

SCUOLA SUPERIORE MERIDIONALE

University of Naples Federico II



DOCTORAL THESIS

in Cosmology, Space Science & Space Technology

Vision-based Geo-Localization of Future Mars Rotocraft in Challenging Illumination Conditions using Deep Learning

Author:

Dario Pisanti

Supervisors:

Dr. Georgios Georgakis

Prof. Michele Grassi

Prof. Giancarmine Fasano

*Submitted in fulfilment of the requirements for
the degree of Doctor of Philosophy in
Cosmology, Space Science & Space Technology.
Coordinator: Prof. Salvatore Capozziello.*



To my parents and my brother

Abstract

The new aerial mobility dimension enabled by Ingenuity has unlocked unprecedented potential for groundbreaking science investigations in astrobiology, geology and climate on Mars. The next generation of Martian rotorcraft will need advanced navigation capabilities to conduct long-range flights over diverse and challenging terrains. Accurate geo-localization within a global reference frame is essential to mitigate cumulative drift from on-board visual odometry, ensuring precise navigation during extended traverses. Absolute visual localization can be achieved in a map-based approach by matching real-time images from the rotorcraft's navigation camera with pre-referenced orbital maps stored on-board. However, significant illumination differences between query observations and on-board maps can challenge visual geo-localization performance, restricting the mission's operation envelope to certain times of day. This work investigates deep-learning-based methods to perform robust Map-based Localization (MbL) in challenging lighting. We proposed a novel multi-modal deep-learning framework that utilizes cross-attention mechanisms to fuse visual and depth data from ortho-projected maps and Digital Terrain Models (DTMs) obtained from orbital assets, effectively leveraging geometric context to learn illumination and scale invariance. To support training and validation, we developed a custom rendering framework to generate a synthetic Mars dataset with HiRISE ortho-images and DTMs, simulating aerial observations under varying lighting and altitude. Comprehensive evaluations show that our multi-modal method improves localization accuracy and robustness to a wide range of lighting offsets between maps and observations compared to single-modality models, including both deep-learning-based and traditional methods. Additionally, the integration of depth information has been shown to provide a degree of scale invariance and enhanced robustness to varying terrain morphology.

Acknowledgements

The true richness of my PhD experience lies not in the countless hours spent at my disordered and busy desks but in the incredible community of researchers, institutions, friends, and family who supported me throughout this complex, uncertain, and marvelous journey. This work would not have been possible without the Scuola Superiore Meridionale and NASA Jet Propulsion Laboratory, to which I express my deepest gratitude for nurturing my passion for space and always pushing me to dare mighty things.

First, I would like to thank my supervisor, Georgios Georgakis, who contributed to shaping me into the researcher I am today with his careful supervision and patient guidance during my time at JPL. His mentorship made me better equipped against the uncertainties that inevitably surround the questions of genuine scientific inquiry. Then, I would like to thank Prof. Salvatore Capozziello and my co-supervisors, Prof. Michele Grassi and Prof. Giancarmine Fasano, for supporting my visiting research periods and for recognizing them as fundamental opportunities for academic growth.

I extend my gratitude to Robert Hewitt for introducing me to the fascinating world of aerial robotics and for being a valued mentor. If I had the chance to contribute to groundbreaking work in rotorcraft autonomous navigation for Mars exploration, this is thanks to Roland Brockers who welcomed me into his team with trust and dedication. My experience on this project was profoundly enriched by my knowledgeable and brilliant teammates, from whom I learned so much: Gerik, Jeff, Tanguy, Luca, Lasse, Simone, Charles, Dario and Axel. I am also thankful to Ben Morrell, and Sheryar Kattak for the fruitful conversations and for the feedback on this work.

Spending most of my PhD working at JPL allowed me to meet some of the most humble, brilliant and supportive researchers. I would like to especially thank Saptarshi Bandyopadhyay for giving me the opportunity to join JPL during my first internship on the Lunar Crater Radio Telescope project and for his priceless advice as a mentor and friend. I will always be grateful to my other mentor and friend, Ashish Goel, whose guidance fostered the critical thinking and rigor needed by a scientist and who remains a constant source of inspiration. I am thankful to all my fellow JPL interns for having filled my research experience with culture and joy. Thanks to my fellow PhD students Giada, Matteo, Rasmi and Maria Grazia for helping me feel less alone as engineer in the intricate and charming world of cosmology. Special thanks to Marcello for having always supported me with his light and bright soul.

This journey would not have been the same without my closest friends, whose

presence and support have been a cornerstone of my PhD experience. To Francesco Musella for being the epic companion of this Odyssey set between Naples and Los Angeles, feeding my aspirations from beginning to end. To Francesco Flora, for making my first years at Scuola Superiore Meridionale so delightful and for celebrating life together with a Dionysian spirit. To Simone, whose lightheartedness has been a true blessing during difficult moments. To Flavio, for the unique and precious bond we built during my first summer at JPL. To Ivan, for being the wild, deep and sweet soul that increased my lust for life during my time at JPL. To Diana, for all the wisdom shared during our rich conversations at the JPL mall. To Carmine, for being the invaluable partner of the funniest adventures across at least seven states during my time at JPL and for sharing his joyful zest for life.

Finally, no words can adequately express the gratitude I owe to my family. To my parents, Maria Rosaria Zannini and Giuseppe Pisanti, their unwavering and unconditional support, especially during my time in California. To my brother, Fabrizio, for keeping on being the most important cardinal point in my life. To Veronica, for gifting me with being the kindest sister I could have ever asked for. To Zio Maurizio, for being the main responsible for my passion for space. To Nonna Yolanda, for her morning prayers and for her immense love. To Nonno Alberto, for who I am today.

Thank you!

New York City, December 2024

Contents

Abstract	v
Acknowledgements	vi
Preface	xvii
1 Introduction	1
1.1 Science enabled by future Mars rotorcraft	2
1.2 The Mars Science Helicopter	3
1.3 Long-range flight challenges on Mars: the Map-based Localization . . .	4
1.4 Contributions of this work	5
2 Related Work	7
2.1 Template matching	7
2.2 Hand-crafted features matching	9
2.3 Deep-learning-based Matching	10
2.3.1 2D-2D matching	10
2.3.2 3D-3D matching	11
2.3.3 Multi-modality	11
3 Methodology	13
3.1 Introduction to Transformer	13
3.2 LoFTR	16
3.2.1 Local Feature CNN	17
3.2.2 Coarse-Level Local Feature Transform	18
3.2.3 Coarse Matching Module	19
3.2.4 Coarse-to-Fine Module	20
3.3 Geo-LoFTR	20
3.4 LoFTR Models Supervision and Training	22
3.5 Map-based Localization Pipeline	23

4	Mars Synthetic Dataset	27
4.1	HiRISE Data	27
4.2	MARTIAN: Mars Aerial Rendering Tool for Imaging and Navigation	31
4.2.1	Terrain and texture modeling	31
4.2.2	Scene setup and camera modeling	33
4.2.3	Lighting	35
4.2.4	Rendering	36
4.3	Dataset generation	38
4.4	LoFTR training set	41
5	Map-based Localization Evaluation	47
5.1	Localization with Fixed Lighting Offset	48
5.2	Effect of Terrain Morphology on Localization Accuracy	52
5.3	Robustness to Changing Solar Angles	59
5.4	Localization Over a Simulated Martian Day	67
6	Conclusions and future work	73
6.1	Key findings	73
6.2	Future Directions	75

List of Figures

1.1	NASA’s Ingenuity helicopter viewed by the Perseverance rover’s rear Hazard Camera on April 4, 2021. Credit: NASA/JPL-Caltech	1
1.2	Mars Science Helicopter. Credit: NASA/JPL-Caltech	4
3.1	Transformer architecture diagram. The encoder is displayed on the left side of the diagram: the decoder is on the right side. Reproduced from Figure 1 of [1], with permission from Google for scholarly use.	14
3.2	Architecture of a dot-product attention layer (a) and a multi-head attention layer (b). Reproduced from Figure 2 of [1], with permission from Google for scholarly use.	15
3.3	LoFTR: architecture overview. Reproduced from Figure 2 of [2], licensed under CC BY 4.0.	17
3.4	Architecture of the LoFTR encoder (a) with a vanilla dot-product attention (a) and linear attention layer (b). Reproduced from Figure 3 of [2], licensed under CC BY 4.0.	18
3.5	Geo-LoFTR: architecture overview. Adapted from Figure 2 of [2], licensed under CC BY 4.0.	21
3.6	Geo-Merger architecture.	21
4.1	HiRISE ortho-rectified image product ESP_046060_1985_RED_C_01_ORTHO of the Jezero crater site, generated with 1 m / pixel resolution from the ESP_045994_1985 and ESP_046060_1985 stereo pairs.	30
4.2	Shading editor configuration for the MARTIAN framework.	33
4.3	Views of the Blender terrain for the Jezero Crater site in MARTIAN. . .	33
4.4	Depth precision computed from Eq. 4.2 for perspective cameras in Blender within the z_c range [0, 14,404] m (<i>left panel</i>) and [0, 200] m (<i>right panel</i>), for different distances n of the near clipping plane of the camera, with depth the buffer stored in 32 bit floating point format and the far clipping plane f at 100000 m.	37

4.5	Gray-scale images (top panel) and normalized depth image (bottom panel) of an orthographic map tile compared with three observations sampled from different locations under the same illumination condition and map/observation scale ratio equal to 1.00 (top query), 2.08 (middle query), 3.12 (bottom query). Generated with MARTIAN.	39
4.6	Gray scale images of a map tile from the Jezero crater site, rendered with different combinations of Sun Azimuth (AZ) and Elevation (EL). Generated with MARTIAN.	40
4.7	. Scale ratio $r_{\text{query/resized}}$ (from Eq. 4.7) between the map window (resized by LoFTR) and the query image, within the altitude range of the observations in the training set.	43
4.8	Examples of LoFTR training pairs formed by map windows with different lighting (<i>left</i>) and query observations (<i>right</i>) at 180° Sun azimuth (AZ) and 40° Sun elevation (EL). Maps are shown with 0° AZ and 30° EL (<i>top</i>), 60° EL (<i>mid</i>), 90° EL (<i>bottom</i>).	45
5.1	. Cumulative distribution of the localization error (<i>left</i>) and pixel reprojection error (<i>right</i>) for the MbL of 500 test observations of the Jezero Crater HiRISE map with different image matching models, and with consistent Sun AZ and EL offsets of 180° and 35° from the ortho-rectified map. Observations are generated in MARTIAN (nadir-pointing, altitude 64-200 m) with reference Sun AZ=180° and EL=40°.	49
5.2	. Matched keypoints between a sample query image (<i>left side of each panel</i>) and a map search area image (<i>right side of each panel</i>) for the Jezero Crater, with fixed Sun azimuth (AZ) and elevation (EL) angles offsets. Each panel shows matches using different image matching models: (a) Geo-LoFTR, (b) Fine-tuned LoFTR, (c) Pre-trained LoFTR, and (d) SIFT. The query observation was generated in the MARTIAN framework (nadir-pointing, altitude 162 m) with a reference Sun AZ of 180° and EL of 40°. The ortho-rectified map was generated with Sun AZ = 0° and EL = 5°.	50
5.3	. Cumulative distributions of the localization error for different matching methods used in the MbL of 500 nadir-pointing test observations within the altitude ranges [64, 112] m (<i>left</i>), [113, 157] m (<i>middle</i>) and [159-198] m (<i>right</i>). The observations were generated in the MARTIAN framework and sampled from the Jezero Crater HiRISE map, with reference Sun AZ=180° and EL=40°. The ortho-rectified map was generated with Sun AZ = 0° and EL = 5°.	51
5.4	Locations of the 200 sampled nadir-pointing observations (altitude 100 m) on rugged (<i>left</i>) and dunal (<i>right</i>) terrains from the Jezero Crater HiRISE map. The images are generated with the MARTIAN framework at a Sun azimuth of 180° and elevation 40°.	53

5.5	Sample nadir-pointing observations of rugged (<i>top rows</i>) and dunal (<i>bottom rows</i>) terrains on the Jezero Crater HiRISE map, at altitudes of 64 m (<i>left column</i>), 100 m (<i>middle column</i>) and 200 m (<i>right column</i>). The images are generated with the MARTIAN framework at Sun azimuth of 180° and elevation 40°	54
5.6	Cumulative distributions of the localization error for the MbL of 200 nadir-pointing observations (<i>each panel</i>) of rugged (<i>top row</i>) and dunal (<i>bottom row</i>) terrains on the Jezero Crater HiRISE map, at altitudes of 64 m (<i>left column</i>), 100 m (<i>middle column</i>) and 200 m (<i>right column</i>). Observations are generated in MARTIAN framework with Sun AZ= 180° and EL= 40° . The map is rendered at Sun AZ= 0° and EL= 5°	55
5.7	Variation of localization accuracy with altitude for the MbL of nadir-pointing observations over rugged (<i>top row</i>) and dunal (<i>bottom row</i>) terrains on the Jezero Crater HiRISE. The accuracy is addressed at the precision level of 0.5 m (<i>left column</i>), 1 m (<i>middle column</i>) and 2 m (<i>right column</i>). Observations are generated in MARTIAN framework with Sun AZ= 180° and EL= 40° . The map is rendered at Sun AZ= 0° and EL= 5°	56
5.8	Matched keypoints between observations (<i>left, in each panel</i>) of rugged terrain and map <i>right, in each panel</i>) for the Jezero Crater HiRISE map, at altitudes of 64 m (<i>left column</i>), 100 m (<i>middle column</i>) and 200 m (<i>right column</i>). The image matching models are: Geo-LoFTR (<i>1st row</i>), pre-trained LoFTR (<i>2nd row</i>), fine-tuned LoFTR (<i>3rd row</i>) and SIFT (<i>4th row</i>) Observations are generated in MARTIAN framework with Sun AZ= 180° and EL= 40° . The map is rendered at Sun AZ= 0° and EL= 5°	58
5.9	Matched keypoints between observations (<i>left, in each panel</i>) of dunal terrain and map <i>right, in each panel</i>) for the Jezero Crater HiRISE map, at altitudes of 64 m (<i>left column</i>), 100 m (<i>middle column</i>) and 200 m (<i>right column</i>). The image matching models are: Geo-LoFTR (<i>1st row</i>), pre-trained LoFTR (<i>2nd row</i>), fine-tuned LoFTR (<i>3rd row</i>) and SIFT (<i>4th row</i>) Observations are generated in MARTIAN framework with Sun AZ= 180° and EL= 40° . The map is rendered at Sun AZ= 0° and EL= 5°	59
5.10	Cumulative distributions of the localization error for the MbL of 500 nadir-pointing observations (altitude 64-200 m) at reference Sun AZ= 180° and EL= 40° , and maps at six different Sun elevation angles and fixed azimuth. Test observations and maps are generated from the Jezero Crater site in the MARTIAN framework.	60
5.11	Variation of localization accuracy with map Sun elevation, for the MbL of 500 nadir-pointing observations (altitude 64-200 m) at reference Sun EL= 40° (<i>dashed line</i>). The localization accuracy is shown at the precision level of 0.5m (a) , 1 m (b), 2 m (c) and 5 m (d). Queries and maps are generated from the Jezero Crater site with the MARTIAN framework, with a consistent Sun azimuth angle of 180°	61

- 5.12 Geo-LoFTR (*left column*) and SIFT (*right column*) matched keypoints displayed for a sample query image (*left side of each panel*) and a map search area image (*right side of each panel*) under four different Sun elevation angles and fixed azimuth. The query observation (nadir-pointing, altitude 142.67 m) is generated from the Jezero crater site in the MARTIAN framework with a reference Sun AZ of 180° and EL of 40° . Match lines are color-coded by confidence score, with red indicating higher confidence and blue indicating lower confidence. 62
- 5.13 Cumulative distributions of the localization error for the MbL of 500 nadir-pointing observations (altitude 64-200 m) at reference Sun AZ= 0° and EL= 10° , and maps at six different Sun azimuth angles and fixed elevation. Test observations and maps are generated from the Jezero crater site in the MARTIAN framework. 63
- 5.14 Variation of the localization accuracy with map Sun azimuth, for the MbL of 500 nadir-pointing observations (altitude 64-200 m) at reference Sun AZ= 0° (*dashed line*). The localization accuracy is shown at the precision level of 0.5m (a) , 1 m (b), 2 m (c) and 5 m (d). Queries and maps are generated from the Jezero Crater site with the MARTIAN framework, with a consistent Sun elevation angle of 10° 64
- 5.15 Geo-LoFTR (*left column*) and SIFT (*right column*) matched keypoints displayed for a sample query image (*left side of each panel*) and a map search area image (*right side of each panel*) under four different Sun azimuth angles and fixed elevation. The query observation (nadir-pointing, altitude 142.67 m) is generated from the Jezero crater site in the MARTIAN framework with a reference Sun AZ of 0° and EL of 10° . Match lines are color-coded by confidence score, with red indicating higher confidence and blue indicating lower confidence. 65
- 5.16 Localization accuracy at 1 m precision level as a function of map Sun elevation (*top*) and azimuth (*bottom*) for nadir-pointing observations across three altitude ranges. In the top row, accuracy is shown for maps with varying elevations at a fixed azimuth of 180° , with query observations set at AZ= 180° and EL= 40° . In the bottom row, maps vary in azimuth at a fixed elevation of 10° , with query observations at AZ= 0° and EL= 10° . Data are generated from Jezero Crater using the MARTIAN framework. 66
- 5.17 Sun trajectory on a local panorama from 77.44°E longitude and 18.43°N latitude on Mars, on 2031-05-10, with positions shown at four Local Mean Solar Times (LMSTs). Sun elevation varies in the range $[-90^\circ, 90^\circ]$ on the y-axis. The x-axis shows the sun azimuth measured clockwise from North, in the $[-180^\circ, 180^\circ]$ range. Adapted from Mars24 [3]. . . . 67
- 5.18 Sample nadir-pointing observations at different Local Mean Solar Times (LMSTs) taken on Mars on 2031-05-10, across three altitude ranges. The reference HiRISE map is taken at 15:00 LMST. The Sun Zenith is at 11:29 LMST. Data are generated from Jezero Crater using the MARTIAN framework. 69

5.19	Cumulative distributions of the localization error for the MbL of 500 nadir-pointing observations (altitude 64-200m) taken at six different Local Mean Solar Times (LMSTs) on Mars, on 2031-05-10. The reference HiRISE map is taken at 15:00 LMST (c). The Sun Zenith is at 11:29 LMST (b). Data are generated from Jezero Crater using the MARTIAN framework.	70
5.20	Localization accuracy at 1 m (<i>top</i>) and 2m (<i>bottom</i>) precision levels as a function of Local Mean Solar Time (LMST) of 500 nadir-pointing observations taken on Mars on 2031-05-10, across three altitude ranges. The reference HiRISE map is taken at 15:00 LMST (<i>dashed black line</i>). The Sun Zenith is at 11:29 LMST. Data are generated from Jezero Crater using the MARTIAN framework.	71

List of Tables

4.1	HiRISE Requirements and Performance Characteristics. Adapted from [4]	29
4.2	Comparison of the left and right HiRISE stereo images.	31
4.3	Main parameters of the Principled BSDF node in Blender for the synthetic Mars terrain.	32
4.4	Sun object Cycles Rendering Engine settings in MARTIAN.	35
4.5	Parameters for the maps and observations image dataset sourced for the generation of the pairs formed by query observations and map windows, used for LoFTR and Geo-LoFTR training.	46
5.1	Localization accuracy at the precision levels of 0.5 m, 1 m, 2 m and 5 m for different matching methods used in the MbL of 500 nadir-pointing test observations in three altitude ranges. The observations are generated in the MARTIAN framework and sampled from the Jezero Crater HiRISE map, with reference Sun AZ=180° and EL=40°. The map is generated with Sun AZ = 0° and EL = 5°.	52
5.2	Localization accuracy at the precision levels of 0.5 m, 1 m, 2 m and 5 m for different matching methods used in the MbL of 200 nadir-pointing test observations at altitudes of 64 m, 100 m and 200 m, for the rugged and dunal terrain morphologies. The observations are generated in the MARTIAN framework and sampled from the Jezero Crater HiRISE map, with reference Sun AZ=180° and EL=40°. The map is generated with Sun AZ = 0° and EL = 5°.	57
5.3	Specifications of points of interests on the solar local trajectory from (77.44°E, 18.43°N) coordinates on Mars, on 2031-05-10. Sun azimuth is measured clockwise from North in Mars24 framework; while it is defined counterclockwise from East in the MARTAIAN framework. Adapted from Mars24[3]	70

Preface

The research was carried out at the Jet Propulsion Laboratory, California Institute of Technology, under a contract with the National Aeronautics and Space Administration (80NM0018D0004).

1 Introduction

The demonstration flights of NASA's Mars Helicopter, *Ingenuity* (Figure 1.1), have marked a groundbreaking milestone in Mars exploration and scientific discovery by introducing the aerial mobility dimension[5].



Figure 1.1: NASA's Ingenuity helicopter viewed by the Perseverance rover's rear Hazard Camera on April 4, 2021. Credit: NASA/JPL-Caltech

The next generation of Mars rotorcraft will require even more sophisticated autonomous navigation capabilities to access diverse terrains under challenging environmental conditions and to support long-range, fully autonomous flights, with the ultimate goal of enabling high-priority investigations in Martian astrobiology, climate, and geology[6, 7]. One of the next evolutionary steps in advancing aerial mobility on the red planet is represented by the Mars Science Helicopter (MSH), a concept of a hexacopter with a payload capacity of up to 5 kg and lateral traverse capability over 10 km, conceived by the NASA Jet Propulsion Laboratory (JPL). To achieve precision during

long-range flights on complex terrains, it's crucial to minimize drift in position estimates generated by the on-board visual odometry in a GNSS-denied environment like Mars. This can be accomplished by registering images captured by the MSH navigation camera on geo-referenced orbital maps, such as those produced by the High Resolution Imaging Science Experiment (HiRISE) camera aboard the Mars Reconnaissance Orbiter. This geo-localization technique is referred to as *Map-based Localization* (MbL). The MbL performance can be significantly impacted by visual disparities between the navigation camera image and the pre-downloaded offline map, particularly due to differences in illumination that occur when flying at a different time of day than when the map was originally collected. This work investigates MbL pipelines designed to enhance robustness under challenging lighting scenarios by leveraging deep-learning image matching models. The research has been carried out at NASA JPL in the frame of the Long-Range Navigation (LORNA) project for MSH. In this chapter we will briefly discuss examples of compelling science that can be enabled by future Mars rotorcraft (Section 1.1), followed by an overview of the Mars Science Helicopter's design (Section 1.2. Section 1.3 then delves into the key research questions and challenges posed by long-range navigation, introducing the map-based localization problem. Finally, Section 1.4 outlines the contributions of this work and presents the overall structure of the thesis.

1.1 Science enabled by future Mars rotorcraft

Compared to present-day rovers, which typically drive three to five sols per week at no more than 100 m/sol, an aerial platform achieving 1km/flight at 100s of m altitude with < 1 flight per sol can significantly extend the geographic range of science investigations and increase the number of science targets that can be inspected in a single mission. Answering critical questions on the nature and the evolution of the early crustal processes on Mars requires measurements on a scale of 100s of km. For instance, measurements from orbit indicate distributed remanent magnetism at Lucus Planum[8]: a 100-km-scale geophysical mapping performed by a Mars hexacopter with a 5 kg payload capability would allow for resolving finer scale magnetic features compared to orbit measurements, thus improving the characterization of the crustal remanent magnetism and unveiling precious information on crust formation and evolution[6]. Another meaningful instance (included in [6]) of science enabled by extended long-range traversability is represented by clays and astrobiology investigations at Mawrth Vallis, an outflow channel located in northwest Arabia Terra and extending for ~640 km. Thousands of clay minerals-rich sites have been mapped from orbit in this region, indicating intense aqueous activity on the early Mars surface and subsurface [9, 10]. An aerial platform like MSH might perfectly fit in-situ sampling operations at multiple, hard-to-reach sites along the channels. Equipped with sampling tools and potentially a microdrill, MSH could locate, sample, and deliver materials to a lander for subsequent analysis onboard [6]. Other critical insights into the planet's climate history are offered by the deposits of water ice found in numerous high- and middle-latitude locations [11, 12, 13]. However, these water ice exposures may be inaccessible and spaced by large distances (10 to 100s of km), making their exploration impracticable for a present-day rover. If we suppose that two target sites are displaced

by 50 km, a rover located at one of the sites and driving at the best of its performance at 100 m/sol, 5 sols a week, would reach the other site in approximately 2 years. A Mars rotorcraft flying 1 km/flight, 5 sols a week, would take approximately 10 weeks, thus completing the mission within a single Martian season.

1.2 The Mars Science Helicopter

The MSH design (Figure 1.2) adopts a hexacopter configuration with 1.28 m-diameter rotors (approximately the same size as Ingenuity's rotors) and a total mass of ~ 31 kg to support a ~ 5 kg science payload, in contrast to Ingenuity's coaxial configuration and dry mass of 1.8 kg. MSH is designed for a maximum of ~ 7 minutes of hover time, with its range linearly increasing with decreasing hover time, up to a maximum range of ~ 14 km with 0 minutes of hover time [7]. The MSH will be deployed on the Martian surface with a novel Mid-Air Helicopter Delivery (MAHD) architecture [14], where the helicopter takes off mid-air from a jetpack, after separating from the aeroshell in which it is stored during the entry phase. This solution bypasses the utilization of a large Perseverance-class rover (like for Ingenuity), enabling the design of a larger rotor size, since the helicopter only needs to fit within the volume constraints posed by the aeroshell. To enable science missions such as the ones mentioned in Section 1.1 MSH's navigation capabilities must be provided with map-based localization pipelines and autonomous on-board safe landing site detection for precision navigation and landing in a wide variety of terrains. In contrast, Ingenuity's navigation system was designed to fly with a flat-terrain assumption and was not equipped with landing hazard detection nor absolute localization capabilities. To accommodate those advanced navigation tasks, the MSH's navigation suite encompasses a down-facing navigation camera capturing gray-scale images with 640×480 resolution and 110° horizontal FOV (similar to Ingenuity), a Sensoror STIM-300 IMU and a down-pointing single-beam Laser Range Finder (LRF). Navigation over complex 3D terrain is performed with a Range-Visual-Inertial-Odometry (Range-VIO) algorithm [15] that overcomes the scale unobservability in traditional VIO pipelines by exploiting the LRF measurement. The camera pose is estimated by Range-VIO, by tracking visual SLAM features in the images subsequently collected by the navigation camera. A Landing Site Detection (LSD) module has been developed for in-flight hazard detection to select safe landing in nominal operations or emergency circumstances [16, 17]. The LSD performs an on-board 3D reconstruction of the terrain using subsequent images captured by the navigation camera in a Structure-from-Motion (SfM) approach. The reconstructed terrain is then analyzed to identify safe landing sites based on the estimated slope of the terrain and the local surface roughness.



Figure 1.2: Mars Science Helicopter. Credit: NASA/JPL-Caltech

1.3 Long-range flight challenges on Mars: the Map-based Localization

Developed as a technology demonstrator, Ingenuity has been designed to perform short-range flights within a 300 m range and at 5 m altitude, with a maximum flight duration of 90 seconds. During Flight 9 on July 5, 2021, it traversed a horizontal distance of 625 m at a maximum altitude of 10 m and a duration time of 166.4 s, while on October 5, 2023, during Flight 61, Ingenuity achieved a maximum altitude of 24 m in a flight duration time of 19.5 s [18]. During its flights, Ingenuity used a VIO algorithm to perform a Visual Relative Navigation (VRL), which produces error accumulation (drift over time) as a byproduct of the recursive use of past pose estimations to provide new ones. During flights 6 to 12, the recorded horizontal traversed distances ranged from 106 m to 625 m at an ~ 10 m altitude, and the drift error was measured within a range of 1.4% to 6.2%, with the maximum error occurring during Flight 9, when Ingenuity flew over high-slope terrain. Its geo-localization in a global frame was performed post-flight with human intervention by manually registering the navigation camera images to a HiRISE map.

