The University of Hong Kong

Faculty of Engineering

Department of Computer Science

COMP 4801 Final year Project 2023/24

Intelligent Robot Design and Implementation

Group: FYP23005

Members:

Kwok Ka Yan (3035705336)

Leung Cham Chung Terry (3035704435)

Tang Gillian (3035705130)

Chan Hiu Tung (3035705180)

Supervisors:

Dr. T. W. Chim

Mr. David Lee

Table of Contents

1 Project Background

1.1 Quadruped Robots

Legged robots have gained significant attention in recent decades, owing to their remarkable ability and potential to operate across different surfaces and terrains [1]. Among all legged robots, researchers identify quadruped robots, which consist of four legs, as the best in terms of mobility, stability of locomotion, as well as ease of control and design [1, pp. 2017].

The history of quadruped robots can be dated back to 1870 when Russian mathematician Chebyshev developed the first walking mechanism. Since the 1960s, when the first autonomous quadruped robot was developed [1, pp. 2018], institutes, universities, and technology firms have been actively developing mobile robots. Over time, quadruped robots have improved gradually, such as being equipped with different intelligent programs and advanced sensors, as well as moving in smoother motion and enhanced speed [1, pp. 2018-2024].

These modern robots are equipped with diverse features and can be found practical in assisting humans in real-world tasks, such as in search and rescue missions as well as commercial activities. For instance, quadruped robots with delivery systems have been designed to carry and deliver medical kits to target locations [2]. Another group of researchers is also developing a system that enables quadruped robots to pick up packages and place them on tables, which demonstrates the potential of quadruped robots in package delivery and the business world [3]. All these examples demonstrate the real-world application and potential of quadruped robots in various fields.

Currently, several quadruped robots are made commercially available in the market. For example, Spot from Boston Dynamics, Go1 from Unitree and CyberDog from Xiaomi. The cost of these commercial quadruped robots ranges from around 1540 USD to 74500 USD [4], [5].

1.2 3D Printing

Quadruped robots have emerged as good learning tools for students to explore further in the field of robotics [6]. While some open-source quadruped robots, such as Pupper and Doggo from Stanford University are available online for students to replicate and study, these two robots require relatively expensive materials like aluminium and carbon fibre [6, pp. 2].

To be more cost and time-effective, a plastic, 3D-printed robot will be more suitable for this project. 3D printing technology offers the advantages of high flexibility and ease of customization. The ability to modify 3D models and print new parts easily facilitates a trial-and-error approach, enabling continuous improvement in the robot design and ensuring that the resulting robot will be best aligned with the project objectives and hardware components.

Therefore, in this project, the quadruped robot will be 3D printed, based on the model developed by the MakerLab of the Department of Computer Science at the University of Hong Kong. PLA will be used as the printing material for the robot’s main components, and further details about the hardware will be discussed in Section 3.

2 Project Objectives

This project aims to leverage the capabilities of 3D printing technology and AI/ML to develop an intelligent quadruped robot with different features and control methods.

Based on the 3D model provided by MakerLab, modifications and adjustments will be implemented to align with the hardware and control framework. The parts of the robot will then be 3D printed using a 3D printer and PLA material, followed by their assembly.

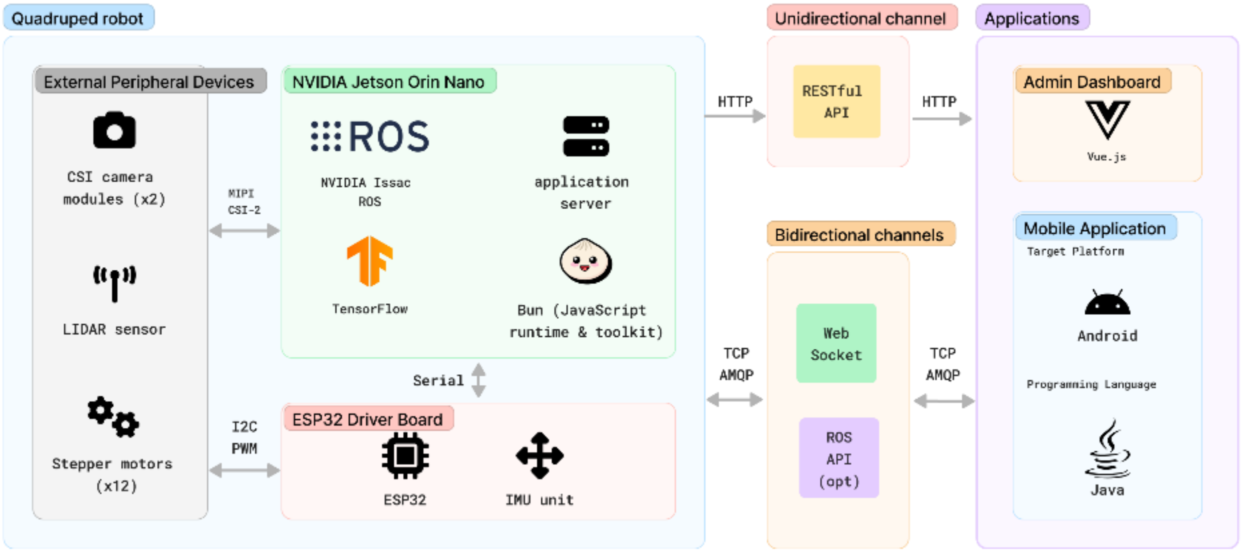
Equipped with a camera and sensors, the robot will be able to autonomously navigate and explore the surroundings. With artificial intelligence and machine learning features, the robot will be capable of learning from the collected environmental data, thereby empowering it to move smoothly and avoid obstacles.

