Traverse the Non-Traversable: Estimating Traversability for Wheeled Mobility on Vertically Challenging Terrain

Chenhui Pan*, Aniket Datar*, Anuj Pokhrel, Matthew Choulas, Mohammad Nazeri, Tong Xu, and Xuesu Xiao

***Abstract*—Most traversability estimation techniques divide off-road terrain into traversable (e.g., pavement, gravel, and grass) and non-traversable (e.g., boulders, vegetation, and ditches) regions and then inform subsequent planners to produce trajectories on the traversable part. However, recent research demonstrated that wheeled robots can traverse vertically challenging terrain (e.g., extremely rugged boulders comparable in size to the vehicles themselves), which unfortunately would be deemed as non-traversable by existing techniques. Motivated by such limitations, this work aims at identifying the traversable from the seemingly non-traversable, vertically challenging terrain based on past kinodynamic vehicle-terrain interactions in a data-driven manner. Our new Traverse the Non-Traversable (TNT) [1] traversability estimator can efficiently guide a downstream sampling-based planner containing a high-precision 6-DoF kinodynamic model, which becomes deployable onboard a small-scale vehicle. Additionally, the estimated traversability can also be used as a costmap to plan global and local paths without sampling. Our experiment results show that TNT can improve planning performance, efficiency, and stability by 50%, 26.7%, and 9.2% respectively compared with a state-of-the-art off-road navigation method on a physical robot platform.**

I. Introduction

Autonomous navigation in off-road environments is an exciting frontier in robotics research. Its ever-growing applications in search and rescue, planetary exploration, mining, and agriculture warrant extensive research and development in off-road robot mobility. The unpredictable nature of off-road terrain combined with the high risk of catastrophic failures presents significant challenges for mobile robots. Navigating through *vertically challenging* terrain with *wheeled* robots [1] in particular can cause robots to flip over, become airborne, or get stuck on the underlying terrain, especially if the obstacles are comparable in size to the robots themselves.

Most existing navigation systems classify off-road terrain into traversable and non-traversable spaces [2]. Robots then employ path and motion planners to traverse on the traversable terrain, usually composed of free spaces on top of stable ground with minimal slope, to safely reach the goal. However, in extremely challenging or time-critical missions, solely planning within traversable regions may not be possible or effective. Traversable terrain may not exist and limiting plans to only traversable regions can lead to overly conservative paths and delays in mission completion. In such scenarios, more aggressive and risky maneuvers through so-called non-traversable terrain may be necessary to achieve timely results.

Recent work on wheeled mobility has shown that even conventional wheeled vehicles have untapped potential to achieve impressive mobility on vertically challenging terrain [1, 3, 4], with only minimal hardware requirements such as all-wheel drive, independent suspensions, and differential locking. This extended capability highlights that with the right combination of hardware and navigation strategies, even simple wheeled robots can overcome obstacles that were previously deemed non-traversable by state-of-the-art autonomous navigation systems. Apart from end-to-end learning [1], another way to explore the possibility of traversing previously non-traversable terrain is through 6-DoF kinodynamic modeling to predict vehicle-terrain interaction and sampling-based planning to roll-out potential future trajectories for evaluation [3, 4]. However, to deploy such methods robots must perform continuous and high-volume sampling across all terrain on any potential future paths. This process involves numerous queries to a complex kinodynamic model to estimate traversability through a set of high-precision, multi-step 6-DoF trajectory rollouts. Such a combination is computationally demanding for resource-limited mobile robots.

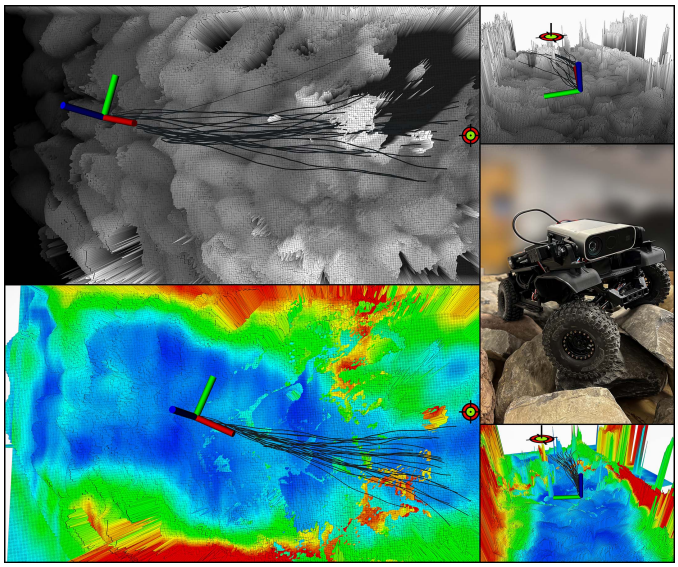

Fig. 1: Guided by our TNT traversability estimator, a sampling-based motion planner with a high-precision kinodynamic model quickly converges to safe trajectories on vertically challenging terrain (bottom, colored traversability map), whereas without the TNT traversability map the planner gets stuck at exploring completely non-traversable area (top, white boulder on the black-white elevation map).

In this work, we introduce Traverse the Non-Traversable (TNT), a terrain traversability estimator for previously non-traversable, vertically challenging terrain. Based on a terrain

[1] https://github.com/RobotiXX/TNT

elevation map, TNT generates a traversability map using previous vehicle-terrain interactions, including roll and pitch angles, discrepancy in vehicle command execution and kinodynamic model prediction. Instead of sampling the entire terrain elevation map and potentially getting stuck on a local optima, TNT filters out obviously non-traversable terrain in advance and allows a motion planner to only explore potentially traversable (while seemingly non-traversable) regions so as to quickly converge to a globally optimal plan. While TNT can guide sampling-based planners like MPPI [5] to improve sample efficiency, it can also be used as a costmap to plan both global and local paths through vertically challenging terrain without sampling. Our physical experiments show that our TNT traversability estimation method can enable both a state-of-the-art sampling-based planner with a high-accuracy 6-DoF kinodynamic model and a search-based planner using a traversability cost function to traverse previously non-traversable, vertically challenging terrain. TNT achieves up to 50%, 26.7%, and 9.2% improvement in success rate, traversal time, and vehicle stability compared with a state-of-the-art method [3].

