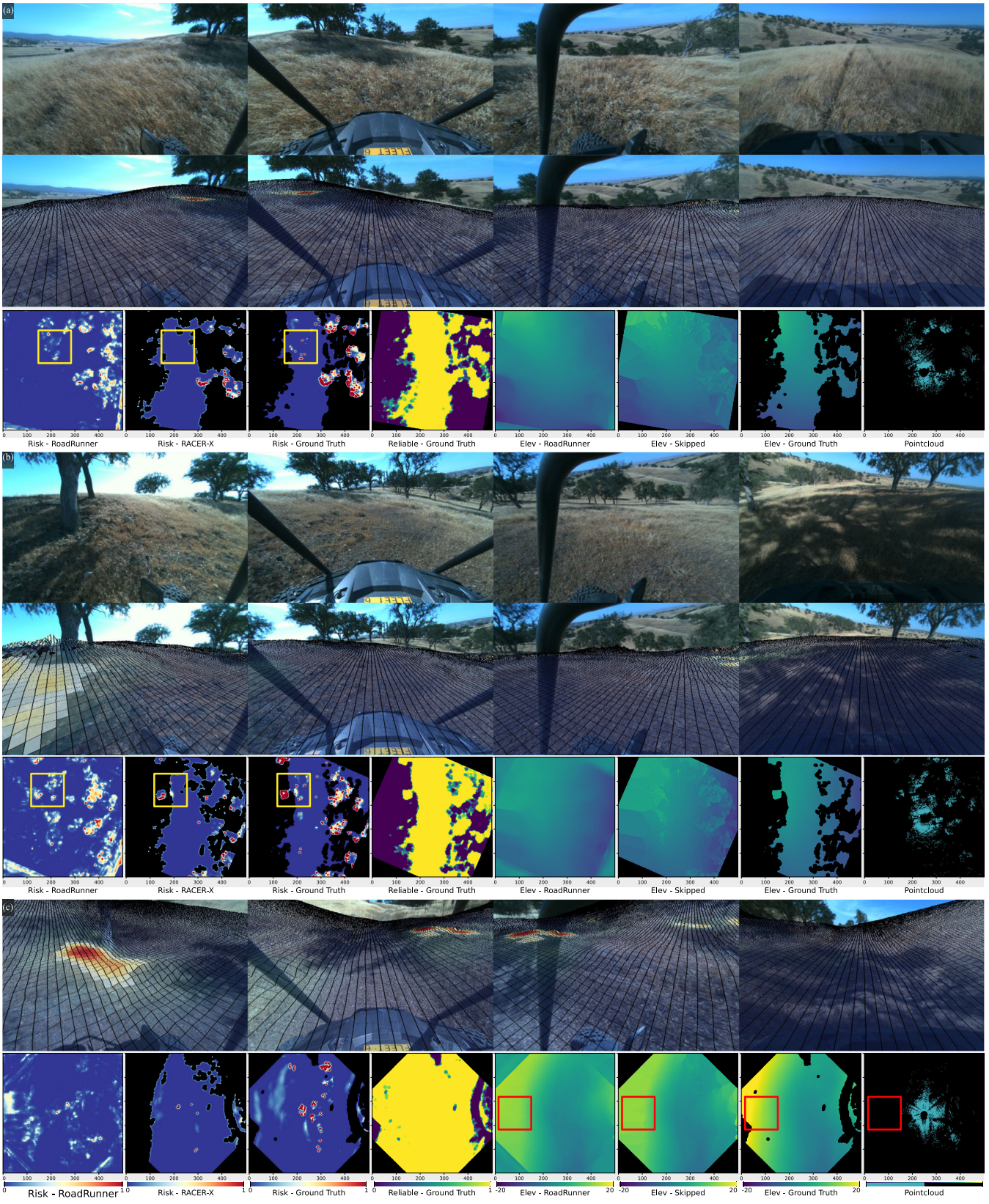

FIGURE 12. (a) and (b) Yellow box highlights successful predictions at long range where X-Racer fails. (c) Red box highlights incorrect elevation prediction, due to missing LiDAR and camera observations. Further details and interpretation of each example can be found in Section IV-F and Fig. 11.

One aspect we did not address was evaluating prediction performance as a function of vehicle velocity. While RoadRunner is inherently designed for handling high speeds, a more rigorous quantitative evaluation is needed. One, e.g., could drive the same trajectory at different speeds and compare the performance of RoadRunner as a function of velocity.