To ensure user-friendly interaction, there will be three ways to control the robot in this project, which include mobile app/keyboard control, audio commands and pose estimation. A dedicated mobile app will be developed to provide a user-friendly interface for users to control the robot. The robot can also interpret and respond to the audio commands and physical gestures of the users.

3 Project Methodology

Given the proposed architecture design, the project will be separated into separate domains: Hardware and firmware, AI/ML backend, application server, and frontend applications, as well as other non-functional requirements such as testing and automated deployment. These domains, along with project management, will be discussed below.

The following diagram shows the tentative system architecture:



3.1 Hardware and Firmware

In terms of hardware, the following electronics will be used: a single board computer (SBC) running ROS to host backend applications, various peripheral devices connected to the SBC such as camera modules and LiDAR sensors, as well as ESP-32 microcontrollers interfacing with the SBC. Also, other physical components will be designed using SketchUp and CAD software, 3D printed, and assembled.

In terms of firmware, the motors and IMU controllers will be implemented, in addition to the serial interface to the SBC via ROS messages. As ESP-32 has embedded WIFI and Bluetooth modules, they will also be leveraged and used throughout the project (I.e., used as serial connection).

It is planned that a first prototype implementing the champ framework will use a Raspberry PI as SBC, ESP32 as microcontroller, and LiDAR sensors. In the following iterations, a Nvidia Jetson Orin Nano Developer Kit will be used as SBC instead as it provides a solid AI inference performance and Jetson software stacks. Besides, hardware peripherals like CSI cameras will be installed and integrated within the system. In addition, the current 3D model for the robot dog will be adjusted to the new configurations.

3.2 AI/ML Backend

To enable intelligent behaviours as stated in the objective, a dedicated AL/ML backend would be implemented, which would be closely integrated with ROS for optimal performance. Nvidia offers a collection of AI packages implemented in ROS 2 nodes as well as hardware acceleration on Jetson devices. Therefore, Nvidia Issac ROS would be deployed to the SBC.

As mentioned above, it is aimed that the quadruped robot would be capable of intelligent tasks involving auto navigation and scene reconstruction. To implement these features, perception AI packages provided by Nvidia would be leveraged. For example, Issac Stereo Perception would be used to perform obstacle avoidance during navigation. Compared to conventional python libraries and frameworks, these packages are optimized towards robotics applications and edge devices like Nvidia Jetson modules, making them ideal for our applications in terms of power efficiency and overall accuracy.

Apart from the topics related to VSLAM (visual simultaneous localization and mapping), the quadruped robot should be able to respond to human instructions via speech and gesture. These methods of control require pose estimation, audio recognition, and potentially natural language processing. Given the unique use cases, the models deployed would be trained with custom datasets to customize the behaviour of the quadruped robot.

3.3 Application Server

While the AI/ML backend enables high-level features mentioned above, it would be less ideal for frontend applications to communicate with ROS with client API libraries solely as frontend applications might have to preprocess raw data fetched and send out multiple API requests for a single action. Therefore, an application server acting as a middleware between ROS nodes and frontend applications would be implemented, which is composed of two main modules.

