MOSU2: Long-range Navigation System with Multi-modal Scene Understanding

Journal Title
XX(X):1–17
©The Author(s) 2026
Reprints and permission:
sagepub.co.uk/journalsPermissions.nav
DOI: 10.1177/ToBeAssigned
www.sagepub.com/
SAGE

Jing Liang*[1], Zhenyu Hou*[2], Senthurbavan Kirubaharan[2], Md Safwan Mondal[2], Ashok Gudivada[2], Xuesu Xiao[2], Dinesh Manocha[3]

Abstract

Robots that travel long distances through public outdoor spaces must do more than avoid obstacles: they must remain on terrain suited to their mobility and decide when to proceed, yield, or detour around pedestrians, cyclists, vehicles, traffic signals, and temporary restrictions. These decisions are difficult because their correctness often depends on how a scene evolves, such as whether an approaching car is slowing or a cyclist will cross the robot's path. Vision-language models (VLMs) can interpret such cues, but they reason over images that lack metric geometry and motion. Supplying this evidence at every step is too slow for real-time navigation, and hand-crafted triggers cannot anticipate the open-ended variety of outdoor situations. We present MOSU2, a long-range navigation system with two-stage trajectory selection, in which first-stage contextual reasoning determines when additional evidence is needed and what to retrieve. Only when an interaction remains unresolved does the first stage trigger the second stage to retrieve geometric and dynamic information and align it with scene context to make navigation decisions. Building on MOSU, MOSU2 follows routes from GPS-based route planning and filters candidate trajectories through multi-modal traversability analysis before selecting a trajectory to move the robot. In real-world experiments spanning stop gestures, construction sites, crosswalks, and roads shared with vehicles, cyclists, and pedestrians, MOSU2 raises mean navigation success from 61.8% to 94.2% and condition-detection accuracy from 72.2% to 91.4% over its single-image predecessor, at about 2.2 times its mean decision latency. Project website: https://jinggm.github.io/mosu2/

Keywords

Long-range outdoor navigation; social navigation; traversability analysis.

Introduction

Long-range robot navigation has advanced significantly in recent years (Gao et al. 2019; Liang 2025). This task requires mobile robots to perceive and reason about heterogeneous environments to reach distant destinations safely, efficiently, and in a socially compliant manner (Gul et al. 2019; DeSouza and Kak 2002; Mavrogiannis et al. 2023). It plays a crucial role in applications such as logistics (Alverhed et al. 2024; Chen et al. 2021), service robotics (Zemke et al. 2020; Berezina et al. 2019), and search and rescue (Davids 2002; Kalita and Thangavelautham 2020).

Despite these advances, long-range navigation remains challenging in diverse and unstructured environments, where robots with different mobility capabilities (Liang et al. 2025; Eder et al. 2023; Wellhausen and Hutter 2021) encounter varying terrain (Weerakoon et al. 2023; Sathyamoorthy et al. 2022), irregular obstacle motion (Wijayathunga et al. 2023), and complex traffic and social interactions (Liang et al. 2026; Payandeh et al. 2025; Kong et al. 2025). For example, a robot must identify terrain that is compatible with its physical capabilities and, before entering the crosswalk, determine whether an approaching vehicle is yielding. These situations require the robot to coordinate global route guidance, local traversability assessment, and contextual reasoning while accounting for the evolving behavior of surrounding road users.

Long-range navigation requires both global guidance to a distant goal and accurate, timely local scene understanding so that the robot can move safely and efficiently, as illustrated in Fig. 1. Existing navigation systems use global SLAM-based maps to support localization and path planning (Gasparetto et al. 2015; Ozturk et al. 2022), while approaches such as ViNT combine learned visual navigation with topological graph-based planning (Shah et al. 2023). These representations provide useful global structure, but they must be built from prior mapping or traversal of the environment, maintaining them over large environments can be costly, and environmental changes can invalidate previously traversable routes. Moreover, global guidance does not necessarily capture the terrain conditions or temporary obstructions encountered during execution. The challenge is therefore to maintain progress toward a distant goal while reconciling global guidance with limited local

[1]Stanford University, CA, USA
[2]George Mason University, VA, USA
[3]University of Maryland, College Park, MD, USA
* indicates equal contribution

Corresponding author:
Jing Liang, Stanford Robotics Center, Computer Science Department, Stanford University, 350 Jane Stanford Way, RM122, Stanford, CA 94305, USA.
Email: jinglms@stanford.edu

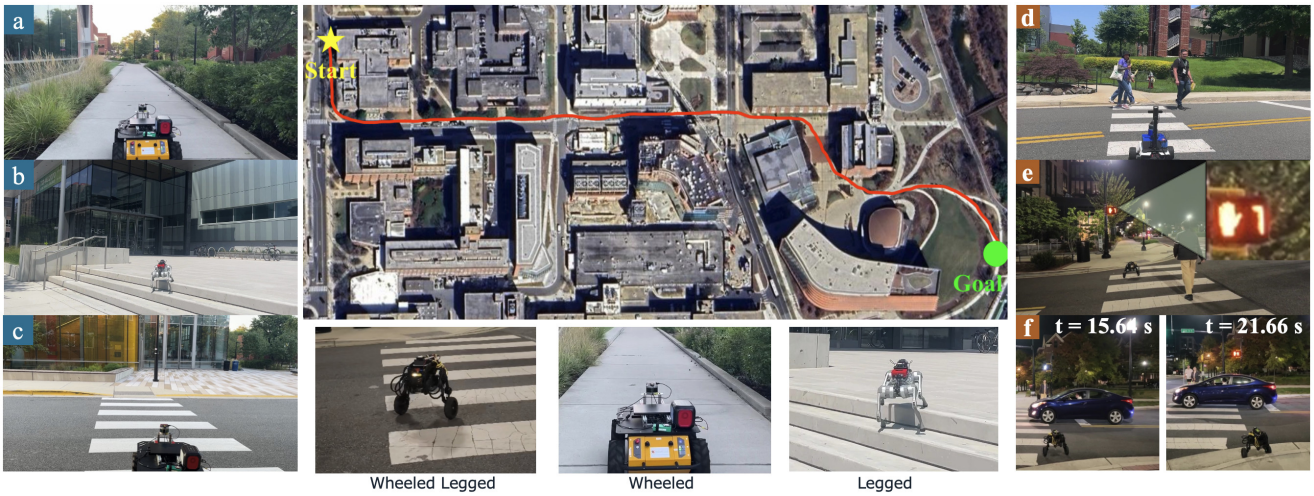


Figure 1. MOSU2 supports long-range outdoor navigation across wheeled-legged, wheeled, and legged robot platforms (bottom row). The satellite view illustrates a route from the start (yellow star) to the goal (green circle). The left-side examples highlight traversability challenges: (a) distinguishing sidewalks from off-road terrain, (b) accounting for differences in robot mobility, and (c) identifying crosswalks for street crossing. The right-side examples highlight social and traffic challenges: (d) interacting with moving agents, such as pedestrians, (e) interpreting traffic signals, including pedestrian signals with countdown displays, and (f) resolving conflicting traffic cues when vehicles violate traffic rules and obstruct the robot's path. MOSU2 combines global route guidance and robot-specific traversability assessment with two-stage reasoning that retrieves local geometric and dynamic information to interpret these conditions and select appropriate motion.

observations. Our framework uses publicly available route information and GPS localization for coarse guidance, while onboard perception determines locally traversable motion without requiring a detailed prebuilt metric map.

Another challenge is to assess traversability according to the robot's morphology and motion capabilities, as illustrated by the terrain variations in Fig. 1 (a-c). Geometric methods maintain local elevation maps to represent terrain shape and obstacles (Fankhauser et al. 2018), while semantic methods estimate traversability to distinguish areas that have similar geometry but different implications for navigation (Kim et al. 2024). Learning-based approaches generate local actions or candidate trajectories directly from onboard observations (Liang et al. 2024a; Shah et al. 2023; Sridhar et al. 2024). Nevertheless, local-map reconstruction remains limited by sensor coverage and occlusion, particularly around corners or in crowded environments, while learned semantic predictions are constrained by the training distributions (Kim et al. 2024). Our framework combines learned trajectory generation with geometric and robot-specific semantic traversability constraints to evaluate candidate motions.

Beyond physical traversability, robots operating in shared public spaces must understand social cues and traffic conditions, as illustrated in Fig. 1 (d-e). Human gestures, pedestrian signals, countdown displays, and vehicle stopping behavior convey information whose meaning depends on the surrounding situation and its evolution over time. Recent approaches use vision-language models (VLMs) to incorporate contextual knowledge into social navigation (Song et al. 2025a; Payandeh et al. 2025). However, methods interpreting individual images can miss the temporal evidence needed to analyze ongoing motion from changes in behavior, and qualitative descriptions do not necessarily provide reliable metric information for planning (Kong et al. 2025). Our approach grounds visual-context reasoning in geometric and dynamic information retrieved from a local map.

Another challenge is translating these complementary, and sometimes conflicting, observations into an executable navigation decision, as illustrated in Fig. 1 (f). For instance, if the pedestrian signal indicates that crossing is permitted but a vehicle is slowly approaching without yielding, the robot must be able to prioritize these conflicting cues to make a safe navigational decision. Existing approaches integrate contextual information through behavioral cost maps (Weerakoon et al. 2025) or VLM-based selection among paths generated using geometric and human-motion information (Fang et al. 2026). Nevertheless, ambiguity among concurrent activities and missing temporal context can still lead to inappropriate prioritization of social cues (Fang et al. 2026). The remaining challenge is to identify which observations matter for the current decision and reconcile their implications with physical feasibility and progress toward the goal. Our approach organizes contextual interpretation, geometric and dynamic analysis, and trajectory selection into two reasoning stages that pass along the information relevant to the navigation decision.

Our previous work, MOSU (Liang et al. 2026), established a long-range navigation framework that combines coarse GPS-based routing with multimodal local scene understanding. It generates candidate trajectories from LiDAR and robot-motion observations, applies semantic traversability constraints using RGB images, and uses a VLM to assess the social appropriateness of candidate trajectories for various robot types. However, although its trajectory generator incorporates temporal sensor inputs, its VLM-based social reasoning does not explicitly associate visual cues with metric geometry and motion history. Consequently, the system has limited understanding of critical interactions that emerge over time, such as whether a vehicle or a

bicycle is approaching, stopping, or yielding. In addition, its predefined prompting and trajectory-scoring pipeline lacks an explicit mechanism for selecting and retrieving additional scene information when the current semantic and geometric information is insufficient to interpret an interaction. These limitations motivate a tighter connection between contextual reasoning, measured road-user states, and navigation decisions.

To address these limitations, we present MOSU2, an extension of MOSU that combines information retrieval and two-stage reasoning for long-range navigation. MOSU2 allows VLMs to retrieve geometric and dynamic information from local maps and interpret it together with social and traffic cues from RGB observations. We also propose a two-stage reasoning architecture that coordinates this process by (1) identifying the information relevant to the current navigation decision and (2) retrieving and interpreting supporting evidence to select trajectories within MOSU's routing and traversability framework. The design leverages a key distinction between interaction detection and interpretation: a single image can flag an approaching vehicle or gesture, but deciding whether to yield requires measured trajectory and speed. MOSU2 thus resolves routine paths in a fast first stage and selectively triggers a second, deliberative stage only when metric geometric and dynamic evidence is needed. Our contributions beyond MOSU are:

- We introduce a method that connects VLM-based contextual understanding with geometric and dynamic information represented in local maps. The method enables VLMs to retrieve relevant map information and relate surrounding geometry and motion to observed social and traffic cues.
- We develop a two-stage architecture that coordinates information gathering and navigation decisions. The first stage determines whether more information is needed for the current situation and, if so, what information is needed and where; given the request from the first stage, the second stage retrieves geometric and dynamic information about the scene according to the context and combines it with traversability assessments and route guidance to select an appropriate trajectory.