II. Related Work

Much of the work on off-road navigation originated from the DARPA Grand Challenge [6] and the LAGR [7] program. Research and development in this field have inspired roboticists to push the boundaries of hardware [4, 8], perception [9, 10, 11, 12], planning [13, 14, 15, 16, 17], modeling [3, 18, 19, 20, 21], control [5, 22], and learning [22, 23, 24, 25, 26] in robotics. In this section, we briefly discuss related work to TNT in terms of traversability estimation and sampling-based planning.

A. Terrain Traversability Estimation

Beyond simple obstacle avoidance, accurate terrain traversability estimation is essential for safe and efficient navigation on off-road terrain. Numerous studies have explored various approaches to this problem, e.g., learning from vehicle-terrain interactions [27, 28] and using a kinodynamic model [29] to identify traversable paths and avoid obstacles. Recent works have leveraged both geometric [30] and semantic [31, 32, 33] modalities for scene understanding and predicting environment elements [34] beyond the perception range. Castro *et al.*, [35] estimated terrain traversability using Bird's Eye View (BEV) images and height maps, supervised by pseudo-ground truth cost derived from IMU data. The resulting costmap guides the robot towards traversable paths. While simpler approaches, such as classifying terrain as traversable or non-traversable [36, 37], have proven effective on flat off-road terrain, estimating traversability on vertically challenging terrain requires a more nuanced approach that extends beyond the simple binary classification. Furthermore, real-time decision-making is also crucial during deployment to ensure safe and efficient navigation. Sampling-based planners require continuous planning to facilitate convergence to optimal paths hence requiring quick traversability estimation. Roadrunner [38] achieves this by fusing sensor information with pre-trained image segmentation into a unified BEV representation.

Recent research has demonstrated the importance of accounting for both aleatoric uncertainty, which arises from partial observability [39, 40], and epistemic uncertainty, which stems from model distribution shift [41, 42]. While Endo *et al.* [43] focused solely on slip predictions for epistemic uncertainty, this limited scope proves insufficient for more complex, vertically challenging environments. EVORA [44] presented a more comprehensive approach to traversability estimation by introducing a terrain traction model based on the ratio of commanded to realized velocities while also incorporating both aleatoric and epistemic uncertainties. This makes EVORA closely aligned with TNT, particularly in using commanded and realized velocity measurements and consideration of epistemic uncertainty to address terrain traversability challenges.

B. Sampling-based Planners

Sampling-based planners have proven to be effective tools for finding optimal paths through complex terrain. While many of these planners operate successfully in 2D spaces, navigating challenging off-road environments often necessitates planners that can operate in $\mathbb{SE}(3)$ [45] to generate agile trajectories in cluttered environments.

Search-based motion planners [4, 46] use a deterministic approach to explore the configuration space. When the heuristic is well-defined, planners like A* [47] and Dijkstra's are optimal. However, in an off-road environment, the high-dimensional configuration space makes achieving optimal performance with these planners very computationally expensive. This is where sampling-based planners [48, 49] have an advantage, as they explore the space flexibly by randomly sampling configurations. With the recent development of using a GPU [5, 50] to parallelize sampling, researchers have gravitated towards sampling in high volumes in the configuration space.

While sampling-based planners excel at efficiently finding solutions in complex off-road scenarios, they often suffer from a short planning horizon and lead to sub-optimal paths due to computation constraints. Guided planners [49] have emerged as a promising solution to this challenge. By leveraging information gained from the environment [51], these planners predict what is beyond the planning horizon to prevent exploration of non-traversable regions and enable risk-aware [52, 53, 54, 55] planning in uncertain environments. Similarly, TNT can assist these planners by guiding their sampling region to only (potentially) traversable spaces.

III. Approach

Our hypothesis for the TNT estimator is that traversability is a simplified form of complex kinodynamic modeling. Prior works have shown that 6-DoF kinodynamic models in $\mathbb{SE}(3)$ are necessary for vertically challenging terrain [1, 3, 4], which take into account current robot state, action, and underlying

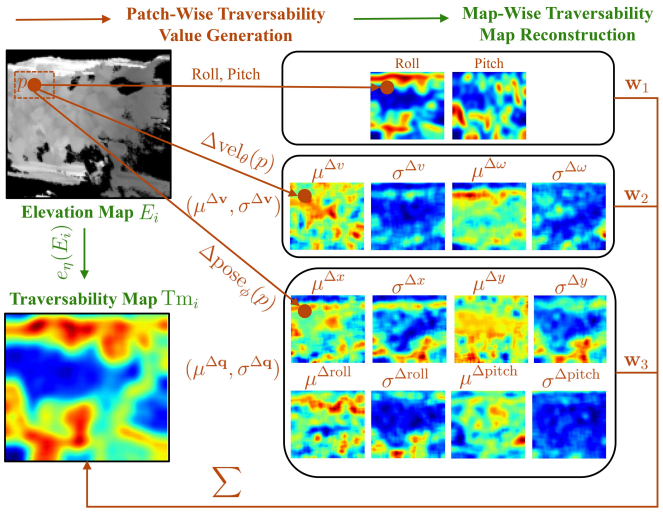

Fig. 2: **TNT Overview.** Based on terrain patches p on the elevation map E_i, three predictors produce roll and pitch angles , velocity discrepancy $(\mu^{\Delta\mathbf{v}}, \sigma^{\Delta\mathbf{v}})$, and pose prediction discrepancy $(\mu^{\Delta\mathbf{q}}, \sigma^{\Delta\mathbf{q}})$, which are combined by $\mathbf{w}_1$, $\mathbf{w}_2$, and $\mathbf{w}_3$ to generate patch-wise traversability values; All traversability values form a traversability map Tm_i, which a map-wise traversability map estimator $(e_\eta(\cdot))$ learns to reconstruct based on terrain patch E_i.

terrain patch beneath the robot to determine the next robot state. A full-scale 6-DoF model, while being precise, cannot be efficiently queried many times in a sampling-based motion planner to reveal the optimal solution through vertically challenging terrain. The key insight of TNT is that certain terrain patches, regardless of robot state and action, will induce undesired robot next state, i.e., being non-traversable. Identifying such terrain patches can limit the sampling space of sampling-based motion planners to only focus on potentially traversable areas in order to find the optimal motion plan to navigate through, or can serve as a costmap for search-based planners without sampling.

