

# HARDWARE ACCELERATED VISUAL LOCALISATION FOR NEXT GENERATION MARS ROVERS

Daniel Townson <sup>(1)</sup>, Niklaus Kamm <sup>(1)</sup>, Mark Woods <sup>(1)</sup>, Mateusz Malinowski <sup>(1)</sup>

<sup>(1)</sup> SCISYS, 23 Clothier Rd., Bristol, BS4 5SS, UK, Email: [markwoods@scisys.co.uk](mailto:markwoods@scisys.co.uk)

## ABSTRACT

Recent developments in space qualified field-programmable gate array (FPGA) technologies have enabled a range of new applications for hardware accelerated algorithms used in space applications. The reduction in execution time and resource usage allows for more complex algorithms to be deployed in a wider range of use cases. One such case is for an on-board data processing such as Visual Odometry (VO), which estimates the location of a vehicle based on current and previous image frames taken by a stereo localisation camera. This paper presents the partial transfer of the VisLoc VO algorithm to an FPGA, discusses its performance, and a wider range of potential applications in space.

## 1. VISLOC VISUAL ODOMETRY

The accurate localisation of a vehicle on the Martian surface is crucial in allowing for continued operation while direct contact to Earth is interrupted. Due to the absence of an external reference system e.g. GPS equivalent and limited direct communications with the rover, determination of the vehicle's current position and attitude has to be carried out locally.

One solution is to use vehicle's wheel odometry. This approach is the least power-consuming, but generates large error over travelled distance, which can be as high as 10% and even greater on terrain where there is a risk of a slippage. To overcome this error, Bundle Adjustment can be introduced, but these techniques are very compute-intense, require images from orbiter, and as such are carried out on the ground.



Figure 1. Example image from Atacama Desert taken by the localisation camera during SEEKER study.

The state-of-the-art solution for Mars rovers' localisation is to make estimates based on visual data gathered by cameras on board of the vehicle [1]. Estimates are based on the relative movement of the camera location from one frame to the subsequent frame, making the location data independent of the terrain and achieving a high degree of accuracy in determination of both position and attitude even in difficult conditions.

This is achieved via the extraction of features from a frame which represent highly recognizable pixels. These features are then matched with the identified features from the previous frame. When a pair is recognized, the difference in location from one frame to the next is used to geometrically infer the movement of the camera in between the frames. This means that visual odometry can be performed even in challenging terrain as long as the image is well-exposed to offer enough definition for features to be recognizable, and the movement between frames is not so large that a high proportion of features have moved off the edge of the frame before the next image is taken.

The Visual Localisation flight software algorithm (VisLoc) was developed for the ExoMars rover [2]. It is based on the core algorithm known as Oxford Visual Odometry (OVO), developed at the University of Oxford [3]. VisLoc was adapted and industrialised by SCISYS over a number of projects, where its core functionality was extensively tested in Mars-representative environments and validated with respect to the ExoMars requirements baseline for visual localisation on Mars

- ESA X-ROB study [4] – the assessment of OVO functionality.
- ESA SEEKER study [5] – OVO used in the Atacama Desert in highly representative terrain to drive autonomous navigation software for ~15km in total. Fig. 1 shows example image taken by the localisation camera. Even though this is the most expected environment, the rover also traversed through more difficult terrain over several 100's of metres as presented on Fig. 2 and Fig. 3. The algorithm dealt with successfully with these complex scenarios.
- ESA SAFER activity [6] – OVO used as a part of autonomous navigation software that was used to perform the first simulation of ExoMars mission with Remote Control Centre. Excellent localisation performance led to precise targeting

of WISDOM (ground penetrating radar) and High Resolution Camera (HRC) instruments.

- The Chameleon project [7] – OVO used as a part of autonomous navigation software to validate terrain sensitive navigation. The total distance of autonomous traverses: 6875m.



*Figure 2. Example of image of overexposed "dried lake" with underexposed background taken by the localisation camera during SEEKER study.*

SCISYS have continued VisLoc development and industrialisation, as part of the European Space Agency's (ESA) ExoMars Rover Mission where the VisLoc algorithm waits for the final acceptance testing to reach a Technology Readiness Level (TRL) of 8. VisLoc will be the first European VO system flown on a Rover system following launch of ExoMars in July 2020.

