HAZARD DETECTION & AVOIDANCE AND OPTICAL NAVIGATION INTEGRATION DEMONSTRATION FOR AUTONOMOUS MOON LANDING APPLICATIONS

Cédric Godin, Ludwik A. Sobiesiak, Jean-François Hamel, Maxime Minville, William Travis, Bernardo Lobo-Fernandes

NGC Aerospace Ltd, 2995 boul. Industriel, Sherbrooke, Québec, J1L 2T9

ABSTRACT

Autonomous landing anywhere on the Moon is a key capability of future private and public missions. This requires hazard detection & avoidance technology as well as terrain relative navigation. Both must be tightly coupled to use the navigation output to compensate for spacecraft motion during the Lidar scan. NGC's integrated system has been tested in both a software in the loop, and a hardware in the loop simulation at NGC's dynamic test facility. Although limited by the fidelity of the simulations resulting from trajectory scaling, some areas for improvement were identified.

Three main developments were conducted during the last year to address these shortcomings: 1) image processing measurement update was changed from epipolar constraint to delta pose extracted from 5-point algorithm (robustness against no motion along the boresight axis); 2) Observe scale of the motion (impossible with a monocular camera-based system) using the Lidar; 3) Offload heavy image processing to FPGA logic to achieve targeted 10 Hz updated rate. The updated is system is currently being tested in the test facility prior to being mounted on a UAV to conduct a full-scale test campaign. The goal is to obtain quantitative performance results in a test environment not suffering form scaling limitations.

1 INTRODUCTION

As both public and private organizations embrace a renewed interest in exploring the Moon, scientists and engineers alike are pushing the envelope of what is possible. To reach lunar sites of interest, it is no longer sufficient or desirable to only land in a conservative safe spot on the surface. Indeed, future missions will require global access, meaning the ability to land anywhere, including near known and unknown hazards. Two key technologies for achieving this goal are Hazard Detection & Avoidance (HDA) and terrain-relative navigation. The HDA function processes camera and Lidar measurements to assess the landing suitability of the area in the vicinity of the planned landing site. The camera image is used to build a map of the shadowed area. A Lidar point cloud is processed to determine the terrain slope and roughness. Terrain-relative navigation determines the spacecraft motion relative to the ground using a sequence of camera images fed through a feature tracking algorithm.

During the descent, the lander motion distorts the Lidar point cloud which negatively affects the reliability of the HDA outputs. Close integration of the HDA function with the navigation system is mandatory to take the lander motion into account when processing the Lidar measurements. Demonstrating the performance of such an integrated system on the ground poses a significant challenge. The concept of using the output from a navigation filter to correct distortion in the Lidar point cloud has been demonstrated during a Software-in-the-Loop (SIL) test campaign in [1]. Results

from Hardware-in-the-loop (HIL) tests have been presented in [2] using NGC Landing Dynamic Test Facility (LDTF).

Three areas for improvement were identified during the HIL campaign:

1. Measurement updates from the image processing in the navigation filter failed to unambiguously determine the relative motion when there is little or no motion along the camera boresight axis.
2. As it is impossible to determine the scale of what is observed through a monocular camera system, an additional input was required to resolve the scale of the measured motion.
3. The image processing is too computationally heavy to run in real time on the flight computer and had to be executed on an external machine.

Work has since been conducted to improve the system performance and a new HIL campaign is underway at the time of writing. The next development step is to conduct a full-scale test campaign using an unmanned aerial vehicle (UAV) to approximate a Moon lander descent trajectory. This will allow the assessment of quantitative performance metrics of the HDA, which cannot be obtained in the LDTF, since Lidar and IMU sensor noise does not scale down with the trajectory nor with the terrain.

This paper highlights the recent developments of NGC relative navigation and hazard detection & avoidance technologies. It is organized as follows: Section 2 gives an overview of the implemented system architecture; Section 3 describes the improvements made to the navigation filter; Section 4 presents the hybrid implementation of the image processing software; Section 5 glimpses into the planned full-scale demonstration; Section 6 summarizes the developments and highlights future work.

