

Real-Time Monitoring and Remote Guidance of Mobile Robots Using Multimodal Digital Twins

Trung Kien La

Dept. of Comp. Science & Engineering
Frankfurt University of Applied Sciences
Frankfurt a.M., Germany
trung.la@stud.fra-uas.de

René Harmann

Dept. of Comp. Science & Engineering
Frankfurt University of Applied Sciences
Frankfurt a.M., Germany
rene.harmann@fb2.fra-uas.de

Eric Guiffo Kaigom

Dept. of Comp. Science & Engineering
Frankfurt University of Applied Sciences
Frankfurt a.M., Germany
kaigom@fb2.fra-uas.de

Abstract—Although digital twins have been playing a pivotal role in the management of the lifecycle of physical robotic systems of systems, they have hardly been employed to guide mobile robots in real-time. In fact, the guidance of such systems requires functionalities, including the perception of targeted locations and avoidance of collisions, that build upon spatial information beyond internal robot states usually acquired using proprioceptive sensors. In this case, exteroceptive sensors help meet this demand. Nevertheless, such sensors have received little attention thus far in the development of digital twins. On the other hand, the completion of various spatial objectives, such as reverse motions, might require the awareness of the historical internal state of the distant robot. For instance, the current energy budget is likely to constrain the reachability of the initial state after a while, even when spatially and kinematically feasible. We therefore embrace these challenges with a multi-modal approach to provide and employ digital twin of mobile robots. We collect data about the internal state and camera-captured neighborhood of the robot in real-time. The robot operator is thereby provided with a multi-dimensional state and perception view that characterizes the robot, elevates situational awareness, and facilitates decision support. We then develop a versatile graphical interface that helps monitor and steer mobile robots. Since the bidirectional approach is intuitive and user friendly, even novices can remotely guide a mobile robot with multi-modal situational awareness. We show the versatility and effectiveness of our approach in use case scenarios in practice.

Index Terms—Robotics, Multimodal Data, Digital Twins, Industry 4.0, Industry 5.0, Society 5.0, Human-Mobile Robot-Interaction, Systems of Systems

I. INTRODUCTION

The Robot Operating System (ROS) stands as a ubiquitous middleware framework in the realm of robotics, serving as a foundational platform for constructing robot systems and developing robot applications. However, for individuals possessing limited or no prior exposure to robot software, navigating the intricacies of ROS can prove challenging. This challenge is particularly pronounced for users who rely on robotic assistance and require an intuitive means of interacting with these machines. In response to this need, a platform-independent web application was meticulously crafted. Designed with accessibility in mind, this application facilitates seamless monitoring and control of locally networked robots. Users gain access to critical information, including battery status and motor temperatures, all presented through an intuitive web interface. Notably, this application is compatible

with any device supporting web browsers, ensuring widespread accessibility and usability. The robotic platform employed



Fig. 1. Picture of the Husky UGV by Clearpath

in our research is the Husky Unmanned Ground Vehicle (UGV), manufactured by Clearpath Robotics. This medium-sized mobile robot, depicted in Fig. 1, operates on the ROS2 distribution Humble Hawksbill. The Husky UGV boasts a substantial maximum payload capacity of 75 kg, rendering it suitable for transporting and accommodating various auxiliary components. Researchers can mount additional robots, peripherals, and specialized tools on this versatile platform.

Its robust design and adaptability make it an ideal choice for a wide range of robotic applications [2].

II. STATE OF THE ART

C. Crick et al.

- r
- teams
- s

A. Maintaining the Integrity of the Specifications

The IEEEtran class file is used to format your paper and style the text. All margins, column widths, line spaces, and text fonts are prescribed; please do not alter them. You may note peculiarities. For example, the head margin measures proportionately more than is customary. This measurement and others are deliberate, using specifications that anticipate your paper as one part of the entire proceedings, and not as an independent document. Please do not revise any of the current designations.

III. IMPLEMENTATION AND DESIGN

The system architecture of the web application can be dichotomized into distinct backend and frontend components. These components synergistically leverage an array of tools and frameworks to facilitate seamless operation.

