Towards a Low-Cost Autonomous Wheelchair Navigation System Using COTS Components

Charlie Guan[1], Zeyang Wang[1], Vivek Burhanpurkar[2], and Jonathan Kelly[1]

I. OVERVIEW

Electric wheelchairs are often prescribed to individuals with mobility challenges. For a subset of users who have upper-body fine motor control impairments due to, for example, spinal cord injury, it is impossible to operate an electric wheelchair using the standard joystick interface. Such individuals must instead rely on other types of assistive control devices (e.g., sip-and-puff switches), which are typically extremely difficult to use. This results in degraded mobility and a substantially deteriorated quality of life.

A robotic navigation system for electric wheelchairs, which would allow the chairs to self-navigate in home and workplace environments, would dramatically improve users' mobility. However, at present, no widely available navigation system for wheelchairs exists, although the problem has been explored since the early 1980s [1]. Part of the reason is cost—much of the research to date has focused on the use of specialized sensing hardware. The prohibitive expense of such hardware makes the near-term, commercial deployment of a viable system unlikely.

Given significant recent advances in (inexpensive) navigation sensor technology and the continued maturation of open source robotics software, our research group recently asked the question: is it possible to build a reliable and low-cost autonomous or semi-autonomous wheelchair navigation platform using commercial-off-the-shelf (COTS) hardware and open source software only? In this extended abstract, we report on our initial progress towards answering this question by developing a prototype wheelchair navigation system with our industrial partners, Cyberworks Robotics, Inc., and Simcoe Habilitation Services, Inc.

II. SYSTEM DESCRIPTION AND CAPABILITIES

Our prototype navigation system (shown in Figure 1) is based on a standard commercial electric wheelchair, to which we have retrofitted a Kinect 2 sensor and related computing equipment. While previous research has focused on varying aspects of autonomy, including doorway traversal, wall following, and obstacle avoidance [2], modern simultaneous localization and mapping (SLAM) software enables the unification of these functions within a common navigation framework. We currently use the libfreenect2 open source library to acquire data from the Kinect 2. The second-generation Kinect has a 512×424 pixel time-of-flight depth sensor and a wide field of view HD video camera. We also use wheel odometry to aid in localization and mapping. All processing is carried out on a commodity laptop powered by an Intel i7 processor.

At present, we have implemented three main software capabilities: large-scale mapping, autonomous map-based navigation, and dynamic obstacle avoidance. We currently use the open source RTAB-Map as our SLAM package (running under ROS, the Robot Operating System) to build and maintain large maps in semi-dynamic environments [3]. An initial map can be built by an operator in real-time, by manually guiding the wheelchair to visit all locations where the platform will be expected to drive. During the mapping process, the SLAM software relies on odometry information to assemble successive point clouds captured by the depth sensor into a 3D map (see Figure 2), and also renders this map into a 2D floor plan. This floor plan must then be validated by the operator and corrected, if necessary, using an interface tool currently in development. RTAB-Map also continually captures RGB images from the Kinect and extracts visual features ('words') that are stored for future lookup to aid in localization and loop closure.

Fig. 1. A commercial electric power wheelchair with the Kinect 2 sensor mounted above the backrest, ensuring a wide field of view.

---

[1] Charlie Guan, Zeyang Wang, and Jonathan Kelly are with the STARS Laboratory, Institute for Aerospace Studies, University of Toronto.

[2] Vivek Burhanpurkar is with Cyberworks Robotics, Inc., and Simcoe Habilitation Services, Inc.

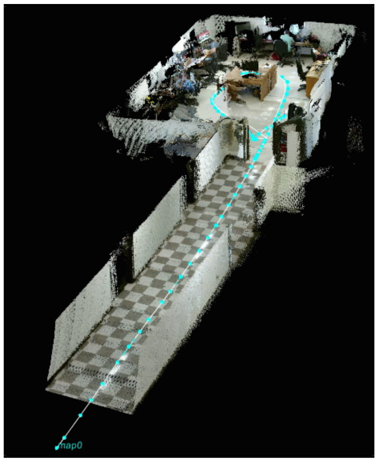

Fig. 2. Three-dimensional map of an office environment generated by RTAB-Map and the ROS Navigation Stack.

