Robust 2D Traversability Mapping for Construction AMRs via Failure-Mode-Aware Fusion of LiDAR Geometry and Monocular Semantics

Manoj Karnekar[1], Om Mandhane[2], and Gautham Ramkumar[3]

***Abstract*— Autonomous Mobile Robots (AMRs) on active construction sites face severe navigational challenges: geometry-based traversability mapping (e.g., LiDAR) misses visually hazardous but geometrically flat surfaces like wet mud and ponding concrete, while abrupt geometry on drivable speed-breakers and inclines produces phantom obstacles. We propose a real-time, failure-mode-aware multimodal traversability pipeline on an NVIDIA Jetson AGX Orin, where LiDAR is the primary geometric safety estimate and monocular semantics act as a selective, class- and confidence-gated corrective signal. The representation retains distinct traversable classes, namely flat road, terrain, and rocky terrain, while flagging construction hazards. We also release a multimodal construction-site dataset from a custom AMR: four closed-loop ROS 2 sequences from two active sites (RGB, depth, LiDAR, IMU, GPS-RTK, odometry) plus 506 annotated frames across 28 semantic classes. By projecting LiDAR onto dense semantic masks, resolving sparsity via morphological in-painting, and applying failure-mode-aware fusion with Patchwork++, the system corrects complementary geometric failure modes for a local AMR costmap.**

***Index Terms*—Traversability mapping, sensor fusion, semantics, LiDAR, construction robotics.**

I. INTRODUCTION

AMR deployment on active construction sites is hindered by unstructured terrain: amorphous hazards (deep mud, ponding concrete) and transient structures (scaffolding, rebar, temporary planks) violate structured-indoor-navigation assumptions. AMRs conventionally rely on 3D LiDAR for ground segmentation (e.g., Patchwork++ [1]), which suffers two complementary failure modes here: (1) false negatives, where flat non-load-bearing hazards like wet mud or ponding concrete read as safe ground; and (2) false positives, where the sharp negative gradient at a speed-breaker's or incline's trailing edge is misread as a lethal obstacle, causing navigational freezing. Purely semantic (camera-based) approaches address recognition but lack the metric reliability collision avoidance needs [3]–[7].

We propose a failure-mode-aware semantic-geometric fusion architecture where LiDAR remains the primary safety estimate and a YOLOv26-m-seg model [8] provides class- and confidence-gated corrective evidence, overriding geometry only for identified failure modes. Rather than learn the final traversability decision end-to-end [6], [9], we use learning only for semantic recognition and keep sensor arbitration an explicit, auditable, deterministic policy. This suits both our 506-image annotated set, which supervises recognition directly without a separate learned traversability target, and the per-decision verifiability a safety-relevant system needs.

Contributions: (1) a deterministic, class- and confidence-gated sensor-arbitration strategy using semantic vision as a selective corrective signal for complementary LiDAR failure modes (flat hazards, phantom obstacles), with LiDAR as fallback; (2) the public multimodal construction-site dataset detailed in Section III. We validate the framework via cross-site evaluation, training on one site and testing perception and fusion jointly on an unseen site.

II. RELATED WORK

Prior fusion work combines visual and geometric terrain cues [3], [4], while recent learned approaches (BEVFusion [5], RoadRunner [7], METAVerse [9]) learn multimodal or traversability representations end-to-end. TNS [13] is closest to our work, fusing elevation-grid slope/step-height features with RGB semantics to zero traversability for non-traversable categories on a 49-ton excavator. Where TNS applies a uniform semantic override, our fusion logic is asymmetric, failure-mode-specific, and confidence-gated, tailored to the unstructured hazards of active building sites: high-confidence flat hazards override geometrically clear LiDAR cells, high-confidence traversable classes suppress geometric false positives from speed-breakers and ramps, and ambiguous classes (e.g., puddles) defer to LiDAR. We also preserve flat road, terrain, and rocky terrain as distinct states rather than a single traversable class, enabling surface-specific navigation costs.

