DOI: 10.13009/EUCASS2023-995
Aerospace Europe Conference 2023 – 10TH EUCASS – 9TH CEAS

Artificial Intelligence in Planetary Exploration: Enabling Autonomous Decision-Making for Spacecraft

Anton B. Ivanov[⋆†] *and Alexey Simonov*[⋆] *and Nidhish Raj*[⋆]
[⋆]*Propulsion and Space Research Center, Technology Innovation Institute*
Masdar City, Abu Dhabi, United Arab Emirates
anton.ivanov@tii.ae · alexey.simonov@tii.ae · nidhish.raj@tii.ae
[†]Corresponding author

Abstract

Artificial Intelligence and Machine Learning (AI/ML) technologies have the potential to revolutionize the field of planetary exploration by enabling spacecraft to make autonomous decisions in real-time. Our work will concentrate on creating an autonomous system for planetary navigation. We are preparing a demonstration rover platform for planetary navigation. Experimental architectures will be based around a specialized processor with additional computational modules to enable fast AI processing. The platform will be equipped with a simple stereo camera setup with an IMU and a sun-sensor. The goal of the research is to identify the most effective combination of sensor, algorithms, and computing platform to enable autonomous traverse. The objectives will be to minimize the power consumption, enable operations during minimal light conditions and achieve faster traverse speeds, compared to state-of-the-art today. This work presents state-of-the-art review and some initial suggestions for implementation.

1. Introduction

While much focus has been on Mars as the next frontier for human colonisation, it is imperative not to overlook the Moon's significance in our cosmic endeavours. Lunar exploration serves as a stepping stone towards the colonisation of the red planet, offering a testbed for cutting-edge technologies and enabling valuable insights into the challenges of future space missions, as well as increasing our understanding of physical and geological processes on the Moon. One technological frontier is the development of architectures for autonomous driving on the Moon, which represents a crucial aspect of lunar surface exploration. The Apollo program's Lunar Buggy, also known as the Lunar Roving Vehicle (LRV), was a remarkable innovation that revolutionised lunar surface exploration during the Apollo missions. Its deployment on Apollo 15, 16, and 17 missions between 1971 and 1972 greatly extended the range of exploration, allowing astronauts to venture further from the lunar module and collect invaluable geological samples. On the other hand, the Soviet Union's Lunokhod program, spanning from 1970 to 1973, featured the Lunokhod rovers, robotic vehicles that explored the lunar surface autonomously. Lunokhod 1 and Lunokhod 2 were the two successful missions, during which these six-wheeled rovers traveled significant distances, conducted experiments, and transmitted valuable data back to Earth.

In the past two decades, a number of robotic missions have been deployed to Mars, including the remarkable Mars rovers such as Sojourner, Spirit, Opportunity, Curiosity, and Perseverance. While these rovers were mostly tele-operated, recent software upgrade to Curiosity and Perseverance enable some degree of autonomous driving and obstacle avoidance.

The Chang'e program, which started in 2010, has included multiple orbiters, landers, and rovers. Chang'e 3 (2013) and Chang'e 4 (2019) successfully landed on the lunar surface. Yutu and Yutu-2 rovers were successfully tele-operated on the lunar surface.

In this work, we aim to explore the recent advancements in architectures for autonomous driving on the Moon. We will examine the challenges specific to lunar surface exploration, the key components of autonomous driving systems, and the integration of artificial intelligence into the platform, while imposing important constraints such as power consumption. We will briefly review the state-of-the-art in Chapter 2. In Chapter 3, we summarise key requirements for a mission to the Moon, which may employ autonomous driving, enabled by AI and present some architectures in Chapter 4.

DOI: 10.13009/EUCASS2023-995

2. State of the art

2.1 Algorithms

Historically it was very limited autonomy that was implemented on robots for ground exploration of celestial bodies. This was in large part due to scarce compute capabilities available on-board of those vehicles. With very limited downlink capability (on the order of 60MB/sol for Curiosity rover) the lack of autonomy moves space exploration literally at snail's pace because humans-in-the-loop have to receive the data and make decisions turn-by-turn.

At the same time the autonomous driving problem is very well studied and solved to a good degree on Earth over the last decade. Now that we have more capable space computers we are looking to move some of these algorithms to space.

An example of classical path planning approach can be seen in [1]. JAXA researchers use the grid based Dijkstra's algorithm for proposed Moon rover. They take into account Digital Elevation Model of the environment (DME). The cost function has terms for distance travelled as well as rover-related costs and environment-related costs. Rover-related costs depend on type of locomotion system (wheels or tracks) and estimated slippage due to slope of the grid elements. External costs include illumination of solar panels that depends on roll/pitch of rover, elevation of the sun and shadows from surrounding terrain. However when authors combine the elements of the cost function they rely on experience and intuition rather than machine learning methods.

In [2], another classical approach, researchers developed a tool for optimal traverse planning for polar regions of the Moon. They use datasets from LRO Camera, Lunar Orbiter Laser Altimeter, Diviner Lunar Radiometer, as well as a fundamental wheel-regolith interaction model and terramechanics model. This tool allows to plan for least-energy traverses maximising illumination. This tool is useful for high-level mission planning, but the map resolution is not sufficient to plan exact trajectories to execute on a small rover.

The Chinese researchers working on Yutu-2 rover described in detail their approach to visual localisation in [3]. The localisation (or more generally state estimation) is a critical subsystem for autonomous driving. As there is no external navigation infrastructure on the Moon (GPS, beacons, etc) the only feasible solution is visual navigation. In the course of visual navigation the robot is trying to infer its self-motion from a sequence of camera images looking at the environment. There are multiple SOTA algorithms to do it. All of them require a lot of compute resources to work reliably well. This is due to high information density of the images and amount of per-pixel processing involved. For example Yutu-2 stereo navigation cameras have resolution of 1024x1024 pixels. The visual navigation also uses IMU measurements, but Yutu-2 mission does not use wheel odometry due to slippage of wheels that is hard to measure. In case of Yutu-2 the localisation was not run on-board, but the images taken approximately every 7m were sent to ground control. The ground teleoperation system used 64-core workstation to process SLAM (bundle adjustment) computations. Due to teleoperation Yutu-2 is quite slow - it covered 120m in first lunar day of operations and over 4 earth years of operation it travelled 1455m total. Even though this was not a truly autonomous visual odometry based driving, it is a good step towards proving that this approach works on the Moon.