For MSH's long-range traverses within ~ 14 m and altitudes up to around 100 meters, the drift is expected to be considerably higher than that of Ingenuity. Relying on a purely visual-relative localization (VRL) approach is not a viable solution for fully autonomous flights. To mitigate this drift in the state estimator, onboard absolute localization needs to be performed online. On Earth, such localization on a global frame would typically employ GNSS (Global Navigation Satellite System) measurements. However, since no such system exists on Mars, a map-based localization strategy must be adopted by registering onboard images from the helicopter's navigation camera with orbital image products that are pre-registered to a global frame and downloaded onboard beforehand.

Map-based localization has been typically performed manually by humans on all Mars rover missions by using data products from HiRISE (High Resolution Imaging Science Experiment) and CTX (Context Camera) onboard the Mars Reconnaissance Orbiter (MRO), including ortho-projected maps and Digital Terrain Models (DTMs) reconstructed from stereo-vision. The resolution of the ortho-images is 0.25 m/pixel for HiRISE and 6 m/pixel for CTX. The first MbL technique successfully executed on the Perseverance Mars Rover with no human in the loop has been proposed in [19]. This employs a template matching approach to match rover images with HiRISE maps at sub-meter accuracy, achieving near-human localization performance. Fully autonomous MbL based on template matching has been performed by the Lander Vision System (LVS) for the Terrain Relative Navigation (TRN) during the Entry, Descend and Landing (EDL) phase of Perseverance over the Jezero Crater in the Mars2020 mission [20]. Both the techniques will be described in more detail in Chapter 2.

Typically, the map-based localization performance suffers most when the onboard imagery is taken under different lighting conditions compared to the orbital image references, which could have an impact on permissible flight lengths as drift accumulates over time or restrict flights to the times of day matching those of the orbital map. This work investigates deep learning-based MbL techniques that improve robustness to lighting variability by learning illumination-invariant image matching models, with the ultimate goal of expanding the mission’s operational envelope in terms of allowable ranges and times of day. To achieve that, we generated a large synthetic Mars dataset consisting of aerial observations and ortho-projected HiRISE-like maps rendered at different combinations of lighting to serve as training and test sets.

Another factor that may challenge MbL performance consists of a scale variance between the map and the query observation images, particularly relevant for the MSH, which needs to operate in a wide range of altitudes up to 100 m. Map-based localization can be impacted by variation in terrain morphology and textures. Complex, high-relief terrain may strengthen visual disparity between the query and map images at different times of day, due to changes in shadow casting. Conversely, textureless terrain can hinder the identification of distinctive features necessary for matching the map and image. These challenges have been carefully considered in generating the training set for the learning-based MbL pipeline and in planning our MbL evaluation.

1.4 Contributions of this work

This work aims to address the challenges of vision-based geo-localization on Mars by advancing the state of the art with:

- A novel pipeline that leverages a multi-modal, deep-learning architecture, fusing visual and depth data from HiRISE maps and DTMs to enable robust MbL under varying lighting conditions and scale differences.
- A custom rendering framework to generate a Mars synthetic dataset, including ortho-rectified maps and aerial observations of realistic Martian landscapes de-

rived from HiRISE DTMs and ortho-images.

- A comprehensive MbL evaluation showcasing the performance of the proposed deep-learning models on dedicated synthetic Mars datasets that span diverse lighting, scale, and terrain morphologies.

Beyond Mars aerial mobility, these contributions could significantly benefit absolute localization tasks in GNSS-denied environments and challenging illumination conditions across different mobility domains, including surface or space-based, for the autonomous robotic exploration of the Solar System and space operations. A key application could consist of advancing Terrain-Relative Navigation during the Entry, Descent, and Landing (EDL) phases of planetary missions. For instance, future missions to the Moon's south pole will face significant visual navigation challenges at the landing stage due to the constantly low Sun elevation resulting in deep shadows and poor lighting, making accurate absolute localization difficult. Similarly, this research can contribute to the autonomous navigation around small celestial bodies, where rapidly shifting lighting alters surface topography, complicating visual perception. These challenges are also relevant to space operations tasks, such as in-orbit servicing or active debris removal, where robust relative navigation around the target artificial bodies may account for a diverse set of illumination conditions. Finally, the research presented in this work shows also meaningful implications for Earth-based operations, such as in scenarios where autonomous Unmanned Aerial Vehicles (UAVs) might be deployed for critical research and rescue missions in damaged or confined indoor environments with poor lighting conditions and no access to GNSS.

The structure of this thesis is as follows: Chapter 2 provides an overview of related work in MbL and image matching techniques. Chapter 3 details the methodology, followed by an in-depth discussion of the Mars synthetic dataset generation within the custom rendering framework in Chapter 4. Chapter 5 presents and analyzes the results of the MbL experiments across various synthetic test sets. Finally, Chapter 6 concludes the thesis and suggests directions for future work.

2 Related Work

Map-based localization belongs to the broader set of Absolute Visual Localization (AVL) techniques designed to determine the position of a UAV in a geo-referenced map. Unlike relative visual localization, which relies on a recursive pose estimation based on past ones and can thus suffer from location error drift over time, AVL enables independent localization by matching a single image captured in real-time with a pre-existing map, making it inherently drift-free. Map-based localization typically consists of matching a query image against a map image, identifying distinctive features or landmarks in both the real-time and reference images, enabling the MbL pipeline to correlate the current view to the corresponding map area, and pinpoint its precise location. However, 2D-2D image-based matching represents only one approach within the broader set of techniques employed in MbL. Similarly, a 2D-3D matching strategy can be used to match the query image on a prior digital terrain model of the map. If also a 3D query model of the terrain is acquired or reconstructed from the UAV's sensor suite, a 3D-3D matching can be performed as well. Recent approaches in MbL increasingly leverage deep learning techniques that enhance robustness to image variance, including methods that benefit from data multi-modality. Extensive reviews on AVL for UAVs are provided in [21, 22]. A more recent review focusing on deep learning for UAV's AVL is included in [23]. The following sections present an overview of MbL approaches that are relevant to the MbL of the MSH, including traditional template-matching methods, hand-crafted feature descriptors, and advanced learning-based models.

2.1 Template matching

Template matching is a technique that consists of comparing a given "template" image against different regions within a target image to find the best match. It works by optimizing similarity measures, such as correlation or normalized cross-correlation, between the template and patches of the target image.

The MARS2020 Lander Vision System integrated a template matching approach in their terrain relative navigation pipeline to perform precise absolute localization during the mission's EDL phase [20]. The LVS was able to successfully localize the lander by fusing landmarks matched between descent imagery and CTX map (6 m/pixel) of the Jezero Crater landing site with inertial measurement unit (IMU) data. The system was

able to reduce the initial 3.2 km position uncertainty at 4200 m altitude to just 40 m at 500 m altitude in 10 s, contributing to an overall targeted landing error margin of 5 m. The LVS pipeline adopted a coarse-to-fine strategy. In the initial coarse stage occurring at 4200 m above ground level (AGL), five large patches in each of three descent images are matched to the onboard map binned to 12 m/pixel, resulting in a position error of 200 m. The fine stage refines this error down to 40 m by matching 150 small patches from each descent image to the full-resolution map of 6 m/pixel. At the coarse-level, the template matching is achieved with a frequency domain correlation algorithm, where the Fast Fourier Transform (FFT) of both templates and map images was used to perform correlation. In the fine-level stage, a spatial correlation algorithm is used due to higher efficiency in matching small search areas.

Another example of successfully applied template matching for MbL on Mars is represented by the Censible framework proposed in [19] and used for absolute localization of Perseverance. The method consists of registering an ortho-mosaic of panoramic stereo images collected by the rover to a HiRISE map. The template matching relies on a modified census transform, a non-parametric local image transform that is invariant to monotonic variations in intensity, applied on both the template and the orbital map. The Censible framework performance has been benchmarked on 264 panoramic images from Perseverance, achieving 0.36 m accuracy. However, despite being collected throughout all the Martian seasons, the majority of images were taken in the afternoon, relatively close to the time of day when HiRISE maps are typically captured, thus excluding challenging lighting corresponding to low Sun elevation angles (e.g., early morning and sunset). To address this limitation, we generated synthetic Mars image test sets to evaluate our MbL pipeline also under these extreme illumination scenarios.

Illumination invariance with template matching in the frequency domain has been investigated in [24], where evidence of illumination-invariant properties of the Phase Correlation to changing lighting has been found via mathematical derivation and experiments. Phase Correlation (PC) is a template matching algorithm based on Fourier shift properties, according to which a translation shift between two similar images produces a linear phase difference in the Fourier frequency domain. The shift can be retrieved by first computing the Fourier Transform (FT) of the two images and then applying an Inverse Fourier Transform (IFT). A PC-based localization pipeline has been used in [24] to match rescaled HiRISE images simulating UAV observations to a HiRISE map taken with a 56° offset in Sun azimuth and 5° in Sun elevation. Their pipeline achieved an overall localization accuracy of 0.35 m, outperforming other template matching methods such as Normalized Cross Correlation (NCC) and Mutual Information (MI) techniques. However, the Sun elevation difference is too small to observe significant visual disparity between the images, and the reference elevation of 50° does not allow the azimuth offset to exhibit significant changes in the shadows' pattern. Moreover, the PC algorithm requires the query image and the map to be largely overlapped (usually more than 1/4 image size).

2.2 Hand-crafted features matching

Feature keypoints represent the most common visual representations for image matching in vision-based tasks. These are generally identified in high-contrast regions of the image, such as corners, edges or blobs. Hand-crafted feature-matching methods can leverage feature detectors and descriptors that are manually designed to identify features with properties invariant to scale, rotation or lighting, in contrast to learning-based feature matching, where these properties are learned by deep learning models. A detector's role is to identify keypoints where distinct patterns in their neighborhood are more likely to be matched across different images. A popular example is the Harris corner detector [25] which located corners based on intensity gradients. Feature descriptors capture and encode the visual information (such as texture, gradient, color, etc.) around each detected keypoint into a compact representation, making it easier to compare between images. Each descriptor is associated with a distance metric (like Euclidean distance or Hamming distance) that defines how closely two descriptors match. Smaller distances according to these metrics suggest that their corresponding keypoints likely represent the same part of a scene or object across images. One of the most popular descriptors is the Scale-Invariant Feature Transform (SIFT) [26], which encodes the orientation and magnitude of image gradients around keypoints detected with a Difference-of-Gaussian (DoG) function, making it highly robust to changes in scale and rotation. ORB (Oriented FAST and Rotated BRIEF) [27] combines a FAST detector [28] with a BRIEF descriptor [29] adapted to achieve rotation invariance. The SURF [30] method uses a Hessian matrix-based detector and a distribution-based descriptor to accelerate feature computation compared to SIFT, while keeping scale and rotation-invariance. [31] proposed an on-board MbL pipeline employing 2D-2D feature-based matching for the absolute localization of the Mars Science Helicopter over a HiRISE map. The map is generated from an ortho-projected image and its corresponding digital terrain model and saved in a database containing feature descriptors and their associated 3D positions in a world reference frame. During flight, the features detected in navigation camera images are matched against the ones in the stored map. The matching process is informed by pose prior uncertainty from the on-board state estimator. An outlier rejection step is applied to the matched keypoints, followed by a Perspective-n-Point (PnP) algorithm that computes the camera pose and thus the drone location in the world frame using the 3D locations of the features stored in the map. The MbL performance was evaluated for SIFT, SURF, ORB and SuperPoint (a deep-learning based approach [32]) on a synthetic Mars dataset sourced from a HiRISE ortho-image of the Jezero Crater landing site and generated using AirSim [33], which uses Unreal Engine 4 as a rendering engine. SIFT has demonstrated superior localization accuracy compared to all the other methods when tested across a range of Sun elevation offsets between observations and the reference map at 90° of elevation. However, its performance significantly declines at a -60° elevation offset, highlighting the need for matching techniques that can better handle challenging lighting variations.

2.3 Deep-learning-based Matching

Deep learning has transformed feature matching by using deep neural networks to generate highly discriminative and robust feature representations. Learning-based methods have surpassed traditional computer vision approaches and now represent the state of the art in modern absolute visual localization for UAVs [23].

Unlike traditional, handcrafted methods that rely on manually designed feature detectors and descriptors, learning-based methods leverage Convolutional Neural Networks (CNNs), as well as architectures such as Transformer originally developed for natural language processing and adapted to computer vision tasks (e.g., Vision Transformer [34]), which can learn feature hierarchies directly from image data. CNNs are particularly effective for extracting features like edges and textures across multiple abstraction levels, making them resilient to changes in lighting and perspective. Transformers add further robustness and distinctiveness by using attention mechanisms to provide features with contextual information and long-range dependencies from the entire image. When trained on large, diverse datasets, both CNNs and Transformers generalize effectively, which enhances AVL performance in challenging, real-world environments. In this section, we categorize deep-learning-based matching approaches relevant to our MbL task into three primary types based on the data modality used: (1) 2D-2D matching, (2) 3D-3D matching, and (3) multi-modality matching.

2.3.1 2D-2D matching

Popular 2D-2D image feature matching techniques are represented by SuperPoint [32] and SuperGlue [35]. SuperPoint is a self-supervised method that uses fully CNNs to generate both keypoints and the corresponding descriptors. SuperGlue uses the features identified by SuperPoint, but focuses on improving keypoints matching using a Graph Neural Network (GNN) with attention mechanisms to learn spatial relationships within and cross-images keypoints. During the MbL evaluation performed by [31] on Mars simulated data, SuperPoint performs slightly worse than handcrafted methods in nominal under relatively low sun elevation offsets between map and observations. However, it performs slightly better than SIFT for drastic illumination changes, showing overall more robustness to lighting than handcrafted feature descriptors.

Among the methods aiming to learn robust image matching under real-world changes in scale, illumination, viewpoint and texture, RoMA [36] offers a coarse-to-fine framework leveraging frozen features from large self-supervised models like DINOv2 [37] to predict features at coarse resolution, combined with CNN-derived features for matching at a finer scale. This approach creates a feature pyramid that is both robust to diverse visual conditions and precise in localization. A popular 2D-2D matching method is the Local Feature TRansformer (LoFTR). LoFTR is a detector-free model for local image feature matching that combines CNNs and Transformers to generate dense pixel-wise correspondences. This method employs a coarse-to-fine strategy, matching features at a coarse level and refining them to sub-pixel accuracy, with the Transformers exploiting the pixels' contextual information to ensure robust performance in low-texture regions.

In this work, we incorporated LoFTR in our MbL pipeline for the localization of the MSH and we fine-tuned it on synthetic Mars datasets. Additionally, we extended its architecture with a novel multi-modal module using cross-attention mechanisms to fuse 3D data from HiRISE digital terrain model with the 2D orbital map, with the aim to improve accuracy and robustness under challenging illumination conditions offsets between observations and maps.

2.3.2 3D-3D matching

3D-3D matching consists of establishing correspondences between different 3D datasets, such as point clouds or voxel grids. Unlike 2D-2D matching, these techniques process spatial data directly, improving robustness to variations in viewpoint, scale, and orientation. Deep-learning-based 3D-3D matching techniques leverage geometric information to learn distinctive and invariant features from 3D data. Meaningful examples are offered by 3DMatch [38], FCGF (Fully Convolutional Geometric Features) [39] and Predator [40] methods. 3DMatch produces feature descriptors from 3D volumetric data by training a CNN on large datasets of 3D scenes. This method effectively learns features that are rotation-invariant and suitable for finding correspondences between point clouds from different viewpoints. FCGF uses sparse 3D convolutions to produce highly distinctive descriptors for each point in a 3D point cloud, allowing for fast and accurate matching across different 3D scans. By learning features from the raw geometric structure of 3D data, FCGF demonstrates high robustness in large-scale environments. Predator performs pairwise point cloud registration by learning how to detect overlapping regions between two unregistered scans and focusing on those areas to identify reliable feature keypoints.

In the context of MbL for navigation on Mars, 3D-3D matching can be adopted in the cases where a 3D model of the observed scene can be reconstructed or sensed in real-time, and thus matched to the offline digital terrain model available from orbital assets. However, aerial platforms such as MSH might face significant constraints in payload capacity, power consumption and computational resources, which may hinder the ability to perform accurate 3D reconstruction during flight.

2.3.3 Multi-modality

Integrating data from multiple sources can significantly enhance MbL accuracy by introducing additional constraints that help resolve ambiguities in identifying correspondences. Multi-modal matching aims to register 2D observations onto maps by leveraging 3D data or other sensor modalities. By properly integrating information from diverse data types, feature representations from one modality complement and enrich the other, adding robustness to the matching process and ultimately improving localization accuracy. For instance, 3D information contained in depth maps, digital elevation models or point clouds could be effectively used to add geometric context to feature descriptors coding visual information and pixel relationships in an RGB image. The Geometry-Aided Matching (GAM) framework [41] is one of the techniques designed to perform 2D-3D matching by combining both appearance-based and geometric cues. GAM operates in

two main steps. First, it applies k-nearest neighbors (kNN) ratio tests to identify candidate matches between 2D features in the query image and 3D features from a Structure-from-Motion (SfM) model. Second, it applies a deep neural network that can leverage the geometric information in the 3D data to filter out outliers out of the candidate match set, based on the observation that some matches might be ambiguous in the visual feature space but distinct in geometric context. 2D-3D matching counting solely on geometric information, rather than visual descriptors, is performed by GoMatch [42]. To bridge the gap between the 2D and 3D data modalities, this neural network architecture converts the query image keypoints into bearing vectors representing their direction relative to the query camera. The same representation is used for the 3D points of the input point cloud, thus unifying the data types into a single modality. The bearing vectors are then encoded in vector representations that undergo self-attention and cross-attention mechanisms to exchange information between them and learn context-rich representations.

The geometric context is used in GAM to learn a more accurate outlier rejection process to apply to pre-acquired candidate matches, rather than learning feature representations effectively fusing the data modalities. On the other hand, GoMatch recasts the 2D-3D cross-modality to an intra-modal matching framework by means of bearing vectors. However, it may lose distinctive information contained in the visual representation of the features, useful for feature matching in scenes with poor geometric context (e.g., flat terrains). Our pipeline aims to retain both the visual and geometric information from the ortho-image and depth maps computed from the associated terrain model by learning a shared representation to provide more reliable matches.

Finally, an example of UAV long-range absolute localization with different sensor modalities can be found in the thermal geo-localization framework proposed in [43]. Thermal geo-localization (TG) for UAVs in Earth applications, particularly in nighttime or poor lighting conditions, is emerging as a vital technology for navigation and localization in GNSS-denied or GNSS-spoofing environments. Unlike vision-based geo-localization, TG relies on aligning infrared images captured by on-board thermal sensors to satellite RGB data. In addition to proposing a novel learning-based technique to perform TG in UAV long-range navigation, [43] is one of the first studies to integrate domain adaptation methods to compensate for the degraded performance due to the limited availability of paired thermal and satellite images.

3 Methodology

Our map-based localization pipeline performs image registration of gray-scale aerial query observations onto HiRISE-like ortho-projected maps employing a LoFTR-based matching model. One model is obtained by fine-tuning LoFTR on a synthetic Mars dataset generated from HiRISE ortho-images and digital terrain models in a custom rendering framework (see Chapter 4). A novel model is then generated by extending the vanilla LoFTR architecture to incorporate depth information from HiRISE maps as an additional data modality to the gray-scale imagery used as input. This new architecture utilizes a cross-attention mechanism to form shared feature embeddings where both depth and visual data can be effectively integrated. Our approach is intended to enhance the matching accuracy by leveraging the geometric information included in the depth data to resolve ambiguities in the visual representation, such as those caused by lighting variations. We refer to this geometry-aided version of LoFTR as *Geo-LoFTR*.

In the following sections, we will first provide a brief introduction to the Transformer (Section 3.1), followed by an overview of the LoFTR method (Section 3.2). Then, Section 3.3 illustrates the Geo-LoFTR architecture and implementation. Finally, Section 3.5 describes how both methods are integrated within our MbL pipeline.

3.1 Introduction to Transformer

Transformer is a type of deep learning architecture first introduced by [1] for Natural Language Processing (NLP) tasks and then increasingly applied to the computer vision domain. Their core functionality consists in using self-attention and cross-attention mechanisms to dynamically weight the relative importance of each unit of an input data sequence (also denoted as *tokens*) to other elements of the same (*self-attention*) or different (*cross-attention*) input data. Examples of tokens can be the words of an input sentence in a NLP task, or image features extracted by a CNN backbone in computer vision applications. Unlike other traditional sequence models in NLP, such as Recurrent Neural Networks (RNNs), which process the data in a sequential manner, Transformer operates on the entire input simultaneously. This approach allows the model to learn how to provide the input data with context-rich representations that effectively capture long-range dependencies within and across data.

The original transformer architecture proposed in [1] is shown in Figure 3.1, with the

encoder displayed on the left side of the diagram and the decoder on the right side.

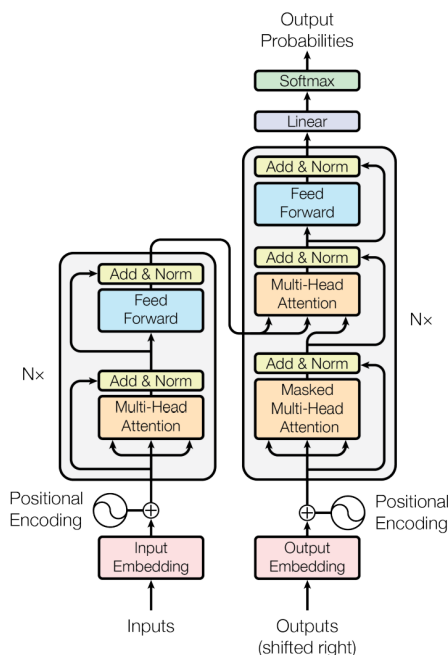


Figure 3.1: Transformer architecture diagram. The encoder is displayed on the left side of the diagram: the decoder is on the right side. Reproduced from Figure 1 of [1], with permission from Google for scholarly use.

Before feeding the encoder's stack, the tokens of an input sequence are first converted into embeddings, i.e., vectors representing their semantic properties. Since each token is processed in parallel by Transformer, positional encoding is added to the embeddings to provide them with positional context in the input sequence. The token embeddings are then passed to a stack of a number of identical encoder layers, where each layer is made of two main sub-layers: the attention layer and a Feed-Forward Neural Network (FFN). The former implements the self-attention mechanism responsible for learning a context-rich representation of the input sequence. The latter is a position-wise fully connected neural network that applies non-linear transformations to each token's embedding by capturing more complex relationships across the input sequence. These sub-layers also incorporate layer normalization to stabilize the learning process and residual connections, which help prevent vanishing gradients during learning by adding the sub-layer input to its output. The encoder output is then passed to the decoder, made of three main components: a masked self-attention layer, an encoder-decoder attention layer and a FFN. The first ensures that each output token is generated based only on previously generated ones. The second layer performs attention over the output of the encoder stack, helping the decoder to incorporate relevant context from the input at each decoding step. Like in

the encoder, the FNN further refines the output representation and finally delivers the processed sequence in the target domain.

To delve more into the details of the attention mechanism, we will refer to the dot-product attention layer (Figure 3.2a), which is typically employed in Transformer.

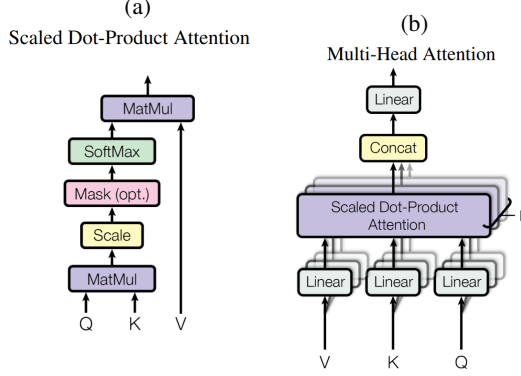


Figure 3.2: Architecture of a dot-product attention layer (a) and a multi-head attention layer (b). Reproduced from Figure 2 of [1], with permission from Google for scholarly use.

Let us consider the input sequence $X \in \mathbb{R}^{N \times D}$, made of N token embedding vectors of dimension D . The input sequence is projected into three distinct representation subspaces: the Query Q , the Key K , and the Value V computed as:

$$Q = XW_Q \in \mathbb{R}^{N \times D_k}, \quad K = XW_K \in \mathbb{R}^{N \times D_k}, \quad V = XW_V \in \mathbb{R}^{N \times D_v}, \quad (3.1)$$

where $W_Q, W_K \in \mathbb{R}^{D \times D_k}$ and $W_V \in \mathbb{R}^{D \times D_v}$ are learnable weight matrices. Each row q_i of Q is a query vector of dimension D_k , representing a token's embedding in the query space. The same applies for the k_i vectors of K in the Key space of the same dimensionality as the Query, while each row v_i of V is a value vector of dimension D_v . In the attention mechanism, each query vector q_i (representing a *question* about the token i to other tokens) is compared with each key vector k_j (representing possible *answers*) by taking the dot-product $q_i k_j^T$. This results in similarity scores measuring the relevance of each key to the query. The dot-products are scaled by $\sqrt{D_k}$ and passed to *softmax* function to get the *attention weights*, which are used to scale the corresponding value vector v_j . For each query vector q_i , the scaled value vectors are summed to produce a context vector that incorporates relevant information from all the tokens in relation to q_i . Since the attention function is computed simultaneously on all the N query vectors of Q , the output of the attention mechanism is given in matrix form, using the notation in [1]:

$$Attention(Q, K, V) = softmax(\frac{QK^T}{\sqrt{D_k}})V \quad (3.2)$$

In the encoder proposed in [1], all the sub-layers and the embedding layers produce outputs with the same dimensions as the input, D , to facilitate the residual connections. If we were using a single-attention head, we would have had $D_k = D_v = D$. However, to enhance the model's capability to capture complex relationships within the input data, Transformer employs a multi-head attention layer. Instead of applying the attention layer on the D -dimensional queries, keys and values vectors, this mechanism executes the attention function in Eq. 3.2 on h different learnable linear projections of the same vectors on subspaces of dimensions d_k , d_k and d_v , respectively, with h being the number of attention heads. In [1], $d_k = d_v = D/h$, where also in this case D is the dimension of both inputs and outputs. The output of the i -th attention head is given by:

$$head^i = Attention(QW_Q^i, KW_K^i, VW_V^i) \quad (3.3)$$

where the matrices $W_Q^i \in \mathbb{R}^{D \times d_k}$, $W_K^i \in \mathbb{R}^{D \times d_k}$ and $W_V^i \in \mathbb{R}^{D \times d_v}$ are the learnable projections. The outputs of the attention heads are then concatenated and projected again with the matrix $W_O \in \mathbb{R}^{d_v \times D}$, producing the multi-head attention output:

$$MultiHead(Q, K, V) = Concat(head^1, ..., head^h)W_O \quad (3.4)$$

In conclusion, the multi-head attention approach enables the model to learn context-rich representations of input data sequences by simultaneously inspecting relationships between tokens at multiple locations in the sequence and over different representation subspaces, all at nearly the same computational cost as a single-head attention layer, due to the reduced dimension of each head.

3.2 LoFTR

LoFTR is made of four different components: the Local Feature CNN, the Coarse-Level Local Feature Transform, the Matching Module and the Coarse-to-Fine Module. An overview of the LoFTR architecture is presented in Figure 3.3, reproduced from [2]. A description of each component will be provided in the following sections.

