DS-LABRNav: Land-Air Bimodal Robot Navigation With Traversable Obstacles Base on Vision-Language Model

Yongjie Li, Wenshuai Yu, *Member, IEEE*, Molong Duan, *Member, IEEE*, Bo Zhang, Zhou Liu, and Qingquan Li

***Abstract*—Land–air bimodal robots (LABR) provide efficient ground locomotion and flexible aerial mobility, making them suitable for complex navigation tasks. Traditional methods cannot distinguish between traversable obstacles (such as curtains) and rigid obstacles, resulting in unnecessary aerial transitions. To address these challenges, DS-LABRNav is proposed as a navigation framework that integrates a Vision-Language Model (Align-DS-V) into land–air planning. This framework consists of an event-triggered Obstacle-Induced Takeoff Trigger (OITT) and a modality-guided occupancy map update. OITT selectively activates the VLM reasoning only prior to a proposed aerial maneuver while the robot is safely on the ground. The occupancy map update then incorporates semantic reasoning results, enabling the planner to re-optimize the trajectory and prioritize land modalities for improved energy efficiency. Specific experiments demonstrate that DS-LABRNav improves navigation efficiency and path optimality while reducing mode-switching costs. Compared to traditional map-based approaches, the framework achieves energy savings of up to 70% in scenarios with traversable obstacles and unknown slopes, validating its effectiveness in real-world LABR deployments.**

***Index Terms*—Land–air bimodal robots, vision-language model, autonomous navigation.**

I. Introduction

LAND-AIR bimodal robots (LABR) have attracted growing attention for their ability to integrate efficient ground locomotion with agile aerial mobility. By switching between modes, these robots can overcome obstacles, access elevated targets, and conserve energy. This versatility makes LABR well suited for applications such as rescue, reconnaissance, and delivery [1], [2], [3], [4], [5]. Autonomous navigation for LABR requires not only reliable planning and obstacle avoidance but also intelligent mode switching. In many real-world environments, obstacles such as curtains or gentle slopes are physically traversable for ground robots. However, conventional planners [6], [7], [8] often classify these elements as rigid barriers, resulting in unnecessary takeoffs and inefficient trajectories. Therefore, effective use of traversable obstacles is essential for LABR navigation in complex environments.

Traditional LABR navigation methods [7] often suffer from limited sensing ranges, leading to conservative behaviors and unnecessary aerial transitions. Learning-based approaches, such as AGRNav [9], improve navigation efficiency in occlusion-prone environments, but their higher inference latency hinders real-time applications. To improve computational efficiency, recent frameworks like HE-Nav [10] have introduced lightweight semantic scene completion networks tailored for cluttered environments. Furthermore, highly efficient architectures such as OccRWKV [11] and OMEGA [12] exploit linear-complexity models to enable real-time 3D semantic occupancy prediction in dynamic environments. Previous work [13] also partially addressed global planning limitations by proposing the Global Keypoint Prediction Network (GKPN), which optimizes hierarchical waypoints to minimize flight distances. As shown in Fig. 1, despite these advances in computational efficiency and global guidance, existing methods remain limited by their reliance on purely geometric representations or basic semantic labels. They lack the high-level reasoning required to distinguish impassable stairs from traversable curtains, resulting in energy-inefficient takeoffs.

Recent advances in Vision–Language Model (VLM) provide a promising solution to the lack of semantic reasoning in traditional navigation pipelines [14], [15], [16], [17]. By achieving zero-shot scene understanding, VLM can assist robot navigation while providing the possibility to assess the traversability of obstacles. For example, LM-Nav [18] successfully navigates hundreds of meters in complex, real-world outdoor environments by combining three independent, pretrained models, demonstrating its ability to follow multi-step instructions. AerialVLN [19] integrates low-altitude image mapping with language-command parsing, enabling drones to execute semantic navigation tasks outdoors from natural-language instructions. However, most existing studies [18], [19], [20], [21] focus on single-modal

Received 7 November 2025; accepted 26 April 2026. Date of publication 29 May 2026; date of current version 5 June 2026. This article was recommended for publication by Associate Editor Yiyang Zhou and Editor Ashis Banerjee upon evaluation of the reviewers' comments. This work was supported in part by Guangdong Basic and Applied Basic Research Foundation, under Grant 2026A1515011407, in part by the Guangdong Laboratory of Artificial Intelligence and Digital Economy (Shenzhen), under Project 23501002, and in part by the National Natural Science Foundation of China, under Grant 62503338. *(Corresponding authors: Zhou Liu; Qingquan Li.)*

Yongjie Li is with the College of Computer Science and Software Engineering, Shenzhen University, Shenzhen 518060, China, and also with the Guangdong Laboratory of Artificial Intelligence and Digital Economy, Shenzhen 518132, China (e-mail: 2450101005@mails.szu.edu.cn).

Wenshuai Yu and Qingquan Li are with the Shenzhen University, Shenzhen 518060, China (e-mail: liqq@szu.edu.cn).

Molong Duan is with the Hong Kong University of Science and Technology, Hong Kong.

Bo Zhang and Zhou Liu are with the Guangdong Laboratory of Artificial Intelligence and Digital Economy, Shenzhen 518132, China (e-mail: liuzhou@gml.ac.cn).

This article has supplementary downloadable material available at https://doi.org/10.1109/LRA.2026.3698272, provided by the authors.