Geometric approaches such as Patchwork++ [1] give robust ground-plane extraction from LiDAR but cannot distinguish flat non-load-bearing hazards from safe ground, nor drivable elevation changes from lethal obstacles. Off-road semantic segmentation datasets such as RUGD [11] and RELLIS-3D [12] extend the Cityscapes paradigm [10] to unstructured environments, but do not capture hazards specific to construction sites, while TNS's Complex Worksite Terrain (CWT) dataset [13] targets excavators with 669 RGB-only images and seven categories. Our release instead focuses on live, active building construction sites and centers on a raw multimodal rosbag corpus (four closed-loop sequences: RGB, depth, LiDAR, IMU, GPS-RTK, odometry) plus a 28-class

[1]Manoj Karnekar is Robotics Lead at FloMobility Pvt. Ltd., Bangalore, India manojkarnekar1@gmail.com

[2]Om Mandhane is a Robotics Engineer at FloMobility Pvt. Ltd., Bangalore, India ommandhane27@gmail.com

[3]Gautham Ramkumar is a Robotics Engineer at FloMobility Pvt. Ltd., Bangalore, India gauthamramkumar03@gmail.com

Fig. 1. The custom AMR platform used for multimodal data collection, purpose-built for autonomous navigation in construction environments.

annotated subset, supporting fusion, mapping, loop-closure, and localization research along with segmentation.

III. CONSTRUCTION-SITE MULTIMODAL DATASET

The data was gathered at two active construction sites (Site 1, Site 2) using a custom AMR (Fig. 1) equipped with an OAK-D PoE Pro RGB-D camera, a Livox Mid-360 LiDAR, IMU, GPS-RTK, and odometry, and stored in ROS 2 MCAP bags for a total of four closed-loop sessions (2 per site; $>$1.75 h). In addition to the 506 annotated frames used here, the raw corpus contains spatial regions that can be revisited and temporal relationships between streams that can be maintained; it can also be used for point-cloud generation, SLAM/loop-closure, GPS-RTK localization, sensor calibration studies, and more hazard-class coverage. A perceptual-hash keyframe extractor is used to extract key frames by selecting those with high variance in RGB frames and retaining a candidate iff $\min(D_H(h_c, h_s)) \geq \tau$ where h_c is the candidate and h_s is a previously saved hash and τ=14. An automated hybrid pipeline of SAM-3 [14], [15] (for hazard mask proposal), YOLO [8] (for background classes), and human validation was used to add 506 keyframes to the $>$3.06M messages (415 train / 91 val). The training data is only from site 1, and site 2 is reserved for evaluation on the zero-shot spatial-generalization task. The dataset consists of rosbags, calibration data, and annotated masks, and is available at `https://huggingface.co/datasets/manojkarnekar/construction-traversability-dataset` [16]. To address privacy concerns, people, text, and vehicle license plates are blurred before release.

The 28-class taxonomy assigns each class a fusion role (Table I): three *traversable* classes (flat road, terrain, rocky terrain); *flat-hazard override* classes (wet mud, ponding concrete) that can override geometrically clear LiDAR cells; a *geometry-dependent* class (puddle) confirming water but deferring depth to LiDAR; *camera-ignored* classes (sky, not visible) that always fall back to LiDAR; and 20 remaining *non-traversable* classes (e.g., rebar, scaffolding, debris, pipes, vehicles, personnel).

TABLE I
28-CLASS TAXONOMY AND FUSION ROLES (ABRIDGED)

Fusion role	**Semantic classes**
Traversable	flat road, terrain, rocky terrain
Flat-hazard override	wet mud, ponding concrete
Geometry-dep. hazard	puddle
Camera-ignored	not visible, sky
Non-traversable (20 classes)	animal, building, ceiling, concrete blocks, machinery, debris, fence, material pile, person, pillar, pipes, pit, pole, rebar, scaffolding, tiles, vegetation, vehicle, wall, wooden planks

IV. SYSTEM ARCHITECTURE AND METHODOLOGY

The system runs a dual-stream pipeline into a unified 2D occupancy grid for the AMR's navigation stack [2]. The *semantic branch* (YOLOv26-m-seg [8]) runs on the Jetson AGX Orin: we take the semantic head's logits, bilinearly upsample them to full image resolution, and apply a softmax over the 28 classes, so every pixel carries a class label together with its posterior probability as a confidence value. The *geometric branch* (Patchwork++ [1], tuned for the Livox Mid-360's 23° mounting tilt) segments the cloud into ground and non-ground returns; non-ground returns are rasterized into the grid as geometric obstacles. RGB and LiDAR messages are paired by header timestamp with a 0.1 s tolerance.