2 SYSTEM ARCHITECTURE

The navigation and landing site assessment prototype system for UAV testing is distributed across three computers as shown in figure 1. The main flight computer is a Q7 single board computer from Xiphos available in a spaceflight-proven configuration [3]. It runs the following processes:

1. Image Processing (IP): detects and tracks features between images.
2. Onboard Software (OBSW): runs the navigation filter, manages the mission, and commands the sensors according to the current mission phase.
3. Hazard Detection and Avoidance (HDA): assesses the safety of the target landing zone and recommends the safest landing site according to the measured surface illumination, slope, and roughness.
4. Parameters management: allows the Ground Control Station (GCS) operator to update the OBSW parameters in flight and set the current mission mode.

The sensor platform computer is a x86 OWL embedded computer from VersaLogic. For space operations, this computer task would be accomplished by the spacecraft bus. It runs the following Robot Operating System (ROS) nodes:

1. Camera driver: acquire images from the camera.
2. Lidar driver: acquire points from the Lidar.
3. Altimeter preprocessing: extract a subset of the full Lidar point cloud corresponding to the Lidar field of regard when it is operating in range finding mode.

4. Lidar preprocessing: extract a subset of interest in direction of the landing site from the full Lidar point cloud and express it in a more natural frame for onboard use.
5. Image Processing COMmunication (IP COM): forward images from the camera to the IP process.
6. OBSW COMmunication (OBSW COM): sends altimeter measurements to the OBSW and retrieves the navigation and IP outputs to make them available on the ROS network.
7. HDA COMmunication (HDA COM): forwards camera images and preprocessed Lidar point cloud to the HDA and retrieves the HDA outputs to make them available on the ROS network.

The GCS computer is a standard Linux desktop or laptop. It acts as a ROS master and runs the graphical interfaces required to operate and monitor the system. It also logs test outputs for future analysis. It would not be used during a space mission as the system would operate autonomously based on the current mission phase. Logging and telemetry would have to be arranged with the spacecraft integrator.

