ISBN: 978-3-95450-238-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2023-WE3A006

DEPLOYMENT AND OPERATION OF THE REMOTELY OPERATED ACCELERATOR MONITOR (ROAM) ROBOT *

Thomas C. Thayer, Namrata Balakrishnan, Maria Alessandra Montironi, Alessandro Ratti
SLAC National Accelerator Laboratory, Menlo Park, California, USA †

Abstract

Monitoring the harsh environment within an operating accelerator is a notoriously challenging problem. High radiation, lack of space, poor network connectivity, or extreme temperatures are just some of the challenges that often make ad-hoc, fixed sensor networks the only viable option. In an attempt to increase the flexibility of deploying different types of sensors on an as-needed basis, we have built upon the existing body of work in the field and developed a robotic platform to be used as a mobile sensor platform. The robot is constructed with the objective of minimizing costs and development time, strongly leveraging the use of Commercial-Off-The-Shelf (COTS) hardware and open-source software (ROS). Although designed to be remotely operated by a user, the robot control system incorporates sensors and algorithms for autonomous obstacle detection and avoidance. We have deployed the robot to a number of missions within the SLAC LCLS accelerator complex with the double objective of collecting data to assist accelerator operations and of gaining experience on how to improve the robustness and reliability of the platform. In this work we describe our deployment scenarios, challenges encountered, solutions implemented and future improvement plans.
Keywords: Robotics, Infrastructure Monitoring, Remote Control

INTRODUCTION

The Linac Coherent Light Source (LCLS) accelerator at SLAC National Accelerator Laboratory is a very extensive machine, spanning multiple miles in length and employing hundreds of interdependent devices, all operating together create hard X-ray free-electron laser pulses. Originally designed to produce pulses at up to 120 Hz, the upgraded LCLS-II accelerator is capable of producing pulses on the order of 1 MHz. In order for the machine to function optimally at such a high repetition rate, the numerous individual devices need to be monitored while the machine is operating. This is ordinarily accomplished using an array of sensors placed in carefully chosen fixed locations around the accelerator housing supervised through remote network connections. In most cases, these sensors are adequate to remotely monitor the accelerator function, however there are many instances where ad-hoc sensor placement is necessary for diagnostics or troubleshooting. Fulfilling the requirements for as-needed sensor placements have, in the past, been executed with movable one-off sensor carts in lieu of permanent installations, but this still requires accelerator downtime for human access to move the carts into their positions. Thus, there is a need for remotely configurable sensor arrangements that do not require beam off conditions.

First presented in [1], the Remotely Operated Accelerator Monitor (ROAM) robot was created as a sensor platform that can be mobilized to extend the capabilities of the fixed location sensor network. The design of ROAM relies on the use of Commercial Off-the-Shelf (COTS) components and open-source software, which permits streamlined development and allows the platform to be easily configurable for different deployment scenarios with minimal modifications. Because of ROAM's intended environment, prudent attention was paid when developing the remote control software to minimize the chances of collision with accelerator components.

In this work, we discuss the deployments and operational challenges faced by ROAM in the LCLS and LCLS-II accelerators. After providing an overview of some similar robots and applications within accelerator environments in Sec. *Related Work*, a review of the hardware and software features of the ROAM robot is given in Sec. *ROAM Overview*. Then in Sec. *Deployment and Challenges*, we describe the challenges and lessons learned from the deployment of ROAM in the accelerator complex. Finally, we provide closing remarks in Sec. *Conclusions* and consider future plans for the application, development, and deployment of the ROAM platform.

RELATED WORK

The deployment of robots for the purpose of accelerator monitoring is a relatively new field with a limited body of research associated with it. Mainly, larger institutions such as the European Organization for Nuclear Research (CERN) have more developed robotics programs. As an example, [2] discussed an omnidirectional wheeled robot with to perform environmental monitoring for radiation, temperature, and oxygen concentration within the Super Proton Synchrotron at CERN. This robot works autonomously, and has its radiation sensor mounted on a movable arm for flexible positioning. It is purpose built, and is designed to work with CERN's already developed robotics infrastructure. This infrastructure is known as CERNTAURO [3], and was created as a framework for real-time controls of mobile robots. It is similar to Robot Operating System (ROS), however it was conceived

* This work was supported in part by the U.S. Department of Energy under contract number DE-AC02-76SF00515. Any opinions, findings, conclusions, or recommendations expressed in this publication are those of the authors and do not necessarily reflect the views of their affiliated agencies.
† {tthayer, namrata, alexmon, ratti}@slac.stanford.edu

ISBN: 978-3-95450-238-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2023-WE3A006

from the ground up to control multiple robots within the CERN accelerator housings.

