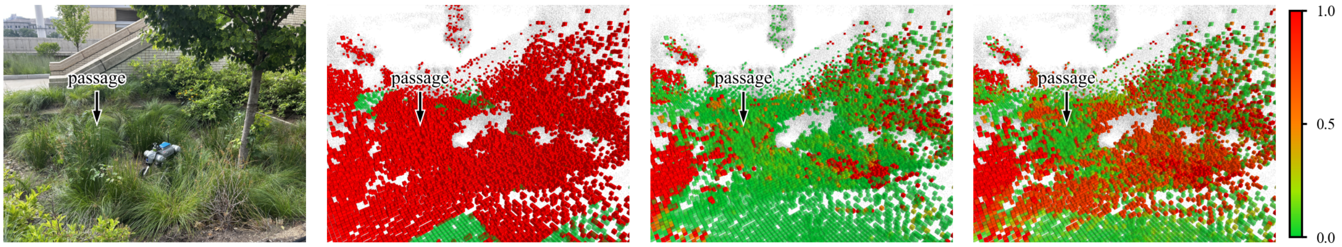
SCOUT: Semantic Class-Specific Online Update for Traversability in Densely Vegetated Environments

Seungje Seong[1], Weichen Qiu[1], Vivek Thangavelu[2], and Aaron M. Johnson[1]



(a) field site (b) geometric (height-only) (c) static-semantic (priors) (d) SCOUT (learned)

Fig. 1. Traversability cost map in dense vegetation, from one drive. (a) The field site is populated with tall grass and brush of moderate difficulty. (b)–(d) The accumulated semantic voxel cost map: (b) a height-only baseline: marks everything above the wheel radius as difficult (82.8% of voxels); (c) static priors: assign all vegetation one predefined cost; (d) SCOUT: learns per-class traversability from proprioceptive feedback. The arrows mark an opening with relatively thin grass growth, which SCOUT identifies as a lower-cost region relative to other areas of the same class based on its interaction and geometric cost assessment.

***Abstract*—Although autonomous navigation in structured urban environments has matured, navigation in unstructured, vegetated terrain remains challenging. Exteroceptive sensing alone cannot distinguish traversable vegetation from rigid obstacles, proprioception-fused approaches require extensive environment-specific pretraining, and both still struggle to drive through dense vegetation. In this abstract, we present SCOUT, a semantic-aware voxel mapping and traversability estimation system that, instead, adapts to the field. Each terrain and obstacle class carries a Gaussian belief of its traversability coefficient, initialized optimistically so ambiguous vegetation is assumed traversable at the start and its experienced cost is learned during the mission. SCOUT generates a cost map using this traversability coefficient, derived by 2D segmentation from its camera, fused with a geometric potential built from the obstacle's density and height observed by the LiDAR. With each new interaction, proprioceptive feedback recursively refines the cost of the obstacles' classes to improve its navigation. Across recorded drives, learned beliefs lift the cost of tall grass (0.37–0.53) to bush level while dirt (0.08–0.17) stays low, bringing the estimated cost closer to what the robot actually experiences on the terrain. In addition, SCOUT rates 68.6% of the map as easily traversable, against 17.2% for a height-only baseline. In these experiments, SCOUT was tested on Unitree's Go2W, but since the costs of obstacles are updated relative to each system's physical interaction with its environment, the algorithm is designed to generalize to other autonomous ground vehicles (AGVs).**

***Keywords*—Semantic-aware mapping, traversability estimation, off-road navigation, online adaptation, field robotics.**

I. Introduction

Autonomous ground vehicles (AGVs) have advanced rapidly in urban settings, but urban land covers less than 1% of the Earth's land surface [1]. The vast majority of unstructured terrain, from agricultural land to forests and natural wilderness, remains underserved by existing systems. A reliable AGV system for unstructured terrain has a wide range of applications, yet the highly variable and diverse nature of vegetative obstacles hinders accurate traversability cost (TC) assessment and a system's generalizability. The traditional approach has been to use exteroceptive sensors: LiDAR point clouds give a binary assessment of whether a space is occupied or not, but cannot distinguish tall and widespread grasses from a boulder [2], while the 2D semantics of a camera tells what an obstacle is but neglects the depth information needed to estimate its density and volume. Recent self-supervised methods fuse exteroceptive and proprioceptive modalities to learn TC through demonstration [3], [4], [5], but require environment-specific data before deployment, with collection ranging from less than half an hour [4] to tens of hours [3].