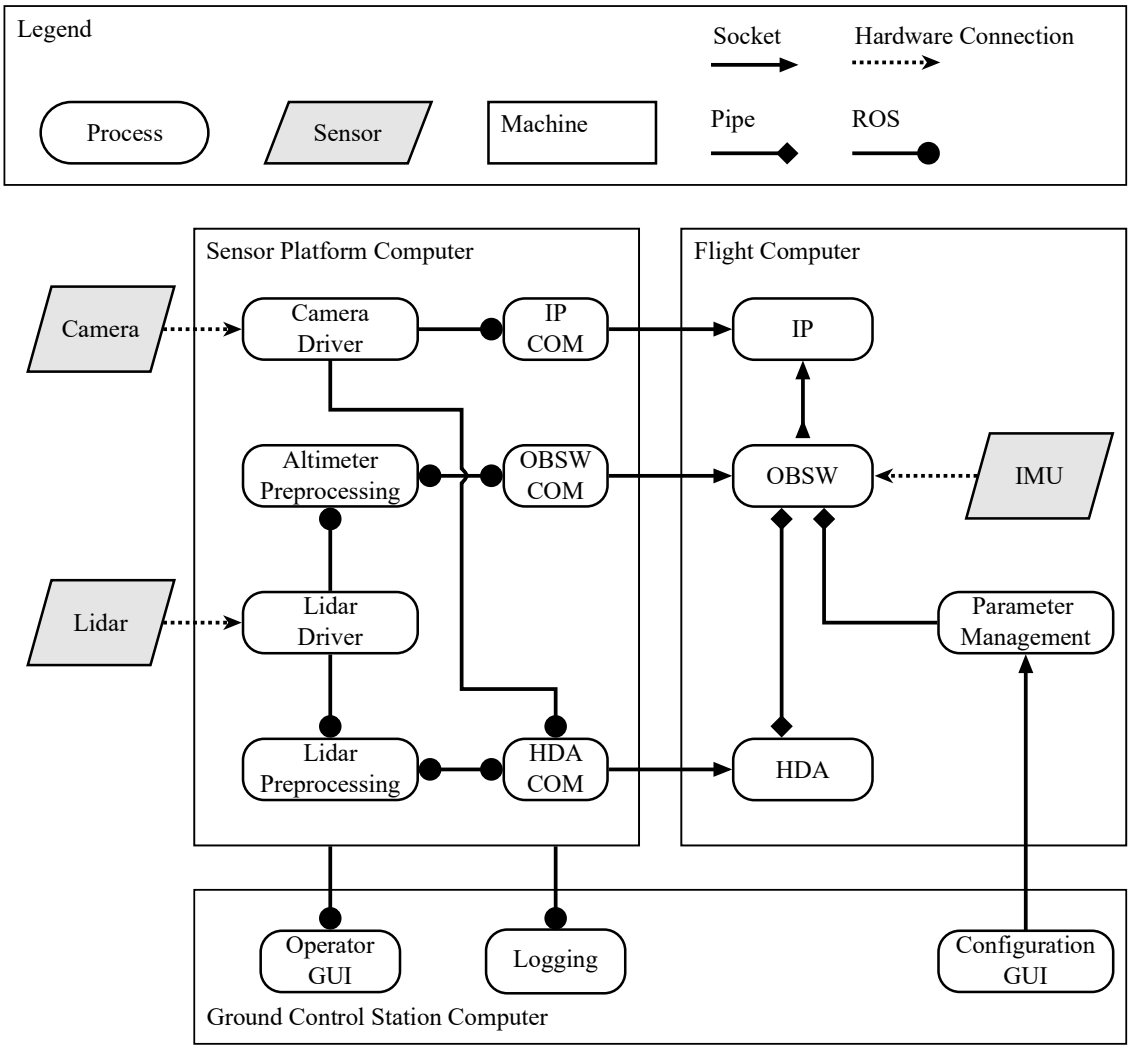

Figure 1. Navigation Landing Site Assessment System Architecture for UAV Testing

The camera is a Blackfly camera from Teledyne FLIR. It is only intended for Earth operations. The IMU is a spaceflight rated STIM318 from Safran [4]. The final demonstration will be conducted with a prototype mini-WABS scanning Lidar from MDA intended for spaceflight. It is the same sensor

used during the HIL campaign in [2]. However, an OS1 spinning Lidar from Ouster is used during preliminary tests until the mini-WABS prototype is integrated.

3 NAVIGATION FILTER UPDATES

3.1 Image Processing Measurement

The previous implementation of the image processing function from [5] used the epipolar constraint of feature tracked between two images as a measurement model for an exteroceptive update of the filter. It was observed for motion cases where there was little to motion along the boresight axis of the camera that the filter would not estimate the motion correctly. The problem is due to there being more than one solution that will minimize the epipolar measurement residual:

- any relative state vector lying in the null space of the epipolar relational matrix;
- if the relative state vector is 0.

When there is no motion along the camera boresight, the filter tends to converge to the null solution and not the true state estimate.

The new baseline design of the filter uses the relative pose change of lander between two instances in time (relative change in position and attitude) as a measurement in the filter. The determination of relative pose is a two-step process. First, the essential matrix relating matching features between two consecutive image feature sets is determined using the 5-point algorithm [6] in combination with RANSAC hypothesis testing. From the essential matrix, the relative positive is extracted and used as the image processing measurement.

3.2 Lidar Altimeter

While consecutive images from the monocular camera provide important relative motion information, the camera alone cannot resolve the scale of the motion. The relative pose measurement described in the previous section yields the direction of relative translation but not a magnitude. To determine the scale, the Lidar sensor is repurposed to provide an additional altimeter measurement, when it is not being used for hazard detection and avoidance functions. The Lidar terrain scan requires only several seconds to complete, so the sensor is available to provide an altimeter measurement for most of the descent.

Online scan pattern adaptation for motion compensation already requires the Lidar sensor to be controlled by the HDA software. The same control process used for scan pattern adaptation can be used to command an “altimeter” scan pattern when the Lidar is not scanning the surface for hazards. The Lidar uses a narrow FOV scan to scan along the sensor boresight to emulate a range finder and provide distance to ground. The terrain scanning and altimeter scanning pattern concepts are illustrated in figure 2.

