A Quadruped High-Level Control Module for Worker Activity Recognition

Robert Ashe, Mani Amani, Conner Sommerfield, Reza Akhavian

Data-informed Construction Engineering (DiCE) Research Lab, Department of Civil and Environmental Engineering, San Diego State University; e-mails: rashe7414@sdsu.edu, mamani5250@sdsu.edu, csommerfield3597@sdsu.edu, rakhavian@sdsu.edu

ABSTRACT

Quadrupeds are becoming increasingly popular in construction engineering research and practice for their affordability and accessibility. These robots navigate uneven terrain commonly found in construction sites, making them suitable vehicles for sensors and monitoring tasks. However, the lack of streamlined and fully-developed client-side software packages inhibits rapid deployment of application-specific models to the field. Furthermore, substantial prerequisite knowledge of computer science and programming significantly impedes the ability of non-experts to adapt the robots to specific applications. In this work, a comprehensive and detailed framework to address this gap in accessibility is presented to facilitate seamless communication between the robotic vehicle, edge devices, sensors, pathfinding algorithms, and a Unity simulation for streamlined mission planning and execution. Additionally, we have conducted a construction-specific case study using the current configuration of the package to demonstrate the efficacy and flexibility of this framework. The findings highlight the potential of accessible software tools in expanding the utility of robotic platforms across various engineering domains.

INTRODUCTION

The use of robotics in the construction engineering field has grown substantially in recent years, driven by the industry's need for enhanced automation and efficiency (Bogue 2018). The availability of robotic platforms by manufacturers such as Unitree Robotics (Unitree Robotics 2025) or Boston Dynamics (Boston Dynamics 2025) has played a significant role in research and development in this space, and their capabilities have proven them to be an attractive and low-cost choice for robotic vehicles (Reese et al. 2024). Despite their obvious utility, the software development kit (SDK) provided with these robots lacks ready-to-use features and high-level abstractions that make it easy for users that do not have extensive knowledge of computer science and programming to adapt these robots to specific tasks. Furthermore, there is no built-in functionality that allows users to easily connect additional sensors and edge devices to the robot. This can be a major obstacle to widespread application in the construction industry due to a lack of a workforce specializing in robotics and the difficulty of integrating custom electronics into robotic vehicles (Cherubini & Navarro-Alarcon 2020).

Building Information Models (BIM) are a rich source of spatiotemporal data that are commonly used in the Architecture, Engineering, and Construction (AEC) industry (Song et al. 2017). The geometry and floor plan represented in a BIM can be utilized as a partially known environment to accomplish a multitude of robotic tasks. The 3D representation of the BIM can be stored in the form of an Autodesk FilmBox (FBX) file and imported into the Unity game engine, an extremely popular software application commonly used in robotics research (Bartneck et al. 2015). Utilizing the Unity game engine to represent real buildings and floor plans in a virtual environment creates a powerful tool for enabling physical interactions in the real world via robots, offering significant benefits to construction robotics research. Moreover, the available packages for the Robot Operating System (ROS) and the Unity Machine Learning Agents Toolkit (ML-Agents) streamline the research and development process in Unity.

The utility of accessible robotic systems like those offered by Unitree Robotics can be improved significantly with integrated sensors and other customizations. They are commonly equipped with visual and infrared cameras, light detection and ranging (LiDAR) sensors, and other specialized instruments to assist in a wide variety of inspection and maintenance tasks (Kohlhepp & Bretthauer 2002). However, integrating these sensors with robotic vehicles is not a trivial task. Programming language incompatibility, complicated dependencies, and network communication can become significant barriers to the rapid deployment of application-specific robots.

In this work, we develop a software package that integrates Unitree robots with BIM and the Unity game engine for seamless mission planning, simplifying the process for users to integrate sensors and customize the robots to their specific needs. The system leverages the Unitree SDK to provide precise control over the robot's linear and rotational movement and includes a modular application programming interface (API) that allows users to easily connect and communicate with a wide range of edge devices, sensors, or external APIs. By bridging these components through a unified interface, the application simplifies robotic operations and enhances adaptability for a wide variety of use cases.

The main contributions of this work are as follows:

1. Creating a high-fidelity 3D environment for BIM-centric mission planning.
2. Developing and distributing a modular, open-source software package that provides streamlined mission planning for quadruped robots in the AEC industry.
3. Conducting a case study using the package and a Unitree Robotics Go1 robot to demonstrate the efficacy and versatility of the package in a construction environment, featuring a novel self-labeling mechanism for dynamically generating labeled data during human activity recognition (HAR).

