Visual Odometry for Extreme Robotics

Sylvain Beuret, Marie Ethvignot and Joachim Despature

***Abstract*— This research investigates the performance of Visual Odometry (VO) algorithms under challenging conditions, such as varying lighting, dynamic occlusions, and high-speed motion. Through a comprehensive review and experimental analysis of feature-based methods (ORB) and direct methods (DSO), we identified significant limitations in low-texture and dynamic environments. While feature-based methods perform well in normal conditions, they struggle under complex scenarios. Direct methods offer advantages but are computationally intensive and less effective in high-speed situations.**

To address these challenges, we integrated a patch affine illumination model with DSO, which showed partial improvement in robustness against dynamic lighting changes. Our results suggest the potential of Visual-Inertial Odometry (VIO) to mitigate some issues but highlight the need for high computational resources. The study underscores the importance of dynamic algorithm selection and advanced sensor fusion techniques to enhance the accuracy and reliability of VO systems.

This research provides a foundation for future advancements aimed at improving autonomous navigation for planetary exploration, emphasizing the need for refined algorithms and optimized computational efficiency to ensure robust performance in diverse and dynamic environments.

I. INTRODUCTION

A key aspect of mobile robotics is to ensure an autonomous and robust navigation of the system. Whether a robot is used for industrial inspection, security or space exploration, it is crucial in all type of applications for it to have an accurate knowledge and tracking of its position and orientation in its environment. A lot of methods and sensors are used to tackle this challenging task, such as Global Navigation Satellite Systems (GNSS), wheel odometry, beacon-based localization systems, Laser imaging Detection and Ranging (LiDAR) or Visual Odometry (VO). Each of these methods have advantages and disadvantages and their use depends on the type of application and environment of the robot [1]. For example, although wheel odometry is the simplest and one of the most utilized method, it lacks precision since an error in the positioning accumulates proportionally over time due to wheel slippage. Another common practice is to use a GNSS (such as the Global Positioning System) for position and orientation tracking. Even though this method is very accurate, it cannot be used for indoors or space applications. Visual Odometry is hence a widely used method for global localization and orientation tracking, especially in space applications like NASA's Mars Exploration Rovers (MER) [2] or NASA's Mars flying robot Ingenuity [3].

This research contributes to a larger initiative aimed at developing a planetary two-wheeled rover, controlled in part through reinforcement learning. The rover is designed to be equipped with three types of cameras: visible spectrum, infrared, and Single Photon Avalanche Diodes (SPAD).In the context of space exploration, the rover must be resilient to various challenging conditions, including dust storms, planetary landings, and difficult lighting situations, such as those found in craters or when the sun is near the horizon. Importantly, the rover must operate autonomously without relying on external sensors or assistance from a Global Navigation Satellite System. This study focuses on the visual odometry aspect of the rover's localization system, a critical component for ensuring the rover's self-sufficiency and accurate positioning in the absence of GNSS support.

Odometry is the process of estimating an agent's (robot or vehicle) change in position and orientation over time using the agent's motion sensors. For example, wheel odometry is the estimation of the agent's pose and orientation over time using its wheels, by counting the number of revolutions of rotary encoders. In a similar way, Visual Odometry is a technique used to estimate the pose and orientation of an agent using a sequence of images taken by one or multiple on-board cameras (monocular or stereo visual odometry).

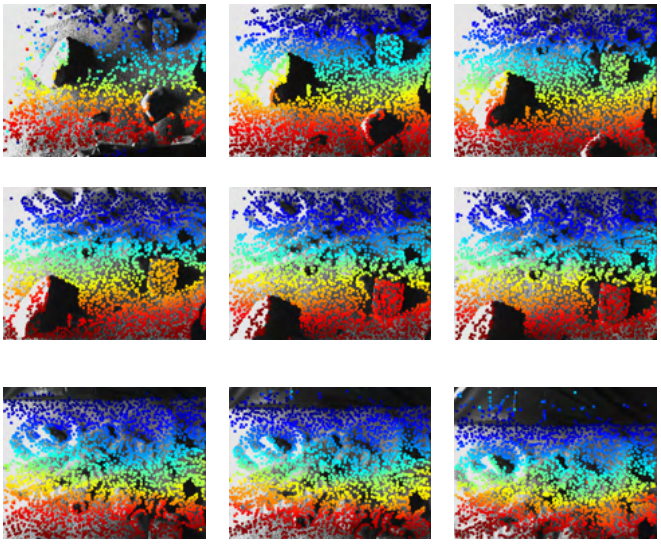


Fig. 1: Direct Sparse Odometry [4] : Sampled color coded depth map of acquired dataset of challenging terrain mimic

Monocular VO cannot directly measure the scale of movement because it only captures 2D projections of the 3D world. The actual distance to objects cannot be

determined without additional assumptions or external information, which yields a scale ambiguity. It typically requires less computational resources than stereo matching but can be more challenging in terms of accuracy and robustness due to the scale ambiguity and reliance on feature tracking in potentially texture-poor environments. Stereo VO on the other hand, can directly measure the depth of objects by triangulating points seen by both cameras. However, it is crucial to use a pair of cameras with a known baseline (distance between the cameras), which provides a consistent scale for all measurements. This scale allows for more accurate and robust motion estimation than monocular VO. It is also possible to use an RGB-D camera, which provides both color (RGB) images and depth (D) information. Since it combines visual and depth information it is more robust for motion estimation.

The main steps of Visual Odometry are the following: image acquisition from one or multiple cameras, feature detection and matching (feature based methods) or tracking of pixels intensity (direct methods), motion estimation and pose integration (integrate the estimated motions over time to obtain the trajectory of the camera). There are two different methods of VO depending on the second step of the algorithm: feature-based method and direct method. These two methods will be detailed later as we use both of them in this project.

Another variant of VO, is Visual Inertial Odometry (VIO), a method that combines the information from the cameras with inertial measurements from an inertial measurement unit (IMU) to estimate the motion (position and orientation) of a camera-equipped agent.

Visual Odometry is also frequently integrated with Simultaneous Localization and Mapping (SLAM) systems to enhance the robustness of the navigation solution. SLAM systems construct a map of the environment by identifying and utilizing features, which in turn helps mitigate drift. This is achieved through loop closure, a process where the system recognizes previously visited locations, thereby correcting the trajectory and reducing cumulative errors.

While VO is a powerful and widely-used technique, there are still several limitations that can impact its performance and accuracy. For examples, some important challenges for VO are the computational cost (high-resolution images and real-time processing) and the light and imaging conditions. Indeed, in order to have an accurate detection and tracking of features across images, there needs to be sufficient illumination and the scene needs to be static with enough texture in the environment. Whether it is for terrestrial or space applications, the limitations of Visual Odometry due to the complexity of the visual fields can drastically impact the estimation of the state of the robot.

In this paper, we try to uncover how to accurately estimate the position of a robot operating across planetary environments under complex visual fields (low light, high dynamic range, high optical depth or fast motion). To achieve this, we focus on three imaging conditions that impact the accuracy of the results of VO: complex lighting environments, dynamically occluded scenes and fast moving scenes. To establish a foundation, we conducted a comprehensive review of existing VO algorithms, analyzing their working principles and limitations. Afterwards, we tested the chosen existing VO algorithm with datasets in normal imaging conditions. Finally, we evaluated the performance of these algorithms with datasets for cases of difficult imaging conditions. The ultimate goal is to understand why these VO algorithms do not perform as well under those conditions to increase the robustness and accuracy under condition of planetary exploration.

The paper is organised as follows: in Section two we review the related and existing work on Visual Odometry in general and about tackling our cases of difficult imaging conditions (complex lighting environments, dynamically occluded scenes and fast motion scenes). In section three we present the methodology we followed to implement and test VO algorithms and relevant datasets. Section four summarizes our results and section five is a discussion about these results. Finally we will establish a conclusion of our research in section six.

II. RELATED WORK

A. *Visual Odometry Algorithms*

This paper explores the development of robust Visual Odometry (VO) algorithms for extreme robotics applications. We achieve this by integrating classic VO algorithms with novel data pre-processing techniques. To establish a foundation, we conducted a comprehensive review of existing VO algorithms, analyzing their working principles, limitations, and various types.

The first occurence of the term ”Visual Odometry” was introduced in a paper by David Nister & al. in 2004 [5]. The key idea of this paper revolves around solving the problem of estimating the relative pose (position and orientation) of two cameras from a minimal set of corresponding 2D-3D point correspondences. Since then, two main branches of VO algorithms have been developed:

*1) **Feature based Method**:* Feature-based visual odometry algorithms detect distinctive features in consecutive images for motion estimation. These algorithms first detect features in the initial image frame, such as corners, edges, or blobs. Then, they match corresponding features between frames using descriptors like SIFT [6], SURF [7], or ORB [8].

After feature matching, these algorithms compute the motion of the camera based on the displacement of matched feature points. Techniques like RANSAC [9] or least squares fitting are often used for robust motion estimation. Once the motion is estimated, it updates the camera’s pose (position

and orientation), representing its new position relative to the initial frame.

Continuously estimating camera motion between frames allows the algorithm to compute the camera's trajectory over time. Feature-based VO algorithms offer robustness to changes in lighting and scene texture and are computationally efficient. However, they may struggle in environments with repetitive patterns or low-texture regions. Overall, they provide an effective approach to visual odometry, widely used in robotics, autonomous navigation, and augmented reality applications.

SURF is designed to be computationally efficient while maintaining robustness to variations in scale, rotation, and illumination. It utilizes integral images and a wavelet-based approach to achieve speedups compared to traditional methods like SIFT. SURF has been widely used in visual odometry applications due to its balance between speed and accuracy, making it suitable for real-time processing in robotics and augmented reality systems.

ORB combines the speed of FAST feature detection with the robustness of BRIEF [10] descriptors, making it suitable for real-time applications such as visual odometry. It has become popular due to its computational efficiency and competitive performance compared to other feature-based methods like SIFT and SURF.

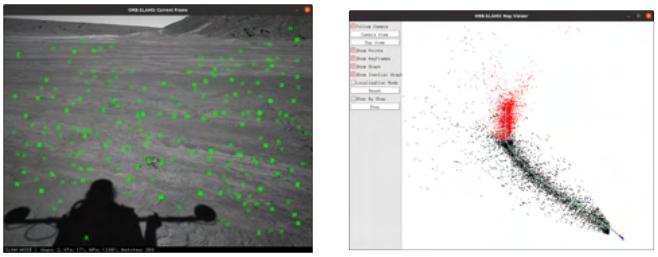


Fig. 2: Example of ORB-SLAM3 run. Left: Live feed with feature detection. Right: Localization and mapping

2) ***Direct Method****:* Direct method visual odometry algorithms operate directly on the pixel intensity values of consecutive images to estimate the motion of the camera or vehicle. These algorithms compute the spatial gradients of pixel intensity values in local neighborhoods across consecutive image frames. These gradients capture the variations in intensity, which are indicative of object edges and texture.