While RoadRunner can provide accurate traversability and elevation information, a better understanding of the uncertainty of the prediction would be beneficial for the planning and fusion of the predictions. For this, one could integrate methods available in the evidential deep learning community to understand the epistemic uncertainty and detect out-of-distribution data specifically.

Finally, RoadRunner does not provide theoretical guarantees, in contrast to X-Racer, where, under specific conditions such as optimal sensing and constrained environments, guarantees can be made based on implemented heuristics.

V. CONCLUSION AND FUTURE WORK

In conclusion, RoadRunner represents a significant leap forward in the realm of mapping and traversability estimation for high-speed off-road autonomous robot navigation. We showcase improved traversability prediction and elevation mapping performance, the benefit of sensor fusion, and a rigorous analysis of the performance for a 16.5-km off-road dataset. With a substantial reduction in latency, RoadRunner has the potential to enable vehicles to operate at higher speeds autonomously in complex unstructured environments. The implications of this approach extend to critical areas such as search and rescue operations and even hold promise for robot applications in extraterrestrial terrain, where reliable traversability understanding is key.

In future research, we will address existing challenges laid out in Section IV-G while exploring novel avenues, including attention-based fusion of information, evidential deep learning methods for uncertainty prediction, and a more physically grounded estimation of traversability concerning vehicle dynamics by exploring the integration of the learned dynamics models. We hypothesize novel data augmentation methods allowing for a coherent augmentation of point cloud and image data in combination with synthetic data may have the potential to further substantially improve the performance and generalization. Finally, we recognize the need for publicly available datasets and benchmarks for off-road terrain understanding, which will allow for comparing various methods across varying off-road environments and ecoregions.

REFERENCES

[1] M. A. Maxon, *The Real Roadrunner*, vol. 9. Norman, OK, USA: University of Oklahoma Press, 2005.

[2] L. Stanislas et al., "Airborne particle classification in lidar point clouds using deep learning," in *Proc. 12th Int. Conf. Field Service Robot.* Cham, Switzerland: Springer, 2021, pp. 395–410.

[3] J. Philion and S. Fidler, "Lift, splat, shoot: Encoding images from arbitrary camera rigs by implicitly unprojecting to 3D," in *Proc. 16th Eur. Conf. Comput. Vis. (ECCV)* Glasgow, U.K.: Springer, Aug. 2020, pp. 194–210.

[4] A. H. Lang, S. Vora, H. Caesar, L. Zhou, J. Yang, and O. Beijbom, "PointPillars: Fast encoders for object detection from point clouds," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 12697–12705, doi: 10.1109/CVPR.2019.01298.

[5] Z. Liu et al., "BEVFusion: Multi-task multi-sensor fusion with unified bird's-eye view representation," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2023, pp. 2774–2781.

[6] C. M. Shoemaker and J. A. Bornstein, "The demo III UGV program: A testbed for autonomous navigation research," in *Proc. IEEE Int. Symp. Intell. Control (ISIC), IEEE Int. Symp. Comput. Intell. Robot. Autom. (CIRA)*, Sep. 1998, pp. 644–651.

[7] A. Kelly et al., "Toward reliable off road autonomous vehicles operating in challenging environments," *Int. J. Robot. Res.*, vol. 25, nos. 5–6, pp. 449–483, May 2006.

[8] D. Kim, J. Sun, S. Min Oh, J. M. Rehg, and A. F. Bobick, "Traversability classification using unsupervised on-line visual learning for outdoor robot navigation," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2006, pp. 518–525, doi: 10.1109/ROBOT.2006.1641763.

[9] R. Hadsell et al., "Learning long-range vision for autonomous off-road driving," *J. Field Robot.*, vol. 26, no. 2, pp. 120–144, Feb. 2009, doi: 10.1002/rob.20276.

[10] J. Guzzi, R. O. Chavez-Garcia, M. Nava, L. M. Gambardella, and A. Giusti, "Path planning with local motion estimations," *IEEE Robot. Autom. Lett.*, vol. 5, no. 2, pp. 2586–2593, Apr. 2020, doi: 10.1109/LRA.2020.2972849.

[11] J. Frey, D. Hoeller, S. Khattak, and M. Hutter, "Locomotion policy guided traversability learning using volumetric representations of complex environments," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Oct. 2022, pp. 5722–5729.