Related Work

Long-range Outdoor Navigation and Traversability

Long-range outdoor navigation requires robots to reconcile guidance toward a distant destination with locally feasible motion. Several approaches address this problem without requiring a detailed global metric map. ViNT (Shah et al. 2023) learns a general-purpose visual navigation policy from heterogeneous robot datasets and combines it with topological planning for extended navigation. NoMaD (Sridhar et al. 2024) further unifies goal-directed navigation and exploration through a goal-conditioned diffusion policy. These approaches demonstrate that coarse global information and learned local navigation can support long-range autonomy without reconstructing the entire environment in metric detail.

Recent methods improve outdoor navigation through broader training data and longer planning horizons. CityWalker (Liu et al. 2025) learns urban navigation policies from large-scale walking and driving videos, using observation and trajectory histories to predict future motion. Long Range Navigator (Schmittle et al. 2025) predicts promising frontier directions from egocentric images, extending the effective horizon of local navigation systems beyond their metric maps. These methods address complementary challenges in generalization and goal-directed progress. However, their navigation interfaces do not explicitly request additional metric evidence to resolve a particular social or traffic ambiguity. This distinction motivates our first reasoning stage, which uses contextual reasoning to identify when route-following requires further investigation of the surrounding scene.

Local traversability provides another essential component of outdoor navigation. ViPlanner (Roth et al. 2024) combines depth and semantic observations to generate paths that account for terrain-dependent traversability, while MTG (Liang et al. 2024a) generates diverse candidate trajectories with objectives that encourage traversability and coverage. However, physical traversability alone does not determine whether a motion is appropriate. For example, a traversable crossing may require waiting for traffic, and an accessible pathway may be temporarily restricted. Our previous MOSU framework (Liang et al. 2026) integrates candidate generation, geometric and semantic assessment, and visual-context evaluation with GPS-based route guidance. MOSU2 retains this navigation foundation while introducing additional reasoning over social, traffic, and temporary environmental conditions.

Social and Traffic-aware Navigation

Social navigation extends collision avoidance by considering how robot motion affects nearby people and whether it conforms to contextual expectations. ORCA (Van Den Berg et al. 2011) computes reciprocal collision-avoidance actions using the relative motion of neighboring agents. Learning-based approaches, such as the attention-based crowd-navigation policy of CrowdSteer (Liang et al. 2021), additionally model human-robot and human-human interactions to anticipate crowd behavior. These methods explicitly account for motion, but their state-based formulations do not directly represent the meaning of gestures, traffic signals, or temporary restrictions.

Vision-language models provide a complementary means of interpreting such contextual information. VLM-Social-Nav (Song et al. 2025a) uses visual-language reasoning to score socially appropriate directions and speeds for a downstream navigation system. Social-LLaVA (Payandeh et al. 2025) develops navigation-oriented visual reasoning through supervised social-scene understanding, while AutoSpatial (Kong et al. 2025) improves spatial reasoning through structured grounding and hierarchical visual question answering. These approaches broaden the range of social situations that navigation systems can handle. However, understanding an interaction from an individual image remains limited when its meaning depends on dynamic information. AutoSpatial explicitly identifies temporal understanding as an open challenge, including cases in which

human intentions become apparent only across successive observations. This limitation motivates combining image sequences with measured spatial and motion information.

Several methods already connect contextual reasoning with geometric planning. BehAV (Weerakoon et al. 2025) translates language instructions into navigation and behavioral requirements, grounds the latter in a behavioral cost map, and integrates this representation with LiDAR occupancy information. Fang et al. (2026) combine human-motion prediction with geometrically feasible path sampling and use a fine-tuned VLM to select a socially appropriate path. Their reported failure cases include missing visual memory, ambiguous simultaneous activities, and limitations in the available candidate paths. These studies establish that contextual reasoning and geometric or dynamic information can be integrated through several interfaces.

MOSU2 investigates a specific interface within this broader direction: the reasoning process requests relevant geometric and dynamic information from a local map and interprets the returned evidence together with consecutive RGB observations. This supports questions whose answers depend on both context and measured motion, such as whether an approaching vehicle is slowing near the intended crossing or whether a pedestrian's gesture is consistent with an available passage. Motion measurements provide supporting evidence rather than a definitive interpretation of intent. The resulting assessment guides selection among trajectories that have already passed geometric and semantic traversability evaluation.

Agentic Robotics and Adaptive Reasoning

Robotic planning agents increasingly use external representations and tools to connect language reasoning with observations of the physical environment. NLMap (Chen et al. 2023) constructs an open-vocabulary scene representation that allows a language-model planner to query object availability and location before generating a plan. SayPlan (Rana et al. 2023) uses semantic search over hierarchical 3D scene graphs, together with conventional path planning and feedback-based replanning, to support large-scale mobile-manipulation tasks. More recently, the SuperNav preprint (Zhang et al. 2026a) organizes navigation around a pretrained multimodal model with tool access, task-progress tracking, and context management. These approaches motivate separating information acquisition from planning while maintaining an explicit connection between the two.

The computational cost of foundation-model reasoning has also motivated architectures that separate deliberation from rapid execution. DualVLN (Wei et al. 2026) combines a VLM that predicts intermediate goals with a lightweight diffusion policy for local motion. Hydra-Nav (Wang et al. 2026) adaptively switches between deliberative reasoning and reactive execution for object-goal navigation. Particularly relevant to outdoor deployment, the Walk With Me preprint (Zhang et al. 2026b) combines public-map routing, high-level VLM scene assessment, and low-level VLA trajectory generation. Its high-level model assesses whether the robot should proceed or stop and wait in situations such as crossings and crowded scenes. These methods demonstrate the relevance of hierarchical and adaptive reasoning, so the use of two stages alone does not distinguish MOSU2.

Our two-stage design instead focuses on deciding when additional evidence is required and what information should be retrieved. The first stage screens the current scene while retaining a subgoal-directed trajectory as the default choice. When further investigation is needed, the second stage examines recent images and retrieves relevant map information before selecting a trajectory and velocity mode. This design reserves the additional image-sequence analysis and map-query process for situations identified as requiring it, while allowing contextual observations to determine the investigation rather than relying exclusively on a predefined list of triggering conditions.

Our Approach

Our approach consists of two parts: the navigation framework inherited from MOSU (Liang et al. 2026) and the extensions introduced in MOSU2. MOSU decomposes long-range navigation into routing and local trajectory generation and evaluation. The routing module uses OpenRouteService and GPS data to provide global guidance for long-range navigation, while trajectory generation and evaluation combine multi-modal sensor observations to generate candidate trajectories, assess traversability costs, and evaluate compliance with social and traffic rules from RGB perception. However, because MOSU does not use geometric and dynamic information about surrounding road users, it cannot determine from a single image whether a nearby pedestrian, cyclist, or vehicle is approaching, yielding, or moving away, so it may stop unnecessarily or proceed prematurely. Building on this framework, MOSU2 introduces a VLM-retrievable local map that provides geometric and dynamic information relevant to the observed scene context. Real-world environments are complicated and contain potentially hazardous conditions that cannot be easily detected by traversability analysis, such as dangerous construction sites or road closures marked only by subtle signs. To handle these challenges, we design a two-stage reasoning architecture. In the first stage, a VLM handles navigation decisions using the available observations together with MOSU's route guidance and trajectory assessments, and determines whether further investigation is needed. When additional information is required, the second stage retrieves relevant map information, coordinates visual-context interpretation with geometric and dynamic analysis, and combines the resulting evidence to select an appropriate trajectory.

Global Direction From Routing

We retain the routing strategy of MOSU (Liang et al. 2026) to provide coarse global guidance for long-range navigation. Given the robot's current GPS position and a target location, the routing module obtains a high-level route from a public routing service, such as Google Maps or OpenRouteService, and represents it as a sequence of intermediate GPS subgoals. These subgoals indicate the intended progression toward the destination, while local trajectory generation and contextual reasoning determine how the robot should move through the observed environment.

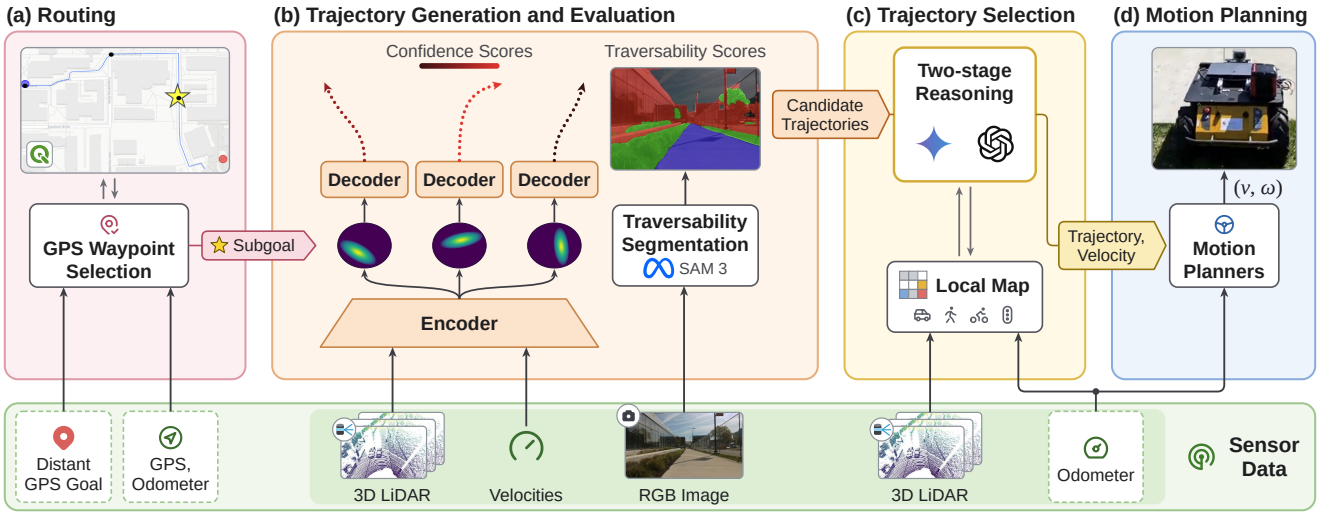


Figure 2. Overview of the MOSU2 architecture. Building on MOSU's routing and traversability framework, GPS-based routing provides subgoals, while LiDAR and robot-velocity observations support candidate trajectory generation. RGB-based semantic segmentation provides complementary traversability assessments. MOSU2 introduces a VLM-retrievable local map and two-stage trajectory selection: the first stage screens the current scene and determines if more analysis is required, and the second stage, when invoked, combines recent images with retrieved geometric and dynamic information to select a trajectory and velocity mode. The motion planner converts the selected motion into linear and angular velocity commands (v, ω).

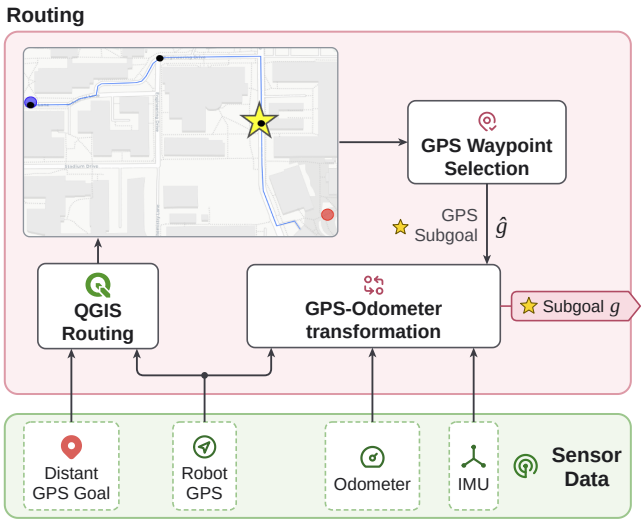


Figure 3. Routing and subgoal transformation. OpenRouteService routing uses the robot's GPS location and the destination to obtain a global route, from which an intermediate GPS waypoint $\hat{g}$ is selected. The waypoint is converted to the odometry frame with metric coordinates by aligning the GPS and robot motion, yielding the subgoal g. This subgoal provides directional guidance for trajectory selection.