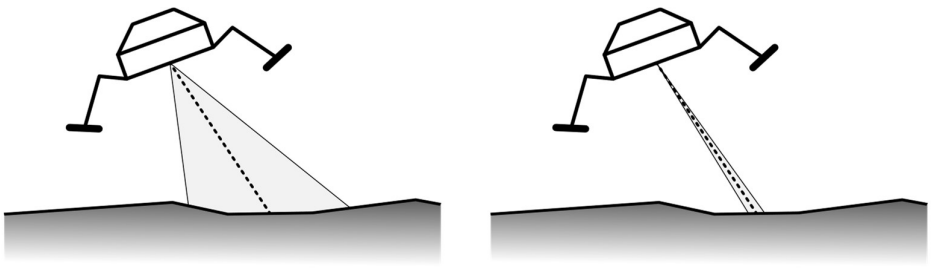

Figure 2. Lidar Sensor Scanning Modes

For the most accurate results, the filter's altimeter measurement should include a terrain elevation model of the approximate landing region, to account for the local variations the altimeter will observe. Implementation of this feature is left for a future phase of the project.

4 IMAGE PROCESSING HYBRID IMPLEMENTATION

The Q7 features a Xilinx Zynq-7020 SOC comprised of a dual-core Advanced RISC Machine (ARM) Cortex-A9 Processing System (PS) and an Artix-7 Programmable Logic (PL). Multiple Advanced eXtensible Interfaces (AXI) allows for communication between the PS and the PL. Each interface is 64 bits wide and can perform one transfer operation per PL clock cycle. Four high performance AXI allow the PL to read and write directly to Random Access Memory (RAM). One Accelerator Coherency Port (ACP) is routed through the ARM Snoop Control Unit (SCU), allowing the PL to perform cache coherent memory operations.

Each image processing step suitable for execution in PL was synthetized in an accelerator from C code using Xilinx Vivado High Level Synthesis software suite. Boundaries were placed between accelerators where data access by the PS was required. Shared memory buffers, allocated using a custom Linux kernel driver, allowed the PS to perform memory operation in the hardware address space used by the PL. To reduce memory operations overhead time, use of the ACP was prioritized to reduce both the need to flush the PL cache to RAM and the need to invalidate PL cache lines. Memory access by the PL were also minimized by employing local buffers synthetized in logic and by rewriting the image processing algorithm to process pixels in a streaming fashion. This way, each accelerator can process around one pixel per PL clock cycle.

Fixed point operations were used when operating on pixel intensities, to reduce the required logic area and improve loop pipelining. Fixed point intensities were also packed in wider datatypes at the boundary of the PL accelerators to maximize the transfer rate of the AXI and ACP. Image coordinates operations were kept in floating point because the algorithm reliance on sub-pixel normalized coordinates would have required a long implementation time.

4.1 Image Processing Algorithm and Accelerators

The image processing algorithm used for detecting and tracking features between captured camera images, introduced in [5], is based on the Lucas-Kanade optical flow algorithm. It uses the expected camera motion estimated by the navigation filter and epipolar geometry as a heuristic to increase the algorithm robustness to large feature displacement without relying on pyramidal image scaling.

Figure 3 presents the execution flow of the algorithm and whether a processing step is performed by a PS function or a PL accelerator. The main steps are:

1. PS – Receive the input image through a TCP socket.
2. PL – Blur the input image and compute its horizontal and vertical gradients. Gradients computing is performed in the same accelerator allowing the computation to start as soon as enough blurred lines are available in a logic buffer.
3. PL – Compute the Harris response from the processed image. PS cache must be invalidated after this operation as the new Harris response is written directly to RAM, thus the PS could fetch stale data from its caches.
4. PL – Correlate the features descriptor with the processed image in the vicinity of the predicted feature locations based on epipolar constraint and spacecraft estimated states.
5. PS – Resolve the features displacement using the correlation values.