A LiDAR point P_{bf} in the robot base frame is transformed to the camera frame via T_{cam}^{lidar} and projected using the pinhole model,

$$u = \left\lfloor f_x \frac{X}{Z} + c_x \right\rfloor, \; v = \left\lfloor f_y \frac{Y}{Z} + c_y \right\rfloor, \tag{1}$$

inheriting the semantic label and confidence at pixel (u, v) when $Z > 0$ and (u, v) is in-bounds, then flattened into a 200×200, 0.1 m local grid (20×20 m, robot-centered in `base_footprint`) initialized to Unknown (-1). Returns above the robot height (1.30 m) are discarded, and where several points share a cell the highest-confidence label wins. Since sparse LiDAR rings leave gaps, we resolve this without computationally heavy neural upsampling: for each Unknown cell we count confident same-tier hits in a 9×9 neighborhood and fill the cell once support is sufficient, requiring one hit for flat road and two for terrain/rocky terrain, at a confidence just above the corresponding override threshold. The stricter tier-2 rule keeps in-painting from propagating the weaker-evidence surface classes.

Fusion applies this class- and confidence-gated arbitration as a deterministic, auditable decision tree per cell: (1) Sky/Not Visible $\rightarrow$ LiDAR only; (2) Wet Mud/Ponding Concrete with conf $>$ 0.60 $\rightarrow$ force Obstacle, below which the cell defers to LiDAR; (3) Puddle $\rightarrow$ confirm water, defer depth decision to LiDAR; (4) sensor agreement $\rightarrow$ output as-is; (5) LiDAR Clear + camera non-traversable with conf $>$ 0.80 $\rightarrow$ camera overrides; (6) LiDAR Obstacle + camera Flat

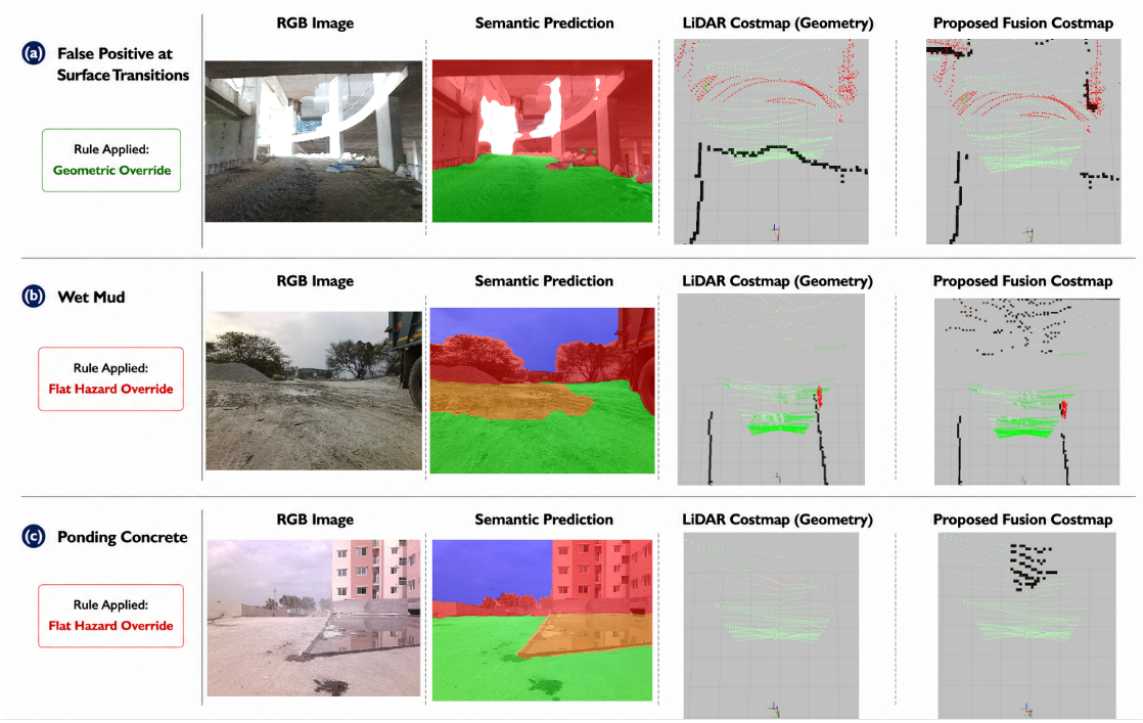

Fig. 2. Qualitative fusion results. (a) LiDAR falsely flags a traversable surface transition, such as a slope or an incline, as an obstacle; the flat road prediction removes it (geometric override). (b–c) Wet mud and ponding concrete are geometrically flat but semantically identified, triggering the flat-hazard override.