To the best of our knowledge [4] is first full scale attempt to apply AI and ML to space autonomy. The authors propose machine-learning capabilities for future NASA's Mars Sample Fetch Rover assuming that NASA HPSC program [5] delivers 100x compute power. The mission main goal is to drive and collect pre-deposited samples, but with extra autonomy capabilities it can do 'drive-by science' as a background task to its main mission. It would automatically select interesting artefacts and highlight them to scientists on Earth while driving autonomously past them. Another use of machine learning here is use of compressed feature representation of images to communicate data back to earth. Yet another way to utilize AI on this mission is using risk- and resource-aware autonomous navigation where a neural network is optimizing the long-range driving path with diverse objectives, such as driving energy minimization, science gain maximization while keeping safety within limits. The authors trained 2 neural networks based on historical Mars datasets, one estimating power required for driving based on input image, another estimating terramechanics and power required based on input image. They also propose networks for estimating probabilities of collisions, and for kinematic settling (minimize contact between terrain and wheels). In summary, with enough available compute the opportunities are truly endless. What is notable in this paper is that authors leverage Robotic Operating System [6], an open source software ecosystem that became de-facto robotics 'lingua franca' in scientific community. It enables modularity of the resulting system and allows one to combine the best algorithms developed by leading robotics labs to solve a particular use-case.

2.2 Hardware

The state of the art in space robotics compute hardware right now is usually a System-on-Chip (SoC) that combines at least some CPU cores, as well as a re-programmable hardware accelerator(s) such as FPGA or GPU, along with

DOI: 10.13009/EUCASS2023-995

optional extra specialised compute units such as DSP etc. The industry is moving away from custom designed IC and ASICs due to long development cycle times, high costs and elevated integration risks.

Two general trends are: 1) re-use of COTS (commercial off-the-shelf) electronic components, with some extra certification for space applications, or 2) custom design SoC based on open architecture, open hardware and open software/firmware. An example of 1 is Mars Ingenuity helicopter that uses Qualcomm Snapdragon SoC. In another instance, the recent Phisat-1 mission evaluated the use of COTS Intel Myriad 2 VPU for cloud detection on a cubesat. An example of 2 is NASA's High Performance Spaceflight Computing (HPSC) program [5] that is developing a general space computer with 100x compute power compared to the currently deployed units.

The computer hardware for the lunar rover can take inspiration from the recent Mars helicopter, Ingenuity. The robotic helicopter was envisaged as a technology demonstrator with a planned duration of 30 sols on Mars and 5 total flights. However, Ingenuity has far outperformed its original mission objective and accumulated 37 flights and survived 663 sols (682 days) as of January 1, 2023. The hardware architecture of Ingenuity [7] is shown in Fig 1. It uses a radiation tolerant FPGA (Microchip ProASIC3) acting as a safety-critical communication hub connecting the sensors, actuators, flight control computer (FCC) and the navigation computer. The FCC is responsible for time critical inner control loop (actuators, attitude control), the trajectory control, and the state-machine of the helicopter. It consists of dual-redundant dual-core Arm Cortex-R5 microcontrollers (TI TMS570 Hercules). The cores of individual microcontrollers work in lock-step which helps in detecting radiation induced upsets. In such a scenario the FPGA switches to the spare microcontroller. The Navigation computer does the state estimation by using camera images and IMU data. A quad-core Qualcomm Snapdragon 801 SoC is used on the Navigation computer, which is connected to the downward facing monochrome Navigation camera. Image processing and navigation filter are assigned to separate individual CPU cores and the remaining cores are assigned to less time-critical tasks such as communication, logging, command handling.

The NASA work [4] summarised in previous section had the resulting algorithms implemented and tested on Athena Rover that has NVIDIA TX2 SoC. But in addition to this the toolchain was created to deploy the same algorithms on HPSC or Snapdragon platforms (820 and 855).

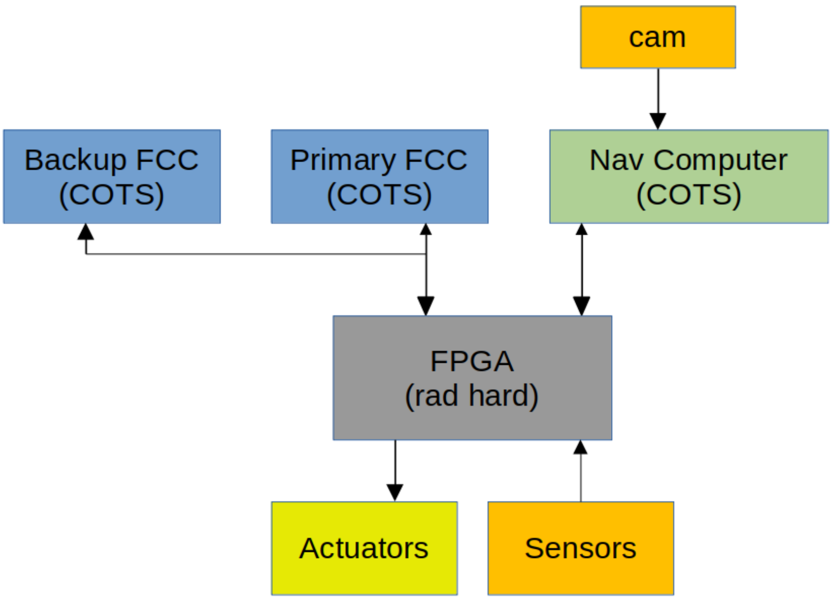

Figure 1: Hardware architecture of the Mars 2020 helicopter Ingenuity taken from [7]. The most important innovation here is that commercial off-the shelf electronics was used on this mission, with the exception of the rad-hard FPGA. This hybrid architecture paves way for much more powerful computers to be flown on future missions.