Digital Object Identifier 10.1109/LRA.2026.3698272

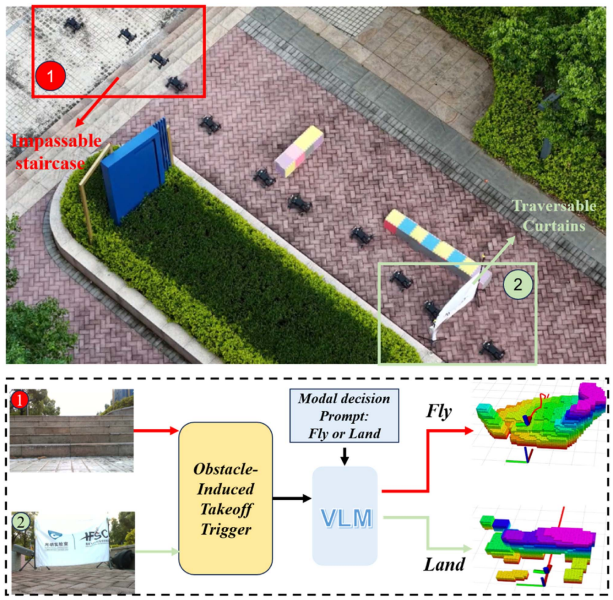

Fig. 1. Real-world scenes and modality decision process. **Top:** Real-world scenarios requiring modal decisions in a land-air bimodal navigation system. Region ① shows an *impassable staircase* that triggers aerial transition, while ② shows *traversable curtains* suitable for land traversal. **Bottom:** Visual input from ① and ② is processed by the obstacle trigger. Based on the image and prompts, VLM infers the passability of obstacles and selects the appropriate mode for trajectory planning.

robots, leaving land–air bimodal systems largely unexplored. Furthermore, continuously invoking cloud-based VLM to generate navigation actions introduces considerable latency and computational overhead, conflicting with the real-time requirements of LABR navigation.

To address these challenges, DS-LABRNav is proposed for land–air bimodal robots. Vision–Language Model are incorporated into the planning pipeline to support traversability reasoning, allowing the distinction between traversable obstacles (e.g. curtains) and rigid barriers without additional training data. Unlike existing methods that rely on continuous VLM queries, the proposed framework adopts an event-triggered mechanism, the Obstacle-Induced Takeoff Trigger (OITT). The VLM inference is activated only in ground mode when the local planner is planning the air trajectory. The resulting modal decisions are used to improve the occupancy map. This strategy avoids unnecessary takeoffs and enables the re-optimization of energy-efficient, ground-preferred paths. In this way, the framework achieves a balance between high-level semantic reasoning and the real-time constraints of LABR navigation.

The contributions of this work are summarized as follows:

- A VLM-based framework DS-LABRNav is proposed to infer obstacle traversability, enabling LABR to distinguish traversable obstacles from rigid barriers.
- An event-triggered mechanism OITT and a modality-guided occupancy map update balance real-time performance and navigation accuracy by invoking VLM inference only when aerial motion is required.
- To the best of our knowledge, this is the first work to validate the effectiveness of VLM-based modality decision-making in LABR navigation. Compared with traditional methods, the proposed approach achieves up to 70% energy savings by avoiding unnecessary aerial maneuvers in traversable obstacle scenarios.

II. Related Work

A. Autonomous Navigation for Land-Air Bimodal Robots

Navigation for land-air bimodal robots (LABR) has transitioned from traditional mapping to learning-based perception. Early mapping planners [7], [22], [23] construct local maps from onboard sensors but frequently fail in complex, unknown terrains due to limited sensing horizons. To enhance global awareness, learning-based approaches like AGRNav [9] utilize semantic scene completion. Recent frameworks have further resolved the associated computational bottlenecks through lightweight BEV perception (HE-Nav [10], OMEGA [12]) and linear-complexity sequence modeling (OccRWKV [11]) for highly efficient, occlusion-aware mapping. Additionally, prior work [13] optimized global guidance via hierarchical land-air keypoint generation. However, despite achieving real-time efficiency and robust geometric mapping, these state-of-the-art methods fundamentally lack deep semantic physical reasoning; consequently, they cannot intelligently differentiate traversable soft obstacles from rigid barriers.

B. Semantic Analysis and Occupancy Mapping

Several prior works [24], [25], [26], [27], [28] have attempted to incorporate visual cues into map updates. For example, [27] created a dense traversability map by annotating a geometric point cloud with semantic labels, enabling autonomous navigation of an off-road vehicle. Moreover, [28] dynamically generates new obstacle maps for different embodiments of a given image by using a list of obstacle categories. However, in our task, without any prior knowledge of obstacles, a suitable semantic map cannot be constructed, requiring the robot to plan costly takeoff trajectories to avoid obstacles.

C. Robot Navigation Based on Vision-Language Model

Recent breakthroughs in Vision-Language Model [18], [29], [30] have catalyzed a paradigm shift in robot navigation, moving beyond purely geometric methods to incorporate semantic understanding and commonsense reasoning. These models trained on vast datasets of text and images can interpret complex scenes and follow natural language instructions, fundamentally enhancing a robot's ability to operate in unstructured environments. Specifically, [18] uses a VLM to associate images with text and then uses a visual navigation model to generate feasible paths. However, existing VLM-based navigation approaches often assume that detected landmarks are non-traversable, which limits their applicability in scenarios where obstacles (e.g., flexible barriers) may be safely crossed. In contrast, our work leverages VLM reasoning not for landmark detection, but for traversability assessment. This semantic understanding is directly integrated into occupancy map updates, enabling land–air bimodal robots to re-optimize their trajectories and avoid unnecessary aerial transitions.

III. Problem Statement

*a) Baseline*Previous research proposed a lightweight two-stage framework that combines global keypoint prediction with

local trajectory optimization [13]. In the first stage, a Global Key points Prediction Network (GKPN) converts a depth image into an observation embedding via a perception network. This embedding, combined with the goal position feature, is then inputted into a network that predicts a keypoint path K consisting of n land and aerial keypoints. In the second stage, the predicted keypoints segment the global path, which is then interpolated and optimized by a mapping-based planner to produce a collision-free trajectory.

b) Task Formulation We consider the problem of autonomous navigation for a land–air bimodal robot (LABR) toward a specified goal in a three-dimensional workspace. Let $\mathcal{S} \subset \mathbb{R}^3$ denote the navigation workspace. The obstacle set is given by $\mathcal{S}_{\text{obs}} \subset \mathcal{S}$. It is partitioned into $\mathcal{S}_{\text{obs}} = \mathcal{S}_{\text{trav}} \cup \mathcal{S}_{\text{rigid}}$ where $\mathcal{S}_{\text{trav}}$ denotes traversable obstacles (e.g., curtains), and $\mathcal{S}_{\text{rigid}}$ denotes non-traversable obstacles.

