Autonomous Hazard Zone Mapping and Worker-Aware Safety System for Construction Sites Using SLAM

Smrithi Srinivasan
School of Computer Science
University of Birmingham
Student ID: 3028838

Abstract—Construction sites are dynamic environments where rapid hazard detection is critical for worker safety. This project uses Simultaneous Localization and Mapping (SLAM) to present an autonomous hazard zone mapping and worker-aware safety system. This mobile robot was made in a simulated construction site to generate maps, find static obstacles with LiDAR, and keep track of workers' positions at all times. A dynamic safety zone mechanism is used, which changes the protective radius based on the robot's speed. This makes the robot more responsive to different types of motion. Hazard detection integrates both obstacle proximity and worker distance to produce behavioral adjustments through a control strategy. Real-time logging of the robot's pose, hazard flags, and the minimum distance between workers makes it possible to do structured experimental evaluation. Results show that adaptive safety zoning raises the minimum distance between people and improves how quickly they respond to hazards compared to fixed safety thresholds. The framework demonstrates the potential of integrating SLAM with dynamic safety logic for improved construction site monitoring. Index Terms—SLAM, hazard detection, construction safety, dynamic safety zone, worker tracking, mobile robotics.

1 Introduction

Construction sites are dynamic, frequently evolving environments that might pose hazards, with workers navigating heavy machinery, provisional structures, and altering configurations. Despite rigorous safety regulations, accidents may still occur when dangers are not consistently identified in real time. Traditional safety methods, such manual inspections and static warning signs, are often limited due to their inability to provide continuous environmental monitoring or swiftly adjust to changes. This underscores the imperative for more sophisticated systems capable of detecting threats and responding swiftly to prevent accidents.

Simultaneous Localization and Mapping (SLAM) enables a robot to build a map of an unfamiliar environment while also localizing itself within that environment. By combining the use of SLAM technology with sensors and adaptive control algorithms, it is possible to develop a robot that maps a work environment while also identifying possible risks and methods to avoid them.

This project aims to develop a SLAM-based autonomous system for safety monitoring on construction sites. The robot employs LiDAR and motion data to incessantly monitor its location and construct a site map. A hazard detection system analyzes LiDAR data to detect barriers and simultaneously monitors the real-time locations of workers to ensure safe distances are maintained. The data on barriers and worker placements is integrated into a unified "hazard alert" system to ensure the identification of all threats.

To enhance safety, the system employs a dynamic safety zone surrounding the robot that adjusts according to the robot's velocity, resulting in a bigger safety zone at higher speeds. Upon the entry of a worker or an impediment within the safety zone, the robot autonomously modifies its behavior to avert a potential accident. The system additionally documents the robot's actions, possible hazards, and nearest proximity to personnel.

This project seeks to enhance safety and efficiency on construction sites through the integration of real-time mapping, intelligent hazard detection, and adaptive safety protocols.

2 System Overview

The proposed system is an automated hazard-aware navigation system operating on a mobile robot equipped with LiDAR and odometry in a construction zone. The robot sustains a mapping of its environment, identifies risks associated with both stationary obstacles and proximate workers, and modifies its movement while recording safety-related data.

2.1 Hazard detection, worker tracking, and dynamic safety zone

A hazard-detection layer analyses LiDAR sectors to assess obstacle distances and detect potential collision threats. A worker-tracking system makes use of worker models while also updating each worker's location at every control step, keeping individual trajectory histories, and calculating real-time distances between robots and workers. A dynamic, velocity-dependent safety zone is created around the robot, which adjusts dynamically based on its current speed, and is used to identify surrounding workers as dangers as they enter this radius.

2.2 State machine, logging, and experimental scenarios

A structured motion state machine integrates obstacle and worker-hazard information to select behaviors such as ”FORWARD” and ”TURN,” modulating linear and angular velocity to avoid risky conditions while advancing through the environment. During each control cycle, the controller documents simulation time, robot pose, LiDAR sector distances, worker distance metrics, hazard flags, and the current state in CSV logs, facilitating offline analysis, parameter optimization of safety thresholds and velocities, and consistent assessment across scripted scenarios involving multiple moving workers and hazards.