The core objective is to align the intensities of corresponding pixels between the reference and current frames. By finding the optimal transformation (rotation and translation) that minimizes the discrepancy in intensity values, the algorithm aligns similar structures in the two frames. Direct methods minimize a cost function that quantifies the differences in intensity between corresponding pixels.

Optimization techniques like gradient descent or Gauss-Newton optimization are commonly employed to iteratively refine the estimated transformation until convergence [11]. Upon convergence, the estimated transformation provides the motion (rotation and translation) of the camera between the reference and current frames. This motion estimation forms the basis for updating the camera's pose and computing its trajectory over time.

LSD-SLAM [12] revolutionized visual SLAM systems by operating directly on image intensities without the need for feature extraction. It introduced efficient optimization techniques for real-time camera pose estimation and dense 3D mapping, enabling robust localization and mapping in large-scale environments.

DSO introduced a novel approach to visual odometry by combining the efficiency of sparse feature-based methods with the accuracy of direct optimization of pixel intensities. It achieved real-time performance while providing accurate motion estimation, making it widely adopted in both research and industry applications.

Both LSD-SLAM and DSO are heavily based on direct VO algorithms while having a few specifications of feature-based algorithms, making them hybrid algorithms.

B. Complex Lighting Environments

In complex outdoor environments with diverse lighting conditions, Direct Sparse Odometry (DSO) proves advantageous for visual odometry.

DSO's direct optimization of pixel intensities ensures robustness to lighting variations, a common challenge in outdoor settings. By directly comparing pixel values between consecutive frames, DSO can effectively handle changes in illumination.

Operating on a sparse set of feature points, DSO focuses on informative features while disregarding irrelevant or noisy regions. This approach helps mitigate issues with repetitive patterns or low-texture areas encountered outdoors.

DSO's real-time performance capabilities make it suitable for dynamic outdoor environments where scenes may change rapidly. Its efficient implementation ensures timely processing of images, crucial for accurate motion estimation.

In the face of challenging conditions like dynamic lighting changes, shadows, and reflections common outdoors, DSO's direct optimization framework proves robust. It can handle these challenges by directly comparing pixel intensities, leading to more reliable motion estimation.

In essence, DSO's combination of direct intensity optimization, sparse feature representation, real-time performance, and robustness to challenging conditions make it well-suited for visual odometry in complex outdoor environments.

Various techniques have been developed to address the challenges posed by illumination changes. Despite DSO showing some ability to handle gradual shifts in lighting, its core reliance on photo-consistency, shared by all direct method VO algorithms, means that significant dynamic changes can still substantially impact accuracy.

An explored path is to constantly assess the illumination level, in order to adapt the camera's exposure time and the environment' map as done in "Robust visual localization in changing lighting conditions" [13]. While this paper

uncovers interesting aspects on the challenges of illumination changes, much of its content may not directly apply to our scenario. This is because the robotic platform tested in the study, the Astrobee [14], operates within the highly controlled and predictable environment of the International Space Station (ISS). Moreover, the ISS environment offers a significant number of trackable features, further enhancing the reliability of VO algorithms in such conditions.

Our chosen approach draws heavily from the research conducted by Kim et al., as presented in their paper "Robust Visual Odometry to Irregular Illumination Changes with RGB-D camera" [15]. This paper introduces a patch affine illumination change model, which enhances the capability of direct visual odometry algorithms to adapt to dynamic changes in illumination conditions. Our intention is to incorporate these advancements into the framework of DSO.

C. *Dynamically Occluded Scenes*

Vehicles and robots that use Visual Odometry for localization and navigation suffer when deployed in environments characterized by high optical depths. These environments include rain and snowfall, smoke, fog, and dust, among others, making these situations dynamic and unpredictable. When parts of the scene are occluded by objects moving into the camera's field of view or by occluded environments, the VO system loses track of the features in those regions which impacts the feature tracking across frames and hence the general trajectory estimation. Moreover, the boundaries between occluded and visible regions can create ambiguities in depth estimation.

Up to the present time, there hasn't been a lot of research to resolve specifically the issue of dynamically occluded environments (such as fog or dust) in Visual Odometry. However, we can still find a few papers and research about occlusion-robust VO approaches in general. Among the most cited works in this field is the paper on Probabilistic Inertial-Visual Odometry (PIVO) [16], which integrates IMU sensors with monocular cameras to maintain reliable pose estimation despite occlusions and feature-poor environments. This paper introduces a probabilistic approach and demonstrates that stronger coupling between the inertial and visual data sources leads to robustness against occlusion and feature-poor environments.

Another notable contribution is LEAP-VO [17], which enhances long-term feature tracking by dynamically estimating and adjusting for occlusions using both visual and temporal information. Some learning-based methods also recently showed some significant improvement of VO performance in dynamically occluded environments. For example, [18] proposes a learning-based VO method that employs optical flow maps and deep learning to dynamically adjust attention weights based on different motion scenarios.

Even though the challenge of dynamically occluded environments hasn't yet been tackled a lot in the context of VO specifically, there has been numerous research proposing methods to remove dynamical occlusions (such as haze or fog) from images, which could be used as a pre-processing step on images in the VO algorithm, and hence have better performance in detecting and tracking features. The main methods used for dehazing images can be classified into three types of methods: prior-based methods, fusion-based methods and learning-based methods.

Prior-based methods are grounded in assumptions about the properties of clear images and the atmospheric scattering model. The atmospheric scattering model provides a theoretical foundation for understanding how light interacts with atmospheric particles, forming the basis for estimating the true scene radiance in hazy conditions. One of the most influential works in this category is the Dark Channel Prior by He, Sun, and Tang (2011) [19]. This method assumes that in a haze-free image, at least one color channel has some pixels with very low intensity, which helps in estimating the transmission map and atmospheric light to effectively remove haze. Other prominent prior-based methods include the Color Attenuation Prior [20], which leverages the disparity between the brightness and saturation of pixels to estimate depth information and subsequently remove haze.

Fusion-based methods enhance image visibility by integrating multiple input images or different representations of a single image, The Multi-Scale Fusion method [21]. It combines several derived images using different enhancement techniques and merges them through a weighted fusion process to improve visibility under hazy conditions. These methods are particularly effective in dynamic environments where conditions change rapidly, as they integrate various image features such as contrast, saturation, and exposure.

Deep learning methods use neural networks to learn the mapping from hazy images to clear ones. Convolutional Neural Networks (CNNs) are trained on large datasets of hazy and clear images to develop robust dehazing models. DehazeNet [22] and AOD-Net [23] are among the most cited works in this field. These networks are designed to directly process hazy images and produce clear outputs by learning intricate features that traditional methods may overlook. The end-to-end learning approach of these models has significantly advanced the performance of dehazing techniques, providing superior clarity and detail preservation.

D. *Fast Moving Scenes*

Image acquisition under fast motion primarily results in motion blur, which occurs due to the movement of either the camera or the object in front of it. This artifact arises when the exposure time is longer than the duration of the movement, causing the captured image to appear blurred rather than instantaneous. Exposure time can vary depending on the environmental conditions; for instance, low light often necessitates longer exposure times, making motion blur possible even with high-quality sensors in challenging situations. Motion blur is a well-known issue in visual odometry and in image processing more generally, leading to

extensive research for solutions. In visual odometry, the usual approach is by deblurring images in preprocessing [24][25] or using robust algorithms if the situation is only occasional [26]. There are three main conventional methods to achieve deblurring: the Wiener filter, the Richardson-Lucy algorithm, and blind deconvolution.

Wiener filtering has been employed for many years in order to reduce noise and blurring on images [27][28]. This technique is a linear low-pass filter aiming to minimize the mean square error between the estimated image and the true image. The cutoff frequency is space dependent, having lower treshold in low-detail regions. It requires an accurate Point Spread Function (PSF) and an estimate of the noise in order to filter. The Point Spread Function describes how each object is deformed in the image. While the Wiener filter is known for its simplicity and speed, it is limited to linear noise and is highly sensitive to the accuracy of the estimated PSF and noise levels to produce high-quality results.

The Richardson-Lucy algorithm is another long-standing yet still relevant technique for image restoration [29][30]. Unlike Wiener filtering, which assumes linear blurring, the Richardson-Lucy algorithm addresses non-linear blurring by employing an iterative deconvolution method based on Bayesian probability. Because the blurring of an image can be explained with a convolution between the input and the PSF. While this technique works for more situations, it still needs a estimation of the Point Spread Function in order to have good results. The iterative aspect leads to better results but can also be a drawback as it is computationally heavier.

The third method works even with an unknown blur kernel thus not needing estimation of PSF nor noise level. Blind deconvolution algorithms work by estimating and resolving blurring at the same time in an iterative manner [31]. This method is the more polyvalent but also the computationally heavier.

These traditional techniques are well-established and used techniques, but in the last decade, much of the research has shifted towards using deep learning to address image deblurring. Early methods incorporated learning-based approaches into certain parts of existing algorithms [32][33]. In recent years, convolutional neural networks are used in order to deblur images by progressively increasing the resolution thereby recovering sharper images [34][35]. While these newer methods are promising and efficient, our study will focus on conventional methods to gain a deeper understanding of the underlying principles, which may be less apparent with deep learning approaches.

Another approach to handle fast motion in visual odometry is to incorporate Inertial Measurement Units (IMU) data as an additional independent module, assisting classical visual odometry with an Extended Kalman Filter (EKF) to achieve Visual-Inertial Odometry. Both modules can run in parallel, providing two independent state estimations that are then fused [36], or they can be combined from the beginning, as in VINS-Mono [37]. This method improves state estimation by distributing the error across independent sources and having two complementary modules as IMUs operate at a much higher rate ($\sim$ 1000Hz) than cameras ($\sim$ 50Hz), making them more accurate for high-speed motion [38][39]. Additionally, IMUs are inexpensive and commonly implemented in most mobile robotics devices.

III. METHODOLOGY

To evaluate the robustness of the chosen visual odometry systems, ORB-SLAM3 and DSO, in extreme situations, we created datasets representing each scenario. Prior to evaluating our dataset, we tested the VO systems on well-documented datasets to establish a baseline.

A. *Visual Odometry Systems*

The first step is to make the different systems work with the dataset used for their evaluation in the documentation in order to prove their performance under the same conditions.

*1) **ORB**:* ORB-SLAM3 [40] is the latest version of ORB published in 2021, a feature-based system widely used, which is the reason why it was chosen, to ensure reliable results and have access to documentation. The installation was performed on a freshly installed Ubuntu 20.04 system, with all necessary dependencies set to the versions specified in the documentation. This approach was taken because the system may not be compatible with newer versions of these dependencies. ORB-SLAM3 has examples running with KITTI [41] and EuRoC MAV [42] (Micro Aerial Vehicle) dataset which have both been evaluated in our study.