Therefore, TNT aims to identify how (non-)traversable a terrain patch is regardless of robot state and action. We separate TNT into two stages: patch-wise traversability value generation and map-wise traversability map reconstruction. The former generates traversability value labels for the latter, which can be queried in real time onboard mobile robots (Fig. 2).

A. Patch-Wise Traversability Value Generation

With the aforementioned simplification, TNT first estimates the traversability value of a terrain patch underneath the robot footprint regardless of robot state and action. The traversability value thus becomes a distribution over different robot states and actions. For vertically challenging terrain, we devise three intermediate metrics to comprise the final traversability value of a terrain patch p: (1) amplitude of roll and pitch angles, (2) velocity discrepancy in terms of the difference between commanded and actual vehicle velocities, and (3) pose prediction discrepancy in terms of the difference between predicted (based on a full 6-DoF model) and actual vehicle poses (from internal or external state estimation).

1) Roll and Pitch Angles: Two most frequent failure reasons on non-traversable vertically challenging terrain are vehicle rollover and getting-stuck due to excessive roll and pitch angles respectively. Therefore, the first component of the patch-wise traversability value is the amplitude of roll and pitch angles on a terrain patch based on a quasi-static physics-based model. The roll and pitch angles are estimated based on the elevation value h_i of the terrain patch pixel under each vehicle wheel i using trigonometry:

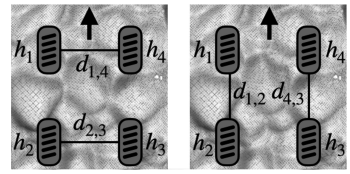

Fig. 3: Roll and Pitch Model.

$$\text{roll}, \text{pitch} = \frac{1}{2}\| \sum_{(i,j)\in\mathcal{I}_{\text{roll},\text{pitch}}} \left(\arctan\left(\frac{h_i - h_j}{d_{i,j}}\right)\right)\|, \quad (1)$$

where $d_{i,j}$ is the distance between the wheels i, j, and the subscript sets for roll and pitch are $\mathcal{I}_{\text{roll}} = \{(1,4),(2,3)\}$ (Fig. 3 left) and $\mathcal{I}_{\text{pitch}} = \{(1,2),(4,3)\}$ (Fig. 3 right), respectively. Notice that Eqn. (1) only considers the terrain patch, not vehicle state and action, and the physics model is deterministic.

2) Velocity Discrepancy: The intuition for the second component of the patch-wise traversability value is that less traversable terrain usually causes more significant discrepancy in terms of vehicle velocities. Particularly, we use the difference between commanded and actual vehicle velocities:

$$\Delta\mathbf{v} = \mathbf{v}_{\text{commanded}} - \mathbf{v}_{\text{actual}},$$

where $\mathbf{v} = (v, \omega)$ includes the linear and angular velocities of the vehicle. However, such a difference is dependent on the robot state and action. For example, if the vehicle is already traveling at a high speed, a steep slope won't cause too much difference between a high velocity command and the actual vehicle speed; However, a stationary vehicle starting on a steep slope with a high velocity command will cause a large difference. Therefore, an elevation map cannot uniquely determine the value of such differences, but rather a distribution over them. For example, extremely rugged terrain will cause a high probability of a large difference, while on a flat ground such a difference is mostly small. So the velocity discrepancy estimator is devised as:

$$(\mu^{\Delta\mathbf{v}}, \sigma^{\Delta\mathbf{v}}) = \Delta\text{vel}_\theta(p), \quad (2)$$

where $\mu^{\Delta\mathbf{v}}, \sigma^{\Delta\mathbf{v}}$ are the mean and standard deviation of the difference in linear (Δv) and angular $(\Delta\omega)$ velocity, whereas the estimator $\Delta\text{vel}_\theta(\cdot)$ is parameterized by learnable parameters θ.

Given a dataset of ground truth $\mathcal{D}_{\Delta\mathbf{v}} = \{\Delta\mathbf{v}_i, p_i\}_{i=1}^N = \{(\Delta v_i, \Delta\omega_i), p_i\}_{i=1}^N$ computed by the difference between the outputs of a vehicle state estimator (e.g., internal Visual-Inertial Odometry or external GPS-RTK or motion capture

system) and velocity commands, we represent $\Delta\text{vel}_\theta(\cdot)$ as a learnable neural network and learn the parameters θ using a negative log likelihood loss:

$$\mathcal{L}_{\mathcal{D}_{\Delta\mathbf{v}}} = \sum_{\substack{\{\Delta\mathbf{v}_i, p_i\} \\ \in \mathcal{D}_{\Delta\mathbf{v}}}} \frac{1}{2}\left(\log(\sigma_i^{\Delta\mathbf{v}^2}(p_i)) + \frac{(\mu_i^{\Delta\mathbf{v}}(p_i) - \Delta\mathbf{v}_i)^2}{\sigma_i^{\Delta\mathbf{v}^2}(p_i)}\right). \tag{3}$$

3) Pose Prediction discrepancy: The last component of the patch-wise traversability value is based on the intuition that it is more difficult to predict vehicle pose on less traversable terrain due to the more complex vehicle-terrain interactions. Therefore, we employ an existing kinodynamic model to predict the pose on a terrain patch $\mathbf{q}_{\text{predicted}}$ and compare it against the actual pose on that patch $\mathbf{q}_{\text{actual}}$:

$$\Delta\mathbf{q} = \mathbf{q}_{\text{predicted}} - \mathbf{q}_{\text{actual}}.$$

