Design and Development of an Autonomous Mobile Patrol Robot

Jonathan Selvanathan, Haoxiang Lang*
Department of Automotive and Mechatronics Engineering, Ontario Tech University, Oshawa, Canada
*haoxiang.lang@ontariotechu.ca

Abstract—Autonomous mobile robots (AMRs) are especially useful for completing tasks that humans find repetitive or unsafe. An industry that would benefit from the integration of an AMR is security, where a robot could be tasked with patrolling areas to report potential concerns, hazards or violations based on the environment. To research and experiment with these possibilities, a versatile robotic platform capable of operating outdoor and indoor amongst cars and humans is required. Commercial and open-sourced AMRs are available and range in sizes; however, there is no existing mobile robotic system in the market that can be directly adopted in the application with all required capabilities. Therefore, a newly designed mobile robotic system, specifically for patrol and policing applications, is proposed in this paper. It is a robust 1/3rd scale car-like robot with a modular, and reproducible hardware architecture. The design and development of the SPR is presented as it aims to expand the possibilities of research and validation in simulation and experimentation. The test result of the designed robot showed its ability to navigate and localize in an environment, displaying its readiness for autonomy.

Keywords-component; Autonomous vehicle; design; hardware; localization

I. INTRODUCTION

Autonomous mobile robots (AMRs) have become increasingly popular in the industry for their ability to complete repetitive tasks continuously, explore unknown environments and expand on what was previously thought possible. For these inherent characteristics, AMRs have become desirable for use in sectors such as equipment inspection [1], agriculture [2] and space exploration [3]. Robots deployed in these settings complete a task that is either unfavorable or impossible for humans to undertake, allowing them to avoid harm. For these reasons, an interesting application for AMRs to be introduced to is security patrol in both outdoor and indoor settings such as parking lots, shopping malls and school campuses. A mobile robot could ensure safe environments for individuals in these areas by conducting regularly scheduled excursions to patrol, detect and report potential misdoings. Such an application emphasizes the benefits of AMRs by ultimately improving the quality of life and well-being of society.

The current challenge and limitation of autonomous surveillance mobile robots are the lack of commercialized systems available in the market. Therefore, developing such a system including hardware and software becomes crucial for the applications of surveillance and policing. There are existing hardware options found in the market including full-scale autonomous vehicles, or smaller-scale research robots. The former is a costly solution to the problem and is limited to outdoor navigation due to its size. The latter is suitable for short-term use; however, it is required for such a patrol robot to be medium to large-sized and easily identifiable by both people and vehicles to avoid potential collisions. Although some commercial robotic companies sell AMRs that fit this requirement, their software is not made open-source and therefore not feasible for use within academia. Nonreconfigurable hardware and software makes integration of new equipment and modifications of internal software difficult. To bridge this gap, the purpose of this project is to design and develop a system including hardware and software that can be utilized as a security patrol robot (SPR). The designed robot in this paper is a 1/3rd scale car-like vehicle for the purpose of research and development of autonomy in this field. The hardware is designed, and fabricated in-house with the main concept of being simplistic, modular, and reproducible. Finally, the low-level controllers and high-level perception and navigation were also developed in the lab to achieve autonomy.
This paper details the design and development of the mobile robot and is organized as follows. Section II reviews the related work in mobile robots. Section III presents an overview of the mechanical design of the robot. Section IV details the designed electrical hardware architecture. Section V presents the low-level controllers of the vehicle in the form of its kinematic model and steering position control. Experimentation and validation of the capability of the system for localization are shown in Section VI. The paper is concluded in Section VII.

I. RELATED WORK

Over the past decade, industry, academia and the open-source community (e.g., ROS community) have made autonomous mobile robots possible. A well-known platform is the Husky Unmanned Ground Vehicle (UGV) developed by Clearpath Robotics [4] which is a medium-sized robot capable of outdoor and indoor navigation. The Husky has a skid-steer drivetrain making it highly maneuverable, and suitable for various research goals such as localization and navigation methodologies. It is a robust and easy-to-use platform making it a popular choice amongst roboticists. However, its software is better suited for general research purpose of mobile robotics. It can’t be easily used in mobile robot applications such as patrolling and policing. In addition, since being a turn-key solution, its electrical hardware is not openly available for others to customize.

