Design and evaluation of an UI for astronauts to control mobile robots on planetary surfaces

Julius Arzberger, Jasper Zevering, Fabian Arzberger, Andreas Nüchter
Computer Science XVII – Robotics
Julius-Maximilians-University
Würzburg, Germany
Julius.arzberger@stud-mail.uni-wuerzburg.de, {jasper.zevering|fabian.arzberger|andreas.nuechter}@uni-wuerzburg.de

Abstract—**With an advancing plan to send humans on planetary missions again, there will be an increasing need for direct on-surface robot control as interaction between robots and humans will increase with increasing sophistication and complexity of tasks. Remote control from Earth with long delays in communication will not be possible through the direct interaction on the surface. Although a higher degree of autonomy of the robots themselves will also play a role, there is a fundamental lack of interfaces for astronauts to control robots on future planetary missions. This research emerged from participation in the AMADEE-24 Mars simulation by the Austrian Space Forum, which focus on researching future human-robot missions on Mars. We designed and evaluated user interface (UI) prototypes for mobile robots operated by astronauts by using a user-centered design approach and incorporated astronaut feedback. We tested the resulting functional prototype in hypothetical scenarios which demonstrated in a low error rate. This affirms the effectiveness of customized user interfaces for specific mission needs. We advocate for adaptability and user involvement to enhance human-robot interaction for planetary exploration. This work not only addresses the need for intuitive robot control systems but also presents potential UI layouts for future human-robotic mission interfaces.**

I. INTRODUCTION

In recent years, there has been a growing interest in Mars exploration and missions, with both government agencies and private companies investing heavily in this field [1]–[3]. With NASA already having rovers on Mars and other institutions, such as ESA, planning to send more to the Red Planet, manned missions to Mars are becoming increasingly feasible. In preparation for this moment, the AMADEE-Programm orchestrated by the ÖWF (Austrian Space Forum) is researching a framework for the development of hardware, workflows, and the science behind future human-robotic planetary surface missions [4]. The following research emerges from participation in the ÖWFs AMADEE-24 mission, a Mars simulation program designed to prepare astronauts for future human-robotic missions on Martian terrain [5]. In this simulation, a small field crew of highly trained analog astronauts (astronaut who stays on earth and does science for manned space flights) with spacesuit simulators are tasked with conducting experiments, reflecting real-time delays between Earth and Mars to the Mission Support Center in Austria, which ranges from 5 to 20 minutes in each way depending on the planets position [6]. A detailed description of the mission to clarify the restrictions and requirements of the work is in section III.

Due to the delay in teleoperation of robots on distant planets, proximate interaction between robots and astronauts reduces the challenges of delayed teleoperation. This arrangement enables astronauts to control the robots directly, facilitating real-time control and responsiveness during extraterrestrial missions. Given that robots are highly complex systems, a critical component of this mission is to develop an intuitive and efficient User Interface (UI) for astronauts to interact with them [7]. In space exploration, there are only a few interactions between robots and astronauts, as remote interaction is the standard, where the robot is controlled by the ground center. Recognizing this gap, our work focuses on the conceptualization and realization of potential Graphical User Interface (GUI) layouts designed for astronauts. Our approach is grounded in a user-centered design process and includes a review of the relevant literature and insightful interviews with analog astronauts.

Tools used for visualizing robot data often have a steep learning curve because they were developed for engineers rather than end-users like astronauts (e.g. [8]). Although astronauts are highly capable individuals, they typically lack a background in computer science or robotics and have limited time to train on complex software. Since training astronauts is expensive, the user interface utilized by astronauts needs straightforward and resilient interaction without sacrificing data richness and control capabilities to shorten the training phase.

In our research, we introduce potential GUI layouts for mobile devices designed for astronaut use in robot control and experiment execution. We are thus moving away from the current status of remote interaction within a command centre and moving into the area of proximate control of the rovers by astronauts directly in the field. In the future, astronauts will have to control robots at least partially to carry out experiments, so an ideal interface for them must be developed that provides simple interaction and is resistant to errors. This effort highlights the incorporation of a user-centered design methodology and astronaut feedback into the development process, underscoring the significance of these insights in shaping the interface designs.

We acknowledge funding from the Elite Network Bavaria (ENB) for providing funds for the academic program “Satellite Technology”

