TOP-Nav: Legged Navigation Integrating Terrain, Obstacle and Proprioception Estimation

Junli Ren*, Yikai Liu*, Yingru Dai, Guijin Wang

Abstract—**Legged navigation is typically examined within open-world, off-road, and challenging environments. In these scenarios, estimating external disturbances requires a complex synthesis of multi-modal information. This underlines a major limitation in existing works that primarily focus on avoiding obstacles. In this work, we propose TOP-Nav, a novel legged navigation framework that integrates a comprehensive path planner with Terrain awareness, Obstacle avoidance and close-loop Proprioception. TOP-Nav underscores the synergies between vision and proprioception in both path and motion planning. Within the path planner, we present and integrate a terrain estimator that enables the robot to select waypoints on terrains with higher traversability while effectively avoiding obstacles. In the motion planning level, we not only implement a locomotion controller to track the navigation commands, but also construct a proprioception advisor to provide motion evaluations for the path planner. Based on the close-loop motion feedback, we make online corrections for the vision-based terrain and obstacle estimations. Consequently, TOP-Nav achieves open-world navigation that the robot can handle terrains or disturbances beyond the distribution of prior knowledge and overcomes constraints imposed by visual conditions. Building upon extensive experiments conducted in both simulation and real-world environments, TOP-Nav demonstrates superior performance in open-world navigation compared to existing methods.**

I. INTRODUCTION

Imagine an outdoor hiking scenario where gait stability and efficient navigation are both critical. The target environment is often marked by intricate obstacles and hazardous terrains that cannot be fully observed through onboard vision. This challenge underscores the need for a comprehensive path planner integrating multi-modal observations. To achieve this, we humans employ experienced guidance and alternative perception modalities such as trekking poles and GPS devices. This example reveals a requirement of numerous trials and a substantial foundation of prior knowledge to accomplish the challenge navigation task.

Although recent advancements in legged locomotion have allowed the robots to navigate various terrains based on a simulation-learned strong controller [5, 20, 21, 39, 41], the complexity of currently insimulable real-world factors makes it impossible for the robot to traverse all potential terrains encountered in reality. As a result, integrating the locomotion controller with only an open-loop path planner often restricts legged navigation to limited scenarios [4, 18, 23, 33].

* Equal Contribution.
Corresponding Author: Guijin Wang (wangguijin@tsinghua.edu.cn)
All authors are with Department of Electronic Engineering, Tsinghua University, Beijing 100084, China.

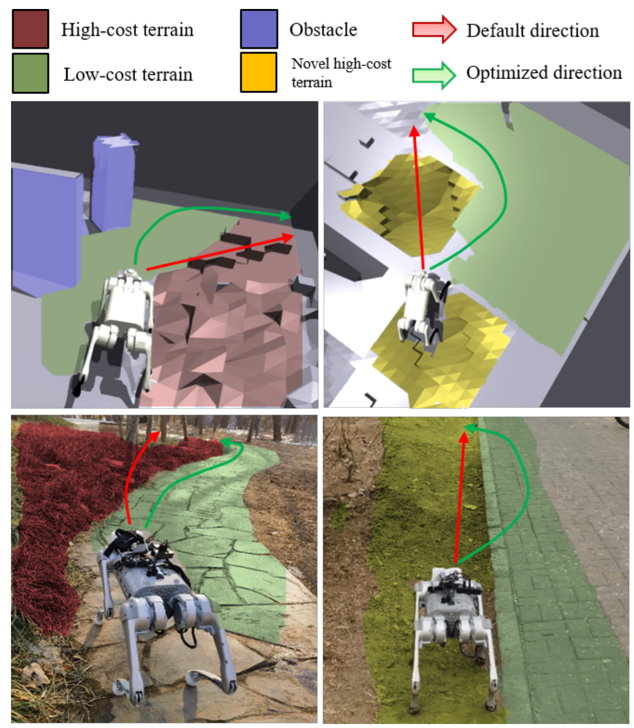

Fig. 1. Example deployment scenarios for **TOP-Nav** in both simulation and the real world. Besides obstacle avoidance, the robot plans an optimized direction on terrains with better traversability. For novel terrains, the robot incorporates proprioception history from previously traversed terrain to infer the traversability.

An effective solution to overcome these limitations is equipping the robot with terrain awareness. A traversable path can be planned based on the robot's preferences on terrains [9, 15] with an appropriate distribution of contact heights and forces [8]. Unlike obstacle estimation, the distinct terrain features are typically encoded as semantic information, where traditional methods collect sufficient data and train segmentation or classification models [16] to learn these features. Nevertheless, compiling an exhaustive catalogue of all conceivable terrains along with their corresponding walking preferences is impractical [11]. Compounding the issue, the dynamic real-world conditions, such as lighting, humidity, and temperature, may introduce inaccuracies in the correspondence between images and walking preferences, especially when relying exclusively on vision in this context [38].

The mentioned challenges can be attributed to the reliance on vision and the ignorance of motion states in path planning. To address this, we complement the vision-only terrain estimator with online corrections from motion evaluations. We construct a proprioception advisor to not only convey information about the traversability cost of novel terrains but also alert the robot to unexpected disturbances, such as invisible obstacles.

By integrating the **T**errain estimator, **O**bstacle estimator, and **P**roprioception advisor, we formulate **TOP-Nav**: a hierarchical path and motion planning framework designed to navigate a quadruped robot through diverse and challenging terrains proficiently. **TOP-Nav** maintains four real-time robot-centric costmaps corresponding to: 1) goal approaching; 2) terrain traversibility; 3) obstacle occupancy; and 4) proprioception advice. The costmaps are synthesized with dynamically weighted factors to ensure a balanced consideration of safety and efficiency. We evaluate **TOP-Nav** both in simulation and on a physical robot, with a comparative analysis against existing approaches that address subsets of the factors (terrain, obstacle, proprioception). In simulation, we construct an environment featuring diverse terrains, including slopes, steps, and random textures, along with various obstacles. For real-world experiments, we deploy our approach in diverse off-road navigation scenarios encompassing common field terrains. Our main contributions are summarized below:

- A comprehensive legged navigation framework Integrating multi-modal observations throughout a task and motion planner;
- A terrain estimator trained from previously collected data to inform the robot of a vision-based terrain traversability;
- Compensating proprioception to offer online corrections for the vision-based estimation of both terrain traversability and obstacles;
- Successful implementation and quantitative validations of the proposed framework in both simulation and hardware.

II. RELATED WORK

The pivotal feature of **TOP-Nav** lies in its integration of the terrain estimator and proprioception advisor. To elucidate this aspect, we present an overview of the relevant literature.

A. Vision and Legged Proprioception Integration

Recent advancements in legged locomotion have showcased a synergistic mechanism for processing vision and proprioception within the context of “How to Walk” [2]. While heightmaps serve as crucial observations for a controller to generate dynamic motions across various terrains [26, 27], proprioception observations can be employed to reconstruct the heightmaps in visually degraded environments [30]. Different from the mentioned efforts focus on the depth channel, we propose a data-efficient solution to extract semantic information from the proprioceptive feedback.

The insights of integrating vision and legged proprioception are further explored in guiding the robot in “Where to Walk”. From the perspective of vision adied navigation, researchers have explored both hierarchical [4, 23] and end-to-end [32] pipelines. These methods require substantial efforts to develop a robust perception module [19], making it difficult and costly to transfer them to different hardwares.

To provide a comprehensive task observation and reduce the reliance on vision systems, recent researches have introduced proprioception to improve task planning. A majority of these works learn proprioception representations along with visual features in simulation, and then implement the cross-modal features through end-to-end [37], hybrid [17] or decoupled methods [40]. Despite the effectiveness demonstrated in these works, the high-dimensional representation space presents challenges for adaptation to novel scenarios and sim-to-real transfer. Alternatively, Fu et al. [14] introduced a hierarchical navigation framework that derives evaluation scores from motion states, yet it overlooks the integration of visual observations.