Because GPS localization is subject to meter-scale errors (Van Diggelen and Enge 2015), subgoal coordinates may not coincide with locally traversable terrain. We therefore use them as directional guidance rather than precise positions to be reached. Following the MOSU implementation, as shown in the Routing block of Fig. 3, consecutive subgoals are spaced approximately $35\,\mathrm{m}$ apart, a distance chosen in relation to the perceptual range of the onboard 3D LiDAR and RGB sensors. This spacing limits reliance on closely spaced GPS coordinates, while local geometric and semantic context assessments guide the robot along traversable terrain. We then transform the GPS subgoal into the robot's odometry frame using the Universal Transverse Mercator projection, and an extended Kalman filter updates this transformation. We have $g = \mathbf{T}\hat{g}$, where $\hat{g}$ contains the GPS latitude and longitude, g is the subgoal in the odometry frame, and $\mathbf{T}$ is the calculated transformation matrix.

During navigation, the active subgoal is transformed into the robot's coordinate frame using the estimated robot pose. The system monitors the approximate distance to this subgoal and advances to the next one when the distance falls below a predefined threshold of $15\,\mathrm{m}$. Route guidance specifies the desired direction of progress, while local traversability, geometric and dynamic information, and social and traffic context determine the appropriate motion toward it.

Trajectory Generation and Evaluation

Given the noisy subgoal from the GPS due to inherent sensor noise and the complex and unstructured outdoor environment, it is very challenging to generate an optimal trajectory to solve the navigation problem at once. Relying solely on geometric or color information is insufficient to fully understand the environment's traversability (i.e., which areas are safe for the robot to traverse), as shown in Fig. 6. Therefore, our approach integrates both geometric and color information for trajectory generation and evaluation. Classical approaches perform traversability analysis of the environment to provide the constraints for trajectory optimization, but these approaches often require complete perception and are computationally slow and heavy. Learning-based approaches use end-to-end models to directly convert high-dimensional perceptual data into low-dimensional trajectories. They are faster but require large amounts of data to cover complex outdoor conditions. Our approach separates this problem into two steps: we first generate trajectories that cover the traversable area in front of the robot, and then use various models to

provide constraints, evaluate the trajectories, and select the best trajectory to follow. In this way, we do not need to build an integrated map for trajectory optimization, and we can also leverage the contextual understanding of large VLMs to handle traffic and social conditions. In this work, we inherit two components from MOSU: (1) CVAE-based multiple trajectory generation, which models the distribution of feasible trajectories based on LiDAR and velocity inputs, and (2) semantic segmentation for traversability constraints, which leverages RGB images to identify navigable surfaces.

CVAE-based Multiple Trajectory Generation: This trajectory generation model understands the geometric information of the environment for traversability analysis through 3D LiDAR point clouds. However, directly modeling the environment using 3D point clouds is memory-intensive, and segmenting large point clouds is computationally expensive (Guo et al. 2020; Sohail et al. 2024). To address this issue, we adopt MTG (Liang et al. 2024a), a lightweight learning-based approach that generates multiple candidate trajectories to cover traversable regions in front of the robot. MTG employs a Conditional Variational Autoencoder (CVAE) (Sohn et al. 2015) to model the distribution of feasible trajectories. Given sensor input $\mathbf{x} = \{\mathbf{l}, \mathbf{v}\}$, where $\mathbf{l} \in \mathcal{L}$ represents a sequence of LiDAR observations and $\mathbf{v} \in \mathcal{V}$ denotes the robot's historical velocities, the model generates a set of N candidate trajectories $\mathcal{T} = \{\tau_1, ..., \tau_N\}$. For each trajectory $\tau_n \in \mathcal{T}$, we have the following formulation:

$$p_\theta(\tau_n|\mathbf{x}) = p_\theta(\tau_n|\mathbf{z}_n, \mathbf{c}), \quad \mathbf{z}_n \sim \mathcal{N}(\mu_n, \nu_n), \tag{1}$$

$$\mathbf{c} = f_\theta(\mathbf{x}), \quad \mu, \nu = e_\theta(\mathbf{x}), \tag{2}$$

$$\mathcal{N}(\mu_n, \nu_n) = h_\theta(\mathcal{N}(\mu, \nu), \mathbf{c}) \tag{3}$$

We slightly abuse notation by using θ to represent the parameters of all neural networks. $\mathbf{c}$ is the conditional vector obtained from an encoder network $f_\theta(\cdot)$. $e_\theta(\cdot)$ is the latent encoder network that takes the sensor inputs $\mathbf{x}$ and predicts the parameters, mean μ and variance ν, of a Gaussian distribution $\mathcal{N}(\mu, \nu)$. In our approach, the LiDAR data is processed by PointCNN (Li et al. 2018), and the velocities are processed by a sequence of linear layers, as shown in (Liang et al. 2024a). Then we linearly transform the distribution to N Gaussian distributions $\mathcal{N}(\mu_n, \nu_n)$ by the learnable linear transformation (neural network) $h_\theta(\cdot)$. We then sample the latent vector $\mathbf{z}_n$ from the distribution $\mathcal{N}(\mu_n, \nu_n)$ and use it as the input to the trajectory decoder $p_\theta(\cdot)$ to generate trajectory τ_n. The variance ν_n of each latent distribution represents its uncertainty and is used to compute the confidence score c_τ of trajectory τ_n. This formulation takes the LiDAR geometric information as input and generates trajectories to cover geometrically traversable areas in front of the robot.

Open-vocabulary Semantic Segmentation for Traversability Analysis: However, the trajectory generation model struggles to accurately detect the boundaries between adjacent traversable and non-traversable areas when they have similar geometric structures (e.g., off-road terrain vs. sidewalks), as shown in Fig. 6(a). To address this limitation, we incorporate semantic information by using SAM3 (Carion et al. 2026), an RGB-based semantic segmentation model with open-vocabulary queries. The model segments the image into multiple regions with different queried semantic categories by predicting the class label of each pixel. We define five traversability categories (Liang et al. 2025): road, sidewalk, vegetation, building, and stairs. Different types of robots have different traversable areas: wheeled robots are restricted to the sidewalk and road categories, whereas legged robots can also traverse stairs.

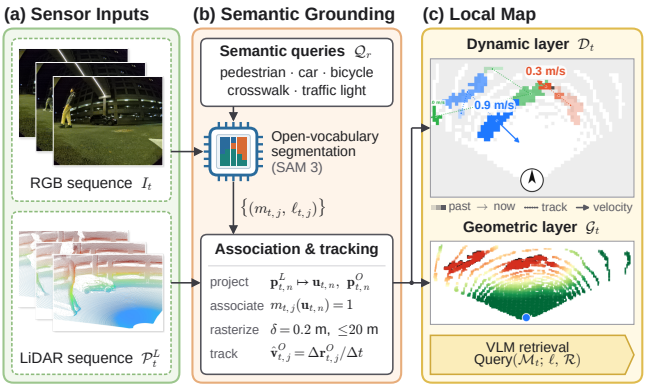


Figure 4. Construction of the VLM-retrievable local map. A VLM generates robot-specific semantic queries for the segmentation of consecutive RGB images. The corresponding LiDAR points are then used to construct a 2D local map that provides the geometric information of the semantic objects and to calculate the motion information of the objects. The constructed geometric and dynamic layers allow VLMs to retrieve relevant information by semantic category and spatial location for social context understanding.

Given the segmented image, we overlay the set of candidate trajectories $\mathcal{T}$, generated by the CVAE-based trajectory generator, onto the image and evaluate their traversability scores. First, we use the Bresenham algorithm (Bresenham 1998) to convert trajectory waypoints into connected pixels in the image. Then, the semantic traversability score s_τ is computed for each trajectory as the ratio of pixels falling in traversable areas to the total number of pixels along the trajectory.

In this work, we decouple the traversability analysis from social and traffic decision-making. This decoupling is reasonable because physical traversability determines which motions are feasible, whereas social and traffic reasoning determines which feasible motion is appropriate. Therefore, unlike the original MOSU, we use these two complementary components only to filter the candidate trajectories before contextual reasoning. For the generated N trajectories, let c_τ^i denote the confidence score from geometric information and s_τ^i denote the semantic traversability score of trajectory $i \in [1, N]$. We calculate the traversability of each trajectory as

$$\kappa_\tau^i = \beta_1 c_\tau^i + \beta_2 s_\tau^i; \qquad \mathcal{T}_s = \{\tau_i \in \mathcal{T} \mid \kappa_\tau^i > \kappa_{\min}\}, \tag{4}$$

where $\beta_{1,2} > 0$ are the weights of the components, $\kappa_{\min}$ is a predefined threshold, and $\mathcal{T}_s$ represents the set of trajectories retained by traversability analysis.

VLM-retrievable Local Map

We construct a local map that associates open-vocabulary semantic information with the geometry and motion of the surrounding environment, as illustrated in Fig. 4. The map represents traffic and social components in the environment, including crosswalks and traffic lights, together with the positions and velocities of nearby cars, pedestrians, and bicycles. These attributes provide evidence for social and traffic decisions beyond the physical traversability assessments described above. By exposing the traffic and social information through queryable function interfaces, the map allows VLMs to retrieve relevant environmental measurements and interpret them together with visual context.

At the beginning of each experiment, a VLM generates a set of semantic queries, $\mathcal{Q}_r$, based on the robot's type and mobility capabilities. The query set is not fixed during a run; during navigation, the VLM can add query terms according to the first-stage request. Given an RGB image I_t, the open-vocabulary segmentation model produces

$$\mathcal{S}_t = F_{\text{seg}}(I_t; \mathcal{Q}_r) = \{(m_{t,j}, \ell_{t,j})\}_{j=1}^{J_t}, \tag{5}$$

where $m_{t,j}$ is the binary mask of segment j, $\ell_{t,j}$ is its semantic label, and J_t is the number of detected segments. We use SAM3 (Carion et al. 2026), consistent with the preceding traversability analysis. The queries include traffic-related concepts such as "traffic light," "car," "pedestrian," and "bicycle," together with additional concepts relevant to the robot platform.

To recover metric geometry, we associate each segmented image with the synchronized LiDAR point cloud $\mathcal{P}_t^L = \{\mathbf{p}_{t,n}^L\}$, expressed in the LiDAR frame L. We project the points onto the image plane to get the distances of the segments. Let $\mathbf{K}$ denote the camera intrinsic matrix, and let $(\mathbf{R}_{CL}, \mathbf{b}_{CL})$ denote the calibrated transformation from the LiDAR frame to the camera frame. A LiDAR point is projected into the rectified image as

$$\mathbf{u}_{t,n} = \pi\big(\mathbf{K}\left(\mathbf{R}_{CL}\mathbf{p}_{t,n}^L + \mathbf{b}_{CL}\right)\big), \tag{6}$$

where $\pi([x, y, z]^\top) = [x/z, y/z]^\top$. Points with valid positive depth and projections within the image are associated with the masks containing their projected pixels.