Pose $\mathbf{q}$ can include a 6-DoF vehicle state, i.e., x, y, z, roll, pitch, and yaw, and/or their higher-order derivatives. Since pose prediction using a forward kinodynamic model is iterative, the predicted x and y components are highly dependent on yaw, so including yaw is redundant in terms of discrepancy. Furthermore, the discrepancy in height z does not provide significantly more insight compared to other pose components. Hence, in our implementation, we use the most informative components x, y, roll, and pitch (Fig. 2). Similar to Eqn. (2), for pose prediction discrepancy, we employ another data-driven estimator:

$$(\mu^{\Delta\mathbf{q}}, \sigma^{\Delta\mathbf{q}}) = \Delta\text{pose}_\phi(p), \tag{4}$$

parameterized by learnable parameters ϕ. Another negative log likelihood loss, similar to Eqn. (3) but with the $\Delta\mathbf{v}$ components replaced by $\Delta\mathbf{q}$, is used to learn ϕ in Eqn. (4):

$$\mathcal{L}_{\mathcal{D}_{\Delta\mathbf{q}}} = \sum_{\substack{\{\Delta\mathbf{q}_i, p_i\} \\ \in \mathcal{D}_{\Delta\mathbf{q}}}} \frac{1}{2}\left(\log(\sigma_i^{\Delta\mathbf{q}^2}(p_i)) + \frac{(\mu_i^{\Delta\mathbf{q}}(p_i) - \Delta\mathbf{q}_i)^2}{\sigma_i^{\Delta\mathbf{q}^2}(p_i)}\right).$$

In our implementation, we use the Terrain-Attentive Learning (TAL) model [3] with 25-step prediction to produce $\mathbf{q}_{\text{predicted}}$.

4) Final Traversability Value: To determine the final traversability value for the terrain patch p, we combine the roll and pitch prediction in Eqn. (1), the velocity discrepancy in Eqn. (2), and pose prediction discrepancy in Eqn. (4):

$$\begin{aligned}&\text{Traversability}(p) = \\ &\mathbf{w}_1 \cdot [\text{roll}, \text{pitch}]^T + \mathbf{w}_2 \cdot [\mu^{\Delta\mathbf{v}}, \sigma^{\Delta\mathbf{v}}]^T + \mathbf{w}_3 \cdot [\mu^{\Delta\mathbf{q}}, \sigma^{\Delta\mathbf{q}}]^T,\end{aligned} \tag{5}$$

where $\mathbf{w}_1$, $\mathbf{w}_2$, and $\mathbf{w}_3$ are weight vectors of different dimensions to prioritize roll and pitch prediction, velocity discrepancy, and pose prediction discrepancy, along with their internal components, e.g., roll and pitch, linear and angular velocities, and 6-DoF pose dimensions.

B. Map-Wise Traversability Map Reconstruction

In principle, the patch-wise traversability value generation can be directly used during deployment, i.e., estimating the traversability of each terrain patch of interest, e.g., along the sampled vehicle trajectories to compute trajectory traversal costs. However, deploying all three models (Eqns. (1), (2), and (4)) on each terrain patch located at and aligned with every vehicle state along hundreds or thousands of sampled trajectories is extremely computational intensive and therefore would defeat the purpose of traversability estimation (consider one could rather query the high-precision 6-DoF kinodynamic model for trajectory cost estimation [3]). Therefore, we use patch-wise traversability value generation to generate training data to learn a more efficient map-wise traversability map estimator. When moving through vertically challenging terrain, upcoming elevation maps in front of the robot are updated using onboard sensors (e.g., RGB-D camera or 3D LiDAR) and vehicle odometry at a low frequency (e.g., 2Hz). Our traversability map estimator takes as input updated elevation maps, and produces traversability values for each elevation pixel in the form of a traversability map.

During training, we collect a variety of elevation maps $\{E_i\}_{i=1}^M$. On each elevation map E_i, we sample many terrain patches p_j by varying the position indices, $m \sim \{1, 2, ..., H\}$ and $n \sim \{1, 2, ..., W\}$, and orientation angle, $\psi \sim \Psi$, where H and W are the height and width of E_i and Ψ is the set of candidate angles (e.g., $\Psi = \{-\frac{\pi}{2}, -\frac{\pi}{4}, 0, \frac{\pi}{4}, \frac{\pi}{2}\}$). We then acquire $p_i^{m,n,\psi} = g(E_i, m, n, \psi)$ with function $g(\cdot,\cdot,\cdot,\cdot)$ producing the patch centered at (m, n) aligned with ψ on E_i. For each $p_i^{m,n,\psi}$, we query the three estimators for roll and pitch (Eqn. (1)), velocity discrepancy (Eqn. (2)), and pose prediction discrepancy (Eqn. (4)) in order to compute the traversability value $\text{Tr}_i^{m,n,\psi}$ for $p_i^{m,n,\psi}$ (Eqn. (5). For each pixel (m, n) in E_i, we define the pixel value on the traversability map as:

$$\text{Tm}_i^{m,n} = \frac{1}{|\Psi|}\sum_{\forall\psi\in\Psi} \text{Tr}_i^{m,n,\psi}, \tag{6}$$

i.e., the traversability map pixel value $\text{Tm}_i^{m,n}$ at position (m, n) is averaged over all traversability values $\text{Tr}_i^{m,n,\psi}$ at all possible vehicle orientations $\forall\psi \in \Psi$. All individual $\text{Tm}_i^{m,n}, \forall m, n$ compose the full traversability map Tm_i. For each elevation map E_i and traversability map Tm_i, we train a traversability map encoder $e_\eta(\cdot)$ with learnable parameters η, in order to minimize a traversability map reconstruction loss:

$$\eta^* = \underset{\eta}{\text{argmin}} \sum_{i=1}^{M} \|e_\eta(E_i) - \text{Tm}_i\|. \tag{7}$$

$e_\eta(\cdot)$ will be used online to produce Tm_i when a new elevation map E_i is available to guide a sampling-based planner or serve as a costmap for a search-based planner.

