IAC-24-A3,IP,57,x90773

Infrared Vision-based Navigation for Planetary Landing

Samuele G. Labò[a]*, Silvestrini Stefano [a], Michelle R. Lavagna [a]?**

[a] *Department of Aerospace Science and Technology, Politecnico di Milano, Italy*
* Corresponding Author, samuelegiuseppe.labo@mail.polimi.it
** Corresponding Author, stefano.silvestrini@polimi.it
? Corresponding Author, michelle.lavagna@polimi.it

Abstract

Amongst the most helpful missions for planetary exploration are those planning to land a spacecraft for on-site analysis and experimentation. The task of landing on a planetary surface is not trivial. The lander must be provided with a fully autonomous navigation system capable of handling the extremely complex and delicate descent towards ground. A revolutionary technology in this field has recently witnessed a great successful implementation with JAXAs SLIM mission: Vision-based Navigation. By extracting salient features from the images of one or multiple on-board sensors and tracking them during motion, position and orientation of the spacecraft for each consecutive frame can be retrieved with respect to the observed scenario. This allows to provide the data necessary to carry out landing navigation and hazard avoidance with unprecedented accuracy. The strategy, however, is strongly dependant on the scene's illumination. This work investigates the possibility of freeing Vision-based Navigation from said limit by assessing the possibility of working in the infrared band. If possible, this would significantly improve the flexibility of landing strategy design for future missions. The study is a preliminary evaluation of the effectiveness of visually assessed algorithms on infrared planetary surface frames. The fundamental idea is to exploit the basic architecture of Vision-based navigation systems for planetary landing but providing infrared images as input in place of visual ones. Through a detailed review of many Computer Vision algorithms and the various approaches exploitable for a spacecraft landing scenario, a selection of the most compatible for the study is made. Realistic image sequences simulating the approach of a lander, provided with a monocular infrared sensor, to the Martian surface are created by manipulating real infrared mosaics from Mars Odyssey and used for extensive testing. Algorithms are executed on these image sequences to find out if Infrared Vision-based Navigation for planetary landing is possible and under which conditions. Specifically, robustness against motion disturbance and for various Martian regions is assessed on real infrared frames taken both during the day and at night. After evaluating feasibility, the effectiveness of some simple performance enhancement strategies is demonstrated and the algorithms most compatible with the application identified. Finally, algorithms are run on a BeagleBone Black board to retrieve indications for realistic computational times. Results suggest that the technology requires a lot of work for optimizing both robustness and computational times but definitely has great potential and should be further investigated.

Keywords: Vision-based Navigation, Infrared, Planetary Landing, Autonomous Landing, Computer Vision

Acronyms/Abbreviations

Vision-based Navigation (VN), Autonomous Guidance Navigation and Control (AGNC), Hazard Detectin and Avoidance (HDA), Inertial Measurement Unit (IMU), Infrared (IR), Computer Vision (CV), Central Processing Unit (CPU), Visual Odometry (VO), Visual Simultaneous Localization and Mapping (V-SLAM), Frame Couple (FC), Entry Descent and Landing (EDL).

1. Introduction

The task of landing on a planetary surface is not trivial. The lander must be provided with a fully autonomous navigation system capable of handling the extremely complex and delicate descent towards ground. A revolutionary technology in this field is Vision-based Navigation which is gaining popularity due to the unprecedented accuracy it allows to achieve both for AGNC and for HDA. Many of these successes have been witnessed on Mars: one of the first missions to ever exploit VN for the landing of a spacecraft was Mars Exploration Rovers. DIMES [1] exploited three images in total to track, between each pair, two Harris features using Moravec's pseudo-normalized correlator and then combined with the IMU and altimetry to produce one horizontal velocity estimate, essential for landing safety. Mars 2020 is a more recent example in which VN has been used even more extensively and for an even more important role: the lander touched down with a position error of less than 5 m by fusing landmark matches between descent images and an on-board map with IMU data [2].

Nonetheless, this technology has many aspects that can and will be improved, as proven by the various projects currently in the works [3,4]. One of the biggest limits of VN in general and especially for the task of landing, is that it is well assessed only for the use of visual images. This inevitably binds performances to the illumination conditions of the scene: in fact, the need for brightness makes certain landing sites, like the lunar south pole [5], and times (night) inaccessible.

The idea presented in this work is to investigate the possibility of overcoming this limit by exploiting infrared frames in place of visual ones. Since the concept of IR VN has scarcely been studied, whit just a handful of papers [6] which are not related to space applications, the work has to be set up as a preliminary technological assessment for the purpose of planetary landing.

1.1 Objectives

The aim of the work is to answer the following questions:

- Could an AGNC system for planetary landing exploiting IR VN work?
- Which would be the permitting and prohibitive conditions for the functioning of such a system?
- Which CV algorithms would perform best on IR frames of a planetary surface and under which conditions?

The potential outcome of this field of research, within which this project poses as a first step, could be the development of an Infrared Vision-based Navigation system part of an AGNC system for planetary landing much more flexible than any developed to date, exponentially broadening the horizon of possibilities for landing sites and times.

1.2 Tools

In the attempt to answer the questions of section 1.2, numerical simulations have been implemented on a performing portable computer (Intel® Core i7-8550U CPU at 1.80 GHz and 16 GB of RAM [7]) at first, and then on a BeagleBone Black board (1GHz ARM® Cortex-A8 based CPU and 512MB DDR3L of SDRAM [8]) to retrieve initial indications of more realistic computational times. All image processing functions of the work are developed using the OpenCV-4.5.3 library [9]. The library's source code is in C++, however, the Python interface is exploited for this work. Data post-processing and other computations have also been coded in Python.

2. Methods

2.1 Algorithm Selection

The first key step is the selection of the algorithms to evaluate. To carry out a coherent choice, a reference VN system architecture should be identified. A frame-to-frame motion estimation strategy [10] with a single monocular camera is deemed as the most compatible with the application at hand since it does not depend on the existence of a map of the landing scene and its effectiveness is well assessed in visual image literature. The work is developed to account for aspects influencing the performance of a system solving either a VO or a V-SLAM [11] problem. Considering that the scope is to investigate the feasibility of IR VN for planetary landing, the most basic and well assessed CV strategies and algorithms have been selected for the task. Specifically, a bunch of detection and description algorithms followed by Brute Force matching [12] (here simply referred to as matching) have been considered as well as some detectors directly followed by Pyramidal Lucas-Kanade tracking [13,14] (here simply referred to as tracking).

Out of all the parts of a standard AGNC system exploiting VN for landing, focus is here place on those highlighted in Fig. 1. These are the ones directly affected by the switch in input from visual to infrared frames.

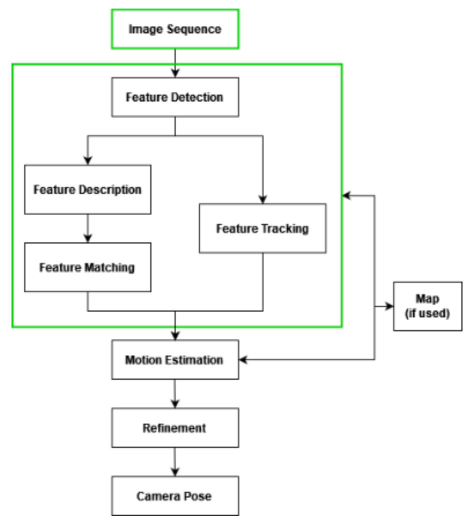


Fig. 1. Typical AGNC system for planetary landing.

There is an extremely large variety of feature detectors and descriptors available in literature and their efficiency strongly depends on the working conditions. Thus, as is well known in the field of CV, there is no generally better performer. The selection of these algorithms must cover most bases but be strict enough to prevent the study from being dispersive. A detailed literature review is performed in search of those keypoint detectors and descriptors which have been proven to perform well on visual images granting invariance to scale, rotation, viewpoint and noise. Similar guidelines are followed for the choice of the aforementioned pairing algorithms, i.e. feature matcher and feature tracker. The detectors and descriptors analysed in this study are summarized in Tab. 1. There are twelve algorithm sequences overall: six detection-description couples each followed by matching, plus the same six detectors each followed by tracking.