For autonomous navigation and obstacle avoidance, we employ the standard ROS navigation stack. The stack ships with a capable global path planner, which uses the 2D floor plan produced by RTAB-Map to compute an obstacle-free path from its current location to a selected goal. The system localizes itself primarily using dead-reckoning odometry. Concurrently, RTAB-Map processes RGB data to correct accumulated dead-reckoning errors by calculating updated pose estimates using recognized visual features.

The global path is then passed to a small-scale planner which builds a real-time map of the immediate vicinity of the wheelchair using live depth data, and adjusts the path to avoid any detected obstacles. A custom filter removes spurious measurements from the raw depth image and then exports a 2D cost map (in which obstacles accrue higher cost). The planner selects the path with the least cost through the navigation space. At present, the control loop operates reliably at 10 Hz, with depth information updated at 3 Hz.

The system is capable of reliably negotiating doorways and other narrow passages while following a smooth and predictable path to its destination. We have found that the system also operates well in complex environments with diverse geometries and scales.

III. RESEARCH CHALLENGES

While we have developed an initial prototype that performs reasonably well in many situations, the general problem of robust autonomous navigation is far from solved (of course). We are now investigating a variety of corner cases and failure modes, which we discuss briefly below.

As with any sensor, the Kinect 2 has some critical limitations. In particular, the unit can have difficulty registering accurate depth information in certain environments. Highly reflective surfaces may return false depth data and light-absorbent materials may produce a very low return signal. Transparent and translucent materials also produce erratic results. These issues could be mitigated by augmenting the infrared depth sensor with other sensors types, although cost would increase.

Due to RTAB-Map's reliance on visual features, localization is difficult in feature-sparse or highly repetitive environments. Importantly, highly repetitive environments may cause aliasing, that is, the false recognition of new environments as previously visited locations. Erroneous loop closure under this circumstance can result in significant mapping errors. A possible solution may be to combine the RGB and depth data to extract more distinct feature signatures.

As implemented, the ROS navigation stack obstacle detection algorithms do not account for floor gaps or other hazardous ground geometry. Also, the system cannot safely reverse because we have no rear-facing sensor.

The frequency and latency of the sensing and control loops necessitate limiting the wheelchair velocity to a modest walking pace, to ensure sufficient time to respond to dynamic obstacles. An upgrade to our on-board computer would partially solve this issue, although we hope to keep power consumption below 150 W (approximately 10% of the capacity of the existing wheelchair power subsystem). A further challenge is that the computer must stably operate at high loads for extended periods of time, without reaching the thermal limits of any of its components, even in high ambient temperatures.

Perhaps the most critical research challenge, that we have yet to address, is to determine how a user will interact with and command the system. There are a myriad of human-robot and human-computer interaction issues to explore. Thus far, we have focussed primarily on navigation performance only.

IV. CONCLUSIONS AND ONGOING WORK

Over the 6-month duration of the project, we have been encouraged by the progress made towards realizing our goal. The ROS software components that drive the system have largely been used 'out of the box', without the need to write significant amounts of custom code. Our opinion is that the development of a viable, cost-effective COTS-based wheelchair navigation system may soon be within reach.

We hope to address the issues mentioned in Section III in the future, and to further improve the robustness and capabilities of the system. Full navigational autonomy has the potential to improve the safety of users and those around them, while greatly reducing operator fatigue.

We are also planning to begin testing our development platforms in busy home, office, and retail environments, in order to assess and validate its real-world performance. This testing will be carried out under the guidance of occupational therapist, ensuring that we meet the needs of the target community.

REFERENCES

[1] R. C. Simpson, "Smart wheelchairs: A literature review.," *Journal of rehabilitation research and development*, vol. 42, no. 4, pp. 423–436, 2005.

[2] R. C. Simpson, E. F. LoPresti, and R. a. Cooper, "How many people would benefit from a smart wheelchair?," *Journal of rehabilitation research and development*, vol. 45, no. 1, pp. 53–71, 2008.

[3] Mathieu Labbe and Francois Michaud, "Online Global Loop Closure Detection for Large-Scale Multi-Session Graph-Based SLAM,"