IV. Experiments

In this section, we present our experiment results of the TNT traversability estimator. We visualize the TNT traversability

map and show that search-based planners like A* can be used to find the most traversable path. We also use TNT in a sampling-based motion planner with a high-precision 6-DoF kinodynamic model and demonstrate improved sample efficiency and navigation performance.

A. Implementations

1) Robot, Testbed, and Data: We implement TNT on the Verti-4-Wheeler (V4W), an open-source, 1/10th-scale robotic platform [1]. V4W is equipped with advanced mobility features, including a low-high gear switch and lockable front and rear differentials, which enhance its performance on vertically challenging terrain. The perception system comprises a Microsoft Azure Kinect RGB-D camera, while an OptiTrack motion capture system provides odometry data during training data collection only, but not for experiments. Real-time elevation mapping is facilitated by an open-source tool [56], which processes depth input from the Azure Kinect camera. Computational tasks are managed by an onboard NVIDIA Jetson Orin NX computer.

To facilitate experimentation, we construct a scaled testing environment. The testbed consists of hundreds of rocks and boulders, measuring 3.1 m$\times$1.3 m with a maximum height of 0.6 m. This environment is highly reconfigurable, allowing for a wide range of experimental scenarios.

The dataset collected on a V4W robot [1] for traversability estimation encompasses human-teleoperated vehicle controls (throttle and steering commands), elevation maps derived from depth images [56], and odometry data from the motion capture system for vehicle state estimation. To ensure comprehensive coverage, the dataset includes a diverse range of 6-DoF robot states, with particular emphasis on capturing maximum roll and pitch conditions. Additionally, the robot is operated at varying speeds within the same rock configuration to gather data for velocity discrepancy estimation. The resulting dataset consists of 70,816 individual elevation maps and 139 minutes of vehicle teleoperation on the rock testbed.

2) Architectures: The prediction models for velocity discrepancy and pose prediction discrepancy consist of a combination of a 7-layer Convolutional Neural Network (CNN) followed by three fully connected layers with the output dimensions of four (mean and standard deviation of linear and angular velocity, v and ω) and eight (mean and standard deviation of x, y, roll, and pitch) respectively (Fig. 2).

The map-wise traversability map estimator consists of a 4-layer CNN with residual connections to encode the elevation map. The encoding is passed through a 3-layer convolution transpose followed by upsampling to construct a 14-channel map of size 320$\times$260, including two channels for roll and pitch, four for velocity discrepancy, and eight for pose prediction discrepancy. This traversability map estimator is the final model used during deployment to generate the traversability map in real-time onboard the robot. The final traversability map is generated by calculating a weighted sum of all 14 channels, whose weights are manually tuned.

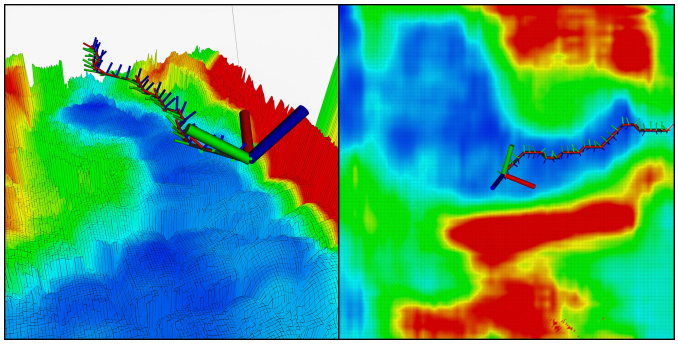

Fig. 4: **Traversability Map Generated by TNT**. The color gradient represents traversability, with blue areas indicating easily traversable terrain and red areas signifying challenging or non-traversable regions. The overlaid path represents the optimal route calculated by the A* algorithm.

B. Traversability Map Visualization and A* Planning

We showcase an example of the traversability map produced by our TNT traversability estimator in Fig. 4. Despite the vertical challenges caused by the rocks and boulders, TNT is able to effectively discern the obviously non-traversable terrain patches from the potentially traversable ones, providing crucial terrain information to downstream mobility tasks, like path planning and motion control. Using the traversability map, we demonstrate that simple search-based planners, such as A*, are able to plan the most traversable path through vertically challenging terrain, without considering vehicle dynamics.

To be specific, we first down-sample the final traversability map to a 31$\times$25 grid using a convolution filter to average the pixel values. Then, the A* algorithm can efficiently compute the optimal path from the robot position to the goal location, minimizing both the traversability cost and Euclidean distance. The A* path can be converted to desired robot positions to be tracked by downstream local planners or controllers, such as the DWA planner [46] or a PID controller.

C. Physical Experiments

We integrate TNT with a sampling-based motion planner, MPPI [50], and deploy it on a physical V4W robot. To address vertically challenging terrain, the MPPI planner employs a state-of-the-art, high precision 6-DoF kinodynamic model, TAL [3], to minimize a cost function considering vehicle trajectories in $\mathbb{SE}(3)$. Previous work [3] demonstrated the computational difficulty in rolling out a large amount of 6-DoF trajectories with high variance using TAL to cover a large portion of the state space, which potentially includes the globally optimal trajectory, thereby compromising real-time planning and navigation performance. In fact, it has been shown that despite its higher kinodynamics accuracy, such a difficulty limits TAL's performance to that of a simplified 6-DoF model decomposed into a planar Ackerman, roll and pitch prediction, and elevation value model, i.e., the WMVCT model and planner [4]. Considering that such a limitation partially motivates TNT (Fig. 1), we showcase the importance

of efficiently biasing the sampling distribution of MPPI using TNT to more efficiently utilize the high-precision, but also high-computation TAL model.

Specifically, we denote the MPPI planner with the TAL model as TAL, whereas TNT indicates that they are augmented by TNT. We also compare with the WMVCT model and planner, denoted as WMVCT. Table I shows the experiment results on a randomly created test course on the vertically challenging testbed (Fig. 5 left).