EuRoC MAV is a visual-inertial dataset with data collected on-board a micro aerial vehicle. It contains stereo images, IMU measurements and accurate ground truth. It was recorded indoor in an industrial environment with eleven sequences with different difficulties. Only the first one, MH01, has been evaluated before changing for a dataset representing more the conditions of the study.

KITTI is a comprehensive benchmark dataset widely used in the fields of computer vision and autonomous driving research. Developed by the Karlsruhe Institute of Technology and the Toyota Technological Institute at Chicago, it includes data collected from a variety of sensors mounted on a car driving in the streets of Karlsruhe, such as stereo cameras, LiDAR, and GPS/IMU systems. The KITTI dataset contains 21 outdoor sequences and each sequence is available in both monocular and stereo vision.

*2) **DSO**:* Since its initial publication in 2016, DSO has undergone minimal updates. We installed it on an Ubuntu 20.04 Virtual Machine. However, due to the elapsed time and the rapid evolution of dependencies, numerous modifications were necessary. These adjustments included updates to DSO's source code, specifically addressing deprecated OpenCV functions, as well as modifications to its dependencies, particularly Pangolin.

DSO was developed concurrently with the TUM-Datasets, which adhere to the calibration setup outlined in "A photometrically calibrated benchmark for monocular

visual odometry" [43]. These datasets were tailored to be compatible with DSO, distinguishing them from other visual odometry datasets with their photometric calibration across all sequences. Each dataset comprises a comprehensive array of components, including the image files themselves, precise photometric values corresponding to the camera settings, the groundtruth, a file linking images to their exposure times, and the exact timestamps of image capture. Furthermore, intrinsic camera parameters are provided, along with a supplementary PNG file known as the vignette. This vignette serves a critical purpose in compensating for radial light fall-off, a phenomenon inherent to camera lenses that results in darker regions towards the periphery of images.

DSO underwent testing by its developers using the EuRoC MAV Dataset [42], albeit with some limitations. Unlike the TUM-Datasets, which boast photometric calibration, the EuRoC MAV Dataset lacks this feature. Despite containing valuable Visual-Inertial Sensor Unit data and calibration files, a crucial component missing from this dataset is the vignette image required for DSO operation. Consequently, attempts to utilize DSO with datasets other than the TUM-Datasets proved unsuccessful.

The results obtained using the TUM-Datasets closely resemble those reported by the original authors, as illustrated by Fig. 3

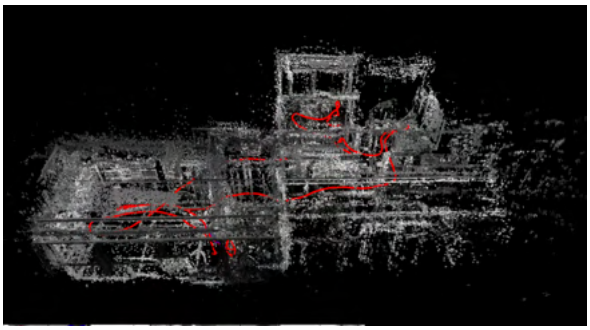


Fig. 3: Mapped trajectory using DSO on TUM-Dataset Sequence 1

B. *Datasets*

In order to assess the algorithms' robustness under extreme conditions, new datasets were curated. Existing datasets were found lacking in adequately representing these particular scenarios, necessitating the creation of new datasets for evaluation purposes.

1) ***Complex Lighting Environments****:* We initially explored "The POLAR Traverse Dataset" [44] as it offered a depiction of desired extra-planetary terrains under contrasting lighting conditions. However, we soon encountered limitations: the proximity of the light source to the terrain led to a pronounced light gradient. This effect stems from the fact that light intensity diminishes inversely proportional to the square of the distance, as described by the following formula:

$$I(r) = \frac{I_0}{r^2}$$

where:

$I(r)$: Intensity of light at distance r from the source
I_0 : Initial intensity of the light source
r : Distance from the light source

While not inherently restrictive, the dataset's images being captured one meter apart, coupled with the limited number of pictures in each sequence, complicated its usability. Another obstacle arose from the absence of a vignette picture essential for accurate photocalibration of the dataset.

We thus decided to create our own dataset, addressing the challenges highlighted by the POLAR Traverse Dataset, using a FLIR Blackfly BFS-U3-04S2 camera. The dataset was captured in a blackout room in order to avoid the interference of external light sources. David Rodríguez had prepared a sandbox filled with sand to emulate extraterrestrial terrains, complete with added obstacles like rocks. To mitigate the light intensity fluctuations observed in the POLAR Traverse Dataset, we opted for a neon light matching the dimensions of the sandbox. This ensured a consistent light intensity aligned with the camera's movement axis. We finally placed the sandbox on a cart, fixed the camera so the cart could pass underneath it.

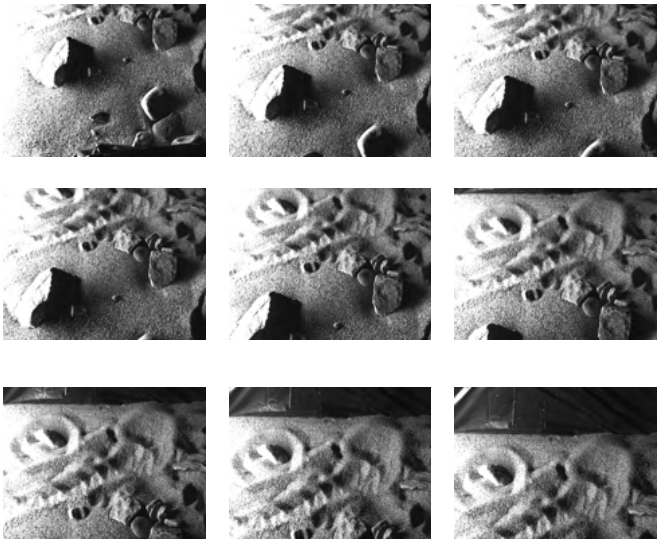


Fig. 4: Acquired dataset (Low-Sun) sample corresponding to Fig.1

We conducted two distinct sequences: one with the neon positioned slightly above the sandbox (High-Sun) and another with the neon in direct contact with the sand (Low-Sun). This approach enabled us to assess DSO performance under varying shadow conditions. However, due to equipment limitations, we were unable to capture ground truth data. Consequently, we recorded the dataset along a straight

path of approximately 30 cm, encompassing around 300 images.

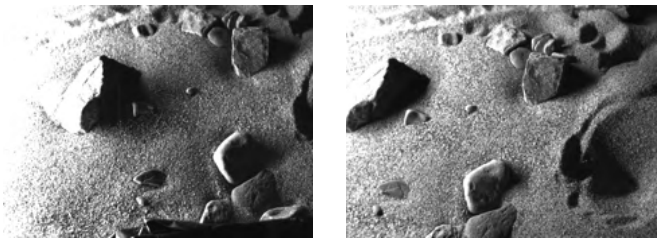


Fig. 5: Comparison of Low-Sun Dataset (Left) and High-Sun Dataset (Right)

To enable the utilization of DSO on our new dataset, we needed to calibrate it employing the same methods as the TUM Datasets. This calibration began by compiling and documenting the camera's intrinsic properties within a camera.txt file. The initial phase involved employing the chessboard calibration method [45]. Following this, an adequate number of chessboard images were captured using the same camera utilized during the dataset's acquisition, enabling the acquisition of the camera's intrinsic parameters.

Compensation for radial light fall-off towards the image edges, induced by the camera lens, necessitated the acquisition of a vignette. This was accomplished by capturing an image of a white wall under consistent lighting conditions using the same camera employed for dataset acquisition.

Subsequently, to ensure compatibility with DSO using the minimum required calibration, a text file was generated using a python script. This file documented each image's name, which required renaming from 0000 to 0300 using another Python script beforehand. Additionally, it included the POSIX timestamp of each image and its corresponding exposure time.

To fully harness the capabilities of DSO, we needed to conduct photocalibration of our camera. With limited resources left from previous users to fully understand the needed steps for this part of the calibration, our initial approach involved capturing a series of images of a white wall with incrementally increasing exposure times. Unfortunately, the software provided alongside the camera lacked this functionality. Consequently, we developed a Python script to acquire this image sequence, which also facilitated automatic logging of the POSIX timestamp and exposure time. This data was crucial for generating the camera's photoconsistency file, logging the values of G which characterizes the relationship between the pixel values in the image and the actual light intensities received by the camera sensor. alongside the image sequence's dense attenuation factor, shown in Fig.5, to insure the white plane is under consistent illumination.

Subsequently, leveraging the previously acquired vignette for the camera, our next objective was to perform vignette calibration. Regrettably, this proved unsuccessful, and due to time constraints, further investigation was not feasible.

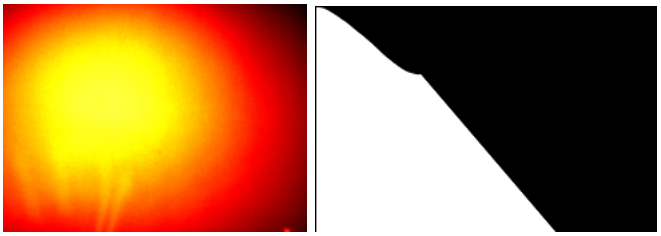
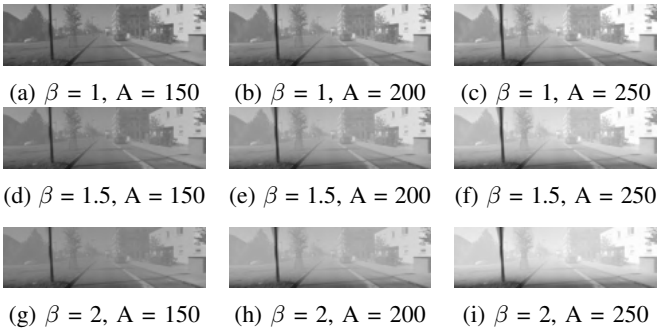


Fig. 6: Left: Lens attenuation of the Blackfly BFS-U3-04S2 Right: Plot of the photometric response function, G of each image of the calibration dataset

*2) **Dynamically Occluded Scenes**:* Given our research about the already existing work on VO in dynamically occluded scenes, we decided that to try to have better performance in these cases, a pre-processing step in the VO algorithm to dehaze the images before analyzing them would be the best option. As summed up in the Related work section, a lot of dehazing methods exist nowadays and are available in open-source. However, in order to test this approach, it was necessary for us to have a relevant VO dataset containing fog, haze or dust storms. Unfortunately, up to this date, no such datasets exist or are available, it was hence imperative to create our own dataset.