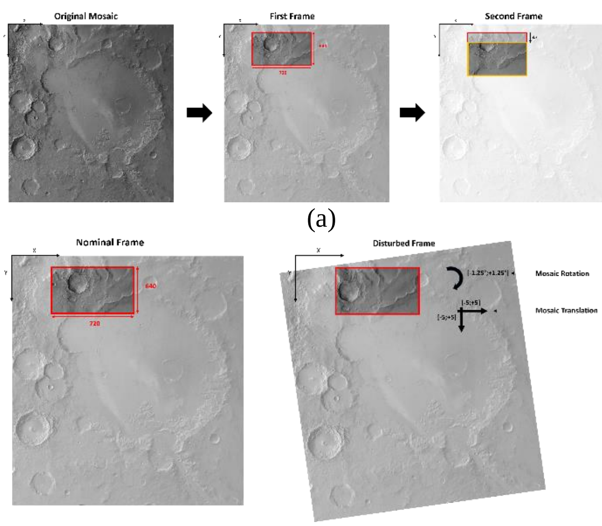
Detector	Descriptor
SIFT	SIFT
FAST	FREAK
STAR	BRIEF
ORB	ORB
BRISK	BRISK

AKAZE	AKAZE

Tab. 1. Summary of selected algorithms.

2.2 Image Sequence

Since the concept of study is quite new and IR images generally do not have any other use for landing, there are no real infrared datasets of a landing sequence. Nonetheless, the Mars Odyssey mission did map the whole surface of Mars in the infrared band through its THEMIS instrument [15]. Therefore, the mosaics created through the THEMIS pictures can be manipulated to simulate the initial segment of a landing sequence. Since the spacecraft orbited the planet at a fixed altitude during the thermal mapping of the surface, this strategy does not allow to account for scale change or out-of-plane rotations. Also, the mosaics have been post-processed and cleaned of sensor noise. Regardless, the manipulation of these infrared mosaics allows to account for in-plane rotations and translations due to motion disturbance, grants the possibility to sample a broad variety of surfaces (more or less rich in ground features), and favours the testing of a fundamental aspect of the technology: daytime performance compared to nighttime performance. Fig. 2 reports a scheme describing the process of creation of the image sequences.



(a)

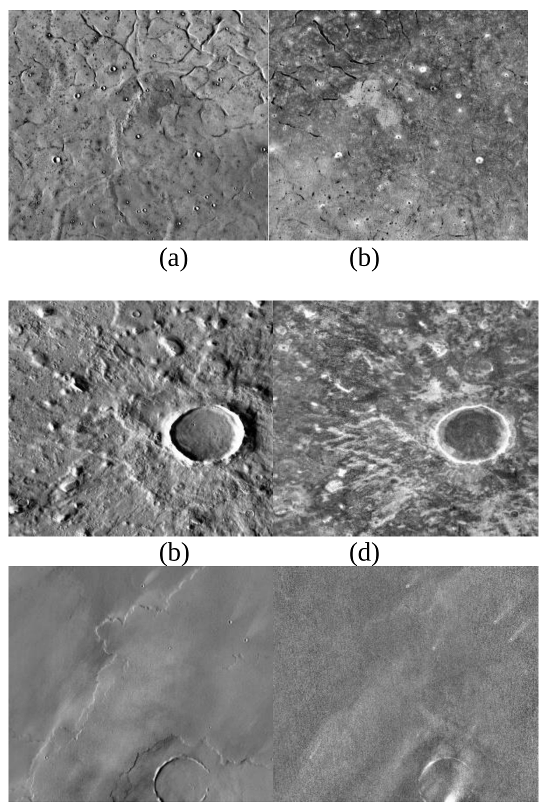
(b)

Fig. 2. Dataset creation process (a) and addition of motion disturbance for the Disturbed datasets (b). Reported values are in pixels. The region framed is the Martian Gusev crater.

A realistic time step between the taking of one picture and the next is $\Delta t = 1$ s, from [16], while a physically sensible motion velocity for the simulated lander is v = 4.7 km/s, from the Tianwen-1 mission [17]. These assumptions, along with the knowledge of the mosaic resolution (100 *mpp*), dictated by the instrument, make the creation of a total of twelve datasets possible. Each image sequence is made of 100 frames of 720*x*640 *px*. The datasets can be divided into two groups: six Nominal datasets for which motion disturbance is not considered and six equivalent but Disturbed datasets where motion disturbance is introduced. The datasets of each group are associated to three different regions of Mars which are Amenthes, Cebrenia and Tharsis. For each of these, one daytime and one corresponding nighttime image sequence are created. The physically sensible disturbance entities along with the other main characteristics of the frame sequences are resumed in Tab. 2.

	Nominal	**Disturbed**
Frames per Sequence	100	100
Frame Resolution	100 *mpp*	100 *mpp*
Frame Size	720*x*640 *px*	720x640 *px*
Nominal Motion	+47 *px*	+47 *px*
Vertical Disturbance	0 *px*	±5 *px*
Horizontal Disturbance	0 *px*	±5 *px*
In-plane Rotation Disturbance	0°	±1.25°
Out-of-plane Rotation Disturbance	0°	0°

Tab. 2. Dataset parameters and properties. The ± symbol indicates that a random value between the two extremes is used.



(a) (b)

(b) (d)

(e) (f)

Fig. 3. First image of sequence of studied Martian regions: Amenthes Day (a), Amenthes Night (b), Cebrenia Day (c), Cebrenia Night (d), Tharsis Day (e), Tharsis Night (f).

2.3 Testing and Evaluation Parameters

Algorithms are executed following the sequence represented in Fig. 1. Feature detection is always the first step, followed either by feature description and then matching, or directly followed by tracking. To determine whether the features matched or tracked between a target and a reference frame couple are correct, the geometry information from the dataset generation process is exploited to back project the pairings of the target on the reference. Then, the Euclidean distance between each keypoint on the reference and its corresponding back-projected point is compared to a threshold of 2.5: if the value is greater, the couple is considered an outlier. This has been done according to [18]. After applying the algorithms to the image sequences, performance parameters are retrieved to assess the efficiency of each one for all the essential aspects. Tab. 3 resumes the performance parameters of interest for the work at hand: these are retrieved for each algorithm on each image sequence. The parameters in the table are only defined for matching for conciseness' sake, the definition is equivalent for tracking parameters.

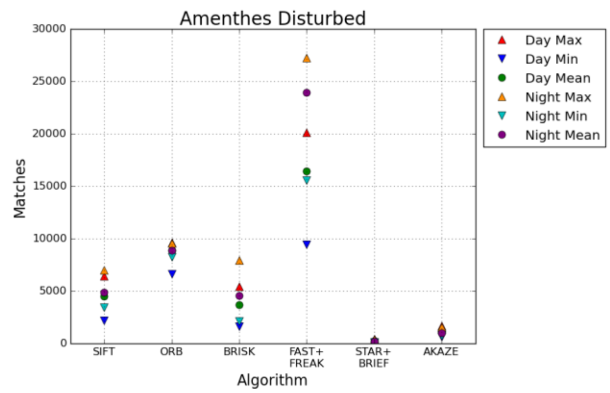
	Symbol	Definition
Matches	m	Total matches found
Correct Matches	Cm	Total correct matches found
Matching Score	MS	Cm/m
Reliability	R	Frames with MS > 0.5
Mean Execution Time	MET	Execution Time Average
Persistence	P	Consecutive FC with Cm>0
Number of Detections	N_d	Total detections for a dataset
Minimum Amount of Information	FC | Cm<50	Frame couples with Cm<50
Zero Information	FC | Cm=0	Frame couples with Cm=0

Tab. 3. Performance parameters.