SYSTEM OVERVIEW

As shown in Figure 1, Unity serves as the main platform for mission planning, handling core functionalities such as pathfinding, motion control, and overall system logic. Communication between Unity, the robot, and external devices or APIs is enabled by a modular backend architecture written in Python. This architecture employs a transmission control protocol (TCP) client-server framework and facilitates communication across all components of the system.

Our construction-specific case study implements the proposed system to enable human activity recognition (HAR) on a construction site. This implementation relies on a video feed supplied by a Raspberry Pi and webcam mounted on a Unitree Go1 robot, which is then processed by a local server running the Python backend. This system supports HAR and additional features by integrating activity recognition deep learning modules, the ElevenLabs (Eleven Labs 2025) API, the OpenAI API, and other custom functionality using the Python backend, with the Unity game engine providing high-level control. This example shows that the proposed system can be customized freely, provided that a Unitree robot is used as the base vehicle.

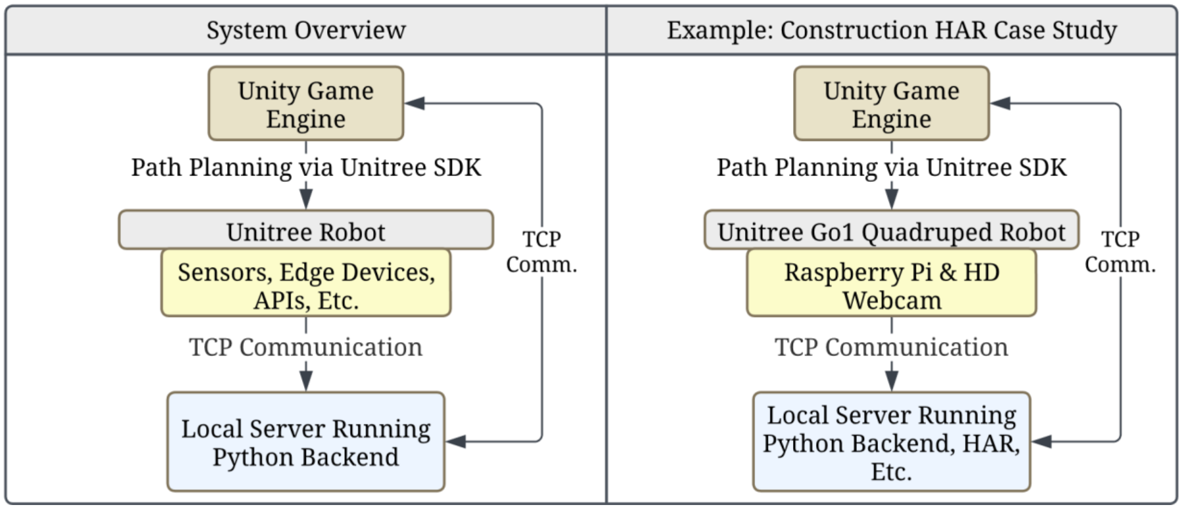

Figure 1. System architecture overview and example configuration for construction HAR case study.

CLIENT-SERVER MODULARITY

To simplify the task of adapting robots to specific applications, the Python backend utilizes a client-server architecture where each capability or feature is managed by its own client-server module. This design improves efficiency by leveraging multiprocessing while providing a generic template users can replicate to implement new features. Communication between Unity and the Python backend is facilitated by TCP and a generic callback method, while features that operate independently from Unity communicate using TCP and generic request-response methods. This configuration combines speed, efficiency, and robust data management while providing users a flexible platform that can be used to integrate sensors, edge devices, and other features to adapt Unitree robots to specific applications.

SELF-LABELING

We evaluated the efficacy of this framework by solving a construction research problem using quadruped robots. These robots are utilized extensively in the construction industry for tasks such as surveillance, monitoring, and data collection. Previous studies have highlighted their effectiveness in applications like construction progress monitoring (Halder et al. 2024) and semantic data collection for the enrichment of BIM models (Zhai et al. 2024). A major challenge in this field is the scarcity of labeled datasets, prompting researchers to focus on synthetic data generation to support emerging machine learning algorithms (Tohidifar et al. 2024). The expense and time-intensive nature of traditional dataset collection exacerbate this issue, making efficient solutions critical for advancing the field.

Semi-supervised learning techniques, such as pseudo-labeling, have been developed to mitigate the challenge of data scarcity across various domains. Pseudo-labeling assigns true labels to unlabeled data by leveraging a confidence threshold to predict the most likely labels for the data (Lee et al. 2013).