[12] J. Frey, M. Mattamala, N. Chebrolu, C. Cadena, M. Fallon, and M. Hutter, "Fast traversability estimation for wild visual navigation," in *Proc. 19th Robotics: Sci. Syst.*, Daegu, Republic of Korea, Jul. 2023, doi: 10.15607/rss.2023.xix.054. [Online]. Available: https://www.roboticsproceedings.org/rss19/p054.html

[13] D. D. Fan, K. Otsu, Y. Kubo, A. Dixit, J. Burdick, and A. Agha-Mohammadi, "STEP: Stochastic traversability evaluation and planning for safe off-road navigation," *CoRR*, vol. abs/2103.02828, Mar. 2021. [Online]. Available: https://www.roboticsproceedings.org/rss17/p021.html

[14] A. Majumdar and M. Pavone, "How should a robot assess risk? Towards an axiomatic theory of risk in robotics," in *Proc. 18th Int. Symp. Robot. Res.*, vol. 10, N. M. Amato, G. Hager, S. L. Thomas, and M. Torres-Torriti, Eds., Puerto Varas, Chile. Cham, Switzerland: Springer, Dec. 2017, pp. 75–84, doi: 10.1007/978-3-030-28619-4_10.

[15] D. A. Pomerleau, "Knowledge-based training of artificial neural networks for autonomous robot driving," in *Robot Learning*. Cham, Switzerland: Springer, 1993, pp. 19–43.

[16] R. Manduchi, A. Castano, A. Talukder, and L. Matthies, "Obstacle detection and terrain classification for autonomous off-road navigation," *Auto. Robots*, vol. 18, no. 1, pp. 81–102, Jan. 2005.

[17] L. D. Jackel, E. Krotkov, M. Perschbacher, J. Pippine, and C. Sullivan, "The DARPA LAGR program: Goals, challenges, methodology, and phase I results," *J. Field Robot.*, vol. 23, nos. 11–12, pp. 945–973, Nov. 2006, doi: 10.1002/rob.20161.

[18] U. Müller, J. Ben, E. Cosatto, B. Flepp, and Y. Cun, "Off-road obstacle avoidance through end-to-end learning," in *Advances in Neural Information Processing Systems*, vol. 18, Y. Weiss, B. Schölkopf, and J. Platt, Eds. Cambridge, MA, USA: MIT Press, 2005.

[19] R. Hadsell et al., "Online learning for offroad robots: Spatial label propagation to learn long-range traversability," in *Proc. 3rd Robot., Sci. Syst.*, Jun. 2007, p. 32.

[20] K. Konolige et al., "Mapping, navigation, and learning for off-road traversal," *J. Field Robot.*, vol. 26, no. 1, pp. 88–113, Jan. 2009.

[21] R. Behringer et al., "The DARPA grand challenge–development of an autonomous vehicle," in *Proc. IEEE Intell. Vehicles Symp.*, Jun. 2004, pp. 226–231.

[22] S. Thrun, M. Montemerlo, and A. Aron, "Probabilistic terrain analysis for high-speed desert driving," in *Proc. Robot., Sci. Syst.*, Philadelphia, PA, USA, Aug. 2006, doi: 10.15607/RSS.2006.II.021.

[23] C. Urmson et al., "A robust approach to high-speed navigation for unrehearsed desert terrain," *J. Field Robot.*, vol. 23, no. 8, pp. 467–508, 2006.

[24] M. Tan and Q. Le, "EfficientNet: Rethinking model scaling for convolutional neural networks," in *Proc. Int. Conf. Mach. Learn.*, Long Beach, CA, USA, 2019, pp. 6105–6114.

[25] K. Viswanath, K. Singh, P. Jiang, P. B. Sujit, and S. Saripalli, "OFFSEG: A semantic segmentation framework for off-road driving," in *Proc. IEEE 17th Int. Conf. Autom. Sci. Eng. (CASE)*, Aug. 2021, pp. 354–359.

[26] D. Maturana, P.-W. Chou, M. Uenoyama, and S. Scherer, "Real-time semantic mapping for autonomous off-road navigation," in *Proc. 11th Int. Conf. Field Service Robot. (FSR)*, Sep. 2017, pp. 335–350.

[27] P. Roth, J. Nubert, F. Yang, M. Mittal, and M. Hutter, "ViPlanner: Visual semantic imperative learning for local navigation," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2024, pp. 5243–5249.

[28] A. Shaban, X. Meng, J. Lee, B. Boots, and D. Fox, "Semantic terrain classification for off-road autonomous driving," in *Proc. Conf. Robot Learn.*, 2022, pp. 619–629.

[29] G. Erni, J. Frey, T. Miki, M. Mattamala, and M. Hutter, "MEM: Multi-modal elevation mapping for robotics and learning," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Detroit, MI, USA, Oct. 2023, pp. 11011–11018.

