Optical Flow-based Techniques for ExoMars Rover Autonomous Navigation

Yang Gao
Surrey Space Centre, University of Surrey, UK
yang.gao@surrey.ac.uk

Abstract

In this paper, we have introduced the baseline design of ExoMars navigation system and proposed a scheme to implement an optical flow package to perform three major tasks, namely visual odometry, target tracking and on-the-fly obstacle avoidance. The proposed scheme could potentially improve the autonomy of a rover and enable it travel longer for each planned traverse.

1. Introduction

Rovers have been increasingly adopted for planetary missions because of the long-range surface mobility they can offer. This includes many past/current/future missions to the Mars such as Pathfinder, MER, and ExoMars, etc. Rover operation on Mars is a challenging task due to the long distance communication with the Earth. For example, the two MER rovers are remotely navigated by the Earth ground station and can sometimes only travel a few meters per Martian day (sol). Increasing autonomy of the rover can help to reduce the ground station intervention and improve the operational efficiency of the rover. For instance, an onboard autonomous navigation system can enable the rover to sense, plan and act automatically and thus quickly. This can be further improved by speeding up the perception, decision and action.

The ExoMars is an ESA flagship mission to search for life signature on Mars due for launch in 2013. Critical to this objective, the ExoMars rover must provide high mobility across potentially rugged terrain. This is planned for the ExoMars rover to achieve navigation autonomously with minimal reliance with the ground station. The currently perceived baseline approach replies on CNES navigation software and stereovision. In broad terms, the CNES solution can be considered as a 'traditional approach' to navigation which employs stereovision and A* type path planning to provide a rover with independent navigation capability. Such methods are known to be computationally intensive. This imposes severe limitations given the relatively scarce onboard computational resources on the rover. Research in the area of autonomous navigation has advanced significantly in recent years particularly with the development of optical flow and ego-motion techniques. It seems prudent therefore to revisit the baseline approach to ascertain if performance can be improved in the light of recent advances.

In this study, we investigate various optical flow based techniques to enhance the existing baseline design. Optical flow algorithms estimate motion velocity vectors which are capable to detect and trace moving objects from 2D images without extensive 3D vision processing. This formulates the major advantage of incorporating optical flow in the ExoMars rover navigation software.

2. ExoMars Rover Navigation Baseline

In the ExoMars rover Phase-A study, a baseline approach to navigation system has been proposed after reviewing three existing approaches namely NASA MER, CNES and LAAS [1,3]. In a broad sense, an autonomous navigation system involves three high-level functionalities: 1) *perception* of the environment including self-localization with respect to landmarks; 2) *decision* on path planning; 3) *action* on path traversal. The currently perceived baseline approach to provide these functions relies heavily on stereovision and CNES navigation software, which has shown a good degree of maturity and a higher efficiency. Nevertheless, it is suggested to consider further developments and improvements. The baseline approach can perform the three high-level functionalities in 1)-3) mentioned before. However, the corresponding low-level functionalities are assembled and configured by the operators depending on the actual rover context and the current mission needs, classified by different *operation mode* (see **Table 1**).

Table 1: Navigation functionalities involved in various operation modes [1]

	Perception				Decision	Action
	Environment data acquisition	Localization	Navigation map building	Target tracking	Path Planning	Locomotion control & monitoring
Direct Control Mode	X	X				X
Safeguarded Mode	X	X			X	X
Science Target Reaching Mode	X	X	X	X	X	X
Long Range Traverse Mode	X	X	X		X	X

Environment data acquisition consists in obtaining geometric and visual information of the environment. The baseline design uses pixel-based stereovision or dense stereovision used by CNES software, including image acquisition, image sub-sampling, image rectification, disparity search, disparity filtering and 3D reconstruction.

Localization consists in estimating both the rover 6 position parameters (3 translations and 3 orientations) and corresponding errors. The baseline approach recommends 3D odometry using fusion data from wheel encoders, steering angles, chassis internal configuration angles, heading gyro, and inertial measurement unit (IMU), and possibly to consider visual odometry based on stereo images.

Navigation map building consists in structuring 3D data provided by stereovision. CNES software can be used to generate 3D Digital Elevation Map (D.E.M.). It is then used to perform terrain navigability analysis based on rover capabilities. This produces a local 2D navigation map to merge and update a global navigation map. For the first three modes, the navigation map is only maintained locally, i.e. on a small surface surrounding the rover.