Alternatively the algorithms can be implemented on space grade processors and FPGAs. Collections of radiation hardened by design (RHBD) and radiation tolerant (RT) FPGAs and microprocessors from major players have been identified and provided in the following table.

DOI: 10.13009/EUCASS2023-995

Name	Vendor	Type	Rad.	LUTs/Freq
GR740	Cobham Gaisler	Multicore CPU	QML-V	250 MHz
PC8548	Teledyne e2v	Multicore CPU	QML-Y	1200 MHz
RAD5545	BAE Systems	Multicore CPU	QML-V	466 MHz
HPDP	ISD / Airbus	Multicore CPU	ECSS	250 MHz
RC64	Ramon Space	Multicore CPU	MIL	130 MHz
Kintex Ultrascale	Xilinx	FPGA	RT	331,680
Virtex-5QV	Xilinx	FPGA	RHBD	131,072
ProASIC3-RT	Microchip	FPGA	RT	35,000
NG-Ultra	NanoXplore	FPGA	RHBD	536,928

2.3 Rover Platform

Even though the actual rover platform and its capabilities are absolutely crucial to satisfy the missing requirements, the goal of this paper is to address the concept for autonomous driving. That will be refined later to reflect the interfaces with the platform, available power budget, moving capabilities etc.

An example of relevant platform, along with the open datasets simulating lunar environment on Earth, can be found in [8]. These results can be used to kick-start the system and software development until the target platform is specified in sufficient detail.

3. Mission description

3.1 Use case description

For this work, we assume that the next generation of lunar rovers will perform the following functions (also see (Figure 2):

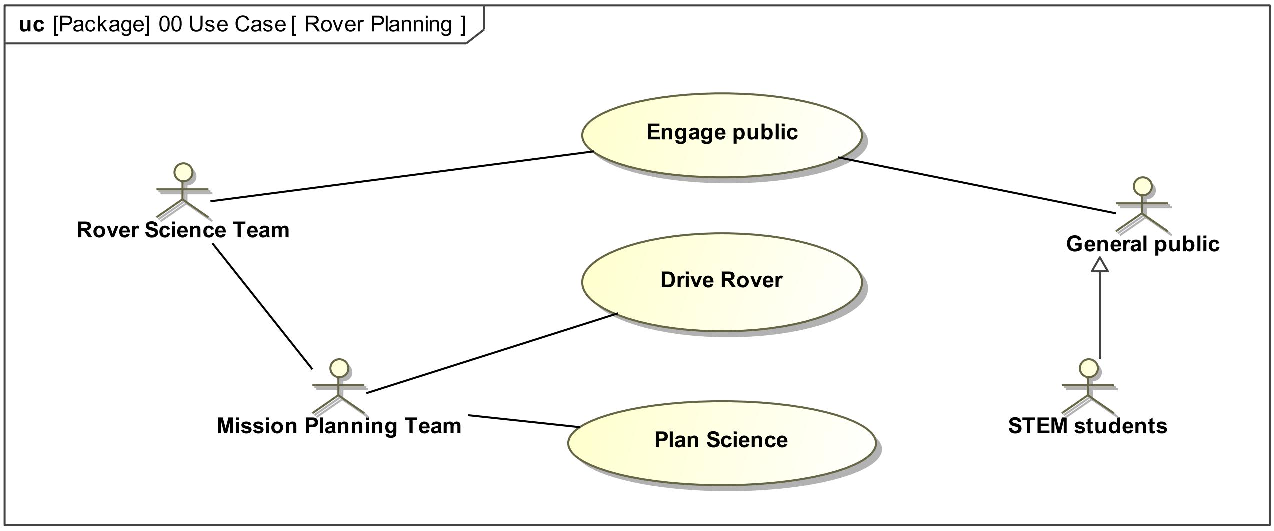

Figure 2: Use case diagram for the Lunar Rover project. The main executing actors are the Science and Mission Planning teams. They are working to benefit general public by releasing science results and media materials.

- Autonomous driving on the Moon's surface: 100km during duration of the mission.
- Send live video stream from the Moon during driving to engage general public
- Open-source some simulation, software, tutorials and collected data to motivate STEM students to chose space as career
- contribute some code and hardware designs to Robotic Operating System community to facilitate international collaboration with the leading space and robotics teams

DOI: 10.13009/EUCASS2023-995

We also assume that the main goal is to collect information about water ice deposits in the South Pole area of the Moon. In order to set autonomy requirements, we propose to beat current records for daily driving and total distance. Science investigation is beyond the scope of this work, but we assume that sample collection and analysis of subsurface soil properties will be the main objectives.

Autonomy requirements will be mapped into hardware and software architecture of the mission in sections below. We note, that this is concept work, without iterations with the rover platform team, which will impose strong driving and power budget requirements.

3.2 Concept of operations

The science operations team will plan weekly scenarios for exploration, based on digital elevations maps of the Moon from LRO (about 1m/pixel resolution), and after examining returned data from beginning of this mission. The Missions Operations team will schedule driving sessions on daily basis, after evaluation of the previous day performance. Autonomous driving will allow operations when the rover is out of communication with ground stations on Earth. In order to set some requirements for our missions, we propose to set a world record in autonomous driving on a celestial body other than Earth of 100km (current record is 45km, but it is likely that Curiosity and Perseverance rovers will beat it).

We assume a simple and straightforward concept of operations for this mission. Geometry of Earth-Moon system allows daily contacts with a selected ground station (given that it is located on the Earth-facing part of the Moon). Our simple evaluation of a scenario is shown in Fig. 3.