Other institutions have started developing robots as well. An interesting example is at the Facility for Antiproton and Ion Research in Europe (FAIR). In [4–6], a robot with semi-autonomous capabilities was designed to inspect the beamline vacuum chambers of the heavy ion synchrotron SIS100 at FAIR. This robot platform is unique in its purpose to visually investigate the accelerator from inside the machine, rather than from outside the machine and inside the accelerator housing. It is meant to reduce the complexity of internal inspections by requiring only one or two parts of the beampipe be opened. At the National Synchronotron Radiation Research Center in Taiwan, which houses the Taiwan Photon Source (TPS), a robot more similar to the design of ROAM was developed. The "PhotonBot", described in [7], is a remote-controlled robot designed for environmental monitoring of the TPS accelerator enclosure during top-up injection mode. Similarly to ROAM, it is designed primarily with COTS components and uses open-source software such as ROS for control. Differently from ROAM, PhotonBot features fixed height placements for its sensors instead of adjustability and lacks radiation monitoring, but the authors mention future possibility of these improvements.

ROAM OVERVIEW

As communicated in Sec. *Introduction*, the ROAM robot primarily consists of COTS hardware and open-source software. This combination allows ROAM to work as a quick prototyping platform through which various deployment opportunities can be achieved with minimal modifications and ancillary development.

Hardware

The ROAM robot's hardware consists of two subsets of devices that work together for the primary objective of accelerator observation with remotely position-able equipment. These are the primary robot components and the configurable sensor platform. A photo depicting the ROAM robot is shown in Fig. 1.

Primary Robot Components ROAM uses a 4WD Rover Pro robot chassis built by Rover Robotics. It is a differential drive chassis with a small 62 cm by 39 cm (24.4 in by 15.4 in) footprint and a large payload capacity. The primary advantages of this platform are its long lasting 294 Wh lithium-ion battery and charging docking station. On battery power alone, the robot can drive continuously for 2 hours or perform stationary monitoring for 12 hours before requiring charging. The charging dock gives the robot the ability to replenish its power supply while out in the field without human assistance, allowing for indefinite continuous operation for multiple missions or long multi-part missions.

On the Rover Pro chassis is an R&D and navigation payload that includes numerous devices required to allow the robot to operate remotely. For computation, an Nvidia Jetson TX2 embedded computer is included which handles all data processing and communicates with the remote operator through a secured wifi network. Of the data which is relayed to the remote operator, among the most critical are the four driving camera feeds (one camera for each forward, reverse, left, and right directions) that allow the operator to view the environment for safe navigation. Additional feedback is provided for wheel odometry by the robot chassis which measures the number of turns for the wheels, and for forward/backward/rotational movements by an inertial measurement unit which measures linear acceleration and angular velocity. For measuring raw distances to nearby objects and obstacles in 3D space, a Velodyne VLP-16 LiDAR is utilized, from which data is processed locally on the TX2 to map the environment.

Figure 1: The ROAM robot hardware with sensors attached. Not shown are the Arduino and LED lamps.