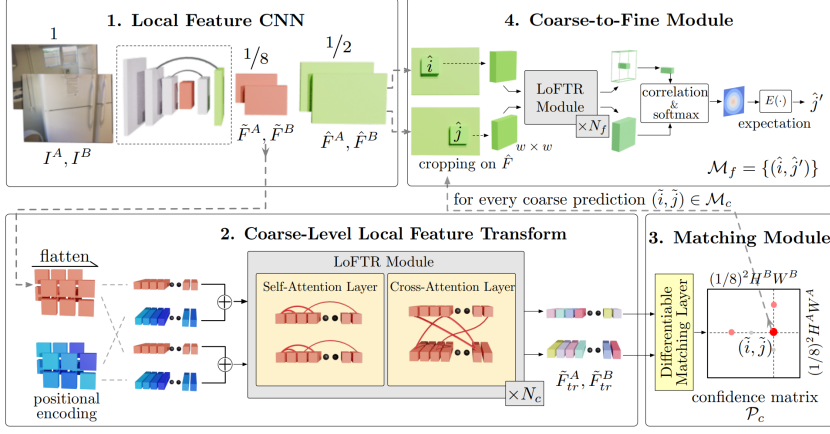


Figure 3.3: LoFTR: architecture overview. Reproduced from Figure 2 of [2], licensed under [CC BY 4.0](#).

3.2.1 Local Feature CNN

Given the image pair I^A and I^B of size $W^A \times H^A$ and $W^B \times H^B$, the first module uses a modified version of ResNet-18[44] as a CNN backbone to extract local features at 1/8 (coarse) and 1/2 (fine) scales of the original images' size. These constitute the coarse-level feature maps $\tilde{F}^A \in \mathbb{R}^{\tilde{W}^A \times \tilde{H}^A \times \tilde{D}}$ and $\tilde{F}^B \in \mathbb{R}^{\tilde{W}^B \times \tilde{H}^B \times \tilde{D}}$ for both the images, and the fine-level ones, $\hat{F}^A \in \mathbb{R}^{\hat{W}^A \times \hat{H}^A \times \hat{D}}$ and $\hat{F}^B \in \mathbb{R}^{\hat{W}^B \times \hat{H}^B \times \hat{D}}$, respectively, where:

$$\begin{aligned} (\tilde{W}^A, \tilde{H}^A) &= 1/8(W^A, H^A), & (\tilde{W}^B, \tilde{H}^B) &= 1/8(W^B, H^B) \\ (\hat{W}^A, \hat{H}^A) &= 1/2(W^A, H^A), & (\hat{W}^B, \hat{H}^B) &= 1/2(W^B, H^B) \end{aligned}$$

The number of channels of the coarse and fine feature maps is set to $\tilde{D} = 256$ and $\hat{D} = 128$ respectively.

In the context of the map-based localization I^A corresponds to the query observation from the MSH's navigation camera and I^B can be a window cropped from a HiRSE ortho-projected map. Further details on the required image pre-processing for query and map windows images will be provided in Section 3.5.

3.2.2 Coarse-Level Local Feature Transform

In the second component of the architecture, the coarse feature maps $\tilde{F}^A, \tilde{F}^B$ are flattened in $\tilde{s}^A$ and $\tilde{s}^B$ $\tilde{D}$ -dimensional vectors, respectively, which are equipped with a positional encoding, forming the flattened input feature maps $\tilde{f}^A \in \mathbb{R}^{\tilde{s}^A \times \tilde{D}}, \tilde{f}^B \in \mathbb{R}^{\tilde{s}^B \times \tilde{D}}$ where $\tilde{s}^A = \tilde{W}^A \tilde{H}^A$ and $\tilde{s}^B = \tilde{W}^B \tilde{H}^B$. The encoding is responsible for providing each feature with unique position information in the image space, in the sinusoidal format, ensuring the transformed features are position-dependent. This allows LoFTR to learn to produce reliable matches also in indistinctive regions. The flattened and encoded feature maps are then processed by the LoFTR module, which employs Transformer to extract position- and context-dependent local features. The input feature maps can be seen as a sequence of data, where each token is represented by a $\tilde{D}$ -dimensional feature vector extracted by a CNN backbone at given pixel coordinates within the coarse-resolution image. The sequence length corresponds to the total number of pixels in the scaled image.

The Transformer architecture adopted in LoFTR is composed of just a sequence of connected encoder layers (Figure 3.4a) with multi-head attention.

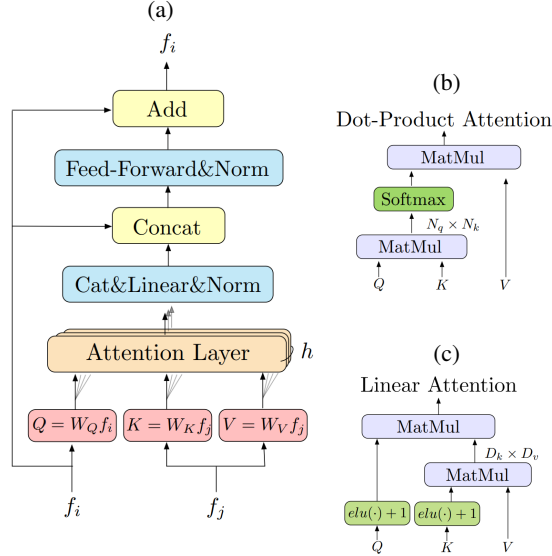


Figure 3.4: Architecture of the LoFTR encoder (a) with a vanilla dot-product attention (a) and linear attention layer (b). Reproduced from Figure 3 of [2], licensed under [CC BY 4.0](#).

The LoFTR module is formed by a sequence of N_c sub-modules, each consisting of a stack of a self-attention and a cross-attention encoder. Each encoder takes two input feature maps $f_i, f_j \in \mathbb{R}^{N \times \tilde{D}}$, which can both be either $\tilde{f}^A$ or $\tilde{f}^B$ for the first encoder (with N being either $\tilde{s}^A$ or $\tilde{s}^B$), and outputs a transformed feature map f_i of the same size as the input. The query, key and value matrices are defined by:

$$Q = f_i W_Q \in \mathbb{R}^{N \times D_k}, \quad K = f_j W_K \in \mathbb{R}^{N \times D_k}, \quad V = f_j W_V \in \mathbb{R}^{N \times D_v}, \quad (3.5)$$

where D_k, D_k, D_v are the dimensions of the queries, keys and values representation subspaces, respectively.

For a self-attention encoder we have $f_i = f_j$, otherwise, the cross-attention encoder applies. In each LoFTR encoder, the attention layer (3.4c) is a modified version of the dot-product attention. The dot product QK^T in a vanilla dot-product attention layer (Figure 3.4b) would introduce computation cost that grows quadratically with N , being prohibitively expensive even for coarse scaling from 640×480 images. The soft-max based-attention can be replaced by a similarity function such as:

$$\text{sim}(Q, K) = \phi(Q) \cdot \phi(K)^T \quad (3.6)$$

where $\phi(\cdot) = \text{elu}(\cdot) + 1$. This kernel applies the Exponential Linear Unit (ELU) activation shifted by 1 to ensure positive output. By leveraging the associativity property of matrix multiplication, it is possible to perform the matrix multiplication $\phi(K)^T \cdot \phi(V)$ first, with a cost complexity of $O(D_k \times D_v)$. Since usually $D_k \times D_v \ll N^2$, the $O(N^2)$ cost complexity of the vanilla dot-product attention is brought down to $O(N)$ by the linear attention layer. The outputs of the LoFTR-module are the transformed feature maps $\tilde{f}_{ir}^A$ and $\tilde{f}_{ir}^B$, which now leverage context-rich representations to ensure better feature matching compared to the input maps $\tilde{f}^A$ and $\tilde{f}^B$ produced solely by the CNN backbone.

3.2.3 Coarse Matching Module

In this module, the transformed features are matched by a differentiable matching layer. Given two features $\tilde{f}_{ir}^A(i)$ and $\tilde{f}_{ir}^B(j)$ at locations i and j in the corresponding transformed maps, a similarity score S is first calculated by:

$$S(i, j) = \frac{1}{D} \langle \tilde{f}_{ir}^A(i) \tilde{f}_{ir}^B(j) \rangle. \quad (3.7)$$

Then, a softmax is applied to both the i and j dimensions of S to obtain the probability of soft mutual nearest neighbor matching:

$$\mathcal{P}_c(i, j) = \text{softmax}(S(i, \cdot))_j \cdot \text{softmax}(S(\cdot, j))_i \quad (3.8)$$

where $\mathcal{P}_c$ is the matches *confidence matrix*. Matches with a confidence higher than a threshold of θ_c and passing a mutual nearest neighbor (MNN) criterion form the set $\mathcal{M}_c$ of the coarse-level match predictions:

$$\mathcal{M}_c = \{(\tilde{i}, \tilde{j}) \mid \forall(\tilde{i}, \tilde{j}) \in \text{MNN}(\mathcal{P}_c), \mathcal{P}_c(\tilde{i}, \tilde{j}) \geq \theta_c\} \quad (3.9)$$

3.2.4 Coarse-to-Fine Module

In this model, the coarse matches are refined to the images original resolution with a correlation-based approach. The process begins by unfolding the fine feature maps $\widehat{F}^A$ and $\widehat{F}^B$ in two sets of $w \times w$ map windows, centered at locations aligned with the coarse feature maps $\widetilde{F}^A$ and $\widetilde{F}^B$. The windows' sets are arranged in matrices of cropped fine features: $\widehat{f}^A \in \mathbb{R}^{\widetilde{s}^A \times (ww) \times \widehat{D}}$ for $\widehat{F}^A$ and $\widehat{f}^B \in \mathbb{R}^{\widetilde{s}^B \times (ww) \times \widehat{D}}$ for $\widehat{F}^B$. Each window feature map $\widehat{f}^A(i) \in \mathbb{R}^{(ww) \times \widehat{D}}$ and $\widehat{f}^B(j) \in \mathbb{R}^{(ww) \times \widehat{D}}$ can be accessed at locations i and j along the same dimensions $\widetilde{s}^A$ and $\widetilde{s}^B$, respectively, of the transformed coarse feature maps $\widehat{f}_{tr}^A$ and $\widehat{f}_{tr}^B$. For each coarse match $(\widetilde{i}, \widetilde{j})$ in the $\mathcal{M}_c$ set, the corresponding window feature maps $\widehat{f}^A(\widehat{i})$ and $\widehat{f}^B(\widehat{j})$ are located at coordinates $(\widehat{i}, \widehat{j})$. A LoFTR module transforms each window feature map by $N_f < N_c$ times, producing two transformed maps $\widehat{f}_{tr}^A(\widehat{i})$ and $\widehat{f}_{tr}^B(\widehat{j})$. The center vector of $\widehat{f}_{tr}^A(\widehat{i})$ is then correlated with all the vectors within $\widehat{f}_{tr}^B(\widehat{j})$, thus generating a heat map that measures the matching probability of each pixel in the $\widehat{j}$ -neighborhood with $\widehat{i}$. This probability distribution is used to compute the expected matched position $\widehat{j}'$ in I^B with sub-pixel accuracy. The final output of LoFTR is the set $\mathcal{M}_f$ of fine-level matches $\{(\widehat{i}, \widehat{j}')\}$, accompanied by a confidence score associated to each match.

3.3 Geo-LoFTR

Geo-LoFTR builds on the original LoFTR architecture by enhancing its local feature extraction module with a novel *Geo-Merger* sub-layer. This additional sub-layer integrates a gray-scale image and the corresponding camera depth map to produce coarse- and fine-level feature maps that incorporate both visual and geometric information. In the context of the map-based localization, this approach aims to enrich the representation of the ortho-projected map with the 3D info from a digital elevation model of the corresponding scene, allowing for more robust feature matching, particularly in challenging lighting scenarios where visual cues alone may lead to degeneracy. An overview of the Geo-LoFTR architecture is shown in Figure 3.5; while the graph of the Geo-Merger layer is illustrated in Figure 3.6.

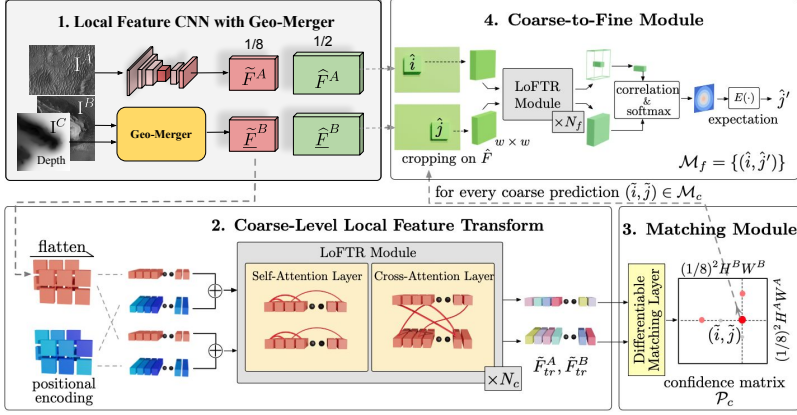


Figure 3.5: Geo-LoFTR: architecture overview. Adapted from Figure 2 of [2], licensed under CC BY 4.0.

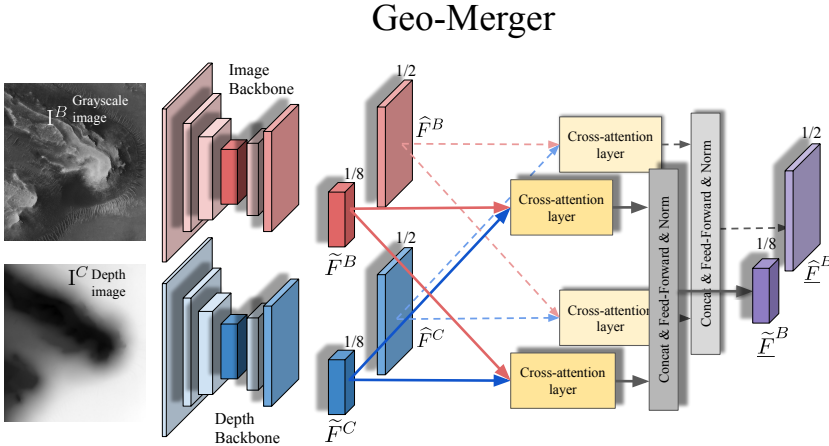


Figure 3.6: Geo-Merger architecture.

Geo-LoFTR takes three inputs: a gray-scale image (I^A) from a query observation, a gray-scale image (I^B) cropped from the ortho-rectified map, a depth image (I^C) of the same map crop, where each pixel contains the component along the map camera’s viewing direction, of the distance from an object in the scene to the camera focal plane. Like in the original LoFTR, I^A is processed by a CNN backbone that produces the feature maps $\tilde{F}^A$ and $\hat{F}^A$ at 1/8 and 1/2 scales, respectively.

In the Geo-Merger layer, I^B is processed by the same backbone as for I^A , referred to as *Image Backbone*; while I^C is fed to a separate backbone that we denoted as *Depth Backbone*. Both backbones employ the modified ResNet-18 architecture proposed by [44], consistent with the original LoFTR framework. However, by defining two different backbones in the implementation with distinct learnable parameters, we enable the model to specifically optimize for the unique characteristics of image and depth data during training, enhancing the processing capability of each modality. The backbones produce the coarse-level feature maps, $\tilde{F}^B$, $\tilde{F}^C$, and the fine-level feature maps $\hat{F}^B$, $\hat{F}^C$. $\tilde{F}^B$ and $\tilde{F}^C$ are provided to two cross-attention layers, representing two different instances of the LoFTR encoder in Figure 3.4a. Each cross-attention layer allows one feature map to be used as context while transforming the other, generating intermediate maps for each modality that incorporate information from the other. These feature maps are then concatenated along the channel dimension and sent to a feed-forward neural network. The FFN is composed of a linear layer, a ReLU activation to introduce non-linearity, and a final linear layer that brings the features’ channel dimension back to the input’s one. The FFN output is then passed to a normalization layer to contribute to stabilizing the training and improving convergence. The final product is the merged coarse-level feature map $\tilde{F}^B$. The same process is repeated for the fine-level feature maps, $\hat{F}^B$ and $\hat{F}^C$, and the merged feature map $\hat{F}^B$ is generated at 1/2 scale. $\tilde{F}^A$ and $\tilde{F}^B$ will be provided to the standard Coarse-Level Local Feature Transform module of LoFTR, while $\hat{F}^A$ and $\hat{F}^B$ will be used in the Coarse-to-Fine Module. As for the vanilla LoFTR, the final product of Geo-LoFTR is the set of fine-level matches $\mathcal{M}_f$ with associated confidence scores.

3.4 LoFTR Models Supervision and Training

LoFTR and Geo-LoFTR share the same type of supervision adopted in [2], with the final loss $\mathcal{L}$ consisting of the coarse-level module loss, $\mathcal{L}_c$, and the fine-level module’s one, $\mathcal{L}_f$:

$$\mathcal{L} = \mathcal{L}_c + \mathcal{L}_f \quad (3.10)$$

The coarse-level loss is a negative log-likelihood loss over the confidence matrix $\mathcal{P}_c$, which we chose to generate with the dual-softmax operator in Eq. 3.8. To determine ground-truth labels for $\mathcal{P}_c$, camera poses and depth maps available for both the query observations and maps in the training set are used in map-to-query and query-to-map point warping functions to compute a set of mutual nearest neighbors between two 1/8-resolution grids, based on their re-projection distances. Denoting this ground-truth

coarse matches as $\mathcal{M}_c^{gt}$, the negative log-likelihood loss $\mathcal{L}_c$:

$$\mathcal{L}_c = -\frac{1}{|\mathcal{M}_c^{gt}|} \sum_{(\tilde{i}, \tilde{j}) \in \mathcal{M}_c^{gt}} \log \mathcal{P}_c(\tilde{i}, \tilde{j}) \quad (3.11)$$

is minimized over the grid points $(\tilde{i}, \tilde{j})$ in $\mathcal{M}_c^{gt}$. The fine-level supervision is performed by using a l_2 loss. Each query point $\hat{i}$ is assigned a variance $\sigma^2(\hat{i})$ which is computed from its heatmap obtained in the coarse-to-fine module. The goal is to minimize a weighted loss, $\mathcal{L}_f$, that prioritizes points with low uncertainty $\sigma^2(\hat{i})$ and is given by:

$$\mathcal{L}_f = \frac{1}{|\mathcal{M}_f|} \sum_{(\hat{i}, \hat{j}') \in \mathcal{M}_f} \frac{1}{\sigma^2(\hat{i})} \|\hat{j}' - \hat{j}_{gt}\|_2 \quad (3.12)$$

where $\hat{j}'_{gt}$ is obtained by warping each $\hat{i}$ from $\hat{F}_{tr}^A$ to $\hat{F}_{tr}^B$ for the vanilla LoFTR ($\hat{F}_{tr}^B$, for Geo-LoFTR) with the ground-truth camera poses and depth.

LoFTR has been trained by [2] on MegaDepth [45], a large-scale dataset consisting of images collected from a variety of scenes, including urban, natural and indoor environments. We will denote this pre-trained model as *Pre-trained LoFTR*. In this work, we fine-tuned both LoFTR and Geo-LoFTR (using the pre-trained model parameters) on a synthetic Mars image dataset generated with a custom framework that we developed in Blender (Section 4). We will refer to the fine-tuned model as *Fine-tuned LoFTR* for the remainder of this work. The performance of Pre-trained LoFTR, Fine-tuned LoFTR and Geo-LoFTR will be evaluated in our MbL pipeline through several experiments in challenging lighting and scale variance (Section 5). The proposed MbL pipeline is described in the following section.

3.5 Map-based Localization Pipeline

The function of an MbL pipeline consists of registering a query observation captured by a camera with known intrinsic parameters over a geo-referenced ortho-projected map. The ultimate goal is to retrieve the query camera pose $(\mathbf{R}_{WC_{query}}, \mathbf{t}_{WC_{query}})$ in the world frame W , where $\mathbf{R}_{WC_{query}}$ is the rotation matrix that aligns W with the camera frame C_{query} and $\mathbf{t}_{WC_{query}}$ is the camera location in W . To achieve that, our pipeline uses the matched keypoints identified in the query image and the map by one of the LoFTR models described above and their corresponding 3D positions in W backprojected from the geo-reference map. Given the potentially large size of HiRISE maps, matching the query observation against the entire map would require prohibitive computational resources, particularly when leveraging CNN backbones. To mitigate this, the map is divided into multiple overlapping windows of reduced size. When no pose prior is available, an exhaustive search can be conducted by attempting to match the query image against the map windows sampled over the entire map. If a pose prior is available, the map windows can be sampled by a search area of size depending on the uncertainty of the pose, using a strategy inspired by [31]. We based the size of the search area on the

point reprojection uncertainty $\Sigma_{u,v}$ for the query image corners when reprojected on the map. Given the query and map camera poses $(\mathbf{R}_{WC_{\text{query}}}, \mathbf{t}_{WC_{\text{query}}})$, $(\mathbf{R}_{WC_{\text{map}}}, \mathbf{t}_{WC_{\text{map}}})$ in the world frame, and the pixel coordinate $(u_{\text{query}}, v_{\text{query}})$ of a 2D point P on the query image, the transformation f that projects this point to map at pixel coordinates (u, v) is composed of the following subsequent transformations:

- backprojection of P from the query image plane to 3D coordinates, $\mathbf{x}^{C_{\text{query}}}$, in the query camera location frame C_{query} :

$$\mathbf{x}^{C_{\text{query}}} = Z \cdot \mathbf{K}^{-1} \begin{bmatrix} u_{\text{query}} \\ v_{\text{query}} \\ 1 \end{bmatrix}$$

where Z is the depth value (i.e., the z-coordinate) obtained from the query depth image at $(u_{\text{query}}, v_{\text{query}})$, and $\mathbf{K}$ is the camera intrinsic matrix.

- Transformation from C_{query} to the position, $\mathbf{x}^W$, of the 3D point in W :

$$\mathbf{x}^W = \mathbf{R}_{WC_{\text{query}}} \mathbf{x}^{C_{\text{query}}} + \mathbf{t}_{WC_{\text{query}}}$$

- Transformation from W to the 3D position, $\mathbf{x}^{C_{\text{map}}}$, in the map camera frame C_{map} :

$$\mathbf{x}^{C_{\text{map}}} = \mathbf{R}_{WC_{\text{map}}}^T (\mathbf{x}^W - \mathbf{t}_{WC_{\text{map}}})$$

- Orthographic projection to the 3D map:

$$\begin{bmatrix} u \\ v \end{bmatrix} = \frac{1}{p_{\text{map}}} \mathbf{x}^{C_{\text{map}}} + \begin{bmatrix} c_{x,\text{map}} \\ c_{y,\text{map}} \end{bmatrix}$$

where p_{map} is the pixel resolution of the map and $(c_{x,\text{map}}, c_{y,\text{map}})$ are the coordinates of the optical axis in the map image plane.

Given the final transformation f is given by:

$$f = \frac{1}{p_{\text{map}}} \mathbf{R}_{WC_{\text{map}}}^T (Z \cdot \mathbf{R}_{WC_{\text{query}}} \mathbf{K}^{-1} \begin{bmatrix} u_{\text{query}} \\ v_{\text{query}} \end{bmatrix} + \mathbf{t}_{WC_{\text{query}}} - \mathbf{t}_{WC_{\text{map}}}) \quad (3.13)$$

the point reprojection uncertainty through first-order error propagation is obtained as:

$$\Sigma_{u,v} = \frac{\delta f}{\delta \eta} \Sigma_{\eta} \frac{\delta f^T}{\delta \eta}, \quad \eta = \{u_{\text{query}}, v_{\text{query}}, \mathbf{t}_{WC_{\text{query}}}, Z, \psi, \theta, \phi\} \quad (3.14)$$

where $\{\psi, \theta, \phi\}$ is the query camera orientation expressed in yaw, pitch and roll angles, and Σ_{η} is set as a diagonal matrix. The prior uncertainty Σ_{η} can be derived, for instance, from the covariance of the Range-VIO estimator. For each of the query image corner points $(u_{\text{query}}, v_{\text{query}})$, we compute the uncertainty in its projected location on the

map image (u, v) . This is quantified along the width and height directions on the map image plane as $3\sigma_u$ and $3\sigma_v$, where σ_u^2 and σ_v^2 are the entries of the covariance matrix Σ_η corresponding to each map projection. The largest uncertainty among all the points is selected as the final search area size.

After the map windows are sampled over the identified search area with specified width and overlap, each window is resized and eventually padded to match the dimensions of the query image for computational efficiency. In case the selected matching model is Geo-LoFTR the map windows are accompanied by their depth information. For each window/query pair, the selected LoFTR model generates a set of matched keypoints with associated confidence scores, and the top 500 matches are retained. Once all the pairs have been processed, the matches are aggregated and further filtered based on a specified confidence threshold. The filtered keypoints on the map are then projected back to the world frame W by using the known map camera pose and the map pixel resolution. The resulting 3D world points and the matched 2D keypoints on the query image are input to a RANSAC-PnP algorithm to estimate the camera pose $(\mathbf{R}_{WC_{\text{query}}}, \mathbf{t}_{WC_{\text{query}}})$.

4 Mars Synthetic Dataset

We generated a large synthetic Mars image dataset with the aim of training and testing LoFTR and Geo-LoFTR, under varying illumination conditions, scale, and complex terrain morphologies. In order to accomplish that, we conceived *MARTIAN* (Mars Aerial Rendering Tool for Imaging and Navigation), a custom framework developed with the 3D computer graphics software Blender, to generate realistic Martian scenes from Digital Terrain Models (DTMs) and ortho-rectified images sourced from HiRISE. *MARTIAN* allows the rendering of both RGB and depth images from simulated camera setups, including perspective and orthographic views, with user-defined camera parameters and poses. In this framework, it is possible to simulate navigation camera observations and orthographic maps at different times of the day for the selected site by properly tuning the orientation of the Sun in the simulated environment. The capability to generate a dataset spanning various combinations of lighting conditions and terrain perspectives enables the creation of multiple map/observation pairs, serving as a robust training set for our deep learning models to learn scale and lighting invariance.

In the following sections, we will first provide an overview of the HiRISE camera and the data products selected for this work. We will then discuss the *MARTIAN* framework in more detail, including HiRISE data import, terrain modeling, camera placement, and scene and lighting configuration. Finally, we will describe the generation of the rendered maps and aerial observations constituting the Mars synthetic dataset, followed by a discussion on how their image and depth information are used to form image pairs for efficient model training.