The associated points are then transformed into the odometry frame O:

$$\mathbf{p}_{t,n}^O = \mathbf{R}_{OL,t}\mathbf{p}_{t,n}^L + \mathbf{b}_{OL,t}, \tag{7}$$

where $(\mathbf{R}_{OL,t}, \mathbf{b}_{OL,t})$ is obtained from the robot's odometry pose and the calibrated LiDAR mounting transformation. Expressing consecutive observations in the same odometry frame accounts for the robot's motion when comparing segment positions. For each segment, its associated point set and representative position are

$$\begin{aligned} \mathcal{P}_{t,j}^O &= \left\{\mathbf{p}_{t,n}^O \mid m_{t,j}(\mathbf{u}_{t,n}) = 1\right\}, \\ \mathbf{r}_{t,j}^O &= \frac{1}{|\mathcal{P}_{t,j}^O|}\sum_{\mathbf{p}\in\mathcal{P}_{t,j}^O}\mathbf{p}. \end{aligned} \tag{8}$$

Here, $\mathbf{r}_{t,j}^O$ is the centroid of the observed segment points. The full point set represents its observed spatial extent. Segments without valid LiDAR support do not receive a metric position estimate.

We maintain observations within a $20\,\text{m}$ radius around the robot, and discretize them into a local grid with resolution $\delta = 0.2\,\text{m}$. Let $\mathbf{o}_t^O \in \mathbb{R}^2$ denote the grid origin in the odometry frame. The cells associated with segment j are

$$\Omega_{t,j} = \left\{ \left\lfloor \frac{\mathbf{p}_{xy}^O - \mathbf{o}_t^O}{\delta} \right\rfloor \,\middle|\, \mathbf{p}^O \in \mathcal{P}_{t,j}^O \right\}, \tag{9}$$

where $\mathbf{p}_{xy}^O$ contains the horizontal (xy) coordinates. This projection preserves the observed layout of an extended region, instead of representing it only by its centroid.

Motion information is estimated from the displacement of corresponding segments across consecutive observations. Let $a_t(j)$ identify the segment in the previous frame associated with segment j at time t. We associate each segment with the nearest segment of the same semantic class. Given an observation interval Δt, the object velocity estimate is

$$\hat{\mathbf{v}}_{t,j}^O = \frac{(\mathbf{r}_{t,j}^O)_{xy} - (\mathbf{r}_{t-1,a_t(j)}^O)_{xy}}{\Delta t}. \tag{10}$$

A velocity estimate is provided only when a valid temporal correspondence and sufficient geometric observations are available; otherwise, velocity is marked as unknown.

The semantic and motion attributes are organized into geometric and dynamic layers,

$$\begin{aligned} \mathcal{G}_t(\mathbf{a}) &= \left\{(j, \ell_{t,j}, \mathbf{r}_{t,j}^O) \mid \mathbf{a} \in \Omega_{t,j}\right\}, \\ \mathcal{D}_t(\mathbf{a}) &= \left\{(j, \hat{\mathbf{v}}_{t,j}^O) \mid \mathbf{a} \in \Omega_{t,j}\right\}, \end{aligned} \tag{11}$$

where $\mathbf{a}$ denotes a grid cell. These layers associate semantic regions with their measured geometry and estimated motion.

Finally, we expose the map through function interfaces that support retrieval by semantic type, spatial location, or their combination. For a requested semantic label ℓ and a spatial region $\mathcal{R}$ expressed as a set of grid cells, a combined query is written as

$$\text{Query}(\mathcal{M}_t; \ell, \mathcal{R}) = \left\{(\mathbf{r}_{t,j}^O, \hat{\mathbf{v}}_{t,j}^O) \mid \ell_{t,j} = \ell,\ \Omega_{t,j} \cap \mathcal{R} \neq \emptyset\right\}, \tag{12}$$

where $\mathcal{M}_t$ denotes the local map with its position, semantic, and dynamic information. We then convert the query results to the robot's frame. The returned measurements provide evidence for the subsequent two-stage reasoning process, which combines them with visual context and route guidance to select a trajectory from $\mathcal{T}_s$.

Two-stage Trajectory Selection

Given the traversable candidate set $\mathcal{T}_s$, we design a two-stage trajectory selection process to balance the computational cost and latency of VLM-based reasoning with the need to recognize context-dependent navigation requirements that call for additional geometric and dynamic features. VLMs can interpret social and traffic cues using contextual knowledge, but detailed analysis of image sequences and geometric/dynamic information retrieval can introduce substantial computation and latency. Meanwhile, manually specified detection rules may not cover the diverse interactions and unfamiliar conditions encountered during

outdoor navigation. We therefore use a VLM in the first stage with a single image input to examine the current observation and identify whether additional geometric and dynamic information is needed, allowing the observed context to guide further investigation. When the available information is sufficient, the system selects a trajectory from $\mathcal{T}_s$ according to the routed subgoal. Otherwise, the second stage analyzes consecutive RGB observations and retrieves relevant geometric and dynamic information from the local map to support trajectory selection and provide a velocity mode. This architecture reserves detailed reasoning for situations identified as requiring further investigation, without requiring every triggering condition to be explicitly specified in advance.

In the first stage, candidate trajectories are projected onto the image plane and labeled with numbers from right to left (Liang et al. 2026). Then the projected trajectories are evaluated according to their progress toward the subgoal. Let $\mathbf{g}_t^B$ denote the routed subgoal expressed in the robot's current coordinate frame, and let $\mathbf{x}_{i,\text{end}}^B$ denote the endpoint of candidate trajectory τ_i in the same frame. For a nonempty candidate set, the default trajectory is selected as

$$\tau_t^{(1)} = \arg\min_{\tau_i \in \mathcal{T}_s} \left\| \mathbf{x}_{i,\text{end}}^B - \mathbf{g}_t^B \right\|_2^2 . \tag{13}$$

This selection uses the geometric and semantic traversability assessments already incorporated into $\mathcal{T}_s$. Besides the selected $\tau_t^{(1)}$, the model also outputs a suggested velocity mode, [normal speed, slow speed, stop] (Liang et al. 2026).

Before executing the selected trajectory, a VLM examines the current RGB observation together with the subgoal and candidate trajectories to determine whether additional scene information is needed:

$$(b_t, \mathcal{U}_t) = F_{\text{screen}} \left(I_t, \mathbf{g}_t^B, \mathcal{T}_s \right), \tag{14}$$

where $b_t \in \{0, 1\}$ indicates whether to invoke the second stage, and $\mathcal{U}_t$ describes the information to be investigated. For example, the first stage may identify a pedestrian interaction, a traffic signal relevant to the route ahead, or a temporary restriction that is not captured by physical traversability alone. The resulting request specifies the relevant scene elements and the unresolved question, such as whether the pedestrian in front is blocking the road or whether the bicycles or cars in traffic on the left are yielding. When no further investigation is requested, the system selects $\tau_t^{(1)}$.

When $b_t = 1$, the second stage receives the information requests and a sequence of the H most recent RGB observations, $\mathcal{I}_t^H$. The image sequence provides temporal context for interpreting actions and changes in the scene. Based on $\mathcal{U}_t$ and these observations, the second stage selects semantic and spatial queries for the local map. Let $\mathcal{Q}_t^{\text{map}}$ denote the requested combinations of semantic labels and spatial regions. The retrieved evidence is

$$\mathcal{Z}_t = \bigcup_{(\ell, \mathcal{R}) \in \mathcal{Q}_t^{\text{map}}} \text{Query} \left(\mathcal{M}_t; \ell, \mathcal{R} \right). \tag{15}$$

For social interactions, the second stage interprets the consecutive images together with the retrieved positions and velocities of relevant pedestrians, cyclists, or vehicles. The images provide evidence about gestures, activities, and interaction context, while the map provides metric information about surrounding geometry and motion. Their combination supports distinctions that affect trajectory and velocity selection, such as whether a nearby road user is continuing toward the robot's path, moving away, or slowing near a crossing. These observations inform the interpretation of an interaction without treating motion alone as definitive evidence of intent. For traffic conditions, the second stage additionally interprets the state of the relevant traffic signal and its relationship to the intended crossing. Signal indications and any visible countdown information are considered together with the crossing geometry and the motion of nearby road users. The map supplies the spatial relationship of surrounding traffic, while the RGB observations provide the visual information needed to interpret the signal state. For irregular conditions, the second stage examines scene details that may change whether an otherwise traversable trajectory is appropriate. Examples include temporary barriers, construction activities, warning signs, and access restrictions. It uses the image sequence to interpret these cues and the retrieved geometry to locate the affected region relative to the candidate trajectories. A scene may involve several of these conditions simultaneously, in which case their evidence is considered jointly.

The second stage combines the retrieved evidence, visual context, and subgoal information to select the final motion:

$$\tau_t, \gamma_t = F_{\text{select}} \left(\mathcal{T}_s, \mathbf{g}_t^B, \mathcal{I}_t^H, \mathcal{Z}_t, \mathcal{U}_t \right), \tag{16}$$

where γ_t indicates the velocity mode: stop, slow, or normal speed. The selection considers progress toward the subgoal together with the social, traffic, and contextual implications of each candidate. The selected trajectory and velocity mode are passed to the motion controller, as shown in Fig. 2, and the selection process is repeated as new observations become available.

Experiments

We evaluate the navigation capabilities inherited from MOSU and the improvements introduced in MOSU2 through two groups of experiments. The first evaluates MOSU's traversability assessment and the ability of a single VLM to interpret social and traffic cues from an RGB image when ranking candidate trajectories. We retain selected results from the conference paper (Liang et al. 2026) to establish the performance of this underlying navigation framework. The second compares MOSU2 with MOSU in real-world social, traffic, and irregular scenarios to examine how MOSU2 affects navigation decisions and the recognition of situations that require a specific response. Together, these experiments assess navigation performance, contextual understanding, and time cost of reasoning.

Experimental Setup

We conduct experiments on three robot platforms, as shown in Fig. 1: Husky, Unitree Go2, and Diablo robots, representing wheeled, legged, and wheeled-legged robots, respectively. Each platform is equipped with calibrated RGB

cameras and 3D LiDAR for perceiving the surrounding environment, as well as GPS for global guidance.

Experiments are performed on a computer with an Intel i9 CPU (31 GB RAM) and an Nvidia RTX 3060 GPU (6.4 GB VRAM). We use ChatGPT-realtime (Wu et al. 2023) or Gemini (Team 2024) for visual-context interpretation and trajectory evaluation, with inference performed in the cloud through API access. The trajectory generator and motion planner operate at 20 Hz and 50 Hz, respectively. For MOSU2, the local map is updated at 2 Hz within a 20 m radius around the robot.

MOSU2 uses the conditional two-stage procedure described above. In the first stage, as in MOSU, the input includes the current RGB observation, the candidate trajectories, and the subgoal in the robot frame. If further investigation is required ($b_t = 1$), the second stage receives from the first stage the names of the semantic items and the type of information needed. According to the request $\mathcal{U}_t$, it retrieves $H = 5$ consecutive RGB observations sampled at 0.2 s intervals and spanning 1.0 s, together with the requested geometric and dynamic information about these items. The second stage then outputs a trajectory number and a velocity mode. If no further investigation is required ($b_t = 0$), the first stage directly outputs the trajectory and velocity mode without invoking the second stage.

Evaluation of MOSU

The experiment is designed to evaluate the efficacy of the system in two aspects: (1) the performance of each component, including point-cloud-based trajectory generation, semantic traversability analysis, and VLM's social understanding capability; (2) Overall MOSU system performance in real-world long-range navigation compared with other approaches.

We first summarize the trajectory selection mechanism used in MOSU (Liang et al. 2026), which provides the baseline for evaluating the extensions introduced in MOSU2. MOSU combines geometric trajectory generation, semantic traversability assessment, and contextual evaluation by a single VLM. At each decision step, candidate trajectories are projected onto the current RGB image and assigned numerical labels according to their endpoint positions in the image. The VLM receives this annotated image together with the relative position of the current subgoal and navigation instructions concerning pedestrian interactions, traffic rules, and robot-specific traversability. It returns a ranked list of trajectories and a velocity mode based on the observed scene.