For data visualization purposes, the server would continuously collect ROS metrics and log messages, which would be stored in a local database. Upon data requests, the server would extract and process the relevant data through RESTful API. For user interactions, the server would communicate with the mobile application via several channels including WebSocket and ROS API. WebSocket would be the primary method of communication as it provides low-latency TCP connection. Meanwhile, ROS API can provide low-level access to ROS nodes for debugging and calibration purposes. Both modules would be deployed with Bun, a lightweight and efficient JavaScript runtime for optimal performance.

3.4 Frontend Applications

Various frontend applications will also be implemented for data visualization and control, including web dashboards and mobile applications. They will be interfaced with the ROS system directly using ROS message APIs or indirectly through backend services using HTTP, WebSocket, or other message queue technologies. Mobile applications will be implemented using Android Studio, while other web applications will use the prevailing web stacks such as MERN or MEVN.

3.5 Non-functional Requirements

Firstly, system validation will be carried out to improve reliability and maintainability, including unit testing, integration testing, and so on. Different testing frameworks will be used depending on the application, such as Platform IO's Unity test framework for microcontrollers, as well as Selenium and Jest for web applications.

Secondly, as the applications are developed and deployed in different environments, containerization technologies such as Docker will be employed, while automated deployment will be implemented with GitHub actions, which are triggered when new versions of the software are available.

3.6 Project Management

Given the complexity of the project, it is preferable to adopt some project management principles and tools to track the progress and adjust the scope accordingly. GitHub Projects will be used for managing and coordinating tasks. Agile development will be adopted to flexibly manage the scope with regards to unforeseen technical roadblocks and ensure earlier system verification due to shorter iteration cycles.

4 Project Schedule and Milestones

The following table shows the tentative schedule of work:

Epic	Task	Priority	Completion ETA
3D printing	Design URDF model using current STL mesh	High	Nov (Phase 2)
	Research other 3d models for use	High	Jan (Phase 2)
	Redesign and assemble 3d models to suit control framework	Medium	Jan (Phase 2)
Firmware Implementation	Implement champ framework firmware for esp32 (motors, MPU)	High	Nov (Phase 2)
	Configure and deploy ROS champ controller on SBC	High	Dec (Phase 2)
	Implement application server to interface with mobile app (WebSocket)	Medium	Feb (Phase 3)
AL/ML features	Configure NVIDIA Issac ROS on SBC	High	Dec (Phase 2)
	Implement motion control based on speech and gesture	High	Dec (Phase 2)
	Implement auto exploration and navigation	High	Feb (Phase 3)
	Provide stereo vision for VR devices	Medium	Feb (Phase 3)
Mobile App	UX design for features	High	Dec (Phase 2)
	App implementation	High	Mar (Phase 3)
Admin dashboard	Implement metrics collection and log forwarding in application server	Medium	Mar (Phase 3)
	Develop web application to display real-time metrics	Low	Mar (Phase 3)
CI/CD pipeline	Auto deployment for SBC	Medium	Nov (Phase 2)
	Auto deployment for mobile app services	Medium	Feb (Phase 3)

References

[1] P. Biswal and P. K. Mohanty, “Development of quadruped walking robots: A Review,” *Ain Shams Engineering Journal*, vol. 12, no. 2, pp. 2017–2031, 2021. doi:10.1016/j.asej.2020.11.005

[2] C. D. Bellicoso *et al.*, “Advances in real-world applications for Legged Robots,” *Journal of Field Robotics*, vol. 35, no. 8, pp. 1311–1326, 2018. doi:10.1002/rob.21839

[3] J. Hooks *et al.*, “Alphred: A multi-modal operations quadruped robot for package delivery applications,” *IEEE Robotics and Automation Letters*, vol. 5, no. 4, pp. 5409–5416, 2020. doi:10.1109/lra.2020.3007482

[4] I. Randall, “Move aside, Spot! Xiaomi launches a $1,540 CyberDog as a cheaper rival to Boston Dynamics’ $74,500 four-legged robot dog,” *Daily Mail*, Aug. 11, 2021

[5] “Unitree go1,” UnitreeRobotics, https://shop.unitree.com/products/unitreeyushutechnologydog-artificial-intelligence-companion-bionic-companion-intelligent-robot-go1-quadruped-robot-dog (accessed Oct. 1, 2023).

[6] Y. N. Zhang and M. Q. An, “Structure and software architecture design of an extremely low-cost quadruped robot for Education,” *Journal of Physics: Conference Series*, vol. 1931, no. 1, p. 012007, 2021. doi:10.1088/1742-6596/1931/1/012007