Enabling active perception through data quality assessment: a visual odometry case

Andrea de Maio, Simon Lacroix

ENABLING ACTIVE PERCEPTION THROUGH DATA QUALITY ASSESSMENT: A VISUAL ODOMETRY USE CASE

Andrea De Maio and Simon Lacroix

LAAS-CNRS, Université de Toulouse, CNRS
7, Avenue du Colonel Roche, 31031 Toulouse, France,
E-mail: {andrea.de-maio,simon.lacroix}@laas.fr

ABSTRACT

The state of the art of perception processes for the autonomy of robots is constantly improving, yet these processes remain mostly pre-configured at the robot design phase. This prevents their adaption to the context at hand, which is all the more needed for long life systems that encounter a large variety of situations. This paper presents work on the modelling of perception processes, exhibiting the need to assess their quality, so as to be able to actively control them. We instantiate the visual odometry case, a crucial functionality for planetary rovers, and define dedicated data quality assessment functions for the elementary processes composing it. These functions are used to monitor the processes, defining control points onto which explore different parameter configurations that better adapt to the context the robot is facing. Preliminary tests are performed using a planetary analogue data set to show the potential of this approach.

Key words: active perception; perception quality assessment; perception optimisation; visual odometry.

1. INTRODUCTION

The sensing technologies for robotics are ever improving, along with the state of the art in data processing and data fusion, that now offers a broad choice of solutions for robotics perception. This will allow to endow future generation of space robotics systems with a much richer perception layer, which will maximise the throughput of exploration missions, *e.g.* by yielding the possibility of autonomous long traverses or autonomous science.
However, such broad perception capabilities come with a large amount of configuration parameters. So far, roboticists configured perception processes to find the best expected trade-off between genericity, applicability to the context in which the robots operate, and constraints on the sensing, communication and computing resources. In other words, data fusion and perception processes of space robots are fixed by design, tailored to the task they are needed for, and cannot be optimised online with respect the conditions under which the robots are operating and the objectives they are trying to achieve. This prevents the adaptation of the perception activities to the context and the task at hand, which may yield poor performances in case of unexpected scenarios, and in the worst cases the inability to provide useful information.
The augmentation of the complexity of perception processes raises a need to *actively* control them, by deciding which data to acquire and how to process them. This follows the *active vision* paradigm [2], which aims at optimising the throughput of perception, that is the relevance and quality of the information provided to the clients of the perception processes. The clients that exploit the two main perception products, environment models and robot localisation, are numerous and of varied nature, from motion control to decisional and planning processes. To a large extent, the quality of the perception products define the efficiency and adaptivity of the robots, and there is a strong interest in optimising this quality.
In robotics, development of active perception schemes have always targeted to specific given tasks: there is a lack of a system abstracting from the type of sensor or from the nature of the task, meant to define a generic, principled approach to active perception. Our research objective is to propose a formal and operational framework to allow the control of robotics perception activities. This entails a formal modelling of the perception tasks, and the definition of optimisation tools that allow to configure perception activities, as well as to control them in real time. The framework aims at having autonomous systems being able to adapt to a larger variety of contexts and situations, without the need of a human in the loop to manually reconfigure perception processes.

Outline. This paper sketches our work in this direction, and focuses on the case of Visual Odometry (VO), a perception process that has shown its importance for planetary rovers. Section 2 first drafts the way we foresee the modelling of the perception processes. The remaining of the paper is dedicated to the specific case of VO: section 3 depicts the models of the underlying processes, section 4 defines the associated metrics that allow to qualify, and section 5 depicts and discusses preliminary results.

2. MODELLING PERCEPTION PROCESSES

The quality of information produced by the perception processes depends on numerous factors. Some are controllable, like the selection of the algorithms, their configuration and composition, or the resources devoted to their execution. Others are not controllable: the nature of the perceived scenes and the environmental conditions have a strong impact on the output quality of the perception processes.
To maximise the output quality of the perception processes, one must intervene on the controllable factors, while tracking and possibly adapting to the uncontrollable ones. For this purpose, one must assess the influence of the controllable factors on the process output quality: this is done by defining functions that disclose this influence, as well as the influence of the non controllable factors, *i.e.* by defining *models* of the perception processes.

