



Research Paper

Development of Autonomous Quadruped Robot: Search and Rescue Autonomous Helper (SARAH)

Paige Andrie¹, Isaac Huth¹, Zachary Derkowski¹, Matt Janicki¹

Wei Shi², Jason Liu^{3,*}

^{1,2,3}(University of Wisconsin-Stout)

*Corresponding Author: Jason Liu

ABSTRACT: This research presents the design and development of SARAH (Search and Rescue Autonomous Helper), a low-cost autonomous quadruped robot intended for applications in search and rescue operations. The system integrates mechanical design, control systems, and perception technologies to achieve stable locomotion, self-balancing, and real-time environmental mapping. SARAH utilizes a Raspberry Pi-based architecture running ROS, combining inverse kinematics, trot gait locomotion, and IMU-based stabilization for dynamic terrain adaptation. Sensor integration includes LiDAR for SLAM-based navigation and ultrasonic sensors for obstacle detection. The project demonstrates that advanced robotic functions, such as autonomous navigation and environment mapping, can be achieved using affordable, open-source components with a total cost under \$350. Experimental results highlight the robot's ability to maintain stability and navigate unstructured environments. Overall, SARAH serves as a proof-of-concept platform that enhances accessibility to quadruped robotics research and supports educational and practical deployment in hazardous environments.

KEYWORDS: Autonomous Robot, Quadruped Robot, Search and Rescue Autonomous Robot

Received 24 May., 2026; Revised 28 May., 2026; Accepted 02 June., 2026 © The author(s) 2026.

Published with open access at www.questjournals.org

I. INTRODUCTION

There are already a multitude of quadrupedal robots in market. The current development of robotic dogs has mostly been for entertainment or emotional support purposes, such as Tombot, which is a robot dog created to aid dementia and hospital patients with the psychological stressors of their situations. On the entertainment side, robot dogs are used mostly as toys for children, which can be seen on a scale from Paw Patrol toys to Sony's Aibo dog. The contribution of robot dogs to this industry has provided a more accessible route for pet ownership for all ages [1].

Another industry that has seen a rise in robotic dogs is emergency response. The New York City Fire Department purchased two robotic dogs in 2022 for aid in search and rescue efforts and has since expanded their arsenal and moved into water hose attachments for the robots, further diversifying the uses for robotic dogs in emergency response situations. Several robots are under development at universities such as MIT, Texas A&M, and John Hopkins University [2]. Most of this development focuses on remote autonomous capabilities – allowing the robot to locate targets without human assistance. Some, like the dog from Texas A&M, have memory capabilities so that the robot doesn't have to relearn a location it has already visited [3].

The goal of the SARAH (Search And Rescue Autonomous Helper) project is to design and build a functional four-legged quadruped robot that is capable of walking, self-stabilizing, and mapping its environment. The project addresses the challenge of stable locomotion on varied terrain while integrating sensors, custom kinematics, and autonomous mapping – demonstrating applied robotics concepts. SARAH serves as a proof-of-concept for deploying low-cost legged robots in search and rescue scenarios where wheeled robots would otherwise fail. SARAH represents a budget-friendly, open-source quadruped platform with practical capability for search and rescue and educational applications. At a total cost of approximately \$350, SARAH demonstrates that meaningful autonomous behaviors – including real-time SLAM mapping, IMU-based body stabilization, and obstacle detection – are achievable without the cost of commercial platforms such

as Boston Dynamics Spot (~\$75,000) or Unitree Go1 (~\$2,700). This makes legged robotics research accessible to students, hobbyists, and smaller research teams who would otherwise be priced out of the field.

II. LITERATURE REVIEW

Quadruped robots have emerged as a critical area of robotics research due to their superior mobility, terrain adaptability, and stability compared to wheeled or bipedal systems. These robots mimic animal locomotion and are particularly advantageous in unstructured environments, including disaster zones, industrial sites, and natural terrains. Over the past eight years, research has increasingly shifted from mechanical design toward full autonomy, integrating perception, control, planning, and learning systems [4].