In this paper we discuss the results of a study investigating the integration of a field-programmable gate array (FPGA) board into the VisLoc algorithm to accelerate its execution time. The aim was to achieve an execution frequency of 1Hz while maintaining full parity between the software-based algorithm and its accelerated counterpart. Currently both software and hardware-accelerated versions of the algorithm are being evaluated for ESA's Sample Fetch Rover (SFR), which intends to cover considerably larger distances than ExoMars in a similar timeframe [8].

## 2. FPGA INTEGRATION

Being part of a navigation system on a Mars rover poses unique constraints for a visual localisation algorithm due to very limited availability of Central Processing Unit (CPU), electrical power, and memory. While the low velocity of the rover can largely eliminate difficulties such as motion blur, which makes it harder to extract corners from smeared images, the Martian environment and rovers themselves offer their own set of unique additional

challenges for visual localisation. Besides the lowered mass and power budgets, low image resolution of 512x512 pixels, stark shadows covering large parts of the image, rover parts protruding into the frame, and the potential for dust storms can greatly hinder the processing of an image. Large parts of images are also covered by the sky, which is not suitable for feature detection. The algorithm must thus be able to process images in real time with minimal resources, while correctly identifying and mitigating a number of factors that could render parts of an image unusable.

This large amount of processing overhead involves many mathematical operations which slow down execution of the algorithm but can be greatly accelerated by making use of pipelining and parallel processing on FPGA hardware. By demonstrating that partial porting of the algorithm's functionality to an FPGA can result in significant reduction of execution time, we could provide a range of options allowing for a trade-off between resource usage and execution time.

Suitability for hardware acceleration was assessed on a case by case basis for individual software modules of the algorithm. Factors such as sequential logic and dependencies within the process could prevent pipelining and parallelization, making modules less suitable for hardware acceleration. Additionally, the limitation in bandwidth of streaming interfaces between the CPU and FPGA introduced fixed data streaming delays before processing on the FPGA could commence, so the architectural design was adjusted to minimize streaming interfaces where possible.



*Figure 3. Typical example of bland, feature poor image from Atacama Desert taken by the localisation camera during SEEKER study*

Special care was given to ensure that outputs of the accelerated algorithm were equivalent to the software version in order to limit the effect on the TRL. This

was validated using a series of simulated trajectories, such as the one in Fig. 4 (please note similarity to the image of Atacama Desert as presented on Fig. 3), representing the simulated conditions on the Martian surface, allowing for statistical analysis of results to 3-sigma accuracy. Parity of results was retained across the simulated reference trajectory.

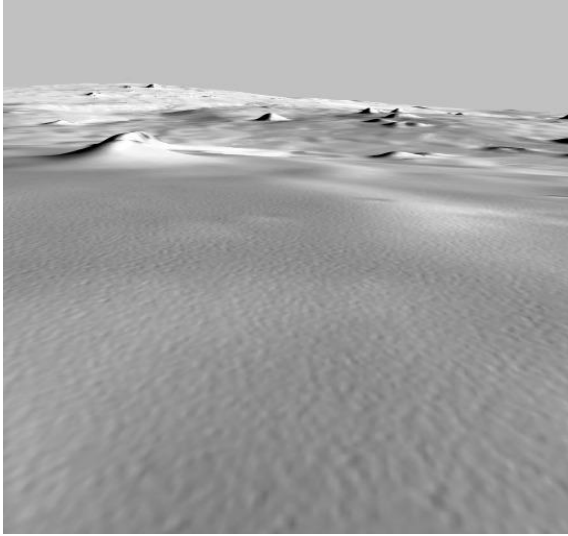


Figure 4. Example of bland frame from an artificial camera using SCISYS simulator.

Adding more functionality to the FPGA resulted in progressively diminishing returns, while increasing the required amount of the board resources. The aim was to target and evaluate execution time when all prominent functions are moved to FPGA, but also to investigate what would be the time gain when Big Re-programmable Array for Versatile Environments (BRAVE) FPGA family is considered [9]. For the purpose of this study, Xilinx Zynq Ultrascale+ MPSoC ZCU102 Evaluation Kit was used with a clock frequency set to 100MHz.

