Terrain-Aware Perceptual Planning for Aerial Vehicles in Martian Environments

Jessica Todd[1], Pedro Roque[1], Adam. A. Johnson[2] and Joel W. Burdick[1]

***Abstract*— Robust state estimation is a cornerstone of successful planetary robotics missions. Visual-inertial odometry (VIO) has emerged as a leading approach in this domain, providing computationally efficient state estimation through the fusion of camera and inertial measurement unit (IMU) data. However, with its reliance on trackable visual features, VIO performance is inherently coupled to the visual quality of the surrounding terrain, which varies significantly across planetary surfaces. To address this challenge, we propose a terrain-aware Model Predictive Path Integral (TA-MPPI) control framework that actively incorporates terrain visual-feature quality into the planning process, guiding the robot towards regions that support reliable state estimation. We evaluate our approach in synthetic worlds and a simulated Mars environment using the PX4-Autopilot software-in-the-loop stack and OpenVINS. Our results demonstrate that terrain-quality-aware planning improves VIO performance compared to terrain-agnostic baselines, highlighting the benefit of incorporating perception awareness into motion planning.**

I. Introduction

Future aerial vehicles on Mars and other planetary bodies will face growing demands from onboard autonomy to navigate safely while maximizing scientific return. Visual inertial odometry (VIO) is widely used for local state estimation for unmanned aerial robotics in GPS-denied environments on Earth, and has recently been demonstrated to great success off-world with the Mars Ingenuity helicopter [1] as the dominant navigation aid between global positioning updates. Future planetary aerial vehicles will likely rely heavily on VIO-based estimation systems [2]. However, the performance of VIO is strongly dependent on the availability of high-quality trackable features, and thus accuracy is affected by both the motion of the vehicle itself (such as motion blur and camera pointing direction) and environmental conditions. Planetary environments, in particular, pose the unique challenge of highly unstructured and low-textured landscapes. Fig. 1 illustrates how poorly textured terrains like sand result in fewer feature detections. This performance limitation was highlighted by the hard impact of the final flight of the Ingenuity helicopter in early 2024, believed to be the result of poor state estimation resulting from the VIO system having too little visual information to work with while flying over a featureless and poorly textured terrain [3].

In this work, we focus on the challenge of navigating the Martian landscape. While many proposed approaches seek to improve robustness by augmenting VIO with additional sensors [4], incorporating alternative feature representations [5], or modifying the estimation algorithm itself to better tolerate feature loss [6], we instead propose incorporating awareness towards visually hazardous terrain directly into the local planning pipeline. By reasoning about the visual characteristics of the environment during planning, the vehicle can proactively select trajectories that maintain reliable feature observations and reduce the likelihood of estimator degradation or failure. Because this approach operates at the planning level, it is broadly applicable across different VIO frameworks and is not tied to any specific sensor modality or feature representation.

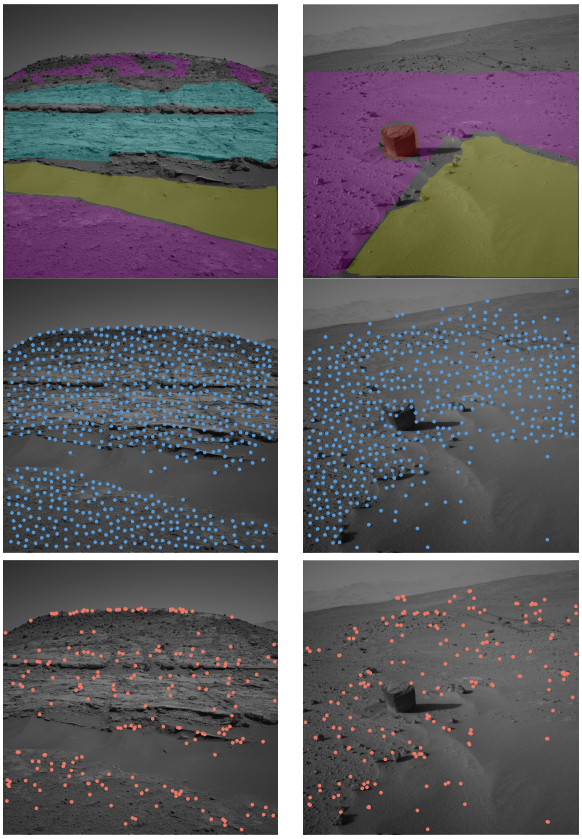


Fig. 1: Terrain classification and feature extraction on two example Curiosity MastCam images. *Top*: Terrain classification using the trained classifier, showing sand (yellow), soil (magenta), exposed bedrock (cyan) and big rock (red). *Middle:* Features extracted by xVIO. *Bottom:* Features extracted by OpenVINS. In both VIO frameworks, far fewer features are extracted in the poorly textured *sand* terrain class.