4.1 HiRISE Data

The High Resolution Imaging Science Experiment is a high-resolution camera orbiting around Mars since 2006, onboard the Mars Reconnaissance Orbiter. With its unprecedented resolution capability (up to 30 centimeters per pixel from 300 km altitude) compared to any existing orbiter imaging of the Martian surface, HiRISE has been serving as an indispensable asset for identifying and selecting safe landing sites for robotics exploration missions. The camera is equipped with a 0.5-meter primary mirror and can capture images in three color bands: blue-green or BG (400-600 nm), red (550-850 nm), and near infra-red or NIR (800-1000 nm). The instrument uses 14 CCD sensors,

with red images being 20,048 pixels wide (with a swath width of 6 km from 300 km altitude), and BG and NIR images being 4,048 pixels wide (with a swath width of 1.2 km from 300 km altitude). The onboard memory holds up to 28 Gbit, with images transmitted in compressed form, typically using JPEG 2000 format. Further specifications are reported in Table 4.1, adapted from [4]. The main HiRISE data products used in this work are Digital Terrain Models (DTMs) and ortho-images, the generation of which involves a meticulous multi-step process aimed at achieving high accuracy and detail [46]. DTMs are derived from stereo pairs of images of the same area on the ground, captured from different viewing angles. Initially, the images are corrected for any optical distortions inherent to the HiRISE camera, with the aid of spacecraft pointing information at each observation time. Then, a triangulation step, also known as bundle adjustment, is performed to align the stereo images to each other and to the global elevation map produced by the Mars Orbiter Laser Altimeter (MOLA). Once triangulated, terrain models are generated through computationally intensive automated processes and then reviewed for any errors and artifacts. The DTM produced in this way typically presents a post spacing (i.e. the distance in meters between individual data points - posts - in the terrain model) about four times the pixel resolution of the input stereo images. HiRISE images are usually 0.25-0.5 m/pixel, thus resulting in DTM resolution of approximately 1-2 m/post. The sourced stereo pairs used to generate the terrain model are then employed to produce ortho-rectified images, which will be referred to as *ortho-images*, where each pixel has been geometrically corrected to represent a nadir view of the terrain, with topographical distortions being eliminated. The final DTMs and ortho-images are then made available through the Planetary Data System (PDS), with the former being in standard PDS image object (.IMG) format and the latter in JPEG2000 format. In this work, we chose to utilize a 1 m/post DTM (product ID: DTEEC_045994_1985_046060_1985_U01) and a 0.25 m/pixel ortho-image (product ID: ESP_046060_1985_RED_A_01_ORTHO) obtained with an equirectangular projection. These have been generated from stereo pairs imaging the Jezero Crater landing site selected for the Mars2020 Mission, over an area of 6.737 km by 14.403 km. The right observation (ESP_045994_1985) of the source stereo pair has been acquired with a resolution of 28.1 cm/pixel on May 19, 2016 at 15:14 local Mars time, with the Sun about 42° above the horizon and with 2.9° sub-solar azimuth, defined as the clockwise angle from the observation reference axis (i.e. the line from the center to the right edge of the image) to the sub-solar point on the Mars surface. The left observation (ESP_046060_1985) has a resolution of 29.6 cm/pixel and has been captured on May 21, 2016 at 15:22 local Mars time, with a Sun angle above the horizon of 40° and 1.7° sub-solar azimuth. We will refer to the Sun angle above the horizon as *Sun elevation angle* (EL) and the opposite of the sub-solar azimuth as *Sun azimuth angle* (AZ), for the remainder of this work. A 1 m / pixel resolution version of the ESP_046060_1985_RED_A_01_ORTHO ortho-image (ESP_046060_1985_RED_C_01_ORTHO) is shown in gray-scale in Figure 4.1, for visualization purposes. Further information on the stereo images is reported in Table 4.2.

Parameter	Performance	Comments
Ground Sample Distance (GSD)	30 cm/pixel	From 300 km altitude
Telescope aperture	0.5 m, f/24	For resolution and SNR
Spectral range	400 to 600 nm 550 to 850 nm 800 to 1000 nm	Blue-Green (BG) Red Near infra-red (NIR)
SNR Blue-Green Red NIR	Typically 100:1 Typically 200:1 Typically 100:1	Achieved with TDI, backside thinned CCDs, and 50 cm aperture
Swath Width Red BG & NIR	>6 km >1.2 km	From 300 km altitude From 300 km altitude
Swath length	>2x swath width	Along track
Data precision	14 bit A/D	12 to 13 bit usable
Data compression	Real-time 14 to 8 bit Up to 16 x 16 binning Lossless compression at SSR	Look-up table Increases areal coverage $\approx 2:1$
Data storage	28 Gbits	All channels
Number of pixels across swath	20,264 Red 4,048 Green and NIR	From swath and pixel scale
TDI line time	$\geq 76 \mu\text{sec}$	To match ground track speed
CCD read noise	<50 electrons rms at 22°C	Achieve SNR at low signal levels
FOV	$1.14^\circ \times 0.18^\circ$	
IFOV	$1 \times 1 \mu\text{rad}$	Detector angular subtense
Relative Radiometry	<1 % pixel to pixel	Absolute 20 %

Table 4.1: HiRISE Requirements and Performance Characteristics. Adapted from [4]

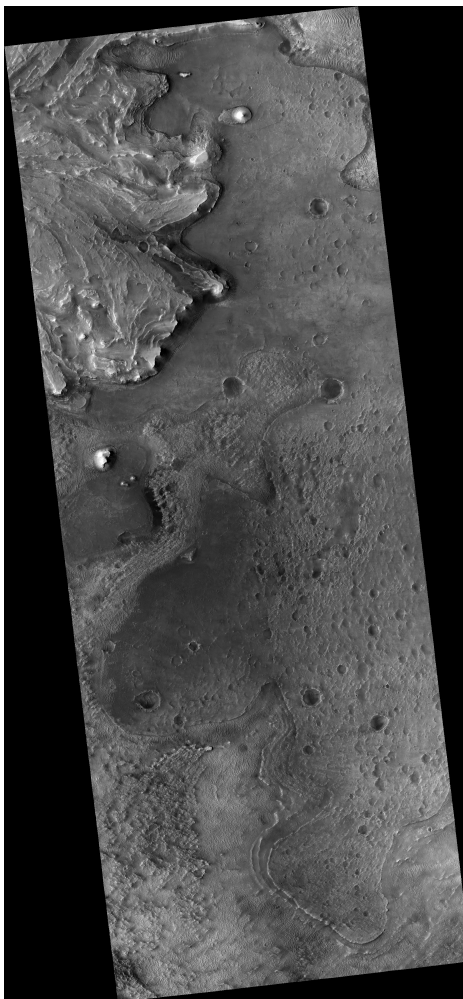


Figure 4.1: HiRISE ortho-rectified image product ESP_046060_1985_RED_C_01_ORTHO of the Jezero crater site, generated with 1 m / pixel resolution from the ESP_045994_1985 and ESP_046060_1985 stereo pairs.

Parameter	Left Image	Right Image
Acquisition date	24 May 2016	19 May 2016
Local Mars time	15:22	15:14
Latitude (centered)	18.431°	18.427°
Longitude (East)	77.438°	77.437°
Spacecraft altitude	278.4 km (173.0 miles)	280.0 km (174.0 miles)
Original image scale range	29.6 cm/pixel (objects 89 cm across)	28.1 cm/pixel (objects 84 cm across)
Emission angle	19.8°	3.8°
Phase angle	30.1°	51.4°
Solar incidence angle	50° (Sun about 40° above the horizon)	48° (Sun about 42° above the horizon)
Solar longitude	157.6°, Northern Summer	154.9°, Northern Summer
North azimuth	96°	97°
Sub-solar azimuth	1.7°	2.9°

Table 4.2: Comparison of the left and right HiRISE stereo images.

4.2 MARTIAN: Mars Aerial Rendering Tool for Imaging and Navigation

MARTIAN is a python-based software integrated in Blender 4.0 , specifically designed to import HiRISE DTMs and ortho-images, and generate synthetic maps and aerial observations of the sourced Martian site under specified lighting conditions, with user-defined camera intrinsic and extrinsic parameters. The framework operates with two modes: *demo* and *dataset*. The former allows for generating perspective and orthographic observations, with depth information, at desired locations of the terrain mesh, with the included possibility to visualize pixel-to-pixel correspondences between perspective-orthographic or perspective-perspective image pairs. The latter allows to generate maps and observational image datasets, with camera locations sampled all over the Martian site with specified altitude and attitude ranges and Sun azimuth and elevation angles.

4.2.1 Terrain and texture modeling

The Jezero HiRISE DTM in PDS-compliant .IMG format was imported in Blender 4.0 using a modified version of the original HiRISE import plug-in, originally developed for Blender 2.7 [47]. This add-on can load the terrain data at a desired resolution within a range (0., 100] % of the full model resolution. This approach enables the user to initially stage a scene with a lower resolution terrain, perform Blender operations such as texture application and scene setup, and then increase the resolution to prepare for the final rendering, thereby optimizing computational time. A triangulation algorithm is then used to create a mesh from the imported terrain data. The algorithm generates a list of valid vertices and faces, leveraging the orthogonal structure of the DTM raster for computational efficiency. Each vertex is defined by a 3D coordinate tuple read from the DTM raster, with the z-coordinate representing the terrain elevation (in meters) and with

the xy-coordinates corresponding to the pixel indices in the raster, scaled by the DTM resolution (in meters / pixel). Each face is defined by a triplet of vertex indices forming triangular elements in the mesh. The triangulation processes the raster row by row, identifying valid quadrilateral areas formed by vertices in adjacent pairs of rows, with each area being subsequently divided into two triangles. This process continues until the entire raster has been properly triangulated. By leveraging terrain metadata, including geographic information and resolution, the modified HiRISE importer also creates a UV map for the generated mesh in Blender, allowing for ortho-images to be draped over the companion DTM and used as a high-fidelity terrain texture. The MARTIAN framework initially imports the Jezero DTM with 10% of its original resolution, allowing for efficient terrain setup. Then, a material is added to the mesh, with its surface shading model being controlled via a Principled BSDF (Bidirectional Scattering Distribution Function) node set in the Blender shading editor. The Principled BSDF integrates multiple material properties, such as diffuse, metallic, subsurface and transmission into a single node, thus allowing to model a wide variety of materials. For the simulated Mars terrain, we set the Principled BSDF with the main parameters shown in Table 4.3.

Parameter	Values	Range	Description
Metallic	0.0	[0.0, 1.0]	Blends between a dielectric and metallic material model. 0 is non-metal, 1 is fully metallic.
Roughness	0.9	[0.0, 1.0]	Controls the smoothness of the surface; lower values create a glossy surface.
IOR (Index of Refraction)	1.45	[1.0, 4.0]	Defines the refractive index, mainly for transparent or refractive materials.
Alpha	0.0	[0.0, 1.0]	Controls the transparency of the material.

Table 4.3: Main parameters of the Principled BSDF node in Blender for the synthetic Mars terrain.

The 0.25 m / pixel Jezero ortho-image is then loaded in the shading editor in .JP2 format to serve as the base texture for the terrain surface. The ortho-image coordinates data are retrieved by a Texture Coordinate Node in the Blender shading editor to ensure that the texture is properly mapped onto the 3D mesh. The complete configuration of the shading editor is shown in Figure 4.2. Finally, the mesh is reloaded with its full resolution of 1 meter per post, ensuring that the textures' alignment is retained, and a .blend file is generated. Figure 4.3 shows some views of the terrain model with texture.

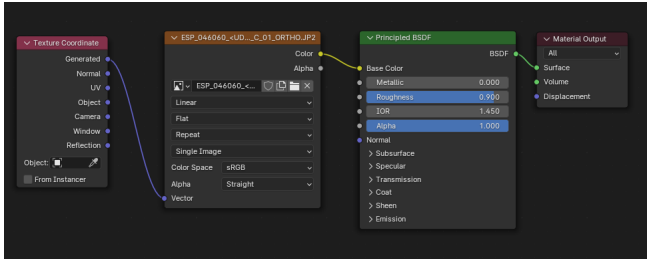


Figure 4.2: Shading editor configuration for the MARTIAN framework.

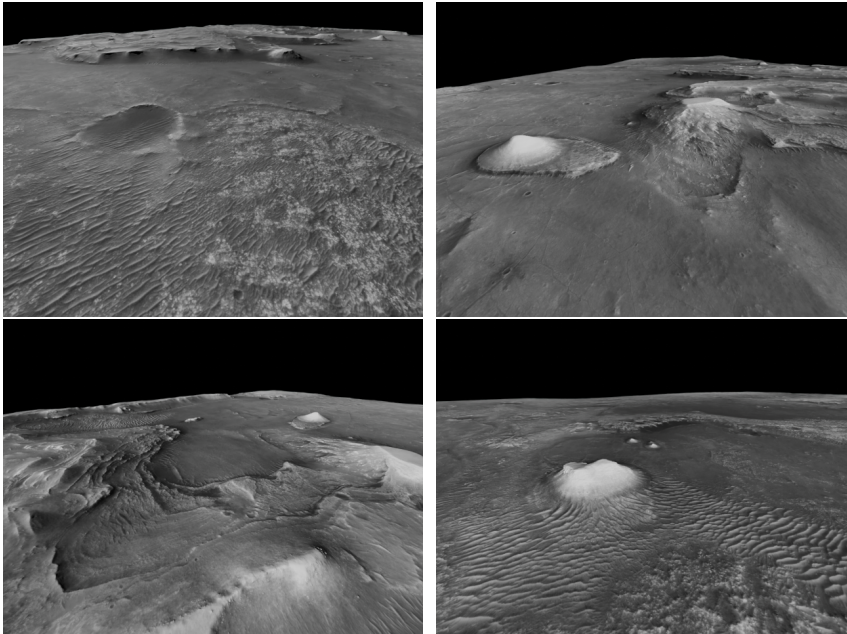


Figure 4.3: Views of the Blender terrain for the Jezero Crater site in MARTIAN.

4.2.2 Scene setup and camera modeling

The MARTIAN framework allows for setting multiple scenes for perspective and orthographic imaging with user-defined camera parameters, enabling the possibility to choose different rendering settings (such as image resolution, pixel aspect ratio and simulated camera exposure) for each camera type. For the perspective projection model, Blender supports the modeling of camera intrinsic elements such as focal length, sensor width and lens distortion. The orthographic projection camera type requires instead the specification of the orthographic width in meters, with the orthographic height being computed

from the pixel resolution set along the image width and height. Blender also allows for specifying the camera clip range in terms of the maximum and minimum visible distance in meters. These parameters can affect the accuracy of the camera depth estimation.

To delve into the details of the camera pose specification, it is necessary to introduce the reference frames used in MARTIAN. The Blender reference frame, also referred to as our *world* reference frame W , is defined as a right-handed Cartesian coordinate system, with the X-axis pointing eastward, the Z-axis pointing to the up-direction and the Y-axis completing the orthogonal set. A frame for the Blender camera object O , can be defined as centered in the camera and with its x, y and z axes initially aligned with the world frame W . The camera object pose with respect to the world frame is described by $(R_{WO}|t_{WO})$, where t_{WO} is the camera body translation vector from the world origin and R_{WO} is the rotation matrix that aligns W to O . In MARTIAN it is possible to input cameras locations in the world frame by providing the (t_{WO_x}, t_{WO_y}) coordinates and the altitude values (in meters) with respect to the terrain at those coordinates. The camera object attitude is specified by assigning the Euler angles yaw (γ), pitch (β), and roll (ϕ) in degrees, following a 312 rotation sequence. Specifically, (γ) represents rotation around the Z-axis, (β) is the rotation around the X-axis, and (ϕ) is the rotation around the Y-axis. To position the camera object above the terrain mesh at the desired altitude, MARTIAN uses a ray-tracing approach. Starting with a high z-coordinate value above the desired location in (t_{WO_x}, t_{WO_y}) , it casts a ray from the camera object origin downward along the Z-axis. The ray's intersection with the terrain mesh is calculated using Blender's *BVHTree* structure, and the camera z-coordinate t_{WO_z} is adjusted to the intersection point plus the altitude specified in input. A BVH (Bounding Volume Hierarchy) tree is a data structure used by Blender to efficiently organize geometric objects in 3D space, where the 3D mesh is divided into hierarchical bounding boxes, facilitating quick spatial queries over the terrain model. Another reference frame, referred to as the *camera* frame C , is conventionally used in robotics for tasks such as perspective projections and camera pose estimation. This frame can be obtained by rotating the camera object frame O by 180° around its X-axis. As a result, the C frame is centered at the camera's optical center (the center of the captured image), with its X-axis pointing to the right along the image width, its Y-axis pointing downward along the image height, and its Z-axis pointing forward toward the terrain, completing the orthogonal set. The pose of the C frame with respect to the world frame W , denoted as $(R_{WC}|t_{WC})$, is given by $t_{WC} = t_{WO}$ and $R_{WC} = R_{WO}R_x(\pi)$, where $R_x(\pi)$ represents a 180° rotation around the X-axis. While the camera object pose $(R_{WO}|t_{WO})$ is mainly used for operations with camera objects in Blender, the pose $(R_{WC}|t_{WC})$ will be adopted to perform perspective and orthographic projections to support the generation of the LoFTR training set (Section 4.4) and in the MbL evaluation chapter (Chapter 5).

4.2.3 Lighting

MARTIAN provides the capability to render scenes in Blender with various illumination conditions by tuning Sun light source parameters such as the irradiance, the apparent disk diameter and orientation in the Blender frame. The irradiance is set by the *strength* parameter (in W/m^2), where higher values cast brighter illumination and shadows in the scene. The angular diameter of the apparent Sun disk as seen by the scene, controlled by the *angle* parameter in radians, impacts the softness or harshness of shadows. The Sun orientation simulates the scene time of the day and it is specified by the user through Elevation (EL), defined as the angle between the Sun and the xy-plane in the Blender frame, and Azimuth (AZ), which is the counterclockwise angle between the sub-solar point on the xy-plane and the x-axis. In the Blender reference frame, the Sun's orientation is represented by Euler angles in a 321 rotation sequence, where the rotation around the z-axis corresponds to AZ, and the rotation around the y-axis is equal to $90^\circ - \text{EL}$. The lighting computations are performed using the Blender Cycles engine. This is a physically-based rendering engine that uses a ray tracing algorithm to accurately simulate light behavior. This algorithm traces light rays from the camera into the scene, bouncing around until they find a light source (like the simulated Sun) or the world background. The method uses both *direct* and *indirect* light sampling, where the samples correspond to the ray paths to trace for each pixel in the final render. The direct light sampling consists in tracing rays directly towards the light source to compute its contribution to each pixel. In the indirect sampling, the rays follow the surface BSDF, allowing the light to bounce off surfaces before reaching others. There are four different types of rays: the *camera rays*, originated straight from the camera; the *reflection* and *transmissions* ones generated by surface reflections and transmissions; the *shadow* rays used to render transparent shadows. Reflection and transmissions rays have additional properties like diffusivity (generation by diffuse reflection or transmission), glossiness (caused by a glossy specular reflection or transmission) and singular (rays generated by a perfectly sharp reflection or transmission). Cycles allows for the controlling further properties such as the maximum number of light bounces and the number of bounces for the different ray types. A list of the main light objects and cycles parameters used in the MARTIAN framework is shown in Table 4.4.

Parameter	Value	Description
Sun Strength	580 W/m^2	The irradiance of the Sun in the scene.
Angle	0.35°	The size of the Sun light according to its angular diameter as seen from Mars.
Device	CPU	Sets the device for rendering (CPU or GPU).
Samples	16	Determines the number of samples per pixel for rendering.
Max Bounces	100	Sets the maximum number of light bounces in the scene.
Min Bounces	10	Sets the minimum number of light bounces in the scene.
Diffuse Bounces	10	Configures the number of diffuse light bounces in the scene.
Glossy Bounces	4	Configures the number of glossy light bounces in the scene.

Table 4.4: Sun object Cycles Rendering Engine settings in MARTIAN.

4.2.4 Rendering

RGB and depth images can be rendered in MARTIAN with customized settings for each individual scene. For each camera view, users can specify image pixel resolution (width and height), pixel aspect ratio, and exposure, allowing scenes to be rendered with distinct visual properties while utilizing the same underlying terrain. The simulated camera exposure can be adjusted for a fine-tuning of the overall brightness level in the rendered image. Depth images from perspective cameras are rendered in Blender by using a depth buffer, also referred to as a z-buffer. This consists of a data structure that records the component, along the camera's viewing direction, of the distance from an object in the scene to the camera at each pixel of the rendered image. In this context, we will refer to this component as the z_c distance, with Z being the camera viewing axis. The depth-buffer z_d stored at the distance z_c from the camera is given by:

$$z_d = \frac{f}{(f - n)} - \frac{1}{z_c} \frac{fn}{(f - n)} \quad (4.1)$$

where z_d is normalized in the range $[0, 1]$, with f and n representing the distances from the far and near clipping planes of the camera, respectively.

For each fragment generated by the rasterizer (the component in the graphics pipeline that is responsible for converting 3D geometric data into a 2D image) at the rendering stage, the z_c distance is compared to the one already stored in the buffer at the same pixel location. If the new fragment is closer to the camera, the depth buffer is updated with the new value; otherwise, the fragment is discarded. This process ensures that objects closer to the camera correctly occlude the ones in the background.

Typically, the depth buffer is stored in fixed-point format and the precision in z_c is given by:

$$\frac{dz_c}{dz_d} = \frac{z_c^2}{S \cdot n} - \frac{z_c^2}{S \cdot f} \quad (4.2)$$

where $S = 2^d - 1$ is the buffer's bit depth, with b representing the number of bits used for the fixed-point representation. For $f \gg n$, Equation 4.2 can be approximated by:

$$\frac{dz_c}{dz_d} \approx \frac{z_c^2}{S \cdot n} \quad (4.3)$$

showing how the accuracy is highest at the near clipping plane and decreases with increasing distance. In the final depth images rendered in Blender from perspective cameras, each pixel stores the z_c values with an accuracy that can be modeled by Eq. 4.2 and $b=32$ bit. Figure 4.4 shows the depth accuracy as a function of z_c for different values of n and a fixed value of f . In the assumption of $f \gg n$, the choice of f doesn't significantly affect the depth accuracy. Considering that our simulated observations will be in the altitude range $[64, 200]$ m and that the maximum size of the map is 14,403 m, it is reasonable to assume that the maximum measurable depth is equal to the one measured from a perspective camera located at the one short edge of the map and pointing at the opposite edge, resulting in 14,403 m. The left panel of Figure 4.4. shows that up to this

depth value, the depth precision is below 0.1 m for most of the distances (one order of magnitude smaller than the DTM accuracy). For observations taken from nadir-pointing perspective cameras in the aforementioned altitude range. The depth-precision is 5 orders of magnitude less than the DTM accuracy, as shown in the right panel of Figure 4.4.

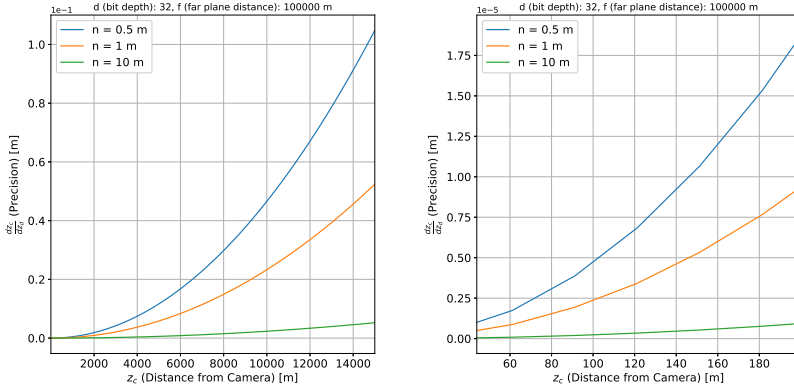


Figure 4.4: Depth precision computed from Eq. 4.2 for perspective cameras in Blender within the z_c range $[0, 14,404]$ m (*left panel*) and $[0, 200]$ m (*right panel*), for different distances n of the near clipping plane of the camera, with depth the buffer stored in 32 bit floating point format and the far clipping plane f at 100000 m.

Operating modes

MARTIAN supports two main operating modes: *demo* and *dataset*. The demo mode offers a quick tool to sample desired locations on the Martian DTM and assess the visual appearance of the imaged scene under different viewpoints and illumination conditions. It generates rendered images and depth maps from two perspective cameras and one orthographic camera, with desired poses specified by xy-coordinates, altitude and the yaw, pitch and roll angles with respect to the world reference frame W defined above. With this mode it is also possible to visualize pixel-to-pixel correspondences between images generated by two overlapping perspective views, or between the perspective and the orthographic images. The demo can be useful to inform the parameters space to be used for the dataset generation process.

The other supported mode renders a set of RGB and depth images from different poses of a perspective camera and an orthographic map of the terrain, with the aim to simulate multiple aerial observations and synthetic HiRISE-like maps under different lighting. Figure 4.1 shows that the HiRISE terrain map is included in a bounding box. A lower-resolution version of the loaded HiRISE ortho-image can be imported in MARTIAN to extract the size and orientation of the valid terrain with respect to the world frame while saving computational resources. This step is crucial to ensure that camera object xy-coordinates (t_{WO_x}, t_{WO_y}) are always sampled from the actual terrain (with valid depth

points) rather than areas of the map bounding box, the pixels of which do not correspond to mesh points (characterized by invalid depth points). A set number of observations, provided by the user, is randomly sampled from the valid terrain areas, with (t_{WO_x}, t_{WO_y}) drawn from a uniform distribution within certain ranges. In order to accomplish that, the terrain orthographic width (w_T) and height (h_T) in meters are initially inferred by an automatic identification process of the terrain corners in the ortho-image, and with the help of the HiRISE map metadata. The terrain center coordinates $c_T = (c_{T_x}, c_{T_y}, c_{T_z})$ are identified in the world frame W , and a terrain reference frame T can be defined with origin in c_T , the x-axis pointing rightward along the terrain width direction, the y-axis pointing upward along the terrain height direction, and the z-axis completing the orthogonal set. Then the translation t_{WT} and the rotation matrix R_{WT} that transform the frame W into T are computed. This process allows for randomly sampling the camera x_T and y_T coordinates in the T frame, within the ranges $[-(w_T - M_w), w_T - M_w]$ and $[-(h_T - M_h), h_T - M_h]$ respectively, where M_w and M_h are width and height margins equal to half of the camera swath width and height at the maximum altitude of the user-defined altitude range. The sampled points are then transformed back to the world frame by using the terrain pose $(R_{WT}|t_{WT})$. At each of the sampled locations, the camera is placed above the mesh with altitude and yaw, pitch, and roll values drawn from user-defined ranges with an uniform random distribution. The camera intrinsics and the pose in terms of t_{WO} and (γ, β, ϕ) , along with altitude data and clipping ranges, are stored for each observation and serve as ground truths for the image and depth data in the dataset. Camera parameters and poses are also stored for the orthographic cameras rendering map images and depth. Due to the large size of the map, the orthographic map images are rendered in tiles for efficient rendering, storage, processing, and retrieval during data loading. Orthographic depth images of the same map are produced in a similar way.

Finally, the MARTIAN dataset mode allows for independent rendering of observations and maps, enabling users to create various combinations of lighting conditions. This flexibility facilitates achieving the desired lighting variability between the maps and observations.

4.3 Dataset generation

We generated a dataset comprising 41 orthographic maps from combinations of Sun azimuth ($0^\circ - 360^\circ$, 45° steps) and elevation ($2^\circ, 5^\circ, 10^\circ, 30^\circ, 60^\circ, 90^\circ$), along with 4,5000 aerial observations with an altitude range $[64, 200]$ m, at Sun $AZ=180^\circ$, and $EL=40^\circ$. Two additional sets of 500 observations have been rendered in the same altitude range with $(AZ=180^\circ, EL=5^\circ)$ and $(AZ=180^\circ, EL=40^\circ)$ respectively, for further testing in challenging illumination conditions. Further observation datasets have been generated for MbL evaluation over distinct terrain morphology (Section 5.2) and different times of day in a simulated Martian day from the Jezero Crater landing site (Section 5.4). The maps are gray-scale images of size $26,949 \times 57,613$ pixels, with a 0.25 m / pixel scale. Depth images of the same size are collected with the same scale, even though the depth resolution is provided by the DTM accuracy of 1 m / pixel. The observations - or queries - are gray-scale images of size 640×480 , with a scale spanning from 0.25 m / pixel to

0.78 m / pixel within the assigned altitude range. These are generated by nadir-pointing perspective cameras with a pinhole camera model characterized by a sensor width of 80 mm, a focal length of 32 mm, 0 lens shifts along the camera x and y directions, and a pixel aspect ratio equal to 1. The sensor height is determined by the image size, sensor width and pixel aspect ratio and is equal to 60 mm. Depth images are also generated for the same observations. The top panel of Figure 4.5 shows a map tile with three observations sampled from when the illumination conditions are the same; while the bottom panel shows the corresponding depth information. Figure 4.6 illustrates the effect of different combinations of Sun azimuth and elevation on the visual appearance of the map tile considered in Figure 4.5.