Researchers as of recently have made custom AMRs open-sourced with detailed documentation for others to recreate and use towards education or research-driven purposes. FITENTH [5] is an example of such, who’s robot is created by modifying a readily-available 1/10th scale RC car with the hardware required for autonomy. This platform with its Ackermann steering geometry is beneficial for those looking to apply research in the area of autonomous racing or car-based navigation on physical hardware. However, it doesn’t have a full-scale vehicle available. Similar open-sourced platforms have since been created by also modifying a 1/10th scale RC-Car. MuSHR [6] for instance, aims to provide a more cost-effective solution by using only off-the-shelf components. Another is XTENTH-CAR [7], whose goal is to equip the car with state-of-art hardware to facilitate the research of computationally expensive algorithms. Chronos [8] on the other hand, builds off a 1/28th scale car with low-cost hardware to enable the development of multi-agent experiments in areas where space is limited. Finally, a more unique vehicle is proposed in [9] which has 8-wheel-drive and 8-wheel-steering. Its size is comparable to the Husky UGV, but provides drive and steering geometry that could be well-suited for off-road environments.

These platforms are not favorable for use in an outdoor setting where other vehicles are present due to their small-scale. In addition, their documented hardware is based on low-voltage systems (5V to 11.1V) which are best for their intended use-case but cannot be extended to larger scale platforms that operate on 48V. A demand is required for a vehicle of medium to large scale that can operate amongst vehicles and people.

II. MECHANICAL DESIGN

In this section, the design and development of SPR’s mechanical platform including its chassis design, drivetrain configuration, actuators and general specifications are presented.

A. Chassis Design

The chassis of the SPR is a custom welded steel frame, highlighted in blue within the isometric and bottom view of robot displayed in Fig 1 and Fig 2. As shown in Fig 2, the front and rear axle assemblies of the robot are mounted to pivot points about the steel chassis through an intermediate link. The respective axles are suspended using gas shocks to complete the vehicle's suspension. The steel frame and compliant suspension allow for the robot to be robust against adverse conditions found when driving either indoors or outdoors. Besides this, the chassis is also used to house several terminal boxes around the robot, which provides ample space for the electronic components of the SPR to be mounted. In addition, several sensors are mounted atop and to the sides of the chassis. The details of the robot’s sensor configuration are detailed in Section IV-C.
B. Drivetrain Design

The drivetrain of the SPR is comparable to a rear-wheel drive (RWD) car where the rear wheels are actuated for propulsion while the front wheels are actuated for steering. The rear axle of the mobile robot is driven by a brushless direct current motor (BLDC) coupled to a 22:1 differential gearbox. The gearbox allows for each rear wheel to spin at varying speeds from each other. This in turn improves the SPR’s traction performance during a turn as the outer wheels are able to spin at a greater rate than the inner wheels. If the gearbox were to be eliminated and replaced with directly driven wheels, the SPR would see increased lateral movement on turns (slip) thus negatively affecting the localization performance of the robot.

The front axle, in contrast, holds the passively propelled wheels of the SPR. Both wheels are connected together by a steering linkage. This linkage is driven by a DC motor where its output shaft is perpendicular to the front axle. The wheels can turn a maximum of 35 degrees in either direction. Each axle of the vehicle has rotors mounted to both ends for the wheels to be fastened to with lug nuts.

C. General Robot Specifications

The physical dimensions of the SPR are comparatively large to most research platform robots available for purchase. The wheelbase, track width and max height are 0.85m, 0.585m and 1.74m. The large size of the robot has several advantages to its intended use case of outdoor security with the most significant being ease of identification by both bystanders and surrounding vehicles. With being an easily spotted vehicle ensures the probability of an accident is decreased. Furthermore, the heavy mass decreases the chances of theft or unwanted relocation of the vehicle when unsupervised. The rated speed of the SPR can be determined upon further analysis of the aforementioned rear drive actuator. With the BLDC motor rated for 3000RPM at 48V/1000W, the rated linear velocity of the vehicle is calculated to be 6.35m/s with consideration of the 22:1 differential gearbox. The robot however, will not be driven at these speeds, and is limited through software mentioned in section V.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Measurement</th>
</tr>
</thead>
<tbody>
<tr>
<td>Wheelbase</td>
<td>0.85m</td>
</tr>
<tr>
<td>Track Width</td>
<td>0.585m</td>
</tr>
<tr>
<td>Height</td>
<td>1.74m</td>
</tr>
<tr>
<td>Tire Radius</td>
<td>0.23m</td>
</tr>
<tr>
<td>Rated Speed</td>
<td>6.35m/s</td>
</tr>
<tr>
<td>Max Steering Angle</td>
<td>35 degrees</td>
</tr>
</tbody>
</table>

III. ELECTRICAL HARDWARE ARCHITECTURE

Prior to the development of the SPR’s hardware architecture, the following design requirements were considered:

- Components are to be selected with availability in mind for ease of reproducibility.
- Safety of the components and more importantly the end user should be strongly considered.
- Hardware with interfaces developed for the robot operating system (ROS) is preferred to leverage the benefits of the open source community.