At each time step t_0, the robot is at position $\mathbf{p}_{t_0} \in \mathbb{R}^3$, receives a depth observation $\mathcal{D}_{t_0} \in \mathbb{R}^{H\times W}$, and is given a goal position $\mathbf{p}_{t_f} \in \mathbb{R}^3$. The objective is to generate a trajectory $\tau(t), t \in [t_0, t_f]$ that drives the robot from the initial position $\mathbf{p}_{t_0}$ to the goal $\mathbf{p}_{t_f}$, while maintaining a safe distance from non-traversable obstacles $\mathcal{S}_{\text{rigid}}$ and exploiting traversable obstacles $\mathcal{S}_{\text{trav}}$ to minimize energy cost.

IV. Method

A. Integrating VLM for Modal Decision

The proposed framework employs a Vision-Language Model (VLM) agent, specifically Align-DS-V [14], to assess obstacle traversability during land mode. Based on DeepSeek R1 [31], this multimodal architecture effectively fuses visual and textual input to perform fine-grained semantic reasoning, allowing the robot to identify material properties that geometric sensors cannot perceive.

To formalize this reasoning process and overcome the semantic ambiguity of single-frame inference, the decision space of the VLM is defined as:

$$\mathcal{A} = VLM(O_{seq}, I_{conf}), \tag{1}$$

where $O_{seq} = \{O_{t_1}, O_{t_2}, O_{t_3}\}$ represents a brief temporal sequence of visual data captured by onboard sensors, and I_{conf} is the confidence-aware natural language prompt. Feeding a temporal sequence rather than a static frame enables the model to observe dynamic physical variations, such as a curtain swaying. A critical challenge in semantic navigation is the inherent ambiguity. To address this, I_{conf} explicitly instructs the VLM to internally evaluate a continuous probability confidence score $C_k \in [0, 1]$ for each individual frame O_{t_k} before making a definitive decision.

Although semantic uncertainty is internally modeled via these continuous scores, the final output must provide an unambiguous, actionable directive to the low-level path planner to prevent control instability. Therefore, the VLM's output is strictly constrained to a discrete binary set $\mathcal{A} \in \{0, 1\}$ (0 for land traversal, 1 for required flight). The selection of $\mathcal{A}$ is governed by a strict multi-frame consensus strategy, evaluating the internal confidence C_k against a conservative safety threshold $C_{th} = 0.85$. Specifically, the framework outputs $\mathcal{A} = 0$ only if the obstacle is semantically identified as soft and $C_k \geq C_{th}$ across all frames in the sequence. If any single frame exhibits ambiguity ($C_k < C_{th}$), the model conservatively defaults to $\mathcal{A} = 1$, thereby prioritizing absolute physical safety. Several examples of these modality decisions are illustrated in Fig. 2.

Fig. 2. The example of Align-DS-V decision-making.

B. Obstacle-Induced Takeoff Trigger for VLM Reasoning

In practical navigation scenarios, invoking a cloud-based Vision-Language Model (VLM) at every planning step is computationally inefficient and prone to network-induced latency. Moreover, in simple and unambiguous environments, navigation can proceed without continuous VLM feedback. For land–air bimodal robots, the energy cost of aerial maneuvers is much higher than that of ground detours, making it essential to minimize unnecessary takeoffs.

To this end, we propose an event-triggered Obstacle-Induced Takeoff Trigger (OITT). OITT is dynamically driven by the local planner based on real-time depth mapping. During land navigation, as the depth camera continuously constructs the local occupancy map, the local planner evaluates ground traversability. If an impending obstacle obstructs the ground path, the planner will propose an evasive aerial trajectory. The VLM query and image capture are initiated at the moment. Because the local planner inherently reacts as soon as the obstacle is reliably mapped within the sensor's optimal range, this event-driven mechanism naturally ensures that the obstacle is optimally positioned within the camera's field of view (FOV). This captures the comprehensive structural and semantic details necessary for reasoning, as shown in the upper half of Fig. 3(a). Specifically, let z_h denote the critical altitude threshold. Given a candidate trajectory τ_i, an event $e_t = 1$ is triggered at time t if:

$$e_t = \begin{cases} 1 & \text{if } \exists\, \tau_i^z > z_h, \\ 0 & \text{otherwise.} \end{cases} \tag{2}$$

Upon trigger activation ($e_t = 1$), the captured temporal sequence is transmitted to the VLM. The system switches to flight mode only if the VLM confirms a rigid obstacle ($e_t = 1 \wedge \mathcal{A} = 1$). Conversely, if $e_t = 1 \wedge \mathcal{A} = 0$, indicating that the obstacle is traversable ($\mathcal{S}_{\text{trav}}$), the occupancy map is updated to replan the ground path.