[30] M. Wigness, S. Eum, J. G. Rogers, D. Han, and H. Kwon, "A RUGD dataset for autonomous navigation and visual perception in unstructured outdoor environments," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Nov. 2019, pp. 5000–5007.

[31] P. Jiang, P. Osteen, M. Wigness, and S. Saripalli, "RELLIS-3D dataset: Data, benchmarks and analysis," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2021, pp. 1110–1116.

[32] A. Valada, G. Oliveira, T. Brox, and W. Burgard, "Deep multispectral semantic scene understanding of forested environments using multimodal fusion," in *Proc. Int. Symp. Experim. Robot. (ISER)*, 2016, pp. 465–477.

[33] D. M. Bradley et al., "Scene understanding for a high-mobility walking robot," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Sep. 2015, pp. 1144–1151, doi: 10.1109/IROS.2015.7353514.

[34] F. Schilling, X. Chen, J. Folkesson, and P. Jensfelt, "Geometric and visual terrain classification for autonomous mobile navigation," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Sep. 2017, pp. 2678–2684, doi: 10.1109/IROS.2017.8206092.

[35] M. Ono, T. J. Fuchs, A. Steffy, M. Maimone, and J. Yen, "Risk-aware planetary rover operation: Autonomous terrain classification and path planning," in *Proc. IEEE Aerosp. Conf.*, Mar. 2015, pp. 1–10.

[36] B. Rothrock, R. Kennedy, C. Cunningham, J. Papon, M. Heverly, and M. Ono, *SPOC: Deep Learning-Based Terrain Classification for Mars Rover Missions*. Reston, VA, USA: American Institute of Aeronautics and Astronautics, 2016, p. 5539, doi: 10.2514/6.2016-5539.

[37] R. M. Swan et al., "AI4MARS: A dataset for terrain-aware autonomous driving on Mars," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2021, pp. 1982–1991.

[38] J. Zhang, L. Lin, Z. Fan, W. Wang, and J. Liu, "S^5mars: Semi-supervised learning for Mars semantic segmentation," 2022, *arXiv:2207.01200*.

[39] E. Goh, J. Chen, and B. Wilson, "Mars terrain segmentation with less labels," in *Proc. IEEE Aerosp. Conf. (AERO)*, Mar. 2022, pp. 1–10.

[40] M. Endo, T. Taniai, R. Yonetani, and G. Ishigami, "Risk-aware path planning via probabilistic fusion of traversability prediction for planetary rovers on heterogeneous terrains," 2023, *arXiv:2303.01169*.

[41] X. Meng et al., "TerrainNet: Visual modeling of complex terrain for high-speed, off-road navigation," in *Proc. Robot., Sci. Syst.*, South Korea, Jul. 2023. [Online]. Available: https://www.roboticsproceedings.org/rss19/p103.html

[42] R. Triebel, P. Pfaff, and W. Burgard, "Multi-level surface maps for outdoor terrain mapping and loop closing," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Oct. 2006, pp. 2276–2282, doi: 10.1109/IROS.2006.282632.

[43] C. A. Brooks and K. Iagnemma, "Self-supervised terrain classification for planetary surface exploration rovers," *J. Field Robot.*, vol. 29, no. 3, pp. 445–468, May 2012.

[44] K. Otsu, M. Ono, T. J. Fuchs, I. Baldwin, and T. Kubota, "Autonomous terrain classification with co- and self-training approach," *IEEE Robot. Autom. Lett.*, vol. 1, no. 2, pp. 814–819, Jul. 2016, doi: 10.1109/LRA.2016.2525040.

[45] M. G. Castro et al., "How does it feel? Self-supervised costmap learning for off-road vehicle traversability," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2023, pp. 931–938.

[46] J. Seo, S. Sim, and I. Shim, "Learning off-road terrain traversability with self-supervisions only," *IEEE Robot. Autom. Lett.*, vol. 8, no. 8, pp. 4617–4624, Aug. 2023.

[47] S. Higa et al., "Vision-based estimation of driving energy for planetary rovers using deep learning and terramechanics," *IEEE Robot. Autom. Lett.*, vol. 4, no. 4, pp. 3876–3883, Oct. 2019.

[48] J. Zürn, W. Burgard, and A. Valada, "Self-supervised visual terrain classification from unsupervised acoustic feature learning," *IEEE Trans. Robot.*, vol. 37, no. 2, pp. 466–481, Apr. 2021.