The VLM ranking is converted into a score r_τ^i, with higher-ranked trajectories receiving larger scores. MOSU then selects a trajectory by combining this score with geometric confidence (c_τ^i), semantic traversability (s_τ^i), and progress toward the subgoal (g_τ^i):

$$\tau^* = \arg\max_{\tau_i \in \mathcal{T}} \left(\alpha_c c_\tau^i + \alpha_s s_\tau^i + \alpha_r r_\tau^i + \alpha_g g_\tau^i \right), \qquad (17)$$

where $\mathcal{T}$ denotes the candidate trajectory set, c_τ^i is the geometric confidence score, s_τ^i is the semantic traversability score, and g_τ^i favors trajectories approaching the current subgoal. The coefficients α_c, α_s, α_r, and α_g weight the respective components. The selected trajectory and velocity mode are passed to the motion controller. This mechanism uses a single RGB image for VLM-based contextual evaluation, without the map retrieval or conditional second-stage reasoning introduced in MOSU2.

We first qualitatively evaluate each component, including traversability analysis using geometric trajectory generation, semantic information, and VLM-based social and traversability understanding of the environment, and then we quantitatively evaluate whether this combination produces traversable trajectories that approach the destination. The quantitative evaluation uses the GND dataset (Liang et al. 2025), which contains observations from ten campuses spanning urban and rural environments. The evaluated scenarios include narrow passages, off-road terrain, crosswalks, and pedestrian interactions, as illustrated in Fig. 5. On this dataset we compare MOSU against MTG (Liang et al. 2024a), DTG (Liang et al. 2024b), and VL-TGS (Song et al. 2025b), NoMaD (Sridhar et al. 2024), ViNT (Shah et al. 2023), and PIVOT (Nasiriany et al. 2024). The results retained here use the original MOSU configuration, including Gemini as the VLM backend.

The metrics in Table 1 are calculated as follows:

- **Traversability:** We overlay the generated trajectory onto the traversability map from the GND dataset (Liang et al. 2025). The traversability score is then calculated as the percentage of fully traversable waypoints over the entire trajectory length.
- **Distance to Target:** $1 - \frac{d_\tau - d_o}{|\tau|}$, where d_τ and d_o represent the distances between the subgoal and the current trajectory and between the subgoal and the optimal ground-truth trajectory, respectively. $|\tau|$ denotes the length of the generated trajectory.

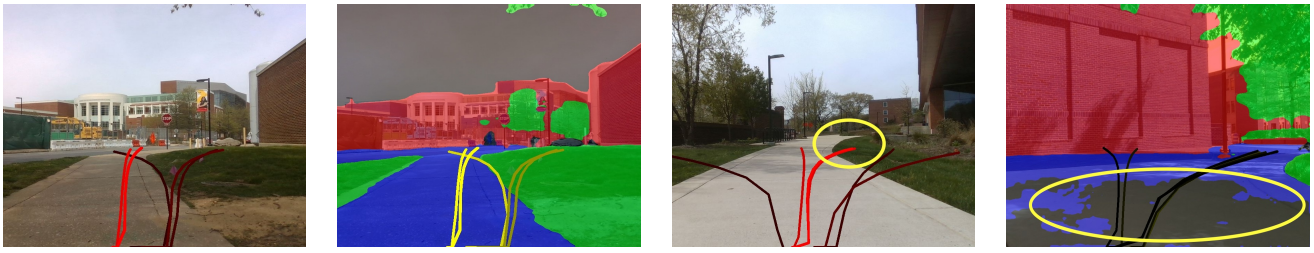
Quantitative results. Tab. 1 reports the retained comparison. MOSU achieves a traversability score of 77%, exceeding the strongest competing result, PIVOT's 70%, by seven percentage points, corresponding to a relative improvement of 10%. MOSU also achieves the highest distance-to-target score of 73%, compared with 70% for VL-TGS and 69% for PIVOT. These results indicate that the combined evaluation improves trajectory traversability while maintaining progress toward the destination under the reported benchmark. The reported inference time for MOSU is 2.30 s, matching PIVOT and closely matching VL-TGS at 2.31 s with Gemini. The remaining methods report inference times between 0.01 s and 0.69 s. The results show that the VLM is the bottleneck in inference time. Therefore, in MOSU2 we leverage the ChatGPT-realtime with shorter inference time for each analysis.

Qualitative results. As shown in Fig. 6 (a), geometry-based trajectory generation (Liang et al. 2024a) performs well in scenarios where geometric structures are easily detected. However, when the elevations of off-road areas and sidewalks are similar, it often fails, as shown in (c). In Fig. 6 (a) and (b), lighter colors indicate higher scores in both geometric confidence and color-based traversability. As shown in (d), semantic segmentation also encounters out-of-distribution scenarios, where large ground areas cannot be properly segmented. Therefore, geometric and semantic information must be combined for reliable traversability analysis, and MOSU2 inherits this design to filter candidate



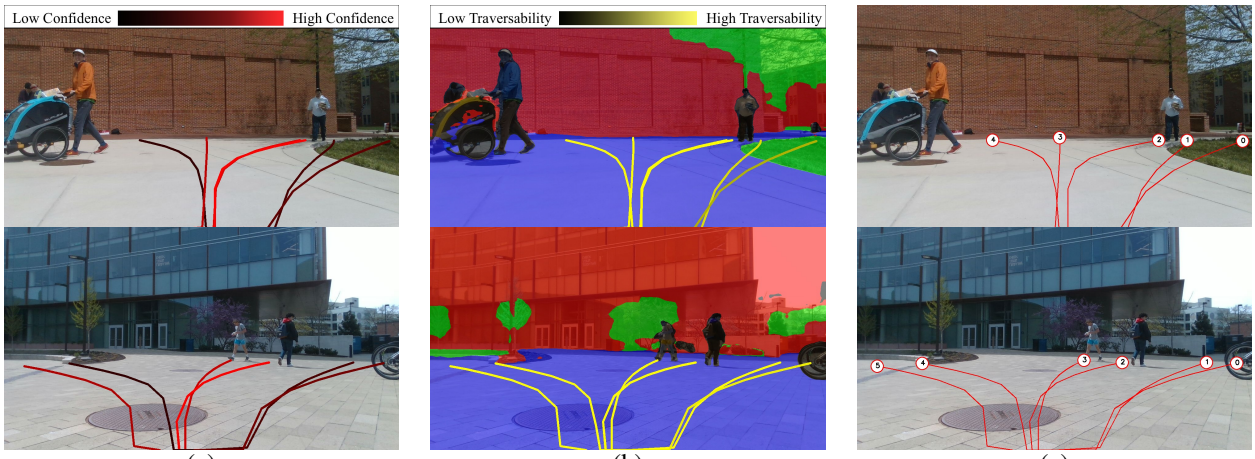
(a) Narrow Space (b) Off-road Terrain (c) Crosswalks (d) Social Scenarios

Figure 5. Complex Outdoor Scenarios: We evaluate the approach in large-scale, complex outdoor environments with various challenging scenarios, such as (a) narrow spaces, (b) areas with dense off-road vegetation, (c) crosswalks, and (d) social situations involving pedestrians.



(a) (b) (c) (d)

Figure 6. Geometric and Semantic Analysis: Both geometric trajectory generation and semantic segmentation perform well in most cases, but there are also failure cases, as marked in yellow circles in (c) and (d). When the elevations of the lawn and sidewalk are similar, the geometric model struggles to perform accurately. Additionally, segmentation fails in out-of-distribution scenarios.



(a) (b) (c)

Figure 7. Qualitative Evaluation: Each column illustrates one of the three components in our trajectory generation pipeline, (a) CVAE-based multiple trajectory generation, (b) semantic segmentation for traversability analysis, and (c) VLM-based trajectory ranking in the outdoor scenarios with various social conditions. In (a) and (b), lighter colors indicate higher scores, confidence scores in (a) and traversability scores in (b). In (c), the VLM ranks trajectories based on social cues. The top example shows a ranking of [3, 0, 1, 2, 4]. The bottom example shows a ranking of [4, 5, 0, 1, 2, 3].

Method	Modality	Traversability (%) ↑	Distance to Target (%) ↑	Inference Time (s) ↓
PIVOT (Nasiriany et al. 2024)	RGB + Language	70	69	2.30
ViNT (Shah et al. 2023)	RGB	57	62	0.69
NoMaD (Sridhar et al. 2024)	RGB	59	61	0.24
MTG (Liang et al. 2024a)	Points	61	64	0.01
DTG (Liang et al. 2024b)	Points	67	66	0.13
VL-TGS (Song et al. 2025b)	Points, RGB, Language	65	70	2.31
MOSU (ours) (Liang et al. 2026)	Points, RGB, Language	**77**	**73**	2.30

Table 1. Quantitative Evaluation: Our approach achieves the best Traversability and comparable Distance to Target.

trajectories. As shown in Table 1, our approach achieves the best traversability among all methods, and the generated trajectories lead to the target closely.

As shown in Fig. 7, beyond traversability analysis, VLMs process labeled trajectories and analyze social cues from the images. In the first row, the VLM ranks the trajectories as [3, 0, 1, 2, 4], given the target at the right front. It suggests following trajectory 3 at normal speed, while other trajectories should be taken at a slower speed when encountering humans. The ranking for the second row is [4, 5, 0, 1, 2, 3] with normal speed. In the experiment, we observed that VLMs take a significant amount of time to process, as shown in Table 1, but they demonstrate strong capability in understanding social cues.

These examples provide qualitative evidence of contextual trajectory ranking. They do not quantify social-cue recognition accuracy or establish the reliability of decisions that depend on accurately measured object motion. The following experiments therefore examine whether access to local geometric and dynamic information improves navigation under social, traffic, and irregular conditions, and at what time cost.

Evaluation of MOSU2

First, we qualitatively show the construction of the geometric and dynamic maps in Fig. 9. While the robot is moving, it maintains the geometric map in the form of an occupancy map. In the second stage of VLM reasoning, the requested semantic items are segmented in the RGB image and projected onto the geometric map to obtain the position and shape of each queried item. Given the input point clouds and RGB images, the middle column of Fig. 9 shows the geometric map, in which the orange items are those queried by the second-stage VLM, and the light green and light gray areas are free space and other occupied cells, respectively. The right column shows the dynamic map, in which we track the objects and store the velocities of the occupied cells.

We then compare MOSU2 with MOSU to evaluate navigation decisions under social, traffic, and irregular environmental conditions. As shown in Fig. 8, the evaluation covers five scenario groups: human gestures; construction/hazard sites; signalized crosswalks with cars, pedestrians and bicycles; roads shared with vehicles, cyclists, or pedestrians; and sidewalk baselines without any traffic or social conditions. These scenarios examine complementary capabilities, including interpreting human gestures, following temporary restrictions, and coordinating robot motion with traffic signals and surrounding road users. MOSU performs contextual trajectory evaluation using a single RGB image and a single VLM query. MOSU2 screens the current observation with a VLM and invokes the second stage only when further investigation is requested. The second stage reasons over consecutive RGB observations and retrieves geometric and dynamic measurements from the local map according to the semantic items requested by the first stage. The evaluation addresses three questions: (Q1) whether MOSU2 improves navigation decisions in the scenarios; (Q2) whether it improves the recognition of conditions that require a specific response, and whether correct recognition leads to correct actions; and (Q3) how much reasoning latency and token usage it adds. Table 2 reports the navigation success rate (Nav. Success Rate), reasoning latency (Latency), condition-detection accuracy (Det. Accuracy), and token usage (Tokens) of both methods in each scenario group. Fig. 10 shows the first-person inputs to the VLMs, the VLM reasoning and outputs, and the third-person view of the robot-environment interactions corresponding to the scenarios.