As shown in Table I , TNT achieves better success rate and traversal time compared to TAL and WMVCT. This is because MPPI without TNT is only guided by the accurate 6-DoF vehicle trajectories produced by the slow TAL model, which must rely on a long horizon in order to rollout until reaching certain terrain patch to determine its cost based on $\mathbb{SE}(3)$ vehicle state. Thus, the requirement on long horizon limits the sample number and variance to a low value due to onboard computation limitation. The robot then lacks efficient anticipation of all possibilities of future paths to explore traversable terrain while preventing completely non-traversable areas. On the other hand, due to the rich traversability information provided by TNT, MPPI can plan with a shorter horizon but still anticipate what is coming up ahead, since all such information has already been efficiently distilled into our traversability map during training. Therefore, TNT helps MPPI to converge faster to regions that are more traversable. Guided by the TNT traversability map, MPPI can rollout a smaller amount of shorter trajectories with a faster convergence time, but still quickly discover the most promising future vehicle trajectory to traverse through the vertically challenging terrain. Similar to the results reported by prior work [3], WMVCT performs similarly or better compared to TAL with a less accurate decomposed 6-DoF kinodynamic model, thanks to the efficient model query time.

We also report the changes in roll and pitch angles, as well as in throttle and steering commands resulted by the three methods. TNT achieves the lowest average and variance across all four metrics, demonstrating a very stable navigation behavior. In most cases, TAL is less stable than WMVCT, except change in roll. The smaller change values in all these four metrics achieved by TNT indicate that the MPPI planner guided by TNT is able to efficiently find traversable paths, without wasting computation and movement effort on exploring undesired, less traversable areas.

D. Outdoor Demonstration

In addition to the indoor physical experiments on the vertically challenging testbed, we also demonstrate that TNT can be deployed in an outdoor off-road environment. The natural outdoor off-road environment is filled with pebbles, rocks, and boulders of a wide range of sizes. Grass, mulch, and gravel are also present next to the rocks. TNT is still able to produce accurate traversability maps to guide subsequent MPPI planner to traverse through previously non-traversable, vertically challenging terrain (Fig. 5 right).

TABLE I: **Experiment Results of TNT, TAL and WMVCT:** success rate, mean traversal time (of successful trials), mean absolute roll and pitch angles, and mean changes in roll, pitch, throttle, and steering.

	TNT	TAL	WMVCT
Success Rate ↑	**9/10**	6/10	8/10
Traversal Time ↓	**17.6s**±2.64s	24.0s±9.88s	21.1s±24.90s
Absolute Roll ↓	**6.6°±7.0°**	9.4°±24.6°	8.3°±11.9°
Absolute Pitch ↓	8.2°±7.4°	**6.9°±10.9°**	8.8°±10.7°
Δ Roll ↓	**0.51°±1.03°**	0.63°±6.07°	0.97°±1.6°
Δ Pitch ↓	**0.53°±0.68°**	0.080°±1.3°	0.57°±1.08°
Δ Throttle ↓	**0.042±0.15**	0.065±0.34	0.053±0.31
Δ Steering ↓	**0.071±0.24**	0.153±0.38	0.087±0.23

Fig. 5: TNT Indoor Experiments and Outdoor Demonstration.

V. Conclusions

We present Traverse the Non-Traversable (TNT), a traversability estimation framework that addresses the limitations of traditional traversability estimation techniques that struggle to identify traversable areas on vertically challenging terrain. The TNT traversability estimator, leveraging data-driven insights from past kinodynamic vehicle-terrain interactions and a physics-based model, enables robots to navigate areas that are deemed non-traversable by traditional traversability estimators. By integrating TNT with a high-precision kinodynamic model and a sampling-based motion planner, or by utilizing it as a costmap for path planning, significant improvements in planning performance, efficiency, and stability have been demonstrated on a physical robot platform. This work paves the way for robots to traverse previously inaccessible environments, expanding their operational capabilities across various domains.

Despite TNT's efficacy in estimating terrain traversability from a purely geometric perspective, i.e., from 2.5D elevation maps, one interesting future direction is to incorporate semantics into the traversability estimator. Grass, mulch, mud, and gravel will exhibit different traversability, although they may look similarly to rocks and boulders in a geometric sense. Terrain granularity and deformability also need to be considered to comprehensively evaluate traversability.

References

[1] A. Datar, C. Pan, M. Nazeri, and X. Xiao, "Toward wheeled mobility on vertically challenging terrain: Platforms, datasets, and algorithms," in *2024 IEEE Interna-*

tional Conference on Robotics and Automation (ICRA). IEEE, 2024, pp. 16 322–16 329.

[2] M. G. Castro, S. Triest, W. Wang, J. M. Gregory, F. Sanchez, J. G. Rogers, and S. Scherer, "How does it feel? self-supervised costmap learning for off-road vehicle traversability," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*, 2023, pp. 931–938.

[3] A. Datar, C. Pan, M. Nazeri, A. Pokhrel, and X. Xiao, "Terrain-attentive learning for efficient 6-dof kinodynamic modeling on vertically challenging terrain," in *2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2024.

[4] A. Datar, C. Pan, and X. Xiao, "Learning to model and plan for wheeled mobility on vertically challenging terrain," *arXiv preprint arXiv:2306.11611*, 2023.

[5] G. Williams, P. Drews, B. Goldfain, J. M. Rehg, and E. A. Theodorou, "Aggressive driving with model predictive path integral control," in *2016 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2016.

[6] G. Seetharaman, A. Lakhotia, and E. P. Blasch, "Unmanned vehicles come of age: The darpa grand challenge," *Computer*, 2006.

[7] L. D. Jackel, E. Krotkov, M. Perschbacher, J. Pippine, and C. Sullivan, "The darpa lagr program: Goals, challenges, methodology, and phase i results," *Journal of Field robotics*, vol. 23, no. 11-12, pp. 945–973, 2006.