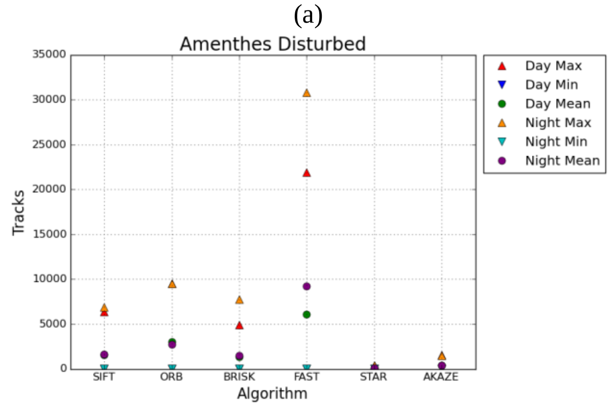
3. Testing

3.1 First Assessment

The first objective is assessing if Vision-based Navigation algorithms can work on infrared planetary surface images and under which conditions. To achieve this, each algorithm sequence presented in section 2.1 is executed on each image sequence presented in section 2.2 and performance parameters from section 2.3 are retrieved for each case. Within the limits of the testing application, results of the First Assessment confirm that Infrared Vision-based Navigation for planetary landing can actually work. Of great importance is the fact that the previous statement is true for both daytime and nighttime. As a matter of fact, an abundance of pairings is found for most algorithms on most datasets with orders of magnitude varying between tens of thousands and hundreds, as shown in Fig. 4. Also, most of these matches or tracks are correct: as highlighted in Fig. 5, average matching and tracking scores vary roughly between 0.9 and 0.6 with values generally lowering when motion disturbance is introduced but not prohibitively. So, motion disturbance does negatively affect both the quantity and quality of pairings found by all algorithms but does not make the application unfeasible. As mentioned, for brightness the opposite is found: there are very little differences between Day and Night performance for the same algorithm sequence meaning that illumination conditions are not a performance-driving factor.

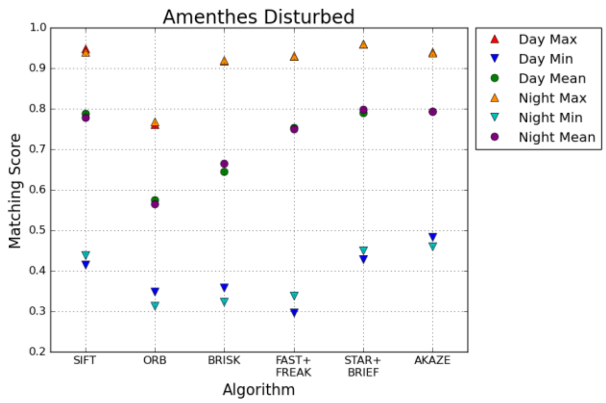


(a)

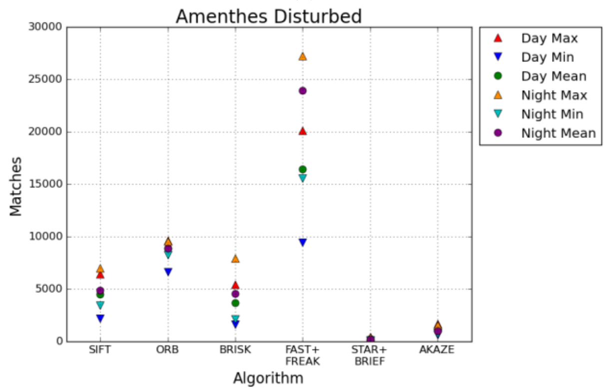


(b)

Fig. 4. Matches (a) and Tracks (b) for the Amenthes Disturbed datasets.



(a)



(b)

Fig. 5. Matching score (a) and tracking score (b) for the Amenthes Disturbed datasets.

As predictable, there are differences depending on the specific case and likely no algorithm, as implemented in this initial evaluation, could satisfy real mission requirements in terms of robustness. However, it is a very promising start. Total failure is rarely witnessed. The only cases are on Tharsis. When detection is performed by STAR roughly 10 keypoints are correctly paired up throughout the dataset and around 20 pairless frame couples are found. No pairings are found on a few frames also in Nominal conditions, meaning that STAR struggles to find and match features on this image sequence and motion disturbance only amplifies the effect. In this particular case, having to work on night frames seems to further affect performance. AKAZE is the only other detector for which a few (less than 10) total failures are registered. This only occurs on Tharsis Night Disturbed and, when motion is not disturbed the algorithm is capable of always finding at least a dozen matches, which implies that disturbances occasionally lead to total match loss. The most favourable region for almost all algorithms seems to be Cebrenia, although average matches are only slightly higher than those of Amenthes. Thus, the infrared images of both these areas can be described as good for finding and pairing features. Algorithms on Tharsis generally show slightly worse robustness with respect to Amenthes and Cebrenia leading to believe that IR VN performance does not depend on the type of ground features in the scene but is favoured by an abundance of them. An interesting observation is that these trends, along with many others found form this analysis, are coherent with those of visual image literature [18,19]. This could imply that frame input discrepancies may not cause significant performance disparities at least for properly post-processed infrared frames. A peculiarity could be that corner detectors such as ORB and BRISK are amongst the least accurate, and above all they tend to be less accurate than their precursor FAST (which they should be an improvement of).

The question then becomes how much processing is required to achieve good results and if a spacecraft would be capable of carrying it out with the computational power available on board. [16] estimates 800 ms to be a reasonable amount of time to dedicate to real-time VN computations for a planetary landing mission. Fig. 6 shows how METs for most algorithms are quite high with respect to the reference. In fact, simulations are run on a high performing computer, but times are of the order of (if not higher than) hundreds of milliseconds. Hence when run in real-time these would probably not complete computations in an acceptable amount of time. The main exception here is STAR (and possibly AKAZE) which has METs in the order of tens of milliseconds. One of the main reasons for these excessive times appears to be the extremely high number of keypoints paired which slows down the overall process. Proof of this is for instance the fact that FAST, the most feature-finding algorithm, is known for having very low computational times but turns out to be the slowest both for matching and tracking in this analysis.

Generally, matching is more robust than tracking but also slower. Additionally, matching performance is much more dependent on the detector-descriptor couple with respect to tracking: in fact, depending on the frame subject the most accurate and reliable algorithm sequence typically has matching as a searcher but with features found and described by different algorithms depending on the specific case. Notice that tracking on average finds less pairs for each respective case. This is due to how tracking works: while for matching a new detection is performed, hence as many features as possible are paired for each FC, tracking does not require a new detection until all the previously found ones are lost, hence the number of paired features progressively lowers until a new detection is performed. This principle is one of the main reasons why the number of tracks is lower but also computational times are better.