As necessary to operate the robot safely within an accelerator enclosure, supplementary hardware was added to the commercially available robot chassis and navigation payload. Light Emitting Diode (LED) lamps were added on all four sides of the robot to illuminate the ground and surroundings when driving the robot in dark conditions, as is typical in the LCLS accelerator enclosure when the beam is running. Ultrasonic range sensors were added on all four sides of the robot as well, to provide backup detection of nearby obstacles. An Arduino Mega 2560 controls the LED lamp brightness, reads output from the ultrasonic sensors, and is also configured to power cycle the LiDAR when needed. The Arduino acts as a watchdog that will reboot the TX2 processor when a fault condition occurs, such as from a single event upset or latchup. Finally, clear acrylic glass panels that activate limit switches wired to the robot's e-stop were added as bumpers to prevent the robot from moving if a collision occurs due to an uncontrolled movement event. Such an event should not happen in normal circumstances, however due to the unpredictable nature of radiation induced

ISBN: 978-3-95450-238-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2023-WE3A006

computer errors, a runaway robot is an unlikely but distinct possibility.

Configurable Sensor Platform In support of the primary mission of ROAM, the robot carries a configurable sensor platform, which consists of an adjustable height scissor lift with a USB hub. The scissor lift can extend the height of the robot from 58 cm (23 in) to 124 cm (49 in) and lift up to 5 kg (11 lbs). This acts as the attachment point for sensors which may be sensitive to their positional height above the ground, and for specifically monitoring something in the accelerator that requires up close attention. Because the scissor lift and devices on it become additional points of contact with obstacles, it is required to be retracted when the robot is moving. The USB hub, being a widely supported interface, allows a substantial variety of COTS devices to be placed on the scissor lift with minimal modifications, making the ROAM robot highly configurable and easily adaptable to many use cases.

The ROAM robot is typically outfitted with a Mirion Ecogamma-g environmental gamma radiation monitor, which is used to survey radiation conditions in certain areas of the accelerator enclosure while beam is running. This gives ROAM the ability to collect environmental radiation data from a new location at any time, which would otherwise require personnel to place a sensor cart or a permanently installed radiation monitor. Also typically installed are two cameras, one which is an RGB webcam with a spotlight, and one that is a Seek thermal camera measuring temperature. The RGB webcam is useful to provide a visual indication of the status of devices within the accelerator tunnel. An example use case would be to check a gauge or device that stopped responding to or is not integrated into the main accelerator control system. The thermal camera has the capability of streaming calibrated temperature measurements for each pixel of its 320×240 resolution image, which allows for thermal-spatial observation. This can be especially helpful for monitoring devices on an accelerator beam line, such as motors, vacuum pumps, and water-cooled magnets.

Software

The ROAM robot's software, like its hardware, can be grouped into two categories. The first consists of software commonly used to control robots in research environments, and consists entirely of open-source software. The second contains software written or configured specifically for accelerator monitoring, where there are extra precautions that must be taken.

Main Robot Architecture The ROAM system software has three layers - an operating system, a robotics middleware, and a set of packages for common high level robotics functions. Running directly on the TX2 processor is an installation of the Ubuntu operating system with long-term support. Because Ubuntu is one of the most widely used open-source operating systems, there is ample support and documentation available to configure and deploy instances for robots, as well as troubleshoot and fix issues that arise with use. It handles the majority of tasks not directly tied to robotics, such as networking and loading hardware drivers. ROS, which is an open-source middleware suite, runs on top of the main operating system and is used to provide the functionality required to turn the system into a robot. ROS organizes the low-level device control for hardware and handles many-to-many communication as message passing so data and commands can flow between different software components.

There are a number of ROS packages freely available that provide high level functionality to ROAM. Perhaps the most important of these are for mapping, localization with sensor fusion, and obstacle tracking. Mapping is accomplished using Simultaneous Localization and Mapping (SLAM), which utilizes the output of LiDAR data to find walls and objects within line of sight of the robot, creating boundaries and eventually a 2D map of the environment as the robot moves around. When not mapping, ROAM employs Advanced Monte Carlo Localization (AMCL) to place the robot with a previously created map. LiDAR based position data from SLAM or AMCL is fused with other sensor data and control inputs using an Extended Kalman filter, which creates a probabilistic position estimate with known accuracy. Obstacle tracking is handled with a voxel grid, which takes objects found with LiDAR, ultrasonic, or other sensors and tracks them in 3D space on the 2D map.