[8] S. Talia, M. Schmittle, A. Lambert, A. Spitzer, C. Mavrogiannis, and S. S. Srinivasa, "Hound: An open-source, low-cost research platform for high-speed off-road underactuated nonholonomic driving," *arXiv preprint arXiv:2311.11199*, 2023.

[9] P. Jiang, P. Osteen, M. Wigness, and S. Saripalli, "Rellis-3d dataset: Data, benchmarks and analysis," in *2021 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2021, pp. 1110–1116.

[10] M. J. Procopio, J. Mulligan, and G. Grudic, "Learning terrain segmentation with classifier ensembles for autonomous robot navigation in unstructured environments," *Journal of Field Robotics*, vol. 26, no. 2, pp. 145–175, 2009.

[11] K. Viswanath, K. Singh, P. Jiang, P. Sujit, and S. Saripalli, "Offseg: A semantic segmentation framework for off-road driving," in *2021 IEEE 17th International Conference on Automation Science and Engineering (CASE)*. IEEE, 2021, pp. 354–359.

[12] M. B. Wigness, S. Eum, J. G. Rogers, D. Han, and H. Kwon, "A RUGD dataset for autonomous navigation and visual perception in unstructured outdoor environments," *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 5000–5007, 2019. [Online]. Available: https://api.semanticscholar.org/CorpusID:210972411

[13] X. Xiao, B. Liu, G. Warnell, J. Fink, and P. Stone, "Appld: Adaptive planner parameter learning from demonstration," *IEEE Robotics and Automation Letters*, vol. 5, no. 3, pp. 4541–4547, 2020.

[14] Z. Wang, X. Xiao, B. Liu, G. Warnell, and P. Stone, "Appli: Adaptive planner parameter learning from interventions," in *2021 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2021, pp. 6079–6085.

[15] Z. Wang, X. Xiao, G. Warnell, and P. Stone, "Apple: Adaptive planner parameter learning from evaluative feedback," *IEEE Robotics and Automation Letters*, vol. 6, no. 4, pp. 7744–7749, 2021.

[16] Z. Xu, G. Dhamankar, A. Nair, X. Xiao, G. Warnell, B. Liu, Z. Wang, and P. Stone, "Applr: Adaptive planner parameter learning from reinforcement," in *2021 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2021, pp. 6086–6092.

[17] X. Xiao, Z. Wang, Z. Xu, B. Liu, G. Warnell, G. Dhamankar, A. Nair, and P. Stone, "Appl: Adaptive planner parameter learning," *Robotics and Autonomous Systems*, vol. 154, p. 104132, 2022.

[18] A. Pokhrel, A. Datar, M. Nazeri, and X. Xiao, "CAHSOR: Competence-aware high-speed off-road ground navigation in SE (3)," *IEEE Robotics and Automation Letters*, 2024.

[19] E. Pagot, M. Piccinini, and F. Biral, "Real-time optimal control of an autonomous rc car with minimum-time maneuvers and a novel kineto-dynamical model," in *2020 IEEE/RSJ international conference on intelligent robots and systems (IROS)*. IEEE, 2020, pp. 2390–2396.

[20] H. Karnan, K. S. Sikand, P. Atreya, S. Rabiee, X. Xiao, G. Warnell, P. Stone, and J. Biswas, "Vi-ikd: High-speed accurate off-road navigation using learned visual-inertial inverse kinodynamics," in *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2022, pp. 3294–3301.

[21] X. Xiao, J. Biswas, and P. Stone, "Learning inverse kinodynamics for accurate high-speed off-road navigation on unstructured terrain," *IEEE Robotics and Automation Letters*, vol. 6, no. 3, pp. 6054–6060, 2021.

[22] X. Xiao, B. Liu, G. Warnell, and P. Stone, "Toward agile maneuvers in highly constrained spaces: Learning from hallucination," *IEEE Robotics and Automation Letters*, vol. 6, no. 2, pp. 1503–1510, 2021.

[23] H. Karnan, E. Yang, D. Farkash, G. Warnell, J. Biswas, and P. Stone, "Sterling: Self-supervised terrain representation learning from unconstrained robot experience," in *Conference on Robot Learning*. PMLR, 2023, pp. 2393–2413.

[24] G. Kahn, P. Abbeel, and S. Levine, "Badgr: An autonomous self-supervised learning-based navigation system," *IEEE Robotics and Automation Letters*, vol. 6, no. 2, pp. 1312–1319, 2021.

[25] X. Xiao, B. Liu, and P. Stone, "Agile robot navigation through hallucinated learning and sober deployment," in *2021 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2021.

[26] Z. Wang, X. Xiao, A. J. Nettekoven, K. Umasankar, A. Singh, S. Bommakanti, U. Topcu, and P. Stone, "From agile ground to aerial navigation: Learning from learned hallucination," in *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2021, pp. 148–153.

[27] L. Wellhausen, A. Dosovitskiy, R. Ranftl, K. Walas, C. Cadena, and M. Hutter, "Where should i walk? predicting terrain properties from images via self-supervised learning," *IEEE Robotics and Automation Letters*, vol. 4, no. 2, pp. 1509–1516, 2019.

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

[29] M. V. Gasparino, A. N. Sivakumar, Y. Liu, A. E. Velasquez, V. A. Higuti, J. Rogers, H. Tran, and G. Chowdhary, "Wayfast: Navigation with predictive traversability in the field," *IEEE Robotics and Automation Letters*, vol. 7, no. 4, pp. 10 651–10 658, 2022.

[30] F. Ruetz, N. Lawrance, E. Hernández, P. Borges, and T. Peynot, "Foresttrav: 3d lidar-only forest traversability estimation for autonomous ground vehicles," *IEEE Access*, 2024.

[31] R. Schmid, D. Atha, F. Schöller, S. Dey, S. Fakoorian, K. Otsu, B. Ridge, M. Bjelonic, L. Wellhausen, M. Hutter *et al.*, "Self-supervised traversability prediction by learning to reconstruct safe terrain," in *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2022, pp. 12 419–12 425.