A. Backend

The web application is a Flask application. Flask is a micro web framework that is characterized by providing the core features to create a python web based application. Flask is also very flexible and highly expandable [4]. The following tools are used in the backend:

- SQLAlchemy is a Python SQL toolkit and can be integrated into Flask as an extension. SQLAlchemy allows users to link Python objects to SQL databases using Object-Relational Mapping (ORM). This allows, for example, database queries to be made using Python code instead of SQL commands. The extension is also database-agnostic. Python code can be used unchanged for various SQL-based databases, such as SQLite, PostgreSQL or MySQL. SQLite3 is used as a database to store ROS and user data. SQLite3 is a serverless database. This characteristic renders it operable in a self-contained manner, devoid of any supplementary software dependencies or intricate configuration settings. Consequently, it exhibits resource efficiency by minimizing the utilization of extraneous computational resources.
- The Roslibjs JavaScript library and the Rosbridge v2.0 protocol are used to establish bidirectional communication with the robots and the web application. The Rosbridge server it contains, provides a WebSocket connection so that web browsers can communicate with Rosbridge. Roslibjs is employed as a JavaScript library to interact with ROS (Robot Operating System) topics, enabling the subscription to and publication of these topics. A quintessential example of this interaction is the

publication of a geometry_msgs/Twist message on the /cmd_vel topic of the Husky robot, which initiates its movement. The geometry_msgs/Twist message is composed of linear and angular vectors, which represent the velocity in free space. Specifically, these vectors are expressed in terms of m/s and rad/s, respectively, in a right-handed coordinate system. In the context of the Husky robot, a linear velocity of 1 m/s in the positive x-direction corresponds to forward movement at a speed of 1 m/s. To induce rotational movement, an angular velocity is specified in the z-direction. Positive and negative values correspond to counterclockwise and clockwise rotations, respectively.

- The Web Video Server is a ROS package that allows HTTP streaming of ROS image topics. This integrates the live image from a Zed 2i camera from StereoLabs into the web interface.
- Flask-Login and Werkzeug.security extensions are used to handle login, logout, and session functions, as well as to enhance user password security through password hashing.

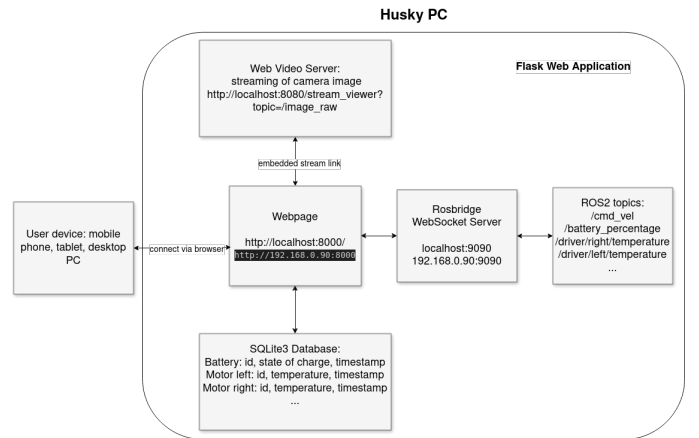


Fig. 2. Backend architecture of the web app

Fig. 2 illustrates the backend architecture and the method by which users can interface with the Husky robot via the web application. This system is designed to operate on any device equipped with a web browser, utilizing the IP address of the Husky's onboard computer for connectivity. The user interface provides real-time access to a stream of visual data from a Zed 2i camera, as well as control options for the robot. Additionally, it displays current ROS data, including metrics such as the battery state of charge (soc) and motor temperatures. A key feature of this system is its ability to track and store historical data in the SQLite3 database. This allows for longitudinal analysis of the robot's operational data, which can be instrumental in performance optimization and troubleshooting. This system provides a flexible and accessible platform for robot control and data monitoring. It underscores the potential of web-based interfaces in enhancing the usability and functionality of robotic systems. The use of an IP-based

connection protocol further emphasizes the system’s adaptability and broad accessibility. The incorporation of a database for data tracking and storage demonstrates a commitment to data-driven decision making and performance optimization.

B. Frontend

C. Equations

D. L^AT_EX-Specific Advice

E. Some Common Mistakes

REFERENCES

- [1] “ros.org,” 2024. [Online]. Available: <https://ros.org>
- [2] “clearpathrobotics.com,” 2024. [Online]. Available: <https://clearpathrobotics.com/husky-unmanned-ground-vehicle-robot/>
- [3] Crick, C., Jay, G., Osentoski, S., Pitzer, B., and Jenkins, O. C. “Ros-bridge: Ros for non-ros users,” in *Robotics Research*, Springer, 2017, pp. 493–504.
- [4] Suraya, Muhammad Sholeh, “Designing and Implementing a Database for Thesis Data,” in *International Journal of Engineering, Science & Information Technology (IJESTY)* Volume 2, No. 1, 2022, pp. 9–14, eISSN: 2775-2674
- [5] “ros.org,” 2024. [Online]. Available: http://wiki.ros.org/rosbridge_suite

IEEE conference templates contain guidance text for composing and formatting conference papers. Please ensure that all template text is removed from your conference paper prior to submission to the conference. Failure to remove the template text from your paper may result in your paper not being published.