Human Activity Recognition (HAR) is a vital area of research within the AEC industry, with applications spanning worker-robot interaction (Shahnavaz et al. 2023), safety enhancement (Frank et al 2019), and construction progress monitoring (Sherafat et al. 2020). Despite its importance, HAR development is hindered by the limited availability of domain-specific datasets, which are costly and difficult to collect. In our construction-specific case study using the proposed framework, we address this limitation by integrating a self-labeling mechanism that generates labeled data dynamically as HAR is performed, demonstrating the potential for semi-supervised learning to streamline dataset creation.

CONCLUSION

In this work, we presented a modular software package for use with Unitree robots that incorporates BIM files, Unity game engine simulations, and a Python backend to integrate various sensors and APIs, creating a flexible and accessible framework for mission planning. Through a construction-specific case study, we demonstrated the system's effectiveness in performing human activity recognition while dynamically labeling data to address dataset scarcity in the construction engineering domain. The proposed framework simplifies the deployment of robotic systems and enhances their adaptability across diverse engineering applications. Future research on this project could involve optimizing real-time performance, adding support for new sensors and customizations, and the expansion of the framework to facilitate the coordinated operation of multiple robotic systems.

PACKAGE AVAILABILITY

Source code for the proposed software package can be found on the DiCE Lab GitHub page (DiCE Lab GitHub 2025).

REFERENCES

Bartneck, C., Soucy, M., Fleuret, K., & Sandoval, E. B. (2015). "The robot engine — making the unity 3d game engine work for hri". 24th IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN), 431– 437. https://doi.org/10.1109/ROMAN.2015.7333561

Bogue, R. (2018). "What are the prospects for robots in the construction industry?" Ind. Robot, 45, 1–6. https://doi.org/10.1108/IR-11-2017-0194

Boston Dynamics. 2025. "Boston Dynamics." Accessed January 13, 2025. https://bostondynamics.com/

Cherubini, A., & Navarro-Alarcon, D. (2020). "Sensor-based control for collaborative robots: Fundamentals, challenges, and opportunities". Frontiers in Neurorobotics, 14. https://doi.org/10.3389/fnbot.2020.576846

Eleven Labs. 2025. "11Labs." Accessed January 13, 2025. https://elevenlabs.io/

DiCE Lab GitHub. 2025. "DiCE Lab GitHub." Accessed January 13, 2025. https://github.com/DiceLabs/go1

Frank, A. E., Kubota, A., & Riek, L. D. (2019). "Wearable activity recognition for robust human-robot teaming in safety-critical environments via hybrid neural networks". 2019

IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) (pp. 449-454). IEEE.

Halder, S., Afsari, K., & Akanmu, A. (2024). "A robotic cyber-physical system for automated reality capture and visualization in construction progress monitoring". https://arxiv.org/abs/2402.07034

Kohlhepp, P., & Bretthauer, G. (2002). “Cooperative service robots for the predictive maintenance of process plants”.

Lee, D.-H., et al. (2013). “Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks”. Workshop on challenges in representation learning, ICML, 3(2), 896.

Reese, R., Kovarovics, A., Charles, A., Koduru, C., Tanveer, M. H., & Voicu, R. (2024). “Optimizing data capture through object recognition for efficient sensor and camera management with a quadruped robot”. SoutheastCon 2024, 1125–1130. https://doi.org/10.1109/SoutheastCon52093.2024.10500066

Shahnavaz, F., Tavassoli, R., & Akhavian, R. (2023). “Robust Activity Recognition for Adaptive Worker-Robot Interaction using Transfer Learning”. Computing in Civil Engineering 2023 (pp. 388-395).

Sherafat, B., Ahn, C. R., Akhavian, R., Behzadan, A. H., Golparvar-Fard, M., Kim, H., ... & Azar, E. R. (2020). “Automated methods for activity recognition of construction workers and equipment: State-of-the-art review”. Journal of Construction Engineering and Management, 146(6), 03120002.

Song, Y., Wang, X., Tan, Y., Wu, P., Sutrisna, M., Cheng, J. C., & Hampson, K. (2017). “Trends and opportunities of bim-gis integration in the architecture, engineering and construction industry: A review from a spatio-temporal statistical perspective”. ISPRS International Journal of Geo-Information, 6(12), 397.

Tohidifar, A., Kim, D., & Lee, S. (2024). “Make it till you fake it: Construction-centric computational framework for simultaneous image synthetization and multimodal labeling”. Automation in Construction, 167, 105696. https://doi.org/https://doi.org/10.1016/j.autcon.2024.105696

Unitree Robotics. 2025. “Unitree.” Accessed January 13, 2025. https://www.unitree.com/

Zhai, R., Zou, J., Gan, V. J., Han, X., Wang, Y., & Zhao, Y. (2024). “Semantic enrichment of bim with indoorgml for quadruped robot navigation and automated 3d scanning”. Automation in Construction, 166, 105605.