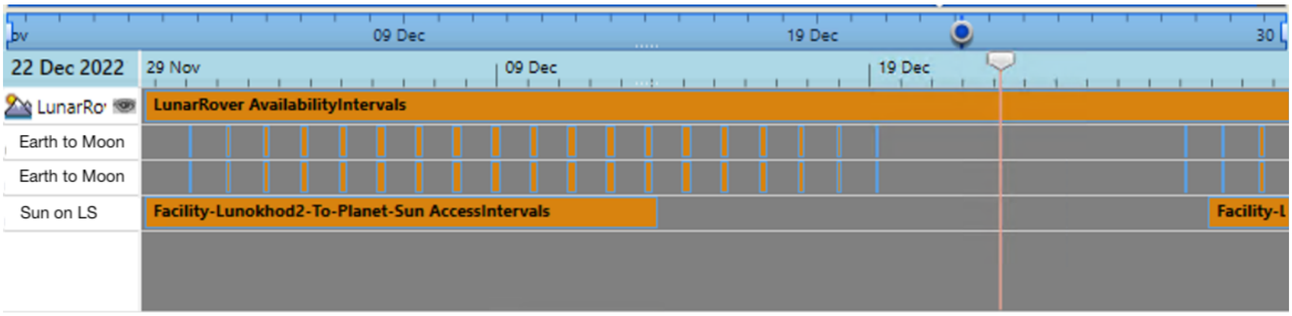

Figure 3: Example scenario (from Nov 29 to Dec 29, 2022) for the Earth-Moon system with the lander near the Lunokhod-2 site and Earth ground station located in the UAE. First line corresponds to ground contacts (seen as thin lines of 5 hours) and second line is when the landing site is sun-lit.

In this particular scenario, our lunar landing site is located near the Lunokhod-2 rover, giving a possibility to investigate why it failed so early in its mission. Ground station is located somewhere in the UAE. Estimating link budgets and details of ground station operations are out of the scope of this study. In the example, we study contact times in the month of November 2022. Sun will be above ground stating on December 7th until 20th December. Details of the energy budget are also out of scope of this study, but probably some time will be needed after lunar sun-rise to recharge the batteries, so not a full 13 days will be available for driving. We have also introduced a severe constraint on the ground station that it is only capable of contacting the rover, when local elevation is above 50 degrees. This resulted in daily 5 hour windows of connection. This time will be changing throughout the year, depending on Earth-Moon position. During these windows we can enable live feed of driving to the ground station. We assume that the rover will not have a direct-to-Earth link and will rely either on the lander capabilities or a telecommunication orbiter. Elevation of the Earth GS will vary between 40 and 60 degree on the rover site during this month.

3.3 Key Requirements

The key driving requirement for this mission is to demonstrate advanced autonomy on the lunar surface, while engaging wide public into the Moon exploration.

Autonomy The rover shall have capability to plan routes autonomously without interference of Mission Control.

Rover Speed The rover shall move at the speed of 1 km per hour [Note: Lunar Rover Vehicle maximum speed was about 12.8km/h (Apollo 15) and 17.7km/h (Apollo 16/17)]

DOI: 10.13009/EUCASS2023-995

Daily driving distance The rover should cover a distance of 500 m per day autonomously (Rationale: to beat , current record of Perseverance: 245 m (February 2022))

Daily mission duration 30 minutes.

Mission Duration The rover should cover at least 100km on the Moon (current record Opportunity: 45km Driving records as of February of 2019). This means about 7 km per 14 days lunar cycle, therefore 14 cycles are needed, therefore total mission duration in Earth days: 392 days (maybe shorter if possible to drive faster)

Public outreach The rover shall transmit live video stream during driving via a relay station.

Science requirements The rover shall collect imaging data. Possibly carry some other simple instrumentation: Radiation sensor, magnetic sensor (to look for ancient magnetic field), monitor flashes in the area (while not driving).

Environmental requirements The rover shall withstand lunar environmental conditions

3.4 Environmental requirements

Radiation environment The electronics has to be designed assuming Total Ionising and Non-Ionising Dose, Single Event Effects (SEE) and Internal Charging (DDC) for a 1 Earth year mission on the Moon.

Dust Properties Described in [9]. Further specification needed for electronics and sensors protection.

Soil Properties Described in [10]. Further specification is out of scope of this paper.

Gravity The rover should operate accounting for acceleration due to gravity of 1.625 m/s^2

4. Proposed architectures

4.1 Algorithms

We propose the autonomy software stack to include modules for: state estimation, local mapping, trajectory planning, control/actuation, supervision. These modules would be deployed on the rover. In addition to that we would have global map and global planner run at ground control. The summary of proposed architecture is outlined in figure 4

For State Estimation we propose to use visual-inertial odometry (VIO). It will use visual observations from navigation stereo camera and inertial measurements from an IMU (gyroscope and accelerometer readings). It will output 6DoF pose as well as linear and angular velocities at the rate sufficient to drive at the speed we target given the computational constraints. The VIO problem is well studied in academia and there are a few robust open source implementations that we can take as baseline in our work. In particular we found OpenVINS [11] framework accurate, fast and not very resource intensive. Alternatively, Basalt VIO [12] offers even better accuracy, but at the expense of higher resource usage. We may also consider Granit [13], a derivative of Basalt developed by German DLR for monocular camera case and highly repetitive textures representative of lunar landscapes. We will calibrate camera-IMU system on earth and expect it to be sufficient, as long as we have calibration results for different temperatures in our operating range. The calibration will be verified from time to time on the Earth by running full bundle adjustment for some representative amounts of data that we would downlink as required.

The Local Mapping module is based on 3D reconstruction from stereo cameras. We propose to use a variation of Semi-Global Matching (SGM) [14] to build stereo disparity image. This algorithm became a de-facto standard for stereo 3D reconstruction in robotics and is easy to implement on specialised hardware like GPU or FPGA [15]. The disparity image from SGM is easily cast into a point cloud which we register in the map using the 6DoF pose from the State Estimation module. We can take Octomap ([16]) as base mapping implementation, but most likely we will optimize both implementation and final map representation to fit our constraints. Final map will be a simple geometric 3D occupancy map with resolution of <5cm. The target range of reliable 3D reconstruction from stereo cameras should be about 5m (depends on baseline length), effectively covering driving distance of 17 seconds at maximum speed of 0.28 m/s. If 3D reconstruction on-board is proven to be too costly computationally we should consider alternative sensors that give 3D information directly, such as ToF sensors or COTS solid-state LIDARs as in recent mobile phones.