Metrics and protocol. A trial is successful when the robot completes the test without collision or human intervention and its behavior satisfies the requirement of the scenario: it stops in response to the stop gesture, passes the construction site on the side indicated by the signs, enters the crosswalk only when the pedestrian signal permits crossing, and yields to road users whose paths conflict with its own. Remaining stopped when proceeding is permitted and safe also counts as a failure. The navigation success rate is the fraction of successful trials in a scenario group. The condition-detection accuracy is computed over decision steps rather than trials: a step counts as correct when the output identifies the condition that characterizes the scenario group, such as the stop gesture, the construction site, the signal state, a conflicting road user, or the absence of any social or traffic condition that governs the robot at the sidewalk baseline. For MOSU, we evaluate the output of its single VLM query, and for MOSU2, the output of the last executed stage. For MOSU2, the first stage is performed at each step; when further investigation is requested, the second stage executes and retrieves the requested information to make the navigation decision. Because VLM inference runs in the cloud, the latency also includes the network round trip. We also report the input and output tokens per decision step. The token counts of MOSU and MOSU2 are both from the same VLM (ChatGPT-realtime). Because the evaluation consists of real-robot trials, the number of trials in each group is limited, so the group-level differences indicate trends rather than statistically established effects.

Navigation performance (Q1). MOSU2 improves navigation success in four of the five scenario groups and matches MOSU at the sidewalk baseline. The unweighted mean navigation success rate increases from 61.8% for MOSU to 94.2% for MOSU2, representing an absolute improvement of 32.4 percentage points. At signalized crosswalks and on roads shared with vehicles, cyclists, or pedestrians, MOSU2 achieves 100% navigation success compared to 42% and 56% for MOSU, respectively. These dynamic settings require reasoning over traffic signal states and surrounding road-user trajectories, where MOSU's single-frame visual scoring frequently leads to overly conservative stops or premature entry. In construction site environments featuring temporary signage and barriers, MOSU2 increases navigation success from 50% to 100%, demonstrating robust path selection around physical restrictions. For human stop gestures, navigation success increases from 61% to 71%, reflecting improved responsiveness to social cues, though minor edge cases remain when gesture visibility is obscured. MOSU fails in this group mainly because its prompt does not specifically address waving or pausing gestures, so it tends to slowly steer around a person instead of stopping. At the sidewalk baseline, both methods achieve 100% navigation success.

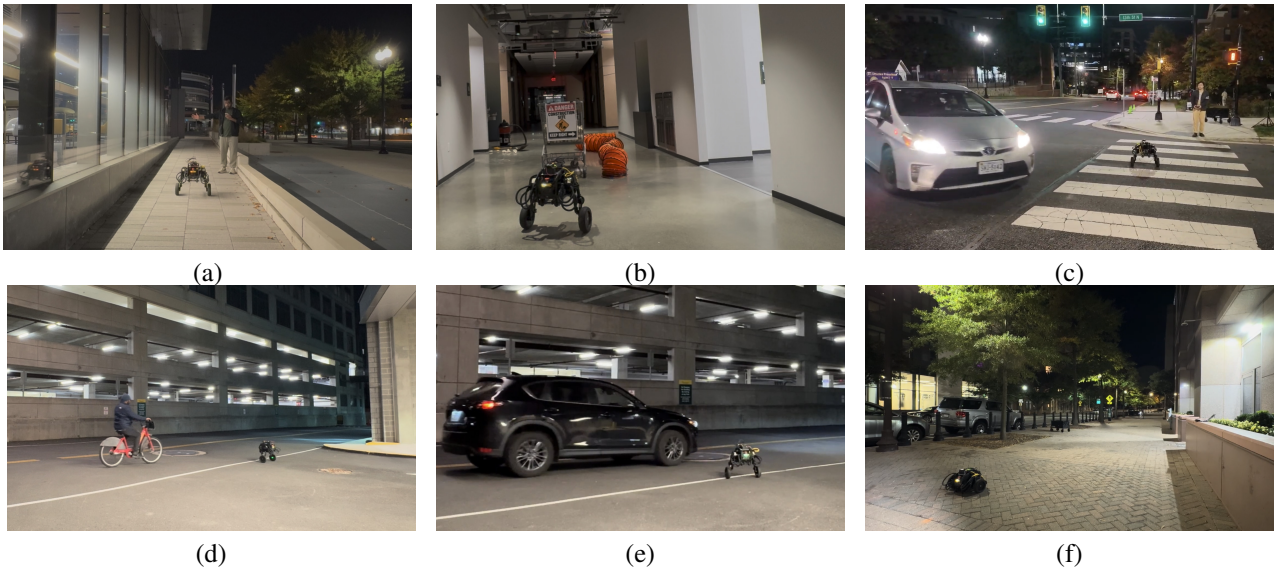


Figure 8. Representative social, traffic, and irregular navigation scenarios used to evaluate MOSU2: (a) responding to a pedestrian's stop gesture, (b) navigating a construction area with warning and directional signs, (c) crossing at a signalized intersection, (d) sharing the road with moving agents, illustrated by a cyclist, (e) intersecting with a vehicle, requiring the robot to analyze the car's velocity to determine if it is yielding, and (f) navigating in sidewalk baseline without any social or traffic situation. Panels (d) and (e) both belong to the shared-road group (Road + Vehicles/Bikes/Peds) in Table 2. These scenarios require interpreting contextual cues and, where relevant, considering the geometry and motion of surrounding road users to determine appropriate robot behavior.

Table 2. Performance Comparison: MOSU vs. MOSU2 across diverse dynamic scenarios. Nav. Success Rate is the fraction of trials in which the robot successfully reached the subgoal, and Det. Accuracy is the accuracy of detecting the social and traffic conditions. Our approach substantially improves the Nav. Success Rate and Det. Accuracy in most of the social and traffic scenarios. Tokens are given as input/output per decision step, and those of MOSU2 are reconstructed from its prompts and outputs.

Scenario	MOSU				MOSU2			
	Nav. Success Rate (%)↑	Latency (s)↓	Det. Accuracy (%)↑	Tokens (in/out)↓	Nav. Success Rate (%)↑	Latency (s)↓	Det. Accuracy (%)↑	Tokens (in/out)↓
Stop gesture	61	**1.51**	70	**821/53**	**71**	2.52	**89**	1,507/135
Construction site	50	**1.68**	**100**	**821/56**	**100**	5.16	**100**	5,386/446
Signalized crosswalk	42	**1.90**	41	**821/65**	**100**	4.72	**93**	4,948/478
Road + Vehicles/Bikes/Peds	56	**1.79**	50	**822/62**	**100**	4.95	**89**	5,142/454
Sidewalk baseline	**100**	**1.74**	**100**	**821/61**	**100**	1.80	86	821/53

Condition detection and navigation decisions (Q2). The unweighted mean condition-detection accuracy rises from 72.2% for MOSU to 91.4% for MOSU2, an increase of 19.2 percentage points. The most pronounced improvement occurs at signalized crosswalks, where detection accuracy improves from 41% to 93%. As shown in the sixth row of Fig. 10, MOSU detects car headlights and other traffic lights and is confused by them, whereas MOSU2 uses the retrieved position of the relevant traffic light to detect the traffic condition more accurately. Detection accuracy also increases from 50% to 89% in shared-road environments with moving vehicles, bicycles, and pedestrians. The third to fifth rows of Fig. 10 show that MOSU2 can estimate the velocities of surrounding objects; when a car is yielding to the robot, MOSU2 keeps the robot moving toward the goal. In contrast, MOSU cannot estimate velocity reliably, so it moves very conservatively. MOSU2 also detects stop gestures more accurately, improving accuracy from 70% to 89%. These gains indicate that MOSU2 recognizes the conditions that require a specific response more reliably than MOSU.

The construction site results highlight the crucial distinction between first-stage detection and second-stage analysis: both MOSU and MOSU2 achieve 100% detection accuracy, yet MOSU2 doubles navigation success from 50% to 100%. This disparity demonstrates that correct scene classification is insufficient on its own; effective autonomy requires grounding detected conditions in measured geometry to understand the shape and semantics of the construction/hazard area. Conversely, at the sidewalk baseline, MOSU2's detection accuracy is 86%, compared with 100% for MOSU, because the first stage does not always recognize that no social or traffic condition governs the robot there; navigation success nevertheless remains at 100% because the traversability analysis foundation is the same as MOSU. Thus, MOSU2 improves overall context recognition and decision reliability without requiring perfect condition classification in every single frame.

Reasoning cost (Q3). The performance gains of MOSU2 come at a higher reasoning cost. Context screening is

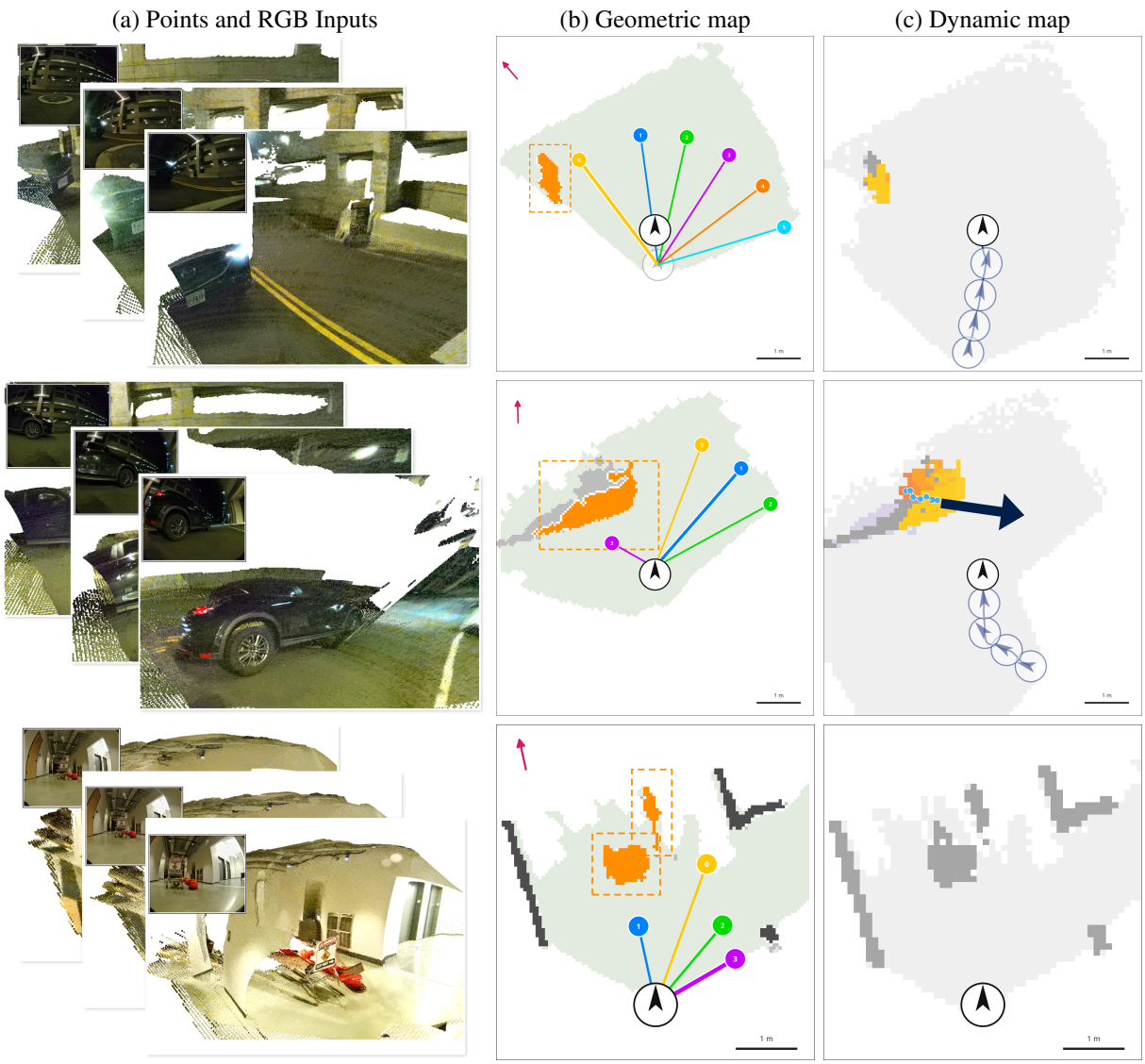


Figure 9. VLM-retrievable local maps in three scenarios (one per row): a stopped car beside the robot's path (top), a car crossing in front of the robot (middle), and a construction site in an indoor corridor (bottom). (a) Consecutive RGB images (insets) and the corresponding LiDAR point clouds colored by the camera images. (b) Geometric layer with the observed free space (light green), segmented objects (orange, dashed boxes), other occupied cells (gray), numbered candidate trajectories, the robot pose (black marker), and the subgoal direction (red arrow). (c) Dynamic layer: the crossing car leaves a track of positions (blue points in the middle figure) with a black arrow to indicate the velocity direction, whereas the stopped car (top) and the construction site (bottom) show no displacement. Blue markers show the robot's recent poses, and the scale bars indicate 1 m.