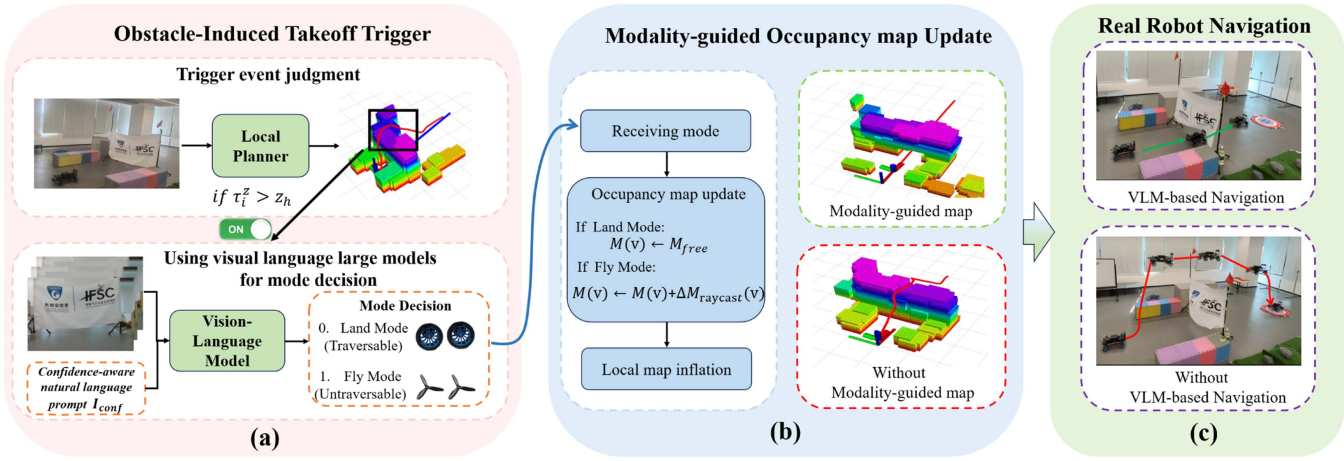

Fig. 3. Overview of the Event-triggered VLM integration pipeline. (a) The Obstacle-Induced Takeoff Trigger module is triggered when takeoff is required and uses VLM to determine the current mode from images and text. (b) The output mode based on VLM guides the occupancy map update. (c) A efficient path can be planned with an modality-guided occupancy map.

C. Modality-Guided Occupancy Map Update

The occupancy map is selectively updated based on the decision to call the VLM after the OITT is triggered, ensuring that semantic reasoning guides trajectory planning. The environment is represented by a voxelized occupancy grid $M(\mathbf{v})$, where each voxel $\mathbf{v}$ is associated with an occupancy probability in log-odds form. These maps are incrementally updated using raycasting from the sensor origin to each valid depth measurement:

$$M_{\text{update}}(\mathbf{v}) = M(\mathbf{v}) + \Delta M_{\text{raycast}}(\mathbf{v}), \tag{3}$$

where $\Delta M_{\text{raycast}}(\mathbf{v})$ represents the confidence-adjusted contribution of the current observation.

Conventional updates treat all detected obstacles as rigid, which often leads to overly conservative trajectories. To overcome this, we integrate the VLM decision $\mathcal{A}$ into the update rule:

$$M(\mathbf{v}) \leftarrow \begin{cases} M_{\text{free}}, & \text{if } \mathcal{A} = 0 \text{ and} \\ & \|\mathbf{p_v} - \mathbf{p}_{\text{cam}}\| \le d_{\text{clear}}, \\ M_{\text{update}}, & \text{otherwise}, \end{cases} \tag{4}$$

where $\mathbf{p_v}$ is the voxel position, and $\mathbf{p}_{\text{cam}}$ is the camera pose. M_{free} denotes that the space is not occupied. When $\mathcal{A} = 0$ (i.e., the region is $\mathcal{S}_{\text{trav}}$), voxels within a clearance distance d_{clear} are explicitly marked as M_{free} to enable more aggressive planning. Otherwise, standard raycasting applies to update the occupancy probability.

As illustrated in Fig. 3(b)–(c), when an obstacle is identified as traversable, the updated occupancy map enables the planner to re-optimize the trajectory for efficient ground navigation. Otherwise, the map remains conservative, and the aerial maneuver is executed. By embedding modal reasoning into map construction, the proposed framework reduces redundant aerial transitions, decreases energy consumption, and improves navigation efficiency.

D. Energy-Saving and Efficient Hierarchical Path Planner

To further improve the quality of the generated paths, hierarchical optimization is applied to refine the initial trajectory obtained from global guidance. The trajectory is formulated as a uniform B-spline of degree p_b, uniquely determined by a set of control points $\mathcal{P} = \{P_0, P_1, \ldots, P_N\}$. Specifically, the trajectory optimization process is separated into ground and aerial segments. For the ground trajectory, the robot is assumed to travel over a planar surface. Hence, only two-dimensional control points are considered

$$\mathcal{P}_{land} = \{P_{t_0}, P_{t_1}, \ldots, P_{t_{L-1}}, P_{t_L}\}, \tag{5}$$

where $P_{t_i} = (x_{t_i}, y_{t_i}),\ \ i \in [0, L]$. Meanwhile, the control points of the aerial trajectory are denoted as $\mathcal{P}_{air}$. The refinement is formulated as a weighted optimization problem designed by Zhang et al. [7] to balance smoothness, safety, and dynamic feasibility. The overall objective is

$$f_{\text{total}} = \lambda_s f_s + \lambda_c f_c + \lambda_f (f_v + f_a) + \lambda_n f_n, \tag{6}$$

where $\lambda_s, \lambda_c, \lambda_f, \lambda_n$are weights for each cost term. f_s is the smoothness term, f_v and f_a are velocity and acceleration penalties, and f_n constrains ex cessive curvature for stable ground motion, as in [7].

The collision cost f_c is defined as the repulsive potential exerted by obstacles at each control point, evaluated with respect to the modality-guided occupancy map $M(\mathbf{v})$:

$$f_c = \sum_{i=p_b}^{N-p_b} F_c\left(d(P_i)\right), \tag{7}$$

where P_i is the i-th control point of the B-spline, and $d(P_i)$ is the distance from P_i to the closest obstacle stored in the updated occupancy map $M(\mathbf{v})$. The potential cost function F_c is given by

$$F_c\left(d(P_i)\right) = \begin{cases} \left(d(P_i) - d_{\text{thr}}\right)^2, & d(P_i) \le d_{\text{thr}}, \\ 0, & d(P_i) > d_{\text{thr}}, \end{cases} \tag{8}$$

where d_{thr} is a clearance threshold ensuring that trajectory points remain safely separated from untraversable obstacles. Meanwhile, regions recognized as traversable in the VLM-guided map are allowed, enabling the planner to generate smoother and more energy-efficient trajectories.

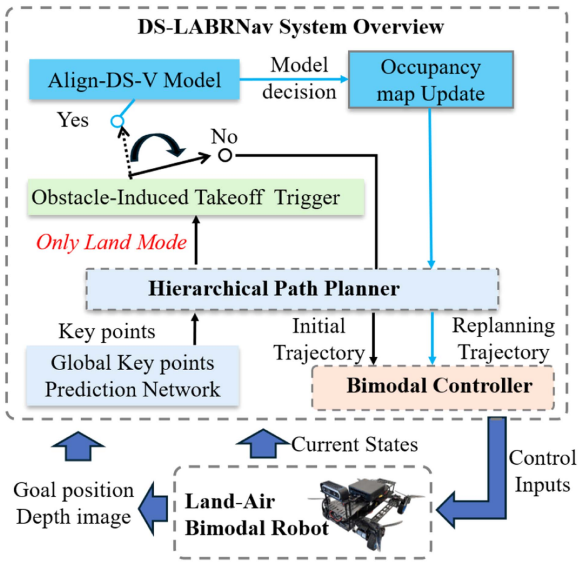

Fig. 4. The overview of DS-LABRNav Framework.