6. PS – Clear lost features and mark features whose descriptor require an update using the correlation values.
7. PS – Replace lost features by detecting new ones using the Harris response.
8. PS – Update or create features descriptor from the processed image in preparation for the next iteration. PS caches must be flushed to RAM after this operation as the PL reads the descriptors directly from RAM, thus could fetch stale data if the caches have not yet been written to RAM.

It is important to note that the PL accelerators are independent. Thus, as the Harris response uses the processed image and is not needed until the feature detection, it runs alongside the feature tracking operations. This means that if the feature list is not empty, the Harris response operation is a zero-time operation.

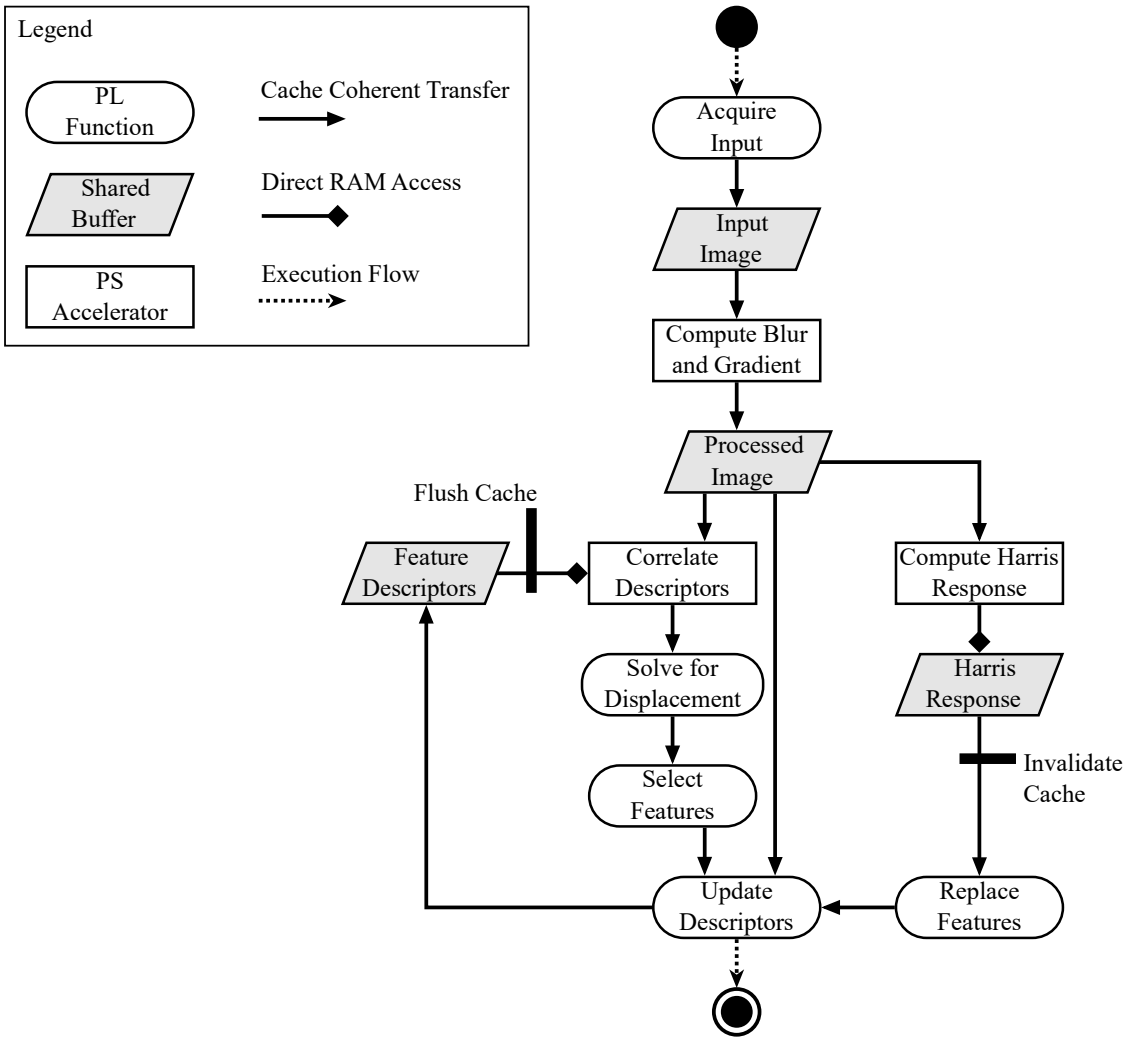

Figure 3. Image Processing Execution Flow

4.2 Profiling Results

Table 1 details the execution time of the image processing accelerators. Total loop time went from 366 ms to 103 ms, which meets the target execution rate required for accurate navigation at higher angular rates. As expected, blurring, gradient, and Harris map all benefited from implementation in logic. It is worth repeating that the Harris map computation time is only blocking on the first run of

the algorithm, when the feature list is empty. It is computed alongside the feature displacement in later runs; thus, the slower tracking operation bounds the execution time.

Table 1: Image Processing Steps Execution Time on the Q7

Step	PS Computation Time	PL Computation Time
Blurring and Gradient	113 ms	17 ms
Harris Map (first iteration)	170 ms	14 ms
Descriptor Correlation (one call)	0.066 ms	0.082 ms
Feature Displacement (all calls)	61 ms	65 ms
Total Loop Time	366 ms	103 ms

Despite the streaming nature of the operation, correlating the feature descriptors with the image did not yield an improvement in runtime. This is likely due to the small size of the descriptor and to the random access required in the image as only fractions of image lines were red.