[1]Jessica Todd, Pedro Roque and Joel Burdick are with Engineering Applied Sciences, California Institute of Technology, Pasadena, CA 90035, USA {jetodd, roque, jburdick}@caltech.edu

[2]Adam Johnson is with the Jet Propulsion Laboratory, California Institute of Technology, Pasadena, CA 91109, USA adam.a.johnson@jpl.nasa.gov

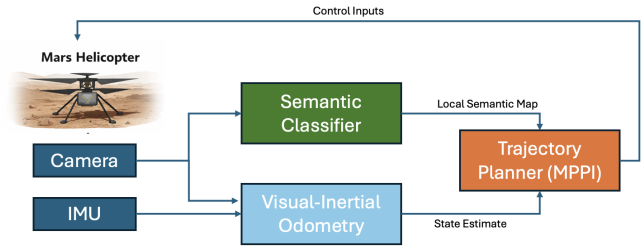


Fig. 2: System diagram for Terrain-Aware MPPI

II. Approach

Our approach combines a semantic classifier with a Model Predictive Path Integral (MPPI) framework to plan trajectories that will reduce state estimation failure. We use a semantic classifier trained on Martian data to categorize the surrounding landscape into terrain classes that are mapped to expected feature quality. While existing perception-aware planning methods are largely feature-centric, seeking to maintain visibility of specific high-quality visual features [7], we instead reason about the expected feature observability of the environment at the level of terrain semantics. We propose a terrain-aware Model Predictive Path Integral (TA-MPPI) planner that uses this classifier to evaluate candidate trajectories by estimating the expected visual feature quality of future camera viewpoints conditioned on the observed terrain. This enables the vehicle to fly a path that maintains views of high quality terrains and thus maintain good state estimation. The proposed system is shown in Fig. 2. By representing perception quality through a probabilistic terrain model rather than individual features, the approach naturally supports the incorporation of prior predictions of terrain quality from external sources such as orbital imagery, previous flights, or scouting robots. This enables the planner to fuse prior environmental knowledge with onboard perception, maintaining informative regions in view while progressing toward a goal.

1) Semantic classifier: We perform terrain classification using DeepLab V3 [8], a popular pixel-wise semantic segmentation model, with a pretrained ResNet-50 backbone. We fine-tuned the model using domain-specific data from the AI4Mars dataset [9]. This dataset consists of ∼326K segmentation full image labels on 35K images from the Curiosity, Opportunity and Spirit rovers. For this work, we limited our dataset to Curiosity images where greater than 50% of the image is labeled (to avoid images dominated by robotic components), resulting in ∼10K images of 1024×1024 pixels, which we used for training (∼5K images) and testing (∼5K images). The AI4Mars dataset includes labels for four terrain types: sand, soil, big rock and exposed bedrock, however given the small fraction of pixels across all images containing big rock ($< 1\%$) and the similar VIO performance in both terrain types, we combined bedrock and big rock into a single *rock* class. The resulting classifier produces per-pixel class probabilities across three terrain types: *sand*, *soil*, and *rock* (see Fig. 1 examples). We take the world belief as the most likely class at each pixel.