Initially, we thought of creating our own outdoor or indoor foggy VO dataset (using a fog generator), but due to a lack of time and means, we did not pursue in this direction. Fortunately, a lot of techniques and methods to synthetically add fog on images exist. The one we decided to use [46] is rather simple and efficient. We tested this method on the dataset they use and changed the scattering coefficient β and the airlight value (parameters of the atmospheric scattering model) to get different types of fog. The higher the β, the denser the fog.

We used this technique on some sequences of the KITTI dataset to constitute our first dataset to test on ORBSLAM3. A picture of the effect of the β coefficient and the airlight are presented on Figure 7.



(a) β = 1, A = 150 (b) β = 1, A = 200 (c) β = 1, A = 250

(d) β = 1.5, A = 150 (e) β = 1.5, A = 200 (f) β = 1.5, A = 250

(g) β = 2, A = 150 (h) β = 2, A = 200 (i) β = 2, A = 250

Fig. 7: Image of the KITTI 01 sequence with different beta and airlight coefficient

Since the end-goal of our research is to develop a robust VO algorithm for a rover in planetary exploration, the KITTI dataset is not representative enough of the environment that our robot will be in. Indeed, on another planet there will probably not be as many features to detect and track than in the streets of Karlsruhe (trees, houses, cars). To meet our needs, we sought a second dataset suitable for robust visual odometry, particularly for space exploration, where other reliable positioning sources are scarce, and the environment is challenging due to the lack of texture in primarily rocky images. We selected the Morocco-Acquired Data set of Mars-Analog eXploration (MADMAX) dataset, recorded in the Moroccan desert, which closely resembles the Martian environment [47].

MADMAX is a dataset published in 2021 aiming to offer a support enabling to benchmark visual-inertial odometry system for autonomous planetary rovers, especially for Mars exploration. The data was acquired with a human portable Dynamic Line Rating (DLR) sensor unit to monitor energy consumption. The dataset contains 36 sequences captured at eight locations with varying environmental conditions, covering a combined trajectory length of 9.2 km. It includes time-stamped recordings from monochrome stereo cameras, a color camera, omnidirectional cameras in stereo configuration, and from an inertial measurement unit. Additionally, ground truth position and orientation data, together with their associated uncertainties, were obtained by a real-time kinematic-based algorithm that fuses the global navigation satellite system (GNSS) data of two body antennas. The dataset also provides calibration information for all cameras, including intrinsic and extrinsic, as well as transformation information between sensors.

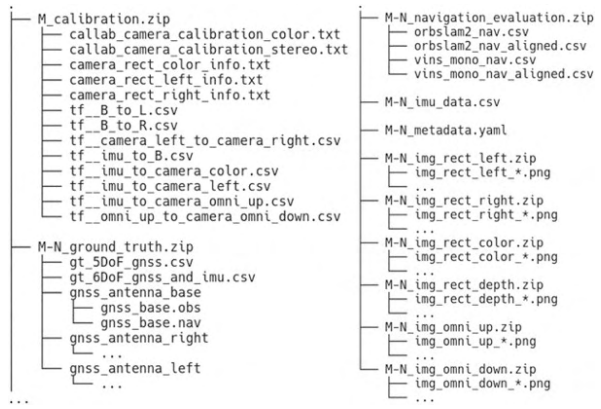


Fig. 8: Structure of the MADMAX dataset for the Location M

In our experiment, we looked at monocular visual odometry therefore, only the left monochrome camera of the stereo setup was used. The data acquisition frequencies used were 14Hz for the monochrome camera, 100Hz for the IMU, and 1Hz for the GNSS receiver for ground truth. Primarily, the sequence A0 was used as its complexity and length were moderate suiting our needs to add synthetic haze afterwards. Our second dataset to test on ORBSLAM3 was the sequence A0 of the MADMAX dataset, on which we added haze synthetically.

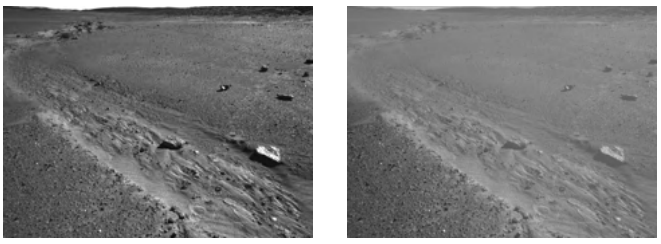


Fig. 9: Image of the MADMAX A0 sequence without fog (left) and with synthetic fog with β = 1 (right)

*3) **Fast Moving Scenes**:* When dealing with rapid motion, the primary artifact in image is motion blur, which is present on some datasets such as the EuRoC MAV dataset [42], specifically in sequences V1-03 and V2-03. However, quantifying motion blur in these datasets is challenging, and is in most cases an unwanted artifact. Consequently, a new dataset was necessary to benchmark the robustness of visual odometry systems against motion blur effectively.

This issue has been previously addressed in order to demonstrate the robustness of the Motion Blur Aware Visual Odometry (MBA-VO) system [25]. Their dataset includes indoor sequences with varying levels of synthetically added motion blur, which would have met our requirements. Unfortunately, this dataset was never published. As a result, we had to create a new dataset.

The baseline for the synthetic dataset aligns with that of the Dynamically Occluded part, utilizing the MADMAX dataset and adhering to the same monocular visual odometry limitations by using only the left monochrome camera. Although the specific sequence used differs from the other part, the constraints regarding length and complexity remain similar. The sequence employed is B1. From this dataset, motion blur can be synthetically added.

Motion blur is a complex topic because blurring can occur for multiple reasons as described in part II. Synthetic creation of motion blur can be done mathematically as shown in 1 described in [48]. Where $\psi(\omega, t)$ is the model describing the influence of optics, shutter, aperture etc. of the camera, Ł(ω, t), ω is the direction of the incoming light and τ is the exposure time.

$$I(\omega, t) = \int_{\tau} \psi(\omega, t) L(\omega, t)\, dt \tag{1}$$

Averaging frames is a simple method for creating motion blur, where a defined number of frames are averaged together to simulate an increased exposure time. This technique is highly effective when the capture framerate is high, ensuring smooth imaging. However, if the framerate is low, stepping artifacts can be introduced.The MADMAX dataset is captured at 14Hz, which is relatively low, prompting an initial exploration of alternative approaches that ultimately proved

ineffective. Thus, averaging frames was chosen as the final method.

The second method tried is viable when the trajectory is known beforehand which is our case with the groundtruth. This approach involves rendering additional frames at intermediary positions through interpolation and then averaging these frames, resulting in better quality motion blur. This method is used in [26], where we adapted it for our needs. However, several issues arose: the rendering time for various levels of motion blur would have taken 4-5 days, and the results were suboptimal because the MADMAX ground truth data is available only at 1Hz, making accurate frame interpolation challenging. Therefore, we came back to the initial method of simple averaging, despite the stepping artifact, creating four levels of motion blur by adjusting the number of frames averaged together. Pictures of the dataset with rendered blur can be seen in Fig 10 with the four levels being 2,4,7 and 12 frames averaged together to make one.

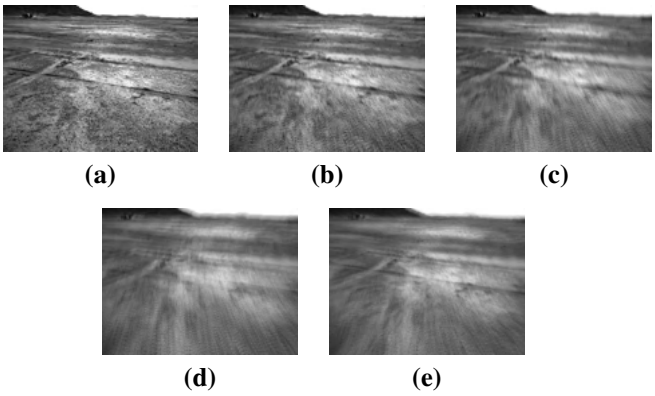


Fig. 10: Pictures of the MADMAX dataset with different levels of blurring **(a)** No blurring **(b)** Average of 2 frames **(c)** Average of 4 frames **(d)** Average of 7 frames **(e)** Average of 12 frames

Finally, the EuRoC dataset was tested on ORB-SLAM3 in Visual Odometry and Visual-Inertial Odometry modes to establish baseline for algorithm efficiency, providing a comparison base for test with other datasets. To run ORB-SLAM3 on the MADMAX dataset, we first had to convert the data to match the structure and file names of EuRoC.

Our plan was to evaluate the standard MADMAX dataset, then test it with the different levels of added motion blur, and finally after the preprocessing step to reduce the blurring. This approach allowed us to compare the different results. The deblurring in the preprocessing employed the three methods described in Part II: Wiener filtering, Richardson-Lucy deconvolution and blind deconvolution.

Granite is a monocular visual odometry algorithm specifically designed for the highly repetitive textures encountered in planetary exploration [49]. This framework, based on Basalt [50], primarly adds the support for monocular systems and rotation-only motion tracking. Although Granite was not the main focus of our study, we briefly evaluated it since it was designed and evaluated using the MADMAX dataset.

C. *Evaluation Method*

1) ***DSO****:* As previously mentioned, our dataset lacks ground truth, but DSO does generate a text file containing estimated coordinates (position and angle). Consequently, we are unable to conduct quantitative comparisons with ground truth data. However, we utilize Pangolin to plot these coordinates, enabling a qualitative evaluation of our method's performance.

2) ***ORB****:* For quantitative evaluation of visual-(inertial) odometry, the process is not straightforward, especially for monocular vision, as the estimated position is precise only to a scale factor, as previously mentioned. A commonly established metric is the Absolute Trajectory Error (ATE), which calculates the error at each point between the estimated trajectory and the ground truth, usually using the root mean square error (RMSE). Another useful metric is the Relative Error (RE), which examines the difference between the movements at each time frame. The main advantage of RE is that the error does not propagate, unlike with ATE [51]. Relative Pose Error (RPE) is another metric designed to evaluate accuracy of SLAM system, comparing the computed map with a groundtruth. In this work, we primarily used ATE.

IV. RESULTS

A. *Complex Lighting Environments*

Throughout the dataset acquisition process, we maintained a consistent camera height and angle. Our objective was to allow movement solely along the x-axis (the length of the sandbox). However, due to limitations in available materials during the dataset capture, complete constraint of the y-axis (width of the sandbox) was unachievable. Consequently, we anticipated a movement trajectory predominantly along the x-axis, albeit with some degree of noise present along the y-axis, prior to conducting DSO testing on this dataset.

After running DSO on our dataset, the trajectory in Fig.11 was plotted.