Accelerator Specific Programming Due to the high radiation environment in which the ROAM robot operates, as well as the lack of radiation hardened components, disruptions to computation are likely to occur. Therefore, ROAM employs a set of three watchdogs to keep it running at all times. The first watchdog is a core function of ROS, which restarts ROS nodes when they are abruptly killed. This includes the Arduino, which communicates through a ROS serial interface. The Arduino acts as a watchdog for itself, making sure communication stays open with ROS, and also for the TX2, which it is capable of restarting via a GPIO reset pin. The TX2 runs a watchdog in the Ubuntu kernel, which will automatically detect problems with the system and other processes including ROS, forcing a reboot of individual processes or the entire system. In Fig. 2 a graph of how the ROAM watchdogs interact is shown.

As the acronym implies, the ROAM robot is a remotely operated system. In order to control the robot remotely, a dashboard was built that allows the operator to view the robot status and send movement commands from a computer on the same network. Data feeds include all of the onboard cameras, the LiDAR and ultrasonic sensors, and a visualization of the SLAM created map. Supplementary modules can be added to the dashboard that stream data from the radiation sensor and control the height of the scissor lift. Additionally, extra navigation software runs locally on the robot to perform collision avoidance, preventing the robot from crashing into the accelerator. It operates to cancel control inputs that would result in a collision, thus provid-

ISBN: 978-3-95450-238-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2023-WE3A006

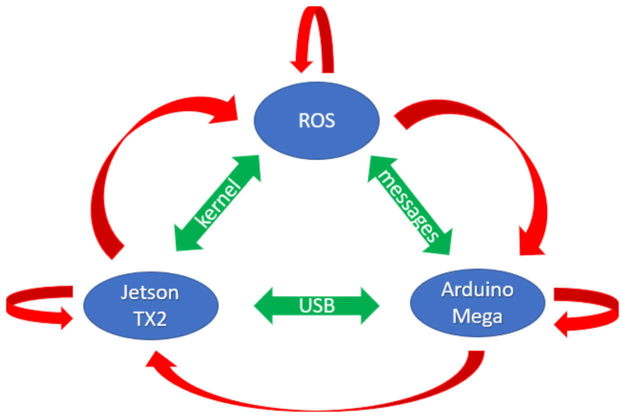

Figure 2: Watchdogs on ROAM are set up such that one is always available to reset a malfunctioning hardware device or restart software. Blue circles represent the systems being watched and the watchdogs on them. Red arrows represent reboot/restart capabilities, leading from a watchdog to a system. Green arrows represent communication between the systems.

ing a fail safe mechanism in case the operator sends a bad command or network issues cause communication delays. This software computes an estimate of the robots future pose based on control inputs, and compares that to the voxel grid occupancy before blocking or allowing the movement.

DEPLOYMENT AND CHALLENGES

Since the initial testing of the ROAM robot in [1], the robot has been deployed to the Beam Transport Hall (BTH) area of the LCLS accelerator. This section will discuss the challenges associated with this deployment, from both an administrative and technical point of view.

Authorizations

In order to deploy ROAM within the LCLS accelerator enclosure, a number of authorizations had to be obtained. Following work planning and control standards is a requirement to ensure safe and reliable operation of ROAM, and therefore plans and procedures were documented at every step.

The initial testing of the ROAM robot to occur in a radiation environment, first described in [1], left the robot without wheels next to a cryomodule radio frequency waveguide. In this configuration, which was maintained for about 4 months, the robot was unable to move and therefore unable to crash. The test showed that a robot which is not specifically radiation hardened can be used in such an environment for some time without worry that it will immediately stop working due to radiation damage. More importantly, data collected showed that the robot would not begin reacting strangely from radiation induced computation error. At no point did the robot's motors begin spinning without explicit commands to do so. This initial testing within the accelerator housing demonstrated that safe operation of ROAM is possible without concern of uncontrolled movement that could damage accelerator components.

Before allowing remote movement of ROAM within the accelerator housing, two different official SLAC approved procedures were written that outlined steps for movement - one for testing and deployment, and another for operation while deployed. Both of these documents required administrative approval when written, and require authorization from appropriate personnel each time they are executed.