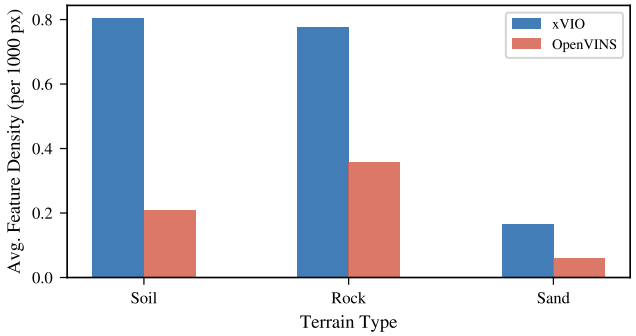


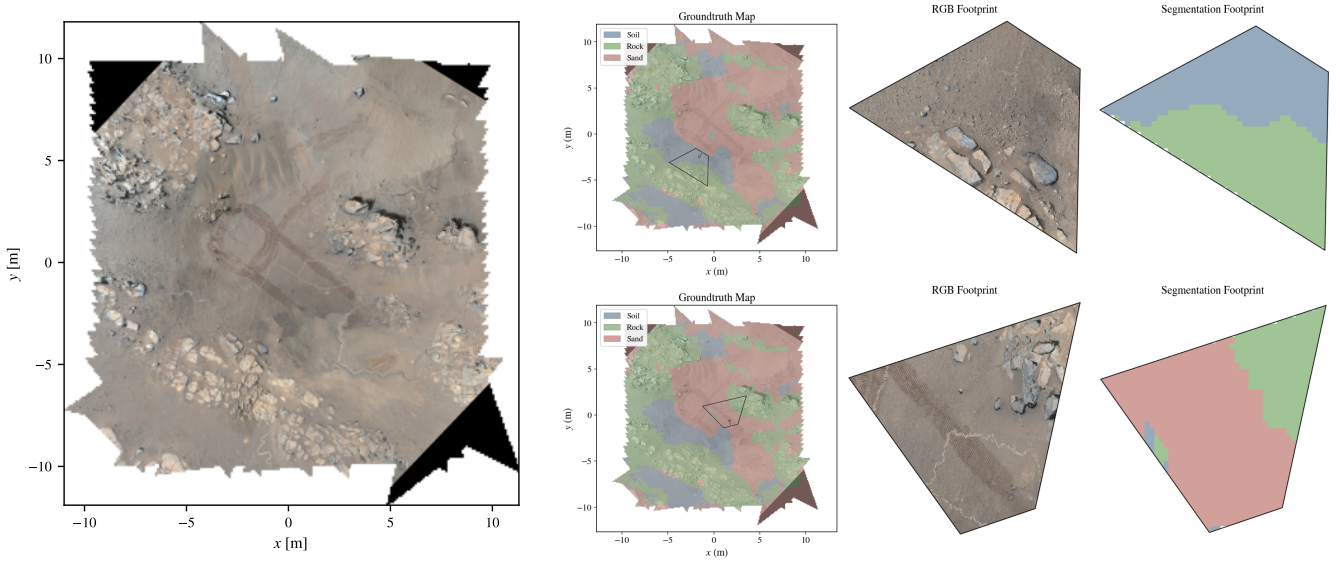
Fig. 3: Average feature density per 1000 pixels in AI4Mars training dataset images across three terrain types, *Soil*, *Rock* and *Sand*. Both xVIO and OpenVINS rely on FAST features but use different filtering methods.

2) Cost estimation: To enable path planning, we design a cost metric based on the performance of VIO frameworks across different terrain types. We focus on two VIO implementations - OpenVINS [10] an opensource VIO framework widely implemented in UAVs, and xVIO [11], the Jet Propulsion Laboratory (JPL)-developed opensource VIO framework. Using the same training dataset as the semantic classifier, we process the images through the feature detection and filtering stages of the xVIO and OpenVINS pipelines to extract feature quality metrics (see Fig. 3). These metrics were then aggregated by terrain class, enabling us to characterize the expected visual feature quality $\mathcal{Q} \in [0, 1]$ associated with each terrain type. For a given camera view of the Martian surface, we derive a viewpoint score S based on the area of a given terrain type and its feature quality, shown in (1).

$$S(\boldsymbol{x}) = \frac{\int_{\mathcal{F}(\boldsymbol{x})} \mathcal{Q}(\mathbf{p}) w(\mathbf{p}) dA}{\int_{\mathcal{F}(\boldsymbol{x})} w(\mathbf{p}) dA} \tag{1}$$

$\mathcal{F}(\boldsymbol{x}) \subset \mathbb{R}^2$ denotes the ground-plane projection of the camera footprint at robot state $\boldsymbol{x}$, computed using homographic projection of the image plane onto the groundplane. Fig. 4 shows some example footprint projections for the Martian world map. $\mathcal{Q}(\mathbf{p}) \in \{\mathcal{Q}_1, ...\mathcal{Q}_k\}$ is the quality score for a point $\mathbf{p} = (x, y)$ on the ground plane in world coordinates, given k terrain types in the environment. $w(\mathbf{p})$ is a spatial weighting function that downweights terrains at the edges of the sensor footprint and where camera distortion is greatest and thus the confidence in the classification is lower. We implemented a Gaussian weighting function $w_{\mathbf{p}} = \exp\left(-\frac{\|\mathbf{p}-\mathbf{p}_c^*\|^2}{2\sigma^2}\right)$ where $\mathbf{p}_c$ is the projection of the camera principal point into the groundplane. Given the angle of the camera and the perspective distortion at the far edge of the sensor footprint, we use $\mathbf{p}_c^* = \mathbf{p}_c + \delta$ to shift the center of the Gaussian towards the drone body in the sensor footprint, concentrating the weights at the edge of the sensor footprint nearer to the drone body.

3) MPPI planner: Costs as represented by terrain types exhibit discontinuities between class types and in unobserved portions of the world. As such, we opt for a sampling-based



(a) Martian reconstruction

(b) Example projected views

Fig. 4: We use a simulated Martian landscape constructed using photogrammetry from Curiosity MastCam images (*left*). Example projected footprints are shown overlaid on the fully classified Martian landscape, showing the projected RGB footprint and resulting semantically segmented footprint (*right*).