SCOUT takes the best of the three approaches and makes the estimation step adaptive and generalizable across platforms. SCOUT creates a 3D voxel map with each voxel carrying a TC, determined by $\mathrm{TC}(v) = \bar{\theta}_v \cdot f_v$, where $\bar{\theta}_v$ is the traversability coefficient of the voxel's semantic class and f_v is a geometric potential built from its observed density d_v and height h_v. During the mission, the per-class belief of the traversability coefficient is continuously updated on the basis of proprioceptive feedback from the system's interaction with its environment. Then, at update time, the traversability coefficients are multiplied by the geometric potential to estimate the TC of the voxels. Coefficients start optimistic, biasing the planner toward attempting ambiguous vegetation and learning the experienced TC from its physical constraints rather than staying away.

The contributions of this abstract are:

[1]Carnegie Mellon University, Pittsburgh, PA, USA. [2]Arizona State University, Tempe, AZ, USA.
{davidseong8914,w3ichen}@gmail.com,
vivekt@asu.edu, amj1@andrew.cmu.edu

- A parametric class-specific traversability formulation, $\mathrm{TC}(v) = \bar{\theta}_v \cdot f(d_v, h_v)$, that separates semantic identity from geometric structure and enables per-class online adaptation (Secs. III-A–III-C).
- A recursive Bayesian update on θ_c driven by wheel-slip and IMU-roughness residuals, replacing the offline self-supervised training typical of prior work with in-situ adaptation (Secs. III-D–III-E).
- A deployed system on a Unitree Go2W wheeled quadruped that integrates RGB segmentation, LiDAR-inertial odometry, voxel-level sensor fusion and a local planner (Secs. IV–V).

II. Related Work

Estimating the TC of vegetated terrain has traditionally relied on exteroceptive sensing, first on the geometry of the scene and later on its semantics, with the cost of each terrain type fixed before the robot is deployed. Recent self-supervised methods learn TC from the robot's driving and proprioceptive sensing. This section reviews these three families, which SCOUT builds on.

Geometric methods infer where the robot can go from the height, density, and volume of the structure around it, usually through 2.5D occupancy or elevation maps that record whether a unit of 3D space is free [6]. Estimating TC solely from geometry forces the estimate to be conservative, since the point clouds from dense grass and from a boulder are indistinguishable, so the planner routes around both [7].

Semantic methods segment the scene into terrain classes and attach a cost to each, distinguishing compliant vegetation from rigid obstacles where geometry cannot [2], [8], [9], [10] (Fig. 2(b) shows an example segmentation). Semantics alone ties TC to the class set that its segmentation model was trained on, and to assigned costs that stay fixed for the mission, so a dense thicket (Fig. 1(a)) reads the same as a few thin stems.

Proprioceptive, self-supervised methods learn the cost of the environment by mapping visual features to proprioceptive feedback [3], [4], [5], with recent work adding evidential uncertainty [11], [12], stability-aware risk [13], visual friction learning [14], and online adaptation [15], [16], [17], grounding the cost in what the robot experiences rather than in an operator's prior. Their supervision is drawn largely from where the robot has already been driven, usually under teleoperation, so the model reproduces the routes it was shown and has little reason to test terrain the operator avoided. ForestTrav [18] and its online successor [19], the closest approaches to ours, keep a persistent 3D voxel map. The successor adapts it through interaction, but updates by LiDAR feature similarity rather than by what an obstacle is, and an operator stages and hand-labels each collision.

Geometric and semantic methods fix the cost of vegetation before the robot touches it; self-supervised approaches revise it in the field, but mostly from demonstrations that confine them to terrain the operator was already willing to cross, so the model never gets feedback on vegetation the operator avoided and stays just as conservative about it, whether or not it is actually traversable. SCOUT leaves each semantic class's cost susceptible to change, so more interactions with a class during a mission constantly refine the TC and adjust the costs of already mapped and future obstacles closer to the experienced TC.