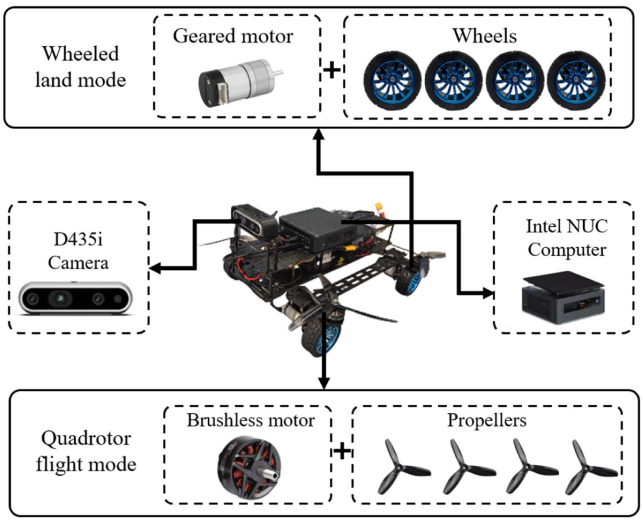

Fig. 5. The structure diagram of the self-developed land-air bimodal robot.

E. System Integration

The integration of the individual modules into a unified navigation pipeline is summarized in Fig. 4. First, the goal position and the current depth image are fed into the Global Keypoints Prediction Network [13], which outputs a set of coarse keypoints covering both land and aerial modes. Next, these keypoints are passed to the Hierarchical Path Planner [7], which generates an initial trajectory. Then, the Obstacle-Induced Takeoff Trigger (OITT) acts as a decision switch that evaluates whether a planned segment requires an aerial maneuver. Crucially, the OITT is designed to trigger VLM queries only when the robot is operating in land mode.

When the trigger is activated, the blue path in Fig. 4 is followed: the OITT invokes the Align-DS-V model to assess obstacle traversability. To handle the several cloud-based inference latency without compromising system stability, the bimodal controller adopts a brake-and-wait state machine. The locomotion controller immediately suspends trajectory tracking and applies active braking to ensure that the chassis remains stationary. Once the VLM's modal decision is received, the occupancy map is updated and trajectory execution resumes. The planner subsequently re-optimizes a ground-preferred trajectory using the updated map whenever feasible. Otherwise, if the trigger is not activated, the black path is followed and the planner executes the original solution without invoking the VLM.

Algorithm 1: Navigation with Traversable Obstacles Base On Vision-Language Model.

Require:Input: Goal position $\mathbf{p}_{t_f}$, robot initial position $\mathbf{p}_{t_0}$, depth image D_{t_0}, occupancy map M_{t_0}, t_0 is the current step.
Output: Optimized trajectory $\tau(t),\ t \in [t_0, t_f]$
Generate keypoints $\mathcal{K} \leftarrow GKPN(D_{t_0}, \mathbf{p}_{t_f})$
Plan trajectory $\tau(t) \leftarrow \text{Planner}(\mathbf{p}_{t_0}, \mathcal{K}, \mathbf{p}_{t_f}, M_{t_0})$
Check OITT trigger condition:
If $e_t == 1$
Mode $\mathcal{A}_{t_0} \in \{0, 1\} \leftarrow \text{VLM}(\text{img}, \text{prompt})$
Update occupancy map:

$$M_{t_0} \leftarrow \begin{cases} M_{\text{free}}, & \text{if } \mathcal{A}_{t_0} = 0 \\ M(\mathbf{v}) + \Delta M_{\text{raycast}}(\mathbf{v}), & \text{otherwise,} \end{cases}$$

Replan $\tau(t) \leftarrow \text{Planner}(\mathbf{p}_{t_0}, \mathcal{K}_t, \mathbf{p}_{t_f}, M_{t_0})$
Else
Accept candidate trajectory $\tau(t)$
End If
Return $\tau(t)$

Finally, the refined trajectory is passed to the bimodal controller for execution. While VLM-guided semantic mapping significantly enhances navigation efficiency, potential reasoning hallucinations (e.g., misclassifying a rigid wall as a soft curtain) inherently introduce safety risks. Because VLM queries are strictly restricted to authorizing ground traversal, any semantic misclassification manifests exclusively as a low-speed ground interaction, fundamentally eliminating the risk of catastrophic high-speed aerial collisions. To provide a hardware-level safety guaranty during the ground traversal of semantically "traversable" regions, the controller continuously monitors chassis motor currents. If either the current threshold is exceeded, the controller immediately issues a stop command to prevent continuous collision. This low-level protective layer operates independent of the high-level semantic module, ensuring robust physical safety across all conditions. The complete pipeline is detailed in Algorithm 1.