2.1. Structuring Perception Processes: Nodes and Compounds

Perception encompasses a variety of processes, ranging from simple data filtering to complex optimisation schemes for state estimation, that must be assembled (composed) in order to fulfill a given functionality, *i.e.* deliver a product such as an environment model or a position estimate. This structure is applicable to all the perception functionalities a given robot must be endowed with. In the context of autonomous navigation, Visual Odometry, SLAM and the generation of a Digital Elevation Map (DEM) are perception functionalities which are composed of several signal processing functions, organised together to return their associated products. As in [7], we adopt the notion of *nodes* and *compounds*, where a node represents an atomic perception function performing elementary operations, and a compound is a composition of nodes assembled to deliver a specific data product. A broader view of this concept is presented in [4], along with a sketch of a taxonomy of nodes and compounds defined by their input and output data types.
The benefits of structuring perception activities into nodes and compounds are the ones of any component based software architecture: *e.g.* reconfigurability, ease of development and maintenance, separation of concerns, reusability, openness, and of course composability. In the context of active perception, *controllability* is one of the most important concern. In the remainder of this section, we present a generic formal model of perception nodes that specifies the influence of their control.

2.2. A Model for Controlling Perception Nodes

Each perception node is characterised by a set of inputs $\boldsymbol{i}$ and a set of outputs $\boldsymbol{o}$ both represented by specific data types (for simple processes, theses sets are singletons). The combination of these input and output types identify the *type* of the process, which can be achieved using different implementations (for instance numerous approaches extract point features from an image – note this may yield slight variations of the definition of the associated descriptor, and hence of the output type: such considerations pertain to the definition of taxonomies of processes and data types, with inheritance mechanisms, which will not be developed here).

A set of input parameters $\boldsymbol{u} = \{u_1, \ldots, u_n\}$, generally linked to the considered implementation, is specified and represents the controllable means to intervene on the process, either at configuration stage or during its execution. Finally a set of data quality assessment functions $\boldsymbol{j} = \{j_1, \ldots, j_m\}$ are defined: they assess the quality of the output as a function of the inputs and parameters. These quality variables are of various nature: they can come along the production of the process output (such as variance for a state estimation process), or are more or less explicitly encoded within the process output (such as the proportion of outliers produced by a data association process). The problem of optimising a perception node is simply stated as finding the optimal set of parameters $\boldsymbol{u}^*$:

$$\boldsymbol{u}^* = \underset{\boldsymbol{u}}{\operatorname{argmin}}\, j(\boldsymbol{u}, \boldsymbol{i}) : j(\boldsymbol{u}^*, \boldsymbol{i}) \leq j(\boldsymbol{u}, \boldsymbol{i})\ \forall \boldsymbol{u} \qquad (1)$$

Other exogenous concerns condition the behavior of the perception nodes, and hence the quality of their output. These are grouped under the term "context", and include for instance light conditions, terrain and texturing levels. The context gathers a series of non controllable, and sometimes even non observable parameters. Some context information can indeed be directly observable, *e.g.* by specific sensors or given information, some are implicitly encoded in the input parameters $\boldsymbol{i}$, and hence not necessarily observable, and finally some are not known at all. Denoting the $\boldsymbol{z} = (z_1, \ldots, z_p)$ the set of uncontrollable yet observable information, which include the inputs $\boldsymbol{i}$, Eq. 1 is rewritten as:

$$\boldsymbol{u}^* = \underset{\boldsymbol{u}}{\operatorname{argmin}}\, j(\boldsymbol{u}, \boldsymbol{z}) : j(\boldsymbol{u}^*, \boldsymbol{z}) \leq j(\boldsymbol{u}, \boldsymbol{z})\ \forall \boldsymbol{u} \qquad (2)$$

Lastly, a set of costs $\boldsymbol{c}$ can be associated to the perception node: it encompasses the various resources consumption (time, memory...), and the optimisation problem then comes to maximising quality to cost ratios.

Finally, we can formalise a model for a perception node $\mathcal{N}$ as:

$$\mathcal{N} = (\boldsymbol{i}, \boldsymbol{o}, \boldsymbol{u}, \boldsymbol{z}, \boldsymbol{j}) \qquad (3)$$

2.3. From Nodes to Compounds

The integration and interaction of perception nodes defines a perception compound $\mathcal{C}$, which produces the final data products to be delivered to the client processes. Most of the times, compounds are defined as a pipeline assembly of nodes, but some nodes can be asynchronously

invoked, and some feedback sequences can be defined. Here the component based model is very helpful.

Optimising a compound $\mathcal{C} = \bigcup_{i=0}^{k} \mathcal{N}_i$ of k nodes comes to find the set of controls $\boldsymbol{U_{0,k}} = \{\boldsymbol{u_0}, \ldots, \boldsymbol{u_k}\}$ for each of the k implied nodes so as to optimise the final product, that is the output of the last involved node $\mathcal{N}_k$:

$$\boldsymbol{U^*}_{0,k} = \underset{\boldsymbol{U_{0,k}}}{\operatorname{argmin}}\, j(\boldsymbol{u_k}, \boldsymbol{z}) \tag{4}$$

Finally, more parameters come into play at compound level. For instance, the frequency at which the whole compound runs can be tuned: this affects not only the resource usages but also the accuracy of the data produced by some nodes. Moreover, a compound can be assembled with different compositions of nodes. Some nodes can be optional and some others can be arranged in different orders. For many nodes it is also common to have different implementations performing the same task in different ways. All these cases deal with the *topology* of the compound, offering another layer of control.