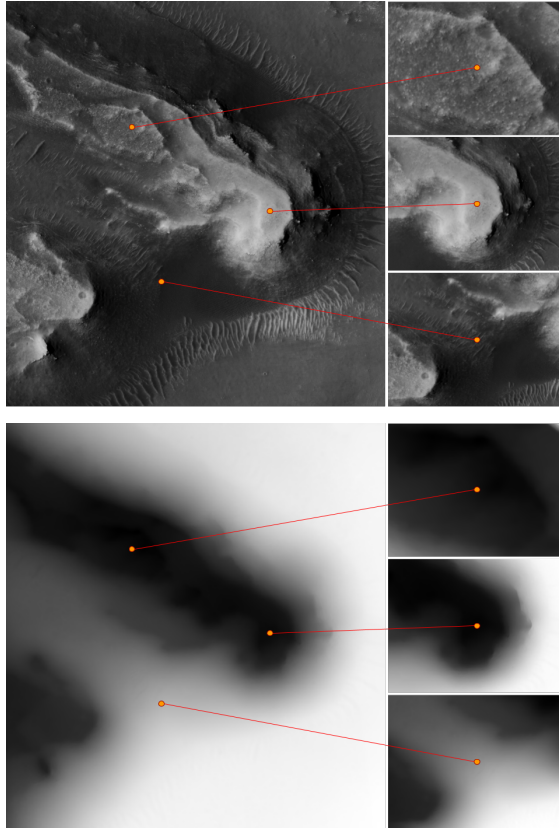


Figure 4.5: Gray-scale images (top panel) and normalized depth image (bottom panel) of an orthographic map tile compared with three observations sampled from different locations under the same illumination condition and map/observation scale ratio equal to 1.00 (top query), 2.08 (middle query), 3.12 (bottom query). Generated with MARTIAN.

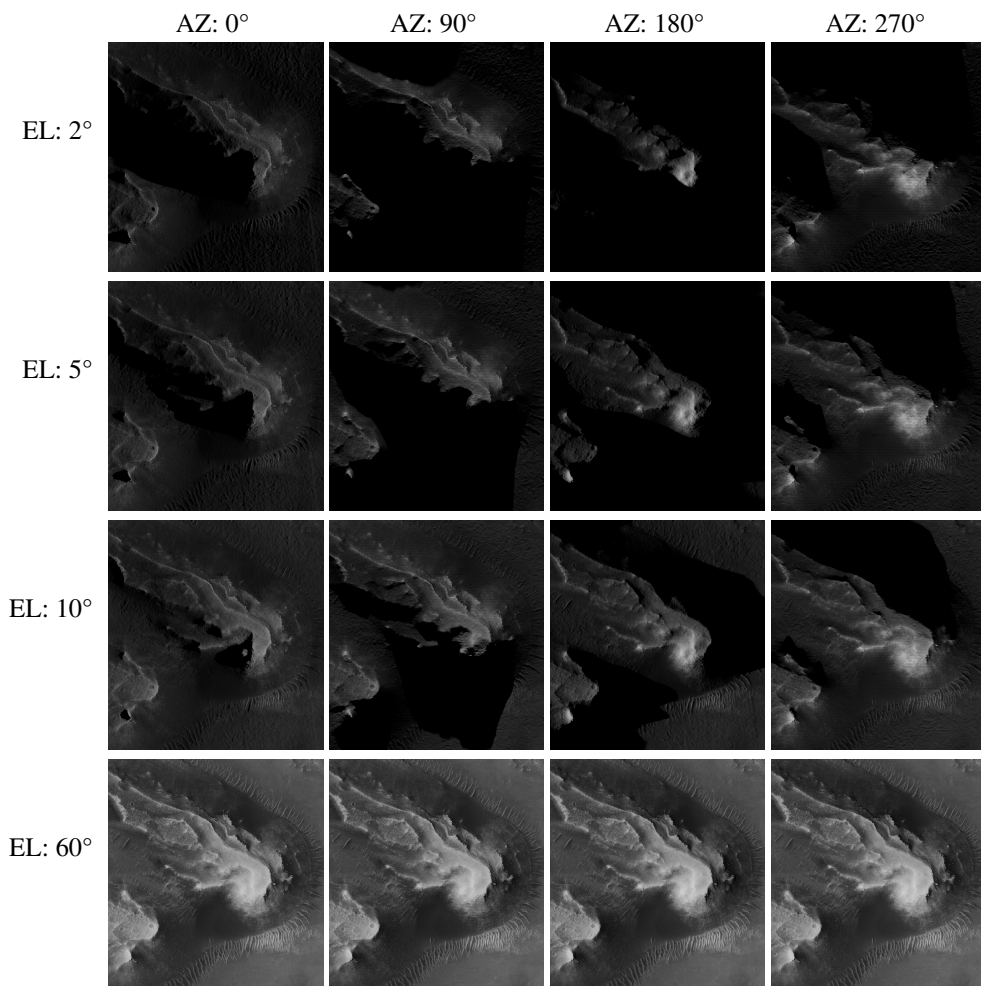


Figure 4.6: Gray scale images of a map tile from the Jezero crater site, rendered with different combinations of Sun Azimuth (AZ) and Elevation (EL). Generated with MARTIAN.

This dataset will be used to form training and test sets for evaluating the ability of LoFTR and Geo-LoFTR to enhance the robustness of the MbL under varying lighting and scale differences between maps and observations. The process will be discussed in more detail in the following section.

4.4 LoFTR training set

A training set for LoFTR and Geo-LoFTR is created by generating pairs made of observations and map windows that show a certain amount of overlap, for multiple combinations of Sun azimuth and elevation angles' offsets between map windows and queries. We also aimed to provide the training pairs dataset with an appropriate level of scale variance, enough for the models to learn effectively without overwhelming the learning process. This is achieved by properly selecting the size of the map windows based on two main sources of scale variability: 1) the map window size reduction operated by the LoFTR models; 2) the variability in the altitude of the observations forming the pair. For simplicity, we will focus on the width scale of the pair of images, with the following reasoning easily extensible to the height dimension. Starting from a condition where map window and query pixel resolutions are matched at 0.25 cm / pixel, which corresponds to an observation altitude of 64 m, images taken at higher altitudes will have a reduced scale. Meanwhile, the image size reduction operated by LoFTR at the image pre-processing stage causes a downscaling in the map windows, which works against the objective of achieving a scale ratio between the query and the LoFTR-resized map window different than unity. This ratio $r_{\text{query/win}}$ between the scales of the query and the map window, and scale ratio r_{resized} between the LoFTR-resized map window and the original one are given by Eq. 4.4 and Eq. 4.5:

$$r_{\text{query/win}} = \frac{p_{\text{win}}}{p_{\text{query}}} = \frac{p_{\text{win}} w_{\text{query}}}{H_{\text{query}}} \cdot \frac{f_{\text{query}}}{S_{\text{query}}} \quad (4.4)$$

$$r_{\text{resized}} = \frac{p_{\text{win}}}{p_{\text{resized}}} = \frac{w_{\text{resized}}}{w_{\text{win}}} \quad (4.5)$$

where:

- p_{query} and p_{win} are the pixel resolutions of the query image and the map window (the latter corresponding to 0.25 m / pixel), respectively;
- p_{resized} is the pixel resolution of the map window resized by LoFTR;
- w_{win} and w_{query} are the map window and query image widths;
- w_{resized} is the width of the resized map window;
- H_{query} is the altitude of the observation;
- f_{query} and S_{query} are the focal length and sensor width of the query camera.

The final scale ratio $r_{\text{query/resized}}$ between the query image and the resize map window is given by:

$$r_{\text{query/resized}} = \frac{p_{\text{resized}}}{p_{\text{query}}} = \frac{p_{\text{resized}}}{p_{\text{win}}} \cdot \frac{p_{\text{win}}}{p_{\text{query}}} = \frac{r_{\text{query/win}}}{r_{\text{resized}}} = \frac{w_{\text{query}}}{w_{\text{resized}}} \cdot \frac{p_{\text{win}}}{H_{\text{query}}} \cdot \frac{f_{\text{query}}}{S_{\text{query}}} \cdot w_{\text{win}} \quad (4.6)$$

In this work, the map window is resized by LoFTR to the same size as the query image, resulting in $w_{\text{query}}/w_{\text{resized}} = 1$ and the ratio $r_{\text{query/resized}}$ (Eq. 4.7) mainly depending on the selected map window size and the observation altitude, given the query camera parameters and the map resolution are fixed quantities.

$$r_{\text{query/resized}} = \frac{p_{\text{win}}}{H_{\text{query}}} \cdot \frac{f_{\text{query}}}{S_{\text{query}}} \cdot w_{\text{win}} \quad (4.7)$$

Eq. 4.7 can also be expressed in terms of the swath widths sw_{query} and sw_{win} covered by the observation and the map window, as follows:

$$r_{\text{query/resized}} = \frac{sw_{\text{win}}}{sw_{\text{query}}} \quad (4.8)$$

To ensure that the terrain surface area captured by the map window is always larger than the one observed by the query camera at its maximum altitude $H_{\text{query}_{\text{max}}}$, the minimum ratio $r_{\text{query/resized}_{\text{min}}}$ is set to 1. In the assumption of using map windows of fixed size, w_{win} is given by:

$$w_{\text{win}} = \frac{H_{\text{query}_{\text{max}}}}{p_{\text{win}}} \cdot \frac{S_{\text{query}}}{f_{\text{query}}} \quad (4.9)$$

By replacing Eq. 4.9 into Eq. 4.7, the scale ratio is determined by:

$$r_{\text{query/resized}} = \frac{H_{\text{query}_{\text{max}}}}{H_{\text{query}}} \quad (4.10)$$

Given our altitude range of [64, 200] m, the scale ratio for the training pairs will vary between 1.000 and 3.125, which may impose excessive strain on the learning process during training. To mitigate this, we divide the altitude range into two sub-ranges: [64, 132] m and (132, 200] m. Each sub-range is assigned a different map window width, calculated such that the minimum scale ratio $r_{\text{query/resized}_{\text{min}}}$ is equal to 1 at the upper limit of each sub-range. By choosing a map window height that maintains the same aspect ratio as the query image (640×480 , i.e. 4:3) which is characterized by a pixel aspect ratio of 1, the scale ratio $r_{\text{query/resized}}$ applies uniformly along both the width and height dimensions. Thus, the map window sizes are 1320×990 pixels for the [64, 132] m sub-range and 2000×1500 pixels for the (132, 200] m sub-range. Figure 4.7 shows the variation of $r_{\text{query/resized}}$ with the observation altitude for the selected map windows' sizes.

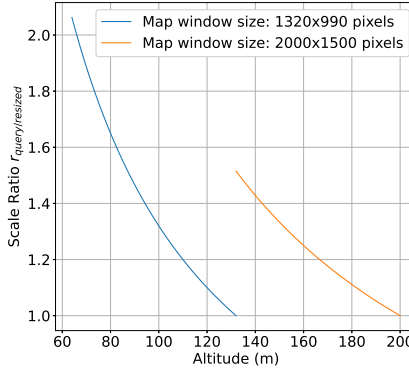


Figure 4.7: . Scale ratio $r_{\text{query/resized}}$ (from Eq. 4.7) between the map window (resized by LoFTR) and the query image, within the altitude range of the observations in the training set.

Once a map, with specified sun azimuth and elevation, and a set of observations, under different lighting conditions, are loaded, the map is divided into windows of sizes and overlap determined by the altitude sub-range of each observation. For each observation in the set, multiple map windows are selected to be paired with the same query image, forming multiple training pairs per observation. The selection process is determined by a concurrent inspection of the percentage of valid depth pixels and its overlap with the query image. Since in the rendered maps the terrain is enclosed in a bounding box (as shown in Section 4.2), the sampled map windows may include areas that do not correspond to portions of the actual surface mesh, with pixels characterized by invalid depth values. A binary *depth mask* is computed for each map window, where 0 (black) is assigned to invalid depth pixels and 1 (white) to the valid ones. If the percentage of black pixels in the depth map is above a specific threshold, the map window is discarded. The training set used in this work has been generated with depth thresholds set to 10%.

Finally, the overlap between the map window and the query image is assessed. The pixels in the query image, treated as 2D key points, are back-projected from the image plane to the query camera frame C_{query} . Given the pixel coordinates (u, v) of a 2D point, its position $\mathbf{x}^{C_{\text{query}}}$ in the query camera frame C_{query} is computed using the inverse of the perspective projection:

$$\mathbf{x}^{C_{\text{query}}} = Z \cdot K^{-1} \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} \quad (4.11)$$

where $\mathbf{x}^{C_{\text{query}}} = [x^{C_{\text{query}}} \ y^{C_{\text{query}}} \ z^{C_{\text{query}}}]^T$ represents the 3D coordinates of the point in C_{query} , and Z is the depth value (i.e., the z -coordinate) obtained from the query depth image at pixel location (u, v) . The matrix K is the camera intrinsic matrix, which is defined as:

$$K = \begin{bmatrix} f_x & 0 & c_{x,\text{query}} \\ 0 & f_y & c_{y,\text{query}} \\ 0 & 0 & 1 \end{bmatrix} \quad (4.12)$$

where f_x and f_y are the focal lengths of the camera in the x and y directions (in pixel units), and $c_{x,\text{query}}$ and $c_{y,\text{query}}$ are the coordinates of the camera's principal point (the optical center) in the image plane. Then, the 3D point $\mathbf{x}^{C_{\text{query}}}$ is transformed into the world frame W using the following transformation:

$$\mathbf{x}^W = \mathbf{R}_{WC_{\text{query}}} \mathbf{x}^{C_{\text{query}}} + \mathbf{t}_{WC_{\text{query}}} \quad (4.13)$$

where $\mathbf{x}^W = [x^W \ y^W \ z^W]^T$ is the position of the point in the world frame, $\mathbf{R}_{WC_{\text{query}}}$ is the rotation matrix from the world to the query camera frame, and $\mathbf{t}_{WC_{\text{query}}}$ is the translation vector of the query camera in W . The 3D points $\mathbf{x}^W$ in the world frame, which were back-projected from the query image, are then transformed into the map camera frame C_{map} using the following transformation:

$$\mathbf{x}^{C_{\text{map}}} = \mathbf{R}_{C_{\text{map}}W} \mathbf{x}^W + \mathbf{t}_{C_{\text{map}}W} \quad (4.14)$$

where $\mathbf{x}^{C_{\text{map}}} = [x^{C_{\text{map}}} \ y^{C_{\text{map}}} \ z^{C_{\text{map}}}]^T$ are the coordinates of the 3D points in C_{map} , $\mathbf{R}_{C_{\text{map}}W}$ is the rotation matrix that transforms points from the world frame W to the map camera frame C_{map} and $\mathbf{t}_{C_{\text{map}}W}$ is the translation vector from the map camera origin to the world origin. Next, the points are orthographically projected onto the 2D map image and their pixel coordinates ($u_{\text{proj}} \ v_{\text{proj}}$) are given by:

$$\begin{bmatrix} u_{\text{proj}} \\ v_{\text{proj}} \end{bmatrix} = \frac{1}{p_{\text{map}}} \begin{bmatrix} x^{C_{\text{map}}} \\ y^{C_{\text{map}}} \end{bmatrix} + \begin{bmatrix} c_{x,\text{map}} \\ c_{y,\text{map}} \end{bmatrix} \quad (4.15)$$

where $p_{\text{map}} = p_{\text{win}}$ is the pixel resolution of the map, and $c_{x,\text{map}}$ and $c_{y,\text{map}}$ are the coordinates of the map camera's principal point. Given a set of map windows that have passed the depth and shadow check for a specific query that has also passed the same checks, the percentage of map-projected points with coordinates ($u_{\text{proj}} \ v_{\text{proj}}$) that fall within the window bounds, relative to the total number of pixels in the window, determines the amount of overlap in the window/query pair. If this overlap does not exceed a threshold (set to 25% for our dataset), the pair is discarded. The retained image pairs, along with their depth information, form the LoFTR training pairs, with associated metadata including query pose, window location in the maps, and depth and shadow masks. Figure 4.8 shows three training pairs examples.

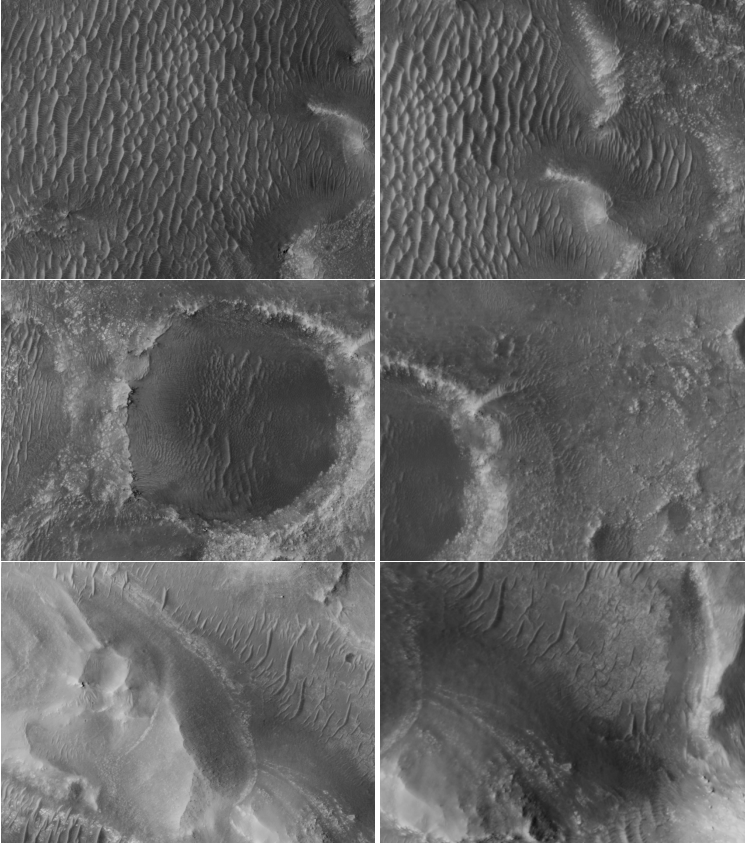


Figure 4.8: Examples of LoFTR training pairs formed by map windows with different lighting (*left*) and query observations (*right*) at 180° Sun azimuth (AZ) and 40° Sun elevation (EL). Maps are shown with 0° AZ and 30° EL (*top*), 60° EL (*mid*), 90° EL (*bottom*).

The training set used in this work is formed by combining 17 rendered maps with Sun azimuth in the range $[0^\circ, 360^\circ]$, with 45° increment, and elevation angles of 30° , 60° and 90° , with 4500 nadir-pointing aerial observations in the 64-200m altitude range, resulting in a total of 150,705 training pairs. The remaining maps at 2° , 5° and 10° in the same azimuth range, correspond to illumination conditions observed in brief portions of the local Sun’s trajectory on a typical day on the Jezero Crater site (see Figure 5.17 in Section 5.4). Therefore, they will be kept as challenging lighting test cases during our MBL evaluation in Chapter 5. Table 4.5 summarizes the properties of the image dataset sourcing the training pairs generation. We trained LoFTR and Geo-LoFTR on the 150,705 training pairs with a batch size set to 4, for six days on the High-Performance Computing clusters of the NASA Jet Propulsion. An independent validation set of 3,400 pairs has been used to regularly assess model performance during training and prevent

over fitting.

Parameters	Maps	Observations
N.o. gray images	17	4500
depth images	17	4500
Image size	$26,949 \times 57,613$	480×640
Pixel resolution	0.25 m / pixel	[0.25, 0.78] m / pixel
Projection type	Orthographic	Perspective
Camera orthographic scale	6737 m	/
Camera focal length	/	32 mm
sensor width	/	80 mm
Clipping plane near range	100 m	0.5 m
far range	100,000 m	100,000 m
N.o. tiles along width	8	1
along height	16	1
Camera location in world frame (t_{WC_x}, t_{WC_y})	(0,0) m	sampled from uniform random distribution of valid terrain points
Camera altitude	4000 m	[64, 200] m
Camera yaw, pitch, and roll in world frame ((γ, β, ϕ))	(0, 0, 0) $^\circ$	(0, 0, 0) $^\circ$
Sun Azimuth angles (AZ)	[0, 360] $^\circ$ with 45 $^\circ$ steps	180 $^\circ$
Elevation angles (EL)	30 $^\circ$, 60 $^\circ$, 90 $^\circ$	

Table 4.5: Parameters for the maps and observations image dataset sourced for the generation of the pairs formed by query observations and map windows, used for LoFTR and Geo-LoFTR training.

5 Map-based Localization Evaluation

This chapter presents a comprehensive evaluation of the performance of the Fine-tuned LoFTR and Geo-LoFTR models in our MbL pipeline, using multiple synthetic Mars datasets generated from the Jezero Crater site within the MARTIAN framework. To ensure an unbiased assessment, none of the test data overlaps with the data used for training. We conducted several experiments to assess the robustness of these methods under changes in lighting, scale and terrain morphologies. The evaluation involved registering nadir-pointing test observations against a 1 km^2 search area cropped from the map. The choice of a relatively large predetermined size for the map search area, rather than one dynamically adjusted based on the pose prior, serves two purposes. First, it allows us to simulate a conservative scenario with a high-uncertainty pose prior. Second, it prevents our evaluation strategy from being overly influenced by variations in the poses themselves, thus ensuring a more consistent assessment of our pipeline’s performance across experiments. Throughout the experiments, the search area was divided into multiple overlapping windows, each sized at 1024×768 pixels and with an overlap of 122 pixels. Each query observation was paired only to the map windows with more than 10% valid depth pixels. The formed pairs were then processed by the LoFTR models, which output matched keypoints on both the map windows and the query observations, along with corresponding confidence scores. The top 500 matches were retained for each map window, with further filtering applied across the entire search area to include only matches meeting a 95% confidence threshold. The refined matches on the query images, the 3D locations of the corresponding ones on the map, and the query camera intrinsics matrix were then fed to a RANSAC-PnP algorithm to retrieve the camera pose $(\tilde{\mathbf{R}}_{WC_{\text{query}}}, \tilde{\mathbf{t}}_{WC_{\text{query}}})$ in the world frame W , with a pixel reprojection error threshold of 1.5 pixels and 1000 iterations.

For benchmarking, we compared our results to the original LoFTR model pre-trained on the MegaDepth dataset and to SIFT, identified by [31] as the most accurate feature matching method for absolute localization over simulated Mars terrain. In the SIFT-based MbL evaluation, up to 4000 SIFT features were extracted from each query image and matched against those from the map, limited to the relevant search area. Feature matching was performed using k-nearest neighbors (KNN) with $k=2$. To ensure robustness, a ratio test was applied, retaining matches only if the ratio of the distance of the closest match

to the distance of the second-closest match was below 0.75.

The following sections detail various aspects of the evaluation. In Section ?? we assessed the MbL performance under a consistent offset in Sun azimuth and elevation angles between the map and the observations. Section 5.2 explores the impact of different terrain morphologies on localization accuracy. Section ?? examines MbL robustness to variation in Sun azimuth and elevation angles. Finally, Section 5.4 investigates model performance throughout different times of a simulated Martian day at the Jezero Crater site.

5.1 Localization with Fixed Lighting Offset

In this evaluation, the MbL performance has been assessed by registering 500 different nadir-pointing test observations with Sun angles set to $AZ = 180^\circ$ and $EL = 40^\circ$, against the search area on an orthographic map rendered with Sun angles set to $AZ = 0^\circ$ and $EL = 5^\circ$. The observations were generated using the MARTIAN framework, with their locations (xy-coordinates in the world frame) randomly sampled across the entire DTM and their altitudes uniformly drawn from a random distribution in the range [65, 200] m. The matching accuracy of the models was assessed by computing the cumulative distribution of the pixel reprojection errors of the matched keypoints. Given the pixel coordinates $(u_{\text{query}}, v_{\text{query}})$ of a matched key-point in the query image plane, the pixel reprojection error is represented by the Euclidean distance from the matched map keypoint, at coordinates $(u_{\text{map}}, v_{\text{map}})$, projected into the query image and resulting in the 2D point of coordinates $(u_{\text{query, proj.}}, v_{\text{query, proj.}})$. Starting from the 2D matched keypoint $(u_{\text{map}}, v_{\text{map}})$ and the corresponding depth value Z , the keypoint is first back-projected to its position, $\mathbf{x}^{C_{\text{map}}}$, in the local map camera frame C_{map} with the orthographic back-projection in Eq. 5.1 and then transformed to the world frame W using Eq. 5.2. The resulting 3D point, $\mathbf{x}^W$, is subsequently transformed into the local query camera frame C_{query} (Eq. 5.3), resulting in the 3D position $\mathbf{x}^{C_{\text{query}}}$ in the query camera frame. Finally $\mathbf{x}^{C_{\text{query}}}$ is projected to the query image plane using the perspective projection equations in Eq. 5.4, giving the query-projected coordinates $(u_{\text{query, proj.}}, v_{\text{query, proj.}})$.

$$\mathbf{x}^{C_{\text{map}}} = \begin{bmatrix} (u_{\text{map}} - c_{x,\text{map}}) \cdot p_{\text{map}} \\ (v_{\text{map}} - c_{y,\text{map}}) \cdot p_{\text{map}} \\ Z \end{bmatrix} \quad (5.1)$$

$$\mathbf{x}^W = \mathbf{R}_{WC_{\text{map}}} \mathbf{x}^{C_{\text{map}}} + \mathbf{t}_{WC_{\text{map}}} \quad (5.2)$$

$$\mathbf{x}^{C_{\text{query}}} = \mathbf{R}_{WC_{\text{query}}}^T (\mathbf{x}^W - \mathbf{t}_{WC_{\text{query}}}) \quad (5.3)$$

$$\begin{bmatrix} u_{\text{query, proj.}} \\ v_{\text{query, proj.}} \\ 1 \end{bmatrix} = \mathbf{K} \mathbf{x}^{C_{\text{query}}} \quad (5.4)$$

The transformations above make use of true observation and map camera poses $(\mathbf{R}_{WC_{\text{map}}}, \mathbf{t}_{WC_{\text{map}}})$ and $(\mathbf{R}_{WC_{\text{query}}}, \mathbf{t}_{WC_{\text{query}}})$ in the world frame. $c_{x,\text{map}}$ and $c_{y,\text{map}}$ are the

coordinates of the orthographic camera’s principal in the image plane, p_{map} is the map resolution and $\mathbf{K}$ is the query camera intrinsics matrix in Eq. 4.12. The localization error is simply given by $\|\mathbf{t}_{WC_{\text{query}}} - \tilde{\mathbf{t}}_{WC_{\text{query}}}\|$. The cumulative distributions of the pixel reprojection error and localization error are shown for the three different LoFTR models in Figure 5.1.

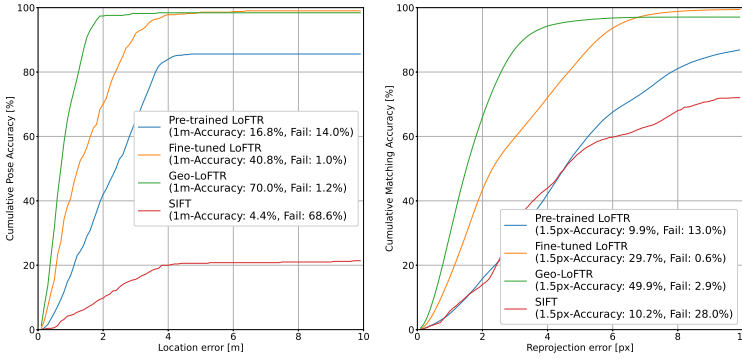


Figure 5.1: . Cumulative distribution of the localization error (*left*) and pixel reprojection error (*right*) for the MbL of 500 test observations of the Jezero Crater HiRISE map with different image matching models, and with consistent Sun AZ and EL offsets of 180° and 35° from the ortho-rectified map. Observations are generated in MARTIAN (nadir-pointing, altitude 64-200 m) with reference Sun AZ= 180° and EL= 40° .