performed at each decision step, with additional second-stage reasoning over recent RGB observations and local-map evidence when further investigation is requested. The unweighted mean decision latency increases from 1.72 s for MOSU to 3.83 s for MOSU2, approximately 2.2 times as long. The increase is concentrated in the scenario groups that invoke the second stage: at the construction site, the signalized crosswalk, and the shared road, MOSU2 takes 4.72–5.16 s per decision, compared with 1.68–1.90 s for MOSU. For the stop gesture and the sidewalk baseline, in which the first stage typically resolves the scene without invoking the second stage, the latency of MOSU2 (2.52 s and 1.80 s) remains much closer to that of MOSU (1.51 s and 1.74 s). MOSU2 also uses on average about 3,560 input and 310 output tokens per decision step, compared with 820 and 60 for MOSU, that is, about 4.3 and 5.3 times as many. The results therefore characterize a trade-off between higher navigation reliability and a higher reasoning cost per decision.

Conclusion

We presented MOSU2, an extension of MOSU for long-range outdoor navigation that combines route guidance, traversability assessment, and contextual reasoning. Building on MOSU's GPS-based routing and multi-modal trajectory generation and evaluation, MOSU2 introduces a VLM-retrievable local map that associates semantic regions with measured geometry and estimated motion. This information complements consecutive RGB observations when the second stage interprets social interactions, traffic conditions, and irregular situations. A two-stage reasoning architecture determines when additional investigation is needed: in the first stage, a VLM screens the current scene and supports

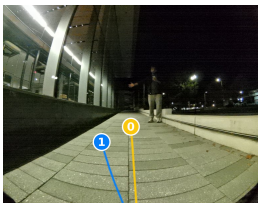
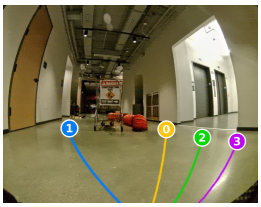
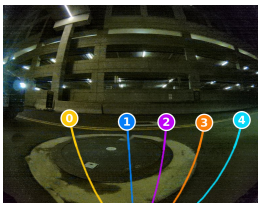
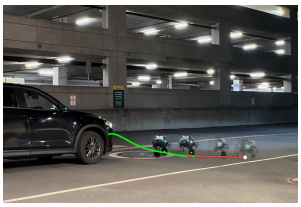
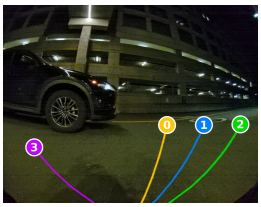
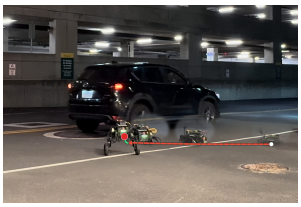
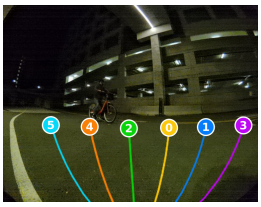
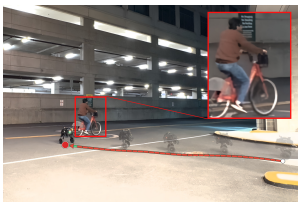
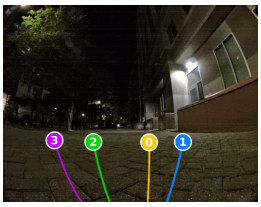
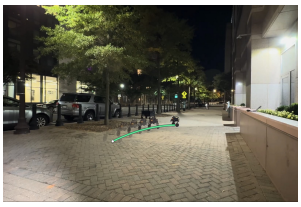
First-Person View	VLM Reasoning & Decision Output	Third-Person View (Execution)
	**MOSU:** mode: Slow, ranking: [1, 0], reason: "A pedestrian is ahead. Path 1 provides more clearance while remaining on the paved walkway...." **MOSU2: Stage 1:** False, "Stop", 0, "no further information needed", "A nearby pedestrian is holding out an open palm in a stop gesture."	
	**MOSU:** mode: Slow, ranking: [0, 2, 3, 1], reason: "The construction sign directs traffic to keep right. Path 0 stays right of the equipment and is closest to the goal; paths 2 and 3 are alternatives. Path 1 approaches the construction equipment." **MOSU2: Stage 1:** True, "Items: construction cart, ducts and sign (static obstacles). Needed: position, and whether the right side is passable. ..." **Stage 2:** Slow, 3, "the site is in the middle of the corridor 5 m ahead, so any path through the center (1 and 0) collides with it or passes too close. The input confirms the sign that the right side is open."	
	**MOSU:** [4, 3, 2, 1, 0], Stop, "A dark vehicle is close on the left, and a round manhole cover and its raised rim sit in front of the robot...." **MOSU2: Stage 1:** True, "Item: black vehicle at the left edge. Needed: position relative to the robot, to judge clearance from path 0, and velocity, to know whether it is stopped or about to pull out." **Stage 2:** Slow, 1, "the goal is 5.7 m away at 29° left, ... The car is stopped, so there is no collision risk now, ..."	
	**MOSU:** [2, 1, 0, 3], "Stop", "A dark SUV is close on the left with its headlight on, and it is moving across or into the robot's path...." **MOSU2:Stage 1:** True, "Item: a black SUV. Needed: position and velocity, to know if it crosses the robot's path. ..." **Stage 2:** Stop, "After the car has passed, take path 0....", "at 2 m/s the car crosses the robot's forward line in about 0.5 s and sweeps across all four candidate paths, so none is safe now. It is only 1 m away, so the robot must not move at all. Once it has crossed to the right, the area in front is clear and path 0 goes toward the goal"	
	**MOSU:** [1, 0, 3, 2, 4, 5], "Stop", "A cyclist is crossing in front of the robot from the left, about 6–8 m away. The goal is 21° right...." **MOSU2: Stage 1:** True, "Item: cyclist (a fast, hard-to-predict road user). Needed: position and velocity, to see where the cyclist will be when the robot would cross." **Stage 2:** Stop, "After the cyclist has passed, take path 0.", "at 1 m/s the cyclist reaches the robot's forward .... robot moving forward would meet the cyclist about 3 m ahead. The robot stops and lets the cyclist cross, which takes about 4–5 s. Path 0 is the direct route to the goal and is clear once the cyclist is past."	
	**MOSU:** [0, 2, 1, 3, 4, 5], "Stop", "The robot is in front of a crosswalk.... Traffic lights ahead show red, and vehicles with bright headlights are further down the road ... The robot should stop until the light and the traffic allow crossing." **MOSU2: Stage 1:** True, "Items: the traffic light (signal state), vehicles (motion) and pedestrians. Needed: which light applies to the robot, and whether the vehicles are moving." **Stage 2 Decision:** Slow, 0, "the pedestrian light at (8, 1) is on, so the robot has the right of way. The vehicles are stopped about 11 m away on the right, and the second light at (5, -10) is on their side of the road.... Slow speed is used because it is dark, the headlights are bright, and a vehicle or pedestrian could still start moving."	
	**MOSU:** [2, 0, 1, 3], "Normal" "This is a brick-paved sidewalk at night with a building..." **MOSU2: Stage 1:** False, Normal, 0, "Path 0 goes straight to the goal and keeps the most distance from the wall on the right and the left edge. Normal speed is fine because the route is clear. I would still watch the brick surface in the dark."	

Figure 10. Qualitative evaluation of MOSU2 across seven representative interactions from the five scenario groups: Left column displays the onboard egocentric camera observation (First-Person View). Middle column shows the textual reasoning output from the two-stage reasoning, including screening decisions, map retrieval, and chosen motion. Right column shows the external observer view (Third-Person View) demonstrating physical execution.

routine subgoal-directed navigation, while the second stage retrieves relevant evidence and uses it to select a trajectory and velocity mode. Together, these components connect contextual understanding with environmental measurements and robot-specific traversability.

In real-world experiments across five scenario groups, MOSU2 achieved 100% navigation success in four groups and 71% in the stop-gesture group, whereas MOSU succeeded in 42–61% of the trials in the four groups other than the sidewalk baseline. MOSU2 detected the relevant conditions with 86–100% accuracy, at the cost of a higher reasoning latency when the second stage is invoked (4.7–5.2 s, compared with 1.7–1.9 s for MOSU) and about four times as many input tokens on average.

Limitations and Future Work

MOSU2 depends on the quality of its geometric, semantic, and dynamic observations. Occlusions, sparse LiDAR returns, segmentation errors, and calibration or odometry inaccuracies can affect the local map. In particular, associating segments by semantic category and spatial proximity can produce incorrect correspondences when several nearby objects share the same category. Changes in the visible portion of an object can also shift its observed centroid and introduce errors into the velocity estimate. Future work will investigate more robust temporal association and motion estimation, together with uncertainty estimates that allow the reasoning process to distinguish reliable measurements from incomplete or ambiguous evidence.

The conditional two-stage architecture depends on the first stage recognizing situations that require further investigation. A missed cue can prevent the second stage from being invoked, while unnecessary escalation increases reasoning cost. Even routine decisions require first-stage inference, and an escalated decision additionally requires map retrieval and second-stage inference. Conditional invocation therefore does not by itself establish a latency advantage over the single-query MOSU baseline. Future work will investigate uncertainty-aware escalation, lightweight screening models, and mechanisms for revisiting decisions as new observations arrive. Evaluation across a broader range of rare and unfamiliar situations will be necessary to characterize the trade-off between reasoning cost and missed contextual requirements.

Even when additional evidence is available, VLM interpretations can remain ambiguous or incorrect. Gestures, signal relevance, and vehicle behavior may admit several explanations, and motion measurements alone cannot establish another road user's intent. With reasoning latencies up to 5.16 s (Table 2), the scene can also change before a decision is executed. Future work will investigate explicit handling of conflicting evidence, checks on observation freshness, and independent verification of selected motions against updated geometric and dynamic observations.

Funding

The authors received no financial support for the research, authorship, and/or publication of this article.