II. Related Work

Our investigation into current literature revealed a lack of specific research on user interfaces for astronauts controlling robots. Addressing this gap, our study leverages established guidelines and research from the broader field of robotic user interfaces. In [9] Filippis et al. outlines an approach to designing Graphical User Interfaces for space rover telenavigation by prioritizing cognitive processes. They focus on improving the users mental workload management and situational awareness for decision-making and problem-solving activities through the methodology of Applied Cognitive Work Analysis (ACWA). ACWA evaluates human cognitive demands in mission scenarios in order to design an interface that meets these demands. However, since their focus is on remotely controlling a rover in a planetary environment with three screens, we have used their insights on what to display rather than how to display it. The focus in [10] is on enhancing human-robot interfaces through the application of data visualization techniques. They identified seven common Data Tasks in human-robot interaction, connecting them with best practices, techniques, and findings from visualization research. These visualization techniques have also been incorporated into our GUI, for example on how to develop and maintain awareness trough a minimap. Dury [11] presents an iterative design process for developing a user interface that enhances operator situation awareness and control in remote robot operations. She discusses the evolution through multiple versions of the interface, highlighting improvements in design and functionality based on user feedback and evaluation results. Her focus is on creating interfaces that provide operators with awareness of the surroundings and status of the robot, leveraging sensor data and multi-touch interaction to reduce cognitive load and improve operational efficiency in tasks such as search and rescue. The design guidelines applied for their user interface has guided our design, as well as the considerations about operator awareness with fused sensor data like a point cloud and camera stream for enhanced situational awareness. In [12] Kawamura et al. divide the UI into different Agents, the User Command Agent to issue commands, the Landmark Map Agent for information regarding the location and control of the robot, and the Navigation Information Agent for sensory data and status. They use a move-to-point approach for navigation, which was chosen as the interaction method for our interface as it has proven to be an effective and intuitive method for controlling robots.

The interfaces described so far require more resources, such as larger screens, because they are not designed to control robots while standing close to them. However, our interface enables astronauts not only to operate it from within the habitat but also to stand adjacent to the robot for direct control. The most related research we know of is the control of the humanoid robot Rollin Justin from DLR [13]. To control the robot, which will also be controlled by the astronauts, they designed a tablet application and a smartwatch user interface. For the tablet, they used a content area as the main interaction and visualization, a command area for manipulation and navigation tasks, and a mode selection to switch between the different interactions. The division into these different components has also been incorporated into our design concepts. In this paper, we adapt the principles and guidelines that shaped the aforementioned interfaces to the context of astronaut-operated systems, designing a user interface for a tablet device for use both inside the habitat and outside to control robots.

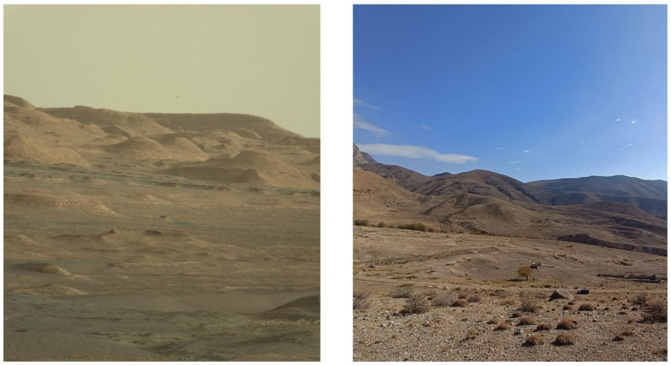

Fig. 1: Comparison between Mars (left) and the AMADEE24 mission environment in the Armenian desert (right), showing the similarity between the test environment and the Martian surface [15].

III. AMADEE-24 Mission

The AMADEE program, build upon the legacy of the PolAres program [14] and led by the Austrian Space Forum, is simulating coordinated human-machine research activities in Mars-like terrain [4]. The PolAres program gained operational expertise through eleven Mars analog field campaigns, the development of high-fidelity spacesuit simulators, ten flights of stratospheric balloons, and a rover program. AMADEE missions, held every two to three years, integrate engineering, scientific, and operational advancements, replicating future planetary surface missions with humans and robots in Mars-like conditions [4]. The AMADEE-24 mission is ÖWFs 14th Mars simulation expedition on Earth in partnership with the Armenia Aerospace Agency. The mission involves trained analog astronauts performing experiments both inside and outside. The chosen location for this simulation is the desert of Armenia, due to its geological and topographical similarity to Mars (see Fig. 1).

It focuses on testing spacesuit simulators and equipment in human-centered interactions, developing testing platforms for life-detection, geoscience techniques, and robotic support tools to aid human missions. Additionally, it seeks to analyze Earths analog environments as models for Mars, increase public engagement with planetary sciences and human exploration, and improve Mars mission management strategies by employing a realistic Mission Support Center model that emphasizes astronaut decision-making frameworks.

The University of Würzburg team, along with two partners (FH Upper Austria and Fachhochschule Würzburg-Schweinfurt), is participating in the iROCS experiment with

Fig. 2: The robots that the astronauts will control: Robot “Lars” (left) with a laser scanner and “Charlie” (right) with four monochrome cameras for rock analysis and the sensor distribution system to launch UWB sensors.

two rovers named “Charlie” and “Lars” as shown in Figure 2. The experiment consists of three parts:

- *UWB:* Stands for UltraWideBand Localization. UWB transmitters, distributed by a sensor-distribution system using compressed air, allows for location tracking by measuring the time of flight (ToF) of signals received by a UWB receiver on the robot Charlie, with the receivers position determined through trilateration or multilateration from four transmitters.
- *GeoSAMA:* Stands for Geo-sampling and Analysis. Four monochrome cameras with different wavelength filters on the robot Charlie are identifying rock samples based on their optical spectrum signatures, aiding in the detection of different rock types [16].
- *SUTerMod:* Stands for Scene understanding and terrain modeling. Image segmentation combined with LiDAR-derived semantic information from the Riegl-Scanner on Lars creates detailed terrain models, enhancing the accuracy of feature identification and classification in various landscapes.