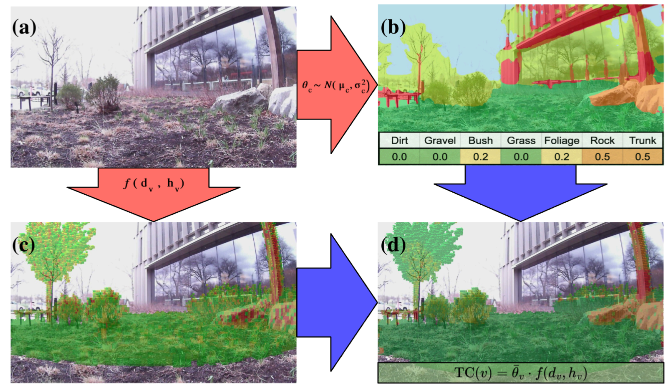


Fig. 2. SCOUT perception pipeline. (a) Raw RGB input. (b) Semantic segmentation with optimistic class priors $\theta_c \sim \mathcal{N}(\mu_c, \sigma_c^2)$; color-coded overlays and the table show initial μ_c values per class. (c) Geometric potential $f(d_v, h_v)$ projected onto the scene as a voxel cost overlay (red = high, green = low). (d) Final cost map after multiplying the class traversability coefficient by the geometric potential, $\mathrm{TC}(v) = \bar{\theta}_v \cdot f(d_v, h_v)$. Grass and dirt voxels are suppressed relative to the geometric-only map; rigid structures retain their full cost.

III. Adaptive Semantic Traversability Mapping

A. Semantic Voxel Mapping

LiDAR scans registered by LiDAR-inertial odometry [20] accumulate into a semantic voxel map [21], [22], voxelized at $r = 0.1\,\mathrm{m}$ in a sliding window. During the drive, each point in the returned point cloud is projected into the corresponding 2D segmentation mask (15 WildScenes [23] classes from DDRNet-23-slim [24]) and votes its class into a voxel of the TC map.

Each voxel stores only its geometric features and semantic class label, never a baked-in cost. Each time the map is output to the planner, a voxel's coefficient $\bar{\theta}_v$ is set to the current belief mean μ_c of its class (Sec. III-C), rather than a value fixed when the voxel was first inserted. Because every voxel of a class shares that same μ_c, revising one class's belief changes the cost of every voxel of that class already in the map the next time the map is output, with no per-voxel update step (validated in Sec. V).

B. Geometric Potential

The geometric potential encodes how much cost a voxel's physical structure would contribute if its class were fully obstructive:

$$d_v = \begin{cases} 1, & \bar{\theta}_v \geq \tau \\ \min(n_v/n_{\min},\, 1)\,, & \text{otherwise,} \end{cases} \tag{1}$$

$$f_v = d_v \left[\alpha_0 + (1-\alpha_0)\min(h_v/h_{\max},\, 1)\right]. \tag{2}$$

Here $d_v \in [0, 1]$ describes the density of a voxel: a voxel whose class already reads as obstacle-like ($\bar{\theta}_v \geq \tau$, τ=0.5) is treated as fully dense regardless of how few points it received, while a nominally safe voxel is trusted as dense only once its point count n_v reaches $n_{\min}$=5. This gate couples semantics to geometry. While a class's learned mean exceeds τ, its voxels forgo the sparsity discount and differentiate by height alone. The bracketed term in (2) is the height factor: it holds at a ground-level floor α_0=0.3 and rises linearly to 1 as height h_v approaches the ceiling $h_{\max}$, the wheel radius (0.09 m), with $h_v = \max(z_v - g(x_v, y_v),\, 0)$ above the local ground estimate g (minimum z within each 2 m cell). A voxel at ground level keeps only α_0 of its class cost, since the wheel can roll over this height. The per-voxel cost $\mathrm{TC}(v) = \bar{\theta}_v \cdot f_v$ (Fig. 2) is then flattened to a 2D grid by column-wise maximum for the planner [25], [26].