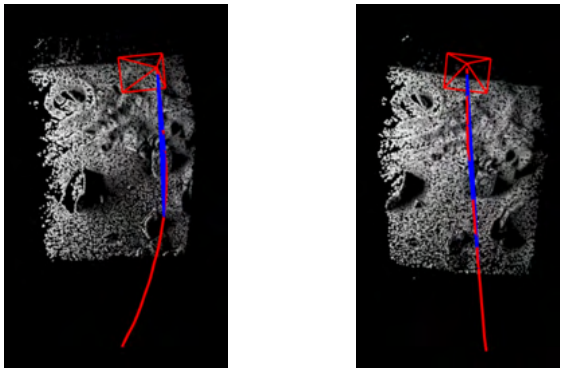


Fig. 11: Plot of camera trajectory and mapping of reference points Left: Low-Sun Sequence Right: High-Sun Sequence

As anticipated, the trajectory depicted in Fig. 11 predominantly follows the x-axis, albeit with notable noise observed

along the y-axis. Since this dataset lacks ground truth and the degree of freedom (DOF) associated with the y-axis remained unlocked, drawing conclusive insights solely from this trajectory is challenging. Here, the information provided by the four locked DOFs, including the z-axis (representing the height of the sandbox), and the camera angles, becomes precious. The evolution of the camera's angle is illustrated in Fig. 12.

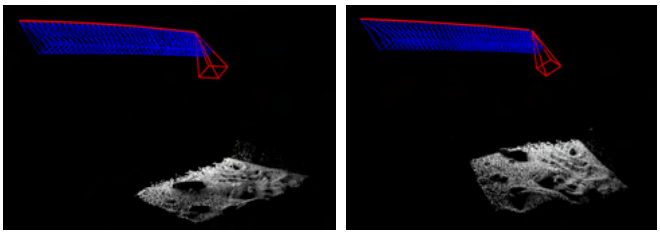


Fig. 12: Plot of camera angle and mapping of reference points Left: Low-Sun Sequence Right: High-Sun Sequence

As depicted in Fig. 12, there is minimal variation observed in the camera's angle and height. Additionally, it is noteworthy that the terrain is accurately represented through the sparse reference points across the two distinct datasets. This qualitative assessment of DSO's accuracy suggests that our hypothesis is correct and that DSO is viable for deployment on extraterrestrial terrains characterized by significant lighting contrasts.

B. Dynamically Occluded Scenes

In order to test ORBSLAM3 on our datasets (with and without fog) to establish a comparison, we first needed to test ORBSLAM3 with the datasets used and tested in their work to verify that we had the same results. To achieve this, we chose to test the MH01 sequence of the EuRoC dataset because of its simplicity. As a measure of performance, we compare our results we the one presented in the ORB-SLAM3 paper with the Absolute Trajectory Error (ATE) metric. The plots and metrics of our implementation and theirs for monocular and stereo vision in the MH01 sequence can be found on the Figure 13.

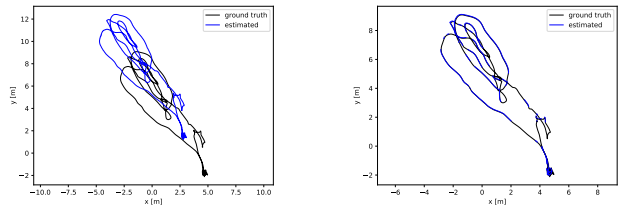


Fig. 13: Plots of trajectories estimated by ORB SLAM 3 for sequence MH with monocular and and stereo vision Left: Monocular MH 01 Right: Stereo MH 01

TABLE I: Comparison of ATE results for monocular and stereo VO on the EuRoC dataset

	Ours	ORB-SLAM3
MH01 monocular	0.015	0.016
MH01 stereo	0.01380	0.029

As we can see, stereo vision enables to have a better performance in Visual Odometry than monocular vision since the estimated path is closer to the ground truth. On the other hand, we see that we get very close results to the ones given in the ORB-SLAM3 paper.

After ensuring that we had similar results on ORB-SLAM3 with the EuRoC dataset, we tested some sequences of the KITTI dataset, to be able to compare them with our foggy dataset (same sequences of KITTI but with synthetically-added haze). We therefore tested the stereo vision sequences 01, 07 and 09 of KITTI on ORB-SLAM3. The evaluation of the results were produced with evo [52]. This evaluation method uses the metrics of Absolute Pose Error (APE) and Relative Pose Error(RPE) and gives exhaustive statistics on these metrics to evaluate the accuracy of the estimated trajectory with respect to the ground truth trajectory. APE measures the difference between the estimated trajectory and the ground truth trajectory. It quantifies the overall accuracy of the estimated trajectory in terms of position and orientation. The APE is computed as the Euclidean distance between corresponding points on the estimated and ground truth trajectories. RPE measures the local accuracy of the trajectory by comparing the relative motion between successive time steps in the estimated trajectory to the relative motion in the ground truth trajectory.
With this evaluation method, we can plot for each KITTI sequence a map of the trajectories (estimated and ground truth), a map of the APE along the trajectory and a map of the RPE along the trajectories. The plots of the trajectories can be found on Figure 14, the plots of the APE along the trajectories on Figure 15 and Table II recaps the mean APE and RPE for each sequence.

TABLE II: Comparison of the mean APE and mean RPE results for sequences 01, 07 and 09 of KITTI dataset

	KITTI 01	KITTI 07	KITTI 09
mean APE	8.697666	0.518549	1.556631
mean RPE	0.048217	0.013699	0.018796

We observed that the accuracy of the estimated trajectories by ORB-SLAM3 were different depending on the dataset used. Indeed, thanks to the mean of the metrics (APE and RPE), we observed that, with ORB-SLAM3, the estimated trajectory for the sequence 07 of the KITTI dataset is more

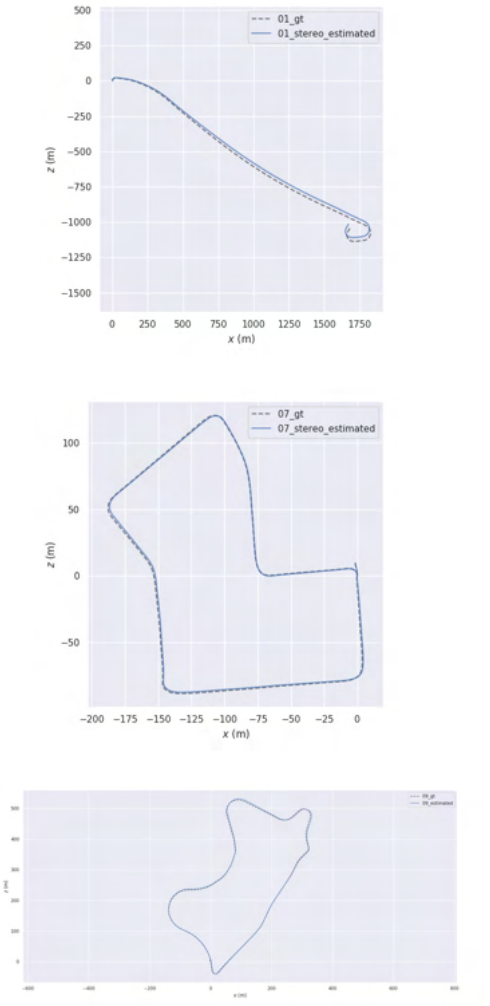


Fig. 14: Plots of trajectories (estimated and ground truth) by ORB SLAM 3 for KITTI sequences 01, 07 and 09 (from top to bottom)

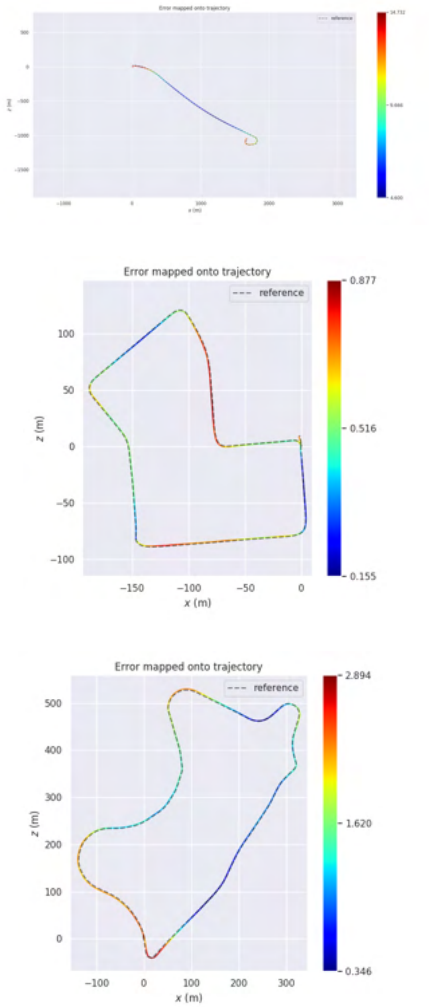


Fig. 15: Plots of APE on trajectory for KITTI sequences 01, 07 and 09 (from top to bottom)

accurate than the other two and that the estimation for KITTI 01 is drastically worse (higher APE and RPE).

The next step was then to test these sequences of the KITTI dataset with synthetically-added fog on ORB-SLAM3. Unfortunately, we could only test our KITTI 01 foggy dataset (with β = 1 and airlight = 150). Figure 16 shows the plotted trajectories (estimated and ground truth) of this sequence and the metrics are summed up in Table III. We observed that with a light fog added on the images, the estimated trajectory was less accurate but still acceptable. We assume that with a denser fog, the results would only worsen.

Finally, the last step was to test a more relevant dataset to our research on ORB-SLAM3, the MADMAX dataset. We tested the A0 sequence of this dataset, as shown in Fig 17, but didn't succeed in testing the foggy A0 sequence we had created.

C. *Fast Moving Scenes*

The first step was to make ORB-SLAM3 work and verify the results on the EuRoC dataset. The same sequence (MH01) as in the previous part was chosen and evaluated in VIO. The Absolute Trajectory Error calculated from our tests and the results reported in the ORB-SLAM3 paper can be seen in Table IV. While there are some differences between our results and those reported, this is not surprising given that the parameters in the original benchmark are likely fine-tuned for each situation. Additionally, the improvement from incorporating the IMU is not significant for such a simple trajectory.

For more complex trajectory from the EuRoC dataset, *Vicon Room 1 03* (V103) and *Vicon Room 2 03* (V203) were the movements are fast and includes some motion blur. The Visual-inertial system improves the results significantly as we can see in Fig 18.