[49] A. J. Sathyamoorthy, K. Weerakoon, T. Guan, J. Liang, and D. Manocha, "TerraPN: Unstructured terrain navigation using online self-supervised learning," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Kyoto, Japan, Oct. 2022, pp. 7197–7204, doi: 10.1109/IROS47612.2022.9981942.

[50] C. Richter and N. Roy, "Safe visual navigation via deep learning and novelty detection," in *Robotics: Science and Systems*. Cambridge, MA, USA, Jul. 2017, doi: 10.15607/RSS.2017.XIII.064. [Online]. Available: https://www.roboticsproceedings.org/rss13/p64.html

[51] J. Seo, T. Kim, K. Kwak, J. Min, and I. Shim, "ScaTE: A scalable framework for self-supervised traversability estimation in unstructured environments," *IEEE Robot. Autom. Lett.*, vol. 8, no. 2, pp. 888–895, Feb. 2023.

[52] J. Ahtiainen, T. Stoyanov, and J. Saarinen, "Normal distributions transform traversability maps: LIDAR-only approach for traversability mapping in outdoor environments," *J. Field Robot.*, vol. 34, no. 3, pp. 600–621, May 2017.

[53] M. V. Gasparino et al., "WayFAST: Navigation with predictive traversability in the field," *IEEE Robot. Autom. Lett.*, vol. 7, no. 4, pp. 10651–10658, Oct. 2022, doi: 10.1109/LRA.2022.3193464.

[54] X. Cai, M. Everett, L. Sharma, P. R. Osteen, and J. P. How, "Probabilistic traversability model for risk-aware motion planning in off-road environments," 2022, *arXiv:2210.00153*.

[55] H. Xue et al., "Contrastive label disambiguation for self-supervised terrain traversability learning in off-road environments," 2023, *arXiv:2307.02871*.

[56] X. Cai et al., "EVORA: Deep evidential traversability learning for risk-aware off-road autonomy," 2023, *arXiv:2311.06234*.

[57] S. Jung, J. Lee, X. Meng, B. Boots, and A. Lambert, "V-STRONG: Visual self-supervised traversability learning for off-road navigation," 2023, *arXiv:2312.16016*.

[58] R. Schmid et al., "Self-supervised traversability prediction by learning to reconstruct safe terrain," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Oct. 2022, pp. 12419–12425.

[59] L. Wellhausen, R. Ranftl, and M. Hutter, "Safe robot navigation via multi-modal anomaly detection," *IEEE Robot. Autom. Lett.*, vol. 5, no. 2, pp. 1326–1333, Apr. 2020, doi: 10.1109/LRA.2020.2967706.

[60] E. Chen, C. Ho, M. Maulimov, C. Wang, and S. Scherer, "Learning-on-the-drive: Self-supervised adaptation of visual offroad traversability models," 2023, *arXiv:2306.15226*.

[61] L. Wellhausen and M. Hutter, "ArtPlanner: Robust legged robot navigation in the field," *Field Robot.*, vol. 3, pp. 413–434, 2023.

[62] C. Cao et al., "Autonomous exploration development environment and the planning algorithms," in *Proc. Int. Conf. Robot. Autom. (ICRA)*, May 2022, pp. 8921–8928, doi: 10.1109/ICRA46639.2022.9812330.

[63] H. Xue, H. Fu, L. Xiao, Y. Fan, D. Zhao, and B. Dai, "Traversability analysis for autonomous driving in complex environment: A LiDAR-based terrain modeling approach," *J. Field Robot.*, vol. 40, no. 7, pp. 1779–1803, Oct. 2023.

[64] N. Hudson et al., "Heterogeneous ground and air platforms, homogeneous sensing: Team CSIRO Data61's approach to the DARPA subterranean challenge," *Field Robot.*, vol. 2, no. 1, pp. 595–636, Mar. 2022, doi: 10.55417/fr.2022021.

[65] A. Bouman et al., "Autonomous spot: Long-range autonomous exploration of extreme environments with legged locomotion," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Oct. 2020, pp. 2518–2525.

[66] G. Kahn, P. Abbeel, and S. Levine, "BADGR: An autonomous self-supervised learning-based navigation system," *IEEE Robot. Autom. Lett.*, vol. 6, no. 2, pp. 1312–1319, Apr. 2021, doi: 10.1109/LRA.2021.3057023.