C. Per-Class Beliefs and Optimistic Priors

Each class c carries $\theta_c \sim \mathcal{N}(\mu_c, \sigma_c^2)$, with θ_c clipped to $[0, 1]$. When $\theta_c = 0$ the class contributes no cost regardless of geometry. When $\theta_c = 1$ it preserves the full geometric potential. θ_c is not a physical parameter of vegetation dynamics (cf. [27]). It compresses stiffness, traction, and ride roughness into a single scalar that a planner consumes. That scalar is the difficulty this robot experiences with the class. Vegetation and terrain classes start optimistic, with a lower μ_c meaning less cost and more traversability: grass, dirt, gravel, mud, and other-terrain at μ_c=0; bush and tree-foliage at 0.2; log, rock, and tree-trunk at 0.5. Each belief starts with variance $\sigma_0^2 = 0.04$.

D. Proprioceptive Measurement Model

SCOUT forms a scalar cost observation from two residuals over non-overlapping windows of $T = 0.5\,\mathrm{s}$:

$$r^{\text{slip}} = 1 - \min(\bar{v}_{\text{body}}/\bar{v}_{\text{wheel}},\ 1)\,, \tag{3}$$

$$r^{\text{imu}} = \operatorname{clip}\left(\frac{\operatorname{std}(a_z)_T - b}{a_{\max} - b},\ 0,\ 1\right), \tag{4}$$

$$z = \lambda_v \left(r^{\text{slip}}\right)^{\gamma} + \lambda_i \left(r^{\text{imu}}\right)^{\gamma}, \tag{5}$$

with z the resulting scalar cost observation for the window, $\bar{v}_{\text{body}}$ the window's displacement divided by its duration, from LiDAR-inertial odometry, $\bar{v}_{\text{wheel}}$ the wheel surface speed from encoders, a_z vertical acceleration, and λ_v the weight given to the slip residual against $\lambda_i = 1 - \lambda_v$ given to the IMU residual (λ_v=0.6). Calibrated on the platform ($b = 1.2\,\mathrm{m\,s^{-2}}$, the median driving vibration on hard dirt and gravel; $a_{\max} = 4.5\,\mathrm{m\,s^{-2}}$), r^{imu} measures the excess vibration over nominal driving rather than absolute vibration. The exponent γ=2 rewards partial progress. A small residual from slip the robot mostly pushed through counts for much less than a residual near 1 from a near-complete stall. Three validity gates reject windows whose slip ratio cannot carry terrain signal: (i) *wheels driving* ($\bar{v}_{\text{wheel}} > 0.3\,\mathrm{m\,s^{-1}}$; odometry noise dominates at crawl), (ii) *near-straight travel* (rotation in place can read as slip), and (iii) *steady command* (a discrepancy between the commanded speed and the physical actuation can read as slip).

Each observation is attributed to the count-dominant class of map voxels in a single 0.4 m-radius cylinder centered on the robot body and vertically anchored to the wheel-contact plane rather than the body origin, keeping the window on the rolling surface instead of reaching into the canopy beside the robot; a window straddling a class boundary assigns all of its evidence to the majority class.

E. Recursive Belief Update

Each windowed observation z updates only the attributed class c via the scalar Gaussian filter [28], with gain $K = \sigma_c^2/(\sigma_c^2 + \sigma_z^2)$, where σ_z^2 is the measurement noise:

$$\mu_c \leftarrow \mu_c + K(z - \mu_c), \tag{6}$$

$$\sigma_c^2 \leftarrow \max\left((1 - K)\,\sigma_c^2,\ \sigma_{\min}^2\right). \tag{7}$$

With $\sigma_0^2 = \sigma_z^2 = 0.04$, the first adverse encounter with an optimistic class has $K = 0.5$, so a window with full slip and heavy vibration, $z \approx 0.8$, moves a 0.2-prior class by $\Delta\mu \approx 0.3$, with no curated dataset. As evidence accumulates, K shrinks and the estimate resists transient spikes. A variance floor of $\sigma_{\min}^2$=0.005 keeps every class revisable; as a lightweight substitute for change-point detection, a residual exceeding $3\sigma_c$ re-inflates the variance to σ_0^2 before the update, letting a drifted class (wet grass, seasonal change) re-learn.