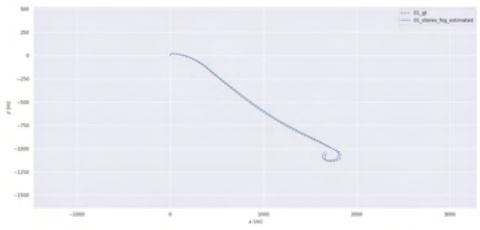


Fig. 16: Estimated and ground truth trajectories for KITTI 01 wth fog run on ORB-SLAM3

TABLE III: Comparison of the accuracy of the estimated trajectories for KITTI 01 with and without fog

	KITTI 01 without fog	KITTI 01 without fog
max APE	14.732210	17.568340
mean APE	8.697666	10.757595
median APE	8.217717	8.535818
min APE	4.600186	6.723626
max RPE	0.294083	0.218725
mean RPE	0.048217	0.048266
median RPE	0.045501	0.045480
min RPE	0.005333	0.007006

The next step was to run ORB-SLAM3 on the MADMAX dataset. Initially, the results were disappointing. The primary issue occurred at the halfway point, where a sharp 360° turn caused the system to lose track of features, leading to reinitialization or a shifted trajectory in the second half, as shown in Fig. 19. This problem arises because ORB-SLAM3 struggles with rotation-only movements.

To address this, we adjusted the ORB parameters to better understand their influence and improve trajectory estimation. The parameters adjusted were:

- *nFeatures*: controls the maximum number of features retained in each frame.
- *scaleFactor*: describes how much the image is scaled down at each level of the pyramid.
- *nLevels*: the number of levels in the pyramid.
- *iniThFAST*: controls the initial number of detected FAST features necessary to begin visual odometry
- *minThFAST*: the minimum threshold for the FAST detector before resetting.

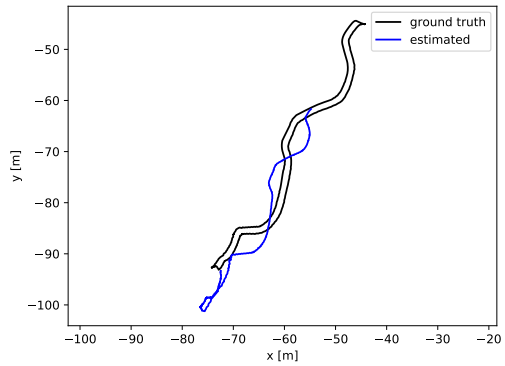


Fig. 17: Mapped trajectory using ORB-SLAM3 on the MADMAX dataset, Sequence A0

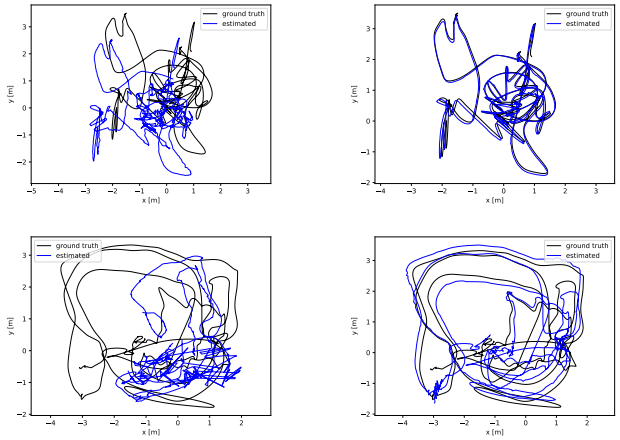


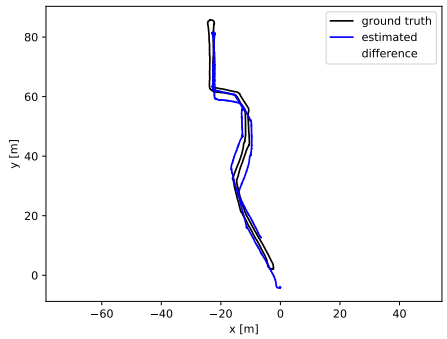
Fig. 18: Mapped trajectory of the EuRoc dataset Top: Vicon Room 1 03 Bottom: Vicon Room 2 03 Left: VO Right: VIO

TABLE IV: Comparison of ATE results for monocular visual odometry on the EuRoC dataset

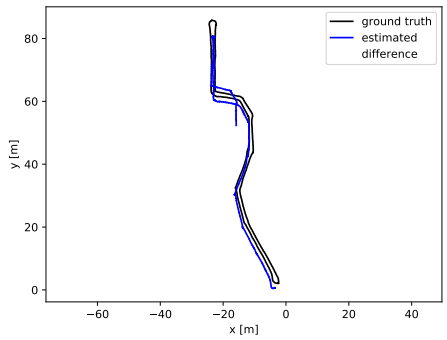
	VO	ORB-SLAM3	VIO	ORB-SLAM3
MH01	0.015	0.016	0.050	0.062
V103	0.183	0.033	0.052	0.037
V203	1.290	-	0.277	0.027

In theory, increasing the number of detected features should improve accuracy, with a tradeoff on computation time. Similarly, adjusting the scale factor and the number of levels in the pyramid, by scaling less at each layer or adding more levels, should allow ORB to detect more features across a wider range of scales, thus increasing precision but at higher computational cost. The last two parameters are different and should not matter as much on the overall performance, but decreasing the minimum number of FAST keypoint detected should reduce the probability of the system to get lost. In practice, the system's complexity means these adjustments do not always yield straightforward improvements. As shown in Table V with some exemple of trajectory in Fig. 19, increasing *nFeatures* does not automatically lead to better results, as the ATE can increase and the number of pairs (keyframes matched between the trajectory estimation and the ground truth) can decrease in some cases. Similar conclusions can be drawn for other parameters, demonstrating that tuning ORB parameters is not a straightforward process and requires fine-tuning for each environment to achieve the best trajectory estimation.

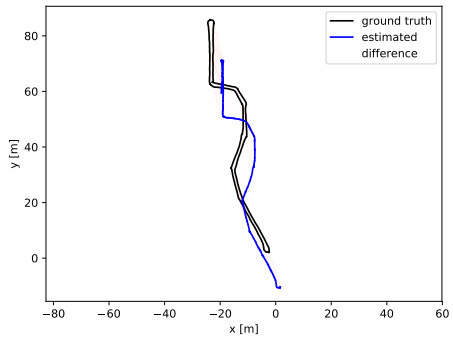
Granite, another visual-inertial algorithm described in Part III, was also briefly tested, as its primary strength is its performance in low-texture environments like those in the MADMAX dataset. However, the initial results were not convincing, leading us to discontinue its use and focus on ORB-SLAM3 instead.



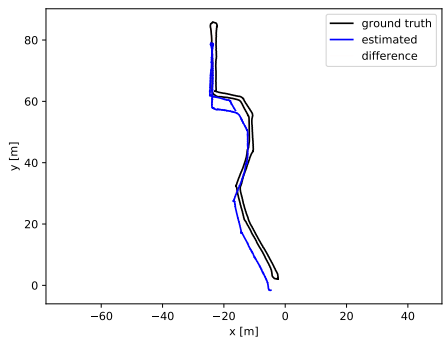
2000 1.2 8 20 6



1500 1.2 8 20 7



2000 1.1 8 20 7



2000 1.3 8 20 7

Fig. 19: Examples of mapped trajectory with the MADMAX dataset with with corresponding ORB parameters (*nFeatures*, *scaleFactor*, *nLevels*, *iniThFAST*, *minThFAST*)

TABLE V: Comparison of results for monocular visual odometry on the EuRoC dataset with different ORB parameters

nFeatures	scaleFactor	nLevels	iniThFAST	minThFAST	pairs	ATE [m]
2000	1.3	11	20	9	189	0.480
1000	1.2	8	20	7	346	1.162
1500	1.2	8	20	7	276	2.876
2000	1.2	8	20	7	347	1.148
2500	1.2	8	20	7	345	5.255
3000	1.2	8	20	7	343	5.246
2000	1.1	8	20	7	221	1.109
2000	1.3	8	20	7	251	1.845
2000	1.4	8	20	7	345	5.932
2000	1.2	9	20	7	348	5.990
2000	1.2	10	20	7	346	5.972
2000	1.2	12	20	7	344	6.917
2000	1.2	8	20	6	345	4.570

The integration of IMU measurements in the evaluation of the MADMAX dataset presented significant challenges. Contrary to expectations, the results were worse with IMU data, as the system failed during abrupt motions where IMU data should theoretically be advantageous. The primary issue likely lies in the calibration of the IMU. The MADMAX documentation lacks precision, complicating the retrieval of necessary calibration data. Additionally, verifying the calibration data is complex. The camera intrinsics were calculated using a slightly different method than the state-of-the-art. Typically, chessboards are used for camera calibration, as mentioned in the calibration process for the complex lighting dataset. This method is also employed in MADMAX, but the larger chessboard size exceeds the borders of the image, rendering standard calibration programs ineffective. Moreover, the free software suggested in the MADMAX documentation is no longer supported. Speculating on how Visual-Inertial Odometry would impact results under blurring, it is likely that the improvement would be minimal during most of the run, except during sharp rotations where the main issues occurred potentially impacting the rest of the run.

The results on the synthetically blurred dataset were straightforward: ORB-SLAM3 struggled significantly, even failing to find enough keypoints to initialize on the first level of blurring. This outcome is understandable, as ORB-SLAM3 relies heavily on detecting distinct features, and blurring effectively diminishes the clarity of these features. In our proposed dataset, every frame contained motion blur, representing an extreme case, yet one that can realistically occur, especially in low-light conditions where longer exposure times are needed to capture sufficient light. These conditions are particularly challenging for visual odometry systems, as the lack of sharp features makes it difficult for the algorithms to accurately track movement and maintain a reliable trajectory.

To explore the preprocessing step aimed at deblurring images, we tested the three conventional methods presented in Section II: Wiener filtering, Richardson-Lucy deconvolution, and blind deblurring. Given that the motion in our dataset is non-linear and the Point Spread Function (PSF) is unknown, blind deblurring is theoretically the most suitable method.

However, it is also the most computationally intensive, significantly increasing the processing time. Therefore, we also tested the other two techniques using estimated PSF and noise. The results were unsatisfactory both in terms of the quality of the recovered images and computation time. Even Wiener filtering, which is relatively simpler, took around 3-5 seconds per image on an Intel Core i5-7300U, which is far too slow for real-time applications. Deep learning-based methods could potentially offer more efficient solutions, but they would require extensive training and computational resources.

V. DISCUSSION

A. *Complex Lighting Environments*

We acknowledge that our study lacks a rigorous scientific approach and measurable outcomes, primarily due to constraints in time. Our project timeline necessitated practicality and efficiency, leading us to prioritize execution over adherence to traditional scientific methodologies.

Because of the elapsed time since DSO's last update, getting it to work initially took an important amount of time, the most important part of our time was dedicated to work around deprecated dependencies, such as Aruco and OpenCV. Consequently, the original plan to integrate DSO with a patch affine illumination model was ultimately abandoned.