The Mapping Module will be enhanced with semantic information data relevant to generating fast and safe driving plans. The semantic information can be inferred from camera images using AI semantic segmentation and may include 'drivable surface', 'small rock', 'big rock', 'crater', 'unknown' classes etc. We will curate a dataset of lunar images we expect to see and label them with the target classes. The training would be done on Earth and the trained network would have to be optimised for our chosen hardware before deployment on the rover.

DOI: 10.13009/EUCASS2023-995

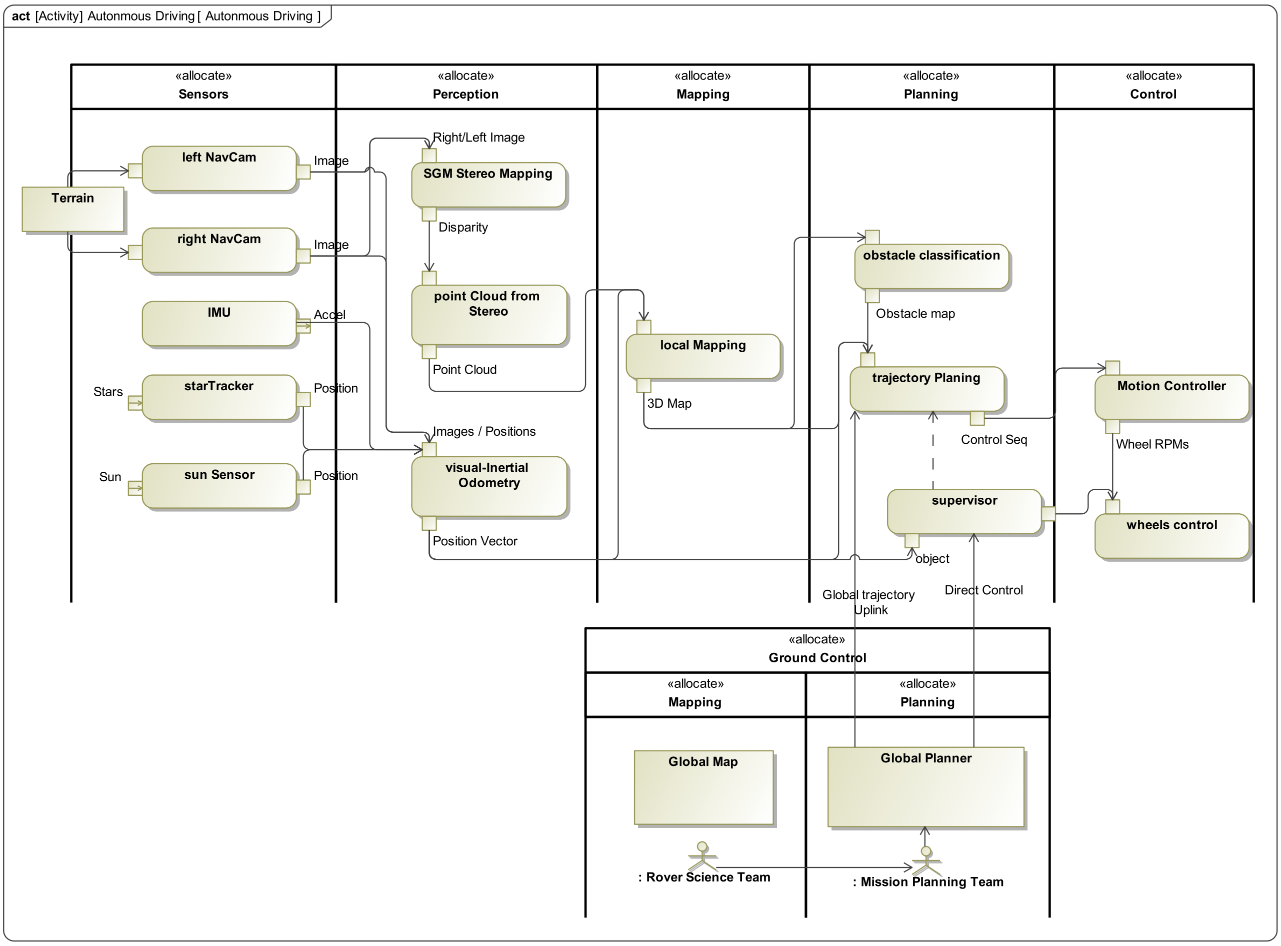

Figure 4: Sensors and Algorithms Architecture for the proposed solution. The rover will take information from local terrain, wheels encoding, Sun position and, possibly, stars (TBD). Sequence of algorithms will be run to map surroundings, identify position, classify obstacles and plan trajectory. The next step will be to estimate computational complexity and whether this solution can be accommodated on the on-board computing hardware. Ground control only has responsibility for Global route planning.

Global path planning will be done daily at ground control to determine high-level route (sequence of waypoints) for the day. It will be planned on a wide area digital elevation map which should be generated offline for the mission using the LRO images of the exploration area. The global map resolution is around 1m per grid cell. The resulting route will be uplinked to the rover daily.

The Trajectory Planning module will use the local map built on-board, along with the daily route sent from the ground control, and the current position from the State Estimation module. It would produce trajectory (a sequence of target x,y coordinates along with timestamps), covering next few seconds of driving. The initial trajectory planning will be done by solving optimization problem that would target: speed of driving, obstacle avoidance, safety metrics such as max roll/pitch, feasibility of the trajectory given kinematic and actuation constraints, as well as required power. It may also optimise for power generation from solar panels. The trajectory re-planning will be done at 1Hz.

As a baseline we would use the trajectory planning approach described in [17]. After first few weeks of operation with availability of actual driving data we may train a neural network for trajectory planning and try it instead of optimization-based planner as in ([18]). Depending on available resources we may try to run both optimization-based and neural network planners in parallel on-board to verify each other predictions.

The Motion Controller module will take the planned trajectory and the current pose and generate actual control commands to apply to the driving system to execute the trajectory. For example for a differential 4 wheel driving platform it would command each wheel to rotate with certain angular velocity. Because all the complex logic of planning a trajectory is implemented in the Trajectory Planner we propose to use a simple model-free controller here such as PIDE, which should be sufficient. The actual commands are executed by the Wheels Controller that is sending the signals to electric motors.