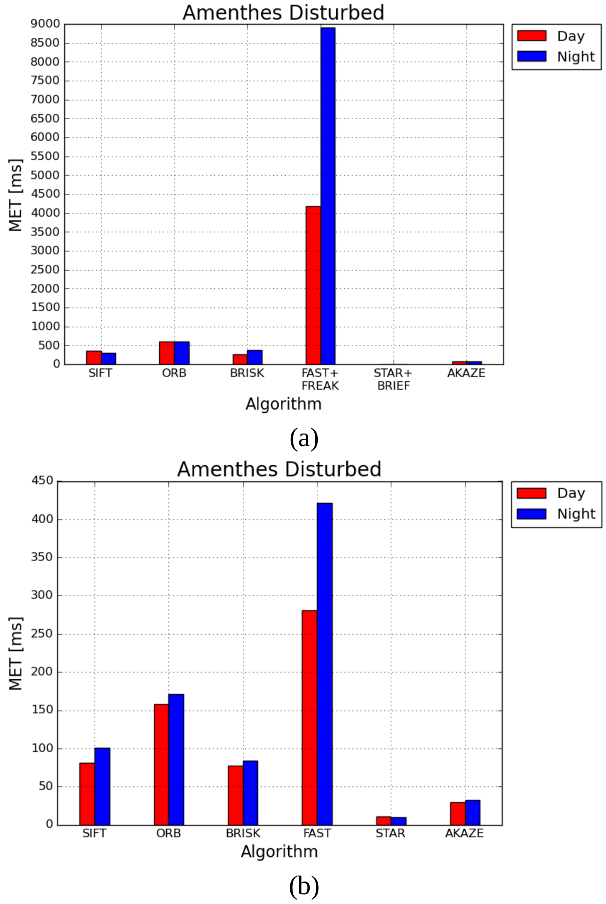
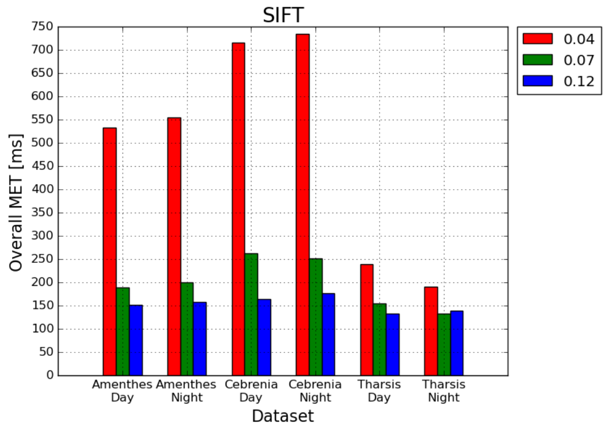


Fig. 6. MET for Amenthes Disturbed for all detectors when matching (a) and tracking (b).

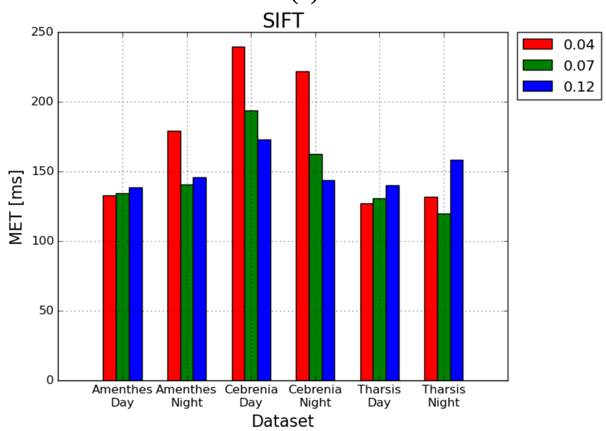
3.2 Threshold Tuning

The previous analysis highlighted promising but improvable results, especially for computational times, but also for robustness. Since an evident connection between the high number of found keypoints and the excessive MET is observed, finding a method to reduce the number of features to handle may be effective. Of course, this must not cause an excessive reduction in the number of features providing valuable navigation information and negatively affect overall robustness. An idea could be to manipulate the threshold of a detector for this purpose. Every feature detector has one or more reference values which divide the candidate features into rejected and outputted ones. This quantity typically reflects how distinguishable a feature is with respect to its surroundings which is closely related to its robustness. By setting the threshold of each detector to be more selective, less features should be found thus reducing computational times. Also, the features left should be more robust. On the other hand, a threshold making the detector excessively discriminatory would lead to greater chances of detection failure. One of the aims of this part of the study is to investigate the potential effectiveness of this strategy for IR VN, but, keeping in mind the wider context of the research, it does not make sense at this stage to conduct a fine threshold tuning for each feature detector. Hence, each algorithm sequence is analysed with three different threshold values empirically selected: the standard one (also used in the First Assessment), an upper limit making the process very selective and a final middle ground value. Apart from the detector threshold change, in this analysis algorithms are run just like in the previous part of the study.

As observable in Fig. 7, most algorithm sequences benefit from threshold tuning in terms of MET. This is true for SIFT, FAST, BRISK and ORB. STAR and AKAZE do not because of their default greater selectiveness, confirming that beneath a certain limit the number of features no longer is one of the main drivers for MET. This can also be observed looking at the trend in Fig. 7 which highlights how the time decrease achieved by increasing selectiveness is progressively lower. Additionally, tracking times benefit less from this strategy, coherently with observations from the previous analysis. Fig. 8 shows that accuracy, but also Reliability and other robustness indicators, do not change significantly, with some exceptions. Results conform to the previous analysis and to literature. A dedicated and more detailed threshold tuning for each detector based on the application would bring greater benefits. Overall, detector threshold tuning is worthwhile for Infrared VN because it allows to improve computational times without losing robustness, but it must be carried out carefully, setting it based on the specific algorithms and working conditions to avoid detection failure. Whether or not this is sufficient to make algorithms acceptable for real-time implementations is difficult to say by looking at the METs on a high performing computer, hence the Realistic Hardware Assessment described in section 3.4. An interesting observation that can be made is that while a lack of improvement on MET for STAR is not concerning, since these were already very low, that of AKAZE leads to its METs being comparable to those of SIFT. Since SIFT is notoriously known for being very robust but too slow for most real-time applications, this indicates that the same is likely true for AKAZE.

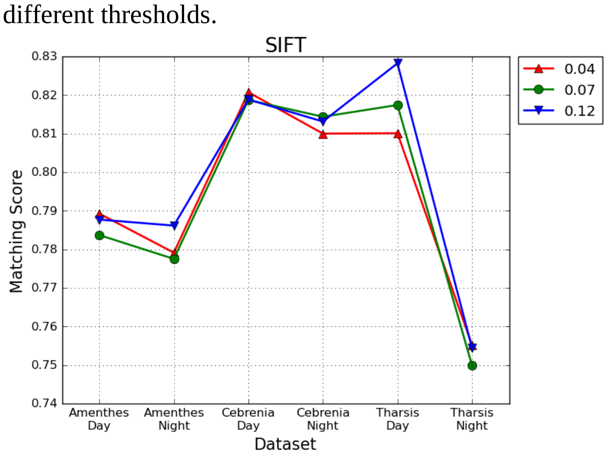


(a)

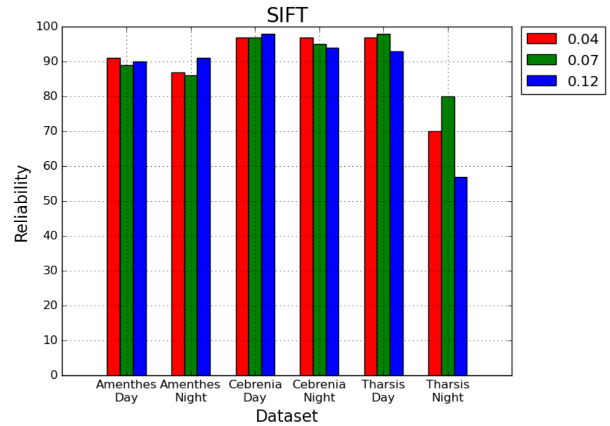


(b)

Fig. 7. SIFT matching (a) and tracking (b) MET for different thresholds.



(a)



(b)

Fig. 8. SIFT matching score (a) and reliability (b) for different thresholds.

Next, the best threshold for each detector is then chosen and each algorithm sequence is compared through a scoring system accounting for the most important performance parameters. For each parameter algorithm sequences are ranked from best to worst on the Disturbed datasets one by one. Then a score (the higher the better) is awarded to each algorithm sequence for each parameter. The scores are also added up in order to assess the overall performance of an algorithm sequence on a particular dataset. STAR matching turns out to be the best performer when working on Amenthes and Cebrenia. It is amongst the most robust and quickest. AKAZE and SIFT matching are good follow ups but are penalized by their high computational times. The only downside to STAR matching is its relatively low Persistence: this is not terrible, since it is around 10 for most frames, but there are better options. Hence, for an application where finding the same features across frames with a high level of consistency, such as a V-SLAM mission, is of great importance and a bit more time could be dedicated to computations, SIFT or AKAZE matching could be a better option. Another interesting observation related to STAR matching is that when moving to the Tharsis datasets, it becomes one of the worst performers. This is due to its high selectivity which makes it good on ground element rich surfaces where a lot of keypoints can be found, but bad for a surface like that of Tharsis. AKAZE tracking is by far the worst performer overall, indicating a high incompatibility between the detection algorithm and the method of feature search. FAST and ORB tracking are generally the best performers on Tharsis. It is important to highlight that this is a case of these algorithm sequences maintaining their performance levels across datasets much better than the algorithm sequences using matching. As a matter of fact, a substantial drop in performance can be witnessed for all matching sequences when going form Amenthes or

Cebrenia to Tharsis. On the other hand, most tracking sequences maintain similar levels of robustness and MET regardless of the framed region. Therefore, when ground element rich terrains are framed, matching sequences tend to perform better overall, while tracking sequences are better for flatter terrains. Regardless, tracking is generally quicker than matching on all datasets. It is important to underline how, coherently whit previous results, brightness conditions do not appear to be a performance driving factor just like the type of ground elements framed (for instance craters or canyons). It is the presence or absence of said elements that makes the difference, although, for the cases analysed in this study, it does not lead to impairment for all algorithm sequences.