Model Predictive Control (MPC) framework to deal with the non-convexity of the objective function and discontinuities in the data. Our terrain-aware MPPI planner optimizes perception objectives by maximizing visibility of 'high feature quality' terrain types. Previous perception-aware MPC approaches have focused on keeping singular points of interest as close to the center of the camera frame as possible [12], or rely on depth sensors for awareness of unexplored regions [13]. In contrast, our terrain-aware method takes semantic information across the entire camera frame into account, ensuring high-quality and trackable features frame to frame. The MPPI optimization problem is given as

$$\min_{u} \sum_{k=0}^{N-1} \ell(\boldsymbol{x}_k, \boldsymbol{u}_k, t_s) + V(\boldsymbol{x}_N) \tag{2}$$
$$\text{s.t.} \quad \boldsymbol{x}_0 = \boldsymbol{s}, \quad \boldsymbol{x}_{k+1} = f(\boldsymbol{x}_k, \boldsymbol{x}_k), \quad 0 \leq k < N$$

where $\boldsymbol{x}_0 = \boldsymbol{s}$ is the problem initial condition, $\boldsymbol{x}_{k+1} = f(\boldsymbol{x}_k, \boldsymbol{x}_k)$ are the discrete-time simplified system kinematics at hover where the state is position and heading $\boldsymbol{x}_k = [\boldsymbol{p}_k, \psi_k] \in \mathbb{R}^3$ and the control input is linear and angular velocities $\boldsymbol{u}_k = [\boldsymbol{v}_k, \omega_k] \in \mathbb{R}^3$. The stage cost is formulated similarly to [13],

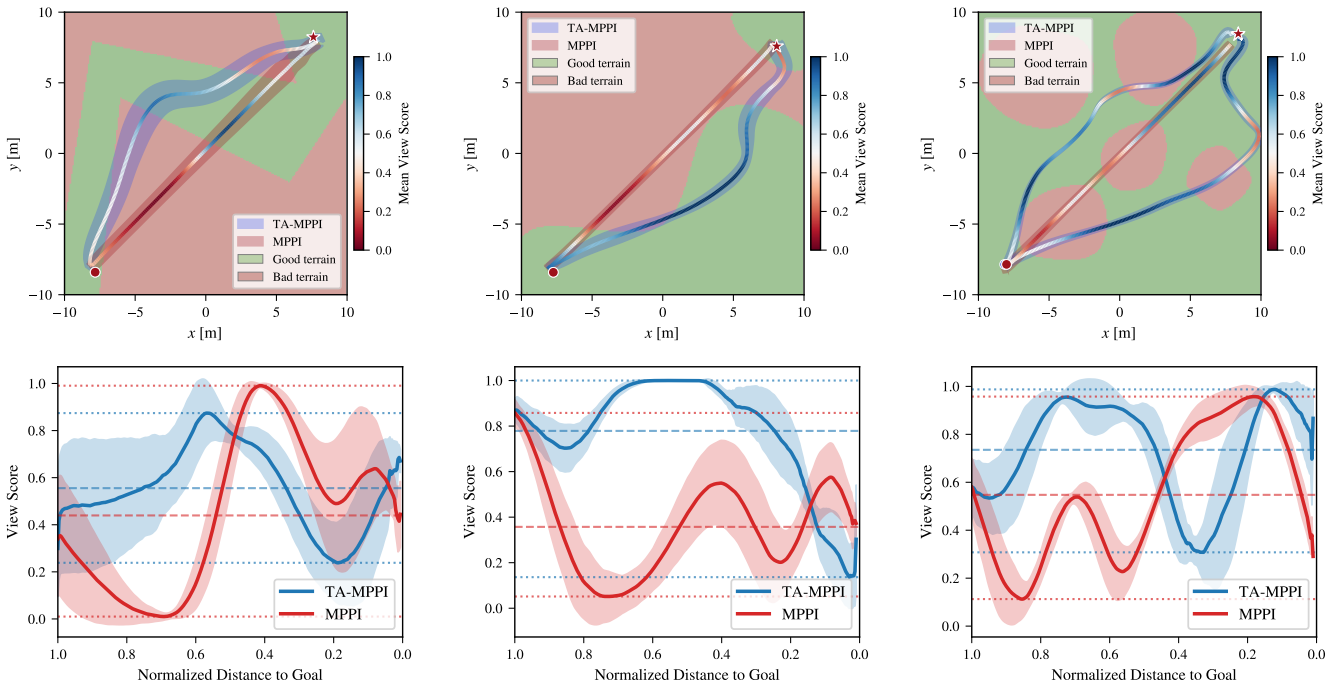
$$\begin{aligned} \ell(\boldsymbol{x}_k, \boldsymbol{u}_k, t_s) = \ell_{goal} + \ell_{action} + \ell_{\psi} \\ + \ell_{perception} + \ell_{boundary} \end{aligned} \tag{3}$$

$$\begin{aligned} \text{where} \quad \ell_{goal} &= c_{goal}\gamma(t_s)d_{goal}, \\ \ell_{action} &= c_{action}\|\boldsymbol{u}_k\|_R^2 \\ \ell_{\psi} &= c_{\psi}e_{\psi}, \\ \ell_{perception} &= c_{perception}(1 - S_k(\boldsymbol{x})), \\ \ell_{boundary} &= c_{boundary} \cdot \mathbb{1}_{\boldsymbol{x}_k \notin map}. \end{aligned}$$