The proposed architecture embodying the desired design goals is detailed in Fig. 3. The result of adhering to these goals is a framework that can be adopted into a variety of mobile robotic applications. The following sections will provide insight to the various modules that constitute the SPR hardware architecture.

A. Power Distribution

The first stage of the electrical design is the power distribution. The power source used to operate all peripherals on the SPR is a 46.8V 80Ah lithium-ion battery with an integrated on-off switch, providing significant run time for outdoor excursions. A 40A fuse is used as the main breaker of the system for overcurrent protection. In order to use this high voltage with low voltage hardware, 12V and 5V DC-DC converters are implemented into the system. Each converters output is wired to its own power distribution board that each contain 12 additional outputs. Each output in this board is wired to a blade fuse that can be selected based on the equipment is to be connected to. The benefit of using such a board is the ability to rapidly test desired peripherals and ensure they are protected from unwanted overcurrent.

B. Motor Controllers

The controller selected for the rear BLDC drive motor is the Vedder Electronic Speed Controller (VESC) version 6. This open-source controller has support for several control schemes including speed and position. To achieve closed-loop control of the motor with the VESC and improve the overall accuracy of the SPR, feedback is sourced from an ABI encoder mounted to the motor’s rear output shaft. This has also in turn improved the performance of the controller to ramp-up and hold the motor at

![Figure 3. SPR electrical hardware architecture block diagram.](image)
low-speeds, common for a mobile robot application such as security patrol.

Control of the steering actuator on the other hand is accomplished using a DC motor controller known as the Roboclaw Solo. A 12-bit resolution absolute position encoder is mounted offset to the axis of rotation of the motor, and is linked using a pulley system as shown in Fig. 2. To interface with both the DC motor controller and encoder, a Teensy 4.1 microcontroller (MCU) is used. A control system is developed and deployed on this MCU to achieve position control of the steering linkage. The working principle of the position control algorithm is further discussed in Section V-B.

To provide added safety to the system, normally-closed (NC) relays are wired in series between the motor controllers and their respective power sources. Each relay is provided a 12V power line to actuate their internal coil and wired in-between is a common emergency stop button. This button provides a means of cutting power to the actuators in a runaway situation while ensuring power is still distributed to various other hardware. This configuration is optimal in a robotic platform where it is vital to be able to stop the robot from causing any harm to its operating environment, but also ensure data can be examined on the onboard PC to possibly determine the root cause of the erratic movement.

C. Sensors

Fig. 1 highlights in total seven exteroceptive sensors that are placed around the SPR to facilitate for experimentation with the platform. Two of the sensors are classified as 2D Light Detection and Ranging (LiDAR) and are mounted towards the front of the vehicle. With the purpose of detecting obstacles in the robot’s path, one of the LiDAR’s (SICK LMS141) is mounted parallel to the ground plane while the other (SICK TiM561) is mounted on a 15-degree decline. The downwards field of view (FOV) ensures obstacles below the elevation of the horizontally mounted LiDAR are detected. The next four sensors on the SPR are 5-megapixel (MP) internet protocol (IP) cameras. These cameras are mounted on the sensor tower within the robot such that they face the front, back, left and right of the vehicle. The combination of these cameras can generate a panoramic view of the SPR’s surrounding environment which can be leveraged in a variety of computer vision applications including object detection and classification, scene analysis and robot localization algorithms. Each of these sensors data output is wired to a network via Ethernet to allow for access over a private network. The final sensor on the SPR is a pan-tilt-zoom (PTZ) camera and is mounted on the top surface of the sensor tower.