Based on the initial findings from our study, we find DSO to be a promising solution for addressing challenges posed by complex lighting conditions. The algorithm effectively navigates through intricate static lighting scenarios. Although it is noteworthy that DSO demands a substantial amount of computational resources, reaching an average processing speed of 7 frames per second on 2020 Intel processors. However, it's important to note that this performance can be significantly enhanced by avoiding the use of Pangolin, which plots all reference points, trajectory, and camera angles. During our study, our primary contribution involved ensuring the functionality of DSO on modern Ubuntu distribution systems. This was achieved by updating all files reliant on deprecated versions of its dependencies and facilitating access to older versions of dependencies that were sometimes challenging to locate. Additionally, we generated a dataset that emulates terrains and lighting conditions akin to those found on other planets and calibrated it to align with DSO's standard parameters to a certain extent. Furthermore, we endeavored to streamline the dataset acquisition and calibration process within the framework of DSO by developing Python scripts. We also enhanced clarity by providing comprehensive steps for dataset, as existing sources lacked explicit instructions.

The suggested course of action in order to increase the algorithm's robustness and making it a reliable tool for the rover in future developments would be the following;

The first priority should be to understand and complete the procedure for DSO's photo-calibration. This step holds potential to significantly improve DSO's accuracy and robustness, particularly in environments with varying lighting conditions. Since this aspect posed a major challenge in our study, the primary hypothesis for our inability to achieve success lies in either a lingering error associated with deprecated code or the possibility that the photometric response function did not exhibit a strictly decreasing trend on our photo-calibration dataset. This anomaly could stem from inaccuracies in the camera's exposure increment function or may simply be attributed to the inherent properties of the camera concerning exposure time adjustments.

To enhance the robustness of DSO further, the incorporation of a patch affine illumination model, as detailed in "Robust Visual Odometry to Irregular Illumination Changes with RGB-D camera" (cite7353893), emerges as the subsequent step. Such integration holds the promise of mitigating the potential for inaccurate odometry information.

While integrating the two aforementioned improvements would undoubtedly boost DSO's performance, it's important to note that they are not mandatory. If prioritizing development time is crucial, these enhancements can be omitted.

Before integrating DSO into a rover for real-time usage, the crucial final step entails developing an OutputWrapper compatible with ROS. This script dictates the format of DSO's output, encompassing the camera's coordinates and angles. Although the camera's angle data may be dispensable given its fixed position on the rover, it could offer valuable insights into the vehicle's state in instances of a malfunctioning or absent IMU.

B. *Dynamically Occluded Scenes*

Due to a lack of time and means, we acknowledge that our research was not as rigorous and deep as we had initially planned. We still created a foggy dataset by synthetically adding haze to some sequences of the KITTI dataset and of the MADMAX dataset. We also successfully tested the ORB-SLAM3 algorithm with the EuroC dataset and afterwards with the KITTI sequences 01, 07 and 09. Finally we were able to test one of the sequences of our foggy dataset and compare it to the original one. For future research, it would be advisable to test different densities of fog on the images. We were able to detect a decrease of the accuracy of the estimated trajectories by ORB-SLAM3 in the presence of dynamical occlusions (fog in our case). Although, the accuracy was not as good, we also observed that ORB-SLAM3 was rather robust and still performed rather well in this case of dynamical occlusions.

However this research still needs to be pursued to fully understand why VO algorithms do not perform as well in the presence of dynamical occlusions and subsequently understand how to improve them to make them more robust to it, especially in the case of planetary exploration.

The first course of action to make our work more robust would be to test a relevant dataset for the application of a rover on a planetary exploration (such as MADMAX)

with synthesized haze. Additionally, To make this research more robust, we suggest creating a real outdoor VO dataset containing haze or fog to test it. Indeed, the method we used to synthesize fog on the images of the dataset made it only possible to add an homogeneous fog. This occlusion turned out to be not as dynamical as a real fog or dust storm would be, and it is probably why ORB-SLAM was as robust in detecting features through it. It would hence be useful to test a more extreme dataset and try to pre-process the images (using dehazing methods) before the VO algorithm analyses them.

If the suggested course of action is not helpful in making ORB-SLAM3 more robust to dynamical occlusions, it might be preferable to try to use Visual Inertial Odometry which is more precise than VO.

C. *Fast Moving Scenes*

Even though the initially proposed plan did not yield optimal trajectory estimation performance, the results with the blurred MADMAX dataset were predictable. Feature-based visual odometry methods, such as ORB-SLAM3, rely heavily on detecting sharp features. The extreme conditions of low texture combined with widespread motion blur created a particularly challenging scenario. The rendering of the motion blur could have been improved by not adding it to every frame or with proper interpolation if the dataset had a higher groundtruth frequency. Despite these difficulties, ORB-SLAM3 demonstrated robustness with the standard MADMAX and EuRoC datasets. However, image preprocessing methods proved to be impractical for real-time systems due to their high computational demands

One promising direction for future research is the exploration of deep learning-based deblurring methods for preprocessing. These methods are significantly faster and could potentially be integrated into real-time systems. However, developing effective deep learning models requires extensive datasets, which can be difficult to create, particularly for systems that need to be robust against not only motion blur but also other artifacts.

Another path worth exploring is the use of datasets with high texture, even though they may be less representative of the ultimate goal of space exploration. High-texture environments can provide more information about the robustness of visual odometry systems, allowing to identify failure points more precisely. This approach could yield valuable insights into the performance limits and strengths of VO systems under varying conditions.

Investigating direct visual odometry systems could also reveal additional strengths. Direct methods, which use the intensity of pixels rather than extracted features, might handle low-texture and blurred environments differently. These systems could provide complementary benefits and help overcome some of the limitations encountered with feature-based methods.

Perhaps the most promising path is the use of visual-inertial odometry. VIO provide high-precision data during fast motions but is not a straightforward implementation. IMUs can compensate for the errors in visual data over short timeframes, potentially enhancing the overall accuracy and robustness of the system but could not be shown in this work.

D. *Simultaneous Implementation of the developed algorithms*

The subject of the integration of three different algorithm inside a single system hasn't yet been mentioned. However it is an important subject as the final goal of the project is to be able to use all the developed algorithms on different camera technologies and be able to switch autonomously during a mission without the need of human interference.

The primary approach to integrating multiple Visual Odometry (VO) algorithms into a robot's navigation system is through dynamic selection. This entails the robot autonomously switching between different VO algorithms based on the prevailing environmental conditions. For instance, when encountering occluded scenes, the robot could utilize the Dynamically Occluded Scenes algorithm. In scenarios characterized by rapid motion, it could seamlessly transition to the Fast Moving Scenes algorithm. Similarly, in environments with challenging lighting conditions, the robot might opt for the Complex Lighting Environments algorithm.

However, implementing dynamic selection poses certain challenges. Firstly, the robot must possess the capability to discern between the various types of scenes accurately. This necessitates robust perception capabilities, which may require sophisticated sensor fusion techniques to interpret data from cameras, LiDAR, IMUs, and other sensors effectively. Additionally, the accuracy of position estimation heavily relies on the performance of the selected algorithm. Therefore, if the chosen algorithm fails to accurately estimate the robot's pose, it could lead to navigation errors and compromise the overall reliability of the system.

An alternative approach to integrating multiple Visual Odometry algorithms into a robot's navigation system is through sensor fusion. Unlike dynamic selection, which involves algorithm switching based on environmental conditions, sensor fusion combines data from multiple cameras simultaneously to enhance pose estimation across various scenarios.

However, implementing sensor fusion with the three VO algorithms introduces challenges related to computational resources. Processing data from multiple cameras concurrently requires significant computational power, potentially straining the onboard computing resources of the robot. This increased computational demand could lead to delays in real-time processing or compromise the system's overall performance if not adequately addressed.

Sensor fusion with the three VO algorithms would involve integrating data from all cameras concurrently, providing a more comprehensive and continuous assessment of the robot's surroundings. This simultaneous fusion of data minimizes the need for real-time decision-making and reduces the

risk of delays or errors associated with algorithm selection.

Moreover, sensor fusion inherently incorporates redundancy and resilience to sensor failures or limitations. While dynamic selection may struggle to cope with sudden changes in environmental conditions, sensor fusion maintains robustness by leveraging diverse sources of visual information simultaneously. This redundancy enhances the system's reliability and robustness, ensuring accurate pose estimation even in dynamic and challenging environments.

Furthermore, sensor fusion enables the exploitation of synergies between the different VO algorithms integrated into the system. By combining data from multiple cameras specialized for specific environmental challenges, sensor fusion enhances the overall performance and adaptability of the system. This collaborative approach allows the robot to navigate more effectively through a wide range of scenarios, from dynamically occluded scenes to fast-moving environments and complex lighting conditions.

In summary, while dynamic selection offers flexibility in adapting to changing environmental conditions, sensor fusion with multiple cameras provides a more comprehensive and robust solution for pose estimation in diverse scenarios. By capitalizing on the strengths of each camera concurrently, sensor fusion enhances navigation performance and autonomy in complex and dynamic environments, reducing the reliance on real-time algorithm selection and improving overall system resilience. However, addressing the computational challenges associated with sensor fusion is crucial to ensure optimal performance and real-time responsiveness of our system.

VI. CONCLUSIONS

In this research, we investigated the performance and robustness of Visual Odometry algorithms under challenging imaging conditions, specifically complex lighting environments, dynamically occluded scenes, and fast-moving scenarios. Our study aimed to identify the limitations of existing VO algorithms and explore potential improvements for enhancing their accuracy and reliability in planetary exploration missions.

Our comprehensive review and subsequent tests revealed that feature-based VO methods, such as those employing SIFT, SURF, and ORB, are generally effective under normal conditions but struggle in low-texture or dynamically changing environments. Direct methods, which leverage pixel intensity gradients, showed promise in handling some of these challenging conditions but still faced significant limitations in fast-moving scenes due to computational demands.

The introduction of Visual-Inertial Odometry presented a notable improvement, as the integration of inertial measurements helped to mitigate some of the issues encountered by purely visual methods. However, VIO's performance was still hampered by low-light conditions and the need for high computational resources.

Through our experimental work with DSO, we identified that while this algorithm performs well in static lighting conditions, its real-time application is constrained by high computational requirements. Additionally, our efforts to integrate a patch affine illumination model to enhance DSO's robustness against varying lighting conditions were partially successful, highlighting the need for further development and optimization.

Looking forward, the dynamic selection of VO algorithms based on environmental conditions and the implementation of sensor fusion techniques were identified as promising paths for future research. Sensor fusion, in particular, offers the potential to combine the strengths of multiple algorithms, providing a more robust and comprehensive solution for pose estimation in diverse and dynamic environments.

In conclusion, while current VO algorithms provide a solid foundation for navigation in extraterrestrial missions, significant advancements are necessary to overcome the challenges posed by complex visual fields. Future work should focus on refining these algorithms, integrating advanced sensor fusion techniques, and optimizing computational efficiency to ensure reliable and accurate navigation for planetary exploration robots.