The goal cost ℓ_{goal} penalizes distance $d_{goal} = \|\boldsymbol{x} - \boldsymbol{x}_{goal}\|^2$ from the desired goal position, subject to a stagnation time t_s, that penalizes insufficient progress to the goal. We compute progress to the goal as the rate of change in goal distance $r(t) = \frac{d_{goal}(t-\Delta t) - d_{goal}(t)}{\Delta t}$ where Δt is the control update interval. When $r(t)$ falls below a threshold r_{th}, stagnation time t_s accumulates, exponentially increasing the goal cost using a stagnation scaling factor $\gamma(t_s) = k_s r_s^t$ where k_s is the stagnation gain and r_s is the stagnation growth rate. This helps the vehicle to escape local suboptimal behavior. ℓ_{act} penalizes control effort. The heading alignment cost ℓ_{ψ} encourages alignment of the vehicle's forward-facing camera with the direction of motion using the deviation e_{ψ} between the velocity vector and yaw angle ψ. The boundary cost $\ell_{boundary}$ is an indicator function that returns 1 if the vehicle position falls outside the established world map, with $c_{boundary}$ giving a large penalty if the vehicle strays outside the bounds of the world. $\ell_{perception}$ captures the semantic cost. We assume the world lies in a flat plane, and define four rays, representing the corners of the camera FOV, that start at the camera origin and inscribe a polygonal sensor footprint in the world space defining our projected viewpoint, for which we calculate the viewpoint score using eq. (1). The terminal cost is usually defined as $V(\boldsymbol{x}_N) = \frac{c_{terminal}}{c_{goal}} l_{goal}$, with $c_{terminal} \gg c_{goal}$.

A. Results

We evaluated the proposed terrain classifier and terrain-aware MPPI planner in a simulation environment, with PX4 [14] providing the flight control stack, Gazebo [15] the vehicle dynamics and onboard sensor simulation, and ROS 2 [16] the middleware used to collect sensor information and send



(a) *Zigzag* world (b) *Crossing* world (c) *Blob* world

Fig. 5: Trajectories over various simulated synthetic terrain worlds. *Top:* The mean trajectory $\mu \pm 1\sigma$for $n = 20$ missions is shown for both the TA-MPPI and MPPI. In all three worlds, the straight line trajectory represents the mean MPPI trajectory. The mean start and goal locations are shown as a circle and star respectively. The instantaneous viewpoint score along the trajectory is represented as the trajectory colormap. Note that for the *Blob* world, the trajectories fell into two distinct homotopic classes of $n = 7$ and $n = 13$ so we chose to depict these as separate mean trajectories. *Bottom:* The mean instantaneous viewpoint score $\mu_S \pm 1\sigma$, with the trajectory mean score (dashed line) and absolute mean min and max score (dotted line) are shown.

commands to PX4. A quadrotor vehicle model with a forward mounted camera, angled at 45^o down, was used to emulate the aerial platform, with a mission objective to autonomously navigate from a specified starting position to a designated science goal. We use a sampling time of $\Delta t = 0.1$ seconds and a rollout length of $N = 20$, with 3000 samples. As our stage cost is sufficiently rich, we do not penalize the terminal state. During the mission, the planner samples both position and potential viewpoints along a trajectory, leveraging the terrain classifier to reason about expected perception quality and generate trajectories that balance progress toward the goal with maintaining informative visual observations of the environment.