The first procedure is an all-encompassing systems check of the robot that has multiple phases. Starting outside the accelerator housing, the ROAM systems are verified working before the robot is allowed to enter its deployment area. Then, inside a safe area of the accelerator housing, operational trials begin, including tests of wheel traction during movement, limit switches, mapping capabilities, obstacle avoidance, networked remote control, and network loss simulation. Data from these tests (such as the map created, stopping distances, battery percentage used, etc) is saved and recorded before the robot is left on its charging station at the deployment location.

The second procedure walks an operator through the steps required to plan and execute ROAM missions while deployed. Similarly to deployment, first the ROAM systems are remotely verified working before starting the mission, including status of sensors, software, and remotely controllable systems excluding driving. Next, the mission plan must be written to include where the robot is going, how long it will stay, what information it will collect, the estimated total mission time, and the estimated battery percent usage. Finally, the procedure describes steps that must be followed during execution and extra information to record for each mission.

Lastly, any repair or modifications made to ROAM while in the accelerator housing requires official SLAC approved procedures in place before action can be taken. This is part of SLAC policy for all equipment in accelerator housings, and not unique to ROAM. This type of procedure requires that all steps and contingencies be laid out before hand, and any deviations from the procedure require a written plan with a second set of authorization from area managers, physicists, and engineers before work can commence. To this end, several procedures were written to accommodate modification or repair of the ROAM robot while deployed in the accelerator housing (including addition/removal of wheels, swapping of controller boards, etc), however several unforeseen problems materialized in the execution of these procedures that could not be immediately fixed. Since ROAM is a mobile platform, removal of the robot from the accelerator housing to perform maintenance or change configurations is now standard.

Technical Complications

During the testing and deployment phases of ROAM, a few technical complications surfaced that required attention. These included wireless networking issues, driving and mapping issues, and hardware limitations. While rel-

ISBN: 978-3-95450-238-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2023-WE3A006