[67] Y. Kim, C. Kim, and J. Hwangbo, "Learning forward dynamics model and informed trajectory sampler for safe quadruped navigation," in *Robotics: Science and Systems XVIII*, K. Hauser, D. A. Shell, and S. Huang, Eds. New York, NY, USA, Jul. 2022, doi: 10.15607/RSS.2022.XVIII.069. [Online]. Available: https://www.roboticsproceedings.org/rss18/p069.html

[68] X. Xiao, J. Biswas, and P. Stone, "Learning inverse kinodynamics for accurate high-speed off-road navigation on unstructured terrain," *IEEE Robot. Autom. Lett.*, vol. 6, no. 3, pp. 6054–6060, Jul. 2021.

[69] H. Karnan et al., "VI-IKD: High-speed accurate off-road navigation using learned visual-inertial inverse kinodynamics," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Oct. 2022, pp. 3294–3301.

[70] M. Wulfmeier, P. Ondruska, and I. Posner, "Maximum entropy deep inverse reinforcement learning," 2015, *arXiv:1507.04888*.

[71] S. Triest, M. G. Castro, P. Maheshwari, M. Sivaprakasam, W. Wang, and S. Scherer, "Learning risk-aware costmaps via inverse reinforcement learning for off-road navigation," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2023, pp. 924–930.

[72] D. D. Fan, A.-A. Agha-Mohammadi, and E. A. Theodorou, "Learning risk-aware costmaps for traversability in challenging environments," *IEEE Robot. Autom. Lett.*, vol. 7, no. 1, pp. 279–286, Jan. 2022, doi: 10.1109/LRA.2021.3125047.

[73] A. Dixit, D. D. Fan, K. Otsu, S. Dey, A.-A. Agha-Mohammadi, and J. W. Burdick, "STEP: Stochastic traversability evaluation and planning for risk-aware off-road navigation; results from the DARPA subterranean challenge," 2023, *arXiv:2303.01614*.

[74] T. Roddick, A. Kendall, and R. Cipolla, "Orthographic feature transform for monocular 3D object detection," in *Proc. 30th Brit. Mach. Vis. Conf.*, Cardiff, U.K., Sep. 2019, p. 285.

[75] M. H. Ng, K. Radia, J. Chen, D. Wang, I. Gog, and J. E. Gonzalez, "BEV-Seg: Bird's eye view semantic segmentation using geometry and semantic point cloud," in *Proc. Robot., Sci. Syst.*, Jul. 2024, doi: 10.15607/RSS.2021.XVII.021.

[76] A. W. Harley, Z. Fang, J. Li, R. Ambrus, and K. Fragkiadaki, "Simple-BEV: What really matters for multi-sensor BEV perception?" in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2023, pp. 2759–2765.

[77] E. Xie et al., "M^2BEV: Multi-camera joint 3D detection and segmentation with unified birds-eye view representation," 2022, *arXiv:2204.05088*.

[78] M. Diaz-Zapata, D. Sierra-Gonzalez, Ö. Erkent, C. Laugier, and J. Dibangoye, "LAPTNet-FPN: Multi-scale LiDAR-aided projective transform network for real time semantic grid prediction," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, Londres, U.K., May 2023, pp. 712–718. [Online]. Available: https://hal.science/hal-03980399

[79] A. Saha, O. Mendez, C. Russell, and R. Bowden, "Translating images into maps," in *Proc. Int. Conf. Robot. Autom. (ICRA)*, Philadelphia, PA, USA, May 2022, pp. 9200–9206, doi: 10.1109/ICRA46639.2022.9811901.

[80] S. Fakoorian, K. Otsu, S. Khattak, M. Palieri, and A.-A. Agha-Mohammadi, "ROSE: Robust state estimation via online covariance adaption," in *Springer Proceedings in Advanced Robotics*. Cham, Switzerland: Springer, 2023, pp. 452–467.

[81] J. Nubert, S. Khattak, and M. Hutter, "Graph-based multi-sensor fusion for consistent localization of autonomous construction robots," in *Proc. Int. Conf. Robot. Autom. (ICRA)*, May 2022, pp. 10048–10054, doi: 10.1109/ICRA46639.2022.9812386.

[82] R. Strudel, R. Garcia, I. Laptev, and C. Schmid, "Segmenter: Transformer for semantic segmentation," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Montreal, QC, Canada, Oct. 2021, pp. 7242–7252, doi: 10.1109/ICCV48922.2021.00717.

[83] T. Overbye and S. Saripalli, "G-VOM: A GPU accelerated voxel off-road mapping system," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Jun. 2022, pp. 1480–1486, doi: 10.1109/IV51971.2022.9827107.