5 FULL-SCALE DEMONSTRATION on UAV

Figure 4. DJI Matrice 600

Testing in LDTF poses scaling problems that prevent full validation of a Moon landing navigation and site assessment system. Range measurement noise does not scale down, preventing the obtention of quantitative results. Thus, it is desirable to perform full scale descent trajectories in a controlled environment. A DJI Matrice 600 rotary wings UAV, shown in figure 4, was chosen due to its availability at NGC and its payload capacity. It will perform descent trajectories in open loop fashion while carrying the navigation and site assessment system.

Sensors and computers were mounted on a custom 3D printed payload frame, resulting in the assembly shown in figure 5. Pictured is the Ouster Lidar used for preliminary testing pending integration of the mini-WABS sensor. The payload also contains a battery for powering the components independently from the UAV.

Two components are added to the system to support the tests: 1) a Microhard ethernet over the air modem for communication between the ground control station and the payload; 2) a VectorNav GNSS/INS unit to provide the ground truth trajectory from a GPS RTK. To control the landing site safety, a flat flight site was selected. Thus, artificial roughness and slope can be added in a controlled fashion by adding obstacles.

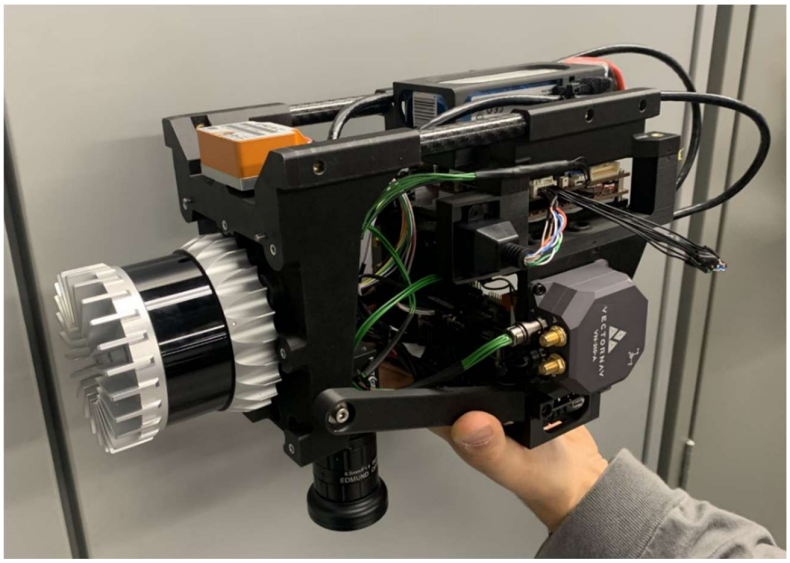

Figure 5. Sensor and Processing Payload

Prior to the flight campaign, the UAV payload mounted was mounted on the robot in NGC LDTF. Figure 6 shows the assembly. This setup allows validation of the correct interaction between the different processes and computers. It will also provide confirmation of the proper performance of the newly implemented algorithms. This will greatly reduce the required flight time, as the system will already be known to be operating correctly prior to the operations at the UAV flight test site.