(1) Mechanical Design and Locomotion Fundamentals

The development of quadruped robots begins with robust mechanical design and gait generation. Studies have emphasized leg configuration, actuation systems, and gait optimization (e.g., trot, pace, bound) to ensure dynamic balance and energy efficiency [5]. Recent literature highlights that quadruped robots provide an optimal trade-off between stability and mechanical complexity, compared with bipeds (less stable) and hexapods (more complex). Design research also addresses fault-tolerant gait mechanisms and climbing capabilities, making these robots suitable for hazardous applications such as inspection and exploration [4]. However, mechanical design alone cannot achieve autonomy. Instead, dynamic locomotion control systems must be integrated to manage real-time adaptations to varying terrains and disturbances.

(2) Control Systems and Motion Planning

Control strategies for quadruped robots have evolved significantly, moving from classical methods (e.g., PID, model-based control) to advanced model-predictive control (MPC) and learning-based approaches. Modern control frameworks integrate terrain perception with motion control, enabling robots to adapt locomotion strategies to environmental changes. These frameworks often rely on real-time terrain mapping and dynamic adjustment of foot placement and body posture [6]. A key advancement in recent years is the use of hybrid control architectures, combining: Model-based methods (for stability and predictability), and Data-driven learning methods (for adaptability). This combination improves both robustness and autonomy, particularly in dynamic and uncertain environments.

(3) Reinforcement Learning and Intelligent Locomotion

Reinforcement learning (RL) has become a dominant paradigm in quadruped locomotion research. RL enables robots to learn gait patterns and control policies autonomously through interaction with simulated or real environments [7]. Recent studies demonstrate that RL-based controllers: Reduce reliance on manual tuning; Improve generalization across terrains; and Enhance adaptability to disturbances. For example, RL frameworks using algorithms such as Proximal Policy Optimization (PPO) have shown improved stability and reduced oscillations in locomotion control. Similarly, imitation learning techniques have further accelerated development by enabling robots to replicate human or animal motion behaviors [8, 9]. Despite these advancements, challenges such as simulation-to-reality transfer (sim-to-real gap) and safety during learning remain critical research topics.

(4) Perception and SLAM for Autonomous Navigation

Perception systems are fundamental to achieving autonomy in quadruped robots. These systems rely on multi-sensor integration, including LiDAR, cameras, IMUs, and depth sensors, to interpret environmental conditions. A primary enabling technology is Simultaneous Localization and Mapping (SLAM), which allows robots to build maps while estimating their position in unknown environments. Research shows that: LiDAR-based SLAM provides high accuracy in mapping [10]; Vision-based SLAM enables richer semantic understanding; and Multi-sensor fusion improves robustness [11]. Recent work integrates SLAM with object detection and semantic perception, enabling advanced tasks such as human interaction and object manipulation. Additionally, perception-based controllers improve locomotion decisions by incorporating terrain classification. Nevertheless, perception challenges persist in low-visibility, reflective, or deformable terrains, requiring more robust sensor fusion techniques.

(5) Autonomous Navigation and Exploration

Autonomous navigation combines perception and control to allow quadruped robots to operate without human intervention. Modern systems implement SLAM-based navigation, path planning, and obstacle avoidance. Recent developments include: ROS2-based navigation frameworks integrating SLAM, path planning, and exploration algorithms [12]; Frontier-based exploration techniques for efficient autonomous mapping; Coverage path planning methods for systematic environment scanning [13]. Applications of these systems have been demonstrated in GPS-denied environments, such as underground mines, where autonomous systems successfully achieved consistent navigation and mapping [14]. Autonomous exploration remains a complex challenge due to dynamic obstacles, limited computational resources and real-time decision-making requirements.

(6) Application Areas

Autonomous quadruped robots are increasingly deployed in diverse domains: industrial inspection and maintenance, search and rescue operations, military and surveillance applications, construction and infrastructure monitoring, hazardous environments (e.g., nuclear plants, firefighting). A notable example is autonomous firefighting systems, where robots can detect and extinguish fires without human intervention [15]. Furthermore, commercial platforms such as Boston Dynamics Spot have accelerated technology transfer by enabling real-world deployment and experimentation.