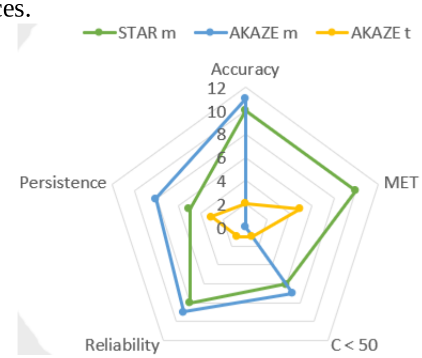


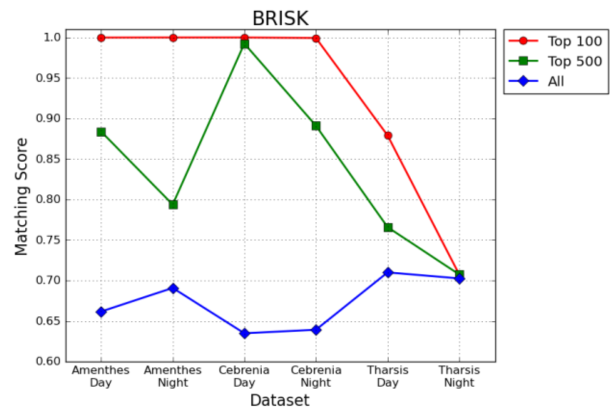
Fig. 9. Plot of the scoring of some algorithm sequences on Amenthes Day.

3.3 Feature Selection

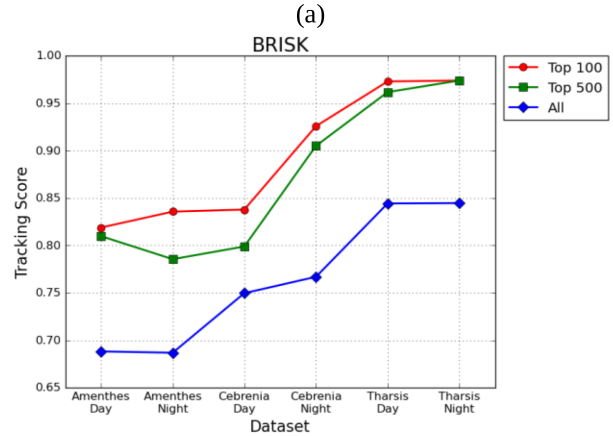
The strategy of tuning the detector's threshold has proven its effectiveness for time optimization. The method generally does not have a negative impact on robustness, but it does not improve it either. Thus, this part of the study aims at investigating the effectiveness of a simple strategy for improving Matching (or Tracking) Score, Reliability, etc. without burdening computational times.

The method applied is the following: for each FC of a dataset, all keypoint pairings are ordered from best to worst according to an error estimation value retrieved during matching or tracking computations. Then, only the top 100 or 500 are considered for calculating the performance parameters, which is the equivalent of only passing these top parings on to the navigation filter in a real system. Specifically, for tracking the L1 distance between patches around the original and the moved point divided by the number of pixels in a window is used [20]. For matching, descriptor distance indicators are used. In particular, the Hamming distance for binary descriptors (which are all except SIFT) and the L2 norm otherwise. The choice of 100 and 500 as the limits for the considered pairings, comes from balancing out literature indications with the number of features found in this study.

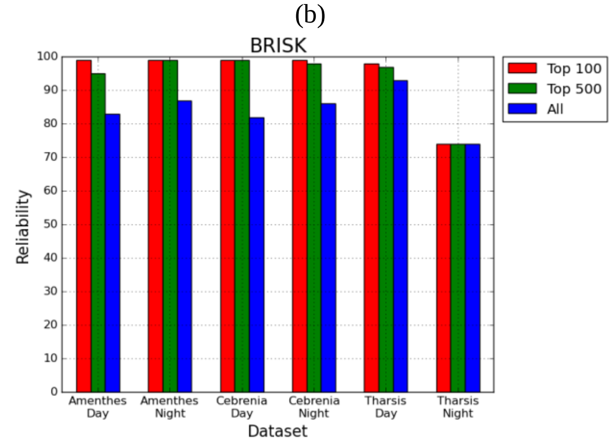
Simulations are run for all algorithm sequences (with their best detector threshold) on the Disturbed datasets and performances are compared exactly as for the Threshold Tuning in section 3.2.



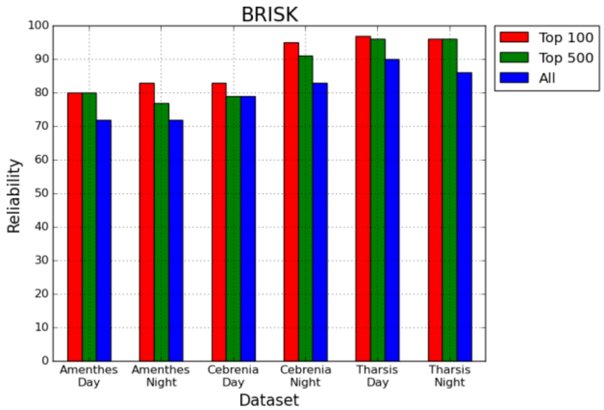
(a)



(b)



(c)



(d)

Fig. 10. BRISK MS (a), TS (b), matching R (c) and tracking R (d) for the two feature limits compared to the previous unlimited case.

All in all, results show that the strategy makes the process more robust for all algorithms. As highlighted by the reference example in Fig. 10, Accuracy and Reliability are substantially improved for most algorithms on most datasets. The main exception is for Tharsis Night where the number of found features is in most cases lower than 100, hence the feature limitation strategy does not make any relevant difference. Computational times do not show any significant variation with respect to the previous parts of the study (although there are some discrepancies ascribable to computational fluctuations), so their acceptability remains to be confirmed.