Declaration of conflicting interests

The authors declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

References

Alverhed E, Hellgren S, Isaksson H, Olsson L, Palmqvist H and Floden J (2024) Autonomous last-mile delivery robots: a literature review. *European Transport Research Review* 16(1): 4.

Berezina K, Ciftci O and Cobanoglu C (2019) Robots, artificial intelligence, and service automation in restaurants. In: *Robots, artificial intelligence, and service automation in travel, tourism and hospitality*. Emerald Publishing Limited, pp. 185–219.

Bresenham JE (1998) Algorithm for computer control of a digital plotter. In: *Seminal graphics: pioneering efforts that shaped the field*, volume 1. New York, NY, USA: Association for Computing Machinery (ACM), pp. 45–50. DOI:10.1145/280811.280913.

Carion N, Gustafson L, Hu YT, Debnath S, Hu R, Suris Coll-Vinent D, Ryali C, Alwala KV, Khedr H, Huang A et al. (2026) Sam 3: Segment anything with concepts. In: *International conference on learning representations*, volume 2026. pp. 138846–138923.

Chen B, Xia F, Ichter B, Rao K, Gopalakrishnan K, Ryoo MS, Stone A and Kappler D (2023) Open-vocabulary queryable scene representations for real world planning. In: *2023 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, pp. 11509–11522.

Chen C, Demir E, Huang Y and Qiu R (2021) The adoption of self-driving delivery robots in last mile logistics. *Transportation research part E: logistics and transportation review* 146: 102214.

Davids A (2002) Urban search and rescue robots: from tragedy to technology. *IEEE Intelligent systems* 17(2): 81–83.

DeSouza GN and Kak AC (2002) Vision for mobile robot navigation: A survey. *IEEE transactions on pattern analysis and machine intelligence* 24(2): 237–267.

Eder M, Prinz R, Schöggl F and Steinbauer-Wagner G (2023) Traversability analysis for off-road environments using locomotion experiments and earth observation data. *Robotics and Autonomous Systems* 168: 104494.

Fang Z, Xiao A, Hsu D and Lee GH (2026) From obstacles to etiquette: Robot social navigation with vlm-informed path selection. *IEEE Robotics and Automation Letters* .

Fankhauser P, Bloesch M and Hutter M (2018) Probabilistic terrain mapping for mobile robots with uncertain localization. *IEEE Robotics and Automation Letters (RA-L)* 3(4): 3019–3026. DOI:10.1109/LRA.2018.2849506.

Gao P, Liu Z, Wu Z and Wang D (2019) A global path planning algorithm for robots using reinforcement learning. In: *2019 IEEE International Conference on Robotics and Biomimetics (ROBIO)*. IEEE, pp. 1693–1698.

Gasparetto A, Boscariol P, Lanzutti A and Vidoni R (2015) Path planning and trajectory planning algorithms: A general overview. *Motion and Operation Planning of Robotic Systems: Background and Practical Approaches* : 3–27.

Gul F, Rahiman W and Nazli Alhady SS (2019) A comprehensive study for robot navigation techniques. *Cogent Engineering* 6(1): 1632046.

Guo Y, Wang H, Hu Q, Liu H, Liu L and Bennamoun M (2020) Deep learning for 3d point clouds: A survey. *IEEE transactions on pattern analysis and machine intelligence* 43(12): 4338–4364.

Kalita H and Thangavelautham J (2020) Exploration of extreme environments with currentand emerging robot systems. *Current Robotics Reports* 1(3): 97–104.

Kim Y, Lee JH, Lee C, Mun J, Youm D, Park J and Hwangbo J (2024) Learning semantic traversability with egocentric video and automated annotation strategy. *IEEE Robotics and Automation Letters* .

Kong Y, Song D, Liang J, Manocha D, Yao Z and Xiao X (2025) Autospatial: Visual-language reasoning for social robot navigation through efficient spatial reasoning learning. *arXiv preprint arXiv:2503.07557* .

Li Y, Bu R, Sun M, Wu W, Di X and Chen B (2018) Pointcnn: Convolution on x-transformed points. *Advances in neural information processing systems* 31.

Liang J (2025) *Robot Navigation in Complex Scenarios*. PhD Thesis, University of Maryland, College Park.

Liang J, Das D, Song D, Shuvo MNH, Durrani M, Taranath K, Penskiy I, Manocha D and Xiao X (2025) Gnd: Global navigation dataset with multi-modal perception and multi-category traversability in outdoor campus environments. In: *2025 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, pp. 2383–2390.

Liang J, Gao P, Xiao X, Sathyamoorthy AJ, Elnoor M, Lin MC and Manocha D (2024a) Mtg: Mapless trajectory generator with traversability coverage for outdoor navigation. In: *2024 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, pp. 2396–2402.

Liang J, Patel U, Sathyamoorthy AJ and Manocha D (2021) Crowd-steer: Realtime smooth and collision-free robot navigation in densely crowded scenarios trained using high-fidelity simulation. In: *Proceedings of the twenty-ninth international conference on international joint conferences on artificial intelligence*. pp. 4221–4228.

Liang J, Payandeh A, Song D, Xiao X and Manocha D (2024b) Dtg: Diffusion-based trajectory generation for mapless global navigation. In: *2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, pp. 5340–5347.

Liang J, Weerakoon K, Song D, Kirubaharan S, Xiao X and Manocha D (2026) Mosu: Autonomous long-range robot navigation with multi-modal scene understanding. In: *International Symposium on Experimental Robotics*. Springer.

Liu X, Li J, Jiang Y, Sujay N, Yang Z, Zhang J, Abanes J, Zhang J and Feng C (2025) Citywalker: Learning embodied urban navigation from web-scale videos. In: *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, pp. 6875–6885.

Mavrogiannis C, Baldini F, Wang A, Zhao D, Trautman P, Steinfeld A and Oh J (2023) Core challenges of social robot navigation: A survey. *ACM Transactions on Human-Robot Interaction* 12(3): 1–39.

Nasiriany S, Xia F, Yu W, Xiao T, Liang J, Dasgupta I, Xie A, Driess D, Wahid A, Xu Z et al. (2024) Pivot: Iterative visual prompting elicits actionable knowledge for vlms. *arXiv preprint arXiv:2402.07872* .

Ozturk U, Akdaug M and Ayabakan T (2022) A review of path planning algorithms in maritime autonomous surface ships: Navigation safety perspective. *Ocean Engineering* 251: 111010.

Payandeh A, Song D, Nazeri M, Liang J, Mukherjee P, Raj AH, Kong Y, Manocha D and Xiao X (2025) Social-llava: Enhancing social robot navigation through human-language reasoning. In: *2025 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, pp. 17192–17198.

Rana K, Haviland J, Garg S, Abou-Chakra J, Reid I and Suenderhauf N (2023) Sayplan: Grounding large language models using 3d scene graphs for scalable task planning. In: *7th Annual Conference on Robot Learning*. URL https://openreview.net/forum?id=wMpOMO0Ss7a.

Roth P, Nubert J, Yang F, Mittal M and Hutter M (2024) Viplanner: Visual semantic imperative learning for local navigation. In: *2024 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, pp. 5243–5249.

Sathyamoorthy AJ, Weerakoon K, Guan T, Liang J and Manocha D (2022) Terrapn: Unstructured terrain navigation using online self-supervised learning. In: *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, pp. 7197–7204.

Schmittle M, Baijal R, Hatch N, Scalise R, Guaman Castro M, Talia S, Khetarpal K, Srinivasa S and Boots B (2025) Long range navigator (lrn) : Extending robot planning horizons beyond metric maps. *arXiv* .

Shah D, Sridhar A, Dashora N, Stachowicz K, Black K, Hirose N and Levine S (2023) Vint: A foundation model for visual navigation. *arXiv preprint arXiv:2306.14846* .

Sohail SS, Himeur Y, Kheddar H, Amira A, Fadli F, Atalla S, Copiaco A and Mansoor W (2024) Advancing 3d point cloud understanding through deep transfer learning: A comprehensive survey. *Information Fusion* : 102601.

Sohn K, Lee H and Yan X (2015) Learning structured output representation using deep conditional generative models. *Advances in neural information processing systems* 28.

Song D, Liang J, Payandeh A, Raj AH, Xiao X and Manocha D (2025a) Vlm-social-nav: Socially aware robot navigation through scoring using vision-language models. *IEEE Robotics and Automation Letters* 10(1): 508–515. DOI:10.1109/LRA.2024.3511409.

Song D, Liang J, Xiao X and Manocha D (2025b) Vl-tgs: Trajectory generation and selection using vision language models in mapless outdoor environments. *IEEE Robotics and Automation Letters* 10(6): 5791–5798. DOI:10.1109/LRA.2025.3559822.

Sridhar A, Shah D, Glossop C and Levine S (2024) Nomad: Goal masked diffusion policies for navigation and exploration. In: *2024 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, pp. 63–70.

Team G (2024) Gemini: A family of highly capable multimodal models. URL https://arxiv.org/abs/2312.11805.

Van Den Berg J, Snape J, Guy SJ and Manocha D (2011) Reciprocal collision avoidance with acceleration-velocity obstacles. In: *2011 IEEE International Conference on Robotics and Automation*. IEEE, pp. 3475–3482.

Van Diggelen F and Enge P (2015) The world's first gps mooc and worldwide laboratory using smartphones. In: *Proceedings of*

the 28th international technical meeting of the satellite division of the institute of navigation (ION GNSS+ 2015). pp. 361–369.

Wang Z, Fang H, Wang S, Luo Y, Dong H, Li W and Gan Y (2026) Hydra-nav: Object navigation via adaptive dual-process reasoning. *arXiv pre-print* URL https://arxiv.org/abs/2602.09972.

Weerakoon K, Elnoor M, Seneviratne G, Rajagopal V, Arul SH, Liang J, Jaffar MKM and Manocha D (2025) Behav: Behavioral rule guided autonomy using vlms for robot navigation in outdoor scenes. In: *2025 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, pp. 7044–7051.

Weerakoon K, Sathyamoorthy AJ, Liang J, Guan T, Patel U and Manocha D (2023) Graspe: Graph based multimodal fusion for robot navigation in outdoor environments. *IEEE Robotics and Automation Letters* 8(12): 8090–8097.

Wei M, Wan C, Peng P, Yu X, Yang Y, Feng D, Cai W, Zhu C, Wang T, Pang J et al. (2026) Ground slow, move fast: A dual-system foundation model for generalizable vision-language navigation. In: *International Conference on Learning Representations*, volume 2026. pp. 12380–12396.

Wellhausen L and Hutter M (2021) Rough terrain navigation for legged robots using reachability planning and template learning. In: *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, pp. 6914–6921.

Wijayathunga L, Rassau A and Chai D (2023) Challenges and solutions for autonomous ground robot scene understanding and navigation in unstructured outdoor environments: A review. *Applied Sciences* 13(17): 9877.

Wu T, He S, Liu J, Sun S, Liu K, Han QL and Tang Y (2023) A brief overview of chatgpt: The history, status quo and potential future development. *IEEE/CAA Journal of Automatica Sinica* 10(5): 1122–1136.

Zemke DMV, Tang J, Raab C and Kim J (2020) How to build a better robot... for quick-service restaurants. *Journal of Hospitality & Tourism Research* 44(8): 1235–1269.

Zhang J, Xu J, Yu Y, Guo J, Hu R, Bao H, Zhou X and Peng S (2026a) Supernav: An agentic navigation system for any task in any scene. *arXiv preprint arXiv:2610.12126* .

Zhang L, Hao X, Bu X, Tang Y, Li H, Lu J, Wei Xs, Ma J, Liu Y, Zhang J et al. (2026b) Walk with me: Long-horizon social navigation for human-centric outdoor assistance. *arXiv preprint arXiv:2604.26839* .