Road (conf $>$ 0.40) or Terrain/Rocky Terrain (conf $>$ 0.50) $\rightarrow$ output Clear; (7) the label grid keeps traversable classes distinct for downstream cost/speed assignment; (8) all other cases fall back to LiDAR.

V. EXPERIMENTS AND RESULTS

The YOLOv26-m-seg model, fine-tuned on 415/91 train/val images, achieved an mIoU of 0.579 and a pixel accuracy of 82.4%. Structurally dominant classes performed well (Sky 0.953, Vegetation 0.858, Flat Road 0.816 mIoU), while rare hazards suffered from sparsity (Wet Mud 0.188 mIoU). Low mIoU on these large, amorphous regions is dominated by boundary error, whereas the flat-hazard override only needs confident interior cells; this is why Wet Mud still triggers reliably downstream (7/7 cases across Tables II and III) despite its weak segmentation score.

To quantify real-time feasibility on the target hardware, we benchmarked standalone YOLOv26-m-seg inference at 640×640 (500 iterations after warm-up), yielding a mean latency of 23.18 ms (P95 23.93 ms, P99 25.49 ms, $\sim$43 FPS). Across the full integrated pipeline—YOLO inference, LiDAR projection, and fusion, over 84 valid ROS 2 samples—mean end-to-end costmap latency was 88.96 ms (P95 111.60 ms, P99 118.95 ms), so the tail does exceed the 100 ms cycle time. This does not stall the planner: the fusion node publishes on a fixed 10 Hz timer decoupled from the perception callbacks, so a late cycle republishes with a marginally staler semantic layer rather than a missed costmap, and a 0.3 s staleness guard flags the case where the geometric and semantic grids drift further apart than that.

Fig. 2 illustrates both corrective mechanisms: geometric override removes phantom obstacles at traversable transitions, and flat-hazard override assigns non-traversable cost to visually identifiable but geometrically flat hazards. Together, these highlight the asymmetric nature of the fusion strategy: semantic evidence is used selectively to correct known geometric failure modes. At the same time, LiDAR remains the primary safety estimate for unresolved cases. The primary evaluation is cross-site: the semantic branch trains only on Site 1, and the full system is scored on unseen Site 2 (Table III); Table II additionally reports same-site Site 1 cases per failure mode. Both tables confirm that the fusion engine resolves most targeted failure modes (inclined slopes, speedbreakers, wet mud, ponding concrete, debris, rebar, pipes) against the ground truth, with the camera column showing that the raw semantic prediction fusion arbitrates against LiDAR.

Limitation: On unseen Site 2, Rocky Terrain was frequently misclassified as Debris (N=11), likely due to overlapping visual texture between loose rock and construction debris, compounded by Rocky Terrain being underrepresented in Site 1-only training. LiDAR correctly read these cells as traversable, but Rule 5 is designed to trust high-confidence semantic evidence over LiDAR in exactly this geometry-clear, camera-flags-hazard scenario, so the high-confidence Debris prediction triggered the override and reported a false obstacle—the inverse of the intended correction, where the classification error propagates through the fusion policy rather than being suppressed by it. Debris also absorbs other similar hazards (e.g., *Big puddle*), acting as a catch-all class; this

TABLE II

FMEA VALIDATION – SITE 1 (PRIMARY/TRAINING)

Scenario	N	Camera	LiDAR	GT	Fused
Inclined slope	6	Flat Road	NT	T	**T**
Speedbreaker	8	Flat Road	NT	T	**T**
Random LiDAR fail	5	Flat Road	NT	T	**T**
Debris	7	Debris	T	NT	**NT**
Metal pipe stack	4	Pipes	T	NT	**NT**
Concrete block	3	Debris	T	NT	**NT**
Weak wooden plank	1	Wooden Plank	T	NT	**NT**
Rebar	3	Rebar	T	NT	**NT**
Ponding concrete	2	Ponding Conc.	T	NT	**NT**
Bushes/vegetation	3	Vegetation	T	NT	**NT**
Animal	1	Animal	T	NT	**NT**
Wet mud	3	Wet Mud	T	NT	**NT**

T = Traversable, NT = Non-Traversable.