In summary, the development of autonomous quadruped robots has progressed rapidly over the past decade, transitioning from basic locomotion platforms to intelligent, adaptive systems capable of complex autonomous operations. The integration of advanced control methods, reinforcement learning, perception systems, and autonomous navigation frameworks has enabled significant improvements in performance and applicability. However, achieving fully robust, general-purpose autonomy remains an open challenge. Continued interdisciplinary research spanning mechanical design, artificial intelligence, and robotics systems engineering is essential for the next generation of quadruped robots.

III. DESIGN OF QUADRUPED ROBOT

3.1 HARDWARE COMPONENTS

The following core components make up the SARAH platform:

- Raspberry Pi 3B+ – main computing platform running Ubiquiti ROS Noetic on Ubuntu 20.04
- RPLidar A1 – 360° laser rangefinder for real-time SLAM mapping (12m range)
- 12x MG996R Servo Motors – joint actuation, 3 per leg (hip, shoulder, knee)
- MPU-6050 IMU – accelerometer used for body stabilization
- 2x HC-SR04 Ultrasonic Sensors – front-facing obstacle detection
- 11.1V 3S 3300mAh LiPo battery – main power source
- 5V UBEC – clean regulated power supply for the Raspberry Pi
- 30A Buck Converter – stepped-down 6V regulated supply for servo motors
- Adafruit PCA9685 PWM Driver – provides all 12 PWM signals for servos

3.2 BILL OF MATERIALS

The final cost for the project was \$348.33 as shown in Table 1. Most components were pre-owned; listed prices reflect current market value at time of ordering. The Raspberry Pi 3B+ and RPLidar A1 were both purchased using the course budget of \$150.

Table 1 Bill of Materials for SARAH project

| Part | Quantity | Cost | Source |
|-------------------------------|----------|-----------------|-------------|
| 3D Printed Chassis | 1 | \$35.00 | Matt |
| Raspberry Pi 3B+ | 1 | \$49.19 | Amazon |
| RPLidar A1 | 1 | \$99.95 | Amazon |
| MG996R Servos | 12 | \$42.78 | Amazon |
| Adafruit PCA9685 PWM Driver | 1 | \$14.95 | Adafruit |
| MPU-6050 IMU | 1 | \$4.00 | Amazon |
| HC-SR04 Ultrasonic Sensors | 2 | \$12.00 | Microcenter |
| 11.1V 3S 3300mAh LiPo Battery | 1 | \$25.00 | Amazon |
| 5V UBEC | 1 | \$18.95 | Amazon |
| 30A Buck Converter | 1 | \$6.50 | Amazon |
| Misc wiring | | \$40.00 | |
| Total Cost | | \$348.33 | |

3.3 SPECIFICATIONS

SARAH uses 12 MG996R digital servo motors controlled by a PCA9685 I2C PWM driver at 50Hz. The 11.1V 3S LiPo battery powers the system through two rails – a 30A buck converter steps voltage down to 6V for the servo motors, while a 5V UBEC provides clean regulated power to the Raspberry Pi. The Pi in turn powers the LiDAR, ultrasonic sensors, and MPU-6050 IMU. With this power distribution, the robot achieves approximately 30 minutes of continuous operation per charge. Physically, the robot is approximately 13 inches long, 6 inches wide, and reaches roughly 10 inches in height while standing with legs extended. The diagonal trot gait runs at 50Hz with a full gait cycle of 1.6 seconds.

3.4 MECHANICAL DESIGN

SARAH's mechanical design is based on the open-source Spot Micro platform, customized and 3D printed in-house. The robot resembles a four-legged dog and consists of a central body, four hip assemblies, and four legs. Each leg attaches to a hip joint and includes an upper leg, lower leg (calf), and foot – providing 3 degrees of freedom per leg and 12 total moving joints. The frame was printed in PLA with an estimated filament cost of \$35 for prototypes and the final body.