To mitigate these limitations, we propose a novel approach within **TOP-Nav** by maintaining a series of lightweight cost maps derived from multi-modal observations. This integration achieves a dynamic balance between vision and proprioception. Furthermore, we leverage the learning-based locomotion controller to derive motion evaluations from the value function, offering an efficient solution without additional training.

B. Terrain Traversability Estimation

Terrain traversability is determined by factors such as terrain geometry, texture, and physical properties [12]. Estimating these features could be achieved by identifying the semantic class with a predefined static traversability score[9, 16, 31, 34]. These solutions exhibit a notable dependency on large-scale datasets [29] or limited to structured environments like urban scenarios [1, 6].

In off-road navigation, the motion states involved in the dynamic interactions between the robot and the environment provide valuable metrics for assessing terrain traversability [10]. These insights have inspired methods that eliminate the need for manual annotation by autonomously deriving terrain traversability from proprioception through self-supervised learning [3, 7, 22, 24, 28, 38]. Nevertheless, the performance of these studies is contingent upon the quality of the collected datasets [12]. To emphasize the challenges in unconstrained navigation, researchers have proposed various approaches to handling novel observations. For instance, Frey et al. [11] updated the traversability estimation network online with anomalies into consideration. Karnan et al. [25] performs nearest-neighbor search in the proprioception space to align visually novel terrains with existing traversability.

Drawing inspiration from those works estimating traversability for novel terrains, we propose a prior-knowledge informed terrain estimator that employs the proprioception advisor as online corrections. Our method diverges from previous approaches primarily in two key aspects: 1) We employ an estimated value function from reinforcement learning to assess terrain traversability, providing a comprehensive evaluation of robot-terrain interactions. 2) By incorporating the proprioception based terrain traversability estimation as online corrections to vision-based predictions, we achieve a more data-efficient approach for estimating traversability on unknown terrains compared to online training.

III. Background

The proposed proprioception advisor leverages recent progress in legged locomotion to provide online motion evaluations for the path planner. This section provides an essential background of this.

Learning-based legged locomotion controllers has been well developed through reinforcement learning, which is generally achieved by updating the policy π within the asymmetric actor-critic training:

$$\pi_1 \begin{cases} \boldsymbol{a}_t = \pi_{\text{actor}}(\boldsymbol{o}_t^p, \boldsymbol{o}_t^i, \boldsymbol{o}_t^e, \boldsymbol{o}_t^h) \\ c_t = \pi_{\text{critic}}(\boldsymbol{o}_t^p, \boldsymbol{o}_t^i, \boldsymbol{o}_t^e, \boldsymbol{o}_t^h) \end{cases}, \tag{1}$$

the locomotion policy receives privileged observation $\boldsymbol{o}_t^p$, proprioception observation $\boldsymbol{o}_t^i$, scanned dots external observation $\boldsymbol{o}_t^e$ and historical observation $\boldsymbol{o}_t^h$ respectively. π_{actor} is modelled as a Gaussian policy and infers the optimized actions $\boldsymbol{a}_t$ to compute the joint positions $\boldsymbol{q}_{\text{des}}$. c_t stands for the estimated value function from π_{critic}, which is updated through:

$$L_t^{\text{critic}} = (c_t - c_t^{\text{targ}})^2, \tag{2}$$

where

$$c_t^{\text{targ}} = \sum_{i=t}^{T} \gamma^{i-t} R(s_i), \tag{3}$$

s_i denotes the robot state, $R(s_i)$ represents the rewards accrued at timestep i, which commonly includes guiding the robot to track a given velocity commands with stable gaits and attitude. A substantial efforts in reward engineering has enabled the value function c_t^{targ} to evaluate a comprehensive set of interactions between the robot and its environment. Therefore, when deploying the learning-based controller, c_t can be utilized to provide closed-loop feedback for the task planner.

The complete training paradigm will involve a second stage, training a depth encoder and a student network to reproducing $\boldsymbol{o}_t^p$ and $\boldsymbol{o}_t^e$ from real world accessible observations $\boldsymbol{I}_d$ (depth image), $\boldsymbol{o}_t^i$ and $\boldsymbol{o}_t^h$. The motion controller will track the planned Δ_{yaw} and v_{lin} and the control action is represented by the 12 desired joint position $\boldsymbol{q}_{\text{des}}$.

IV. Method

A. System Overview

As illustrated in Fig. 2, the proposed **TOP-Nav** pipeline connects a path and motion planner to tackle the task of legged navigation. To efficiently generate a collision-free path and stable motion states, we highlight a terrain estimator and a proprioception advisor inside the pipeline. In addition to combining these modules on the costmap (Fig. 2), the proprioception advisor provides online corrections for the terrain estimation. Such integration informs the robot of the traversability of unknown terrains and invisible obstacles, allowing **TOP-Nav** to handle open-world task without being constrained by visual conditions or prior knowledge.

To elucidate the task configuration, the robot is given a point goal with its location $\boldsymbol{p}_{\text{goal}}$ and has to approach the goal within a limited time. The location of the robot is denoted as $(\boldsymbol{p}_{\text{base}}, r_{\text{base}})$. For external observation, we utilize a bio-channel perception module for both the task and motion planner. This module includes depth ($\boldsymbol{I}_d$) and RGB ($\boldsymbol{I}_{\text{rgb}}$) channels. The path planner computes the desired velocity command $(v_{\text{lin}}, \Delta_{\text{yaw}})$ for the robot, which is tracked by the motion planner along with the appropraite joint position signal.

B. Integrated Path Planner

A key feature within the path planner of **TOP-Nav** is the integrated cost map $\boldsymbol{M}_C$ which offers a comprehensive estimation that encompasses terrain traversability $\boldsymbol{M}_T$, obstacles $\boldsymbol{M}_O$, proprioception advice $\boldsymbol{M}_P$ and the goal approaching map $\boldsymbol{M}_G$. This integration addresses both visible and unexpected external disturbances against the robot.

To keep a balance between navigation efficiency and safety, we formulate the combination of $\boldsymbol{M}_C$ with dynamic scaling factors as follows:

$$\boldsymbol{M}_C = \boldsymbol{M}_P + \boldsymbol{M}_O + \alpha_T \boldsymbol{M}_T + \alpha_G \boldsymbol{M}_G, \tag{4}$$

$$\alpha_T = \frac{k_{T_1}}{1 + e^{-k_{T_2}(d-d_0)}}, \ \alpha_G = \frac{k_{G_1}}{1 + e^{-k_{G_2}(t-t_0)}} + k_{G_0}, \tag{5}$$

here $\boldsymbol{M}_i$ are 2-d matrix with a spatial resolution of $0.15m$, d_0, t_0, k_T, k_G are hyperparameters, t denotes current time consuming and d denotes the distance from the point goal at the current step. We accord the highest priority to $\boldsymbol{M}_P$ and $\boldsymbol{M}_O$ since they represent non-traversable locations where the robot cannot pass through. The terrain traversability scale α_T decreases as the robot approaches the target. This design is made considering that, as the robot nears the target (d decreases), taking a detour to avoid a challenging yet traversable terrain would be an inefficient behaviour. Conversely, the goal approaching scale α_G increases with the duration t of the task.