In the generic case, given the large parameter space (even if it is reduced by the fact that all parameters are not independent) and the various interleaved semantics between quality measures j and process inputs and outputs, such an optimisation problem is intractable. The next section will illustrate the difficulty of the problem for the case of Visual Odometry, a rather simple pipeline compound.

3. MODELLING VISUAL ODOMETRY

Visual odometry is one of the first localization means developed for mobile robotics that exploited vision. Since pioneering work that dates back to the 90's, VO has featured a large number of approaches and methodologies: sparse vs. dense, using monocular vision or stereovision, and many more [12, 1]. We focus here on the most classic instance of VO, which derives the motion between two positions at each of which a pair of stereovision images is acquired, by matching point features extracted in the images (Fig. 1). This instance of VO is well adapted to the limited resources typically available on board planetary exploration rovers, and has been extensively used on the Mars Exploration Rovers [9].

3.1. Involved Processes

This VO scheme is a compound which integrates four nodes: (1) a keypoint extractor, that takes as input an image acquired by a camera and outputs a set of keypoints; (2) a data association algorithm that matches two sets of keypoints extracted from two different images; (3) triangulation, that associates 3D coordinates of keypoints matched between two stereoscopic images, and (4) a motion estimator that computes the relative motion between

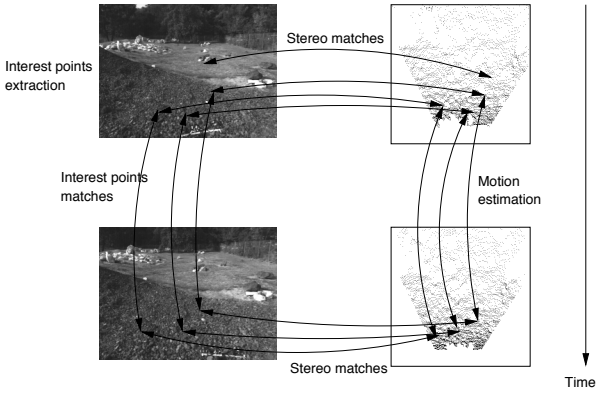


Figure 1: Principle of feature-based stereo VO

two stereoscopic image pairs, using the associated keypoints matches and 3D coordinates. Fig. 2 summarises how these nodes are pipelined.

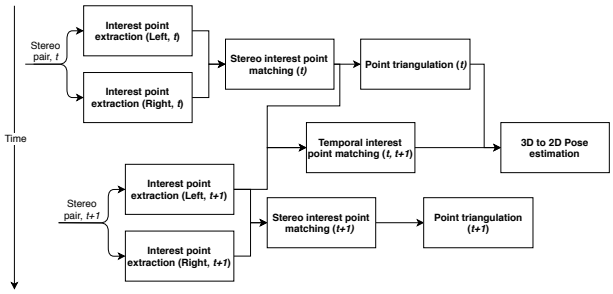


Figure 2: Feature-based visual odometry with 3D to 2D motion estimation workflow

The final output of the overall process is a 3D transformation between the two times (or positions) $t-1$ and t at which the stereoscopic images were collected:

$${}^{t-1}\boldsymbol{T}_t = \begin{bmatrix} {}^{t-1}\boldsymbol{R}_t & {}^{t-1}\boldsymbol{p}_{t-1,t} \\ \boldsymbol{0} & 1 \end{bmatrix} \tag{5}$$

where ${}^{t-1}\boldsymbol{R}_t$ and ${}^{t-1}\boldsymbol{p}_{t-1,t}$ respectively represents the orientation and translation of the stereo reference frame at time t w.r.t to the same frame at $t-1$. The rover localisation is then estimated by the composition of the motion estimation matrices in between each acquired image pairs:

$${}^{O}\boldsymbol{T}_t = {}^{O}\boldsymbol{T}_1\, {}^{1}\boldsymbol{T}_2 \ldots {}^{t-1}\boldsymbol{T}_t \tag{6}$$

where O is the origin reference frame, generally defined at the beginning of a mission/traverse in a specified site.

3.2. Details of the Involved Process

We briefly depict here the models of the four perception nodes. The most important part of the models for their

controllability being the expression of the quality assessment functions $j(\boldsymbol{u}, \boldsymbol{z})$, a specific section is devoted to their derivation.

