Directory Navigation with Robotic Assistance

Tristan Brodeur[1], Alex Cater[2], Jean Chagas Vaz[3], and Paul Oh[4]

***Abstract*—Malls are full of retail store and restaurant chains, allowing customers a vast array of destinations to explore. Often, these sites contain a directory of all stores located within the shopping center, allowing customers a top-down view of their current position, and their desired store location. In this paper, we present a customer centric approach to directory navigation using a service robot, where the robot is able to display the directory, and guide customers to needed destinations.**

***Index Terms*—Service Robot, Path Planning, Navigation, FURo-S, Customer Interaction.**

I. INTRODUCTION

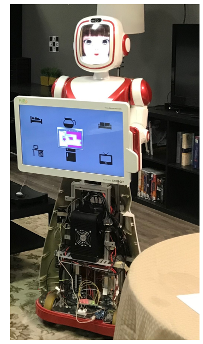
Directory maps placed in shopping malls and popular tourist destinations provide individuals a general overview of their surroundings, as well as a quick and efficient way to find their needed destination. Directory maps can be static posters placed along the entrances of malls, displaying an overview of the mall along with your current position, or a dynamic touchscreen that plots the path to your destination. However, once an individual has moved away from the map and begun their journey to their desired destination, recollection of the route to their destination quickly begins to fade.

In this paper, we address the problem of directory navigation with the use of a FURo-S service robot, as shown in Figure 1. Additionally, we exhibit a convenient user interface, which the customer uses to communicate their desired destination with the robot, wherein it then navigates by using an enhanced A* algorithm [1].

II. RELATED WORK

Several robotic platforms have been released to the public allowing companies cooperation and augmentation within their storefronts. Robots with attached tablets are particularly in high demand, as companies can easily display vital information to the customers. An example would be REEM [2], a humanoid service robot designed specifically for densely packed domestic environments. REEM is equipped with an array of proximity sensors allowing it to autonomously navigate its environment safely, while also sporting an LCD touch screen allowing customer interactivity. The REEM navigation platform is similar to that of the FURo-S, using a differential drive system, but with two caster wheels instead of one, allowing greater stability. The FURo-S was chosen due to its larger screen size, which will be the primal interactivity between the robot and the observer.

Vittorio Perera *et al.*[3] used the Pepper service robot to autonomously navigate within a hotel building. Using the ROS framework, Perera *et al.* were able to efficiently implement SLAM on the robot. Utilizing the included natural language capabilities of the Pepper robot, they were able to ask customers approaching the front desk questions and take requests, only resorting to the tablet when a response cannot be determined. However, in habitually noisy environments, such as that of a shopping mall, interactive tools such as speech would be of little use. Therefore, our go-to approach for communication will be on-screen information.



(a) FURo-S

Fig. 1: FURo-S Service Robot

III. PLATFORM

The robotic platform we used in our application is the FURo-S smart service robot provided by Future Robotics, shown in Figure 1. The FURo-S is meant to provide smart customer service in densely populated tourist areas. The FURo-S comes with an 360 degree array of ultrasonic sensors, a motion-sensing Microsoft Kinect, and an integrated 32-inch ultrasonic touch screen.

The FURo-S is a differential drive robot with two driven wheels in the front and two caster wheels located in the back, with non-holonomic constraints. FURo-S is able to navigate via dead-reckoning through two motor encoders placed on the two front wheels, or via optical navigation by way of infrared depth sensing through the Microsoft Kinect.

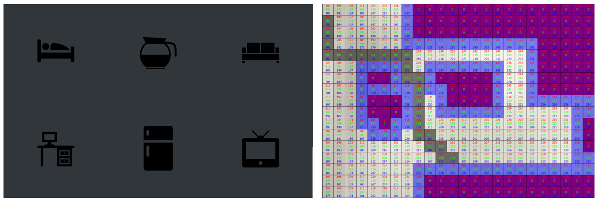
Our application uses dead-reckoning as our primary navigation technique due to its easy computation. Given the fact that the resolution of the included encoders reaches 10,00 pulses per revolution (PPR), position error accumulation over

[1]Tristan W. Brodeur is with the Department of Computer Science, University of Nevada Las Vegas (UNLV), Las Vegas, Nevada 89119, USA brodeurtristan@gmail.com

[2]Alex Cater is with Department of Computer Engineering, University of Nevada Las Vegas (UNLV), Las Vegas, Nevada 89119, USA cater@unlv.nevada.edu

[3]Jean C. Vaz is with the Department of Mechanical Engineering, University of Nevada Las Vegas (UNLV), Las Vegas, Nevada 89119, USA chagasva@unlv.nevada.edu

[3]Paul Oh is with Faculty of Mechanical Engineering, University of Nevada Las Vegas (UNLV), Las Vegas, Nevada 89119, USA paul.oh@unlv.edu



(a) Directory Listing (b) Map Generated

Fig. 2: Directory Listing and Resultant Map Generation

time is due to gear and wheel slippage, rather than encoder resolution. In our experimentation, we observed an average drift in distance traveled of $0.1m$ for every $50m$ travelled on a concrete flat surface. The impact of this positioning error is insignificant in our case, so no error correcting is applied.