To first evaluate the performance of the TA-MPPI, we simulated three synthetic worlds with different challenging scenarios for the path planner (see Fig. 5). The *Zigzag* world consists of an angled corridor of *good* (highly featured) terrain running through *bad* (low featured) terrain that prevents straight line navigation to the goal. This world is intended to test the ability of the TA-MPPI to trade off goal distance with maintaining a good viewpoint score. The *Crossing* world incorporates a poorly featured *bad* region around the goal location, testing the ability of the planner to not get stuck in a local minima by remaining in *good* terrain and not progressing to the goal. The final *Blobs* world distributes regions of poor quality terrain across the world, similarly to an obstacle avoidance scenario. In all three cases, the start and goal locations were placed in the lower left and upper right corners, respectively, with a 1-meter uniformly random offset on the start and goal positions and 45-degree uniformly random initial heading ψ over different simulation trials. For each pair of start and goal locations, the mission was run twice, once with our TA-MPPI and once with a baseline MPPI that included only goal, action and boundary costs. For these simulations, $\mathcal{Q}_{good} = 1$ and $\mathcal{Q}_{bad} = 0$. The resulting trajectory plots (Fig. 5) show that the TA-MPPI moves the drone towards the high quality terrain while consistently making progress towards the goal. For *Crossing*, the TA-MPPI is able to maintain a higher viewpoint score compared to the baseline MPPI, until right at the goal, a consequence of the last bad terrain region. In constrast, *Blobs* and *Zigzag* see an initial improvement in performance of TA-MPPI over MPPI, but due to the geometry of the world and the dynamics of the vehicle, the MPPI does sometimes have a mean higher score towards the latter part of the mission. It is important to note that in all three worlds, TA-MPPI maintains a higher mean trajectory viewpoint score across missions, and avoids catastrophically low absolute minimum

viewpoint scores, unlike the MPPI. These very low view scores are indicative of feature-poor terrain and thus would put the vehicle at risk of navigational failure.

Our second experiment tested the TA-MPPI on a simulated Martian landscape. The ground-truth world consists of a photogrammetry-generated Martian landscape constructed from Mars Curiosity MastCam imagery[1] (shown in Fig. 4), providing realistic terrain geometry and appearance representative of planetary surface conditions. For this experiment we ran OpenVINS alongside the TA-MPPI to understand how feature quality is varying across the mission. The nature of the photogrammetry reconstruction, which uses a triangulated mesh to build a 3D environment, introduces a number of synthetic artifacts into the reconstruction which can be picked up and tracked by OpenVINS FAST feature detector. To counter this limitation and more closely emulate the performance observed on the AI4Mars dataset (see Fig. 1 and 3) and indeed on the Ingenuity helicopter itself, we introduced a dropout mask into the OpenVINS pipeline, which drops out features in sandy regions, so that there is an observable difference between feature counts for different terrain regions. Based on the feature performance for OpenVINS on the AI4Mars data, we set $\mathcal{Q}_{sand} = 0, \mathcal{Q}_{soil} = 0.5$ and $\mathcal{Q}_{rock} = 1$. For these missions, start and goal locations were randomized across the world, and TA-MPPI and MPPI run for each set of locations. Two of the 20 simulated trajectories are shown in Fig. 6c and 6d, showing how the TA-MPPI moves the drone to keep good terrain in view of the camera as it moves towards the goal. Fig. 6a shows the mean difference in viewpoint score for TA-MPPI and MPPI, where $\Delta S = S_{\text{TA-MPPI}} - S_{\text{MPPI}}$ throughout the missions. The positive difference demonstrates that TA-MPPI is consistently achieving higher viewpoint scores compared to the baseline MPPI. Likewise, Fig. 6b shows the mean difference in visual features counts used by OpenVINS across the two algorithms. TA-MPPI consistently achieves a higher feature count. By using our terrain-aware MPPI planner, the quadrotor alters its path to remain in 'high quality' terrains during its mission, resulting in a higher feature count than baseline methods, and thus lower chance of state estimation failure during the mission.

III. Conclusions and Future Work

In this work, we present Terrain-Aware MPPI (TA-MPPI), a sampling-based motion planning framework that explicitly accounts for terrain type to improve Visual-Inertial Odometry performance in autonomous aerial vehicles. Comparative evaluation against a terrain-agnostic MPPI baseline demonstrates that TA-MPPI selects trajectories that prioritize viewpoints of high quality terrain types, maintaining a higher number of mean tracked features and avoiding high-risk regions of low feature count that could result in estimation failure.

We plan to extend the performance of the TA-MPPI scheme by integrating risk-aware planning formulation to improve robustness in uncertain and partially observable environments, as well as pairing TA-MPPI with a terrain-aware global planner for large traversals. A major limitation we encountered during this work was the lack of public usable Martian data that contained a wide selection of terrain types, and demonstrated failure cases that realworld missions have encountered. Consequently we will be investigating methods for synthetic terrain generation, conditioned on real Martian data, with the goal to demonstrate our work on a broader array of terrains.

[1]Online. Available at `https://skfb.ly/opQMQ` on April 18, 2026.

A. Acknowledgements

This work was supported by a grant from the Center for Autonomous Systems and Technologies, California Institute of Technology. A portion of this research was carried out at the Jet Propulsion Laboratory, California Institute of Technology under contract with the National Air and Space Administration.