With the integrated map $\boldsymbol{M}_C$, we select the optimal waypoint $\boldsymbol{p}_{\text{way}}$ based on the lowest combined cost.

$$\boldsymbol{p}_{\text{way}} = \underset{\boldsymbol{p} \in \boldsymbol{E}}{\operatorname{argmin}} \frac{1}{\|\boldsymbol{p}_{\text{base}} - \boldsymbol{p}\|} \sum_{(x,y)=\boldsymbol{p}_{\text{base}}}^{(x,y)=\boldsymbol{p}} \boldsymbol{M}_C(x,y), \tag{6}$$

Here $\boldsymbol{E}$ denotes the set of points on the edge of $\boldsymbol{M}_C$. For each point $\boldsymbol{p}$ in $\boldsymbol{E}$, we calculate the average combined cost along the path from the robot location $\boldsymbol{p}_{\text{base}}$ to the edge point $\boldsymbol{p}$. The optimal waypoint $\boldsymbol{p}_{\text{way}}$ is then chosen as the point with the lowest path cost.

After determining the optimized local target $\boldsymbol{p}_{\text{way}}$, the target direction is calculated based on the relative position $\Delta_{\text{yaw}} = \arctan(\boldsymbol{p}_{\text{way}} - \boldsymbol{p}_{\text{base}}) - r_{\text{base}}$, here r_{base} denotes the yaw direction of the robot base. The linear velocity is constrained to mitigate the impact of high angular variation with $v_{\text{lin}} = v_0 e^{-k\Delta_{\text{yaw}}}$, thereby preventing abrupt and substantial turning at high speeds.

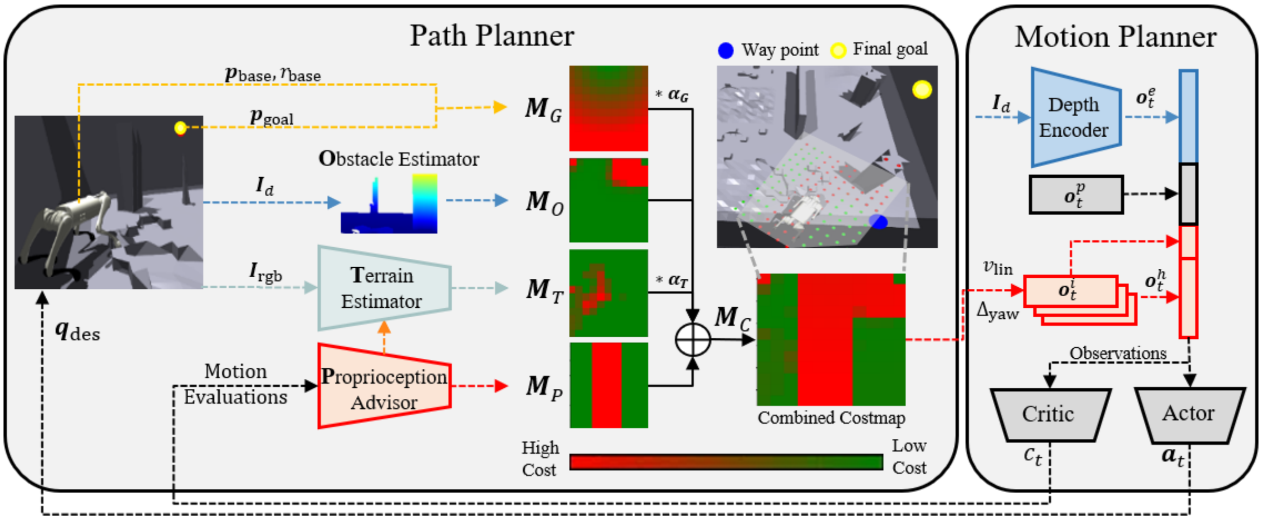

Fig. 2. **TOP-Nav**: the hierarchical path and motion planning framework to tackle the problem of legged navigation with a point goal. The path planner synthesizes relative distance, terrain traversability, obstacle occupancy, and motion evaluations into a combined cost map, from which it computes waypoints based on the overall cost considerations. The motion evaluations are extracted from a learning-based motion planner, which receives depth observation as input.

Obstacle Estimation and Localization: In simulation, the robot has access to the ground truth location. For each point $\boldsymbol{p} = (x, y)$, $\boldsymbol{M}_G(x, y) = \|\boldsymbol{p}_{\text{goal}} - \boldsymbol{p}\|$. In real world, we set a target direction r_{goal} to compute the goal map $\boldsymbol{M}_G(x, y) = \|r_{\text{goal}} - \arctan(\boldsymbol{p} - \boldsymbol{p}_{\text{base}})\|$. We construct the obstacle estimation based on the depth channel $\boldsymbol{I}_d$. The perceived obstacles are converted into point clouds, and for each point in $\boldsymbol{M}_O$, we compute the signed distance $\boldsymbol{M}_{\text{SDF}}$ to the closest obstacles within distance $d_{\max}$, the cost of obstacles therefore can be computed as:

$$\boldsymbol{M}_O(x, y) = \frac{\max(0, d_{\max} - \boldsymbol{M}_{\text{SDF}}(x, y))}{d_{\max}}. \tag{7}$$

C. Proprioception Advisor

We design the proprioception advisor to identify motion abnormalities that may arise from unexpected external disturbances. A straightforward approach is to use the velocity tracking error [11], but its effectiveness can be hindered by the noisy velocity estimation on real hardware. Considering the interactions between the robot and its environment, the ideal metrics should be both observable and comprehensive. To address this, we utilize the value function c_t estimated from π_{critic}, which synthesizes multiple metrics and provides more stable results by incorporating historical observations.

The original π_{critic} requires privileged observations, we train another value function estimation network using observable motion states for the proprioception advisor to be deployed on hardware. The estimated motion evaluation c_t is normalized using a sigmoid function. As demonstarted in Fig. 3, the motion evaluation will decline sharply when the robot encounters a locomotion failure, potentially due to transitioning onto challenging terrain. Implementation details for the reward design and normalization are provided in the appendix.

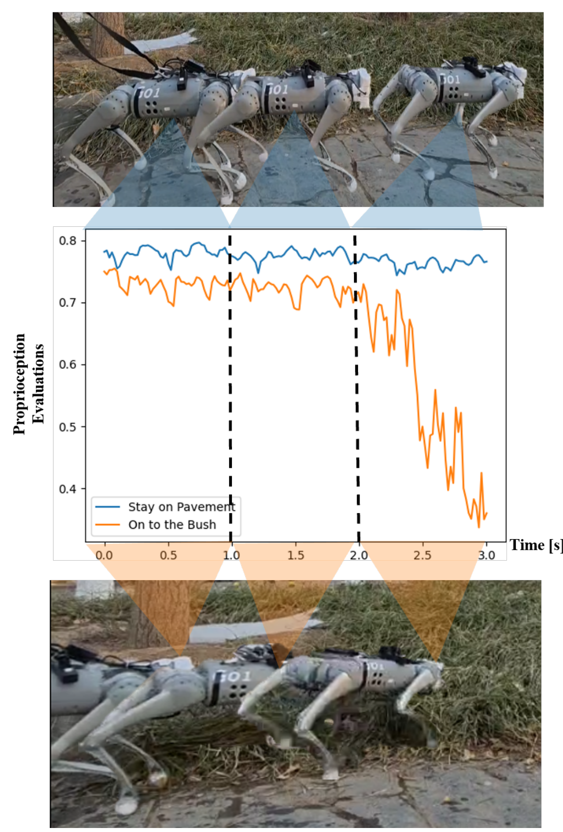

Fig. 3. We demonstrate the variation of the normalized c_t when the robot traverses on different types of terrain. The traversal on bush reveals unstable postures and a failure to track velocity commands, resulting in a decrease in c_t. Conversely, navigating on pavement exhibits a stable c_t.