3.5 ELECTRICAL CIRCUIT / ELECTRONICS DESIGN

The electrical system is organized around two separate power rails driven from a single 11.1V LiPo battery. The servo power rail runs through a 30A buck converter stepped down to 6V, feeding all 12 MG996R servos via the PCA9685 PWM driver. The logic power rail runs through a 5V UBEC providing clean regulated power to the Raspberry Pi, which in turn supplies 3.3V to the MPU-6050 IMU and 5V to the HC-SR04 ultrasonic sensors. The RPLidar A1 connects directly via USB. The PCA9685 and MPU-6050 share the Pi's I2C bus on GPIO pins 2 and 3, with the PCA9685 at address 0x40 and the MPU-6050 at address 0x68 as shown in Figure 1.

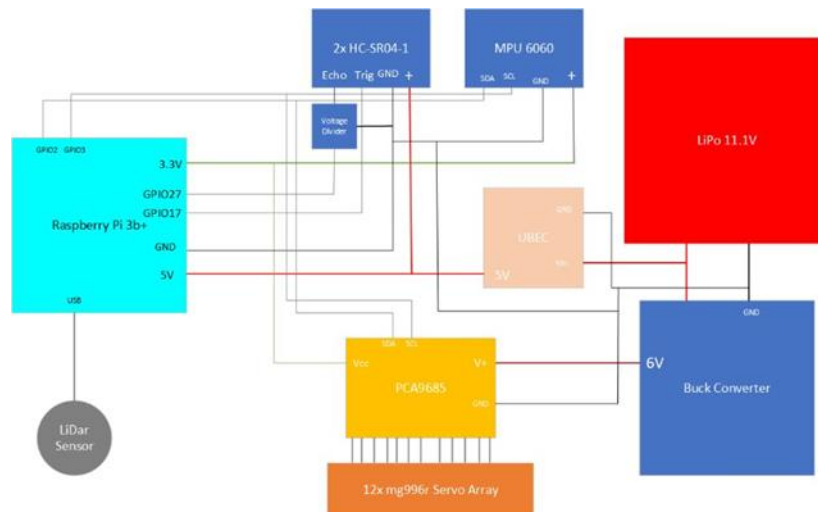


Figure 1: Electrical Wiring Diagram for SARAH

3.6 CONTROLLER DESIGN

(1) Software Architecture

All software runs on the Raspberry Pi 3B+ under ROS Noetic (Robot Operating System), a middleware framework that manages communication between independent software nodes via a publish/subscribe messaging system. The overall loop follows a Sense → Plan → Act architecture running at 50Hz. Sensor nodes read hardware and publish data to named topics; the gait and kinematics nodes consume that data and compute motor commands; the servo driver node executes those commands via I2C. A parallel SLAM pipeline runs independently, processing lidar scan data to build a real-time map of the environment.

(2) Inverse Kinematics

Each of SARAH's four legs has 3 degrees of freedom – hip (lateral swing), shoulder (fore/aft), and knee (elbow bending). To position a foot at a desired location in a 3D cartesian space, a custom analytical inverse kinematics (IK) solver node was implemented from scratch. The solver takes a target foot position (x, y, z) in millimeters and computes the required joint angles for all three servos using trigonometry and the law of cosines – the exact same mathematics used in industrial robot arms. The hip angle is solved using basic right-triangle geometry; the knee angle is solved via the law of cosines given the two link lengths and reach distance; and the shoulder angle is derived from that knee result. All angles are clipped to $\pm 90^\circ$. Left and right legs are mirrored using direction masks applied to output angles, allowing the same solver to work for all four legs as shown in Figure 2.