IV. Platform and Field Dataset

SCOUT's perception, mapping, cost-update and planning stack runs fully onboard a Unitree Go2W wheeled quadruped (Livox MID-360 LiDAR, RGB camera, IMU, wheel encoders, NVIDIA Jetson Orin, ROS 2). FAST-LIO2 odometry [20], mapping, and segmentation run as asynchronous loops. Each LiDAR increment is fused against the most recent 2D segmentation mask at its capture stamp, so a slow segmenter reduces vote density rather than misaligning labels.

The evaluation dataset comprises four teleoperated drives at a single field site, through mixed grass, brush, and tree cover adjacent to rigid structures (a fifth 52 s recording is excluded as too short).

V. Validation and Preliminary Results

A. Belief Convergence

Fig. 3 shows per-class belief traces $\mu_c(t)$ from replaying the measurement model of Sec. III-D over recorded drives, with hindsight attribution against the accumulated map. Over four drives, fully gated bush-attributed windows show a higher slip residual (median 0.66, n=7) than grass-attributed ones (median 0.60, n=39), though the bush sample is small and seen during only one drive. The grass in the test environment carried substantial slip, which is why the learned μ raises the grass to the bush-level cost despite its lower residual per-window. On the other hand, the site's shorter brush let the robot push through, so a moderate bush cost is consistent.

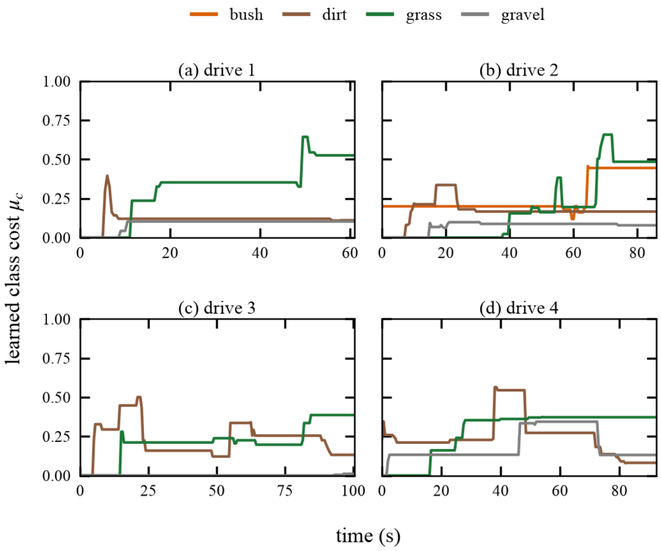


Fig. 3. Belief traces $\mu_c(t)$ from proprioceptive replay of four drives. In (b), the thicket entanglement corrects bush $0.19 \rightarrow 0.46$ in one window (final 0.44 ± 0.12); grass converges to 0.48 ± 0.14 from real drag in the tall grass plus footprint attribution at the thicket boundary. The early dirt spike, corrected as benign observations accumulate, demonstrates the drift mechanism. Across drives and without hand-tuning, grass settles at 0.37–0.53, dirt at 0.08–0.17, and gravel near free.

B. Cost Ablation

Three configurations share perception, mapping, and geometry: *geometric* (height-only, where cost is height above local ground divided by the wheel radius, saturating at 1, so flat ground is free), *static-semantic* (frozen priors), and *SCOUT* (post-update beliefs from the Fig. 3 replay). The semantic configurations differ only by swapping the per-class means μ_c in $\bar{\theta}_v$, so their comparison is exact on identical map data. The geometric baseline drops the α_0 floor rather than setting $\theta_c \equiv 1$, which would deny it easy voxels by construction. Fig. 1(b)–(d) shows the three maps. Over the 29,744 mapped voxels (easy < 0.2, moderate $[0.2, 0.8)$, difficult ≥ 0.8), geometric marks 17.2%/0.0%/82.8% easy/moderate/difficult, effectively binary, since one voxel layer exceeds the $0.09\,\mathrm{m}$ wheel radius. The static priors give 76.0% / 10.1% / 13.8%, and SCOUT 68.6% / 17.3% / 14.2%. The learned beliefs shift the mass from easy into moderate, as the measured drag demands; whether this shift improves navigation is the closed-loop question of Sec. VI.