[32] X. Meng, N. Hatch, A. Lambert, A. Li, N. Wagener, M. Schmittle, J. Lee, W. Yuan, Z. Chen, S. Deng *et al.*, "Terrainnet: Visual modeling of complex terrain for high-speed, off-road navigation," *arXiv preprint arXiv:2303.15771*, 2023.

[33] S. Jung, J. Lee, X. Meng, B. Boots, and A. Lambert, "V-STRONG: Visual self-supervised traversability learning for off-road navigation," in *2024 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2024, pp. 1766–1773.

[34] A. Shaban, X. Meng, J. Lee, B. Boots, and D. Fox, "Semantic terrain classification for off-road autonomous driving," in *Conference on Robot Learning*. PMLR, 2022, pp. 619–629.

[35] M. G. Castro, S. Triest, W. Wang, J. M. Gregory, F. Sanchez, J. G. Rogers, and S. Scherer, "How does it feel? self-supervised costmap learning for off-road vehicle traversability," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2023, pp. 931–938.

[36] J. Seo, S. Sim, and I. Shim, "Learning off-road terrain traversability with self-supervisions only," *IEEE Robotics and Automation Letters*, vol. 8, no. 8, pp. 4617–4624, 2023.

[37] J. Frey, M. Mattamala, N. Chebrolu, C. Cadena, M. Fallon, and M. Hutter, "Fast traversability estimation for wild visual navigation," *arXiv preprint arXiv:2305.08510*, 2023.

[38] J. Frey, S. Khattak, M. Patel, D. Atha, J. Nubert, C. Padgett, M. Hutter, and P. Spieler, "Roadrunner–learning traversability estimation for autonomous off-road driving," *arXiv preprint arXiv:2402.19341*, 2024.

[39] P. Ewen, A. Li, Y. Chen, S. Hong, and R. Vasudevan, "These maps are made for walking: Real-time terrain property estimation for mobile robots," *IEEE Robotics and Automation Letters*, vol. 7, no. 3, pp. 7083–7090, 2022.

[40] X. Cai, M. Everett, J. Fink, and J. P. How, "Risk-aware off-road navigation via a learned speed distribution map," in *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2022, pp. 2931–2937.

[41] J. Seo, T. Kim, K. Kwak, J. Min, and I. Shim, "Scate: A scalable framework for self- supervised traversability estimation in unstructured environments," *IEEE Robotics and Automation Letters*, vol. 8, no. 2, pp. 888–895, 2023.

[42] X. Cai, J. Queeney, T. Xu, A. Datar, C. Pan, M. Miller, A. Flather, P. R. Osteen, N. Roy, X. Xiao *et al.*, "Pietra: Physics-informed evidential learning for traversing out-of-distribution terrain," *arXiv preprint arXiv:2409.03005*, 2024.

[43] M. Endo, T. Taniai, R. Yonetani, and G. Ishigami, "Risk-aware path planning via probabilistic fusion of traversability prediction for planetary rovers on heterogeneous terrains," in *2023 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2023, pp. 11 852–11 858.

[44] X. Cai, S. Ancha, L. Sharma, P. R. Osteen, B. Bucher, S. Phillips, J. Wang, M. Everett, N. Roy, and J. P. How, "EVORA: Deep evidential traversability learning for risk-aware off-road autonomy," *IEEE Transactions on Robotics*, 2024.

[45] S. Liu, K. Mohta, N. Atanasov, and V. Kumar, "Search-based motion planning for aggressive flight in SE (3)," *IEEE Robotics and Automation Letters*, vol. 3, no. 3, pp. 2439–2446, 2018.

[46] D. Fox, W. Burgard, and S. Thrun, "The dynamic window approach to collision avoidance," *IEEE Robotics & Automation Magazine*, vol. 4, no. 1, pp. 23–33, 1997.

[47] P. E. Hart, N. J. Nilsson, and B. Raphael, "A formal basis for the heuristic determination of minimum cost paths," *IEEE Transactions on Systems Science and Cybernetics*, vol. 4, no. 2, pp. 100–107, 1968.

[48] S. Karaman and E. Frazzoli, "Sampling-based algorithms for optimal motion planning," *The international journal of robotics research*, vol. 30, no. 7, pp. 846–894, 2011.

[49] J. D. Gammell, S. S. Srinivasa, and T. D. Barfoot, "Informed RRT*: Optimal sampling-based path planning focused via direct sampling of an admissible ellipsoidal heuristic," in *2014 IEEE/RSJ international conference on intelligent robots and systems*. IEEE, 2014, pp. 2997–

3004.

[50] G. Williams, A. Aldrich, and E. A. Theodorou, "Model predictive path integral control: From theory to parallel computation," *Journal of Guidance, Control, and Dynamics*, 2017.

[51] M. Sivaprakasam, S. Triest, W. Wang, P. Yin, and S. Scherer, "Improving off-road planning techniques with learned costs from physical interactions," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2021, pp. 4844–4850.

[52] L. Sharma, M. Everett, D. Lee, X. Cai, P. Osteen, and J. P. How, "RAMP: A risk-aware mapping and planning pipeline for fast off-road ground robot navigation," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2023, pp. 5730–5736.

[53] L. L. Beyer, G. Ryou, P. Spieler, and S. Karaman, "Risk-predictive planning for off-road autonomy," in *2024 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2024, pp. 16 452–16 458.

[54] J. Tordesillas, B. T. Lopez, M. Everett, and J. P. How, "Faster: Fast and safe trajectory planner for navigation in unknown environments," *IEEE Transactions on Robotics*, vol. 38, no. 2, pp. 922–938, 2021.

[55] X. Xiao, J. Dufek, and R. R. Murphy, "Robot risk-awareness by formal risk reasoning and planning," *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 2856–2863, 2020.

[56] T. Miki, L. Wellhausen, R. Grandia, F. Jenelten, T. Homberger, and M. Hutter, "Elevation mapping for locomotion and navigation using gpu," in *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2022.