Each robot has a different payload for the experiments. Charlie has four identical monochrome cameras with each an 8mm focal lens and a different wavelength filter for sample analysis of rocks, one intel RealSense T265 Camera for visual-inertial odometry, a Velodyne VLP-16 laser scanner which produces a laser scan that is used by the astronauts for orientation purposes, and two launchers for distributing UltraWideBand (UWB) Sensors. Lars has a SICK LMS-100 laser scanner for simple mapping, one intel RealSense T265 camera for visual-inertial odometry which also streams camera images so the astronauts can use them to orientate themselves, and a Riegl VZ-400 for precise mapping. During the AMADEE mission, the astronauts control the robots and experiments at close range due to the simulated delay between Earth and Mars. Consequently, a robust and user-friendly user interface is required, which is designed in the following section.

Task no.	**UWB**	**GeoSAMA**	**SUTerMod**
1)	Launch Sensor 1	Place the sample in front of robot	Drive robot to area of interest
2)	Rotate robot 90°	Make an image of the sample	Start the laser scan and wait
3)	Redo Task no. 1) and 2) three times for sensors 2-4		
4)	Start calibration process		

TABLE I: The steps the astronauts must take to perform each experiment

IV. DESIGN OF THE USER INTERFACE

The user interface design involves three steps. First, we conduct a requirements analysis to gain insights into the astronauts needs for the experiments. Next, we review the literature to identify design concepts and valid design guidelines in robot interface design. Finally, we create several low-fidelity paper prototypes based on the findings, one of which was developed into a functional prototype for the AMADEE mission. We used the insights from [17] as a guideline for the design of the human-robot interface. In the following, we describe each step.

A. Requirements Analysis

The initial phase of a requirements analysis involves conducting a task analysis, executed following Nielsen’s guidelines [18]. This process involves two discrete steps: Firstly, collect information about the tasks and their objectives, and secondly, analyze the execution of these tasks to achieve the set goals. A task analysis results in a detailed description of the tasks that users perform to achieve a goal, including the sequence of steps they take, and the information they need to complete the task successfully. The task analysis for robot navigation resulted in the following user needs in the language of the user:

1) **Connection:** Is there any connection to the robot?
2) **Orientation:** In which direction is the robot pointing? Are there any obstacles?
3) **Control:** How to select a destination for the robot? Is it arrived?
4) **Real-Time Feedback:** Is the robot in a controllable state? Are there any errors?

Due to time constraints preventing the astronauts from receiving detailed training on the experiments, we record and document the individual steps in a procedure list. This list is then provided to the astronauts during the mission to guide them through the experiments. I holds a list for carrying out the experiments. The next step was to search the literature on HCI, HRI, and robotic systems for design guidelines and concepts for visualization.

B. Design Concepts

We use a three-parted structure for our user interface since it has proven effective in robot teleoperation found in the literature [12], [13], [19], [20]. Additionally, we used multiple design guidelines for designing robot teleoperation for our UI, since guidelines on proximate interaction where not found in the literature [21], [22]. Although our GUI is not aimed at teleoperation, but at controlling the robots at close range, these generally applicable guidelines are used as a starting point for the development of a new GUI. These guidelines emphasize the significance of a user-friendly layout. This is achieved by organizing the interface in a way that reflects the users tasks and workflows. Additionally, its essential to use efficient interaction design by employing single-click interactions. Furthermore, its crucial to present information clearly, for example, by using visual cues like color coding to highlight critical information. [23] and [24] are the basis for our structure on the tablet device, focusing on touch interaction, visual ergonomics, and navigational clarity, such as locating elements by taking into account the users hand and finger position.

C. Low-fidelity prototypes

Based on the results of the task analysis and the literature review, our front-end consists of the following parts:
1) **Navigation Area:** An Exocentric (birds-eye) view and an Egocentric (third-person) view of the map.
2) **Feedback Area:** Showing the connection status between the tablet and the robot as well as any errors relating to the robot.
3) **Mode Area:** Change between driving and operation mode for the different experiments as well as sending commands to the robot to execute the experiments.

We designed four low-fidelity user interfaces with this set of functionalities, which can be seen in Figure 3. Each prototype differs in the arrangement and scale of the areas. In the following, the Main View is referred to as the egocentric view, and the Second View to the exocentric view.

- **Version 1** integrates the Second View over the Main View, positioning mode, and command areas to the right, with camera controls adjacent to the Main View.
- **Version 2** distinguishes the Second View in a separate window to the right, placing mode switches below and relocating the command area and camera controls to the left.
- **Version 3** elevates the Mode Area to the top, adhering to design principles that prioritize important UI elements at the top for deliberate interaction while maintaining the Second View and command area arrangement similar to Version 2 but larger.
- **Version 4** evolves from Version 3, expanding the command panel for greater clarity and moving camera controls to the Main Views bottom as well as the relocation of the automatic driving modes to the modes palette at the top.

Based on the feedback on first impressions from two astronaut interviews, version 4 of the prototype was further developed into a functional prototype.