Target tracking is to localize the target in the successive images taken during the motions and to estimate the relative rover/target position. No baseline algorithm is yet specified.

Path Planning is to determine safe trajectories to execute in order to reach the goal specified by the operators. The operator will specify an initial trajectory in the 2nd and 3rd operation modes. In safeguarded mode, the path planning algorithm is to check whether the initial trajectory is feasible and if not an alternative trajectory must be specified. In the target reaching mode, the path planning algorithm updates (or slight modifies) the initial trajectory according to the evolution of the relative rover/target position. In the long range traverse mode, the motion generation algorithm is to define the trajectories autonomously for the rover to reach the goal. For the last mode, CNES approach can be applied.

Locomotion control and monitoring is to control the rover actuators in order to faithfully move according to the trajectory planned and monitor rover motions to detect any dangerous situation and reacts accordingly. To control the wheels motion, two modules are involved based on developments of Solero rover at EPFL in order to minimize wheel slippage. The first module involves derivation of mathematical equations that represent geometrical state of the rover (relative and absolute wheels and links positions) and physical equations modelling wheel-ground interaction and forces distribution. Such a model is necessary to understand the physical constraints on each wheel and to select the appropriate commands. The second module uses the information provided by the physical model and selects the best set of commands for the wheel motors (torque commands). Locomotion is monitored by checking various parameters during the motions, such as rover position with respect to the reference trajectory, attitude parameters, attitude angles, internal chassis configuration angles, and behaviour of localization algorithms, etc. If locomotion errors exceed given thresholds, the rover is stopped.

The ground operators are to select the algorithms depending on the environment context and the operation mode, and to trigger the functionality. For instance, in the long range traverse mode the rover can stop, observe the surroundings plan a route and then proceed with the planned trajectory using CNES approach; while in the safeguarded and target tracking modes the rover can run perception and decision functionalities when it is moving. It is clear that open issues are suggested in the existing baseline approach (such as visual odometry, target tracking, obstacle avoidance, and locomotion control and monitoring) and options are welcomed to implement new techniques in the light of the recent advances. In this study, we are to investigate optical flow based techniques and access their feasibilities to fill in the gaps and compromise with the baseline approach framework. Optical flow algorithms in general offer a robust strategy to detect image motions and identify characteristic features (e.g.

landmarks, targets or obstacles) without extensive processing of stereovision. This enables the rover to take images and plan its path while in motion (react on-the-fly), and thus to travel longer distance for each traverse.

3. Optical Flow

Optical flow (OF) is a technique inspired by the navigation systems of insects and birds. It describes the apparent motion (direction & speed) of the brightness patterns in the image, which can be derived from consecutive 2D images without the need for complex 3D object recognition. This is based on the principle that the moving pattern in the image causes temporal variation of the image brightness or intensity. Given a sequence of images, OF approximates local image motion based upon local image intensity derivatives. That is, in 2D it specifies how much each image pixel moves between adjacent images. It is assumed that all temporal intensity changes are due to motion only[1]. This technique can therefore extract the moving patterns in the image that may well represent useful objects in the scene such as obstacles or landmark features. The optical flow vectors are derived as follows:

Firstly, assume $I(x,y,t)$ is the image intensity of pixel (x,y) at time t and moves by $\delta x, \delta y$ in time δt to $I(x+\delta x, y+\delta y, t+\delta t)$. Since $I(x,y,t)$ and $I(x+\delta x, y+\delta y, t+\delta t)$ are the images of the same point, we assume

$$I(x+\delta x, y+\delta y, t+\delta t) = I(x,y,t) \quad (1)$$

This assumption is true to a first approximation (small local translations) provided $\delta x, \delta y, \delta t$ are not too big. If performing a 1st order Taylor series expansion of **Error! Reference source not found.**, we obtain:

$$I(x+\delta x, y+\delta y, t+\delta t) = I(x,y,t) + \frac{\partial I}{\partial x}\delta x + \frac{\partial I}{\partial y}\delta y + \frac{\partial I}{\partial t}\delta t + H.O.T \quad (1)$$

where H.O.T (Higher Order Terms) is small and can safely be ignored. Using the above two equations, we obtain 2D motion constraint equation below:

$$\frac{\partial I}{\partial x}\delta x + \frac{\partial I}{\partial y}\delta y + \frac{\partial I}{\partial t}\delta t = 0 \quad \text{or} \quad (2)$$