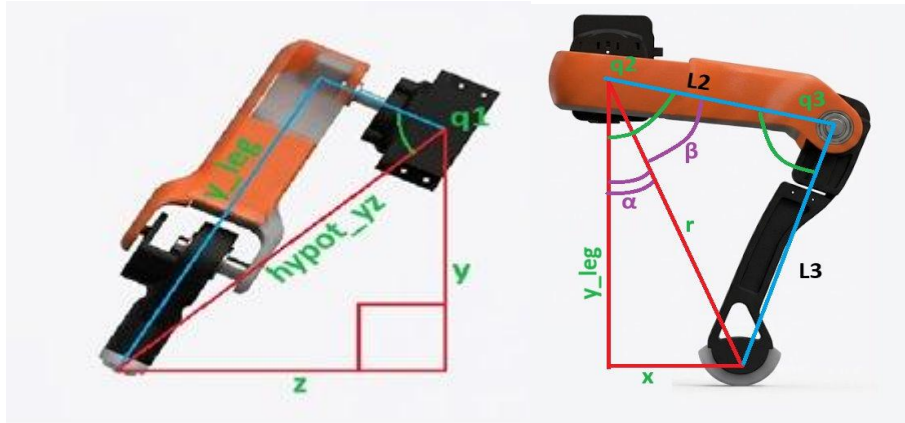


Figure 2: Inverse Kinematics for SARAH

(3) Gait Development

A trot gait was implemented after an initial crawl gait proved unstable under load. The trot moves diagonal leg pairs simultaneously – right-front with left-back, and left-front with right-back – which naturally balances weight distribution. Each leg alternates between two phases: a swing phase where the foot is lifted and moved forward in an arc trajectory, and a stance phase where the foot is planted and pushed backwards to propel the body. The swing arc uses a linear trajectory peaking at 25mm of clearance. To ensure smooth walking at varying speeds, a feedforward prediction algorithm places the foot ahead of its neutral position at the expected landing point, accounting for body velocity during the time the foot is in the air. The full gait cycle runs at 50Hz with a 1.6-second period.

(4) IMU Body Stabilization

The MPU-6050 inertial measurement unit provides roll and pitch angle data used to keep SARAH's body level during locomotion. Raw accelerometer readings are smoothed using a low-pass filter to eliminate vibration noise from the servos. The filtered angles are then used to compute a height correction for each foot based on lever-arm geometry and law of sines – the further a foot is from the robot's center, the larger the correction is applied when the body tilts. This correction is applied to each foot's target position before the IK solver runs, causing legs on the downhill side to extend and legs on the uphill side to retract, keeping the body approximately level. The system is calibrated on startup by averaging 50 samples to establish a zero reference for the robot's resting orientation.

(5) Hector SLAM Mapping

SARAH uses Hector SLAM to build a 2D map of its environment in real time using the RPLidar A1. Unlike traditional SLAM systems, Hector SLAM requires no wheel odometry – it determines the robot's position purely through scan matching, comparing each new lidar sweep against the accumulated map to determine movement. The lidar publishes scan data over ROS, which hector_mapping processes to produce a grid map at a 5cm resolution. Map data is transmitted over a Tailscale VPN network connection to a desktop PC running Rviz for visualization. The SLAM pipeline runs in parallel with the motion system and has no effect on the robot's control.

(6) Obstacle Detection

Two HC-SR04 ultrasonic sensors are mounted at the front of SARAH for last-resort collision prevention. Each sensor emits a pulse and measures time-of-flight to compute distance. A median filter over five readings removes spike noise. If either sensor detects an object within 10cm, the obstacle detection node overrides the keyboard velocity command with a stop signal, halting all movement until the path is clear. Due to a hardware voltage compatibility issue with the left sensor's echo pin, only the right sensor is used in the final implementation. Views of completed SARAH are shown in Figure 3-5.



Figure 3 View of completed SARAH, rear



Figure 4: View of completed SARAH, front



Figure 5: View of completed SARAH, front

(7) Dashboard

A sensor dashboard was implemented as a stretch goal using Python and the Kivy GUI framework. The dashboard provides a unified view of SARAH's sensor data – displaying live roll and pitch angles from the IMU, distance readings from the ultrasonic sensors, and a live feed of the Hector SLAM grid map as shown in Figure 3. All data is accessed by subscribing to the same ROS topics used by the robot's control nodes, requiring no modifications to existing code. The dashboard runs on a separate laptop connected to the robot over the Tailscale network.