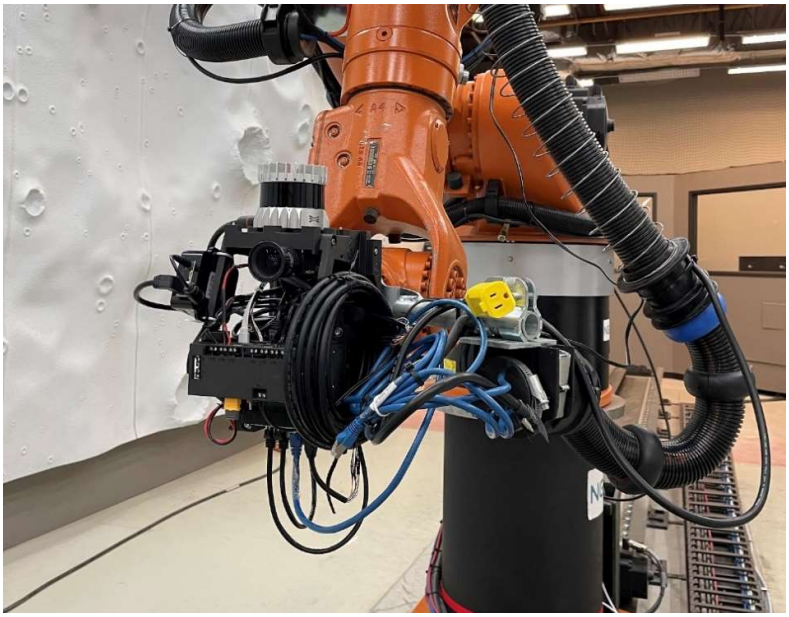

Figure 6. Sensor and Processing Payload Mounted on the Robot

6 CONCLUSION

HDA terrain relative navigation are key technologies to enable global access to the moon. They require close integration to compensate for the spacecraft's motion during its descent. HIL testing uncovered some performance issue when deploying an integrated system in a LDTF which were addressed in the last year. First a new navigation filter image processing measurement update

formulation was devised to avoid singularity when little to no motion along the camera boresight axis is present. Second, an altitude update using the Lidar sensor used by the HDA was added to resolve the scale of the motion observed by the monocular camera system. Third, some steps of the image processing algorithm were synthesized as PL accelerators to run in real time on the flight computer. The system is currently mounted in the LDTF to confirm its proper operation before moving on to a full-scale test campaign using a UAV which is expected to take place over the summer of 2023.

7 Acknowledgements

We acknowledge the support of the Canadian Space Agency (CSA) [20STDPO01].

8 References

[1] Sobiesiak L.A., et al. *Autonomous Moon Landing Guidance, Navigation and Control Systems Development and Validation,* International Conference on Guidance, Navigation and Control, Online, 2021.

[2] Hamel J.-F., et al. Hazard Detection & Avoidance Integration and Demonstration for Autonomous Moon Landing, International Astronomical Congress, Paris, France, 2022.

[3] https://xiphos.com/wp-content/uploads/2015/06/XTI-2001-2020-f-Q7S-Spec-Sheet.pdf

[4] https://sensonor.azurewebsites.net/media/q2cej5tw/ts1657-r11-datasheet-stim318.pdf

[5] Simard-Bilodeau V., Autonomous Vision-Based Terrain-Relative Navigation for Planetary Exploration, Ph.D. Thesis, Université de Sherbrooke, 2015.

[6] Nistér D., *An Efficient Solution to the Five-Point Relative Pose Problem,* IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 26, no. 6, pp. 756-770, Princeton, NJ, USA, 2004.