$$\nabla I(x,y,t)\cdot \vec{v} + I_t(x,y,t) = 0$$

where $\nabla I = (\frac{\partial I}{\partial x}, \frac{\partial I}{\partial y})$ is the spatial intensity derivative, and $I_t = \frac{\partial I}{\partial t}$ is temporal intensity derivative. The field of optical flow or image velocity vector $\vec{v} = (v_x, v_y)$ can be calculated at each pixel in a 2D image based on (2. This means we need to solve one equation with two unknowns, which has a consequence of the *aperture* problem. Normal velocity then is a local phenomenon and occurs when there is insufficient local intensity structure to allow a full image velocity to be recovered. In this case, only the component of velocity normal to the local intensity structure (e.g. an edge) can be recovered. The tangential component of the velocity cannot be recovered. The problem of computing full image velocity then becomes finding an additional constraint that yields a second different equation in the same unknowns.

There are several algorithms to calculate the OF field of $\vec{v}$ from (2. Barron *et al* provided an in-depth survey of different algorithms and classified them into four methods including differential, matching, energy-based and phase-based [4]. This remains the definitive comparison study in the area. This survey highlights the gradient-based image-matching algorithm proposed by Lucas and Kanade [5] as effective across both synthetic and real world image sequences. As far as this study is concerned, we adopt Lucas and Kanade as the baseline approach to calculate the optical flow vectors. The baseline approach implements a weighted least-square (LS) fit of local first-order constraint in (2. A constant model of $\vec{v}$ is obtained in a small spatial neighborhood $\Omega \in \Re^{n\times n}$ by minimizing:

$$\sum_{x,y\in\Omega} W^2(x,y)[I_x v_x + I_y v_y + I_t]^2 \quad (3)$$

where $W(x,y)$ denotes a window function that gives more influence to constraints at the centre of the neighborhood than those at the periphery, containing typically 2D Gaussian coefficients. The solution to (3 is given by:

$$\begin{bmatrix} \Sigma W^2 I_x^2 & \Sigma W^2 I_x I_y \\ \Sigma W^2 I_y I_x & \Sigma W^2 I_y^2 \end{bmatrix} \begin{bmatrix} v_x \\ v_y \end{bmatrix} = -\begin{bmatrix} \Sigma W^2 I_x I_t \\ \Sigma W^2 I_y I_t \end{bmatrix} \quad (4)$$

[1] This is assumed that 1) no occlusion (one object moving in front of or behind another object), 2) no specularities in the scene, and 3) all objects in the scene are rigid, no shape changes, unless the above mentioned scenarios being modeled

4. Optical Flow for ExoMars Rover Navigation

The optical flow field can be used to identify objects and local path of the rover, and thus to predict collision and future course. We have provided an example in Figure 1 to demonstrate how this can be done using two successive images on rocky terrain. Characteristic features can be extracted from the flow field, including landmarks to refer to, targets to follow or even obstacles to avoid. These capabilities can address a number of open issues of the ExoMars rover navigation system as mentioned before, such as visual odometry, target reaching and on-the-fly obstacle avoidance. In literature, studies have demonstrated navigation techniques for both indoor and outdoor applications using OF. It is therefore proven to be a valid technique for real-time implementation.

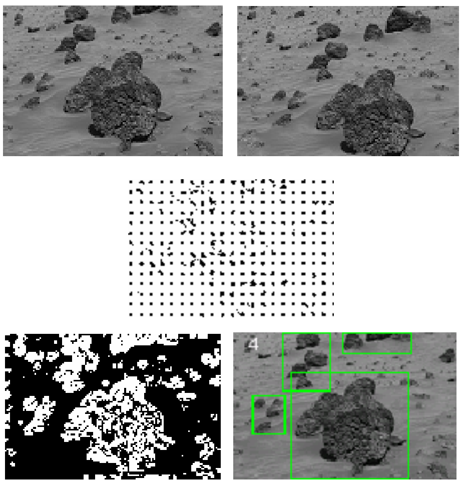

Figure 1: Above: two consecutive images; Middle: optical flow field calculated using Lucas & Kanade algorithm; Bottom: moving pattern & objects window

In this study, we propose an add-on package to the ExoMars navigation software which implements the optical flow algorithms and provide useful information on identifiable features and motions. The plug-in OF module can interface with the existing modules as suggested by the workflow chart in Figure 2 and address the following three key issues:

Visual Odometry

In this context, visual odometry is to estimate movement of the rover from a sequence (2 or more) of 2D images taken while moving. CNES recognizes this as the best candidate to periodically improve odometry estimation [6]. OF module can select characteristic features (landmarks) in the 2D images that are stable enough to be identified in the next images, e.g. rocks, horizon, etc. Existing example is the autonomous helicopter project of Carnegie Mellon University which has successfully implemented OF to determine position of the helicopter for its on-board visual navigation system [7].