While further transfer of functionality to the FPGA, potentially even to the extent of fully running the algorithm on the hardware, is theoretically possible, it was not deemed feasible to achieve within the time constraints available.

### 3. TESTING AND VALIDATION

VisLoc was thoroughly tested on both simulated and real Mars representative trajectories. Real trajectories were taken from the SEEKER trial in the Atacama Desert in Chile, where a rover moved through an analogue environment autonomously for several kilometres [5]. These trajectories allowed for testing under very realistic circumstances, providing real camera images in a very close approximation of the operational environment on Mars. Simulated trajectories meanwhile were used for targeted testing of specific scenarios, such as areas which reflect a lot

of light, and low texture ground materials as shown in Fig. 5.

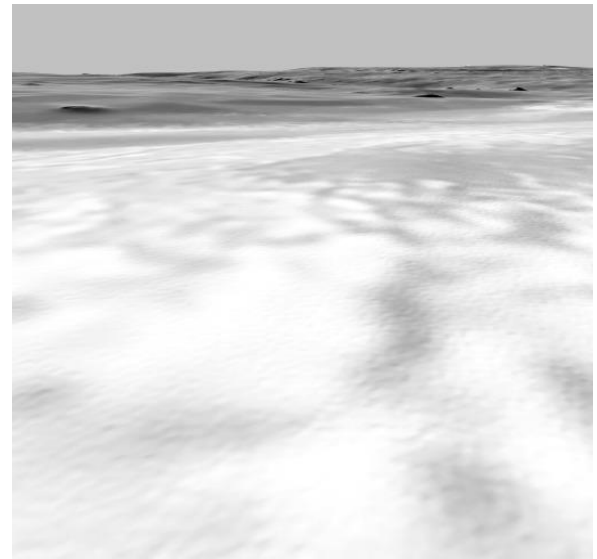


Figure 5. Image with overexposed foreground from the reference trajectory

After verifying that VisLoc was capable of meeting its requirement of an accuracy of 1% of total distance travelled in both real and simulated trajectories, the algorithm was tested under extreme conditions to determine the limitations of environments that were still sufficient for accurate visual localisation.

Individual test cases were created investigating the effects of Optical Depth (OD), camera shutter speed and vehicle velocity. All these tests were done over the same 380m-long reference trajectory where default values were: velocity: 0.1m/s, shutter speed: 1ms, and OD ( $\tau$ ): 0.1. All images were acquired with 1s interval.

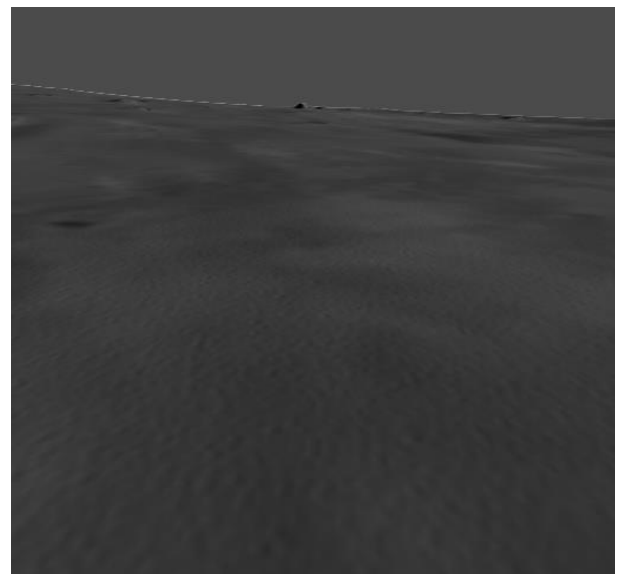


Figure 6. Simulated image at OD 2.6

VisLoc was found to be still within the required accuracy for velocities of at least 0.16m/s, double the intended velocity of SFR. It was also capable of handling optical depth values of at least 2.6, simulating severe dust storms or very dark dusk or dawn conditions using images such as the one in Fig. 6. Shutter speed appeared so be the most impactful factor on algorithm performance, with VisLoc being capable of processing images at a shutter speed of at least 100ms.

#### 4. PERFORMANCE ANALYSIS

Fig. 7 shows the reduction of the execution time as algorithm functions are gradually transferred to the FPGA on both processors. The order of the transferred functions is in this case not sequential but instead functions were prioritized based on their suitability for the FPGA and their overall execution time during a single instance of the algorithm.