TABLE III

FMEA VALIDATION – SITE 2 (UNSEEN)

Scenario	N	Camera	LiDAR	GT	Fused
Wet mud	4	Wet Mud	T	NT	**NT**
Inclined slope	11	Flat Road	NT	T	**T**
Big puddle	1	Debris	T	NT	**NT**
Vegetation	3	Vegetation	T	NT	**NT**
Debris/rock	9	Debris	T	NT	**NT**
Stack of pipes	9	Pipes	T	NT	**NT**
Rebar stack	2	Rebar	T	NT	**NT**
Rocky road	1	Rocky Terrain	NT	T	**T**
Rocky terr. (misclass.)	11	Debris	T	T	NT (fail)

T = Traversable, NT = Non-Traversable.

points to a need for cross-site Rocky Terrain (and broader hazard-class) annotations, a use case for the released raw rosbag corpus. We also note the scope of this evaluation: the scenarios in both tables were selected because LiDAR alone fails on them, so they characterize failure-mode coverage rather than a general comparison against camera-only or uniform-override policies. Cell-level costmap metrics over full frames, sensitivity analysis on the gating thresholds, robustness under degraded vision (dust, low light, glare), and closed-loop navigation trials remain future work.

VI. CONCLUSION

We presented a failure-mode-aware multimodal traversability system for construction AMRs that treats LiDAR and semantic vision as complementary rather than interchangeable: a deterministic, class- and confidence-gated fusion engine rejects visually hazardous but geometrically flat hazards and suppresses phantom obstacles on drivable elevation changes, while preserving flat road, terrain, and rocky terrain as distinct surface categories for downstream cost assignment. We also release a multimodal construction-site dataset—four closed-loop ROS 2 sequences from two active sites plus a curated 506-frame, 28-class annotated subset—to support further construction perception, fusion, and mapping research. The complete system runs in real time on an NVIDIA Jetson AGX Orin, integrated with a custom AMR navigation stack.

REFERENCES

[1] S. Lee, H. Lim, and H. Myung, "Patchwork++: Fast and robust ground segmentation solving partial under-segmentation using 3d point cloud," *arXiv:2207.11919*, 2022.

[2] D. V. Lu, D. Hershberger, and W. D. Smart, "Layered costmaps for context-sensitive navigation," in *IROS*, 2014, pp. 709–715.

[3] F. Schilling, X. Chen, J. Folkesson, and P. Jensfelt, "Geometric and visual terrain classification for autonomous mobile navigation," in *IROS*, 2017, pp. 2678–2684.

[4] L. Zhou, J. Wang, S. Lin, and Z. Chen, "Terrain traversability mapping based on lidar and camera fusion," in *ICARA*, 2022, pp. 217–222.

[5] Z. Liu et al., "Bevfusion: Multi-task multi-sensor fusion with unified bird's-eye view representation," in *ICRA*, 2023, pp. 2774–2781.

[6] Q. Zhu et al., "Learning-based traversability costmap for autonomous off-road navigation," *arXiv:2406.08187*, 2024.

[7] J. Frey et al., "Roadrunner: Learning traversability estimation for autonomous off-road driving," *arXiv:2402.19341*, 2024.

[8] Ultralytics, "Ultralytics yolo26: Unified real-time end-to-end vision models," *arXiv:2606.03748*, 2026.

[9] J. Seo, T. Kim, S. Ahn, and K. Kwak, "Metaverse: Meta-learning traversability cost map for off-road navigation," *arXiv:2307.13991*, 2023.

[10] M. Cordts et al., "The cityscapes dataset for semantic urban scene understanding," in *CVPR*, 2016, pp. 3213–3223.

[11] M. Wigness et al., "A rugd dataset for autonomous navigation and visual perception in unstructured outdoor environments," in *IROS*, 2019, pp. 5000–5007.

[12] P. Jiang, P. Osteen, M. Wigness, and S. Saripalli, "Rellis-3d dataset: Data, benchmarks and analysis," in *ICRA*, 2021, pp. 1110–1116.

[13] T. Guan, Z. He, R. Song, D. Manocha, and L. Zhang, "Tns: Terrain traversability mapping and navigation system for autonomous excavators," in *RSS*, 2022.

[14] N. Carion et al., "Sam 3: Segment anything with concepts," *arXiv:2511.16719*, 2025.

[15] A. Kirillov et al., "Segment anything," in *ICCV*, 2023, pp. 4015–4026.

[16] M. Karnekar, O. Mandhane, and G. Ramkumar, "construction-traversability-dataset (Revision 717f9f0)," Hugging Face, 2026. Available: `https://huggingface.co/datasets/manojkarnekar/construction-traversability-dataset`. doi: 10.57967/hf/9971.