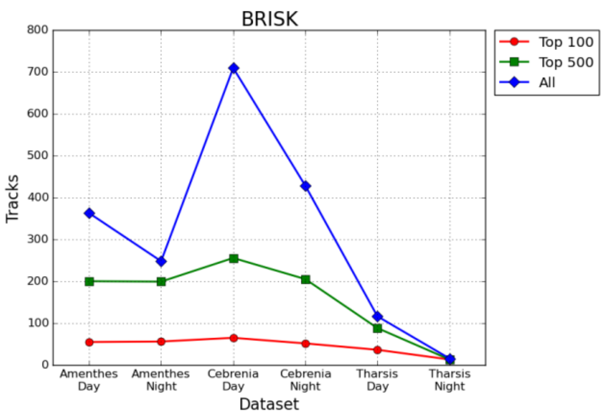
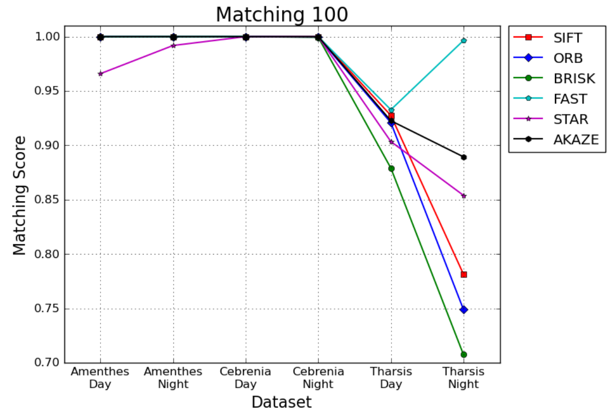
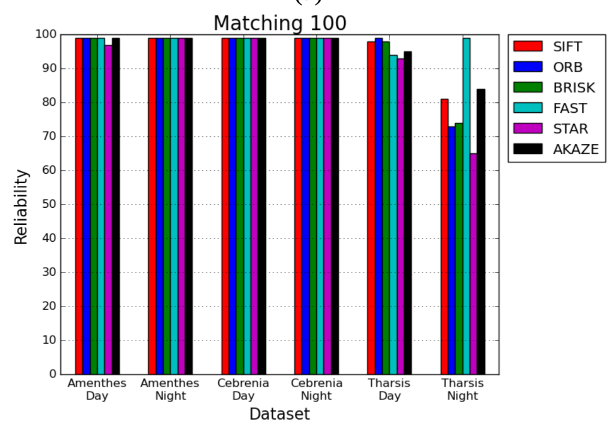


Fig. 11. BRISK Tracks for the two feature limits compared to the previous unlimited case.

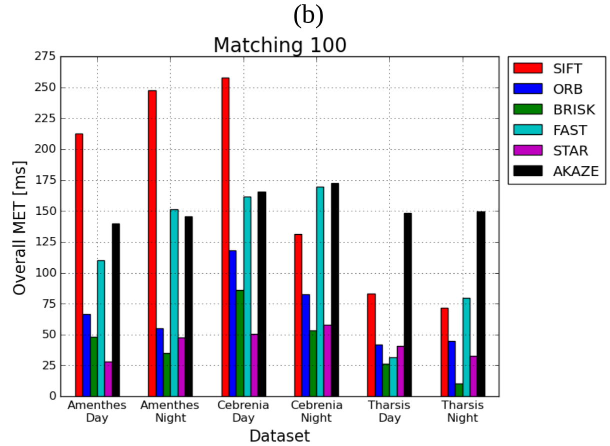
Also, the choice of the number of features to limit strongly affects the benefits depending on the specific case and algorithms involved. This is made clear by looking at the tracking algorithms which generally end up having to work with an insufficient number of features when the limit is set to 100, see Fig. 11. Because of how the performance enhancing strategy works, Minimum amount of Information, Zero Information and Persistence do not change in any case.



(a)



(b)

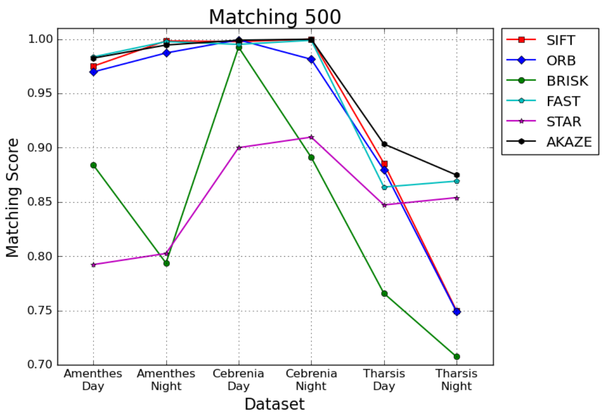


(c)

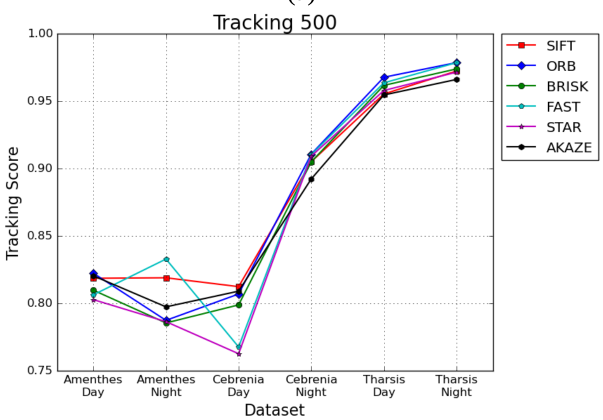
Fig. 12. Comparison of MS (a), matching R (b) and matching MET (c) for the 100 limit.

Performance comparison in this case shows much less clear and more variable best and worst performers. This is due to the substantial increase in robustness for all algorithm sequences. The trend is easy to observe looking at MS and R for Amenthes or Cebrenia in Fig.

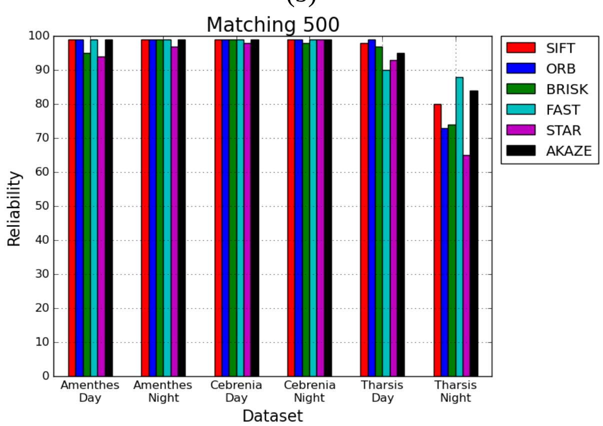
12: no Accuracy drops below 0.95 and, excluding STAR, they are all close to 1.0. Therefore, comparing the scores creates a MS leaderboard in which the differences are minimal. A consequence of this is that MET becomes a much more determining factor for evaluating overall performance.



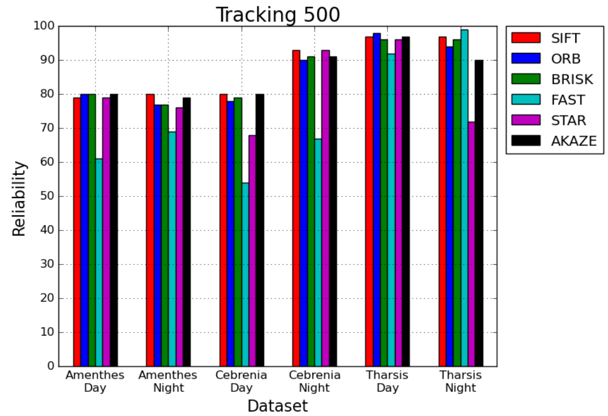
(a)



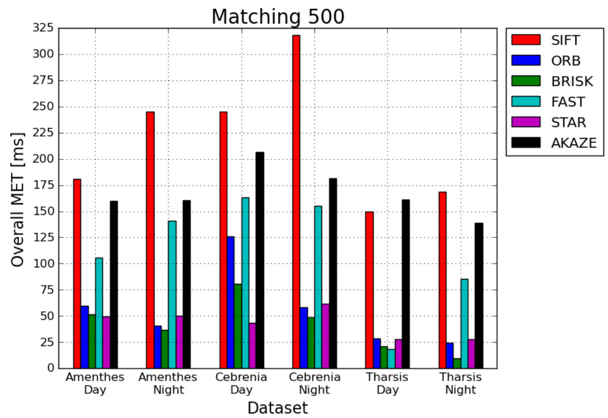
(b)



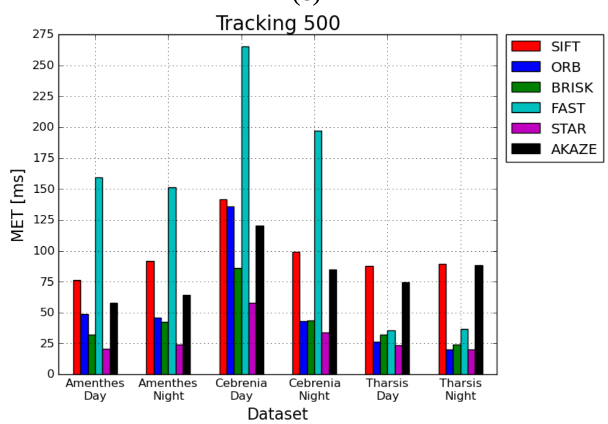
(c)