Within the path planner, we first directly utilize the motion evaluations to enhance the robot's awareness of unforeseen disturbances. This involves estimating collisions ($\boldsymbol{M}_P$) beyond the obstacle estimator and integrating them into the combined cost map ($\boldsymbol{M}_C$). Previous works generate fix-sized virtual obstalces when proprioception feedback reaches a certain threshold [14], the design overlooks the variation of motion evaluations within each interval. We propose an continuously varied proprioception cost along the y-axis in $\boldsymbol{M}_P$.

$$\boldsymbol{M}_P(x,y) = \frac{1-\mathrm{Norm}(c_t)}{e^{k_P(\|y_{\mathrm{base}}-y\|)}}, \tag{8}$$

here k_P is hyperparameter. Since a lower motion evaluation indicates a potential challenging terrain or an invisible obstacle in the current direction of the robot, we allocate c_t to the centriod column in $\boldsymbol{M}_P$ and decrease it towards the edges.

D. Vision-based Terrain Estimation

The proposed terrain estimator incorporates terrain traversability into the path planner. Firstly, we construct a mapping from visual observations to reference terrain traversabilities using prior knowledge.

Given that terrains like stairs, slopes, or discrete small steps can be effectively identified from depth observations, to which our locomotion controller can respond appropriately, our task-level terrain estimation module focuses on semantic features such as color and texture. These features are highlighted by the RGB channel, specifically pertaining to terrains such as grass, snow, and cement. To correspond the RGB observation into real-world coordinates, we apply the perspective transformation on $\boldsymbol{I}_{\mathrm{rgb}}$ to map the pixels to $\boldsymbol{M}_{\mathrm{BEV}}$, which has corresponding coordinates as $\boldsymbol{M}_T$. In subsequent processing, we discretize $\boldsymbol{M}_{\mathrm{BEV}}$ into patches (Fig. 4) and assign the same difficulty within each patch.

To accomplish vision-based terrain estimation, the robot first identifies the observed terrain from $\boldsymbol{M}_{\mathrm{BEV}}$. We achieve this by training a terrain classification network π_{terrain} from offline collected data $\boldsymbol{D}_{\mathrm{terrain}}$. Considering that such inference falls short when faced with unfamiliar terrains, we calculate the predictive entropy to approximate the uncertainty U of the inference at current observation [35]. Since the approximation requires Monte Carlo simulation, we perform K stochastic forward inferences with different data augmentations at each timestep, with P_i^k denoting the predicted probability for label i at trail k. We estimate the expectation as the average of the predicted probability $P_i = \sum_k P_i^k/K$. Then the predicted explicit terrain names T and corresponding uncertainty U for the current patch can be calculated as:

$$U = -\sum_i P_i \log P_i \text{ and } T = \operatorname*{argmax}_i(P_i). \tag{9}$$

For safe navigation, terrain traversability signifies the robot's preference or stability in traversing a particular terrain, which has been explored from either self-supervision [11] or human demonstrations [24]. We leverage experience from these efforts to establish the terrain traversability (Fig. 4) as the operator preference. As we demonstrated in Fig. 3, such prior knowledge-based terrain traversability aligns with the proprioception evaluations.

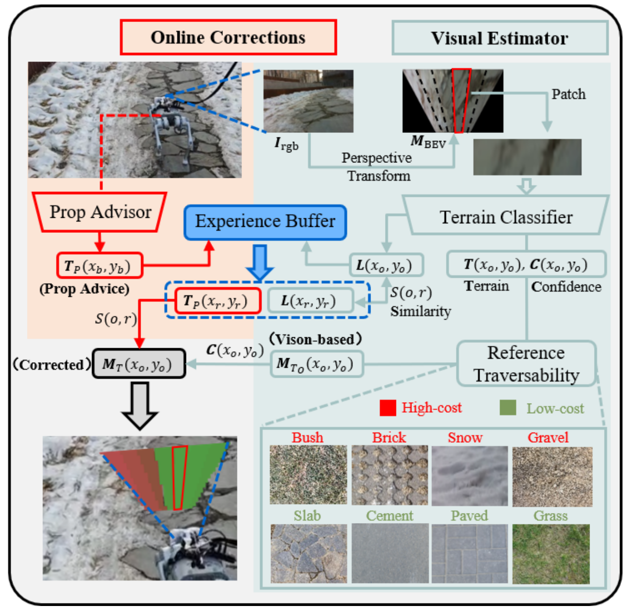

Fig. 4. The proposed terrain estimator incorporates a visual estimator and online corrections. The visual estimator infers the terrain name T from the BEV map with its uncertainty U, and the corresponding terrain traversability $\boldsymbol{M}_{T_O}$ for T is interpreted based on prior knowledge. The online corrections involve recording the motion evaluations $\boldsymbol{M}_{T_P}$ by the proprioception advisor and retrieving them from the experience buffer when encountering similar observations. In the figure, (x_b, y_b), (x_o, y_o), and (x_r, y_r) represent the locations of the robot base, observation point, and historical position, respectively.

E. Online Terrain traversability Corrections

Through the seamless integration of vision and proprioception, we provide online corrections for the vision-based traversability estimation $\boldsymbol{M}_{T_O}$ without additional training (Fig. 4). This enables the robot with terrain awareness beyond the limitations of the collected data $\boldsymbol{D}_{\mathrm{terrain}}$.

We address the mechanism by which the robot recalls the traversability of a terrain once it has been seen and traversed within the navigation process. With the robot location $(x_{\mathrm{base}}, y_{\mathrm{base}})$, we record a duration of $1s$ proprioception advice to indicate the terrain traversability at the robot location:

$$\boldsymbol{T}_P(x_{\mathrm{base}}, y_{\mathrm{base}}) = 1 - \mathrm{Norm}(\overline{c_{t-k:t}}). \tag{10}$$

Besides $\boldsymbol{T}_P(x_{\mathrm{base}}, y_{\mathrm{base}})$, we can access a latent feature $\boldsymbol{L}(x_{\mathrm{base}}, y_{\mathrm{base}})$ extracted by the classifier π_{terrain} when the same patch was observed a few steps ago, we record both $\boldsymbol{T}_P$ and $\boldsymbol{L}$ at location $(x_{\mathrm{base}}, y_{\mathrm{base}})$ into an experienced list $\boldsymbol{P}_e$. Now given a new observation at location (x_o, y_o), We can find a traversed and seen patch (x_r, y_r) that looks closest to (x_o, y_o)

based on cosine similarity:

$$S_{(o,r)} = \underset{r \in \boldsymbol{P}_e}{\operatorname{argmax}} \cos\left(\boldsymbol{L}(x_r, y_r), \boldsymbol{L}(x_o, y_o)\right), \quad (11)$$

Note that we also have the historical traversability $\boldsymbol{T}_P(x_r, y_r)$ infered by the proprioception advisor at (x_r, y_r), we can compute a proprioception adapted traversability with the similarity $S_{(o,r)}$ to access the historical proprioception correction $\boldsymbol{M}_{T_P}(x_o, y_o)$:

$$\boldsymbol{M}_{T_P}(x_o, y_o) = \frac{\boldsymbol{T}_P(x_r, y_r)}{1 + e^{-k_{T_3}(S_{(o,r)} - S_0)}}, \quad (12)$$

here k_{T_3} and S_0 are hyperparameters.

Due to the delay in estimation from the proprioception history, we intend for $\boldsymbol{M}_{T_P}$ to be utilized when encountering novel terrains where $\boldsymbol{M}_{T_O}$ becomes unreliable. Therefore, we normalize the visual uncertainty $\boldsymbol{U}(x_o, y_o)$ into confidence $\boldsymbol{C}(x_o, y_o)$ to adjust the contribution of $\boldsymbol{M}_{T_O}$ and $\boldsymbol{M}_{T_P}$:

$$\boldsymbol{M}_T(x_o, y_o) = \left(\frac{\boldsymbol{M}_{T_O} - \boldsymbol{M}_{T_P}}{1 + e^{-k_{T_4}(C - C_0)}} + \boldsymbol{M}_{T_P}\right)(x_o, y_o), \quad (13)$$

here k_{T_4} and C_0 are hyperparameters.