D. Functional prototype

The main features of the UI are an automatic driving mode, a manual driving mode, and mission modes consisting of UWB, GeoSAMA, and SUTerMod. Mode selection dynamically updates the Command Panel with appropriate task-specific buttons (see Figure 4). A drop-down menu enables switching between the two experiment-conducting robots (see Figure 4 no. 5). Camera views can be adjusted using zoom controls and a virtual joystick (see Figure 4 Camera Control), while the Main View hosts options to toggle the 3D model, a camera stream, and the two view modes (see Figure 4 no. 6-8). Robot Lars uses the RealSense camera stream, which produces higher resolution images than Charlies monochrome cameras due to its filters, which provide clarity and detail only in well-lit conditions. In auto-drive mode (see Figure 4 no. 1), users select any point on the map for the robot to navigate towards. This mode provides data about the distance to the selected point and a countdown in meters until the robot reaches its destination, enhancing situational awareness (see Figure 4 Command Area).

The manual driving mode (see Figure 4 no. 2) offers granular control with buttons for forward/backward movement (specified in meters, see Figure 6a) and turning clockwise/counterclockwise (specified in degrees, see Figure 6b). The astronauts control the robots steering manually by setting the distance and angle for the direction of the robot.

In the UWB panel (see Figure 4 no. 3), the user selects the trigger to launch the Sensor Distribution System (SDS), indicated by a green border around the selected trigger and activation of the launch button (see Figure 6c). Launched triggers turn gray to indicate completion. Following the deployment of all sensors, a calibration sequence needs to be performed by the robot, displaying an on-screen warning via a pop-up window informing the astronauts that the robot will move through the area. The calibration starts when the user confirms a pop-up dialogue. Once the calibration is complete, a subsequent pop-up will report the successful completion of the process and instruct the astronaut to maneuver the robot within the newly established sensor perimeter. Using the pop-up windows, the users attention is shift to a critical event, following the policy of notifying the user of critical events such as automatic robot operation to enhance awareness. The use of pop-up windows is discussed in VII.

The GeoSAMA Panel (see Figure 4 no. 4) is streamlined, featuring a single button to initiate a scanning sequence with the four cameras (see Figure 6d). To ensure the camera is capturing the sample, a camera stream displays one camera view, allowing astronauts to verify the sample positioning within the frame (see Figure 5).

The SUTerMod panel is accessible exclusively when Lars—the robot designated for this experiment—is active in the GUI. Initiating the scan is a one-button operation. Upon

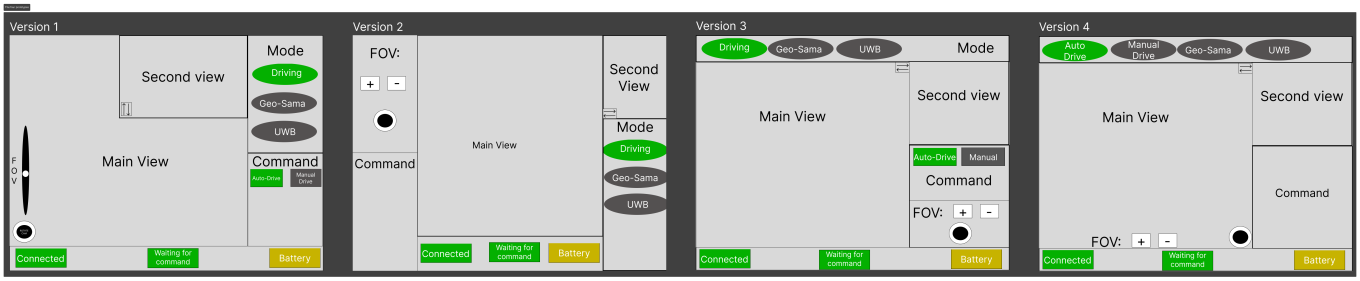

Fig. 3: The four designed low-fidelity prototypes, differ in the arrangement and scale of the areas.