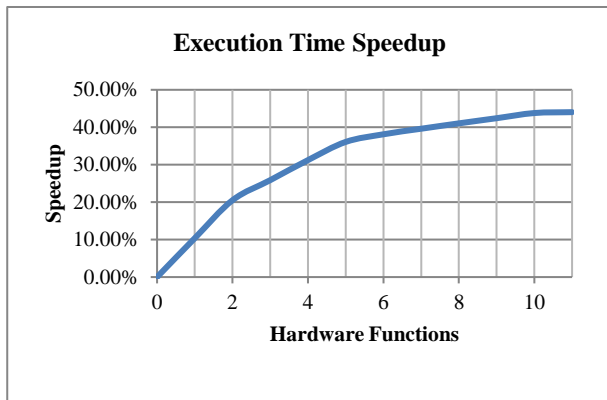


Figure 7. Execution time gain based on the number of functions moved on hardware

The transfer of the full set of selected functions resulted in a total acceleration of VisLoc algorithm by 44.03%. The execution time of accelerated functions themselves was reduced by 88.24%. The most suitable functions for hardware acceleration are image processing and corners extraction. Other parts of algorithm, even though they still benefit from FPGA, implement more sequential logic and there is therefore less speed gain to be made. It is expected that for high processor frequency the data transfer overhead would overcome the gain from the parallelisation.

Given the limited time of this study this activity was based on a LEON2 processor running at 96MHz, as baselined for ExoMars. Even though LEON4 with clock frequency of 250MHz is intended as SFR baseline processor, we expect similar execution time speedup as presented on Fig. 7. For new processor and targeted FPGA clock frequency, the set of identified functions would need to be reassessed. It is possible that some of them would no longer be

suitable for hardware acceleration given the CPU performance.

By moving only four functions to FPGA we have established that:

- Assuming a roughly linear correlation between LEON2 and LEON4 processors frequencies and execution time (i.e. not taking into account L2 cache available on LEON4) we estimate that VisLoc would take less than 1s to execute Visual Odometry (VO) on SFR-representative hardware.
- These functions would fit a single NG-Medium board from BRAVE family while maintaining 50% margin on resources

Together with moving VisLoc to FPGA, a set of SFR-relevant trajectories was defined in order to test VO under the additional harsh conditions. To make it even more difficult, trajectories were defined using very bland images as those presented earlier on Fig. 4 and Fig. 5. For all these trajectories VisLoc on FPGA was very robust and maintained the maximum error below 1%. Fig. 8 presents an example output from such a trajectory. Given the limited time on the study, only a few tests were conducted which validated VisLoc robustness on each of the above parameters individually. We do appreciate that more extensive test campaign may be required for the actual mission, however, initial results are very promising and indicate that VisLoc should not have many problems in meeting its performance requirements when severe conditions are expected.

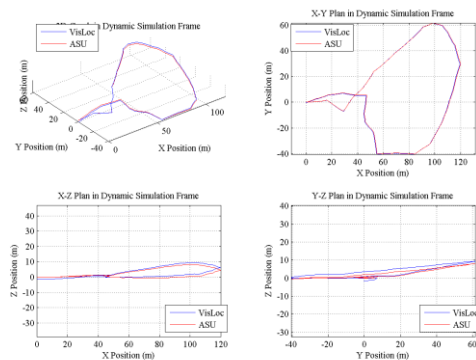


Figure 8. Sample trajectory output, estimated attitude vs ground truth

#### 5. OTHER APPLICATIONS

In parallel with the evaluation of the original OVO algorithms the Autonomy and Robotics Group at SCISYS also continued their development of science autonomy techniques [6]. These were integrated and tested with our guidance, navigation, and control system (including VO) in representative field tests in the Atacama [7]. This allowed us to identify potential areas of functional overlap and acceleration via an

FPGA. This work has now been taken further in our ESA Novelty Or Anomaly Hunter (NOAH) activity and we will report further on its conclusions in the full paper [10].