(d)



(e)



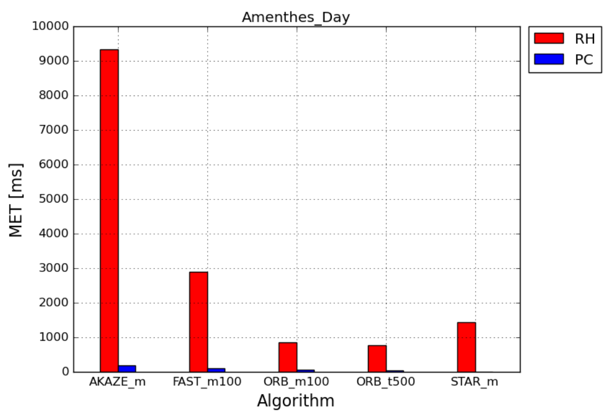
(f)

Fig. 13. Comparison of MS (a), TS (b), matching (c) and tracking (d) R, matching (e) and tracking (f) MET for the 500 limit.

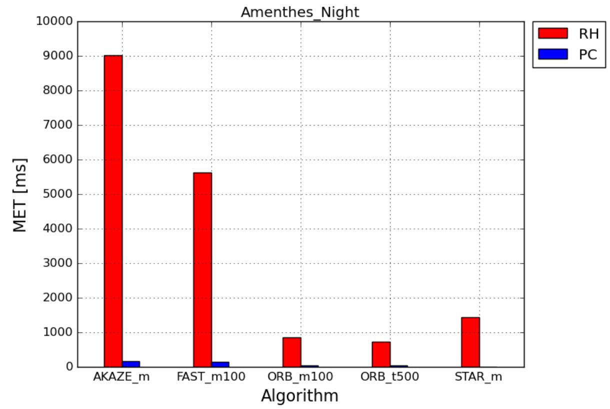
Scoreboards show that algorithms with corner detectors are the most favoured by feature limitation. These achieve the most significant improvements overall, with performances much more coherent with those found in literature [18,19]. ORB and BRISK performances are now similar if not better than those of FAST, proving that it was the inclusion of all found features to lead to inconsistencies rather than the switch of input from visual to IR frames. Overall, the significant distinction between matching and tracking performance, as well as that between corner and filter-based detectors, found in the previous study depending on the type of surface framed, are less marked. This means that, thanks to this post-processing strategy, more algorithm sequences show good potential for working in an Infrared VN system for planetary landing. Because of the overall robustness boost previously discussed, it is not worth highlighting best and worst performing algorithms for each case since these vary quite a bit due to minimal differences. Nonetheless, it is worth mentioning that STAR matching no longer is amongst the best performers (although its performances do remain good) while AKAZE tracking does remain one of the worst algorithm sequences overall. The algorithm sequences showing the most consistently good performances across all datasets appear to be ORB and BRISK tracking and potentially also matching.

3.4 Realistic Hardware

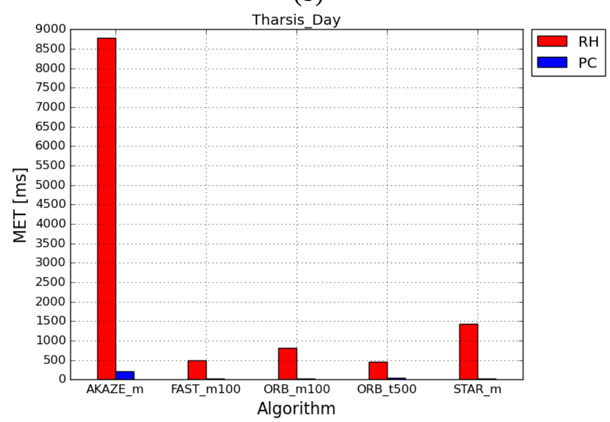
A shortcoming of the previous parts of the analysis is that algorithms are ran with a computational power much higher than that available in space. Therefore, no realistic MET indication can be retrieved. This section aims at providing an initial indication of CV algorithms' computational times on infrared images by running them on a BeagleBone Black board [8].



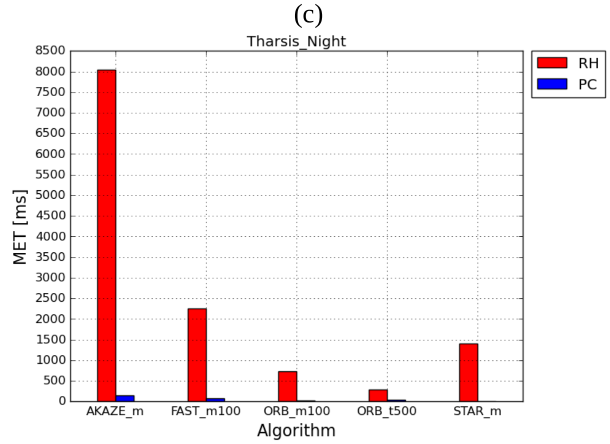
(a)



(b)



(c)



(d)

Fig. 14. Comparison of Portable Computer (PC) vs Realistic Hardware (RH) METs for Amenthes Day (a) and Night (b) and for Tharsis Day (c) and Night (d).

Fig. 14 reports the results found for Amenthes, which is also representative of Cebrenia, and those of Tharsis. For conciseness, only the most promising algorithms overall have been studied for each Disturbed dataset. AKAZE matching is the only exception since previous outputs led to believe that it would be too slow for real-time planetary landing applications and counterproof is desirable. On Amenthes and Cebrenia none of the algorithms grant acceptable MET for the reference application [16], since no MET is lower than 800 ms. For Tharsis there are some instances where METs are lower that the reference limit; this area is, however, the one on which algorithms struggle the most in terms of robustness. It is also important to keep in mind that the reported times are averages while the reference time limit should not be exceeded for any FC of a dataset. So, these results represent very useful preliminary indications which should not, however, be taken as absolute and unchangeable references.

The outcome is expected since time-optimizing strategies are always adopted in Vision-based Navigation, but none were implemented in this analysis. The fact that regardless there are some instances where average times come close to and others which satisfy the reference limit nonetheless, is promising. It is also important to keep in mind that, depending on the application, the time available to complete computations may differ quite a bit. There are many strategies available from literature that can help improve computational times. These are likely to be effective for IR VN since, from a computational perspective, using visual or IR frames does not make a difference. Also, nowadays there are many space-approved units designed to optimize VN computations which would be effective for this application.

The quickest algorithm sequence overall is ORB paired with LK tracking, closely followed by the same detector but with BF matching as the searcher. These are the algorithm sequences which occasionally respect the 800 *ms* threshold, hence being the most promising for real-time implementation. The fact that FAST with FREAK matching goes from being one of the slowest algorithms on Amenthes and Cebrenia, where it has a lot of keypoints to handle, to one of the quickest on Tharsis, where the number of features decreases significantly is counterproof of the importance of the number of keypoints to handle for the computational time of some algorithms. On the other hand, the fact that AKAZE and STAR with BRIEF matching have consistent METs across all datasets confirms that the influence of the number of features to handle is true up to a certain limit. Also, the slowness of AKAZE confirms the suspicion of it not being suited for a real-time application with time scales in the order of those required by planetary landing. STAR displays an interesting result since it increases computational times of two orders of magnitude, going from being the quickest algorithm out of all to never satisfying the 800 *ms* requirement. One reason could be associated to the fact that STAR is a filter-based detector: in fact, the other filter-based detector, AKAZE, executed on the board slows down with a similar entity of two orders of magnitude with respect to when it is run on the pc. Hence, a limited computational capacity likely affects more the processing of a detector exploiting filters rather than corners, which is what ORB and FAST do. A second reason could be associated to the multiple controls that STAR carries out for refining the definition of its keypoints. These are more and more complex than those of ORB for instance. Once again, the limitation of computational power likely slows down the implementation of these checks, weighting significantly on the overall time for feature detection.