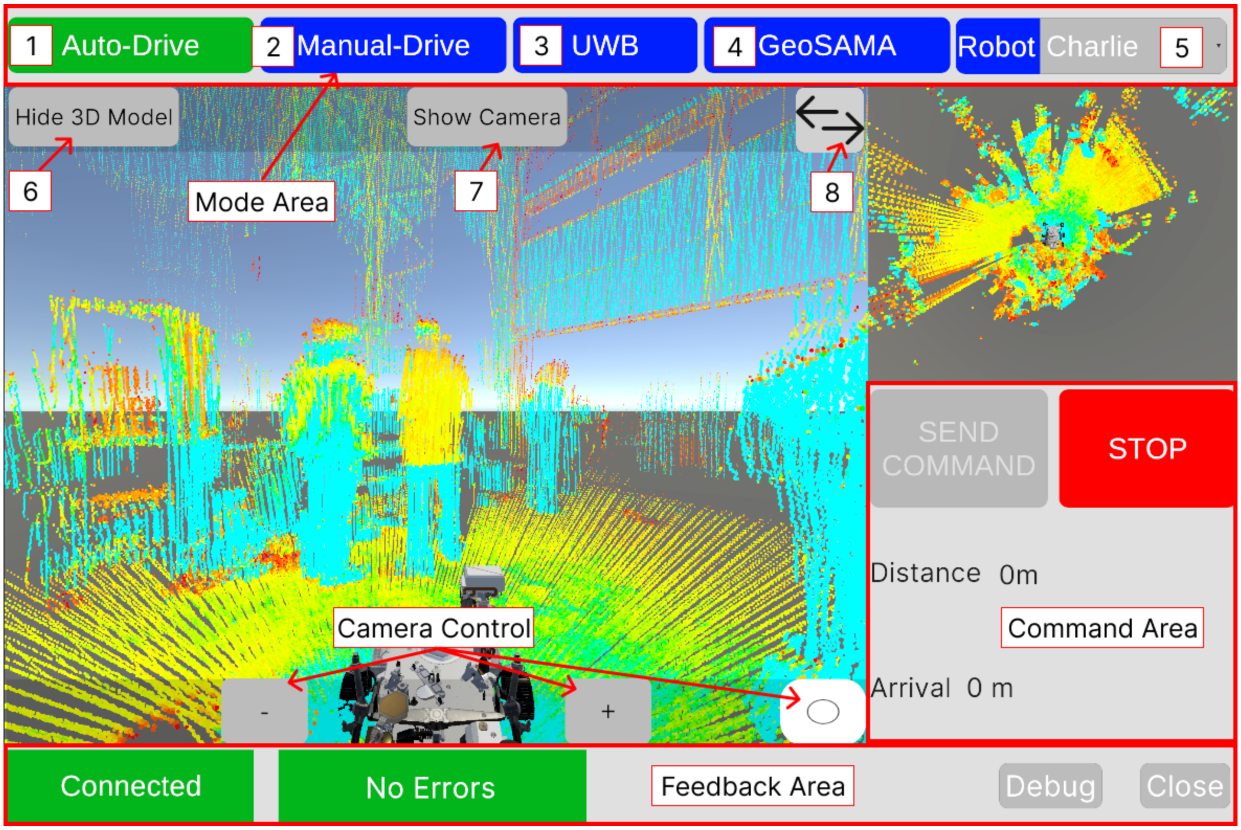

Fig. 4: The User Interface when starting the application, showing the point cloud produced by the laser scanner from “Charlie”.

activation, a progress bar displays advancement and a warning with a countdown cautions against moving the robot during the scanning process for optimal accuracy (see Figure 6e).

V. IMPLEMENTATION OF THE USER INTERFACE

We program the user interface in Unity version 2022.3.10f1, as Unity provides an easy way to send and receive messages through the “Unity Robotics Hub” package, as well as visualizing them through their visualization package [25]. As end device we use the Microsoft Surface 7 Pro with an Intel i3-1115G4 and 8 GB of memory, as this device has enough memory to process the point clouds from robot Charlie. Since the astronauts gloves are not touch-enabled, the tablet is to be controlled with the corresponding pen. Controlling the tablet with a pen is not optimal and will be discussed later in section VII. To establish communication within the system, we use the MikroTik Groove A52 antenna for both robots. This setup allows the robots and the tablet to seamlessly connect to a 5 GHz Wi-Fi network. The connection is facilitated by a third antenna that acts as an access point, creating a unified network platform for all devices to connect to. Subsequently, the robots establish a TCP endpoint to receive and send ROS messages originating from the user interface. Utilizing the ROS TCP Connector package integrated into Unity, a connection to this endpoint was formed, enabling the bidirectional flow of messages with the Robot Operating System (see Figure 7).

VI. EVALUATION OF THE HIGH-FIDELITY PROTOTYPE

As part of the evaluation of our UI, we conducted two interviews with analog astronauts from the ÖWF, which are an important part of the AMADEE-24 mission. First, we carry out semi-structured interviews to identify any challenges

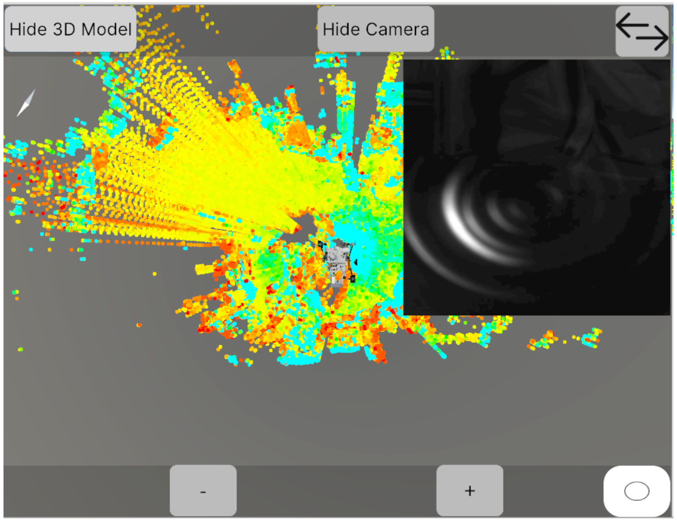

Fig. 5: The camera stream from "Charlie" at the top-right corner of the main view with the point cloud in the exocentric view. The image is produced from one of the four cameras. Trough the filter it appears black when no sunlight is given.