The control hardware is comprised of two parts, the motor controller and the on-board PC. The motor controller controls the motor speed and the on-board PC computes obstacle detection and path planning. An Arduino Mega2560 is used to extract encoder values, while also receiving orientation values via an external IMU. These sensor values are then sent to the serial port on the on-board PC, allowing us to implement our dead-reckoning navigation.

IV. Implementation

Implementation of the directory navigation robot is separated into three steps. First, we detect the presence of an individual using the proximity sensors on the FURo-S. Following the presence sensing, a list of icons are displayed on the screen, allowing the user to select the desired location. After a selection is made, the FURo-S determines the shortest path to the destination using an augmented A* algorithm, and directs the customers to their requested destination.

A. Human Interaction with FURo-S

Optimally, the initial state of the robot's display is a splash screen with a greeting being displayed to the user. As a user approaches, the robot will display the directory of stores located within the shopping mall. Due to large amounts of foot traffic typically associated with densely populated pedestrian areas, the ultrasonic sensors will not trigger a wake up call unless a certain amount of time τ_s has passed from the initial sensor trigger. After the screen displays the directory, the user will tap on the icon, and a confirmation message will appear confirming the selected location. After the selection is confirmed, a map of the area will appear showing the path to the goal, and the robot will begin its navigation. A sample of the directory listing and the resultant map are shown in Figure 2.

B. Navigation Implementation

1) *Path Planning:* Each listing on the user interface has associated with it a tuple containing the x and y coordinates within the grid map. We use a modified A* algorithm [1] that adds a parameter p(n), namely the number of turnings the robot makes during its traversal. Whereas a typical A* algorithm may find the shortest path, penalizing turns reduces travel time along the generated path trajectory, while also finding the next optimal path.
2) *Map Generation:* In our implementation, we use a drawn map of the environment, whereby obstacles are represented by black marks. Using OpenCV, we measure the intensity value of each pixel, and add that pixel location to the map as obstacles. This image is then passed to our map generation method, where an appropriate configuration space is generated. A number of pixels α are set as boundary limits around each obstacle, which account for the diameter of the FURo-S wheel base. As shown in Fig. 2b, the free space is represented as white pixels and the generated path is represented as gray pixels. In this case, the starting position is the upper left corner, and after the user selects the TV icon, both the map and path are generated, and the FURo-S begins its navigation towards the target.

V. Conclusion

A directory navigating robot is proposed in this paper. Through communication via a friendly user interface, a user is able to have a robot direct them to their needed destination. An optimal path trajectory is created via an augmented A* algorithm that cuts down on travel time, and obstacle avoidance is capable via on-board sensors provided by the FURo-S. The experimentation was proven to be successful, as the resultant path demonstrated fewer turns on average compared to the original A* algorithm, allowing FURo-S to navigate to its destination quicker. The user interface was proven to be easy to use, and can be quickly changed to fit the needs of any environment. For future work, we intend to use map generation via added LIDAR sensors, and localization within the provided map will allow optimal path traversal.

References

[1] P. Sudhakara and V. Priyanka, *TTrajectory Planning of a Mobile Robot using Enhanced A-Star AlgorithmIndian Journal of Science and Technology,* November, 2016.

[2] F. Ferro and L. Marchionni, *"REEM: A Humanoid Service Robot" In ROBOT2013: First Iberian Robotics Conference,* pp. 521-525, 2014.

[3] V. Perera, T. Pereira, J. Connell, and M. Veloso, *"Setting Up Pepper For Autonomous Navigation And Personalized Interaction With Users"*, 2017.

[4] R. Arumugam, V.R. Enti, L. Bingbing, W. Xiaojun, K. Baskaran, F.F. Kong, A.S, Kumar, K.D Meng, and G.W. Kit, *"DAvinCi: A cloud computing framework for service robots"In Robotics and Automation (ICRA),* pp. 3084-3089, Alaska, USA, May, 2010.