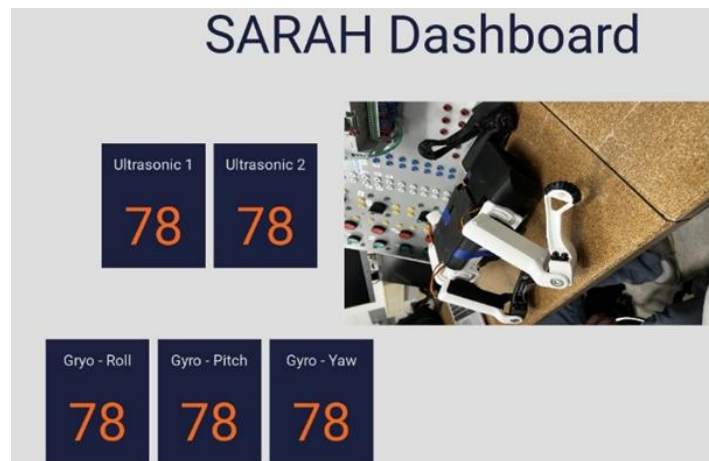


Figure 6: SARAH dashboard

3.7 TROUBLESHOOTING

Several significant challenges were encountered and resolved throughout the development:

- Startup instability – The robot would behave erratically on startup due to the IMU calibrating before the servos had fully settled. Resolved by adding a startup delay and averaging 50 accelerometer samples after the robot reached its neutral standing position.
- Crawl gait instability – The initial crawl gait produced excessive rocking and was mechanically unstable under servo load. Switched to a diagonal trot gait which distributes weight more evenly and proved significantly more stable.
- MPU-6050 library bug – The `mpu6050-raspberrypi` Python library returned all zeros because it did not send the required wake-up command to the sensor on startup. Resolved by switching to direct register access via the `smbus` library.
- Buck converter current limiting – The original power setup caused servo brownouts under load. Upgraded to a 30A buck converter and added a dedicated UBEC for the Pi to separate logic and servo power rails.
- Left ultrasonic sensor – The HC-SR04 echo pin outputs 5V but the Raspberry Pi GPIO only tolerates 3.3V, causing unreliable readings. The left sensor was disabled in software and only the right sensor is used.
- ROS network connectivity – connecting Rviz on a desktop PC to the robot over a standard network was unreliable. Resolved using Tailscale VPN which provides very stable peer-to-peer connectivity between the Pi and desktop regardless of network configuration.

3.8 DISCUSSION AND REFLECTION

The project was divided among four team members. Paige handled soldering, ultrasonic sensor driver implementation, trot gait tuning, and the sensor dashboard. Matt led hardware assembly, component selection, and all ROS software architecture including inverse kinematics, gait development, IMU stabilization, and SLAM integration. Zach configured the Raspberry Pi and contributed to gait tuning. Isaac contributed to project documentation, background research, and power system design. All four members participated in testing, validation, reports and presentations throughout the semester. From a teamwork perspective, the project highlighted the importance of initial planning, the need for thorough research before implementation, and just how mechanically and mathematically complex four-legged locomotion is.

This project provided hands-on experience with a broad range of robotics concepts well beyond the scope of typical coursework. Key learning outcomes included a deep understanding of 3-DOF inverse kinematics and how analytical geometry translates into robot joint control; the principles of legged locomotion and gait design including swing and stance phases, diagonal trot timing, and feedforward foot placement prediction; sensor fusion between IMU data and kinematic control for real-time body stabilization; and simultaneous localization and mapping using a laser rangefinder without odometry. The ROS middleware framework was essential to the entire project, demonstrating how complex multi-sensor robotic systems are architected in industry and research. More importantly, the project illustrated the gap between simulated or theoretical robotics and the practical challenges of real hardware – servo mismatches, power distribution, sensor noise, and network communication all required real engineering solutions.