The three LoFTR models largely outperform SIFT in localization accuracy. Both Geo-LoFTR and fine-tuned LoFTR show considerable performance improvement over the pre-trained model, in terms of the percentage of observations with localization error at the 1 m precision level (1m-Accuracy), and the percentage of observations that failed to be registered on the map. Despite the slightly higher failure rate, Geo-LoFTR led to an improvement of about 29% and 53% in 1m-Accuracy over the fine-tuned and the pre-trained models, respectively, suggesting that the MbL benefited from the geometric information present in the depth data. The improvement over SIFT is larger than 65% in 1m-Accuracy. Figure 5.2 shows examples of matched keypoints on a sampled observation and the map search area with fixed Sun azimuth and elevation offsets for the pre-trained LoFTR, fine-tuned LoFTR, Geo-LoFTR and SIFT.

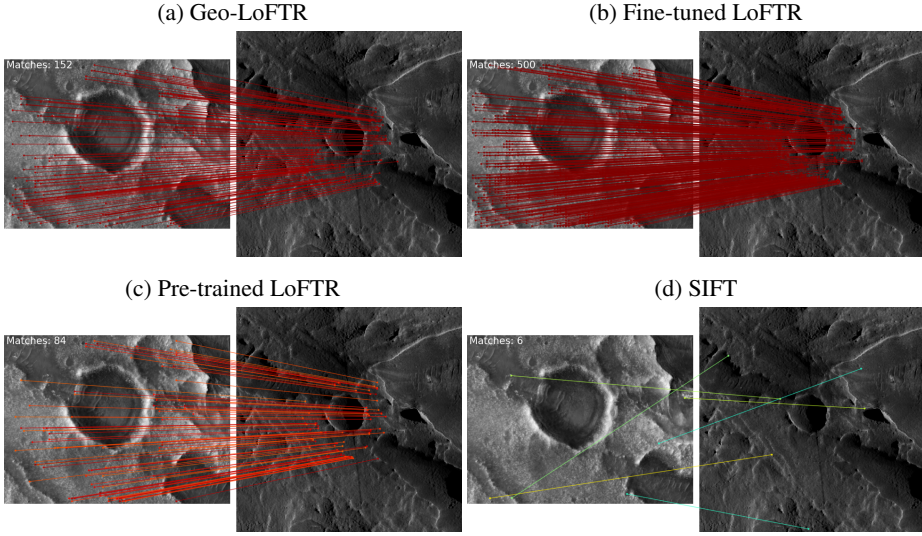


Figure 5.2: . Matched keypoints between a sample query image (*left side of each panel*) and a map search area image (*right side of each panel*) for the Jezero Crater, with fixed Sun azimuth (AZ) and elevation (EL) angles offsets. Each panel shows matches using different image matching models: (a) Geo-LoFTR, (b) Fine-tuned LoFTR, (c) Pre-trained LoFTR, and (d) SIFT. The query observation was generated in the MARTIAN framework (nadir-pointing, altitude 162 m) with a reference Sun AZ of 180° and EL of 40° . The ortho-rectified map was generated with Sun AZ = 0° and EL = 5° .

To assess the models' robustness to scale variation, the localization accuracy for the same test set is evaluated for subsets of queries within three different altitude sub-ranges. The results are shown in Figure 5.3, with the same lighting offset between map and observations as in Figure 5.1. The localization failure rates and accuracy at the precision levels of 0.5 m, 1 m, 2 m and 5 m are reported for the different matching methods and altitude range in Table 5.1. Geo-LoFTR outperforms the other methods in accuracy across the altitude sub-ranges and precision levels, except at 5m-Accuracy where the fine-tuned model is the most performant. Fine-tuned LoFTR achieves slightly better performance in terms of localization failure rate. Moreover, Geo-LoFTR demonstrates greater robustness to altitude variations than the fine-tuned model, with a drop of only 10% in 1m-Accuracy between the 156-199 m and 64-122 m ranges, compared to a 24% accuracy drop of the fine-tuned LoFTR. The variation in accuracy with altitude of the fine-tuned LoFTR seems more prominent than the geometry-aided counterpart also at the 0.5 m and 2 m precision levels, suggesting that providing a geometric context contributes to scale invariance. A possible explanation is that depth data constrains matches by providing consistent pixel-to-pixel depth relationships across altitudes, reflecting terrain elevation alone. This added layer of geometric consistency likely enhances Geo-LoFTR's ability to learn accurate matches by reducing ambiguity from appearance-based features

alone.

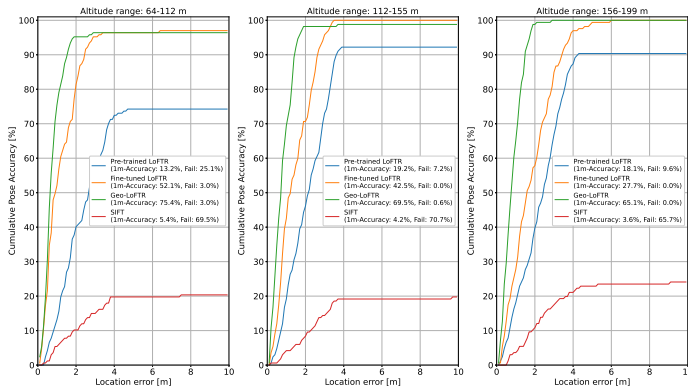


Figure 5.3: . Cumulative distributions of the localization error for different matching methods used in the MbL of 500 nadir-pointing test observations within the altitude ranges [64, 112] m (*left*), [113, 157] m (*middle*) and [159-198] m (*right*). The observations were generated in the MARTIAN framework and sampled from the Jezero Crater HiRISE map, with reference Sun AZ=180° and EL=40°. The ortho-rectified map was generated with Sun AZ = 0° and EL = 5°.

5.2. Effect of Terrain Morphology on Localization Accuracy

Altitude range	Method	0.5 m	1 m	2 m	5 m	Fail
64-112 m	Geo-LoFTR	30.4%	75.4%	95.2%	96.3%	3.0%
	Fine-tuned LoFTR	21.6%	52.1%	81.3%	96.3%	3.0%
	Pre-trained LoFTR	4.1%	13.2%	40.1%	74.2%	25.1%
	SIFT	1.2%	5.4%	10.1%	19.7%	69.5%
112-155 m	Geo-LoFTR	34.8%	69.5%	98.2%	98.8%	0.6%
	Fine-tuned LoFTR	12.0%	42.5%	70.7%	100%	0.0%
	Pre-trained LoFTR	4.8%	19.2%	46.4%	92.2%	7.2%
	SIFT	0.6%	4.2%	8.6%	19.1%	70.7%
156-199 m	Geo-LoFTR	29.5%	65.1%	98.9%	99.9%	0.0%
	Fine-tuned LoFTR	12.6%	27.7%	58.0%	99.4%	0.0%
	Pre-trained LoFTR	6.6%	18.1%	40/0%	90.4%	9.6%
	SIFT	0.0%	3.6%	11.2%	22.9%	65.7%

Table 5.1: Localization accuracy at the precision levels of 0.5 m, 1 m, 2 m and 5 m for different matching methods used in the MbL of 500 nadir-pointing test observations in three altitude ranges. The observations are generated in the MARTIAN framework and sampled from the Jezero Crater HiRISE map, with reference Sun AZ=180° and EL=40°. The map is generated with Sun AZ = 0° and EL = 5°.

5.2 Effect of Terrain Morphology on Localization Accuracy

The effect of the terrain morphology on the MbL accuracy with different matching models is investigated by generating two sets of 600 nadir-pointing observations in two different areas of the Jezero Crater HiRISE map, uniformly sampled over the DTM at three fixed altitudes of 64 m, 100m and 200 m. One area is characterized by mid-to-high variations in elevation, such as cliffs, rocky outcrops or steep slopes: we dubbed this terrain *rugged*. The other area under consideration is mainly dominated by dunes and will be referred to as *dunal* terrain. Both the areas are shown in Figure 5.4, with the sampled locations of 200 nadir-pointing observations at 100 m altitude, which form 1/3 of the test dataset per terrain category. Sample observations generated in the MARTIAN framework for the two terrain morphologies are displayed at the different altitudes in Figure 5.5.

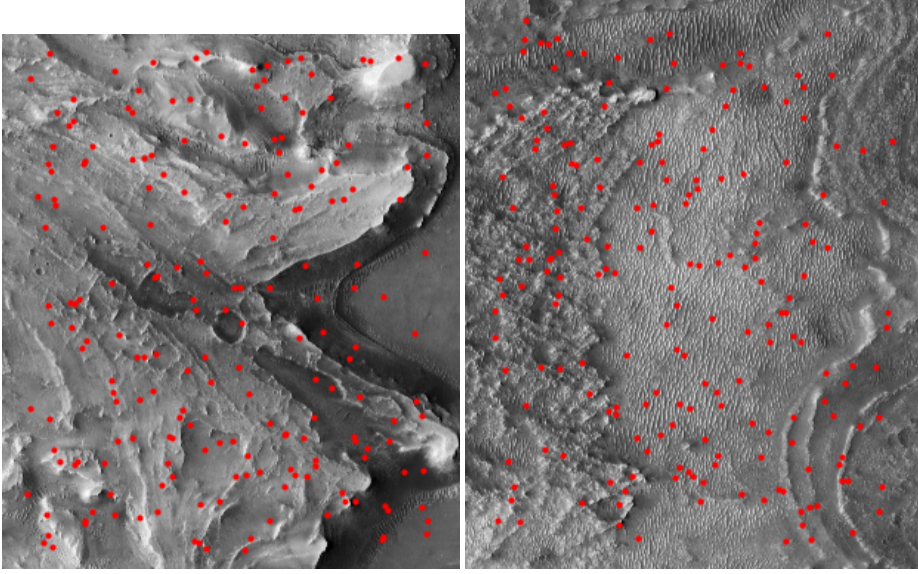


Figure 5.4: Locations of the 200 sampled nadir-pointing observations (altitude 100 m) on rugged (*left*) and dunal (*right*) terrains from the Jezero Crater HiRISE map. The images are generated with the MARTIAN framework at a Sun azimuth of 180° and elevation 40° .

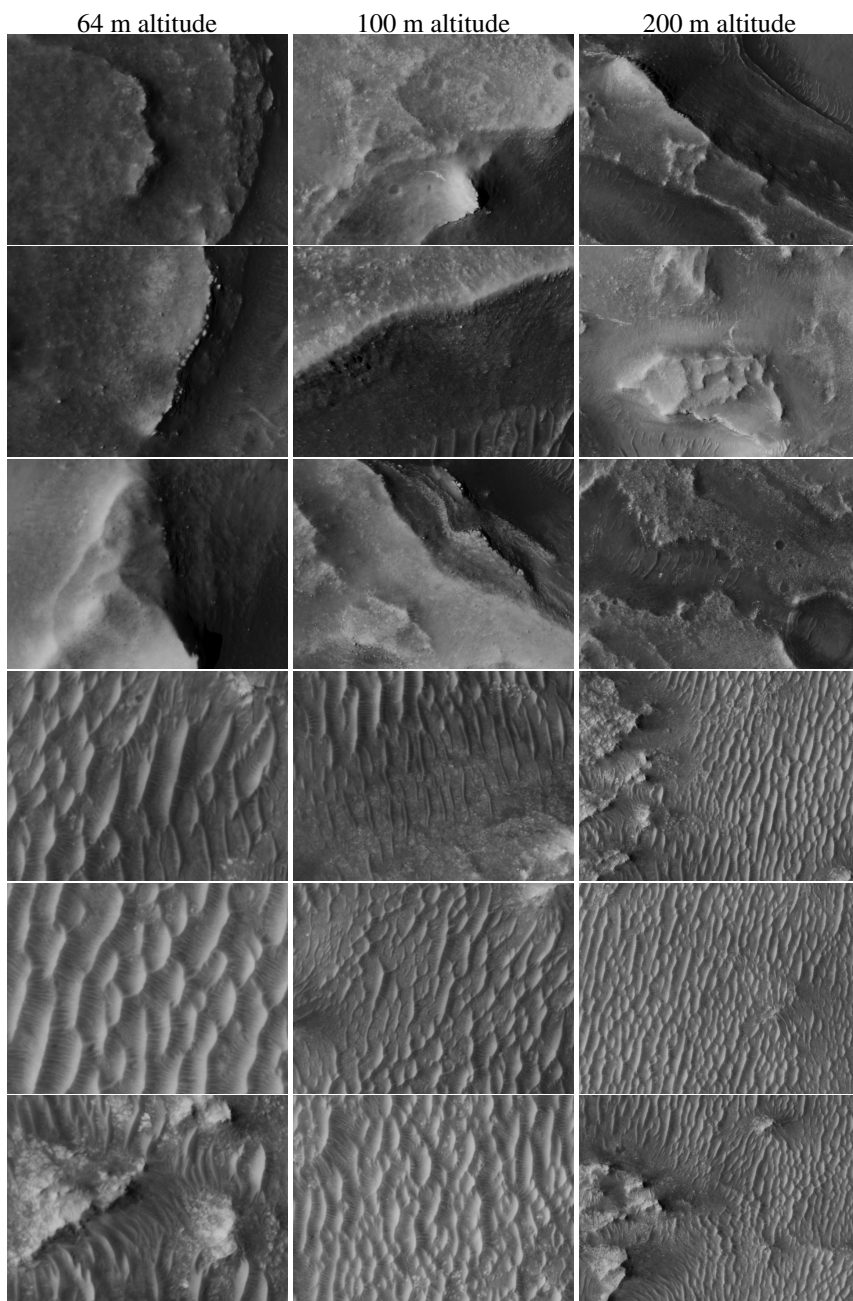


Figure 5.5: Sample nadir-pointing observations of rugged (*top rows*) and dunal (*bottom rows*) terrains on the Jezero Crater HiRISE map, at altitudes of 64 m (*left column*), 100 m (*middle column*) and 200 m (*right column*). The images are generated with the MARTIAN framework at Sun azimuth of 180° and elevation 40°.

Figure 5.6 reports the cumulative distributions of the localization error for the MbL of the test observations over both the rugged and dunal terrains at the different altitudes, under consistent sun azimuth and elevation offsets of 180° and 5° from the map. Figure 5.7 reports the localization accuracy at 0.5 m, 1 m and 2 m precision levels as varying with altitude under the same illumination configuration in Figure 5.6, for the two different terrain types. The values are reported in Table 5.2, along with localization failures.

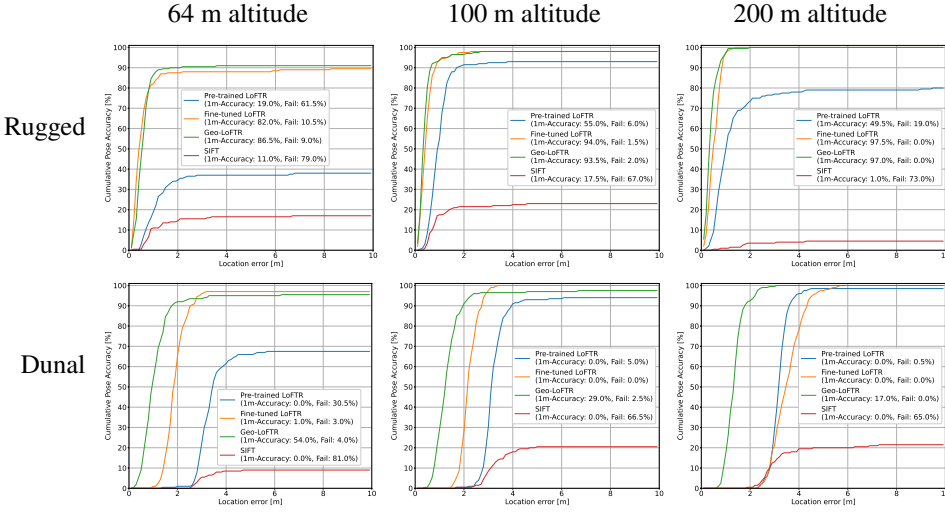


Figure 5.6: Cumulative distributions of the localization error for the MbL of 200 nadir-pointing observations (*each panel*) of rugged (*top row*) and dunal (*bottom row*) terrains on the Jezero Crater HiRISE map, at altitudes of 64 m (*left column*), 100 m (*middle column*) and 200 m (*right column*). Observations are generated in MARTIAN framework with Sun AZ= 180° and EL= 40° . The map is rendered at Sun AZ= 0° and EL= 5° .

Localization over the dunal terrain proves more challenging than over rugged terrain across different matching methods. This is because the repetitive patterns formed by the dunes are more likely to produce ambiguous matches due to the lack of unique, distinctive visual features. However, analyzing the 1m and 2m precision levels (Figure 5.7), the decline in localization accuracy over the dunal terrain compared to the rugged terrain is more significant for the matching models relying solely on visual information. For instance, at 100 m altitude, the drop in the 1m-Accuracy is approximately -64% for Geo-LoFTR and -94% for Fine-tuned LoFTR. Similar behavior is observed at 64 m (-32.5 % Geo-LoFTR, -81 % Fine-tuned LoFTR) and partially at 200 m (-80.5 % Geo-LoFTR, -97.5 % Fine-tuned LoFT), indicating Geo-LoFTR's greater robustness to terrain morphology than its visual counterpart. This might be explained by the depth information contributing to rule out degeneracy in feature matching caused by the repetitive dune patterns. This robustness appears more clear at the 2 m precision level, where the localization accuracy gap between the dunal and rugged terrains is bounded within a 10% range for Geo-LoFTR across 64-200 m altitudes. In contrast, the fine-tuned

model shows a significant accuracy drop from -26% at 64 m to -92% at 200 m, revealing less robustness to scale variation compared to the geometric method. Figure 5.7 shows these accuracies also for the cases with no Sun angles offsets between map and queries. The introduction of a lighting offset in the rugged terrain reduces the performance of all the methods at 1 m and 2 m precision levels, especially at lower altitudes. In the dunal terrain, lighting offsets affect the 2m-Accuracy to a greater extent for the fine-tuned model compared to Geo-LoFTR. Notably, SIFT exhibits the largest performance degradation with Sun angle offsets on the rugged terrain, and both SIFT and the pre-trained LoFTR completely fail on the dunal terrain. Figure 5.8 and Figure 5.9 show the matched keypoints produced by the different methods on the map search area and sample observations for the rugged and dunal terrains, respectively, at multiple altitudes. In summary, LoFTR models trained on the Mars dataset enhance localization accuracy over both terrain types compared to the pre-trained LoFTR and SIFT. Additionally, the integration of depth information in Geo-LoFTR appears to increase robustness to terrain morphology and scale variations beyond what visual feature representations alone can achieve.

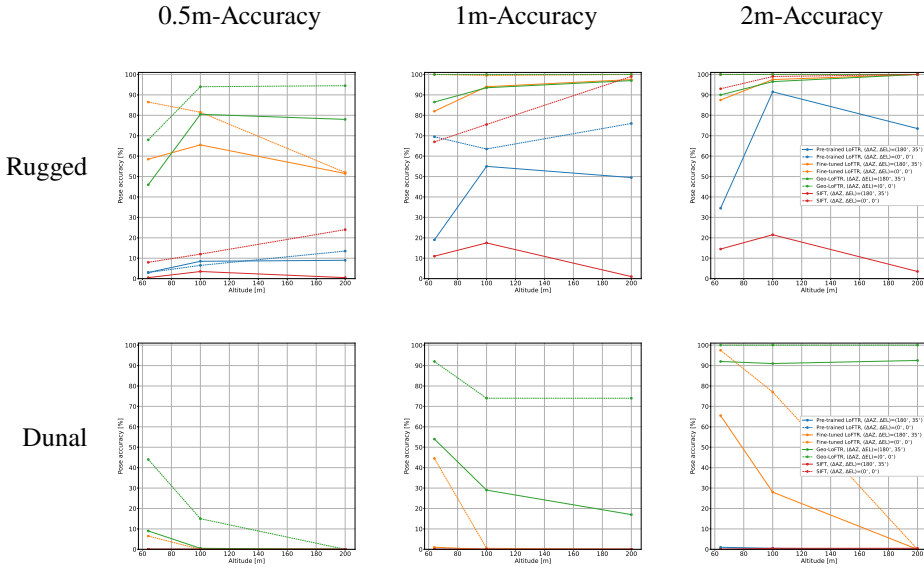
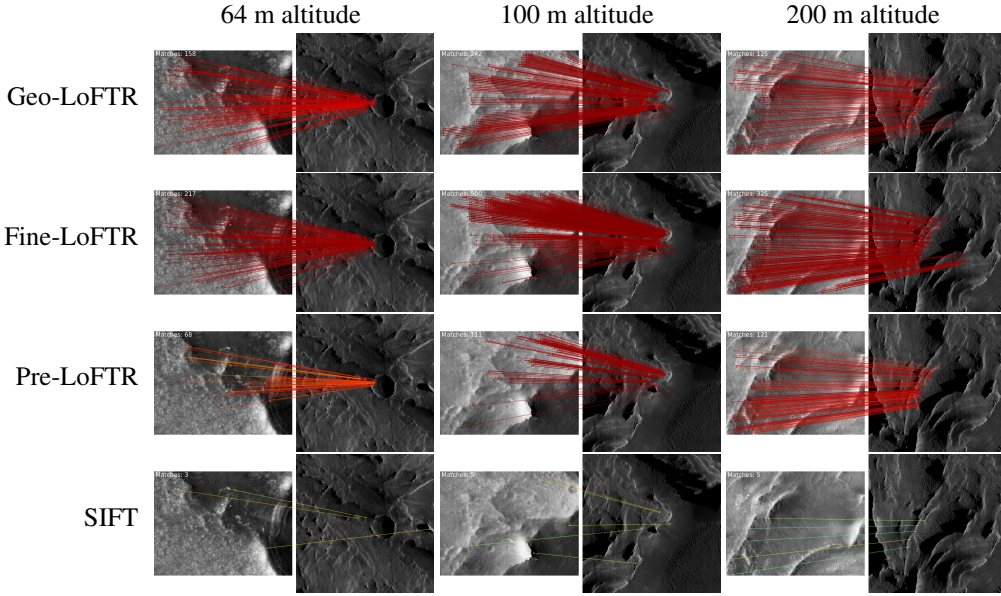


Figure 5.7: Variation of localization accuracy with altitude for the MbL of nadir-pointing observations over rugged (*top row*) and dunal (*bottom row*) terrains on the Jezero Crater HiRISE. The accuracy is addressed at the precision level of 0.5 m (*left column*), 1 m (*middle column*) and 2 m (*right column*). Observations are generated in MARTIAN framework with Sun AZ=180° and EL=40°. The map is rendered at Sun AZ=0° and EL=5°.

Terrain	Altitude	Method	0.5 m	1 m	2 m	5 m	Fail
rugged	64 m	Geo-LoFTR	46.0%	86.5%	90.0%	90.0%	9.0%
		Fine-tuned LoFTR	58.5%	82.0%	87.5%	87.5%	10.5%
		Pre-trained LoFTR	3.0%	19.0%	34.5%	34.5%	61.5%
		SIFT	0.5%	11.0%	14.5%	14.5%	79.0%
	100 m	Geo-LoFTR	80.5%	93.5%	96.5%	96.5%	2.0%
		Fine-tuned LoFTR	65.5%	94.0%	97.5%	97.5%	1.5%
		Pre-trained LoFTR	8.5%	55.0%	92.2%	91.5%	6.0%
		SIFT	3.5%	17.5%	21.5%	21.5%	67.0%
	200 m	Geo-LoFTR	78.0%	97.0%	100.0%	100.0%	0.0%
		Fine-tuned LoFTR	51.5%	97.5%	100.0%	100.0%	0.0%
		Pre-trained LoFTR	9.0%	49.5%	91.5%	73.5%	19.0%
		SIFT	0.5%	1.0%	3.5%	3.5%	73.0%
dunal	64 m	Geo-LoFTR	9.0%	54.0%	92.0%	95.0%	4.0%
		Fine-tuned LoFTR	0.0%	1.0%	65.5%	97.0%	3.0%
		Pre-trained LoFTR	0.0%	0.0%	1.0%	66.0%	30.5%
		SIFT	0.0%	0.0%	0.0%	9.0%	81.0%
	100 m	Geo-LoFTR	0.5%	29.0%	91.0%	97.0%	2.5%
		Fine-tuned LoFTR	0.0%	0.0%	28.0%	100%	0.0%
		Pre-trained LoFTR	0.0%	0.0%	0.5%	93.0%	5.0%
		SIFT	0.0%	0.0%	0.5%	20.5%	66.5%
	200 m	Geo-LoFTR	0.0%	17.0%	92.5%	100.0%	0.0%
		Fine-tuned LoFTR	0.0%	0.0%	0.0%	97.5%	0.0%
		Pre-trained LoFTR	0.0%	0.0%	0.0%	98.5%	0.5%
		SIFT	0.0%	0.0%	0.5%	20.0%	65.0%

Table 5.2: Localization accuracy at the precision levels of 0.5 m, 1 m, 2 m and 5 m for different matching methods used in the MbL of 200 nadir-pointing test observations at altitudes of 64 m, 100 m and 200 m, for the rugged and dunal terrain morphologies. The observations are generated in the MARTIAN framework and sampled from the Jezero Crater HiRISE map, with reference Sun AZ=180° and EL=40°. The map is generated with Sun AZ = 0° and EL = 5°



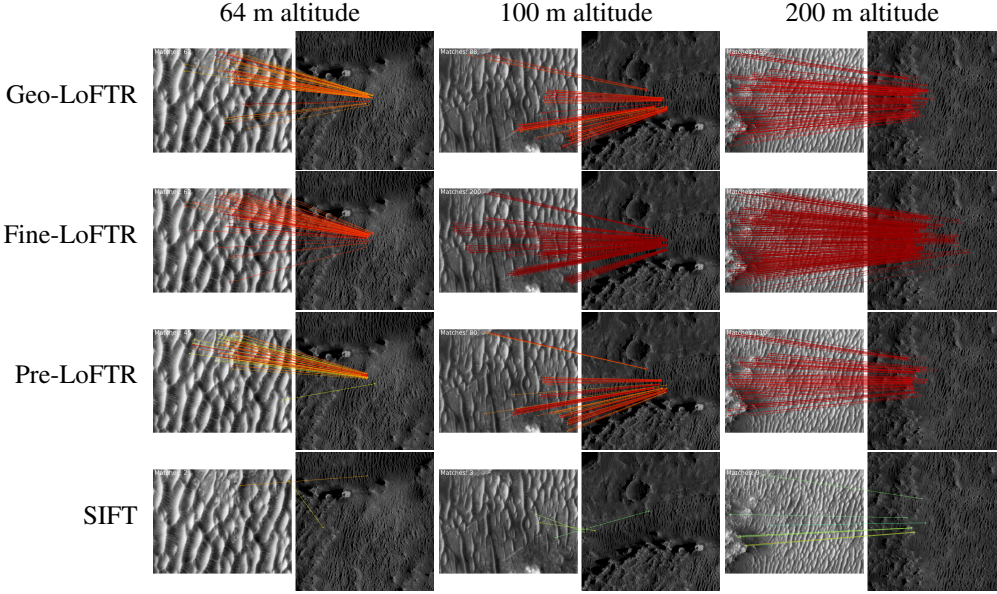


Figure 5.9: Matched keypoints between observations (*left, in each panel*) of dunal terrain and map (*right, in each panel*) for the Jezero Crater HiRISE map, at altitudes of 64 m (*left column*), 100 m (*middle column*) and 200 m (*right column*). The image matching models are: Geo-LoFTR (*1st row*), pre-trained LoFTR (*2nd row*), fine-tuned LoFTR (*3rd row*) and SIFT (*4th row*). Observations are generated in MARTIAN framework with Sun AZ=180° and EL=40°. The map is rendered at Sun AZ=0° and EL=5°.