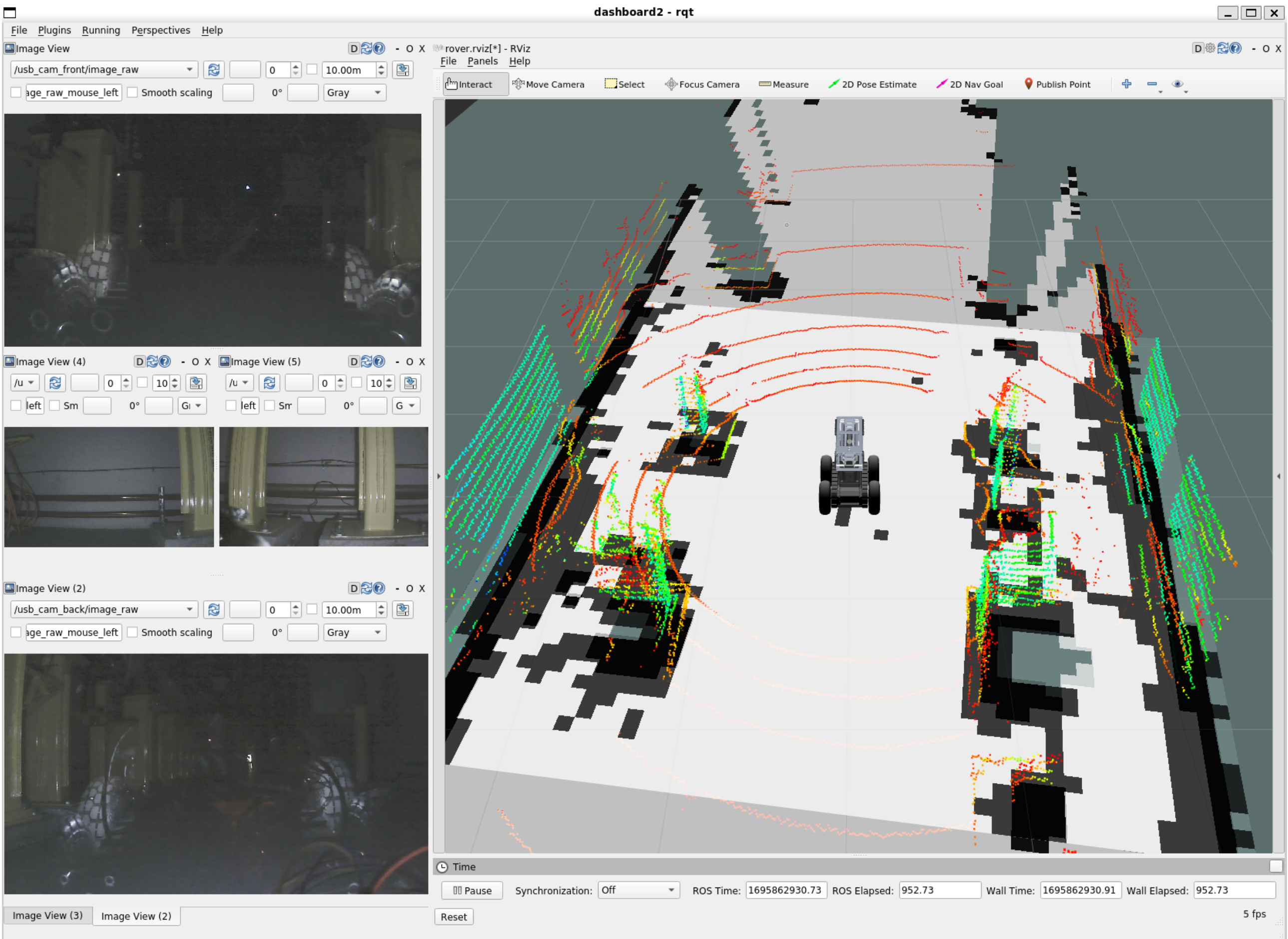

Figure 3: The ROAM remote dashboard, showing the driving camera feeds (left) and the SLAM created map along with voxel grid occupancy and LiDAR data (right).

atively unsophisticated in nature, these complications are nevertheless particular within the context of accelerators.

One troublesome challenge involved the use of the accelerator housing's preexisting wireless network for controlling the ROAM robot. Because the robot moves long distances and between various pieces of equipment, wireless communication is necessary to operate it. However, wireless access points are placed relatively sparsely along the accelerator housing and therefore many areas have poor reception and high latency. The issue is exacerbated by interference caused by radio frequency producing devices. This can be fixed with the introduction of extra access points in problem areas, and indeed was required when deploying ROAM to the BTH area of the LCLS accelerator. In particular, deploying an additional access point in the same location as the robot has proven necessary to keep communication from dropping repeatedly. Wireless access points are also susceptible to radiation damage and redundancy is recommended in case any access point in the operational range of the robot stops working.

During operation of the ROAM robot, other troubles were discovered with driving and mapping. One such trouble was the glare produced by LED lamps reflecting off the acrylic bumpers. As can be seen in Fig. 3, the lights on the left and right sides of the robot produce glare and reflections visible in the front and rear cameras. This impedes visibility while driving the robot in the dark environment of the accelerator enclosure, making it difficult to see obstacles on the ground directly in front of or behind the robot. Another issue encountered was the difficulty in judging the distance driven and area of the accelerator housing the robot was in. Landmarks in many areas are repetitive and not useful for determining position. ROAM uses quick response (QR) codes to act as artificial landmarks, but these are difficult to incorporate into the SLAM software and do not add much intuitive information for a remote operator to interpret.

Regarding the COTS hardware used on the ROAM platform, none of which are radiation hardened, there are a number of issues that occurred. There were frequent cases of the Arduino losing communication and being restarted by the watchdog, however it is not possible to know if this was caused by a radiation-induced single even upset or some other malfunction. Regardless, it showed that the watchdog was able to perform its function successfully. There were

ISBN: 978-3-95450-238-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2023-WE3A006

also instances of other devices requiring frequent rebooting, but again it is unknown if they were due to radiation. For these devices (usually the RGB and thermal cameras on the scissor lift), the software was restarted automatically due to watchdogs, however this would not fix them and instead they required a full power cycle. Multiple software restarts without a power cycle had the unintentional effect of filling the TX2 memory with log files and eventually causing Ubuntu to crash before a fix was implemented. To this end, none of the COTS hardware has yet to be permanently affected by the radiation environment after a cumulative period of over one year in the LCLS and LCLS-II accelerator housings.