The outputs of this part of the study prove that there is a lot of work to be done before achieving a feasible AGNC system exploiting Infrared Vision-based Navigation for planetary landing. However, results also promising, indicating that, by exploiting the many points in common between visual and IR VN, a sufficiently robust and quick real time implementation could be feasible.

4. Discussion

The results of this study demonstrate the potential feasibility of using IR VN for the task of landing on a natural surface. In particular, the work has highlighted how the use of infrared frames allows CV algorithms to perform well in poor brightness conditions. Additionally, the study has highlighted how motion disturbance effects do have an impact on the quality of the results but do not make the application unfeasible.

Differences in best and worst performing algorithms sequences depending on the working conditions are coherent with visual image literature [18,19], and the fact that regardless there are some algorithm sequences finding reasonably abundant and robust pairings in relatively low computational times is promising. This outcome is compatible with the relatively few studies on purely infrared VN found in literature such as [21] where the use of IR sensors allowed to achieve similar performances when navigating a helicopter in the night with respect to its navigation during the day with visual VN. Nonetheless, the passage from earth-bound applications to the field of space is not trivial. This is one of the first studies approaching the topic of IR VN applied to the space field and the topic of planetary landing in particular. Hence, the outcomes presented in this paper are not to be underestimated.

The many points in common found between visual and infrared VN results are another important outcome of this work. These can be exploited to reduce times taken in the development of a full AGNC system exploiting IR VN for planetary landing. In fact, the strategies and methods that have been found to work best for the visual case are likely going to work well for the infrared one too. Thus, giving an encouraging starting point for future developments.

Finally, it is important to remark on the limited conditions that could be addressed in this study which calls for future research analyzing how effects such as scale change and out-of-plane rotation affect IR VN performance. The studies should also perpetuate in greater detail the investigation on computational times, although the fact that Vision-based Navigation has already been successfully used for landing on natural surfaces suggests that this should not be a major issue.

5. Conclusions

The work here presented is a preliminary feasibility assessment for the use of Infrared Vision-based Navigation at the scope of performing AGNC for planetary landing. At first, numerical simulations have been run on IR frames of the surface of Mars to evaluate CV algorithm performance and the influence of effects such as motion disturbance and absence of sunlight. Secondly, similar simulations have been implemented aiming to assess the effectiveness of two simple performance enhancing strategies as well as comparing various algorithm sequences to figure out which show the most promising performances overall. Finally, the most interesting algorithm sequences are further tested with numerical simulations run on a BeagleBone Black board to retrieve more realistic time indications.

In this preliminary assessment of the feasibility of an Infrared Vision-based Navigation system for planetary landing, no result has indicated the unfeasibility of the implementation. Most algorithms on most datasets are quite robust although performances, especially computational times, are (as of now) not good enough for real missions. One of the most important outcomes of the study is that illumination conditions are not a performance driving factor for this technology, indicating the feasibility of carrying out EDL at night. Also, the fact that many parallels with visual image literature, including the effectiveness of efficiency enhancing strategies, have been found is promising for accelerating the development of this technology. All the promising outcomes of this study are a clear indication that further research on the topic of IR VN is undoubtedly worth pursuing.

References

[1] M. Maiomone, et al. Computer Vison on Mars. *International Journal of Computer Vision*, 75(1):67–92, 2007. doi: https://doi.org/ 10.1007/s11263-007-0046-z.

[2] A. E. Johnson, et al. Implementation of a Map Relative Localization System for Planetary Landing. *Journal of Guidance, Control and Dynamincs*, 46(4), 2023. doi: https://doi.org/10.2514/1. G006780.

[3] R. R. Sostaric, et al. The SPLICE Project: Safe and Precise Landing Technology Development and Testing. *In AIAA Scitech 2021 Forum*, 2021. doi: http://dx.doi.org/10.2514/6.2021-0256.

[4] B. Maass, et al. Crater Navigation System for Autonomous Precision Landing on the Moon. *Journal of Guidance, Navigation and Dynamics*, 43(8),2020. doi: https://doi.org/ 10.2514/1.G004850.

[5] W. Guangfei, et al. Illumination conditions near the moon's south pole: Implication for a concept design of china's chang'e7 lunar polar exploration. *Acta Astronautica*, Vol. 208, 74–81, 2023. doi: https://doi.org/10.1016/j.actaastro.2023.03.022.

[6] B. R. Van Manen, et al. Thermal Simultaneous Localization and Mapping, 2023. https://encyclopedia.pub/entry/49458. Last visited: 2/05/2024.

[7] HP Development Company. HP ProBook 450 G5 Notebook PC Product Specifications, 2023. https://support.hp.com/gb-en/document/c05682645. Last visited: 14/12/2023.

[8] BeagleBoard.org Foundation. BeagleBone Black Overview, 2023. https://docs. beagleboard.org/latest/boards/beaglebone/black/ch04.htmlv. Last visited: 14/12/2023.

[9] OpenCV: Open Source Computer Vision Library, 2023. https://github.com/opencv/opencv. Last visited: 11/12/2023.

[10] R. Hartley and A. Zisserman. Multiple View Geometry in Computer Vision. *Cambidge University Press*, The Edinburgh Building, Cambridge, CB2 2RU, UK, 2003.

[11] K. Yousif, et al. An Overview to Visual Odometry and Visual SLAM: Applications to Mobile Robotics. *Intel Ind Syst*, 1:289–311, 2015. doi: https://doi.org/10.1007/s40903-015-0032-7.

[12] OpenCV. Feature Matching. https://docs.opencv.org/3.4/dc/dc3/tutorial_py_matcher.html. Last visited: 22/01/2024.

[13] B. D. Lucas and T. Kanade. An Iterative Image Registration Technique with an Application to Stereo Vision. In *Proceedings of the 7th International Joint Conference on Artificial Intelligence*, pages 674–679, 1981.

[14] J. Bouguet. Pyramidal Implementation of the Lucas Kanade Feature Tracker Description of the algorithm. http://robots.stanford.edu/cs223b04/ algo_tracking.pdf.

[15] P. R. Christensen, et al. The Thermal Emission Imaging System (THEMIS) for The Mars 2001 Odyssey Mission. *Russell, C.T. (eds) 2001 Mars Odyssey*, pages 85–130, 2002. doi: https://doi.org/10.1007/978-0-306-48600-5_3.

[16] S. Silvestrini, et al. Optical navigation for Lunar landing based on Convolutional Neural Network crater detector. *Aerospace Science and Technology*, 123:107503, 2022. doi: https://doi. org/10.1016/j.ast.2022.107503.

[17] W. Hangong, et al. Tianwen-1 Mars entry vehicle trajectory and atmosphere reconstruction preliminary analysis. *Astrodynamics*, 6(1):81–91, 2022.

[18] D. Mukherjee, et al. A comparative experimental study of image feature detectors and descriptors. *Machine Vision and Applications*, 26:443–466, 2015. doi: https://doi.org/10.1007/s00138-015-0679-9.

[19] S. A. K. Tareen and Z. Saleem. A Comparative Analysis of SIFT, SURF, KAZE, AKAZE, ORB, and BRISK. In *2018 International Conference on Computing, Mathematics and Engineering Technologies (iCoMET)*, pages 1–10, 2018. doi: https://doi.org/10.1109/ICOMET.2018.8346440.

[20] OpenCV. Object Tracking. https://docs.opencv.org/3.4/dc/d6b/group__video__track.html#ga473e4b886d0bcc6b65831eb88ed93323. Last visited: 9/01/2024.

[21] J. Delaune, et al. Thermal-Inertial Odometry for Autonomous Flight Throughout the Night. In *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1122–1128, 2019. doi: https://doi.org/10.1109/IROS40897.2019.8968238.