5.3 Robustness to Changing Solar Angles

In this section, we investigate the robustness of the matching models to MbL under varying Sun angles. To study the effect of the Sun elevation angles only, we registered the test set of 500 nadir-pointing observations at Sun AZ=180° and EL=40° (used in Section 5.1) onto different maps taken at the same azimuth of the observations and elevations equal to 2°, 5°, 10°, 30°, 60° and 90°. It is important to note that elevations below 30° were not encountered by the Geo-LoFTR and Fine-tuned LoFTR models during training. Moreover, for a Martian day at the Jezero Crater site, offsets such as on 2016-05-24 (i.e., the date of one of the HiRISE images used in this work), the Sun elevation varied in the range [7.9°, 82.6°] from 6:00 to 17:00 Local Mean Solar Time (LMST), with sunrise and sunset occurring at 05:11 LMST and 17:32 LMST respectively. Therefore, elevations below 10° occupy a very brief portion of Sun’s local trajectory, representing exceptional cases in the Martian surface operations. Nevertheless, we chose to include these cases in our evaluation to assess the models’ ability to generalize and perform effectively under challenging lighting conditions. The cumulative distributions of the observations’ localization error at six different sun elevation offsets are reported for

all the matching methods in Figure 5.10. Figure 5.11 shows the variation of localization accuracy with the sun elevation of the map at different precision levels. Geo-LoFTR outperforms all the other methods in localization accuracy across the entire range of sun elevation offsets and at all the precision levels except for the 5m-Accuracy, where its performance is comparable to the fine-tuned model. Analyzing the 1 m precision level, Geo-LoFTR is 87% accurate in the case of 0° offset in sun EL between map and observations, showing an improvement of +38% over the fine-tuned model, and +63% over SIFT. Below 10° EL of the map (-30° offset from the reference observation), the performance of all the methods is significantly affected, with Geo-LoFTR being 17% accurate at the very challenging lighting case of map EL= 2° , and still more accurate than Fine-tuned LoFTR (9%), the pre-trained model (2%) and SIFT (0.6%). However, Fine-tuned LoFTR shows the best failure localization rate overall, with a 5.8% and 46.8% rates at the low map sun elevations of 2° and 10° , respectively, compared to 7.0% and 53.6% for Geo-LoFTR. Figure 5.12 shows the matched keypoints on observations and the map found by Geo-LoFTR and SIFT, which result in the best and worst methods in terms of 1m-accuracy over the entire set of sun elevation offsets. Four different map sun elevations are displayed in the figure.

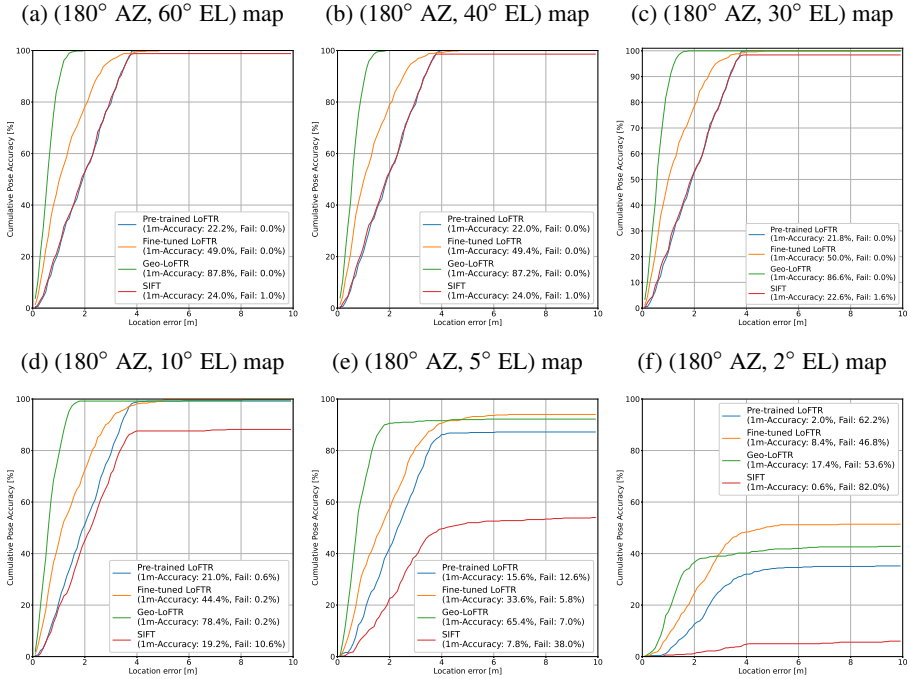


Figure 5.10: Cumulative distributions of the localization error for the MbL of 500 nadir-pointing observations (altitude 64-200 m) at reference Sun AZ= 180° and EL= 40° , and maps at six different Sun elevation angles and fixed azimuth. Test observations and maps are generated from the Jezero Crater site in the MARTIAN framework.

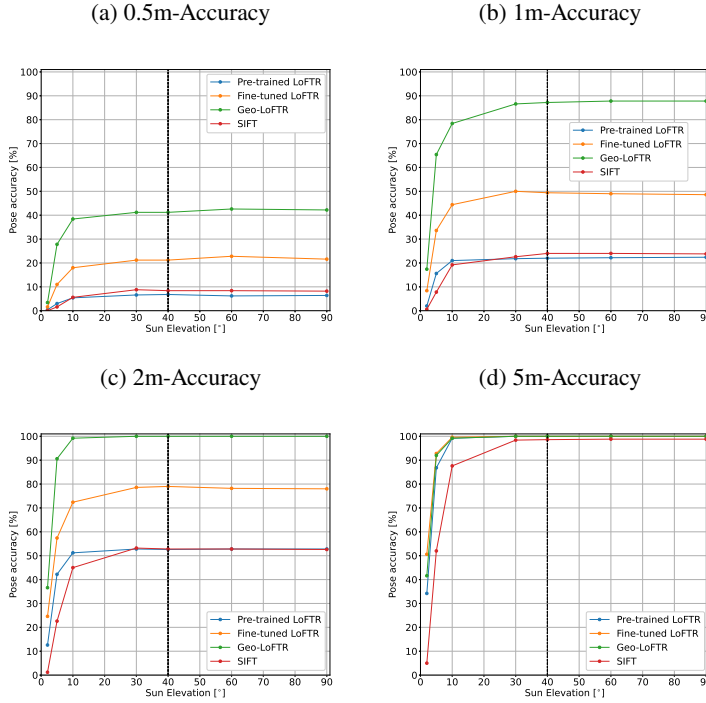


Figure 5.11: Variation of localization accuracy with map Sun elevation, for the MbL of 500 nadir-pointing observations (altitude 64-200 m) at reference Sun EL=40° (*dashed line*). The localization accuracy is shown at the precision level of 0.5m (a) , 1 m (b), 2 m (c) and 5 m (d). Queries and maps are generated from the Jezero Crater site with the MARTIAN framework, with a consistent Sun azimuth angle of 180°.

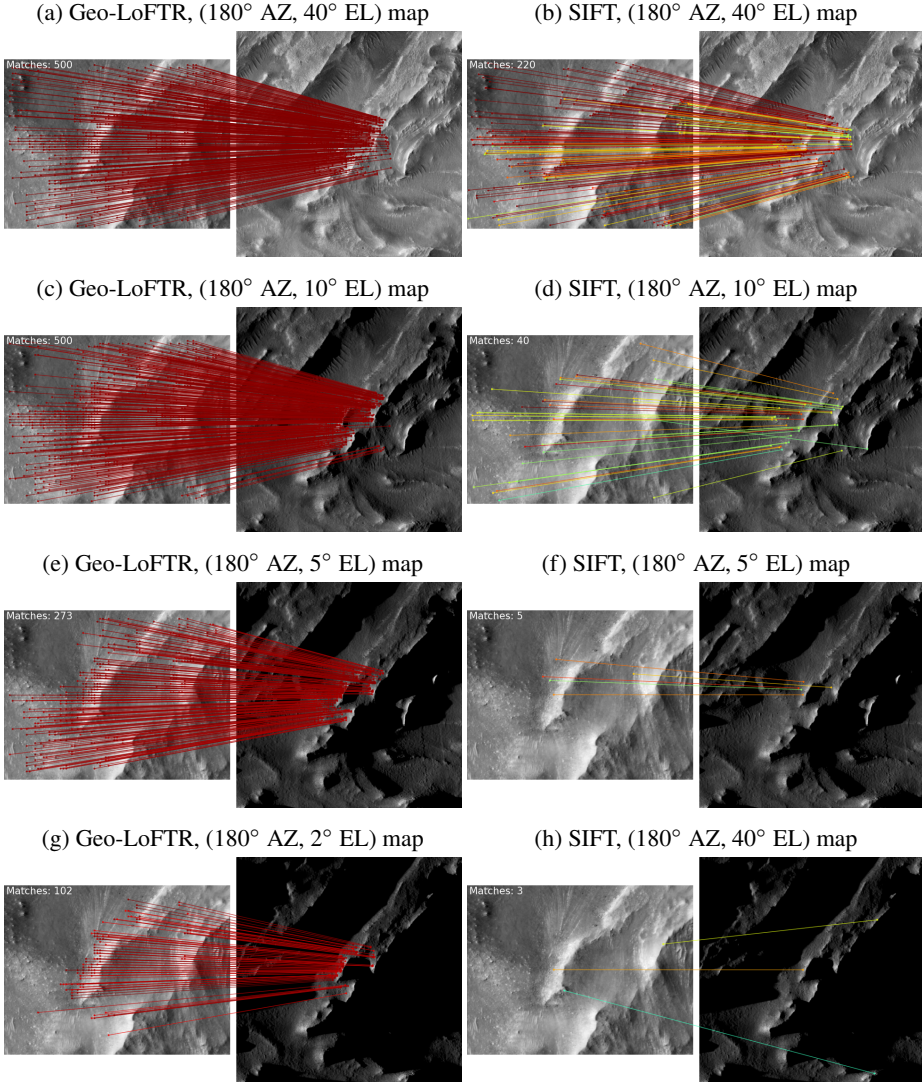


Figure 5.12: Geo-LoFTR (*left column*) and SIFT (*right column*) matched keypoints displayed for a sample query image (*left side of each panel*) and a map search area image (*right side of each panel*) under four different Sun elevation angles and fixed azimuth. The query observation (nadir-pointing, altitude 142.67 m) is generated from the Jezero crater site in the MARTIAN framework with a reference Sun AZ of 180° and EL of 40°. Match lines are color-coded by confidence score, with red indicating higher confidence and blue indicating lower confidence.

To study the effect of sun azimuth on localization accuracy, we generated a different

set of 600 nadir-pointing observations in the 64-200 m altitude range, under 0° Sun AZ and 10° EL to allow the shadow patterns to be more affected by change in azimuth. Different maps are generated at 10° EL and azimuth varying in the 0° - 360° range with 45° steps. Figure 5.13 shows the cumulative distributions of the localization error for the MBL of the test observations at six different sun azimuth offsets. Figure 5.14 shows the variation of localization accuracy with the sun azimuth of the map at the precision levels of 0.5 m, 1 m, 2 m and 5 m. The azimuth angles are reported in the range $[-180^\circ, 180^\circ]$ for visual clarity. Also in this experiment, Geo-LoFTR outperforms all the other methods, except for the 5m-Accuracy, where Fine-tuned LoFTR performs slightly better. Compared to the effect of the sun elevation variation, the accuracy of the LoFTR models trained on the Mars synthetic data results more robust to map sun azimuth offsets from the reference observation, with the 2m-Accuracy varying in a bandwidth of 8.5% for Geo-LoFTR and 12.5% for Fine-Tuned LoFTR. Such robustness is not observed for the pre-trained LoFTR model and SIFT, with a 2m-Accuracy varying in a bandwidth of 12% for the former and 38.5% for the latter. The matched keypoints on a sample observation and map search area are displayed for Geo-LoFTR and SIFT in Figure 5.15.

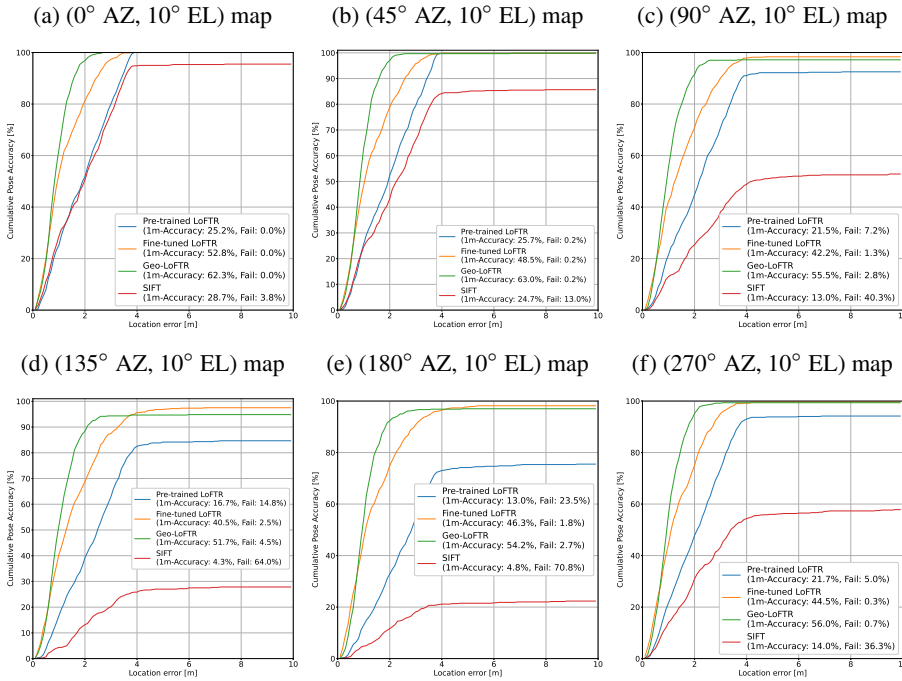


Figure 5.13: Cumulative distributions of the localization error for the MBL of 500 nadir-pointing observations (altitude 64-200 m) at reference Sun AZ= 0° and EL= 10° , and maps at six different Sun azimuth angles and fixed elevation. Test observations and maps are generated from the Jezero crater site in the MARTIAN framework.

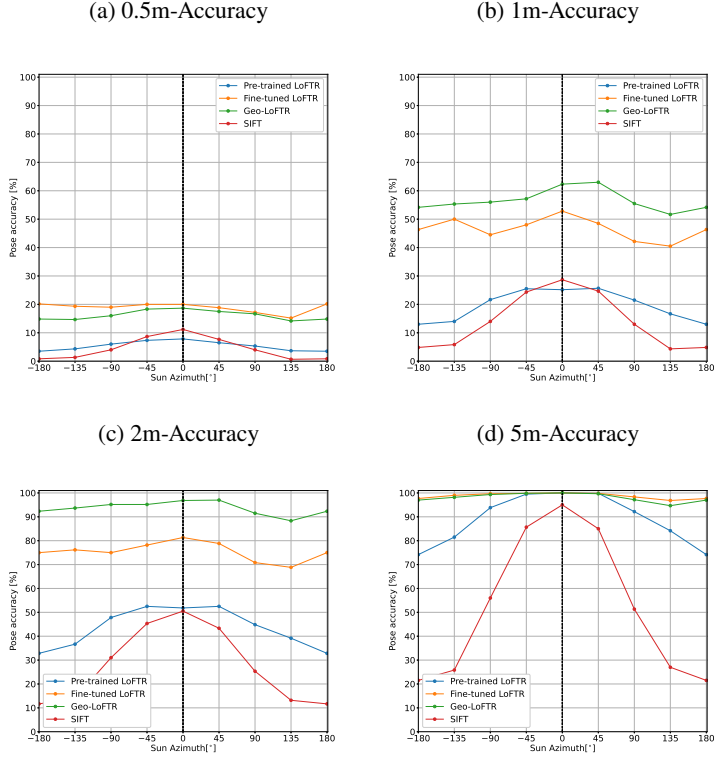


Figure 5.14: Variation of the localization accuracy with map Sun azimuth, for the MbL of 500 nadir-pointing observations (altitude 64-200 m) at reference Sun AZ=0° (*dashed line*). The localization accuracy is shown at the precision level of 0.5m (a) , 1 m (b), 2 m (c) and 5 m (d). Queries and maps are generated from the Jezero Crater site with the MARTIAN framework, with a consistent Sun elevation angle of 10°.

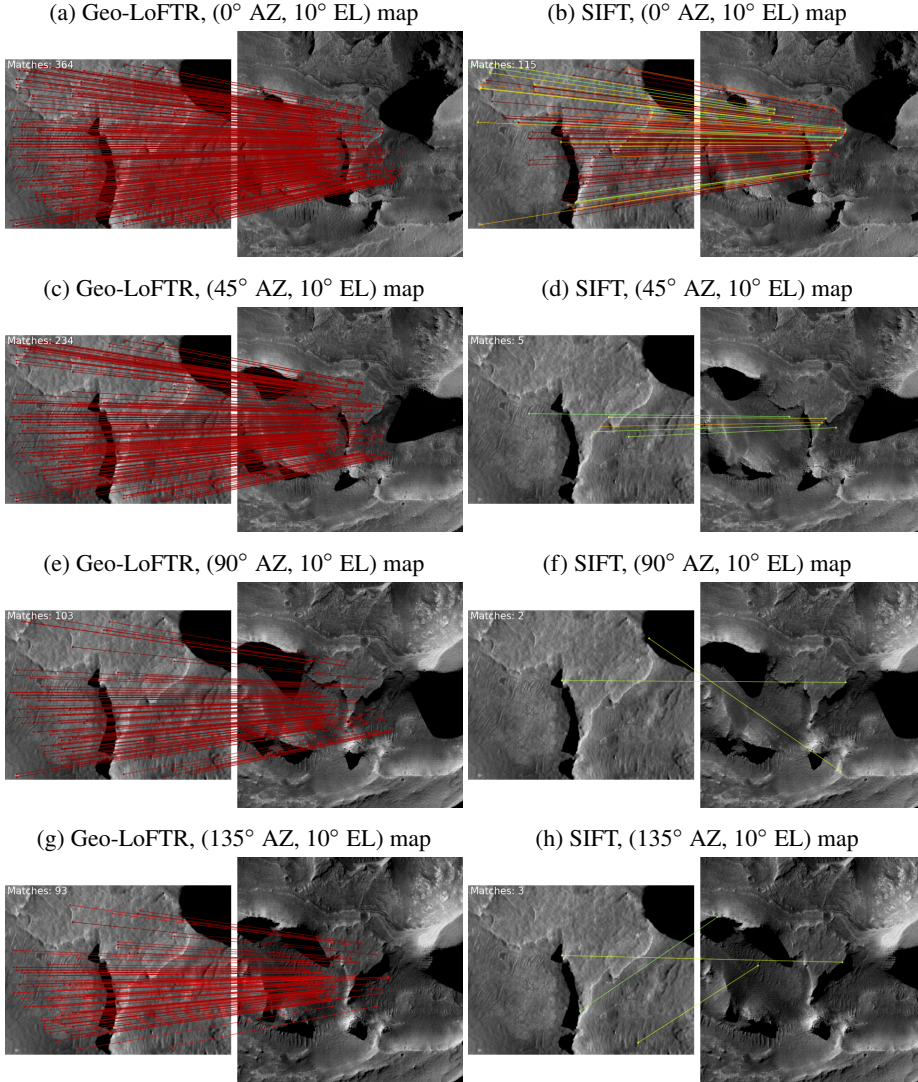


Figure 5.15: Geo-LoFTR (*left column*) and SIFT (*right column*) matched keypoints displayed for a sample query image (*left side of each panel*) and a map search area image (*right side of each panel*) under four different Sun azimuth angles and fixed elevation. The query observation (nadir-pointing, altitude 142.67 m) is generated from the Jezero crater site in the MARTIAN framework with a reference Sun AZ of 0° and EL of 10°. Match lines are color-coded by confidence score, with red indicating higher confidence and blue indicating lower confidence.

The impact of scale variation under lighting offsets in elevation and azimuth is also

investigated by inspecting the MbL accuracy at 1 m precision level of the test observations within three different altitude sub-ranges (Figure 5.16). Geo-LoFTR accuracy results are clearly more robust to altitude variations than the fine-tuned model over the entire set of sun elevation offsets. Picking a map sun elevation of 10° for instance, the drop in 1m-accuracy is only of -13% between the 156-199 m and 64-113 m altitude ranges for Geo-LoFTR, compared to a -28% accuracy drop for Fine-tuned LoFTR. The improved scale invariance of the geometry-aided model over the other methods is confirmed also under lighting variations in terms of sun azimuth offsets, thus providing yet another piece of evidence of the benefits introduced to localization accuracy by depth data.

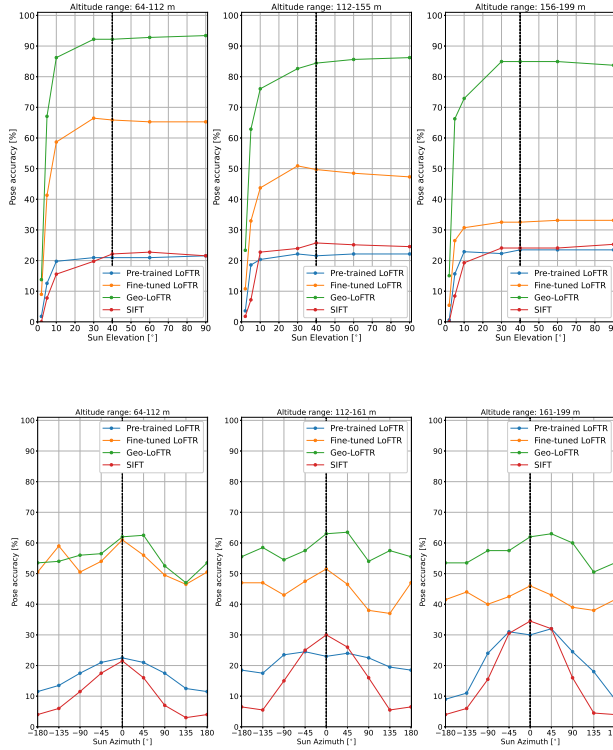


Figure 5.16: Localization accuracy at 1 m precision level as a function of map Sun elevation (*top*) and azimuth (*bottom*) for nadir-pointing observations across three altitude ranges. In the top row, accuracy is shown for maps with varying elevations at a fixed azimuth of 180° , with query observations set at $AZ=180^\circ$ and $EL=40^\circ$. In the bottom row, maps vary in azimuth at a fixed elevation of 10° , with query observations at $AZ=0^\circ$ and $EL=10^\circ$. Data are generated from Jezero Crater using the MARTIAN framework.

5.4 Localization Over a Simulated Martian Day

The MbL performance is investigated for observations taken at different Local Mean Solar Times (LMSTs) during a simulated Martian day on the Jezero Crater HiRISE map at (77.44°E, 18.43°N) coordinates. We used the Mars24 [3] software developed by NASA Goddard Institute for Space Studies, to compute the Sun’s local trajectory for the selected site for a given date. The chosen date, 2031-05-10, ensures that the Sun reaches a zenith at a relatively high elevation angle of 86.7°, allowing a broad range of elevation angles to be observed throughout the Martian day. The sun trajectory is shown in Figure 5.17, where the Sun azimuth is measured clockwise from North in Mars24, differently from our MARTIAN framework, where it is defined counterclockwise from East. The Sun elevation is defined with the same convention in both the Mars24 and the MARTIAN frameworks.

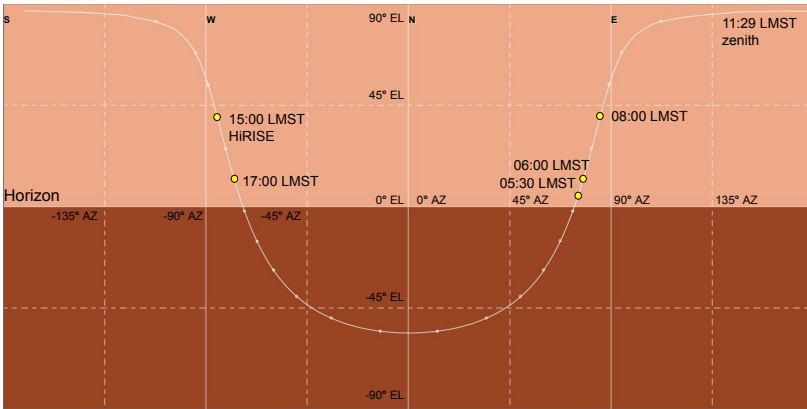


Figure 5.17: Sun trajectory on a local panorama from 77.44°E longitude and 18.43°N latitude on Mars, on 2031-05-10, with positions shown at four Local Mean Solar Times (LMSTs). Sun elevation varies in the range $[-90^\circ, 90^\circ]$ on the y-axis. The x-axis shows the sun azimuth measured clockwise from North, in the $[-180^\circ, 180^\circ]$ range. Adapted from Mars24 [3].

We generated nadir-pointing observations of the Jezero Crater site in our MARTIAN framework at different LMSTs highlighted in Figure 5.17, with 500 observations taken at each LMST in the 64-200 m altitude range, for a total of 3000 rendered grayscale images accompanied by 3000 corresponding depth images. We supposed that a HiRISE map is taken at 15:00 LMST, with the Sun at 39.9° elevation and 274.9° azimuth in the

Mars24 framework (175.1° in MARTIAN). An ortho-graphic map has been rendered in MARTIAN at these sun angles and used as a reference map for the MbL of the observations at different LMSTs. Figure 5.18 shows some observations sampled from the test sets at each LMST. Table 5.3 reports the LMSTs of the test points along with Earth time in UTC, Sun elevation, and Sun azimuth in both the Mars25 and MARTIAN frameworks. Sunrise and sunset points are included. The cumulative distributions of the MbL error at each LMST are shown for the different matching methods in Figure 5.19; while Figure 5.20 shows the localization error at 1 m and 2 m precision levels in three altitude subranges of the 64-200 m range. Analyzing the 1m-accuracy, the LoFTR trained on the Mars datasets outperformed the pre-trained LoFTR model and SIFT, which performed in the 20-30% accuracy range. SIFT proves more accurate than Pre-trained LoFTR in times of day between 8:00 LMST and 15:00 LMST. However, its accuracy rapidly declines for observations taken earlier than 8:00 LMST underperforming the base LoFTR model, especially at lower altitudes. Geo-LoFTR results in being the best localization method at 1 m and 2 m precision levels for most of the Martian day, except when observations are taken at 5:30 LMST, where its performance significantly degrades. Despite the overall higher accuracy, Geo-LoFTR is also observed to be less stable than the fine-tuned model throughout the Martian day. However, it is confirmed again to be more robust to altitude changes than its visual counterpart, providing yet another evidence of the contribution provided by geometric context in achieving scale invariance.