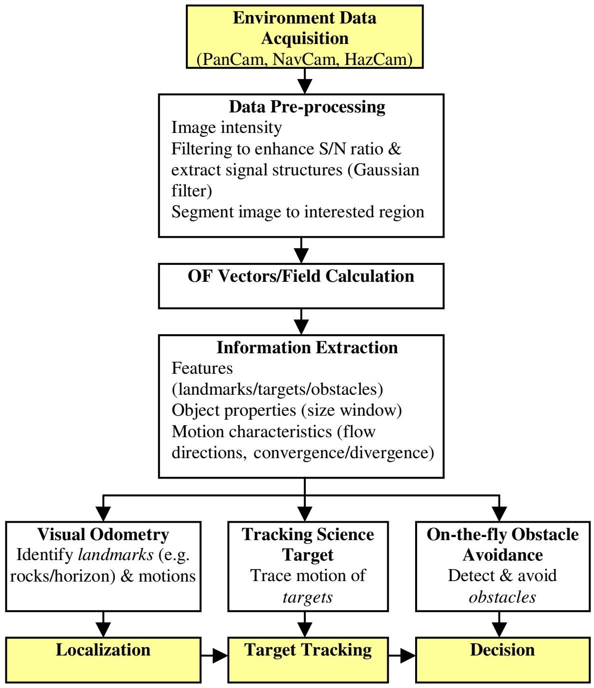

Figure 2: OF Module Workflow

Target tracking

This task is to identify and follow target based on 2D image processing until close enough to the goal for 3D model generation. OF module can provide flow vectors of the target feature in the 2D image and to predict the future course. CNES suggested footprint for this process should be less than 2 m, which OF can easily cope with. This technique has been demonstrated extensively for terrestrial applications such as tracking cars for traffic control.

One-the-fly obstacle avoidance

The task is to detect obstacles, local path of the rover and thus avoid collision. This needs to be performed by the rover under safeguarded mode. OF module can identify obstacles in consecutive 2D images and potentially apply balance strategy to avoid obstacles (as the birds and insects do). Many existing examples are available for indoor mobile robot navigation such as by MIT and NIST.

OF module takes input of consecutive 2D images from the environment data acquisition module. This is relaxed to any on-board vision sensors, e.g. PanCam, NavCam or HazCam. Primarily, the key outputs of the OF module are OF field and image moving patterns. The information will be further analysed and interpreted as characteristic features (e.g. landmarks/targets/obstacles), properties of those features such as size window, and motion characteristics such as directions, divergence or convergence. These results are used in determining functionalities of visual odometry, target tracking and obstacle avoidance. It thus interfaces with three functional modules of the navigation software at the bottom, namely localization, target tracking and decision.

5. Simulations Using MER Images

MER rovers equipped with PanCam, NavCam and HazCam have obtained many images of the Martian surface. Figure 3 shows a collection of these images taken at site 11 in Sol 55 of the mission. Here, we applied the OF algorithm on a number of sample images to test its capability of extracting features. The objective of these tests is to demonstrate reliability of the technique and fast processing speed. All simulations were run using a PC of 500MHz processor and 200MB memory. All images are originally in the size of 1028 x 1028 pixels.

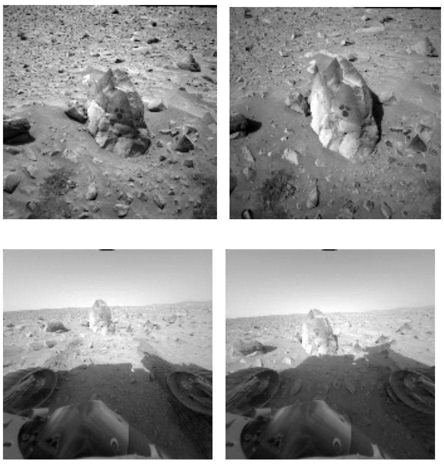

Figure 3: Image collections by MER Spirit in Sol 55 at Site 11: NavCam (top); HazCam (bottom)

The first simulation uses two consecutive images from NavCam in Figure 3. Footprint of the rover in between of taking these images is approximately 0.5 metres. The images were pre-processed to lower resolution images of 128 x 128 pixels before applying OF algorithm. Results shown in Figure 4 include OF field and extracted features. Computation time is less than 0.1 second. The process time can be further reduced if we use even smaller images such as 64 x 64 pixels. As shown in Figure 4, the OF field obtained in both cases can extract similar features.