astronauts have encountered with previous systems during space missions, their recommendations for enhancements, and the attributes they believe are critical for a user interface to effectively control mobile robots in space. Their responses are based on their experiences from previous AMADEE missions involving robot control, as well as her expertise as an aerospace engineer at ESA. Astronauts report several challenges while operating robots:

- **Connection Issues:** Delays and poor connections leads to outdated camera feeds, affecting operational efficiency.
- **Range Limitations:** Loss of control outside a certain range necessitated manual retrieval of the robots.
- **Perception Challenges:** The perception of size and distance through the cameras is currently identified as a limitation for effective robot operation, impacting astronauts ability to perform tasks accurately.
- **Interface and Manual Quality:** The need for intuitive interfaces is emphasized alongside the importance of well-structured manuals and error messaging.

Feedback on essential UI functionalities includes:

- **Sensors:** The importance of proximity and distance sensors to gauge the immediate environment of the robot is highlighted.
- **Terrain Analysis:** Automatic recognition of impassable obstacles by the robot is suggested.
- **Emergency Stop Mechanism:** A simple and quick method to halt the robots movement in emergencies is deemed crucial.
- **Telemetry:** Detailed telemetry views including the robots movement, location, and status is considered essential for informed operation.
- **Camera Navigation:** Intuitive camera control and switching is emphasized for effective visual monitoring.
- **Navigation Information:** Displaying planned navigation paths is suggested to improve operator situational awareness.

Secondly, the astronauts engaged with our prototype through experiments framed as hypothetical scenarios. We designed the scenarios to simulate tasks related to UWB, GeoSAMA, and SUTerMod, following the set of procedures in I. The practical assessment aims to identify any operational errors or deviations from the intermediate steps we expect. We requested feedback from the astronauts after each experiment to determine the clarity of instructions. The astronauts performed the experiments with almost no errors. However, a mistake was made when one of them tried to turn the robot to the left. They entered a negative value for the rotation, but the GUI required a positive value in addition with the "Counterclickwise" button to turn the robot in the required direction. The astronauts propose several improvements to enhance system safety and reliability. These include discarding commands if the robot loses connection to prevent unintended actions upon reconnection, the ability to halt the robot or restrict mode changes while in motion for improved safety, and implementing a deadman switch that would automatically stop the robot if the controlling tablet is accidentally dropped.

VII. DISCUSSION

Regarding the layout of the UI, a division into three areas (Mode, View, Status) is suitable for similar applications, as shown by the evaluation in VI. However, the spatial arrangement of these segments within the GUI necessitate adjustment depending on the specific requirements of the mission, the size of the display, and the complexity of the tasks. The design is currently limited to use for the AMADEE mission, as the robots require minimal interaction to complete the straightforward tasks. Looking ahead to more complex missions that require extensive interaction - such as those involving multiple inputs and continuous monitoring of numerous parameters - a more sophisticated display method will be paramount. In such scenarios, the use of tabs is helpful, which effectively emulates a multi-screen layout in a single display unit, by grouping windows with spatial or semantic correspondence together [26]. This allows complex tasks to be handled by organising the interface into switchable views, each dedicated to a specific aspect of the mission. Alternatively, instead of adopting a mission-driven approach that customizes the GUI for specific mission needs and objectives, another option is a general approach where all robots and experiments are controllable through a single interface. Yet, the low error rate observed in our experiment advocates for the mission-driven design for future missions. This method, by removing unnecessary elements, prevents operators from being overwhelmed by irrelevant information, offering a streamlined interface free from clutter and extraneous features.

The observed error rate of 1 during the evaluation with the astronauts suggests that the design of the GUI is effective in facilitating the completion of the task with a limited amount of training. We assume that the low error rate is due to the

(a) Manual Drive Panel for driving (b) Manual Drive Panel for turning (c) UWB Panel (d) GeoSAMA Panel (e) SUTerMod Panel

Fig. 6: The different panels displayed in the command area, depending on the selected mode.

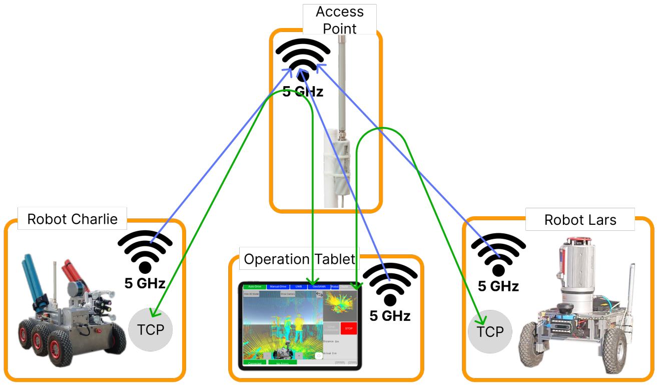

Fig. 7: The communication between the robots and the tablet. The tablet and the robots connect to the access point (blue arrows) and exchange data via the TCP endpoint from the rover over the 5 GHz network (green arrows).