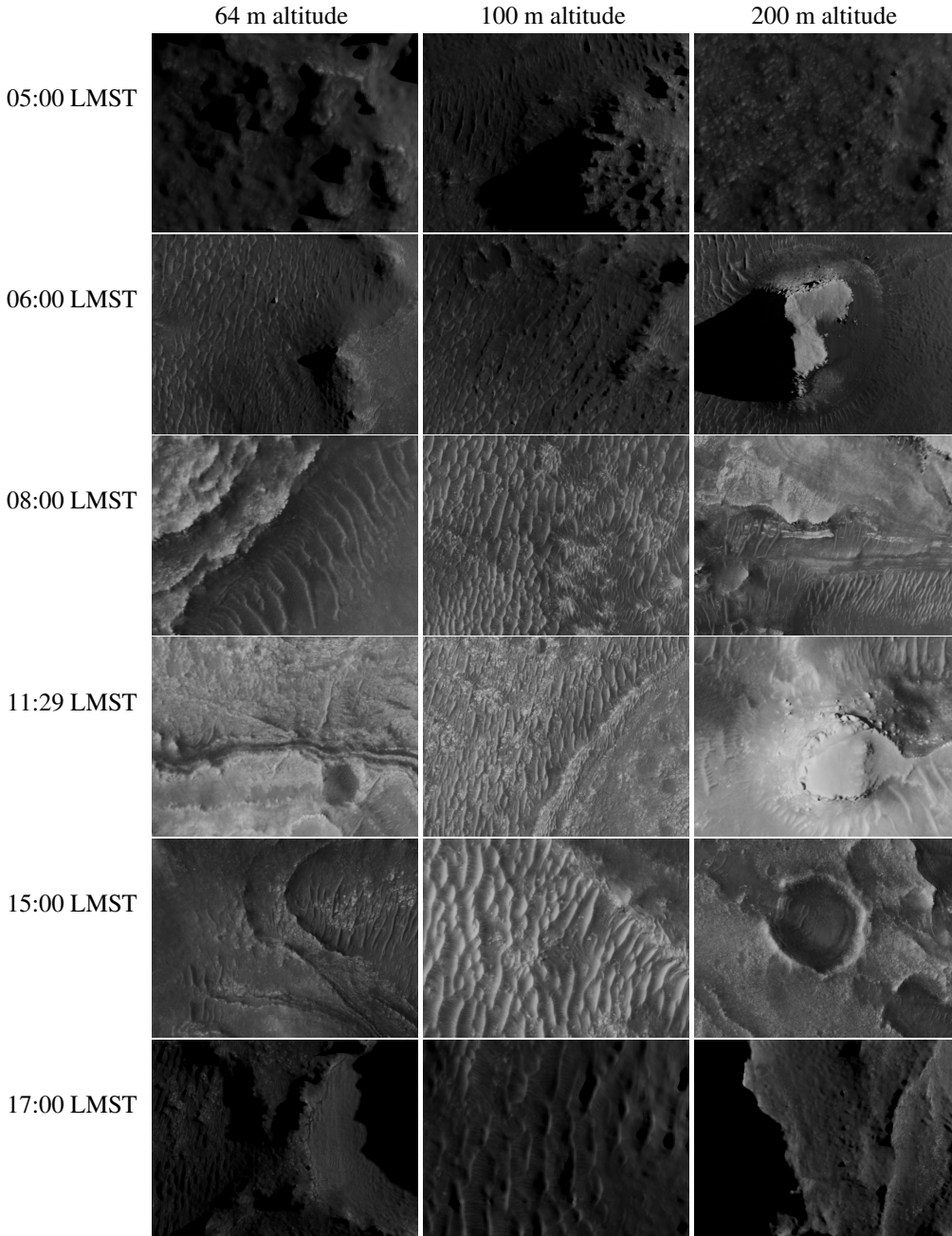


Figure 5.18: Sample nadir-pointing observations at different Local Mean Solar Times (LMSTs) taken on Mars on 2031-05-10, across three altitude ranges. The reference HiRISE map is taken at 15:00 LMST. The Sun Zenith is at 11:29 LMST. Data are generated from Jezero Crater using the MARTIAN framework.

5.4. Localization Over a Simulated Martian Day

LMST	Earth Time (UTC)	Sun EL (Mars24/MARTIAN)	Sun AZ (Mars24)	Sun AZ (MARTIAN)
05:07 (rise)	06:42	-0.2°	73.4°	16.6°
05:30	07:05	5.0°	75.2°	14.8°
06:00	07:36	12.1°	77.4°	12.6°
08:00	09:39	40.1°	85.0°	5.0°
11:29 (zenith)	13:14	86.9°	179.7°	270.3°
15:00 (HiRISE)	16:50	39.9°	274.9°	175.1°
17:00	18:54	11.6°	282.6°	167.4°
17:52 (set)	19:46	-0.2°	286.5°	163.5°

Table 5.3: Specifications of points of interests on the solar local trajectory from (77.44°E, 18.43°N) coordinates on Mars, on 2031-05-10. Sun azimuth is measured clockwise from North in Mars24 framework; while it is defined counterclockwise from East in the MARTAIAN framework. Adapted from Mars24[3]

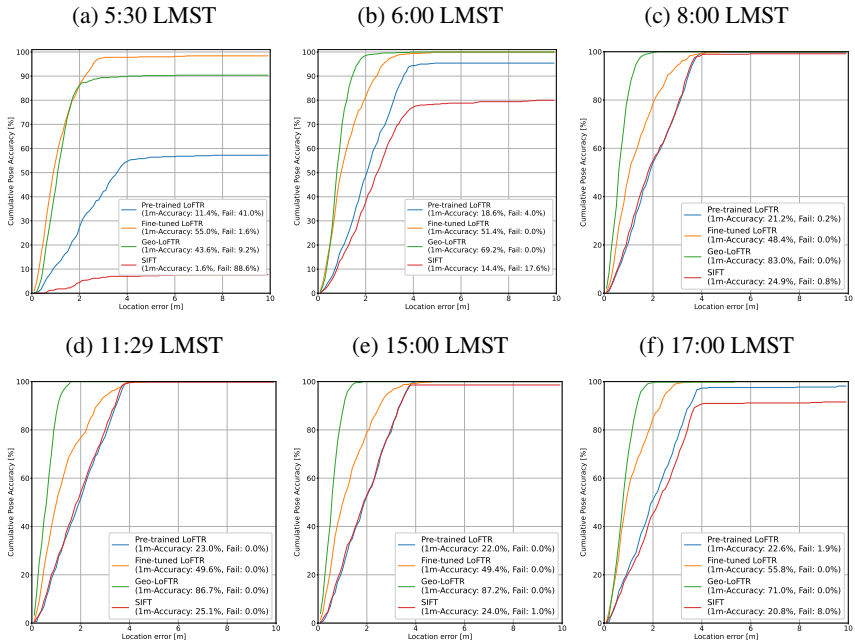
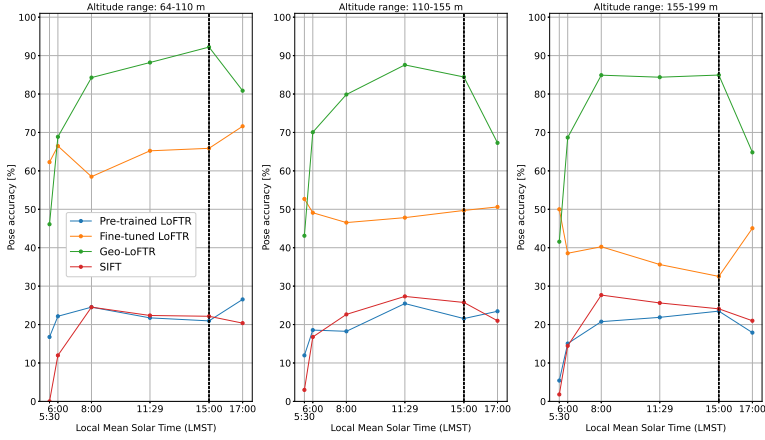


Figure 5.19: Cumulative distributions of the localization error for the MbL of 500 nadir-pointing observations (altitude 64-200m) taken at six different Local Mean Solar Times (LMSTs) on Mars, on 2031-05-10. The reference HiRISE map is taken at 15:00 LMST (c). The Sun Zenith is at 11:29 LMST (b). Data are generated from Jezero Crater using the MARTIAN framework.

(a) 1m-Accuracy



(b) 2m-Accuracy

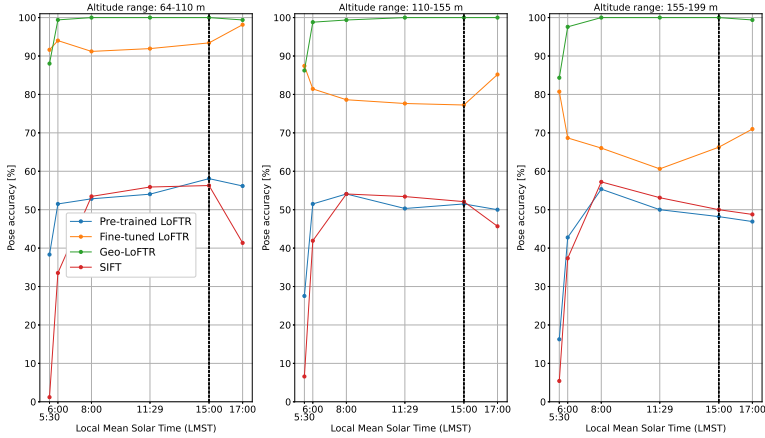


Figure 5.20: Localization accuracy at 1 m (*top*) and 2m (*bottom*) precision levels as a function of Local Mean Solar Time (LMST) of 500 nadir-pointing observations taken on Mars on 2031-05-10, across three altitude ranges. The reference HiRISE map is taken at 15:00 LMST (*dashed black line*). The Sun Zenith is at 11:29 LMST. Data are generated from Jezero Crater using the MARTIAN framework.

6 Conclusions and future work

In this work, we investigated deep-learning-based methods to achieve robust Map-based Localization (M_{BL}) for the long-range navigation of future Mars rotorcraft (e.g., the Mars Science Helicopter - MSH) under challenging illumination conditions. Our main contributions include adapting LoFTR, a feature-matching model integrating Convolutional Neural Networks with Transformer, into an M_{BL} framework that registers aerial images onto geo-referenced ortho-projected maps with a window selection strategy depending on pose prior uncertainty. We further proposed Geo-LoFTR, a novel multi-modal image matcher that incorporates map depth data using Digital Terrain Models (DTMs) obtained from orbital assets. By integrating cross-attention mechanisms, Geo-LoFTR learns a shared feature embedding that merges depth and visual information, aiming to leverage geometric context to enhance the robustness of feature representations against illumination changes, facilitating reliable matches.

To support model training and evaluation, we developed MARTIAN, a custom rendering framework in Blender, which allowed us to generate synthetic Mars datasets derived from HiRSE DTMs and ortho-images. LoFTR and Geo-LoFTR models were trained on query observation/map window pairs formed from combinations of 17 ortho-projected maps and 4,500 nadir-pointing aerial observations across diverse lighting conditions and scales, for a total of 150,000 training pairs. Over 4,000 test observations and 24 maps with controlled lighting, scale, and terrain morphology variations were further rendered to rigorously evaluate M_{BL} performance under varying conditions and against established methods, such as SIFT and the pre-trained LoFTR model.

6.1 Key findings

Fixed Sun Azimuth and Elevation Offsets

M_{BL} evaluations over observations with challenging offsets of 180° in Sun azimuth (AZ) and 35° in elevation (EL) from a reference map at (180° AZ, 5° EL) proved that Geo-LoFTR consistently outperformed other methods, particularly in 1-meter precision localization accuracy (1m-Accuracy). With a 29% improvement over the fine-tuned model, a 53% improvement over the pre-trained LoFTR and a 65% improvement over SIFT, Geo-LoFTR's integration of depth information leveraged the geometric consistency provided by DTMs, especially across altitude variations. This integration likely

contributed to scale invariance, as depth data constrains matches with pixel-to-pixel depth relationships depending on DTM's elevations only, thus mitigating the effect of altitude changes.

Terrain Morphology Robustness

Experiments across high-relief terrains and dune areas sampled from the Jezero Crater site have been conducted to assess MbL performance over different terrain morphologies. On dunal terrains, which contain repetitive patterns prone to ambiguous matches, all the methods show a decline in localization accuracy compared to rugged terrain. However, Geo-LoFTR exhibited greater robustness at 1m and 2m accuracy levels, outperforming both fine-tuned LoFTR and SIFT. This suggests that the inclusion of depth information in Geo-LoFTR helps to resolve ambiguities inherent to visual data in repetitive pattern scenarios by providing a geometric context that stabilizes matching.

Lighting Variation (Sun Azimuth and Elevation)

Geo-LoFTR proved higher localization accuracy compared to other methods across a broad range of Sun elevation and azimuth angles, with the fine-tuned model performing comparably only at the 5m-Accuracy level. Specifically, Geo-LoFTR maintained a 1m-accuracy in a 78-88% range over map Sun elevations between 10° and 90° , with observations taken at 40° . In contrast, the 1m-accuracy bounds are 45-50% for fine-tuned LoFTR and 19-24% for SIFT. MbL at challenging map elevations below 10° showed a significant decline in accuracy, but Geo-LoFTR still proved to be the most accurate overall.

Martian Day Localization Performance

In an evaluation across a simulated Martian day on the Jezero Crater site, Geo-LoFTR outperformed other models, especially at 1m and 2m precision levels, though with slightly less stability across different times of day compared to the fine-tuned LoFTR. Geo-LoFTR struggles more than fine-tuned LoFTR for observations at 5:30 Local Mean Solar Time (LMST) corresponding to 2° in Sun elevation (sunrise is at 5:07 LMST for selected date and location), especially at lower altitudes. Nonetheless, Geo-LoFTR's robustness to altitude variations reinforced the benefits of depth data, with significant accuracy improvements in observations from 6:00 LMST to 17:00 LMST, with sunset occurring at 17:52 LMST.

6.2 Future Directions

These findings set the stage for future research into multi-modal feature representation interpretability to deepen the understanding of the learned depth-visual relationships and further justify the indications that emerged from our MbL evaluation. Future efforts could include adopting contrastive learning techniques to refine the shared embeddings further, potentially improving localization performance.

One limitation of this work consists in the nadir-pointing nature of the aerial observations. While this setup serves as an initial investigation into robustness against illumination changes, future work needs to address robustness analysis to viewpoint changes. New versions of LoFTR and Geo-LoFTR can be trained on datasets including observations taken with specified ranges in yaw, pitch and roll angles, to be evaluated on dedicated test sets.

Future directions in validating our MbL pipeline should also include performance evaluations on lower-resolution orbital maps, such as CTX. Our method could register real observations collected during Mars 2020's Entry, Descent, and Landing (EDL) phase on CTX maps, benchmarking against state-of-the-art template matching methods. Lastly, LoFTR and Geo-LoFTR models fine-tuned on dedicated synthetic datasets simulating imagery from Ingenuity, can be used to evaluate our MbL pipeline's ability to register real Ingenuity images on HiRISE maps.

Bibliography

- [1] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, “Attention is all you need,” 2023. [Online]. Available: <https://arxiv.org/abs/1706.03762>
- [2] J. Sun, Z. Shen, Y. Wang, H. Bao, and X. Zhou, “Loftr: Detector-free local feature matching with transformers,” 2021. [Online]. Available: <https://arxiv.org/abs/2104.00680>
- [3] NASA Goddard Institute for Space Studies, “Mars24: Sunclock - a mars solar time and solar longitude calculator,” version 8.3.1, released on 2023-05-18. Accessed: 2024-10-28. [Online]. Available: <https://www.giss.nasa.gov/tools/mars24/>
- [4] University of Arizona, “HiRISE Camera Technical Specifications,” <https://www.uahirise.org/specs/>, 2024, [Accessed: October 7, 2024].
- [5] H. F. Grip, D. Conway, J. Lam, N. Williams, M. P. Golombek, R. Brockers, M. Mischna, and M. R. Cacan, “Flying a helicopter on mars: How ingenuity’s flights were planned, executed, and analyzed,” in *2022 IEEE Aerospace Conference (AERO)*, 2022, pp. 1–17.
- [6] J. Bapst, T. J. Parker, J. Balaram, T. Tzanetos, L. H. Matthies, C. D. Edwards, A. Freeman, S. Withrow-Maser, W. Johnson, E. Amador-French, J. L. Bishop, I. J. Daubar, C. M. Dundas, A. A. Fraeman, C. W. Hamilton, C. Hardgrove, B. Horgan, C. W. Leung, Y. Lin, A. Mittelholz, and B. P. Weiss, “Mars Science Helicopter: Compelling Science Enabled by an Aerial Platform,” *Bulletin of the AAS*, vol. 53, no. 4, mar 18 2021, <https://baas.aas.org/pub/2021n4i361>.
- [7] T. Tzanetos, J. Bapst, G. Kubiak, L. P. Tosi, S. Sirlin, R. Brockers, J. Delaune, H. F. Grip, L. Matthies, J. Balaram, S. Withrow-Maser, W. Johnson, L. Young, and B. Pipenberg, “Future of mars rotorcraft - mars science helicopter,” in *2022 IEEE Aerospace Conference (AERO)*, 2022, pp. 1–16.
- [8] B. Langlais, E. Thébault, A. Houliez, M. Purucker, and R. Lillis, “A new model of the crustal magnetic field of mars using mgs and maven,” *Journal of Geophysical Research: Planets*, vol. 124, no. 6, pp. 1542–1569, 2019. [Online]. Available: <https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1029/2018JE005854>

- [9] J. L. Bishop, C. Gross, J. Danielsen, M. Parente, S. L. Murchie, B. Horgan, J. J. Wray, C. Viviano, and F. P. Seelos, "Multiple mineral horizons in layered outcrops at mawrth vallis, mars, signify changing geochemical environments on early mars," *Icarus*, vol. 341, p. 113634, 2020. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0019103519304853>
- [10] B. L. Ehlmann and C. S. Edwards, "Mineralogy of the martian surface," *Annual Review of Earth and Planetary Sciences*, vol. 42, no. Volume 42, 2014, pp. 291–315, 2014. [Online]. Available: <https://www.annualreviews.org/content/journals/10.1146/annurev-earth-060313-055024>
- [11] A. Bramson, C. Andres, J. Bapst, P. Becerra, S. W. Courville, C. M. Dundas, S. M. Hibbard, J. J. W. Holt, S. Karunatillake, A. Khuller, M. T. Mellon, G. A. Morgan, R. W. Obbard, M. R. Perry, E. I. Petersen, N. E. Putzig, H. G. Sizemore, I. B. Smith, D. E. Stillman, and P. Wooster, "Mid-Latitude Ice on Mars: A Science Target for Planetary Climate Histories and an Exploration Target for In Situ Resources," in *Bulletin of the American Astronomical Society*, vol. 53, May 2021, p. 115.
- [12] A. G. Galofre, C. N. Andres, P. Becerra, A. Bhardwaj, A. Bramson, F. Butcher, P. R. Christensen, S. J. Conway, A. Coronato, E. Hauber, S. Hibbard, J. P. Knightly, T. Meng, G. R. Osinski, E. I. Petersen, J. Plaut, J. Rabassa, A. M. Rutledge, L. Sam, J. Serla, and K. X. Whipple, "A Comparative View of Glacial and Periglacial Landforms on Earth and Mars," *Bulletin of the AAS*, vol. 53, no. 4, mar 18 2021, <https://baas.aas.org/pub/2021n4i101>.
- [13] C. M. Dundas, A. M. Bramson, L. Ojha, J. J. Wray, M. T. Mellon, S. Byrne, A. S. McEwen, N. E. Putzig, D. Viola, S. Sutton, E. Clark, and J. W. Holt, "Exposed subsurface ice sheets in the martian mid-latitudes," *Science*, vol. 359, no. 6372, pp. 199–201, 2018. [Online]. Available: <https://www.science.org/doi/abs/10.1126/science.aao1619>
- [14] J. Delaune, J. Izraelevitz, S. Sirlin, D. Sternberg, L. Giersch, L. P. Tosi, E. Skliyan-skiy, L. Young, M. Mischna, S. Withrow-Maser, J. Mueller, J. Bowman, M. S. Wallace, H. F. Grip, L. Matthies, W. Johnson, M. Keennon, B. Pipenberg, H. Patel, C. Lim, A. Schutte, M. Veismann, H. Cummings, S. Conley, J. Bapst, T. Tzanetos, R. Brockers, A. Jain, D. Bayard, A. Chmielewski, O. Toupet, J. Burdick, M. Gharib, and J. Balaram, "Mid-air helicopter delivery at mars using a jetpack," in *2022 IEEE Aerospace Conference (AERO)*, 2022, pp. 1–20.
- [15] 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, p. 2421–2428, Apr. 2021. [Online]. Available: <http://dx.doi.org/10.1109/LRA.2021.3058918>
- [16] R. Brockers, J. Delaune, P. Proença, P. Schoppmann, M. Domnik, G. Kubiak, and T. Tzanetos, "Autonomous safe landing site detection for a future mars science helicopter," in *2021 IEEE Aerospace Conference (50100)*, 2021, pp. 1–8.

- [17] P. Schoppmann, P. F. Proença, J. Delaune, M. Pantic, T. Hinzmman, L. Matthies, R. Siegwart, and R. Brockers, “Multi-resolution elevation mapping and safe landing site detection with applications to planetary rotorcraft,” 2021. [Online]. Available: <https://arxiv.org/abs/2111.06271>
- [18] C. I. o. T. NASA Jet Propulsion Loaboratory, “Ingenuity’s flight log details,” NASA Jet Propulsion Laboratory, Tech. Rep., 2024, accessed from the NASA JPL website. [Online]. Available: <https://science.nasa.gov/mission/mars-2020-perseverance/ingenuity-mars-helicopter/>
- [19] J. Nash, Q. Dwight, L. Saldyt, H. Wang, S. Myint, A. Ansar, and V. Verma, “Censible: A robust and practical global localization framework for planetary surface missions,” in *2024 IEEE International Conference on Robotics and Automation (ICRA)*, 2024, pp. 8642–8648.
- [20] A. E. Johnson, Y. Cheng, N. Trawny, J. F. Montgomery, S. Schroeder, J. Chang, D. Clouse, S. Aaron, and S. Mohan, “Implementation of a map relative localization system for planetary landing,” *Journal of Guidance, Control, and Dynamics*, vol. 46, no. 4, pp. 618–637, 2023. [Online]. Available: <https://doi.org/10.2514/1.G006780>
- [21] A. Couturier and M. A. Akhloufi, “A review on absolute visual localization for uav,” *Robotics and Autonomous Systems*, vol. 135, p. 103666, 2021. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0921889020305066>
- [22] Y. Xu, L. Pan, C. Du, J. Li, N. Jing, and J. Wu, “Vision-based uavs aerial image localization: A survey,” *Proceedings of the 2nd ACM SIGSPATIAL International Workshop on AI for Geographic Knowledge Discovery*, 2018. [Online]. Available: <https://api.semanticscholar.org/CorpusID:53428751>
- [23] A. Couturier and M. A. Akhloufi, “A review on deep learning for uav absolute visual localization,” *Drones*, vol. 8, no. 11, 2024. [Online]. Available: <https://www.mdpi.com/2504-446X/8/11/622>
- [24] X. Wan, J. Liu, H. Yan, and G. L. Morgan, “Illumination-invariant image matching for autonomous uav localisation based on optical sensing,” *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 119, pp. 198–213, 2016. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0924271616301113>
- [25] C. G. Harris and M. J. Stephens, “A combined corner and edge detector,” in *Alvey Vision Conference*, 1988. [Online]. Available: <https://api.semanticscholar.org/CorpusID:1694378>
- [26] D. G. Lowe, “Distinctive image features from scale-invariant keypoints,” *International Journal of Computer Vision*, vol. 60, pp. 91–110, 2004. [Online]. Available: <https://api.semanticscholar.org/CorpusID:174065>

- [27] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, “Orb: An efficient alternative to sift or surf,” in *2011 International Conference on Computer Vision*, 2011, pp. 2564–2571.
- [28] E. Rosten and T. Drummond, “Machine learning for high-speed corner detection,” in *Computer Vision – ECCV 2006*, A. Leonardis, H. Bischof, and A. Pinz, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2006, pp. 430–443.
- [29] M. Calonder, V. Lepetit, C. Strecha, and P. Fua, “Brief: Binary robust independent elementary features,” in *Computer Vision – ECCV 2010*, K. Daniilidis, P. Maragos, and N. Paragios, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2010, pp. 778–792.
- [30] H. Bay, T. Tuytelaars, and L. Van Gool, “Surf: Speeded up robust features,” in *Computer Vision – ECCV 2006*, A. Leonardis, H. Bischof, and A. Pinz, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2006, pp. 404–417.
- [31] R. Brockers, P. Proença, J. Delaune, J. Todd, L. Matthies, T. Tzanetos, and J. B. Balaram, “On-board absolute localization based on orbital imagery for a future mars science helicopter,” in *2022 IEEE Aerospace Conference (AERO)*, 2022, pp. 1–11.
- [32] D. DeTone, T. Malisiewicz, and A. Rabinovich, “Superpoint: Self-supervised interest point detection and description,” 2018. [Online]. Available: <https://arxiv.org/abs/1712.07629>
- [33] S. Shah, D. Dey, C. Lovett, and A. Kapoor, “Airsim: High-fidelity visual and physical simulation for autonomous vehicles,” 2017. [Online]. Available: <https://arxiv.org/abs/1705.05065>
- [34] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby, “An image is worth 16x16 words: Transformers for image recognition at scale,” 2021. [Online]. Available: <https://arxiv.org/abs/2010.11929>
- [35] P.-E. Sarlin, D. DeTone, T. Malisiewicz, and A. Rabinovich, “Superglue: Learning feature matching with graph neural networks,” 2020. [Online]. Available: <https://arxiv.org/abs/1911.11763>
- [36] J. Edstedt, Q. Sun, G. Bökman, M. Wadenbäck, and M. Felsberg, “Roma: Robust dense feature matching,” 2023. [Online]. Available: <https://arxiv.org/abs/2305.15404>
- [37] M. Oquab, T. Darcet, T. Moutakanni, H. Vo, M. Szafraniec, V. Khalidov, P. Fernandez, D. Haziza, F. Massa, A. El-Nouby, M. Assran, N. Ballas, W. Galuba, R. Howes, P.-Y. Huang, S.-W. Li, I. Misra, M. Rabbat, V. Sharma, G. Synnaeve, H. Xu, H. Jegou, J. Mairal, P. Labatut, A. Joulin, and P. Bojanowski, “Dinov2: Learning robust visual features without supervision,” 2024. [Online]. Available: <https://arxiv.org/abs/2304.07193>

-
- [38] A. Zeng, S. Song, M. Nießner, M. Fisher, J. Xiao, and T. Funkhouser, “3dmatch: Learning local geometric descriptors from rgb-d reconstructions,” 2017. [Online]. Available: <https://arxiv.org/abs/1603.08182>
- [39] C. Choy, J. Park, and V. Koltun, “Fully convolutional geometric features,” in *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019, pp. 8957–8965.
- [40] S. Huang, Z. Gojcic, M. Usvyatsov, A. Wieser, and K. Schindler, “Predator: Registration of 3d point clouds with low overlap,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2021, pp. 4267–4276.
- [41] H. Yu, Y. Feng, W. Ye, M. Jiang, H. Bao, and G. Zhang, “Improving feature-based visual localization by geometry-aided matching,” 2022. [Online]. Available: <https://arxiv.org/abs/2211.08712>
- [42] Q. Zhou, S. Agostinho, A. Osep, and L. Leal-Taixé, “Is geometry enough for matching in visual localization?” 2022. [Online]. Available: <https://arxiv.org/abs/2203.12979>
- [43] J. Xiao, D. Tortei, E. Roura, and G. Loianno, “Long-range uav thermal geo-localization with satellite imagery,” in *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2023, pp. 5820–5827.
- [44] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” 2015. [Online]. Available: <https://arxiv.org/abs/1512.03385>
- [45] Z. Li and N. Snavely, “Megadepth: Learning single-view depth prediction from internet photos,” 2018. [Online]. Available: <https://arxiv.org/abs/1804.00607>
- [46] R. L. Kirk, E. Howington-Kraus, M. R. Rosiek, J. A. Anderson, B. A. Archinal, K. J. Becker, D. A. Cook, D. M. Galuszka, P. E. Geissler, T. M. Hare, I. M. Holmberg, L. P. Keszthelyi, B. L. Redding, W. A. Delamere, D. Gallagher, J. D. Chapel, E. M. Eliason, R. King, and A. S. McEwen, “Ultrahigh resolution topographic mapping of mars with mro hirise stereo images: Meter-scale slopes of candidate phoenix landing sites,” *Journal of Geophysical Research: Planets*, vol. 113, no. E3, 2008. [Online]. Available: <https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1029/2007JE003000>
- [47] P. contributors, “Blender hirise dtm importer,” 2024, version 1.0. [Online]. Available: <https://github.com/phaseIV/Blender-Hirise-DTM-Importer>
-