By integrating the adapted $\boldsymbol{M}_T$ as the terrain traversability into the path planner, our system demonstrates rapid adaptability to different terrain and visual conditions.

V. Evaluations

A. Experimental Setup

We assess **TOP-Nav** in both simulation and the real world. Considering the sim2real gap primarily lies in the RGB semantic information, we assume the robot has access to ground-truth terrain traversability in simulation. Therefore, our simulation experiments were designed to emphasize the following evaluations:

- The improvements of **T**errain awareness in navigating challenge environments.
- The improvements of the proposed **P**roprioception advisor in navigating the challenge environments.
- The improvements in locomotion by selecting waypoints with higher traversability.

Relatively, in addition to the evaluations mentioned above, our real-world experiments specifically focus on assessing the enhancements provided by the proposed terrain estimator.

Simulation Settings: Our simulation experiments are conducted within Nvidia Isaac Gym. We create a grid of 8×8 independent navigation cells. Each cell is $5m \times 5m$ in size, featuring a robot assigned to a point goal navigation task. The robot and point goal is randomly generated, with a minimum initial distance of $5m$ within the cell. $\boldsymbol{M}_C$ is configured with dimensions $(1.2m, 1.5m)$ centered around the robot's location at $\boldsymbol{p}_{\text{base}}^{\text{map}} = (0.45m, 0.75m)$. This design prioritizes a light-weight, real-time updated path planner and maintains consistency with the height map used in locomotion. The environments include randomly generated obstacles and terrains. Each cell is equipped with at least one *wall* obstacle, two *column* obstacles and the remaining space is divided into $1m \times 1m$ sections. For terrain traversability assignments, we uniformly partition the obstacle-free space into difficulty levels [0, 0.25, 0.5, 0.75, 1] and generate terrain with various heights of steps and the intensity of irregular terrain corresponding to the difficulty levels. This difficulty serves as ground-truth walking preferences in simulation. The simulation experiments are conducted 25 times in each of the 64 navigation cells.

Real-World Settings: For real-world evaluations, we conduct outdoor navigation tasks in 5 different scenarios involving challenge obstacles and various terrains, with each scenario replicated 5 times for each method under investigation. To obtain a more comprehensive observation in the real world, we configure the cost map $\boldsymbol{M}_C$ with dimensions of $(3m, 3m)$. The location of robot is at $\boldsymbol{p}_{\text{base}}^{\text{map}} = (0m, 1.5m)$ in real world $\boldsymbol{M}_C$. The criteria for success include reaching the goal within the specified time constraints, consistent with the simulation setting.

Metrics: We evaluate **TOP-Nav** with the following metrics:

- *SR* (Success Rate): The percentage of successful experiments. We define a success experiment as approaching the point goal into $0.5m$ within 20 seconds
- *TD* (Terrain Difficulty): The percentage of average traversed terrain costs, *TD* provides the ground truth traversed terrain costs within each episode.
- *UT* (Unstable Time): The percentage of unstable motion states ($|roll| > 0.15$ or $|pitch| > 0.15$) in each episode.
- *VFT* (Velocity Tracking Failure): The average percentage of velocity tracking failures ($\|v_{\text{lin}} - v_{\text{act}}\| > 0.2$) in each episode.
- *AEC* (Average Energy Consumption): The average energy consumption ($\tau \dot{q}$) [13] in each successful episode.

Among them, *UT*, *VFT* and *AEC* demonstrate walking states while completing navigation tasks in challenging terrains., *TD* provides a reference terrain traversable cost, measuring the improvements in terrain-aware path planning for gait stability. We calculate the variance across different navigation scenarios to assess the robustness of the proposed method.

Baselines: Beyond ablation study, we compare **TOP-Nav** against state-of-the-art legged navigation frameworks and segmentation based terrain estimators. VP-Nav [14] integrates vision and proprioception to develop a collision detector and a fall predictor within the navigation pipeline. GA-Nav [16] achieves terrain segmentation relying solely on vision.Sterling [24] learns terrain traversability in a self-supervised manner by assigning traversability to terrains based on those that share similar proprioception representations. However, this approach requires prior training for encountered terrains.

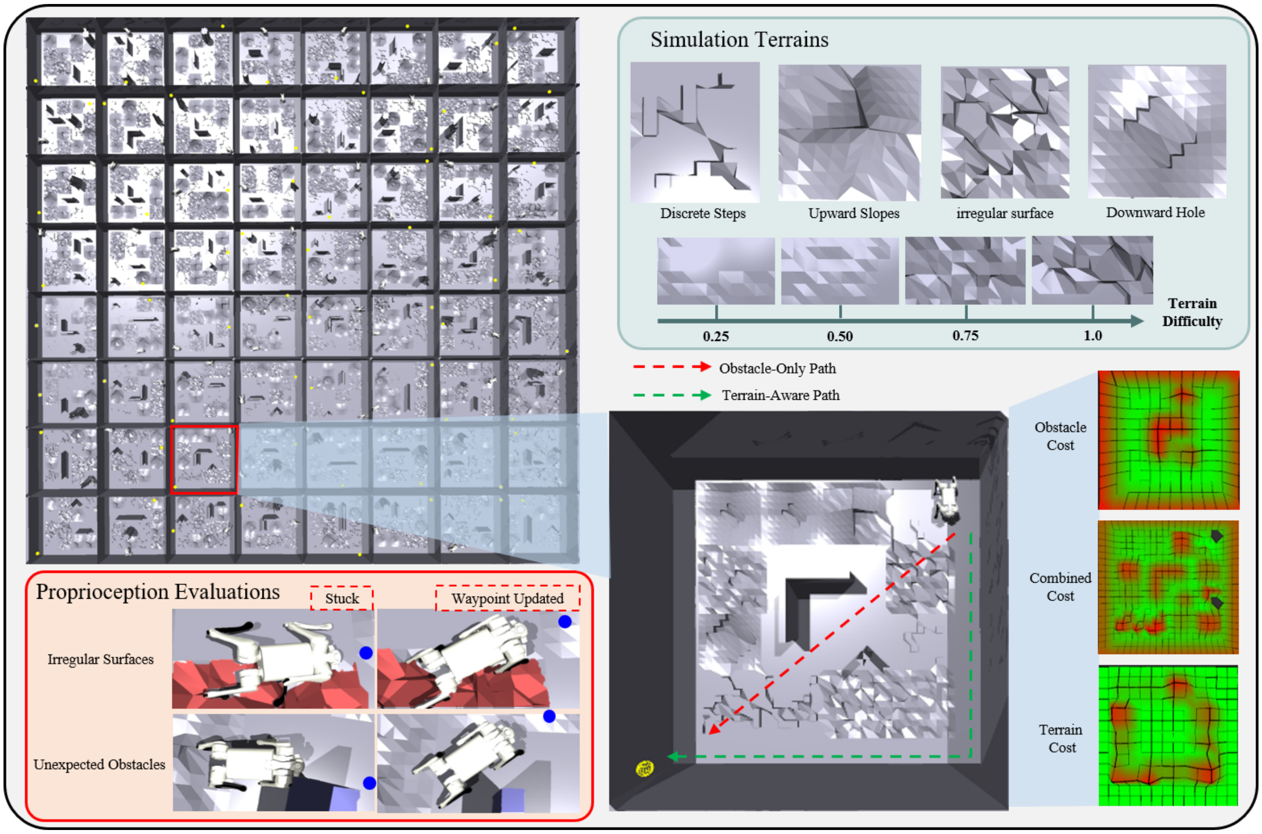

Fig. 5. We establish a parallel navigation evaluation environment in simulation, featuring 8×8 independent navigation cells. Each cell consists of randomly generated challenging terrains with distinct traverse difficulty, which is marked by the irregularity and complexity of the terrain. The proposed terrain awareness navigation framework plans an optimal path to navigate challenging terrains. The robot demonstrates the capability to recover from unexpected obstacles or irregular terrains with the proprioception advisor.