References

[1] D. S. Bayard, D. T. Conway, R. Brockers, J. H. Delaune, L. H. Matthies, H. F. Grip, G. B. Merewether, T. L. Brown, and A. M. S. Martin, "Vision-Based Navigation for the NASA Mars Helicopter," in *AIAA Scitech 2019 Forum*. American Institute of Aeronautics and Astronautics, 2019.

[2] J. Delaune, D. S. Bayard, and R. Brockers, "Range-Visual-Inertial Odometry: Scale Observability Without Excitation," *IEEE Robotics and Automation Letters*, vol. 6, no. 2, pp. 2421–2428, Apr. 2021. [Online]. Available: https://ieeexplore.ieee.org/document/9353193

[3] NASA, "NASA Performs First Aircraft Accident Investigation on Another World - NASA," Dec. 2024. [Online]. Available: https://www.nasa.gov/missions/mars-2020-perseverance/ingenuity-helicopter/nasa-performs-first-aircraft-accident-investigation-on-another-world/

[4] M. Alessandro 1, C. Adrian 2, G. G. . . D. o. Aerospace, P. d. T. Mechanical Engineering, P. d. l. C. . giorgio. guglieri@polito.it 2 Dronomy, and adrian.carrio@dronomy.es, "Enhancing Visual–Inertial Odometry Robustness and Accuracy in Challenging Environments," p. 71, 2025, num Pages: 71.

[5] X. Li, C. Liu, and X. Yan, "Robust Visual-Inertial Odometry with Learning-Based Line Features in a Illumination-Changing Environment," *Sensors*, vol. 25, no. 16, Aug. 2025. [Online]. Available: https://www.mdpi.com/1424-8220/25/16/5029

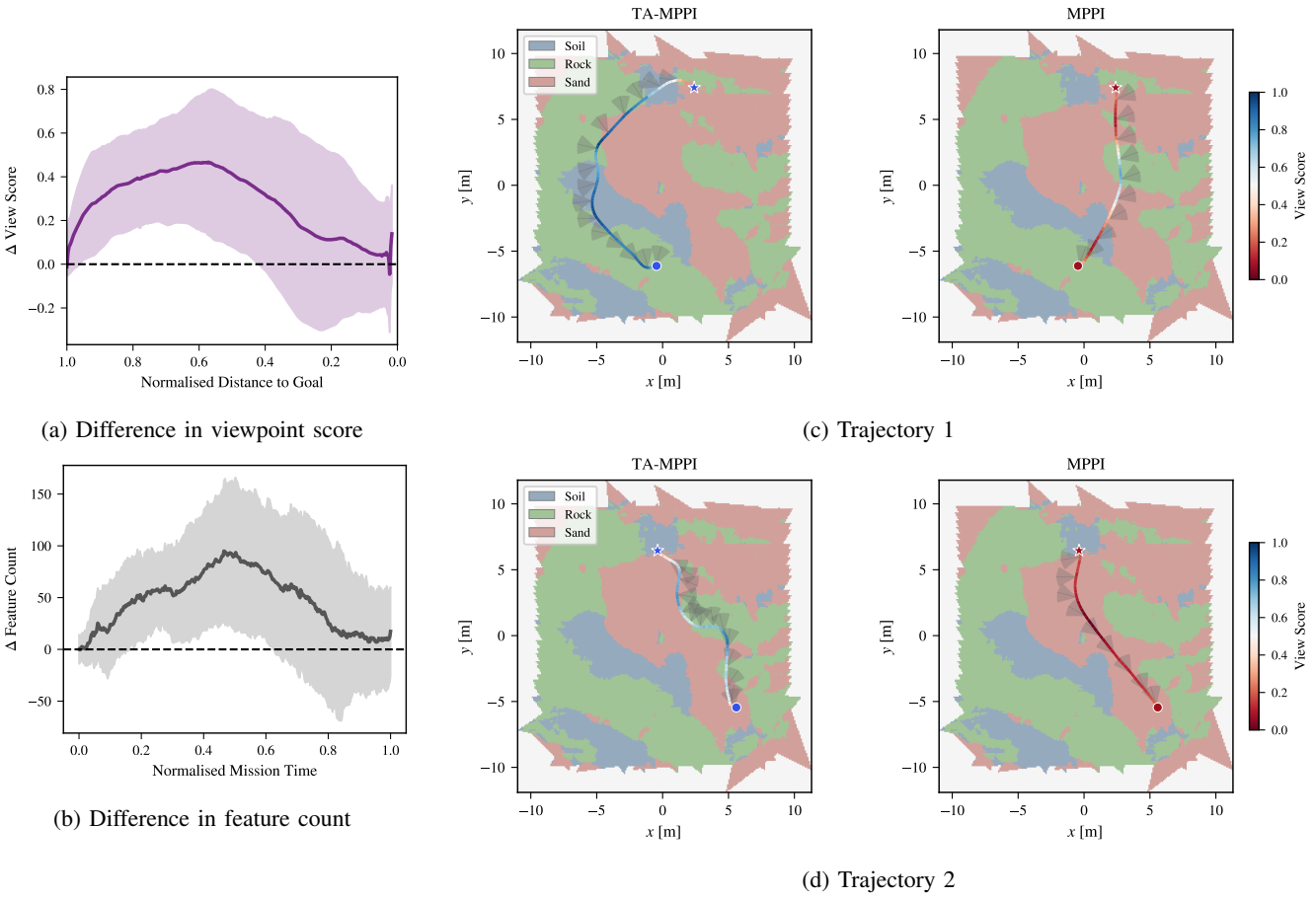
[6] R. Nakashima and A. Seki, "Uncertainty-Based Adaptive Sensor Fusion for Visual-Inertial Odometry under Various Motion Characteristics," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*, May 2020, pp. 3119–3125. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/9197397