3 Hazard Detection Pipeline

The hazard detection pipeline converts raw sensor data and worker positions into a single unified hazard signal that directly drives the robot's behaviour. It combines static obstacle checks from LiDAR with worker-aware checks based on robot–worker distance.

3.1 Sensor preprocessing

- At each control step, the LiDAR point cloud is partitioned into three angular sectors: LEFT, FRONT, and RIGHT.
- For each sector, the controller extracts a representative distance (for example, the minimum range) to describe how close the nearest obstacle is in that direction.

3.2 Static hazard detection

- A static obstacle hazard is raised when objects are detected too close in any relevant sector.
- In particular, a danger flag is triggered if the distance in the FRONT sector falls below a predefined threshold ”HAZARD DISTANCE,” i.e., when

  $$"FRONT" < "HAZARD - DISTANCE"$$

  .

3.3 Worker detection and tracking

- Workers are represented as separate entities (WORKER1–WORKER11) with known ground-truth positions that are updated at every control cycle.
- For each worker, the controller computes the Euclidean distance to the robot in the horizontal plane using

  $$d = \sqrt{(x_r - x_w)^2 + (y_r - y_w)^2}$$

  where (x_r, y_r) is the robot position and (x_w, y_w) is the worker position.
- A worker-specific hazard flag, worker hazard, is set when any worker's distance falls within the current safety radius.

3.4 Unified hazard logic

- The final hazard signal provided to the state machine is the logical combination of both sources.
- The robot treats the situation as unsafe and initiates avoidance behaviour whenever ”Danger OR Worker hazard”
- valuates to true, ensuring that both static obstacles and nearby workers can independently trigger a safety response.

3.5 Dynamic Safety Zone Logic

The dynamic safety zone is a speed-aware safety mechanism that enlarges or shrinks the robot's protection radius depending on how fast it is moving. This allows the robot to be more conservative at higher speeds while remaining efficient at lower speeds.

3.6 Speed-dependent safety radius

The controller computes a safety radius that scales linearly with the robot's current speed using a minimum and maximum radius:

$$\text{Dynamic radius} = \text{MIN} + \frac{\text{speed}}{\text{MAX SPEED}} \times (\text{MAX} - \text{MIN})$$

At low speeds, the radius stays close to MIN, while at higher speeds it approaches MAX, giving the robot more space to react.

3.7 Worker hazard condition and state change

For each worker, the robot compares the current robot–worker distance ”dist” with this dynamic radius.

- If

  $$"dist" < "dynamic_radius", then worker_hazard = True$$

- When worker hazard is true, the motion state machine transitions from forward motion into an avoidance state such as TURN, causing the robot to slow down or steer away from the worker.

3.8 Benefit over static thresholds

By replacing a fixed safety distance with a speed-dependent radius, the system maintains larger buffers when the robot is moving quickly and avoids overly conservative behavior when it is nearly stationary. This makes the hazard response more proportional to risk and improves safety compared to static thresholds, especially in dynamic, crowded environments.

4 Configuration of Scenario in Webots

The experiments were carried out in a simulated construction site environment in Webots, with static barriers and moving workers that cross the path of the robot.The robot operates via a basic state machine with two primary modes: FORWARD and TURN, transitioning upon the detection of dangers. At every step of the simulation, the controller records the attitude of the robot, the hazard indicators created by the LiDAR scanner, the worker hazards, and the distance to the nearest worker, making it easier to analyze the responses of the robot to the dynamic workers.

5 Experimental Design and Metrics

5.1 Experimental Conditions

The system undergoes assessment through a series of controlled simulation runs that methodically alter essential safety parameters. The robot's forward velocity is altered throughout trials to examine the impact of varying speeds on danger contacts and worker proximity. The minimum and maximum safety radius parameters are modified to evaluate cautious versus aggressive dynamic safety zones. Ultimately, each configuration undergoes testing in scenarios involving many mobile workers traversing diverse routes, thus generating varied and reproducible hazard situations surrounding the robot.