DOI: 10.13009/EUCASS2023-995

As an alternative to separate Trajectory Planner and PIDE Controller we may investigate Model Predictive Control. Given approximate model of the rover kinematics, soil properties, state of power system, electric drive characteristics etc it would re-plan the control sequence on receding time horizon for next few seconds to achieve tracking of the desired trajectory. A good description of this approach is found in [19]. It should be noted that this is a more computationally intensive approach that is usually used for highly agile systems with highly non-linear dynamics. It is also possible to use recent advances in Neural Network Model Predictive Control (NNMPC) [20] to control the rover.

The Supervision module would be an independent module monitoring the entire system and interfering if it detects anomalies or takes an external command from the mission control team. Use of AI to detect faults in space hardware is a developing field. Typically faults are very rare in spacecraft and there is not enough statistics to identify faulty behaviour. However, we plan to conduct some experiments with spacecraft modeling to understand whether AI approaches can be used for Fault Detection Isolation and Recovery (FDIR). Typically this is achieved by generating labelled training AI data using a software model of spacecraft operation and injecting faults in it.

4.2 Hardware

The hardware of our autonomous driving system will include navigation sensors and navigation computer.

For the sensors we propose to have a stereo navigation camera and inertial measurement units (IMUs). For redundancy we plan to have 3 image sensors mounted on a line, with focal planes aligned. If any one of them fails we can still operate remaining 2 as a stereo camera pair.

We look to use COTS global shutter cameras because they have constantly improved over several decades and the available open-source algorithms are easy to use, modify and test. The global shutter camera disadvantages are power consumption and low dynamic range. If power budget does not allow for stereo cameras we can consider monocular state estimation. It is less resource intense, but has more uncertainty about metric scale.

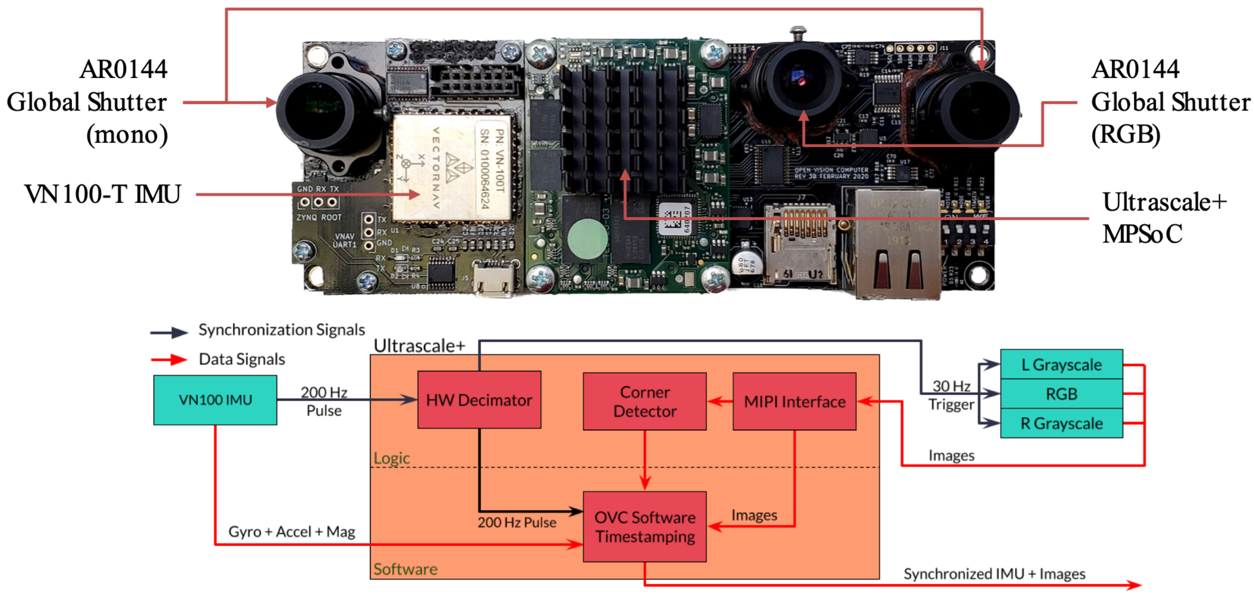

Figure 5: OVC rev.3B appearance and architecture from [21]

As the Moon does not have atmosphere and the operation will be during the lunar day we expect to encounter high dynamic range (HDR) scenes of the landscape, i.e. there will be very bright and pitch dark areas in the same image. This makes it challenging to find single exposure setting for the entire image. The problem is especially pronounced at the polar regions as the Sun elevation over the lunar horizon is low which makes shadows a significant part of the picture field of view. If global shutter cameras are found inadequate for the task we propose to look at event cameras (also known as neuromorphic cameras or dynamic vision sensors). This is a new technology that emerged only within the last decade. It is still in heavy research phase, especially the algorithms. However, there are recent publications about event cameras suitability for planetary missions, such as [22] from NASA JPL.

We also plan to have 2 identical IMUs for FDIR purposes. IMU should be tactical grade MEMS sensors, such as VectorNav VN-100, temperature calibrated for the entire operations temperature range.

One option for the computer on the target rover hardware can be a Zync Untrascale+ or 7000 or Kintex UltraScale MPSoC from Xilinx combining ARM CPU cores and FPGA fabric. The use of FPGA allows best tradeoff between

DOI: 10.13009/EUCASS2023-995

performance and re-programming of the system after deployment. The updated FPGA logic configuration can be tested on Earth and uplinked to the rover. Also, use of FPGA fabric allows to duplicate IP cores for redundancy and run them in parallel. Xilinx line of products use ECC memory for single-bit error detection and correction. Also, in case of single event effects, especially latchups, the FPGA can be reconfigured to avoid the problem places in the fabric. In any case we will have to test the final hardware with the relevant radiation and temperature exposures.