Another similar simulation was carried out on two HazCam in Figure 3. Same image size results in similar computation time as in the first simulation. The difference in this case is footprint between these images is much larger (approximately 1~2 m). As shown in Figure 5, the OF module can still identify characteristic features accurately and quickly even under considerable footprints.

The simulation shows the OF can work friendly and effectively with the real Martian images. The proposed OF module provides robust results for both small and large footprints. It is both workable and tolerable to low-resolution images. This indicates the potential of producing more responsive decision, allowing the rover to traverse longer in between these decisions, and eventually improving process efficiency of the navigation software. This will bring behaviour-based technique into the perception, trajectory planning and execution bypassing the computationally expensive steps. Ideally this will help to reduce the rover stops for 3D perception and eventually improve operation efficiency of the existing system. Furthermore, in case of malfunction to the globe navigation system such as stereo camera failure, local navigation using OF can help to command the rover.

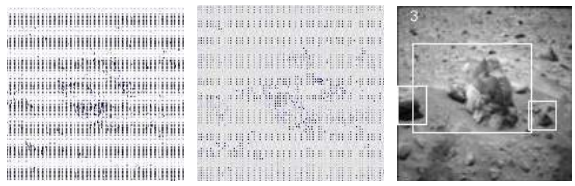

Figure 4: OF field128x128 (left), OF field 64x64 (middle), object window (right)

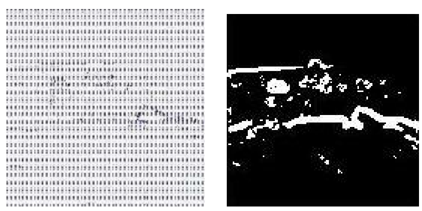

Figure 5: OF field (left) & moving patterns (right)

6. Conclusions

The major contributions of this study are: 1) having proposed a scheme and three scenarios to implement OF into ExoMars rover navigation software and complement to the baseline framework without adding additional sensors; 2) having demonstrated the capability and fast processing speed of the OF algorithms on real Martian images which makes the proposed scheme more promising.

In this study, we have not investigated in detail how to translate the OF calculation into control commands due to lacking of understanding and information in the following steps. These need to be worked out in future. We think it is important to understand the commands configuration of the interfacing modules and to better decide how to integrate them together. Robustness of the proposed techniques needs to be studied more closely. This would require a lot more tests on images with different size, resolutions and footprints with respect to requirements under different operation modes or tasks.

7. Acknowledgments

This work is supported by UK Particle Physics and Astronomy Research Council under CREST Project on ExoMars Rover Technology Development (PP/D006457/1).

8. References

[1] Rover Team (2004), Pasteur Exobiology Payload & Rover for ExoMars TN2/TN3 - Design Concepts & Trade-off Analysis, TN4 - ExoMars Rover/Pasteur: System & Subsystem Design Report, EADS Astrium

[2] Y. Gao, Optic Flow based Navigation Techniques – Simple, Fast and Reliable Methods to Enhance ExoMars Rover Navigation Software, PPARC, Reference No. PP/D006457/1, pp. 1-10, May, 2007

[3] CNES Autonomous Navigation: Basic description and preliminary requirements. DTS/AE/SEA/ER/2003-009

[4] Barron, J.L., Fleet, D.J., Beauchemin, S.S. (1994), Performance of Optic Flow Techniques, *IJCV* 12(1): 43-77

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

[6] L. Rastel and M. Maurette, "Autonomous Navigation: A Development Roadmap for ExoMars," Proc. ESA Advanced Space Technologies for Robotics & Automation (ASTRA), ESTEC, Noordwijk, Netherlands, November, 2006.

[7] Amidi, O. (1996), An Autonomous Vision-Guided Helicopter. *Ph.D. Thesis*, Electrical & Computer Engineering Department, Carnegie Mellon University