Interest point extraction. We have selected the ORB detector for its speed in terms of computation time while maintaining acceptable scale and rotation invariance [11]. A given number of features is extracted from the input image, if more features than requested can be extracted the best n are returned based on a quality score (the "response" of the detector). Following the node model notations (Eq. 3), we have:

- $\boldsymbol{i}$ is an image,
- $\boldsymbol{o}$ is the set of detected keypoints and associated descriptors,
- $\boldsymbol{u}$ encompasses the target number of extracted features n, the pyramid levels, the scaling factor and a detection threshold.

Interest point matching. By comparing keypoints descriptors, the matcher associates keypoints between two images:

- $\boldsymbol{i}$ is two sets of keypoints (with their associated descriptors) extracted in two images,
- $\boldsymbol{o}$ is a set of matched (paired) keypoints,
- $\boldsymbol{u}$ includes the number of k-best matches to return for each keypoint, a cross-check flag request and a criterion to validate matches (e.g. descriptors distance based, best x$\%$).

Note that when applied to rectified stereo images, the matcher search for each feature is narrowed to the same vertical coordinate (*i.e.* the corresponding epipolar line) on the other image $\pm$ 1-2 pixels as an error offset.

3D points triangulation. The data association performed by the matcher on two stereo images allows to estimate their 3D coordinates through a simple triangulation process that does not exhibit any control parameter:

- $\boldsymbol{i}$ is a set of keypoint matches established from a pair of stereo images, and the stereoscopic bench calibration matrices,
- $\boldsymbol{o}$ is a set of 3D points,
- $\boldsymbol{u}$ is an empty set.

Motion estimation. Two approaches for this process are possible: 3D to 3D and 3D to 2D [12]. The first approach finds the roto-translation aligning two sets of corresponding points in $\mathbb{R}^3$, resorting to a least-squares method using singular value decomposition [13]. The second approach minimises the image reprojection error and is now more commonly applied ([10] showed that the 3D to 2D motion estimation performs better than the 3D to 3D due to the large depth error carried by the triangulation process).

We use the latter approach, along with a RANSAC scheme that allows to discard wrong matches possibly produced by the matcher:

- $\boldsymbol{i}$ is a set of 3D points observed at time $t-1$, their corresponding image points in one camera frame at time t and the camera intrinsic calibration matrix,
- $\boldsymbol{o}$ is a transformation matrix as in Eq. 5,
- $\boldsymbol{u}$ is a set of parameters for the RANSAC scheme, and an optional motion guess (e.g. based on a motion model or on wheel odometry).

3.3. Context

Note that the context $\boldsymbol{z}$ has not been made explicit for any of the four nodes that define our instance of VO. Indeed, the context here mostly pertains to the environment (terrain visual appearance, illumination), which directly (and strongly) influences the first node of the pipeline, that is keypoints extraction. All the following processes are then consequently affected via the output of this first node (number of extracted keypoints and histogram of the associated responses).

3.4. Other Parameters

As introduced in Sec. 2.3, other parameters pertain to the overall compound. For visual odometry, deciding the process frequency not only accounts for different resource consumption, but also influences the precision of the estimation. Assuming the robot velocity is defined, the problem is dual with controlling the spatial frequency and linked to the *keyframe selection* problem. A high frequency reduces the likelihood of errors in tracking features but is not always achievable due to operational constraints in terms of resources, especially on space hardware. Furthermore, due to the presence of noise in the images and the keypoint extraction process, a too frequent estimation yields a higher drift compared to a slower estimation that is still able to correctly match features.

4. DATA QUALITY ASSESSMENT

For control purposes, it is essential that each node is associated with one or more data quality assessment functions $j(\boldsymbol{z}, \boldsymbol{u})$ that express the influence of the control parameters $\boldsymbol{u}$ on the quality of the outputs $\boldsymbol{o}$ as a function of the inputs and context information. Depending on the considered perception node, this step is not always straightforward. For instance, there is no generic quality metrics applicable to any type of data. While it is typical to resort to uncertainty measures (as covariance of the motion estimates), this is not always possible for many reasons, from the lack of uncertainty in the problem modelling to the impossibility to measure covariance for the input data. Furthermore, the explicit composition of the node data quality assessment functions is hardly feasible, given the variety of data and quality metric types.

Broadly speaking, there are two options to define such data quality assessment functions:

- Model-based approaches, from close-form derivations to rules-of-thumb.
- Data-based approaches, which calls for machine learning approaches where a model representing the characteristics of the data is produced for different contexts and both successful and unsuccessful cases.

Below we present some tentative data quality assessment functions for the nodes that define our VO scheme. They are preliminary, and show the difficulties of finding good measures to assess the quality of a perception process.