5.2 Performance Metrics

Performance metrics are documented at each timestep throughout the simulation and then recorded in a CSV log for offline analysis. Each run's log records the minimum distance to any observed worker over time, the frequency of activated hazard flags (including both static obstacle hazards and worker dangers), and the overall duration spent in hazard-related states. Furthermore, all state transitions of the motion state machine (e.g., FORWARD → TURN → FORWARD) are documented, facilitating a thorough analysis of the robot's responses to encroaching workers and obstacles.

6 Results and Discussion

The preliminary results indicate that the dynamic safety zone logic improved how the robot managed proximity to workers. Across the tested runs, the minimum distance maintained between the robot and the nearest worker increased when using the speed-dependent safety radius compared to fixed thresholds. Hazard detection events were also triggered earlier at higher robot speeds, meaning the controller raised danger and worker-hazard flags sooner, giving the state machine more time to initiate turning or slowdown actions. The worker-aware component of the hazard logic further reduced the number of very close interactions, since workers could trigger avoidance behavior even when static obstacle distances were still above the LiDAR-based threshold. Continuous logging of pose, hazard flags, and worker distance across parameter sweeps enabled a basic sensitivity analysis, showing how changes in speed and safety radius settings affected both safety margins and the frequency of hazard states.

Despite these encouraging trends, the current evaluation has important limitations. Worker positions are obtained from ground-truth simulation data, so the system does not yet include realistic perception noise, occlusions, or misdetections that would arise with camera- or LiDAR-based people detection in the real world. All experiments are conducted in a simulator with idealized sensing and controlled worker trajectories, which may not fully capture the unpredictability and clutter of an actual construction site. Translating the approach to hardware will require reliable real-time worker detection and tracking, robust sensor calibration, and careful tuning of safety parameters to account for delays, actuation limits, and uneven terrain. There will also be practical challenges around validating safety behavior under real regulatory and site constraints, including how conservative the safety zone must be and how to integrate the system with existing site procedures.

7 Conclusion

This work demonstrates the power of SLAM-based mapping with a hazard detection pipeline on a mobile robot to safely navigate a congested construction site. Dynamic worker-aware safety regions enhance reaction to varying speed and crowd dynamics, enabling earlier hazard detection and avoidance than static approaches. Logging of pose, hazard status, and worker proximity provides valuable insights for safety analysis and optimization, which has a high potential for real-time execution on a construction robot.

The complete source code, Webots world files, and documentation for this project are available in the following GitHub repository:
https://github.com/smrithi699-sgss/Autonomous-Hazard-Zone-Mapping-and-Worker-Aware-Safety-System-for-Construction-Sites-Using-SLAM/tree/main
This contains the read.me file and the csv log file.

8 Reference

- Group 2, "Autonomous hazard zone mapping and worker-aware safety system for construction sites using slam," 2025, project Proposal, Uni- versity of Birmingham.
- J. Teizer, T. Cheng, and Y. Fang, "Real-time location tracking for construction safety," Automation in Construction, vol. 19, no. 4, pp. 492– 504, 2010.
- I. Jeelani, K. Han, and S. Albert, "Real-time vision-based worker localization and hazard detection in construction," Automation in Construction, vol. 129, p. 103816, 2021.
- H. Zhang, M. Golparvar-Fard, and T. Cheng, "Automated construction site safety monitoring using SafeBIM models and real-time location tracking systems," in Proc. 41st Int. Symp. Automation and Robotics in Construction (ISARC), 2024, pp. 1–8.
- A. Rahman and L. Petersson, "Safety considerations for autonomous modular robotics in collaborative environments," Robotics and Autonomous Systems, vol. 156, p. 104207, 2022.
- Y. Li, J. Wang, and C. Chen, "Periodically collected real-world construction dataset for SLAM (ConSLAM)," in Proc. IEEE Int. Conf. Robotics and Automation (ICRA), 2024, pp. 1–7