[7] X. Wu, S. Chen, K. Sreenath, and M. W. Mueller, "Perception-Aware Receding Horizon Trajectory Planning for Multicopters With Visual-Inertial Odometry," *IEEE Access*, vol. 10, pp. 87 911–87 922, 2022, conference Name: IEEE Access. [Online]. Available: https://ieeexplore.ieee.org/document/9863844/?arnumber=9863844

[8] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 40, no. 4, pp. 834–848, Apr. 2018. [Online]. Available: https://ieeexplore.ieee.org/document/7913730

[9] R. M. Swan, D. Atha, H. A. Leopold, M. Gildner, S. Oij, C. Chiu, and M. Ono, "AI4MARS: A Dataset for Terrain-Aware Autonomous Driving on Mars," in *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, Jun. 2021, pp. 1982–1991. [Online]. Available: https://ieeexplore.ieee.org/document/9523149

[10] P. Geneva, K. Eckenhoff, W. Lee, Y. Yang, and G. Huang, "OpenVINS: A Research Platform for Visual-Inertial Estimation," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*. Paris, France: IEEE, May 2020, pp. 4666–4672. [Online]. Available: https://ieeexplore.ieee.org/document/9196524/



(a) Difference in viewpoint score

(b) Difference in feature count

(c) Trajectory 1

(d) Trajectory 2

Fig. 6: *Right:* Mean difference in viewpoint score $\mu_{\Delta S} \pm 1\sigma$ (*top*) and feature count $\mu_{\Delta Count} \pm 1\sigma$ (*bottom*) between TA-MPPI and MPPI (Δ = TA-MPPI - MPPI) for $n = 20$ missions in the *Martian* world. *Left:* Example trajectories for the *Martian* world, showing the instaneous view score along the mission, with the start and goal locations shown as a circle and star respectively. The yaw angle ψ represented as the grey wedge illustrates where the camera is pointing at various points along the trajectory.

[11] J. Delaune, D. S. Bayard, and R. Brockers, "xVIO: A Range-Visual-Inertial Odometry Framework," *arXiv:2010.06677 [cs]*, Oct. 2020, arXiv: 2010.06677. [Online]. Available: http://arxiv.org/abs/2010.06677

[12] D. Falanga, P. Foehn, P. Lu, and D. Scaramuzza, "PAMPC: Perception-Aware Model Predictive Control for Quadrotors," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Oct. 2018, pp. 1–8. [Online]. Available: https://ieeexplore.ieee.org/document/8593739/

[13] Y. Zhai, R. Reiter, and D. Scaramuzza, "PA-MPPI: Perception-Aware Model Predictive Path Integral Control for Quadrotor Navigation in Unknown Environments," Sep. 2025, arXiv:2509.14978 [cs] version: 1. [Online]. Available: http://arxiv.org/abs/2509.14978

[14] L. Meier, D. Honegger, and M. Pollefeys, "PX4: A node-based multithreaded open source robotics framework for deeply embedded platforms," in *2015 IEEE International Conference on Robotics and Automation (ICRA)*, May 2015, pp. 6235–6240, iSSN: 1050-4729. [Online]. Available: https://ieeexplore.ieee.org/document/7140074

[15] N. Koenig and A. Howard, "Design and use paradigms for Gazebo, an open-source multi-robot simulator," in *2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, vol. 3, Sep. 2004, pp. 2149–2154 vol.3. [Online]. Available: https://ieeexplore.ieee.org/document/1389727

[16] S. Macenski, T. Foote, B. Gerkey, C. Lalancette, and W. Woodall, "Robot operating system 2: Design, architecture, and uses in the wild," *Science Robotics*, vol. 7, no. 66, p. eabm6074, 2022. [Online]. Available: https://www.science.org/doi/abs/10.1126/scirobotics.abm6074