Interest point extraction. In order to assess the quality of extracted visual features, we define a figure of merit based on the keypoint response. The response is generally higher in strong features, which are in turn easier to match in successive frames. The average response over N keypoints can be computed

$$j = \sum_{n=0}^{N} \frac{response(n)}{N} \tag{7}$$

where N is the number of extracted keypoints and $response(n)$ is the response value for a given keypoint. Keeping in mind that any node which is not the last produces data for the next one, it is possible to reason in terms of utility for the subsequent process. In this case it is possible to learn or build predictive models for data fitness. For instance, the objective of keypoint extraction is to maximise their matchability. A learning based approach trying to predict this factor and to select matchable keypoints is presented in [8] and has been applied to the Structure-from-Motion (SfM) problem. Alternatively, it is possible to discretise the set of input parameters for the feature extraction process and estimate over a representative dataset what configuration leads to the highest ratio of matched keypoints.

$$j = \frac{|\text{matches}|}{|\text{keypoints}|} \tag{8}$$

This may not be sufficient since we want to maximise correct matches rather than just the number of matches. We can then estimate the percentage of correct matches w.r.t. the response of keypoints. Trivially, this turns out to be significantly higher in points with a high response value.

Interest point matching. Having enough matches is crucial for the execution of a visual odometry algorithm. The lack of good matches, or matches at all, easily leads to incorrect pose estimation. It is not only necessary to produce matches but also to ensure their correctness. Evaluating the matching distance can help to assess the quality of the matches, taking into account that a good number of matches can be more robust to outliers. In this case the function j can be the average of the matching distance and $\boldsymbol{u}$ mostly revolves around the percentage of matches to accept.

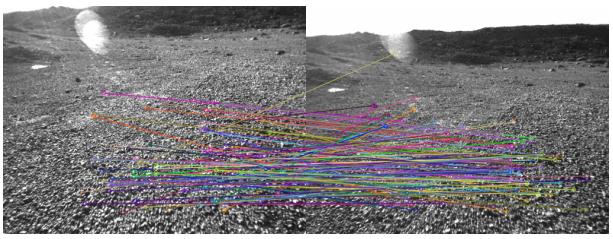
$$j = \frac{\sum_{m=0}^{M} d(m)}{\sum_{n=0}^{N} \frac{d(n)}{N}} \tag{9}$$

Eq. 9 computes a utility value based on the ratio between the sum of distances $d(m)$ of the M accepted matches and the average distance of the entire set of N matches. By not dividing the numerator by M (i.e. not using the average), the function favours larger sets of validated matches.

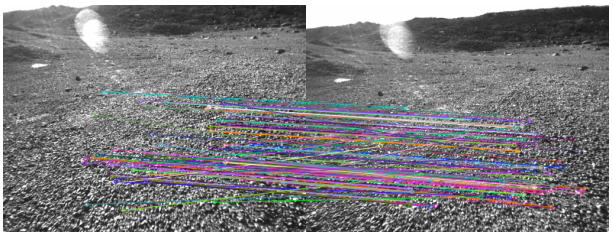
Triangulation. A correct stereo matching process enables to produce 3D points through triangulation. The precision of the keypoint extraction and of the matches, along with their position w.r.t. the camera positions impacts the 3D estimated covariance. As in [3], it is possible to compute the covariance matrix C_e of a euclidean 3D point starting from its corresponding coordinates in the stereo images and their respective covariance matrices in 2D. In stereovision, for small angle differences between two points, it is common to have a high covariance over the depth axis. It is desirable to minimise the presence of uncertain points in order to reduce the error at pose estimation stage.

Motion estimation. Finally, as the 3D to 2D estimation is incorporated in a RANSAC scheme it is possible to consider the number of inliers given a desired reprojection error, which can be tuned. It is worth to note that [5] and [14] introduced methods for including observation uncertainty into the PnP problem. It is part of our planned future works to incorporate the stochastic aspect into the 3D to 2D pose estimation node. It also worth to

remark that in case of a 3D to 3D estimation, an uncertainty measure in form of a covariance matrix is naturally obtained during the computation of the rotation matrix [13].



(a) Fine gravel results in a noisy image producing wrong matches between two time instants (left t, left $t+1$). Far points are filtered by default due to high depth error.



(b) The same image produces many more acceptable matches if compared with another after a smaller motion has been performed.