clear labelling of the buttons, which directly correspond to the names of the respective experiments, as well as the clearly defined procedures. The encountered error occurred when an astronaut attempted to execute a leftward turn by entering a negative angle value. Contrary to the users expectation, the GUI require a positive value to be entered alongside the activation of the “Counterclockwise” button. In the specific test run there was a misalignment between user expectations based on common physics and engineering principles and the design logic of the GUI. To address this, we propose a two-button layout as an alternative to the current design. This layout will use positive values for forward or rightward actions and negative values for backward or leftward actions, which will likely resonate with the astronauts propensity for vectorial thinking. While the proposed approach reduces the physical space requirements of the control panel, it also increases the cognitive load for the user. This is due to the need for the user to mentally map the positive and negative values to their respective physical directions. On the other hand, the current multi-button layout requires more physical space but minimizes cognitive load by providing specific buttons for frequent actions. This design choice reduces the need for mentally translating values into actions, which leads to lower cognitive load. To further validate these findings and explore potential improvements, additional experiments need to be carried out.

The pen for GUI manipulation offers the advantage of enabling precise interaction without covering screen elements, which is a concern with manual finger inputs. Nonetheless, astronauts encountered difficulties in handling the stylus due to the thickness of their gloves, which are not optimized for the fine motor control required to handle a pen. Addressing this challenge, we propose to embed a stylus tip directly into the gloves fingertip. While effective, this strategy necessitate considerable modifications to the spacesuit, entailing substantial costs. A more cost-effective alternative is to implement a pen holder attached to the astronauts glove. This method aims to provide a secure location for the stylus, minimizing the risk of it being dropped and allowing for quick retrieval and use. If the graphical user interface is optimized for finger input, potential errors are eliminated like not being able to use the user interface when the pen falls or the battery runs out. This design decision requires larger UI components to compensate for the imprecision of finger use compared to stylus use. This has implications for the overall layout and available screen space, affecting how information is presented and interacted with on the GUI.

The control of the robot is a click-and-wait approach due to our implementation, which does not guarantee smooth control of the robot as would be the case with a joystick, for example. The manual control panel is used for fine-tuning the robot (e.g. to place a stone clearly visible in front of the camera), but it is limited in controlling the position of the robot only in one direction at a time. Another option would be to control the position of the robot with a joystick, which is also possible when wearing gloves. Since a joystick is limited in the assignment of its buttons and mental mapping is required to link a joystick button with the respective action, controlling the experiments via a graphical user interface is preferable. Merging the two input methods would combine the advantages of both modalities, resulting in fast control of the robots and intuitive execution of the experiments via the GUI.

We incorporate pop-up windows in our user interface, as they are widely used to draw users attention towards important information or actions. Pop-ups have a positive impact on user interaction and ensure that crucial information is noticed [27].

When overusing pop-ups or displaying irrelevant information, users generally experience a high degree of irritation and dissatisfaction. [27]. This agrees with the feedback from astronauts who confirm the use of pop-up windows with a good warning to inform about technical conditions that are out of normal behavior or to get further instructions.

VIII. Conclusions and Future Work

In this paper, we conceptualized the design of four user interface prototypes for controlling robots in space missions. These prototypes are based on a literature review and the implementation of human-computer interaction methods such as a requirements analysis and astronaut interviews. Based on the expertise of analog astronauts from the ÖWF, we have developed one design proposal into a functional prototype. With this prototype, astronauts are able to navigate the robot and complete mission-related tasks. This prototype was then evaluated by the astronauts, who completed hypothetical tasks on the GUI without any errors.

The results of the interviews provided further recommendations for future iterations of the UI or similar robotic interfaces. Moving forward, future experiments will need to demonstrate the usability of the UI by incorporating extensive astronaut feedback collected through questionnaires to support the findings with quantitative data. The participation in AMADEE provides a basis for the design of an optimal user interface for astronauts and led to a possible approach which, however, needs to be tested in different scenarios and compared with other solutions in the future. We plan to improve the functionality and usability of the GUI by visualizing more sensor data and providing better scans to provide more situational awareness to the astronauts, and by making it more robust against connection problems. Investigating the optimal navigation method, including the click-to-move approach versus alternative input modalities like joysticks, remains essential.

Regarding the hardware used in our approach there are also shortcomings, such as the limited Wi-Fi range due to the use of a 5 GHz WLAN and the conventional antenna in the used tablet. Other antennas with a greater range and the use of other frequencies (e.g. 2.4 GHz) will have to be investigated in the future. In addition, the UI cannot be controlled without a stylus due to the thickness of the astronauts gloves. The gloves also make it difficult to hold the stylus, so one solution would be to integrate the stylus tip into the glove or to combine the tablet with a joystick for robot control.

With the increasing number of robots and the overall goal of having astronauts permanently on Mars, a standardized GUI approach that meets all needs is needed. Robots are likely to change from single-task robots with special scientific purposes to more everyday working machines on Mars. Therefore, the GUIs used will need to shift from a single-use experimental design to a more holistic approach. General guidelines and international standards need to be established. This will lead to a fruitful use of human-robot interaction for future space exploration.