TABLE I
SIMULATION RESULTS WITH COMPARISON EXPERIMENTS AND ABLATION STUDY

Method	*SR* (%) ↑	*TD*(%) ↓	*UT*(%) ↓	*VFT*(%) ↓	*AEC* ↓
VP-Nav	65.62 ± 2.38	47.18 ± 0.79	36.18 ± 0.54	26.09 ± 0.42	101.80 ± 49.61
wo/Terrain	68.00 ± 2.57	45.83 ± 0.70	34.92 ± 0.59	25.06 ± 0.46	101.70 ± 46.55
wo/Proprioception	62.75 ± 1.64	$\mathbf{20.47 \pm 0.17}$	32.87 ± 0.48	26.66 ± 0.57	$\mathbf{88.17 \pm 15.31}$
Obstacle-Only	52.81 ± 2.30	47.79 ± 0.82	41.45 ± 0.79	39.13 ± 0.99	96.82 ± 48.03
TOP-Nav	$\mathbf{73.62 \pm 1.21}$	22.65 ± 0.19	$\mathbf{27.21 \pm 0.24}$	$\mathbf{18.14 \pm 0.16}$	92.08 ± 24.39

B. Simulation Evaluations

Improvements with Terrain Awareness: We illustrate a comprehensive terrain and obstacle map within one nav-cell in Fig. 5, exemplifying that the robot equipped with terrain awareness can plan a path with higher traversability. As demonstrated in Table I, when integrated with terrain awareness, **TOP-Nav** surpasses the VP-Nav baseline by approximately 8% in success rate (*SR*). We observe that **TOP-Nav** and the *wo/Proprioception* methods achieve a *TD* of nearly 20%, which is half of the *TD* achieved in methods without terrain awareness. These results indicate that the proposed path planner empowers the robot to select terrains with higher traversability, leading to a significant improvement in the success rate of navigation.

Improvements with Proprioception Advisor: The simulation environments are cluttered with various obstacles and difficult terrains, leading to potential locomotion failures such as getting stuck on irregular terrain surfaces or colliding with unseen obstacles during directional changes (Fig. 5). We demonstrate that the proposed proprioception advisor could address those challenges: In contrast to methods without proprioception, our approach exhibits an approximate 11% enhancement in *SR*. This is evident in the comparison between **TOP-Nav** and wo/Proprioception, as well as the comparison between wo/Terrain and Obstacle-Only. Such ablations also indicate that the terrain traversability integration and the proprioception

Fig. 6. In *scene 1-3*, the green line represents the trajectory of **TOP-Nav**, while the red line represents Obstalce-Only. Our demonstrations for *scene 4,5* are divided into phases that we provide details of the cost map integration in the figure.

advisor each contribute to performance improvement. Meanwhile, even without terrain awareness, the proposed system outperforms VP-Nav by 2.5%, signifying an improvement in our integration of the proprioception advisor compared to existing methods.

Advancements in Locomotion: The metrics of *UT* and *VFT* evaluate the locomotion states. We observe that **TOP-Nav** achieves the lowest *UT* and *VFT*, indicating that selecting simpler terrains contributes to locomotion stability. In the absence of the proprioception advisor, wo/Proprioception and Obstacle-Only exhibit a significant increase in *VFT* due to the robot getting stuck by unexpected obstacles. We evaluate energy consumption with the assumption that when traversing simpler terrain, the robot should exhibit more natural gaits. As a result, **TOP-Nav** and wo/Proprioception exhibit a 10% reduction in energy consumption compared to methods without terrain awareness.

C. Real World Evaluations

As shown in Fig. 6, **TOP-Nav** is evaluated in different environments featured with diverse terrains and obstacles. Among them, the terrains encountered in *scenes 1-3* were provided in the terrain classifier, we assess the improvements introduced by the vision-based terrain estimator in these experiments. *scenes 4-5* are designed to assess the effectiveness of the online correction module when encountering **novel terrains**. In *scene 4*, the terrain classifier does not include high-cost gravel. In *scene 5*, the robot encounters terrains with no prior knowledge, including a slippery detergent surface. The evaluations in *scene 6* demonstrate that with the proposed proprioception advisor, the robot is able to avoid invisible obstacles.

Due to significant velocity estimation errors in the real

TABLE II
EVALUATION RESULTS ON SCENE 1,2,3.

Method	*SR* ↑	*UT*(%) ↓	*AEC* ↓	*Time*(%) ↓
GA-Nav	13/15	19.80	74.42	68.76
VP-Nav	11/15	23.02	71.45	80.12
Obstacle-Only	12/15	22.75	71.09	79.72
TOP-Nav	**15/15**	**7.79**	**59.07**	**65.03**

TABLE III
EVALUATION RESULTS ON SCENE 4 (NOVEL TERRAINS).

Method	*SR* ↑	*UT*(%) ↓	*AEC* ↓	*Time*(%) ↓
GA-Nav	5/5	**7.98**	62.57	67.42
VP-Nav	4/5	26.43	70.30	77.90
Obstacle-Only	5/5	42.33	72.60	70.64
TOP-Nav	5/5	14.30	**57.46**	**62.72**

robot, we exclude *VFT* in the real-world experiments. The *Time* metric measures the proportion of time the robot takes to complete the navigation task. *ST* in scenario 5 assesses the time taken by the robot from encountering a transparent object to completely avoiding the obstacle.

Improvements with the Visual Estimator: The evaluation results for *scenes 1-3* are presented in Table II. **TOP-Nav** successfully completes all 15 trials with superior walking stability and minimal energy consumption. These results affirm the successful deployment of the proposed system on real hardware, allowing the robot to select waypoints on terrains with better traversability and thus execute more natural gaits. In contrast, GA-Nav, trained on RUGD [36], demonstrates notable limitations when confronted with open-world navigation. We observe that the improvements of **TOP-Nav** over GA-Nav mainly stem from: 1) In *scene 1,2*, GA-Nav fails to identify the slabbed road as a low-cost terrain from the bush and gravel. 2) In *scene 3*, GA-Nav incorrectly identifies the lower part of the obstacle as a cement floor, which could be addressed by our integration of obstacles and terrain estimation. Nevertheless, GA-Nav surpasses VP-Nav and Obstacle-Only, which navigates without terrain awareness. With Obstacle-Only, the robot gets stuck by the stones or Grassroots, leading to a higher time consumption. While VP-Nav effectively alerts the robot to locomotion failures that could be brought by such terrains, the absence of terrain awareness prevents the robot from successfully navigating out of these challenging terrains, resulting in inferior locomotion performance.

Improvements with the Online Correction: To evaluate the efficacy of the online corrections for providing the robot with traversability on novel terrains, we conduct experiments in *scene 4* using a terrain classifier trained without gravel. The degraded classifier exhibits an accuracy of **96.30%** on the paved road with an average confidence of **0.87**, whereas the inference on gravel has an average confidence of **0.43**. We provide a navigation example in Fig. 6 and divide it into four phases. In *phase 1*, despite M_P indicating that the robot is traversing a challenging terrain, the goal map M_G guides the robot to continue moving forward until this information is incorporated into M_{T_P}. In *phase 2*, an online correction is applied as the robot encounters similar gravel terrain, leading to an increased cost brought by M_{T_P} in the current direction. Consequently, the robot turns right to tap on the paved road. In *phase 3*, the robot avoids an observed obstacle. In *phase 4*, the historical proprioception continues to indicate the presence of high-cost gravel terrain on the left. The quantitative results are provided in Table III. Despite GA-Nav possessing complete prior knowledge and correctly identifying the paved road in scene 4, our online corrections only lead to a 7% increase in *UT*, mainly due to the forward movement in *phase 1*. VP-Nav allows the robot to exit the gravel terrain with the safety advisor, but it can not retain this information. As a result, the target orientation prompts the robot to return to the gravel, causing increased time consumption and instability.