D. Communication Protocols

The central processing unit of the robot is a fan-less industrial computer, chosen for its small form factor and large number of inputs and outputs (I/O). As shown in Fig. 3, the motor controllers, MCU and sensors are connected to the computer with their respective communication interfaces including Universal Serial Bus (USB), and Ethernet via network switch. The computer’s operating system is a Linux distribution known as Ubuntu 20.04 in order to run the desired software framework for the vehicle known as the robot operating system (ROS) [10].

<table>
<thead>
<tr>
<th>Component</th>
<th>Specification</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>BLDC Motor Controller</td>
<td>VESC 6 75V</td>
<td>[12]</td>
</tr>
<tr>
<td>ABI Encoder</td>
<td>1024 Pulse Per Revolution</td>
<td>[13]</td>
</tr>
<tr>
<td>DC Motor - Steering</td>
<td>12V 220W Motor</td>
<td>[14]</td>
</tr>
<tr>
<td>DC Motor Controller</td>
<td>IMC410 - Roboclaw Solo</td>
<td>[15]</td>
</tr>
<tr>
<td>Absolute Position Encoder</td>
<td>RS485 Output</td>
<td>[16]</td>
</tr>
<tr>
<td>Microcontroller</td>
<td>Teensy 4.1 MCU</td>
<td>[17]</td>
</tr>
<tr>
<td>SICK LiDAR - Horizontal</td>
<td>LMS141</td>
<td>[18]</td>
</tr>
<tr>
<td>SICK LiDAR - Downwards</td>
<td>TiM561</td>
<td>[19]</td>
</tr>
<tr>
<td>IP Camera</td>
<td>5 Megapixel</td>
<td>[20]</td>
</tr>
<tr>
<td>Pan-Tilt-Zoom Camera</td>
<td>1080P 30 frames-per-second</td>
<td>[21]</td>
</tr>
<tr>
<td>Fan-less Computer</td>
<td>Nuvo-5002LP Intel i7-6700TE</td>
<td>[22]</td>
</tr>
<tr>
<td>Network Switch</td>
<td>16-Port Gigabit</td>
<td>[23]</td>
</tr>
<tr>
<td>Travel Router</td>
<td>GL-AR7505-EXT</td>
<td>[24]</td>
</tr>
<tr>
<td>Battery</td>
<td>18650 46.8V 80Ah Li-Ion</td>
<td>[25]</td>
</tr>
<tr>
<td>12V DC-DC Step Down Converter</td>
<td>SD-500L-12</td>
<td>[26]</td>
</tr>
<tr>
<td>5V Step Down DC-DC Converter</td>
<td>48V to 5V</td>
<td>[27]</td>
</tr>
<tr>
<td>DC Power Distribution Board</td>
<td>12 Position and Fuse</td>
<td>[28]</td>
</tr>
</tbody>
</table>

a. Where possible, the exact part number used is provided along with the site of purchase in the “Reference” column. If not, that part is no longer available for purchase. Therefore, relevant specifications are detailed along with a link to a similar product.

IV. LOW-LEVEL CONTROL

In this section, the robot’s underlying control methodology is presented. The main focus of the developed controllers is to provide a foundation for autonomy to be built off from. This entails low-level operation of the SPR in the form of accurate velocity and joint position control. Towards the realization of this goal, the kinematic model of the car-like robot is introduced, presenting the importance of such control and its use in odometry estimation. The control scheme to achieve closed-loop position control of the steering angle is detailed. It should be noted that the velocity estimation of the vehicle is achieved using the VESC and its ROS driver developed by F1TENTH [5].

A. Kinematic Model

A well-known kinematic model used to represent the motion of car-like vehicles operating at slow speeds is the bicycle model [29]. In this model, the front two wheels of the vehicle are represented by one central wheel. The steering angle of the vehicle, δ, is the angle made between this central wheel and the longitudinal axis of the vehicle. Likewise, the rear two wheels are also represented by one wheel centered about them; however, is fixed to the body. Since it is assumed that the robot is restricted to movement on the ground plane, the vehicle has 2 Degrees of
Freedom (DoF). As a result, the state of the robot can be described by \([x, y, \psi]\) where \((x, y)\) represents the position of the vehicle’s center of gravity (CoG) in the world frame and \(\psi\) as its orientation [30]. It is important to note that the CoG in this case is considered the center of the robot, therefore the wheelbase distance, \(L\), is even on both sides of the CoG.