Another option is to use emerging RISC-V based CPUs with FPGA fabric on the same SOC, for example Microchip Polarfire. The RISC-V architecture offers a compelling option for space missions due to its inherent advantages in reliability, flexibility, and openness. As we noted above, HPSC initiative have chosen RISC-V based CPU to power future NASA missions. RISC-V's simple and streamlined instruction set architecture reduces the complexity of hardware design, minimizing the potential for errors and enhancing fault tolerance. Additionally, the modular nature of RISC-V allows for customizable and scalable designs, enabling space missions to tailor the architecture to specific requirements, such as very low power consumption and utilization of AI/ML algorithms onboard.

Our future steps will concentrate on creating prototypes employing existing solutions (e.g. Zync Ultrascale+ based), as well as novel RISC-V based to trade-off for future lunar missions.

One of the motivating implementation examples, is the architecture of the Open Vision Computer (OVC) [23], in particular OVC revision 3b, developed by Open Robotics Foundation in collaboration with GRASP laboratory at UPenn. The hardware, software and firmware is provided open source and easy to manufacture and modify to our needs. The design and high level architecture of the OVC3b system is presented in figure 5. This is a single PCB carrier board integrating sensors (2 AR0144 global shutter mono cameras, 1 AR0144 RGB camera and VectorNav VN100-T IMU) and Xilinx UltraScale+ MPSoC XCZU4EV (Trenz TE0820). The device measures about 14cm x 5cm x 5cm and weighs about 200gr. The cameras are connected directly to FPGA fabric and the first steps in image processing happen with zero latency on FPGA, as the data flows to DRAM to be used by CPUs for further processing. Some parts of the VIO computer vision pipeline are implemented on FPGA. The IMU and cameras are hardware synchronised which is critical for accuracy of VIO localisation. We envisage to fully implement high-volume low-latency data processing steps (histogram equalization, image rectification, feature detection, image segmentation, stereo disparity etc) on FPGA and the higher level, more complex logic on CPUs.

Following examples presentated above one possible high level hardware architecture is presented in Fig. 6. We consider this solution a first iteration. This architecture will allow us to start developing the algorithm pipeline, while working on other hardware architectures. The next step is to procure a rover prototype platform to allow for testing in realistic conditions (e.g. UAE desert, similar to a lunar environment).

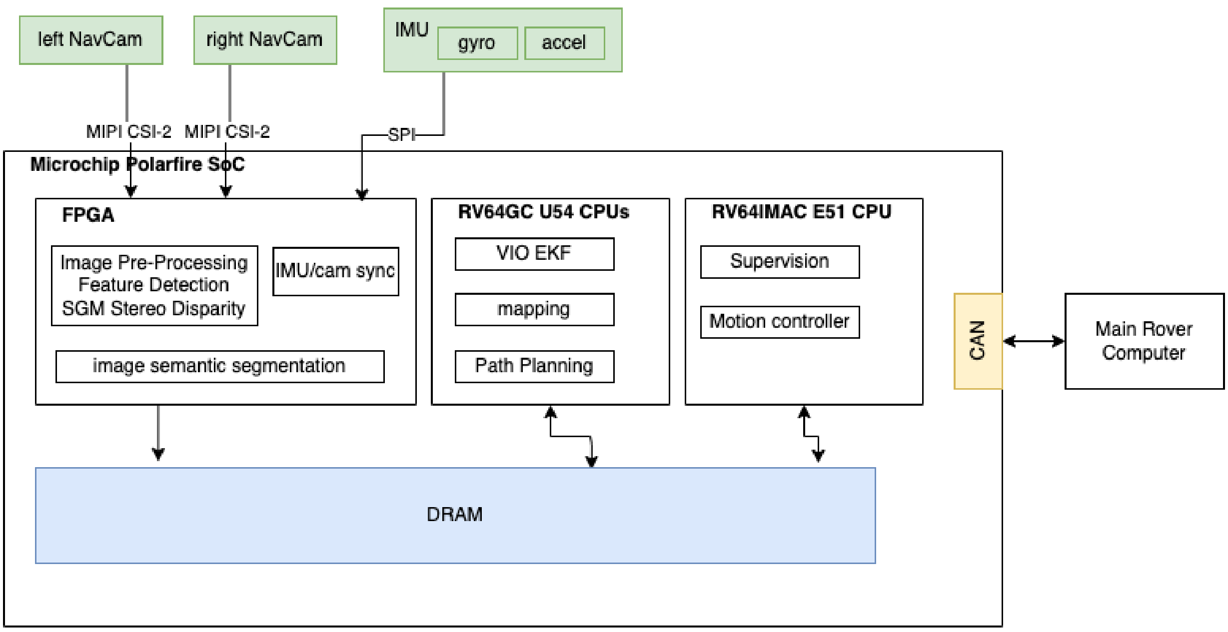

Figure 6: Proposed High-level Hardware Architecture for a Lunar autonomy subsystem. It will run the algorithms shown in 4

DOI: 10.13009/EUCASS2023-995

5. Conclusions

The integration of artificial intelligence (AI) in autonomous driving on the Moon represents a significant leap forward in planetary surface exploration. This paper has proposed software and hardware architectures that may power future autonomous lunar vehicles, highlighting their essential components and functionalities. The utilization of AI and machine learning (ML) methods in object classification will be instrumental in enabling rovers to accurately identify and analyze lunar features, enhancing scientific investigations. Moreover, the incorporation of AI-driven path planning algorithms will facilitate efficient navigation and optimize traversal across the challenging lunar terrain. Additionally, AI-based error detection mechanisms may increase the reliability and safety of autonomous systems, mitigating risks and improving mission success rates. We are now working on implementation of a prototype hardware architecture, as well as software pipeline, which will make use of the AI/ML methods. The combination of these advancements in AI and autonomous driving technologies paves the way for improved data analysis and longer traverses on the Moon, allowing scientists to collect more data than ever before.

References

[1] M. Sutoh, M. Otsuki, S. Wakabayashi, T. Hoshino, and T. Hashimoto. The right path: Comprehensive path planning for lunar exploration rovers. *IEEE Robotics and Automation Magazine*, 22(1):22–33, 2015.