Figure 3: Matching quality difference changing the spatial frequency

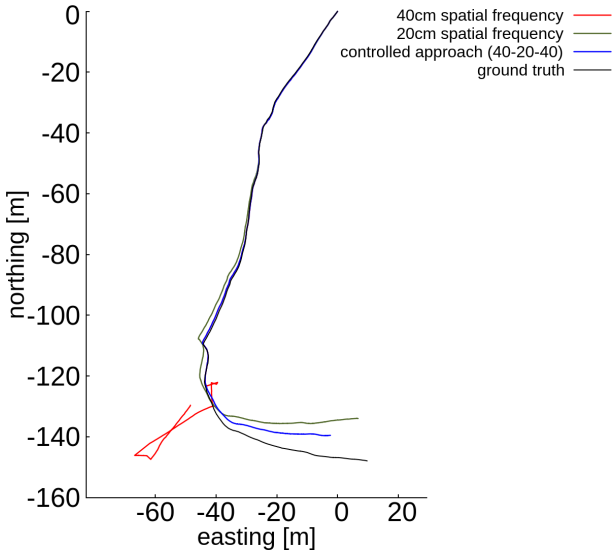
5. PARAMETERS CONTROL AND MANIPULATION

We identify two control threads: *passive-active perception*, on which we focus in this paper, bounded in the control of the perception process itself, and *active perception*, aiming to tighten the link between control, perception and planning by acting at different abstraction level to serve the perception layer. In this section we show the execution of our visual odometry algorithm with data evaluation of several nodes in different contexts.

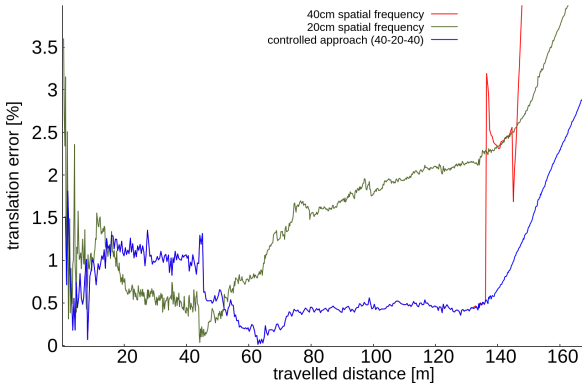
Fixing the data quality assessment functions, we aim to control the set of input parameters $\boldsymbol{u}$ to apply Eq. 2 along with other compounds parameters. Despite an autonomous reconfiguration of perception nodes is outside the scope of this paper, it is worth to show control means and control points to our use case. The advantage of our approach lies in the easiness of changing the input parameters of a node at runtime, especially if done in the scheme of a predictive model, *i.e.* before faults and with no need to backtrack.

5.1. Data Sets

The chosen data set is the Devon Island Rover Navigation Dataset collected by the Autonomous Space Robotics



(a) Pose estimation using different strategies. Coordinates are expressed w.r.t. the origin



(b) Errors in percentage of the travelled distance

Figure 4: In green, estimation performed at ~20cm/frame spatial frequency, the red line, a ~40 cm/frame, is largely overlapped with the blue line, a controlled execution using both frequencies at different stages. The ground truth is represented with a black line. The controlled approach proves to be more accurate. The 40 cm error suddenly jumps after not finding inliers in the fine gravel area.

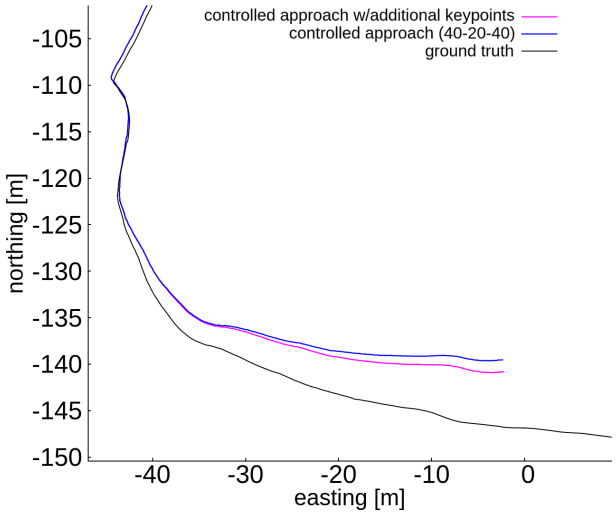
Lab of the University of Toronto [6]. The data has been collected at a Mars/Moon analogue site in the Canadian High Arctic region. It represents relevant characteristic distinctive of planetary-like environments. The data set features a set of of rectified stereo pair images collected over a 10 km traverse. Two different resolutions are provided: 1280x960 and 512x384. To simulate a planetary exploration set-up we use the smaller resolution, which is less intensive in terms of computation. The results will be shown for the first part of the traverse, approximately 200 m. The result of visual odometry will be evaluated against the ground truth obtained with a differential GPS.