The velocity of the vehicle, \(v\), is the tangential velocity found at the CoG. The angle between \(v\) and the longitudinal axis of the robot is \(\beta\), also known as the slip angle and is defined as

\[
\beta = \tan^{-1}\left(\frac{\gamma}{L}\right) \tag{1}
\]

The continuous-time differential equations representing the motion of the vehicle are the following, where the inputs to the model are \(u = (V, \delta)\). A full derivation of this model can be found in [29], [30].

\[
\dot{x} = v(t) \cos(\psi(t) + \beta(t)) \tag{2}
\]

\[
\dot{y} = v(t) \sin(\psi(t) + \beta(t)) \tag{3}
\]

\[
\dot{\psi} = \frac{v(t)}{L} \sin(\beta(t)) \tag{4}
\]

B. Steering Position Control

The developed steering control algorithm is required to drive the front wheels to a desired angle whilst providing feedback of its current angle. This closed-loop response is required for determining the error of the position control algorithm in addition to estimating the odometry of the SPR. In addition, the ability to receive and drive to a commanded angle is a prerequisite for a variety of autonomous navigation systems. As mentioned in section III-B and IV-B, and displayed in Fig. 2, the steering assembly is composed of a DC motor, absolute position encoder, linkage and wheels. The first step taken in creating the control algorithm is polling for the encoder’s data at the extreme turning angles of ±35 degrees. With this data, a linear relationship between the polled encoder position and approximate steering angle of the robot is found. Therefore, the position error of the system can be computed by subtracting the computed current steering angle to any desired angle. A PID control law is then implemented and is used to drive the error between the current steering angle and the desired angle towards zero by outputting PWM values to the steering actuator. An overview of the steering control algorithm is presented in Fig 4.

V. ROBOT EXPERIMENTATION AND RESULTS

In this section, the SPR is tested for its ability to maneuver in an environment whilst accurately localizing itself using Dead Reckoning and simultaneous localization and mapping (SLAM). These algorithms are chosen as they validate the system’s ability to fully utilize the developed hardware and low-level controllers towards completing high level tasks. The first localization method used is Dead Reckoning, which is the use of the robots odometry information towards estimating the pose of the vehicle [31]. This uses the discretized formulation of the kinematic equations (2), (3), (4) with feedback of the actual velocity and steering of the vehicle to determine its state. The SLAM method used is a popular implementation known as Gmapping [32]. A full explanation of this methodology is available in [33]. In short, the algorithm is provided raw laser range data from the onboard sensor in addition to odometry information towards constructing a map of the environment whilst computing the robots position. The experiment takes place indoors in a rectangular area measuring 5m by 7m populated with items such as chairs, desks and a couch. The robot is tele-operated over a U-turn trajectory in this space, taking approximately 45 seconds from beginning to end. Besides the localization algorithms, an indoor GPS system with an accuracy of ±2cm is implemented to represent the ground truth over the trajectory.

Figure 5 shows the SPR’s x and y position over the trajectory using Dead Reckoning, SLAM (gmapping) and the indoor GPS. It can be seen that the robot has an error in its Dead Reckoning estimation, which can be attributed to wheel slippage. The pose maintenance performed by the SLAM algorithm, showed an improved estimate of the robot’s position, closely resembling the ground truth from the indoor GPS. As a result of this experiment, it is evident the SPR is capable of successfully moving in an environment whilst determining its position in the world. This is a significant building block toward achieving full autonomy with the system.

![Figure 5. SPR’s position estimation over U-Turn trajectory](image)

VI. CONCLUSION

In this paper, we present the design and construction of a unique 1/3rd scale car-like mobile robot for research and development in a security patrol setting. The robot’s simplistic,
modular and reproducible electrical hardware architecture is detailed. The benefit of this generalized electrical hardware architecture is its ability to be adopted into a variety of mobile robotic designs regardless of the use-case. Next, controllers that interface with the electronics is shown to accomplish low-level operation of the vehicle. The SPR’s developed system is validated through tele-operated navigation of a U-turn trajectory. This experiment has shown the robots ability to successfully maneuver in an indoor environment. The results also show the vehicles ability to localize within its operating environment using Dead Reckoning and SLAM while being compared to the ground truth supplied by an indoor GPS system. Future work with this robot includes the implementation of motion planning and obstacle avoidance towards achieving autonomy in a variety of settings.

ACKNOWLEDGMENT

This research was funded by the NSERC Discovery Program, grant number RGPIN-2017-05762.

REFERENCES