VI. Limitations and Ongoing Work

Running the full stack on the Go2W's onboard compute bounds both model size and rate. The current segmenter, DDRNet-23-slim, is a small 512×512-input model on a low-rate, low-resolution camera, and the results in Figs. 1 and 3 are bound by these limitations. The brush labeled as grass could penalize the grass, inflating σ_c^2. SCOUT has so far been validated under manual operation, which has shown that it learns and updates the TC of obstacles and terrain. It has not yet been tested in closed-loop autonomous trials, where SCOUT's own planner drives the robot using the cost map, or in a wider range of vegetation densities and varieties.

VII. Conclusion

SCOUT is a step toward autonomous exploratory navigation in unseen vegetated environments. It generates a cost map fusing 2D semantics with 3D geometry, with semantic class costs updated by physical interaction, whereas prior approaches either lean on one modality and forfeit accuracy or adaptivity or require learning from operator's demonstration. Future work rightsizes compute and moves to closed-loop trials in denser vegetation.

References

[1] X. Liu, Y. Huang, X. Xu, et al., "High-spatiotemporal-resolution mapping of global urban change from 1985 to 2015," *Nature Sustainability*, vol. 3, no. 7, pp. 564–570, 2020.

[2] A. J. Sathyamoorthy, K. Weerakoon, T. Guan, M. Russell, D. Conover, J. Pusey, and D. Manocha, "VERN: Vegetation-aware robot navigation in dense unstructured outdoor environments," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, 2023.

[3] G. Kahn, P. Abbeel, and S. Levine, "BADGR: An autonomous self-supervised learning-based navigation system," *IEEE Robot. Autom. Lett.*, vol. 6, no. 2, 2021.

[4] 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)*, 2022.

[5] M. V. Gasparino, A. N. Sivakumar, Y. Liu, A. E. B. Velasquez, V. A. H. Higuti, J. Rogers, H. Tran, and G. Chowdhary, "WayFAST: Navigation with predictive traversability in the field," *IEEE Robot. Autom. Lett.*, vol. 7, no. 4, 2022.

[6] K. Weerakoon, A. J. Sathyamoorthy, U. Patel, and D. Manocha, "TERP: Reliable planning in uneven outdoor environments using deep reinforcement learning," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, 2022.

[7] Z. Jian, Z. Liu, H. Shao, X. Wang, X. Chen, and B. Liang, "Path generation for wheeled robots autonomous navigation on vegetated terrain," *IEEE Robot. Autom. Lett.*, vol. 9, no. 2, pp. 1764–1771, 2024.

[8] T. Guan, D. Kothandaraman, R. Chandra, A. J. Sathyamoorthy, K. Weerakoon, and D. Manocha, "GA-Nav: Efficient terrain segmentation for robot navigation in unstructured outdoor environments," *IEEE Robot. Autom. Lett.*, vol. 7, no. 3, 2022.

[9] S. Matsuzaki, H. Masuzawa, and J. Miura, "Image-based scene recognition for robot navigation considering traversable plants and its manual annotation-free training," *IEEE Access*, vol. 10, pp. 5115–5128, 2022.

[10] S. Gummadi, M. V. Gasparino, G. Capezzuto, M. Becker, and G. Chowdhary, "ZeST: An LLM-based zero-shot traversability navigation for unknown environments," arXiv:2508.19131, 2025.

[11] X. Cai, S. Ancha, L. Sharma, P. R. Osteen, B. Bucher, S. Phillips, J. Wang, M. Everett, N. Roy, and J. P. How, "EVORA: Deep evidential traversability learning for risk-aware off-road autonomy," *IEEE Trans. Robot.*, vol. 40, pp. 3756–3777, 2024.

[12] X. Cai, J. Queeney, T. Xu, A. Datar, C. Pan, M. Miller, A. Flather, P. R. Osteen, N. Roy, X. Xiao, and J. P. How, "PIETRA: Physics-informed evidential learning for traversing out-of-distribution terrain," *IEEE Robot. Autom. Lett.*, vol. 10, no. 3, pp. 2359–2366, 2025.