5.2. Experiments

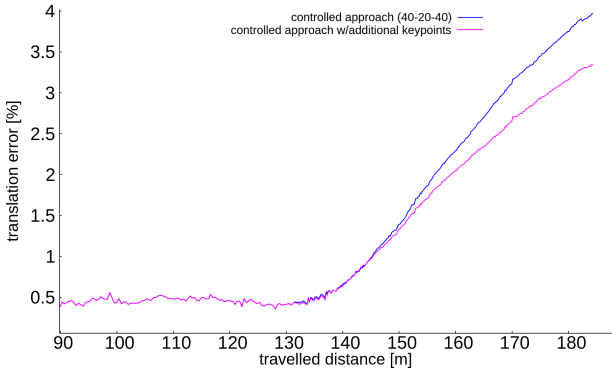
It is impossible to manually explore the huge search space of all the parameters involved in visual odometry. Driven by our knowledge, as most roboticians do nowadays, we configured our algorithm based on a priori knowledge of the scene, experience and empirical results obtained by trial and error. Nonetheless, a robot cannot always predict what situations it is going to face and has to prove adaptive to the operational context.

At some point during the traversal, the terrain changed from well textured rocks producing repeatable keypoints to a fine gravel layer which made the matching process much harder (Fig. 3a). By reducing the spatial frequency between two consecutive frames a much better tracking can be achieved (Fig. 3b). This could lead to think a higher frequency would yield better results in any case, but as mentioned in Sec. 3.4, a too low spatial frequency negatively impacts the estimation, leading to a higher drift. In this case, a relevant the criteria to select the frequency is the percentage of inliers produced by the pose estimation (Fig. 6b). The algorithm kept a 40 cm spatial frequency for the first part of the traverse, reducing it to 20 cm in case of fine gravel areas. Once the scene returned better matches, the spatial frequency is reset to the initial value to limit error accumulation (Fig. 4).

As indicated by the drop in the number of inliers, it can be beneficial (or even necessary) to modify a subset of the input parameters. For instance, increasing the number of keypoints in the first stage of the algorithm allows to feed a higher number of 3D points to the pose estimation stage, slightly improving its performances (Fig. 5). Note that extracting a too high number of keypoints is demanding in terms of computational time and is done only when necessary, as indicated by the the control point. Without action, *i.e.* proceeding with the same spatial frequency, the algorithm produced five gross errors in the pose (jumps), with erroneous translations and rotations (Fig. 6a).



(a) Pose estimation increasing the number of features during a controlled execution



(b) Errors in percentage of the travelled distance

Figure 5: Temporarily increasing the number of extracted keypoints in noisy area can help achieving significantly better motion estimation

6. CONCLUSIONS AND FUTURE WORKS

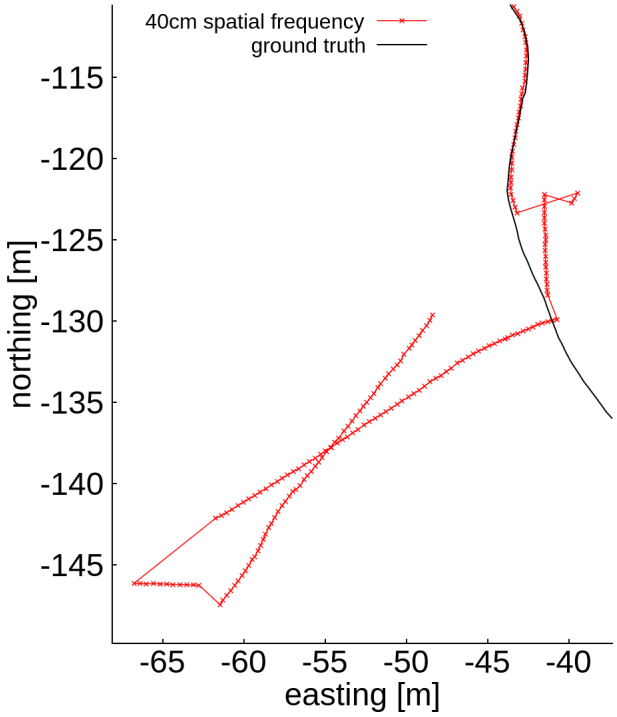
Perception processes can be formally modelled but it is still a hard task to link their model to the instantiations. We proposed an initial model to help controlling these processes. Despite a scheme to autonomously reconfigure the nodes is yet to be defined, this work shows how it is possible to evaluate either atomic functions or compounds through data quality assessment functions. Monitoring these figures of merit can trigger control of the perception node and compound parameters.

Future extensions of this work directly point towards the definition of a reasoning framework to control perception processes. Adapting to the slightest change in the operational scenario is necessary to produce the best possible results regardless of the working conditions. Nonetheless, the search space for the optimal set of controllable parameters is very large, making a blind search approach unfeasible. Resorting to predictive models helps dealing with this problem by supplying a belief for an optimal configuration. This kind of models can be trained and used to find an initial configuration for the input parameters. The search can then be performed in the neighbourhood of this configuration.