CONCLUSIONS

In this work, we have surveyed a body of literature discussing robotics for use in accelerators, summarized the main features of the ROAM platform, and scrutinized the difficulties encountered when deploying the robot in the LCLS accelerator. Because of the unique conditions encountered within the setting of accelerator enclosures, robots are often purpose built and programmed to operate inside them, meeting stringent requirements. ROAM attempts to use COTS components and open-source software for the same objective, with the intent of speeding up development and composing a more flexible platform. After its initial testing and first deployment within an operating accelerator housing, much experience was gained in obtaining authorization to drive the robot remotely, and some technical challenges that hindered progress were addressed. Overall, ROAM has proven capable as a mobile sensor platform for accelerator monitoring.

Future Work

Despite successes, there is still future work needed. For example, as discussed in Sec. *Deployment and Challenges*, the ROAM robot is best removed from the accelerator enclosure when repairs or maintenance are required, which means it cannot continue going on missions. SLAC does currently have a second ROAM robot that could replace the first to minimize mission downtime, however it is configured for development and does not have all sensors installed. Eventually, ROAM should be expanded to allow multiple robots to operate at once, extending data collection coverage across the whole of the LCLS accelerator. On the topic of sensors, ROAM can be improved by incorporating additional sensors that collect different types of data. Some candidates include a magnetometer array, to detect changes in magnetic fields as the robot drives around, and an ionization chamber, for detecting ionizing radiation caused by events other than gamma rays like x-rays and beta particles. On the software side, improvements can be made to the mapping and localization software to better take advantage of QR codes as landmarks. This would allow the robot to build enhanced maps of the accelerator housing and give remote operators better perception of the robot's actual position. Finally, as a long term goal, ROAM could be improved by navigating using optimally safe trajectories implemented as autonomous navigation.

REFERENCES

[1] T. C. Thayer, M. A. Montironi, and A. Ratti, "Roam: A remotely operated accelerator monitor", in *IEEE/ASME Int. Conf. Mechatron. Embed. Syst. Appl. (MESA)*, 2022, pp. 1–6.

[2] C. P. Sesmero, L. R. Buonocore, and M. D. Castro, "Omnidirectional robotic platform for surveillance of particle accelerator environments with limited space areas", *MDPI Appl. Sci.*, vol. 11, p. 6631, 2021. doi:10.3390/app11146631

[3] M. D. Castro, M. Ferre, and A. Masi, "Cerntauro: A modular architecture for robotic inspection and telemanipulation in harsh and semi-structured environments", *IEEE Access*, vol. 6, pp. 37506–37522, 2018. doi:10.1109/ACCESS.2018.2849572

[4] N. Schweizer and I. Pongrac, "Semi-Autonomous Device for Visual Inspection of Vacuum Beamlines of Particle Accelerators", in *Proc. IPAC'17*, Copenhagen, Denmark, May 2017, pp. 4528–4530. doi:10.18429/JACoW-IPAC2017-THPVA042

[5] N. Schweizer and I. Pongrac, "Autonomous Topography Detection and Traversal for an Inspection Robot Within the Beamline of Particle Accelerators", in *Proc. IPAC'18*, Vancouver, Canada, Apr.-May 2018, pp. 3943–3945. doi:10.18429/JACoW-IPAC2018-THPAL128

[6] N. Schweizer and I. Pongrac, "Visual Inspection of Curved Particle Accelerator Beam Pipes with a Modular Robot", in *Proc. IPAC'19*, Melbourne, Australia, May 2019, pp. 4135–4137. doi:10.18429/JACoW-IPAC2019-THPTS014

[7] T. Y. Lee, B. Y. Chen, T. W. Hsu, B. Y. Huang, W. Y. Lin, and C. H. Kuo, "A remote-controlled robot-car in the TPS tunnel", in *J. Phys.: Conf. Ser.*, vol. 1350, p. 012147, 2019. doi:10.1088/1742-6596/1350/1/012147