TABLE IV
EVALUATION RESULTS ON SCENE 5 (NOVEL SLIPPY TERRAINS).

Method	*SR* ↑	*UT*(%) ↓	*AEC* ↓	*ST*(*s*) ↓
VP-Nav	3/5	2.64	102.04	**63.05**
Obstacle-Only	2/5	3.88	119.77	66.14
Sterling	2/5	3.85	113.75	76.52
TOP-Nav	5/5	**1.19**	**83.40**	67.62

Evaluations on Challenge Novel Terrains: In Scene 5, we present a more challenging scenario where the robot encounters terrains without any prior information. A black tarpaulin covered with slippery detergent is laid out in front of the robot, and the target direction is set in this facing direction. To mitigate the disturbances brought by reflections from detergents, the obstacle cost M_O is not considered in this experiment. As demonstrated in Fig. 6, the experiment could be divided into three phases. In *phase 1*, the robot moving forward along the goal direction, but the slippery terrain causes a decline in motion evaluations, resulting in a higher M_P. Simultaneously, the traversability cost of the slippery tarpaulin is recorded along with the visual features. In *phase 2*, the robot begins to turn aside to avoid the high-cost central area. In *phase 3*, the cost of M_G guides the robot to return to the original direction, where it can navigate on the wooden floor with better traversability to complete the task. The results presented in Table IV validate the effectiveness of our method in maintaining stability and improving the success rate by avoiding challenging terrain. Among the comparisons, Obstacle-Only keeps moving forward on the slippy surface and exhibit the worst motion performance. We include Sterling in this comparison to demonstrate the importance of online corrections when incorporating proprioception into terrain estimation. Without online corrections, Sterling behaves similarly to Obstacle-Only, unable to adapt to novel terrains and continuing

TABLE V
EVALUATION RESULTS ON SCENE 6 (INVISIBLE OBSTACLES).

Method	*SR* ↑	*UT*(%) ↓	*AEC* ↓	*ST*(*s*) ↓
VP-Nav	5/5	1.45	42.97	3.30
Obstacle-Only	0/5	/	/	$+\infty$
TOP-Nav	5/5	**0.79**	**34.63**	**1.98**

forward. Compared with the result in *scene 4*, we demonstrate that **TOP-Nav** can handle novel terrains effectively without relying on operator preferences. This showcases the ability of the system to perform open-world navigation.

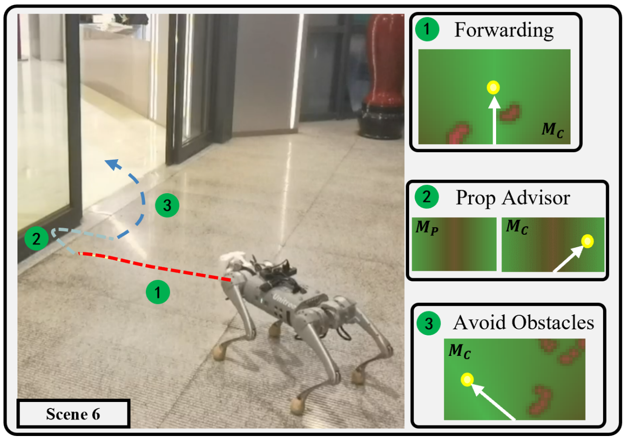

Fig. 7. In *scene 6*, the robot avoids unseen obstacles with the proprioception advisor.

Invisible Obstacles: The proposed proprioception advisor provides close-loop feedback to alert the robot to invisible obstacles. We assess this capability in *scene 6*, where a glass wall is located along the planned path. In *phase 1*, the obstacle estimator is oblivious to the presence of the glass wall, and the robot continues moving forward. In *phase 2*, the robot collides with the glass wall, causing a rapid increase in M_P along the direction of movement. As a result, the robot adjusts its waypoint to steer clear of the high-cost area, successfully avoiding the glass wall. The quantitative results are provided in Table V.

CONCLUSION

We present **TOP-Nav**, a legged navigation system that achieves closed-loop integration of visual and proprioception at both task and motion planning levels. Specifically, not only have we incorporated vision-based obstacle and terrain estimation into the navigation framework, but we have also utilized the proprioception advisor to make online corrections on these vision-only estimations. To validate the proposed system, we conducted extensive experiments in simulated environments featuring diverse terrains and obstacles. Furthermore, the navigation system was successfully deployed on a real robot, and quantitative experiments were conducted. These evaluation results underscore the success of our system in achieving open-world navigation, surpassing limitations posed by terrain, obstacles, or prior knowledge. Compared to existing approaches that couple vision and proprioception or navigate with terrain awareness, our system excels not only in achieving a higher success rate in challenge navigation, but also in demonstrating more stable gaits and lower energy consumption. In the future, we aim to extend the application of the proposed system to longer-distance navigation tasks through precise localization from radar. Furthermore, we plan to enhance the task planner with data-sufficient models such as Transformer.

REFERENCES

[1] Hassan Abu Alhaija, Siva Karthik Mustikovela, Lars Mescheder, Andreas Geiger, and Carsten Rother. Augmented reality meets computer vision: Efficient data generation for urban driving scenes. *International Journal of Computer Vision*, 126:961–972, 2018.

[2] Ananye Agarwal, Ashish Kumar, Jitendra Malik, and Deepak Pathak. Legged locomotion in challenging terrains using egocentric vision. In *Conference on Robot Learning*, pages 403–415. PMLR, 2023.

[3] Xiaoyi Cai, Siddharth Ancha, Lakshay Sharma, Philip R Osteen, Bernadette Bucher, Stephen Phillips, Jiuguang Wang, Michael Everett, Nicholas Roy, and Jonathan P How. Evora: Deep evidential traversability learning for risk-aware off-road autonomy. *arXiv preprint arXiv:2311.06234*, 2023.

[4] Ken Caluwaerts, Atil Iscen, J Chase Kew, Wenhao Yu, Tingnan Zhang, Daniel Freeman, Kuang-Huei Lee, Lisa Lee, Stefano Saliceti, Vincent Zhuang, et al. Barkour: Benchmarking animal-level agility with quadruped robots. *arXiv preprint arXiv:2305.14654*, 2023.

[5] Xuxin Cheng, Kexin Shi, Ananye Agarwal, and Deepak Pathak. Extreme parkour with legged robots. *arXiv preprint arXiv:2309.14341*, 2023.

[6] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3213–3223, 2016.

[7] Mohamed Elnoor, Adarsh Jagan Sathyamoorthy, Kasun Weerakoon, and Dinesh Manocha. Pronav: Proprioceptive traversability estimation for autonomous legged robot navigation in outdoor environments. *arXiv preprint arXiv:2307.09754*, 2023.

[8] Gian Erni, Jonas Frey, Takahiro Miki, Matias Mattamala, and Marco Hutter. Mem: Multi-modal elevation mapping for robotics and learning. In *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 11011–11018. IEEE, 2023.