[84] G. Williams et al., "Information theoretic MPC for model-based reinforcement learning," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2017, pp. 1714–1721, doi: 10.1109/ICRA.2017.7989202.

[85] J. Gibson et al., "A multi-step dynamics modeling framework for autonomous driving in multiple environments," in *Proc. IEEE Int. Conf. Robot. Automat.*, May 2023, pp. 7959–7965.

[86] S. Fakoorian, A. Santamaria-Navarro, B. T. Lopez, D. Simon, and A.-A. Agha-Mohammadi, "Towards robust state estimation by boosting the maximum correntropy criterion Kalman filter with adaptive behaviors," *IEEE Robot. Autom. Lett.*, vol. 6, no. 3, pp. 5469–5476, Jul. 2021, doi: 10.1109/LRA.2021.3073646.

[87] MMSegmentation Contributors. (2020). *MMSegmentation: OpenmmLab Semantic Segmentation ToolBox and Benchmark*. [Online]. Available: https://github.com/open-mmlab/mmsegmentation

[88] Duality Robotics. (2023). *Duality Robotics—Falcon Pro Simulator*. [Online]. Available: https://www.duality.ai/product

[89] G. Williams, P. Drews, B. Goldfain, J. M. Rehg, and E. A. Theodorou, "Information-theoretic model predictive control: Theory and applications to autonomous driving," *IEEE Trans. Robot.*, vol. 34, no. 6, pp. 1603–1622, Dec. 2018, doi: 10.1109/TRO.2018.2865891.

[90] I. Loshchilov and F. Hutter, "Decoupled weight decay regularization," in *Proc. 7th Int. Conf. Learn. Represent. (ICLR)*, New Orleans, LA, USA, May 2019. [Online]. Available: https://openreview.net/forum?id=Bkg6RiCqY7

[91] L. N. Smith and N. Topin, "Super-convergence: Very fast training of neural networks using large learning rates," 2017, *arXiv:1708.07120*.

JONAS FREY (Student Member, IEEE) received the M.Sc. degree in robotics, systems and control from the Swiss Federal Institute of Technology (ETH Zürich), Zürich, Switzerland, in 2021, where he is currently pursuing the Ph.D. degree with the Robotic Systems Laboratory.

He is with the MPI ETH Center for Learning Systems, Max Planck Institute, Tübingen, Germany. His research interests lie in the fields of perception, navigation, and locomotion, and how it can be used for the deployment of mobile robotic systems.

MANTHAN PATEL (Student Member, IEEE) received the B.S. degree in mechanical engineering from IIT Kharagpur, Kharagpur, India, in 2021, and the M.Sc. degree in robotics, systems and control from the Swiss Federal Institute of Technology (ETH Zürich), Zürich, Switzerland, in 2024.

He is currently a Research Engineer at the Robotic Systems Laboratory, ETH Zürich. His research interests lie in various aspects of field robotics with a particular focus on perception for robots.

DEEGAN ATHA received the B.S. degree in electrical engineering from Purdue University, West Lafayette, IN, USA, in 2017, and the M.S. degree in computer science from Georgia Institute of Technology, Atlanta, GA, USA, in 2022.

He is currently a Robotics Technologist with the Perception Systems Group, Mobility and Robotic Systems Section, NASA Jet Propulsion Laboratory, Caltech, Pasadena, CA, USA. He is currently the Perception Lead of JPL's team in the DARPA RACER project. He has previously served as a Principal Investigator for the ShadowNav task developing absolute localization methods for long-distance lunar autonomy operating in darkness. His research focuses on the infusion of robotic perception and learning into autonomous systems operating in unstructured environments.

JULIAN NUBERT (Student Member, IEEE) received the M.Sc. degree in robotics, systems and control from the Swiss Federal Institute of Technology (ETH Zürich), Zürich, Switzerland, in 2020, where he is currently pursuing the Ph.D. degree with the Robotic Systems Laboratory.

He is with the MPI ETH Center for Learning Systems, Max Planck Institute. His research interests lie in robust robot perception and how it can be used to deploy mobile robotic systems.

Mr. Nubert received the ETH Silver Medal and was awarded the Willi-Studer-Price for his accomplishments during his master's studies.

DAVID FAN (Member, IEEE) received the Ph.D. degree in robotics from Georgia Institute of Technology, Atlanta, GA, USA, in 2021.

He worked as a Research Associate at the NASA Jet Propulsion Laboratory, Caltech, Pasadena, CA, USA, from 2019 to 2023, on the DARPA Subterranean Challenge and the DARPA RACER Program. He is currently the Chief Technology Officer of Field AI, Inc., Irvine, CA, USA. He is also a Robotics Researcher and Technologist. His research covers perception, planning, and controls using machine learning techniques, with a focus on safety-critical applications and fielding robots in challenging real-world scenarios.