V. Experiments

A. Platform Development of Land-Air Bimodal Robot

The land-air bimodal robotic platform presented herein achieves seamless integration of land and air navigation functionalities, as illustrated in Fig. 5. To enhance flight endurance, a carbon fiber composite frame underwent structural optimization to minimize weight while preserving structural integrity, resulting in a total mass of 3.4 kg. The platform is outfitted with critical sensing components: a D435i RGB-D camera for environmental perception and an embedded IMU module facilitating

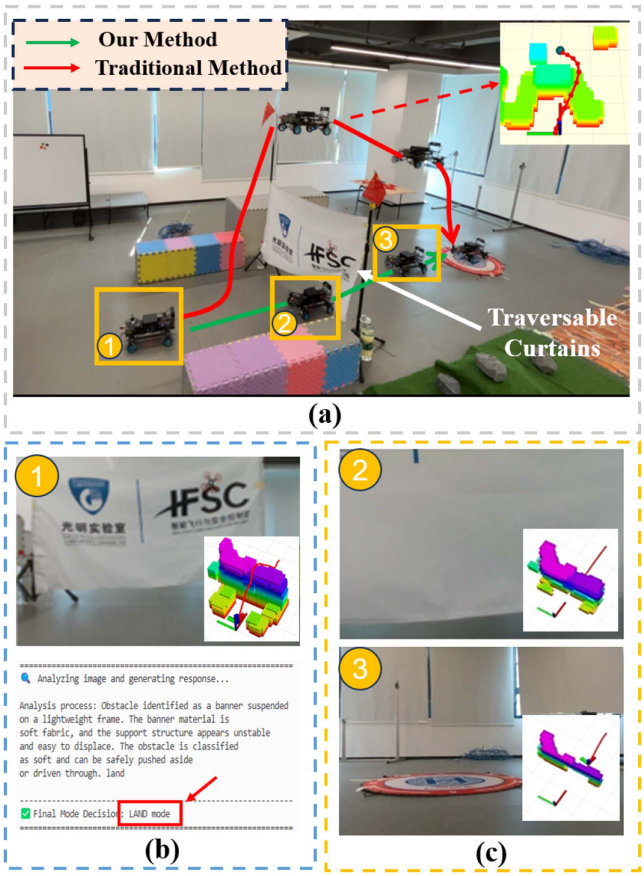

Fig. 6. (a) Navigation trajectories: our method (green) avoids unnecessary aerial transitions, while the traditional method (red) switches to air mode. (b) Top: input image; bottom: VLM output indicating final mode decision. (c) Key execution moments under our method.

real-time attitude estimation. For autonomous navigation capabilities, an NUC-based computing unit was integrated to satisfy the processing demands of navigation tasks. Motion control is executed through a dual-controller architecture, comprising a PIXHAWK-based flight controller and an STM32-based terrestrial locomotion controller, which jointly implement the desired movement strategies. Power is managed via a 19 V 5S lithium battery, which delivers 15 minutes of airborne operation and 1 h of ground runtime.

B. Ablation Study

To quantitatively characterize the individual contributions of the proposed modeling components, we conducted an ablation study. It is worth noting that standard rigid-body robotic simulators (e.g., Gazebo [32] or Isaac Sim [33]) struggle to authentically replicate the complex visual ambiguity and physical deformability of soft materials (such as fabric curtains). Therefore, to ensure the validity of both semantic reasoning and physical interactions, the ablation experiments were conducted exclusively in the real-world indoor scenario depicted in Fig. 6(a).

We compared three configurations: (i) geometric method (w/o VLM), which conservatively treats all obstacles as rigid; (ii) continuous VLM (w/o OITT), updating the occupancy map via cloud queries (e.g., 3 s); and (iii) our framework. The results

TABLE I
ABLATION STUDY IN THE REAL-WORLD INDOOR SCENARIO

Method	Time(s)	Energy (J)
Geometric Method (w/o VLM)	14.2	5797.23
continuous VLM (w/o OITT)	20.4	864.66
DS-LABRNav(ours)	**12.9**	**807.87**

TABLE II
BATTERY AND ENERGY CONSUMPTION PARAMETERS

Parameter	Value
Battery Capacity	5300 mAh
Rated Power	98.1 Wh
Battery Weight	560 g
Operating Voltage	19 V
Land Energy Consumption	$\approx$ 87.7 J/s
Air Energy Consumption	$\approx$ 504.3 J/s

are shown in Table I. Although the geometric method successfully navigated, it consumed maximum energy(5797 J) by forcing aerial maneuvers over the curtain. The continuous VLM correctly identified the traversable curtain, saving locomotion energy, but its lack of an event trigger introduced frequent "decision pauses", increasing total navigation time by over 58%. In contrast, DS-LABRNav achieved an optimal balance: the VLM updates significantly reduced energy consumption by enabling ground traversal, while the OITT mechanism minimized network latency, ensuring real-time navigation.

C. Framework Evaluation in Real-World Environments

We evaluated the proposed navigation framework across multiple scenarios in terms of navigation efficiency, energy consumption, and travel distance under different locomotion modes. Energy consumption was estimated using the measurements summarized in Table II. As the achievable energy savings depend on environmental conditions, we considered two representative real-world scenarios: an indoor environment with visually deceptive soft obstacles and an outdoor environment containing an unknown slope.

a) Soft Obstacle We conducted experiments to evaluate the effectiveness of the proposed navigation strategy in environments containing passable obstacles. As shown in Fig. 6, the setup includes typical low-risk materials such as fabric curtains, which visually resemble rigid barriers but can be traversed by land mode. The goal was deliberately positioned behind such obstacles to test the decision-making ability of the system. Three navigation strategies were compared: (i) a traditional mapping-based planner (TABV) [7], representing state-of-the-art geometric motion planning without global guidance or semantic reasoning; (ii) recent state-of-the-art semantic and occlusion-aware frameworks (AGRNav, HE-Nav), which excel at efficient spatial occupancy prediction and geometric obstacle avoidance; (iii) GKPN-based method [13], which minimizes flight distance but still assumes all obstacles are impassable; and (iv) the proposed DS-LABRNav.