[9] Parker Ewen, Adam Li, Yuxin Chen, Steven Hong, and Ram Vasudevan. These maps are made for walking: Real-

time terrain property estimation for mobile robots. *IEEE Robotics and Automation Letters*, 7(3):7083–7090, 2022.
[10] David D Fan, Kyohei Otsu, Yuki Kubo, Anushri Dixit, Joel Burdick, and Ali-Akbar Agha-Mohammadi. Step: Stochastic traversability evaluation and planning for risk-aware off-road navigation. *arXiv preprint arXiv:2103.02828*, 2021.
[11] Jonas Frey, Matias Mattamala, Nived Chebrolu, Cesar Cadena, Maurice Fallon, and Marco Hutter. Fast traversability estimation for wild visual navigation. *arXiv preprint arXiv:2305.08510*, 2023.
[12] Jonas Frey, Shehryar Khattak, Manthan Patel, Deegan Atha, Julian Nubert, Curtis Padgett, Marco Hutter, and Patrick Spieler. Roadrunner–learning traversability estimation for autonomous off-road driving. *arXiv preprint arXiv:2402.19341*, 2024.
[13] Zipeng Fu, Ashish Kumar, Jitendra Malik, and Deepak Pathak. Minimizing energy consumption leads to the emergence of gaits in legged robots. *arXiv preprint arXiv:2111.01674*, 2021.
[14] Zipeng Fu, Ashish Kumar, Ananye Agarwal, Haozhi Qi, Jitendra Malik, and Deepak Pathak. Coupling vision and proprioception for navigation of legged robots. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17273–17283, 2022.
[15] Lu Gan, Youngji Kim, Jessy W Grizzle, Jeffrey M Walls, Ayoung Kim, Ryan M Eustice, and Maani Ghaffari. Multitask learning for scalable and dense multilayer bayesian map inference. *IEEE Transactions on Robotics*, 39(1):699–717, 2022.
[16] Tianrui Guan, Divya Kothandaraman, Rohan Chandra, Adarsh Jagan Sathyamoorthy, Kasun Weerakoon, and Dinesh Manocha. Ga-nav: Efficient terrain segmentation for robot navigation in unstructured outdoor environments. *IEEE Robotics and Automation Letters*, 7(3):8138–8145, 2022.
[17] Tairan He, Chong Zhang, Wenli Xiao, Guanqi He, Changliu Liu, and Guanya Shi. Agile but safe: Learning collision-free high-speed legged locomotion. *arXiv preprint arXiv:2401.17583*, 2024.
[18] David Hoeller, Lorenz Wellhausen, Farbod Farshidian, and Marco Hutter. Learning a state representation and navigation in cluttered and dynamic environments. *IEEE Robotics and Automation Letters*, 6(3):5081–5088, 2021.
[19] David Hoeller, Nikita Rudin, Dhionis Sako, and Marco Hutter. Anymal parkour: Learning agile navigation for quadrupedal robots. *arXiv preprint arXiv:2306.14874*, 2023.
[20] Jemin Hwangbo, Joonho Lee, Alexey Dosovitskiy, Dario Bellicoso, Vassilios Tsounis, Vladlen Koltun, and Marco Hutter. Learning agile and dynamic motor skills for legged robots. *Science Robotics*, 4(26):eaau5872, 2019.
[21] Fabian Jenelten, Junzhe He, Farbod Farshidian, and Marco Hutter. Dtc: Deep tracking control. *Science Robotics*, 9(86):eadh5401, 2024.
[22] Sanghun Jung, JoonHo Lee, Xiangyun Meng, Byron Boots, and Alexander Lambert. V-strong: Visual self-supervised traversability learning for off-road navigation. *arXiv preprint arXiv:2312.16016*, 2023.
[23] Simar Kareer, Naoki Yokoyama, Dhruv Batra, Sehoon Ha, and Joanne Truong. Vinl: Visual navigation and locomotion over obstacles. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 2018–2024. IEEE, 2023.
[24] Haresh Karnan, Elvin Yang, Daniel Farkash, Garrett Warnell, Joydeep Biswas, and Peter Stone. Sterling: Self-supervised terrain representation learning from unconstrained robot experience. In *Conference on Robot Learning*, pages 2393–2413. PMLR, 2023.
[25] Haresh Karnan, Elvin Yang, Garrett Warnell, Joydeep Biswas, and Peter Stone. Wait, that feels familiar: Learning to extrapolate human preferences for preference aligned path planning. *arXiv preprint arXiv:2309.09912*, 2023.
[26] Ashish Kumar, Zipeng Fu, Deepak Pathak, and Jitendra Malik. Rma: Rapid motor adaptation for legged robots. *arXiv preprint arXiv:2107.04034*, 2021.
[27] Joonho Lee, Jemin Hwangbo, Lorenz Wellhausen, Vladlen Koltun, and Marco Hutter. Learning quadrupedal locomotion over challenging terrain. *Science Robotics*, 5 (47):eabc5986, 2020.
[28] Gabriel B Margolis, Xiang Fu, Yandong Ji, and Pulkit Agrawal. Learning to see physical properties with active sensing motor policies. *arXiv preprint arXiv:2311.01405*, 2023.
[29] Xiangyun Meng, Nathan Hatch, Alexander Lambert, Anqi Li, Nolan Wagener, Matthew Schmittle, JoonHo Lee, Wentao Yuan, Zoey Chen, Samuel Deng, et al. Terrainnet: Visual modeling of complex terrain for high-speed, off-road navigation. *arXiv preprint arXiv:2303.15771*, 2023.
[30] Takahiro Miki, Joonho Lee, Jemin Hwangbo, Lorenz Wellhausen, Vladlen Koltun, and Marco Hutter. Learning robust perceptive locomotion for quadrupedal robots in the wild. *Science Robotics*, 7(62):eabk2822, 2022.
[31] Pascal Roth, Julian Nubert, Fan Yang, Mayank Mittal, and Marco Hutter. Viplanner: Visual semantic imperative learning for local navigation. *arXiv preprint arXiv:2310.00982*, 2023.
[32] Nikita Rudin, David Hoeller, Marko Bjelonic, and Marco Hutter. Advanced skills by learning locomotion and local navigation end-to-end. In *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 2497–2503. IEEE, 2022.
[33] Joanne Truong, Denis Yarats, Tianyu Li, Franziska Meier, Sonia Chernova, Dhruv Batra, and Akshara Rai. Learning navigation skills for legged robots with learned robot embeddings. In *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 484–491. IEEE, 2021.
[34] Kasi Viswanath, Kartikeya Singh, Peng Jiang, PB Sujit,

and Srikanth Saripalli. Offseg: A semantic segmentation framework for off-road driving. In *2021 IEEE 17th international conference on automation science and engineering (CASE)*, pages 354–359. IEEE, 2021.

[35] Guotai Wang, Wenqi Li, Michael Aertsen, Jan Deprest, Sébastien Ourselin, and Tom Vercauteren. Aleatoric uncertainty estimation with test-time augmentation for medical image segmentation with convolutional neural networks. *Neurocomputing*, 338:34–45, 2019.

[36] Maggie Wigness, Sungmin Eum, John G Rogers, David Han, and Heesung Kwon. A rugd dataset for autonomous navigation and visual perception in unstructured outdoor environments. In *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 5000–5007. IEEE, 2019.

[37] Ruihan Yang, Minghao Zhang, Nicklas Hansen, Huazhe Xu, and Xiaolong Wang. Learning vision-guided quadrupedal locomotion end-to-end with cross-modal transformers. *arXiv preprint arXiv:2107.03996*, 2021.

[38] Xinjie Yao, Ji Zhang, and Jean Oh. Rca: Ride comfort-aware visual navigation via self-supervised learning. In *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 7847–7852. IEEE, 2022.

[39] Chong Zhang, Nikita Rudin, David Hoeller, and Marco Hutter. Learning agile locomotion on risky terrains. *arXiv preprint arXiv:2311.10484*, 2023.

[40] Chong Zhang, Jin Jin, Jonas Frey, Nikita Rudin, Matias Eduardo Mattamala Aravena, Cesar Cadena, and Marco Hutter. Resilient legged local navigation: Learning to traverse with compromised perception end-to-end. In *41st IEEE Conference on Robotics and Automation (ICRA 2024)*, 2024.

[41] Ziwen Zhuang, Zipeng Fu, Jianren Wang, Christopher Atkeson, Soeren Schwertfeger, Chelsea Finn, and Hang Zhao. Robot parkour learning. *arXiv preprint arXiv:2309.05665*, 2023.