[2] E.J. Speyerer, S.J. Lawrence, J.D. Stopar, P. Gläser, M.S. Robinson, and B.L. Jolliff. Optimized traverse planning for future polar prospectors based on lunar topography. *Icarus*, 273:337–345, 2016.

[3] Y. Ma, S. Liu, B. Sima, B. Wen, S. Peng, and Y. Jia. A precise visual localisation method for the chinese chang e-4 yutu-2 rover. *Photogrammetric Record*, 35(169):10–39, 2020.

[4] M. Ono, B. Rothrock, K. Otsu, S. Higa, Y. Iwashita, A. Didier, T. Islam, C. Laporte, V. Sun, K. Stack, J. Sawoniewicz, S. Daftry, V. Timmaraju, S. Sahnoune, C.A. Mattmann, O. Lamarre, S. Ghosh, D. Qiu, S. Nomura, H. Roy, H. Sarabu, G. Hedrick, L. Folsom, S. Suehr, and H. Park. Maars: Machine learning-based analytics for automated rover systems. 2020.

[5] Wesley Powell. NASA High Performance Spaceflight Computing. https://www.nasa.gov/directorates/spacetech/game_changing_development/projects/HPSC, 2022. [Online; accessed 03-Jan-2023].

[6] Steven Macenski, Tully Foote, Brian Gerkey, Chris Lalancette, and William Woodall. Robot operating system 2: Design, architecture, and uses in the wild. *Science Robotics*, 7(66):eabm6074, 2022.

[7] Håvard F Grip, Johnny Lam, David S Bayard, Dylan T Conway, Gurkirpal Singh, Roland Brockers, Jeff H Delaune, Larry H Matthies, Carlos Malpica, Travis L Brown, et al. Flight control system for nasa's mars helicopter. In *AIAA Scitech 2019 Forum*, page 1289, 2019.

[8] Olivier Lamarre, Oliver Limoyo, Filip Marić, and Jonathan Kelly. The Canadian Planetary Emulation Terrain Energy-Aware Rover Navigation Dataset. *The International Journal of Robotics Research*, 2020.

[9] A. V. Zakharov, L. M. Zelenyi, and S. I. Popel. Lunar dust: Properties and potential hazards. *Solar System Research*, 54(6):455–476, 2020. Cited By :12.

[10] A. T. Basilevsky, M. I. Malenkov, V. A. Volov, A. M. Abdrakhimov, N. A. Kozlova, A. E. Zubarev, and I. E. Nadezhdina. Estimation of the strength of the lunar soil by the depth of the lunar rover wheel tracks. *Solar System Research*, 55(4):285–308, 2021.

[11] Patrick Geneva, Kevin Eckenhoff, Woosik Lee, Yulin Yang, and Guoquan Huang. OpenVINS: A research platform for visual-inertial estimation. In *Proc. of the IEEE International Conference on Robotics and Automation*, Paris, France, 2020.

[12] Vladyslav Usenko, Nikolaus Demmel, David Schubert, Jörg Stückler, and Daniel Cremers. Visual-inertial mapping with non-linear factor recovery. *IEEE Robotics and Automation Letters*, 5(2):422–429, 2020.

[13] M. Wudenka, M. G. Müller, N. Demmel, A. Wedler, R. Triebel, D. Cremers, and W. Stürzl. Towards robust monocular visual odometry for flying robots on planetary missions. In *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 8737–8744, 2021.

DOI: 10.13009/EUCASS2023-995

[14] Heiko Hirschmüller. Stereo processing by semiglobal matching and mutual information. *IEEE transactions on pattern analysis and machine intelligence*, 30 2:328–41, 2008.

[15] D. Hernandez-Juarez, A. Chacón, A. Espinosa, D. Vázquez, J.C. Moure, and A.M. López. Embedded real-time stereo estimation via semi-global matching on the gpu. *Procedia Computer Science*, 80:143–153, 2016. International Conference on Computational Science 2016, ICCS 2016, 6-8 June 2016, San Diego, California, USA.

[16] Armin Hornung, Kai M. Wurm, Maren Bennewitz, Cyrill Stachniss, and Wolfram Burgard. OctoMap: An efficient probabilistic 3D mapping framework based on octrees. *Autonomous Robots*, 2013. Software available at `https://octomap.github.io`.

[17] Chris Cunningham, Joseph Amato, Heather L. Jones, and William L. Whittaker. Accelerating energy-aware spatiotemporal path planning for the lunar poles. In *2017 IEEE International Conference on Robotics and Automation (ICRA)*, pages 4399–4406, 2017.

[18] G. Sakayori and G. Ishigami. Energy-aware trajectory planning for planetary rovers. *Advanced Robotics*, 35(21-22):1302–1316, 2021.

[19] Tomáš Báča, Daniel Hert, Giuseppe Loianno, Martin Saska, and Vijay Kumar. Model predictive trajectory tracking and collision avoidance for reliable outdoor deployment of unmanned aerial vehicles. 11 2018.

[20] Nathan A. Spielberg, Matthew Brown, and J. Christian Gerdes. Neural network model predictive motion control applied to automated driving with unknown friction. *IEEE Transactions on Control Systems Technology*, 30(5):1934–1945, 2022.

[21] Cladera F. et al. The open vision computer: An integrated sensing and compute system for mobile robots. In *ROScon 2022*, 2022.

[22] Florian Mahlknecht, Daniel Gehrig, Jeremy Nash, Friedrich M. Rockenbauer, Benjamin Morrell, Jeff Delaune, and Davide Scaramuzza. Exploring event camera-based odometry for planetary robots. *IEEE Robotics and Automation Letters*, 7(4):8651–8658, 2022.

[23] Morgan Quigley, Kartik Mohta, Shreyas S. Shivakumar, Michael Watterson, Yash Mulgaonkar, Mikael Arguedas, Ke Sun, Sikang Liu, Bernd Pfrommer, Vijay Kumar, and Camillo J. Taylor. The open vision computer: An integrated sensing and compute system for mobile robots. In *2019 International Conference on Robotics and Automation (ICRA)*, pages 1834–1840, 2019.