REFERENCES

[1] M. O. A. Aqel, M. H. Marhaban, M. I. Saripan, and N. B. Ismail, "Review of visual odometry: types, approaches, challenges, and applications," *SpringerPlus*, vol. 5, 10 2016.

[2] M. Maimone, Y. Cheng, and L. Matthies, "Two years of Visual Odometry on the Mars Exploration Rovers," *Journal of field robotics*, vol. 24, pp. 169–186, 3 2007.

[3] A. Mallios, P. Ridao, D. Ribas, M. Carreras, and R. Camilli, "Toward autonomous exploration in confined underwater environments," *Journal of field robotics*, vol. 33, pp. 994–1012, 11 2015.

[4] D. C. Jakob Engel, Vladlen Koltun, "Direct sparse odometry," 3 2018.

[5] D. Nister, O. Naroditsky, and J. Bergen, "Visual odometry," in *Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2004. CVPR 2004.*, vol. 1, pp. I–I, 2004.

[6] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International journal of computer vision*, vol. 60, pp. 91–110, 2004.

[7] H. Bay, T. Tuytelaars, and L. Van Gool, "Surf: Speeded up robust features," in *Computer Vision–ECCV 2006: 9th European Conference on Computer Vision, Graz, Austria, May 7-13, 2006. Proceedings, Part I 9*, pp. 404–417, Springer, 2006.

[8] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, "Orb: An efficient alternative to sift or surf," in *2011 International conference on computer vision*, pp. 2564–2571, Ieee, 2011.

[9] M. A. Fischler and R. C. Bolles, "Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography," *Communications of the ACM*, vol. 24, no. 6, pp. 381–395, 1981.

[10] M. Calonder, V. Lepetit, C. Strecha, and P. Fua, "Brief: Binary robust independent elementary features," in *Computer Vision–ECCV 2010: 11th European Conference on Computer Vision, Heraklion, Crete, Greece, September 5-11, 2010, Proceedings, Part IV 11*, pp. 778–792, Springer, 2010.

[11] M. Bloesch, S. Omari, M. Hutter, and R. Siegwart, "Robust visual inertial odometry using a direct ekf-based approach," in *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 298–304, 2015.

[12] J. Engel, T. Schöps, and D. Cremers, *LSD-SLAM: Large-Scale Direct Monocular SLAM*. 1 2014.

[13] P. Kim, B. Coltin, O. Alexandrov, and H. J. Kim, "Robust visual localization in changing lighting conditions," in *2017 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 5447–5452, 2017.

[14] T. Smith, J. Barlow, M. Bualat, T. Fong, C. Provencher, H. Sanchez, and E. Smith, "Astrobee: A new platform for free-flying robotics on the international space station," in *International Symposium on Artificial Intelligence, Robotics, and Automation in Space (i-SAIRAS)*, no. ARC-E-DAA-TN31584, 2016.

[15] P. Kim, H. Lim, and H. J. Kim, "Robust visual odometry to irregular illumination changes with rgb-d camera," in *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 3688–3694, 2015.

[16] A. Solin, S. Cortes, E. Rahtu, and J. Kannala, "Pivo: Probabilistic inertial-visual odometry for occlusion-robust navigation," in *2018 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pp. 616–625, 2018.

[17] M. Coppola, K. N. McGuire, C. De Wagter, and G. C. H. E. De Croon, "A survey on Swarming with Micro Air vehicles: Fundamental challenges and constraints," *Frontiers in robotics and AI*, vol. 7, 2 2020.

[18] X.-Y. Kuo, C. Liu, K.-C. Lin, and C.-Y. Lee, "Dynamic attention-based visual odometry," in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pp. 160–169, 2020.

[19] K. He, J. Sun, and X. Tang, "Single image haze removal using dark channel prior," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 12, pp. 2341–2353, 2011.

[20] Q. Zhu, J. Mai, and L. Shao, "A fast single image haze removal algorithm using color attenuation prior," *IEEE Transactions on Image Processing*, vol. 24, no. 11, pp. 3522–3533, 2015.

[21] C. O. Ancuti and C. Ancuti, "Single image dehazing by multi-scale fusion," *IEEE Transactions on Image Processing*, vol. 22, no. 8, pp. 3271–3282, 2013.

[22] B. Cai, X. Xu, K. Jia, C. Qing, and D. Tao, "Dehazenet: An end-to-end system for single image haze removal," *IEEE Transactions on Image Processing*, vol. 25, no. 11, pp. 5187–5198, 2016.

[23] B. Li, X. Peng, Z. Wang, J. Xu, and D. Feng, "Aod-net: All-in-one dehazing network," in *2017 IEEE International Conference on Computer Vision (ICCV)*, pp. 4780–4788, 2017.

[24] A. Pretto, E. Menegatti, M. Bennewitz, W. Burgard, and E. Pagello, "A visual odometry framework robust to motion blur," in *2009 IEEE International Conference on Robotics and Automation*, pp. 2250–2257, 2009.

[25] P. Liu, X. Zuo, V. Larsson, and M. Pollefeys, "MBA-VO: Motion Blur Aware Visual Odometry," *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, 10 2021.

[26] C. Mannila, *Robustness of State-of-the-Art Visual Odometry and SLAM Systems*. Dissertation, Unknown, 2023.

[27] P. A. M and N. Wiener, "The Extrapolation, Interpolation and Smoothing of Stationary Time Series, with Engineering Applications.," *Journal of the Royal Statistical Society. Series A, General*, vol. 113, p. 413, 1 1950.

[28] L. Guan and R. Ward, "Restoration of randomly blurred images by the Wiener filter," *I.E.E.E. transactions on acoustics, speech, and signal processing*, vol. 37, pp. 589–592, 4 1989.

[29] W. H. Richardson, "Bayesian-Based iterative Method of image restoration*," *Journal of the Optical Society of America*, vol. 62, p. 55, 1 1972.

[30] N. Y.-W. Tai, N. P. Tan, and M. S. Brown, "Richardson-Lucy Deblurring for Scenes under a Projective Motion Path," *IEEE transactions on pattern analysis and machine intelligence*, vol. 33, pp. 1603–1618, 8 2011.

[31] A. Levin, Y. Weiss, F. Durand, and W. T. Freeman, "Understanding and evaluating blind deconvolution algorithms," *2009 IEEE Conference on Computer Vision and Pattern Recognition*, 6 2009.

[32] C. J. Schuler, M. Hirsch, S. Harmeling, and B. Schölkopf, "Learning to deblur," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 38, no. 7, pp. 1439–1451, 2016.

[33] J. Sun, W. Cao, Z. Xu, and J. Ponce, "Learning a convolutional neural network for non-uniform motion blur removal," 2015.

[34] S. Nah, T. H. Kim, and K. M. Lee, "Deep multi-scale convolutional neural network for dynamic scene deblurring," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 257–265, 2017.

[35] X. Tao, H. Gao, X. Shen, J. Wang, and J. Jia, "Scale-recurrent

network for deep image deblurring," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8174–8182, 2018.
[36] A. I. Mourikis and S. I. Roumeliotis, "A multi-state constraint kalman filter for vision-aided inertial navigation," in *Proceedings 2007 IEEE International Conference on Robotics and Automation*, pp. 3565–3572, 2007.
[37] T. Qin, P. Li, and S. Shen, "Vins-mono: A robust and versatile monocular visual-inertial state estimator," *IEEE Transactions on Robotics*, vol. 34, no. 4, pp. 1004–1020, 2018.
[38] D. Scaramuzza and Z. Zhang, *Aerial Robots, Visual-Inertial Odometry of.* 1 2020.
[39] M. Servières, V. Renaudin, A. Dupuis, and N. Antigny, "Visual and Visual-Inertial SLAM: state of the art, classification, and experimental benchmarking," *Journal of sensors*, vol. 2021, pp. 1–26, 2 2021.
[40] C. Campos, R. Elvira, J. J. G. Rodríguez, J. M. M. Montiel, and J. D. Tardós, "ORB-SLAM3: an accurate Open-Source library for Visual, Visual–Inertial, and Multimap SLAM," *IEEE transactions on robotics*, vol. 37, pp. 1874–1890, 12 2021.
[41] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun, "Vision meets robotics: The KITTI dataset," *The international journal of robotics research*, vol. 32, pp. 1231–1237, 8 2013.
[42] M. Burri, J. Nikolic, P. Gohl, T. Schneider, J. Rehder, S. Omari, M. W. Achtelik, and R. Siegwart, "The EuRoC micro aerial vehicle datasets," *The international journal of robotics research*, vol. 35, pp. 1157–1163, 1 2016.
[43] J. Engel, V. Usenko, and D. Cremers, "A photometrically calibrated benchmark for monocular visual odometry," *arXiv preprint arXiv:1607.02555*, 2016.
[44] T. F. Margaret Hansen, Uland Wong, "NASA POLAR Traverse Dataset," 2023.
[45] Z. Zhang, "A flexible new technique for camera calibration," *IEEE Transactions on pattern analysis and machine intelligence*, vol. 22, no. 11, pp. 1330–1334, 2000.
[46] L.-A. Tran, C. N. Tran, D.-C. Park, J. Carrabina, and D. Castells-Rufas, "Toward improving robustness of object detectors against domain shift," in *2024 International Conference on Green Energy, Computing and Sustainable Technology (GECOST)*, pp. 01–05, 2024.
[47] L. Meyer, M. Smíšek, A. F. Villacampa, L. O. Maza, D. Medina, M. J. Schuster, F. Steidle, M. Vayugundla, M. G. Müller, B. Rebele, A. Wedler, and R. Triebel, "The MADMAX data set for visual-inertial rover navigation on Mars," *Journal of field robotics*, vol. 38, pp. 833–853, 3 2021.
[48] F. Navarro, F. J. Serón, and D. Gutierrez, "Motion blur rendering: state of the art," *Computer graphics forum*, vol. 30, pp. 3–26, 1 2011.
[49] M. Wudenka, M. G. Muller, N. Demmel, A. Wedler, R. Triebel, D. Cremers, and W. Sturzl, "Towards robust monocular visual odometry for flying robots on planetary missions," *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 9 2021.
[50] V. Usenko, N. Demmel, D. Schubert, J. Stuckler, and D. Cremers, "Visual-Inertial mapping with Non-Linear factor recovery," *IEEE robotics automation letters*, vol. 5, pp. 422–429, 4 2020.
[51] Z. Zhang and D. Scaramuzza, "A tutorial on quantitative trajectory evaluation for visual(-inertial) odometry," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 7244–7251, 2018.
[52] M. Grupp, "evo: Python package for the evaluation of odometry and slam.." https://github.com/MichaelGrupp/evo, 2017.