References

[1] European Space Agency, “Mars express.” https://www.esa.int/Space_in_Member_States/Germany/Mars_Express, 2024. Accessed: 2024-02-03.

[2] NASA, “Mars exploration program.” https://mars.nasa.gov/#mars_exploration_program/4, 2024. Accessed: 2024-02-03.

[3] SpaceX, “Human spaceflight - mars.” https://www.spacex.com/humanspaceflight/mars/, 2024. Accessed: 2024-02-03.

[4] Austrian Space Forum (OeWF), “Amadee program.” https://oewf.org/amadee-programm/, 2024. Accessed: 2024-02-03.

[5] Austrian Space Forum (OeWF), “Amadee-24.” https://oewf.org/amadee-24/, 2024. Accessed: 2024-02-03.

[6] NASA, “Communications with the mars 2020 rover.” https://mars.nasa.gov/mars2020/spacecraft/rover/communications/, 2020. Accessed: 2024-02-03.

[7] M. Goodrich and A. Schultz, “Human-robot interaction: A survey,” *Foundations and Trends in Human-Computer Interaction*, vol. 1, pp. 203–275, 01 2007.

[8] ROS.org, “Rviz.” https://wiki.ros.org/rviz, 2024. Accessed: 2024-02-03.

[9] L. D. Filippis, E. Gaia, G. Guglieri, M. Re, and C. Ricco, “Cognitive based design of a human machine interface for telenavigation of a space rover,” *Journal of aerospace technology and management*, vol. 6, pp. 415–430, 2014.

[10] D. Szafir and D. A. Szafir, “Connecting human-robot interaction and data visualization,” in *Proceedings of the 2021 ACM/IEEE International Conference on Human-Robot Interaction*, pp. 281–292, 2021.

[11] B. Keyes, M. Micire, J. Drury, and H. Yanco, *Improving Human-Robot Interaction through Interface Evolution.* 02 2010.

[12] K. Kawamura, P. Nilas, K. Muguruma, J. A. Adams, and C. Zhou, “An agent-based architecture for an adaptive human-robot interface,” in *36th Annual Hawaii International Conference on System Sciences, 2003. Proceedings of the*, pp. 8–pp, IEEE, 2003.

[13] P. Birkenkampf, D. Leidner, and N. Lii, “Ubiquitous user interface design for space robotic operation,” 06 2017.

[14] Austrian Space Forum (OeWF), “Polares science.” https://oewf.org/en/polares-science/, 2024. Accessed: 2024-02-03.

[15] “Mars 2020 Mission Raw Images.” https://mars.nasa.gov/mars2020/multimedia/raw-images/. Accessed: 2024-02-03.

[16] M. Gmöhling, S. Köhl, J. Zevering, D. Borrmann, S. Ferrari, L. Penasa, R. Pozzobon, and A. Nüchter, “Non-invasive identification of lunar rocks with optical and lidar systems,” in *Proceedings of the 4th International Planetary Caves Conference 2023 (PCC ’23)*, Universities Space Research Association, Lunar and Planetary Institute, 05 2023.

[17] J. A. Adams, “Critical considerations for human-robot interface development,” in *Proceedings of 2002 AAAI Fall Symposium*, pp. 1–8, 2002.

[18] Nielsen Norman Group, “Task analysis.” https://www.nngroup.com/articles/task-analysis/, 2024. Accessed: 2024-02-03.

[19] L. Poli, “User interface for a group of mobile robots,” *UWA*, 2013.

[20] F. Driewer, M. Sauer, and K. Schilling, “Design and evaluation of an user interface for the coordination of a group of mobile robots,” in *RO-MAN 2008 - The 17th IEEE International Symposium on Robot and Human Interactive Communication*, pp. 237–242, 2008.

[21] G. Adamides, G. Christou, C. Katsanos, M. Xenos, and T. Hadzilacos, “Usability guidelines for the design of robot teleoperation: A taxonomy,” *IEEE Transactions on Human-Machine Systems*, vol. 45, 2015.

[22] D. Labonte, P. Boissy, and F. Michaud, “Comparative analysis of 3-d robot teleoperation interfaces with novice users,” *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 40, no. 5, pp. 1331–1342, 2010.

[23] Nielsen Norman Group, “Tablets.” https://www.nngroup.com/reports/tablets/, 2024. Accessed: 2024-02-03.

[24] K. Salo, L. Arhippainen, and S. Hickey, “Design guidelines for hybrid 2d/3d user interfaces on tablet devices - a user experience evaluation,” 01 2012.

[25] Unity Technologies, “Unity robotics hub.” www.github.com/Unity-Technologies/Unity-Robotics-Hub, 2024. Accessed: 2024-02-03.

[26] R. R. Murphy and S. Tadokoro, “User interfaces for human-robot interaction in field robotics,” *Disaster Robotics: Results from the ImPACT Tough Robotics Challenge*, pp. 507–528, 2019.

[27] J. V. Bittner and R. Zondervan, “Motivating and achievement-eliciting pop-ups in online environments: A user experience perspective,” *Computers in Human Behavior*, vol. 50, pp. 449–455, 2015.