The most immediate improvements would focus on gait quality and sensor integration. Reducing foot drag during the trot, improving turning stability at speed, and implementing adaptive step timing based on actual

foot contact detection would significantly improve the locomotion naturalness. On the sensing side, fully integrating the dashboard with live SLAM map updates and reliable dual-sensor obstacle detection would complete the original design intent. Repositioning the ultrasonic sensors to provide downward-facing coverage would add cliff and stair detection – an important safety feature for search and rescue applications. Additionally, a proper wiring harness would improve reliability over the current off-the-shelf, prototype-grade connections.

IV. CONCLUSION

The natural progression of SARAH is toward greater autonomy and terrain capability. Upgrading to higher-torque digital servos with position feedback, combined with a new Jetson Nano compute platform, would enable much faster control loops, computer vision integration, and the deployment of machine learning-based locomotion trained in simulation. Adding force-sensitive resistors to the feet would enable adaptive gait timing based on actual ground contact rather than fixed timers – a very significant step toward more natural walking. Longer term, the SLAM mapping capability positions SARAH well for full autonomous navigation using the ROS navigation stack, which would allow the robot to plan paths, avoid obstacles, and operate without human teleoperation – fully implementing the search and rescue function.

REFERENCES

- [1]. Criollo Rodríguez, E., et al. (2025). A review of quadruped robot control and perception based on augmented reality. IEEE ETCM Conference.
- [2]. Fan, Y., Pei, Z., Wang, C., Li, M., Tang, Z., & Liu, Q. (2024). A review of quadruped robots: Structure, control, and autonomous motion. *Advanced Intelligent Systems*.
- [3]. Misenti, S., et al. (2025). Experimental evaluation of commercial quadruped robots: Stability and performance. *International Journal of Intelligent Robotics Applications*.
- [4]. Majithia, A. M., et al. (2024). Design, motions, capabilities, and applications of quadruped robots: A review. *Frontiers in Mechanical Engineering*, 10.
- [5]. Li, Q., et al. (2025). Quadruped robots: Bridging mechanical design, control, and applications. *Robotics*, 14(5), 57.
- [6]. Kang, H., Wang, Z., Wang, K. et al. Motion control of quadruped robots in complex terrain. *J. King Saud Univ. Comput. Inf. Sci.* 37, 301 (2025). <https://doi.org/10.1007/s44443-025-00292-z>
- [7]. Gurram, M., Uttam, P. K., & Ohol, S. S. (2024). Reinforcement learning for quadrupedal locomotion: Current advancements and future perspectives. *arXiv preprint*.
- [8]. Escudero-Villa, P., et al. (2025). Quadruped robot locomotion based on deep learning rules. *Engineering Proceedings*, 87.
- [9]. Mirza, K. Z., & Singh, S. (2025). Imitation learning for legged robot locomotion: A survey. *Frontiers in Robotics and AI*.
- [10]. Chen, J., & Dellaert, F. (2022). A1 SLAM: Quadruped SLAM using onboard sensors. *arXiv preprint*.
- [11]. Chen, G., & Hong, L. (2023). Research on environment perception system of quadruped robots based on LiDAR and vision. *Drones*, 7(5), 329.
- [12]. Brekke, V., Berge, E. O., Dybdahl, E., Singh, J., & Tyapin, I. (2026). ROS2-driven navigation and sensor platform for quadruped robots. *Robotics*, 15(4), 70.
- [13]. Raheema, J. Y., et al. (2024). Autonomous exploration and mapping on quadruped robots. *ISRR Proceedings*.
- [14]. Gao, Y., & Awuah-Offei, K. (2026). Efficient autonomous navigation of quadruped robots in underground mines. *arXiv preprint*.
- [15]. Baird, C., & Nokleby, S. (2026). Development of an autonomous firefighting quadruped robot with a novel vision-based fire segmentation system. *Letters in Robotics*.