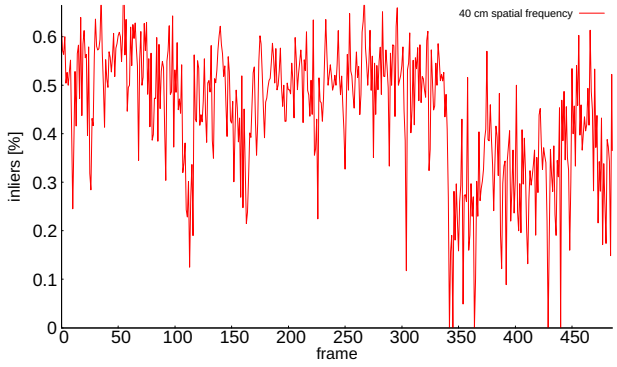
Furthermore, additional elements can be leveraged to optimise the perception outcome. Controlling the robot motion can indirectly achieve the same result as controlling the process execution frequency. Additionally, the viewpoint selection problem directly conditions the output of all the nodes in the compound. Steering the pan-tilt unit of a camera towards more textured areas can make the difference between an efficient estimation and the lack of convergence. These all represent actions that can be carried out serving the perception layer, driven by the observations carried out by data quality assessment functions.

7. ACKNOWLEDGEMENTS

This work has been partically supported by the InFuse H2020 project, funded under the European Commission Horizon 2020 Space Strategic Research Clusters - Operational Grants, grant number 730014.



(a) Close up view of erroneous estimation with low frequency



(b) Inliers percentage at pose estimation stage

Figure 6: Maintaining a low spatial frequency fixed, the algorithm produces five completely wrong estimations which are reflected in five drops of inliers to zero.

REFERENCES

[1] Aqel, M. O., Marhaban, M. H., Saripan, M. I., and Ismail, N. B. (2016). Review of visual odometry: types, approaches, challenges, and applications. *SpringerPlus*, 5(1):1897.

[2] Bajcsy, R. (1988). Active perception. *Proceedings of the IEEE*, 76(8):966–1005.

[3] Beder, C. and Steffen, R. (2006). Determining an initial image pair for fixing the scale of a 3d reconstruction from an image sequence. In *Joint Pattern Recognition Symposium*, pages 657–666. Springer.

[4] De Maio, A. and Lacroix, S. (2017). Towards a versatile framework to integrate and control perception processes for autonomous robots. In *12th national conference on Software & Hardware Architectures for Robots Control & Autonomous CPS*.

[5] Ferraz Colomina, L., Binefa, X., and Moreno-Noguer, F. (2014). Leveraging feature uncertainty in the pnp problem. In *Proceedings of the BMVC 2014 British Machine Vision Conference*, pages 1–13.

[6] Furgale, P., Carle, P., Enright, J., and Barfoot, T. D. (2012). The devon island rover navigation dataset. *The International Journal of Robotics Research*, 31(6):707–713.

[7] Govindaraj, S., Gancet, J., Post, M., Dominguez, R., Souvannavong, F., Lacroix, S., Smisek, M., Hildalgo-Carrio, J., Wehbe, B., Fabisch, A., et al. (2017). Infuse: A comprehensive framework for data fusion in space robotics. In *14th Symposium on Advanced Space Technologies in Robotics and Automation*, page 8p.

[8] Hartmann, W., Havlena, M., and Schindler, K. (2014). Predicting matchability. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 9–16.

[9] Maimone, M., Cheng, Y., and Matthies, L. (2007). Two years of visual odometry on the mars exploration rovers. *Journal of Field Robotics*, 24(3):169–186.

[10] Nistér, D., Naroditsky, O., and Bergen, J. (2004). Visual odometry. In *Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on*, volume 1, pages I–I. Ieee.

[11] Rublee, E., Rabaud, V., Konolige, K., and Bradski, G. (2011). Orb: An efficient alternative to sift or surf. In *Computer Vision (ICCV), 2011 IEEE international conference on*, pages 2564–2571. IEEE.

[12] Scaramuzza, D. and Fraundorfer, F. (2011). Visual odometry [tutorial]. *IEEE robotics & automation magazine*, 18(4):80–92.

[13] Sorkine-Hornung, O. and Rabinovich, M. (2017). Least-squares rigid motion using svd. *Computing*, 1:1.

[14] Urban, S., Leitloff, J., and Hinz, S. (2016). Mlpnp-a real-time maximum likelihood solution to the perspective-n-point problem. *arXiv preprint arXiv:1607.08112*.