[13] Z. Yoon, L. Y. Zhu, J. Lu, L. Gan, and Y. Zhao, "STATE-NAV: Stability-aware traversability estimation for bipedal navigation on rough terrain," *IEEE Robot. Autom. Lett.*, vol. 11, no. 2, pp. 2338–2345, 2026.

[14] T. Fu, Z. Zhan, Z. Zhao, Y. Du, S. Su, X. Lin, E. T. Esfahani, K. Dantu, S. Chowdhury, and C. Wang, "AnyNav: Visual neuro-symbolic friction learning for off-road navigation," arXiv:2501.12654, 2025.

[15] M. Sivaprakasam, S. Triest, C. Ho, S. Aich, J. Lew, I. Adu, W. Wang, and S. Scherer, "SALON: Self-supervised adaptive learning for off-road navigation," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, 2025.

[16] J. Hindel, S. Bultmann, H. Masnavi, D. Cattaneo, and A. Valada, "Self-supervised online robot-agnostic traversability estimation for open-world environments," arXiv:2605.28442, 2026.

[17] M. Elnoor, K. Weerakoon, G. Seneviratne, R. Xian, T. Guan, M. K. M. Jaffar, V. Rajagopal, and D. Manocha, "VLM-GroNav: Robot navigation using physically grounded vision-language models in outdoor environments," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, 2025.

[18] F. A. Ruetz, N. Lawrance, E. Hernández, P. V. K. Borges, and T. Peynot, "ForestTrav: 3D LiDAR-only forest traversability estimation for autonomous ground vehicles," *IEEE Access*, vol. 12, pp. 37192–37206, 2024.

[19] F. A. Ruetz, N. Lawrance, E. Hernández, P. V. K. Borges, and T. Peynot, "Online adaptive traversability estimation through interaction for unstructured, densely vegetated environments," *IEEE Trans. Field Robot.*, vol. 3, pp. 1–21, 2026.

[20] W. Xu, Y. Cai, D. He, J. Lin, and F. Zhang, "FAST-LIO2: Fast direct LiDAR-inertial odometry," *IEEE Trans. Robot.*, vol. 38, no. 4, 2022.

[21] J. McCormac, A. Handa, A. Davison, and S. Leutenegger, "SemanticFusion: Dense 3D semantic mapping with convolutional neural networks," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, 2017.

[22] A. Rosinol, M. Abate, Y. Chang, and L. Carlone, "Kimera: An open-source library for real-time metric-semantic localization and mapping," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, 2020.

[23] K. Vidanapathirana, J. Knights, S. Hausler, M. Cox, M. Ramezani, J. Jooste, E. Griffiths, S. Mohamed, S. Sridharan, C. Fookes, and P. Moghadam, "WildScenes: A benchmark for 2D and 3D semantic segmentation in large-scale natural environments," *Int. J. Robot. Res.*, vol. 44, no. 4, pp. 532–549, 2025.

[24] H. Pan, Y. Hong, W. Sun, and Y. Jia, "Deep dual-resolution networks for real-time and accurate semantic segmentation of traffic scenes," *IEEE Trans. Intell. Transp. Syst.*, vol. 24, no. 3, pp. 3448–3460, 2023.

[25] S. Macenski, F. Martín, R. White, and J. G. Clavero, "The Marathon 2: A navigation system," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, 2020.

[26] S. Macenski, S. Singh, F. Martín, and J. Ginés, "Regulated pure pursuit for robot path tracking," *Autonomous Robots*, vol. 47, no. 6, pp. 685–694, 2023.

[27] C. Ordonez, R. Alicea, B. Rothrock, K. Ladyko, J. Nash, R. Thakker, S. Daftry, M. Harper, E. Collins, and L. Matthies, "Characterization and traversal of pliable vegetation for robot navigation," in *Proc. Int. Symp. Experimental Robotics (ISER)*, 2018.

[28] S. Thrun, W. Burgard, and D. Fox, *Probabilistic Robotics*. MIT Press, 2005.