ALI AGHA (Member, IEEE) is a Co-Founder of Field AI, Inc., a company dedicated to advancing next-generation robotic autonomy in complex and offroad environments. Before founding Field AI, Inc., he was a Technologist and the Group Leader of the NASA Jet Propulsion Laboratory, Caltech, Pasadena, CA, USA. During his tenure at JPL, Caltech, he was the Principal Investigator and led NASA's team in several flagship autonomy-focused projects, including the DARPA Subterranean Challenge, DARPA Racer, and the prototype Mars Helicopter-Rover coordinated autonomy. Prior to his work at JPL, he was with MIT, Cambridge, MA, USA, and Qualcomm Research, San Diego, CA, USA, leading technical efforts in perception and planning for autonomous robotic vehicles.

CURTIS PADGETT received the Ph.D. degree in computer science from University California, San Diego, CA, USA, in 1997.

He is a Supervisor at the Perception Systems Group and a Principal at the Robotics Section, NASA Jet Propulsion Laboratory, Caltech, Pasadena, CA, USA, where he has worked on machine vision problems for over 30 years. He leads research efforts focused on aerial and maritime imaging problems, including navigation support for landing and proximity operations; path planning for sea surface vehicles using International Regulations for Preventing Collisions at Sea (COLREGS); automated, real-time recovery of structure from motion; precision georegistration of imagery; automated landmark generation and mapping for surface relative navigation; and stereo image sea surface sensing for navigation on water and image-based, multiplatform contact range determination. His research interests include pattern recognition, image-based reconstruction, and mapping.

PATRICK SPIELER (Member, IEEE) received the B.S. degree in microengineering and the M.S. degree in robotics and autonomous systems from the Swiss Federal Institute of Technology Lausanne (EPFL), Lausanne, Switzerland, in 2014 and 2017, respectively.

He worked at iRobot, Pasadena, CA, USA, and Astrocast, Lausanne, Switzerland, a space company building communication satellites. He was a Research Engineer with the Autonomous Robotics and Control Laboratory (ARCL), Caltech, Pasadena, CA, USA, where he led the Autonomous Flying Ambulance project and Leonardo, the first flying-walking robot. He is currently a Robotics Technologist with the Aerial Mobility Group, NASA Jet Propulsion Laboratory, Caltech. He is the Principal Investigator of the JPL team for the DARPA RACER project.

MARCO HUTTER (Member, IEEE) received the M.Sc. and Ph.D. degrees in design, actuation, and control of legged robots from the Swiss Federal Institute of Technology (ETH Zürich), Zürich, Switzerland, in 2009 and 2013, respectively.

He is currently an Associate Professor of robotic systems and the Director of the Center for Robotics, ETH Zürich. He is a Co-Founder of several ETH startups, such as ANYbotics, Zurich, Switzerland, and Gravis Robotics, Zurich. He is also the Director of Boston Dynamics AI Institute Zurich Office, Zurich. He is the Principal Investigator of the NCCRs robotics, automation, and digital fabrication. His research interests are in the development of novel machines and actuation concepts together with the underlying control, planning, and machine learning algorithms for locomotion and manipulation.

Dr. Hutter was a recipient of the ERC Starting Grant and the winner of the DARPA SubT Challenge.

SHEHRYAR KHATTAK (Member, IEEE) received the B.S. degree in mechanical engineering from GIKI, Topi, Pakistan, in 2009, the M.S. degree in aerospace engineering from Korea Advanced Institute of Science and Technology (KAIST), Daejeon, South Korea, in 2012, and the M.S. and Ph.D. degrees in computer science from the University of Nevada at Reno, Reno, NV, USA, in 2017 and 2019, respectively.

He was a Post-Doctoral Researcher at the Swiss Federal Institute of Technology (ETH Zürich), Zürich, Switzerland. He is currently a Robotics Technologist at the Perception Systems Group, NASA Jet Propulsion Laboratory (JPL), Caltech, Pasadena, CA, USA. His work focuses on enabling resilient robot autonomy in complex environments through multisensor information fusion. He is the Principal Investigator (PI) for the Multirobot Autonomous Intelligent Search and Rescue Task at JPL. He previously served as the Perception Lead for JPL's team in the DARPA RACER project and Team CERBERUS and the winner of the DARPA Subterranean Challenge.