## 6. CONCLUSION

Over the course of the study it was determined that hardware acceleration using an FPGA can reduce the overall execution time of VO algorithm. Transfer of algorithm functionality to hardware can be performed in a modular manner while maintaining navigation accuracy, and can still yield significant acceleration, as long as software modules are previously scrutinized for hardware suitability. A fixed delay created by the bandwidth of streaming interfaces between CPU and FPGA hardware introduces a latency into the system. Given a modular design, modules should thus be chosen in a manner that minimizes the need for transfer of large data volumes between software and hardware. Given a sufficiently high processor frequency, hardware acceleration and the associated reduction in TRL of the new platform may become unnecessary depending on specific mission constraints. In certain cases, software execution may even outpace the hardware implementation of an individual module due to the head start granted by the lack of data streaming delay. In such a case, a modular approach to hardware acceleration may not be the best choice for the given algorithm, as a full software solution requires significantly less development time, while a full hardware solution will bring significantly higher reduction in execution time. Modular approaches are thus most suitable to algorithms that feature isolated sections of highly complex mathematical operations with limited dependencies, which can be implemented on an FPGA with maximum yield in terms of execution time while minimizing the requirement for data exchange.

Estimated execution time of VisLoc on Leon4 with FPGA, as baselined for SFR, meets the 1s requirement while maintaining the full parity with the software version (TRL 8 subject to the final acceptance testing). Additional simulation tests indicate that VisLoc should be able to maintain its performance requirements in high OD environments, for rover velocities higher than nominal, and when exposed to motion blur due to long shutter speed.

This work has also demonstrated our wider ability to deploy complex algorithms onto FPGA in an efficient manner. This allows us to introduce more intelligent solutions, such as machine learning, into the space domain.

## 7. REFERENCES

- [1] M. Maimone, Y. Cheng and L. Matthies, "Two years of visual odometry on the Mars Exploration Rovers," *Journal of Field Robotics, Special Issue on Space Robotics*, vol. 24, pp. 169-186, 2007.
- [2] D. Townson, M. Woods and S. Carnochan, "EXOMARS VISLOC – THE INDUSTRIALISED, VISUAL LOCALISATION SYSTEM FOR THE EXOMARS ROVER," in *International Symposium on Artificial Intelligence, Robotics and Automation in Space (i-SAIRAS)*, Madrid, Spain, 2018.
- [3] W. Churchill and P. Newman, "Experience Based Navigation: Theory, Practice and Implementation," Oxford University, Oxford, 2012.
- [4] A. Shaw, M. Woods, W. Churchill and P. Newman, "ROBUST VISUAL ODOMETRY FOR SPACE EXPLORATION," in *12th Symposium on Advanced Space Technologies, Robotics and Automation*, Noordwijk, 2013.
- [5] M. Woods, A. Shaw, E. Tidey, B. V. Pham, L. Simon, R. Mukherji, B. Maddison, G. Cross, A. Kisdi, W. Tubby, G. Visentin and G. Chong, "Seeker - Autonomous Long-range Rover Navigation for Remote Exploration," *Journal of Field Robotics*, pp. 940-968, 2014.
- [6] M. Woods, A. Shaw, I. Wallace, M. Malinowski and P. Rendell, "Demonstrating Autonomous Mars Rover Science Operations in the Atacama Desert," in *I-SAIRAS*, Sapporo, 2010.
- [7] M. Woods, A. Shaw, I. Wallace and M. Malinowski, "The Chameleon Field Trial: Toward Efficient, Terrain Sensitive Navigation," in *13th Symposium on Advanced Space Technologies, Robotics and Automation*, Noordwijk, 2015.
- [8] A. Merlo, J. Larranaga and P. Falkner, "Sample Fetching Rover (SFR) for MSR," in *Advanced Space Technologies, Robotics and Automation*, Noordwijk, 2013.
- [9] NanoXplore, *From eFPGA cores to RHBD System-On-Chip FPGA*, Noordwijk: ESA, 2018.
- [10] S. Karachalios, M. Woods, S. Schwenzer and L. Joudrier, "NOVELTY OR ANOMALY HUNTER: TOWARDS FLIGHT READY AUTONOMOUS SCIENCE USING STATE OF THE ART MACHINE & DEEP LEARNING," in *15th Symposium on Advanced Space Technologies, Robotics and Automation*, Noordwijk, 2019.