As shown in Fig. 6(a), the proposed method (green) successfully maintains ground navigation by identifying traversable obstacles, while the traditional method (red) switches to aerial mode. The quantitative results are summarized in Table III and further illustrated in the Fig. 8. While AGRNav, HE-Nav,

TABLE III
PERFORMANCE COMPARISON IN SOFT OBSTACLE AND SLOPE

Task	Method	Dist.(Land/Air)	Time(s)	Energy(J)
Soft Obstacle	TABV [7]	0.5m/5.3m	13.8	5203.42
	AGRNav [9]	0.7m/5m	13.5	4713.36
	HE-Nav [10]	1m/4.4m	12.6	4087.77
	GKPN [13]	1.2m/4.2m	12.8	3836.27
	DS-LABRNav	**5.4m/0m**	**12.2**	**787.27**
Unknown Slope	TABV [7]	0m/5.6m	12.2	5392.43
	AGRNav [9]	0.2m/5.4m	11.9	4924.46
	HE-Nav [10]	0.6m/4.8m	**11.3**	4162.87
	GKPN [13]	1.2m/3.8m	12.4	3564.66
	DS-LABRNav	**4.8m/0m**	14.8	**1150.82**

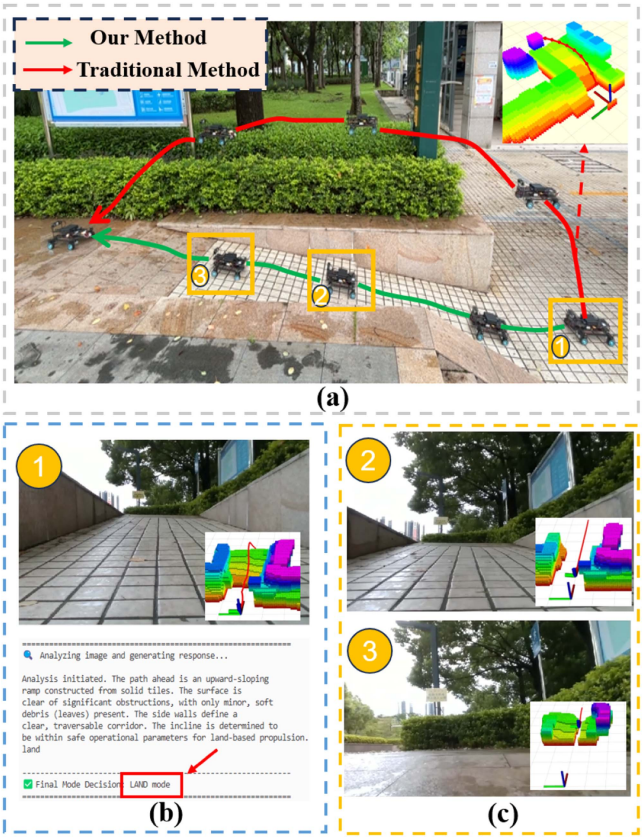

Fig. 7. Experiments in outdoor scenarios with different methods. Three local planning events (1-3) of our method along the path with corresponding first-person images and occupancy maps.

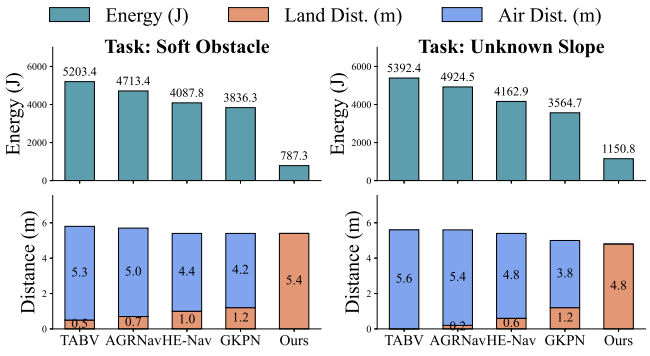

Fig. 8. Quantitative results of indoor and outdoor real environmental.

and GKPN exhibit stable geometric trajectory planning, they fundamentally lack physical property reasoning. Consequently, they inevitably trigger energy-intensive aerial maneuvers (all recorded flight distances exceeded 4 meters). In contrast, our DS-LABRNav operates entirely in land mode, achieving the shortest navigation time (12.2 s) and the lowest energy consumption (787 J).

b) Unknown Slope To further evaluate the capabilities of the proposed system in outdoor environments, we conducted field experiments on unknown urban slopes, as shown in Fig. 7. Before the experiment, there was no prior information about the slope. Similar to the soft obstacle experiment, various navigation methods were compared.

Table III and Fig. 8 summarizes the quantitative results of the outdoor experiments. The traditional TABV and recent AGRNav, HE-Nav achieved extremely fast navigation times, with HE-Nav being the fastest at 11.3 s. However, due to their lack of semantic terrain understanding, they relied almost exclusively on aerial mode to bypass the slope, resulting in high energy consumption (all exceeding 4100 J). The GKPN-based method [13] reduces flight distance by generating aggressive keypoints. However, on sloped ramps these keypoints may lie mid-way along the ramp, where they appear embedded in obstacles in the local map, leading to suboptimal navigation. In contrast, our proposed framework completed all trials successfully while maintaining land traversal throughout the entire ramp. Although the navigation time was slightly longer (14.8 s) due to the VLM inference pause and slower ground locomotion, the energy consumption was drastically reduced to 1150.82 J—saving approximately 70% compared to the SOTA baselines. These results demonstrate that the proposed approach leverages VLM inference and modality-guided map updates to avoid unnecessary takeoffs, enabling energy-efficient navigation.

VI. CONCLUSION

In this work, a navigation framework named DS-LABRNav was presented for land–air bimodal robots. The framework integrates an Obstacle-Induced Takeoff Trigger (OITT) with a VLM-guided occupancy map update to overcome the limitations of conventional methods that conservatively classify all obstacles as non-traversable. VLM inference is triggered only when an aerial maneuver is predicted, minimizing query overhead. The refined occupancy map is subsequently used by the planner to re-optimize trajectories, thereby avoiding unnecessary takeoffs and improving overall energy efficiency.

A limitation of the current framework is its reliance on cloud-based VLM inference, which may introduce network latency and restrict deployment in communication-constrained environments such as underground mines or disaster sites. Although the proposed event-triggered mechanism significantly reduces the frequency of VLM queries, the dependence on cloud connectivity remains a practical constraint in certain real-world scenarios.

Future work will focus on deploying lightweight onboard VLMs to reduce reliance on cloud inference and enable fully offline semantic reasoning. In particular, we plan to integrate compact, edge-compatible VLMs directly on the onboard platform, allowing the system to perform low-latency semantic navigation without external network connectivity. Additionally, the proposed framework will be extended to more complex unstructured environments with dynamic obstacles and stricter real-time constraints.

REFERENCES

[1] D. C. Schedl, I. Kurmi, and O. Bimber, "An autonomous drone for search and rescue in forests using airborne optical sectioning," *Sci. Robot.*, vol. 6, no. 55, 2021, Art. no. eabg1188.

[2] J. Hwangbo et al., "Learning agile and dynamic motor skills for legged robots," *Sci. Robot.*, vol. 4, no. 26, 2019, Art. no. eaau5872.

[3] Z. Liu and L. Cai, "Large-angle and high-speed trajectory tracking control of a quadrotor UAV based on reachability," in *Proc. Int. Conf. Robot. Automat.*, 2022, pp. 1983–1988.

[4] H. Shi, L. Shi, M. Xu, and K.-S. Hwang, "End-to-end navigation strategy with deep reinforcement learning for mobile robots," *IEEE Trans. Ind. Informat.*, vol. 16, no. 4, pp. 2393–2402, Apr. 2020.

[5] Z. Liu and L. Cai, "Simultaneous planning and execution for quadrotors flying through a narrow gap under disturbance," *IEEE Trans. Control Syst. Technol.*, vol. 31, no. 6, pp. 2644–2659, Nov. 2023.

[6] X. Zhou, Z. Wang, H. Ye, C. Xu, and F. Gao, "Ego-Planner: An ESDF-free gradient-based local planner for quadrotors," *IEEE Robot. Automat. Lett.*, vol. 6, no. 2, pp. 478–485, 2020.

[7] R. Zhang, Y. Wu, L. Zhang, C. Xu, and F. Gao, "Autonomous and adaptive navigation for terrestrial-aerial bimodal vehicles," *IEEE Robot. Automat. Lett.*, vol. 7, no. 2, pp. 3008–3015, Apr. 2022.

[8] A. Chandrashekhar, A. Rawate, A. Dhanamathi, R. Dasari, R. P. Shinkar, and P. G. , "Deep reinforcement learning for autonomous drone navigation in cluttered environments," in *Proc. 5th Int. Conf. Recent Trends Comput. Sci. Technol.*, 2024, pp. 41–45.

[9] J. Wang et al., "AGRNav: Efficient and energy-saving autonomous navigation for air-ground robots in occlusion-prone environments," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2024, pp. 11133–11139.

[10] J. Wang et al., "HE-Nav: A high-performance and efficient navigation system for aerial-ground robots in cluttered environments," *IEEE Robot. Automat. Lett.*, vol. 9, no. 11, pp. 10383–10390, Nov. 2024.

[11] J. Wang et al., "OCCRWKV: Rethinking efficient 3D semantic occupancy prediction with linear complexity," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2025, pp. 10915–10921.

[12] J. Wang et al., "OMEGA: Efficient occlusion-aware navigation for air-ground robots in dynamic environments via state space model," *IEEE Robot. Automat. Lett.*, vol. 10, no. 2, pp. 1066–1073, Feb. 2025.

[13] Y. Li et al., "A two-stage lightweight framework for efficient land-air bimodal robot autonomous navigation," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2025, pp. 11165–11171.

[14] J. Ji et al., "Align anything: Training all-modality models to follow instructions with language feedback," 2024, *arXiv:241215838*.

[15] J.-B. Alayrac et al., "Flamingo: A visual language model for few-shot learning," in *Proc. Adv. Neural Inf. Process. Syst.*, 2022, vol. 35, pp. 23716–23736.

[16] G. Team et al., "Gemini: A family of highly capable multimodal models," 2023, *arXiv:2312.11805*.

[17] P. Wang et al., "Qwen2-VL: Enhancing vision-language Model's perception of the world at any resolution," 2024, *arXiv:2409.12191*.

[18] D. Shah et al., "LM-Nav: Robotic navigation with large pre-trained models of language, vision, and action," in *Proc. Conf. Robot Learn.*, 2023, pp. 492–504.

[19] S. Liu, H. Zhang, Y. Qi, P. Wang, Y. Zhang, and Q. Wu, "AerialVLN: Vision-and-language navigation for UAVs," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2023, pp. 15384–15394.

[20] B. Yu, H. Kasaei, and M. Cao, "L3MVN: Leveraging large language models for visual target navigation," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2023, pp. 3554–3560.

[21] M. Khanna et al., "GOAT-Bench: A benchmark for multi-modal lifelong navigation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2024, pp. 16373–16383.

[22] D. D. Fan et al., "Autonomous hybrid ground/aerial mobility in unknown environments," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2019, pp. 3070–3077.

[23] N. Pan, J. Jiang, R. Zhang, C. Xu, and F. Gao, "SkyWalker: A compact and agile air-ground omnidirectional vehicle," *IEEE Robot. Automat. Lett.*, vol. 8, no. 5, pp. 2534–2541, May 2023.

[24] V. Vasilopoulos et al., "Reactive semantic planning in unexplored semantic environments using deep perceptual feedback," *IEEE Robot. Automat. Lett.*, vol. 5, no. 3, pp. 4455–4462, Jul. 2020.

[25] X. Cai, M. Everett, J. Fink, and J. P. How, "Risk-aware off-road navigation via a learned speed distribution map," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2022, pp. 2931–2937.

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

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

[28] C. Huang, O. Mees, A. Zeng, and W. Burgard, "Visual language maps for robot navigation," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2023, pp. 10608–10615.

[29] X. Zhao, W. Cai, L. Tang, and T. Wang, "ImagineNav: Prompting vision-language models as embodied navigator through scene imagination," in *Proc. 13th Int. Conf. Learn. Representations*, 2025, pp. 94387–94401.

[30] M. Elnoor et al., "Robot navigation using physically grounded vision-language models in outdoor environments," 2024, *arXiv:2409.20445*.

[31] DeepSeek-AI, "DeepSeek-R1: Incentivizing reasoning capability in LLMs via reinforcement learning," 2025, *arXiv:2501.12948*.

[32] N. Koenig and A. Howard, "Design and use paradigms for gazebo, an open-source multi-robot simulator," in *Proc. 2004 IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2004, vol. 3, pp. 2149–2154.

[33] V. Makoviychuk et al., "ISAAC gym: High performance GPU based physics simulation for robot learning," 2021, *arXiv:2108.10470*.