Practical Implications for the Design of Mobile Assistive Robots for Quadriplegics Using a Service Dog Model

Kan Chen, Songkan Tang, Hairong Jiang, Student Member, IEEE, Juan P. Wachs, Member, IEEE, and
*Bradley S. Duerstock, Member, IEEE

Abstract— An increasingly popular area of study in Assistive Robots (ARs) is the capability to transport objects for persons with mobility impairments. Some of the biggest challenges in integration of mobile ARs related to the design of a usable and effective control interfaces. A design model, inspired by the essential functions offered by service dogs assisting quadriplegic owners, is proposed to establish the key design criteria and features for enhanced implementation of effective and inexpensive mobile ARs for these users.

In this paper, emphasis is placed on evaluating multiple control modalities including keyboard, voice and gesture recognition as well as vision-based autonomous control to efficiently navigate a Turtlebot mobile robot through obstacle courses. Though voice and gesture recognition provide natural and intuitive human-robot interaction, they were slower and more error prone than keyboard operation. Pilot experiments show gesture input often required the same number of commands as keyboard control but was slower. Autonomous navigation completed courses quickly even when unanticipated obstacles were introduced and more errors were generated.

Keywords—Assistive Robots; multimodal human-robot interface; quadriplegics; bio-inspired design

I. INTRODUCTION

Assistive robots (ARs) have the potential to enable persons with disabilities (PWDs) to more independently perform certain tasks that are routinely accomplished by persons without disabilities, such as feeding, dressing, and object retrieving. ARs comprise of many different types of robots depending upon the type of task to be performed. Robotic manipulators are used for grasping, holding, and manipulating daily living objects. Mobile robots, instead, are required for monitoring the environment by self-navigation, delivering and retrieving objects and user tracking.

In this pilot study, the design criteria necessary for a multi-purpose mobile robot that is useful for transporting objects as well as remotely communicating with others for quadriplegic users is studied. The most important tasks that a mobile robot can perform were weighed in comparison to the practical considerations of placing a mobile robot in the regular routine and home environment of PWDs. The relative importance of these tasks; however, varies according to disability, individual needs, and other practical and social considerations, such as cost, size of living space, and culture [1]. In our particular study, we adopted a bio-inspired approach in the design of human-robot interface (HRI) with control modalities and special features for assisting those with mobility impairments. We had focused on quadriplegics due to spinal cord injury (SCI). The goal of this work is to determine a practical, yet inexpensive, mobile AR that would have commercial benefits for this target population in the performance of a simple tasks. In order to determine ideal usability and functional requirements for a mobile AR, we based the most critical design criteria on the historical assistance guidelines provided by service dogs for quadriplegic owners.

A. Bio-Inspired Design for Mobile AR

The concept of mobile robots inspired on service dogs has been adopted before. Nguyen and Kemp [2] focused on the development of a robot with multiple functions, similar to those provided by service dogs and consequently modified a home environment to the completion of those functions. Determining a set of functions that service dogs can accomplish is helpful but not enough for a practical implementation of a mobile AR. However, Nguyen and Kemp [2] did not apply this bio-inspired model to the flexibility of HRI that exists between service dog and owner with a disability. Likewise, the mechanics of their robot does not follow the physiologic design of a service dog by including a height-adjustable, human-like manipulating arm.

Whereas, the mobile robot platform that we tested is more simplified and focuses on these basic design requirements:

- independent AR operation by users with disabilities through an accessible, multimodal HRI from a wheelchair, bed, or other environments,
- intelligent control of the AR to function semi-autonomously to decrease user operation and fatigue, and
- the AR must be adaptable to variable environmental conditions including unanticipated obstacles and noise.

B. Proposed Functions of Mobile AR

A selection of functions based on this basic design approach for the service dog-inspired model does not need to be exhaustive. Providing only a small number of functions may make this mobile AR practical for the user with a disability as long as it is cost effective and as reliable as a service dog. Here are five categories of functions that service dogs provide, which we require of an effective mobile AR for quadriplegic users.
• It can travel to learned positions autonomously when commanded. It can self-navigate around simple obstacles.
• Find household members that it recognizes and expects at particular locations (telepresence).
• Transport or retrieve, carry, drop off trained objects that the operator wants.
• Operate buttons, switches, pull handles, and push doors that are easily accessible.
• Companionship and a level of social interaction

Though many of these functions are currently available in different prototype mobile robots, many lack key features which make it either inaccessible or less attractive to users with disabilities. We explore the development of a low-cost mobile AR based on these bio-inspired design criteria and compare its usability for PWDs and task performance.

II. RELATED WORK

A. Accessible Human-Computer Interaction

A user-operated computer is often needed to interact with the mobile robot. For the quadriplegic AR operator there are several HCI interfaces that can be used. Sip and puff [3], speech recognition (SR) [4], eye tracking [5], tongue control [6], keyboard, and gesture recognition are all commonly used to control a computer interface [7]. Though there are benefits and drawbacks to each input modality, ideally uncumbered or “hands-free” HCI provides a natural and intuitive interface, such as with voice or gesture recognition [8]–[10]. These modalities do not require direct contact with a controller, such as with joysticks, keyboards, or sip and puff. Using the bio-inspired model both verbal [2] and gesture commands are readily understood by service dogs.

B. Mobile Robot Platform

The transportation of objects is a major task based on this bio-inspired model. However, mobile ARs have also included wheelchair co-robots, which are meant to transport PWDs with some degree of autonomy [11]. The same technical challenges and technologies are often shared by wheelchair co-robots as well as mobile ARs that can carry small objects. Autonomous navigation is important in order to relieve the operator from having to remotely control the mobile AR. To accomplish self-navigation sensors are needed. The wheelchair co-robot literature discusses many different sensor array systems, including touch, infrared (IR), and ultrasound. These sensors are unreliable in detecting drop-offs, such as the top of stairs or curbs [11]. Kinect® sensors which combine color video with IR depth sensors are commonly used for 3-D vision-based sensing in robotics. This technology is closest to being bio-inspired as it allows for autonomous navigation with object avoidance as well as 3-D object recognition and facial detection [12]. In the home environment sensing and wheeled mobility is less challenging than outside locations, which is not addressed.

III. METHODS

A Turtlebot™ mobile robot platform was used for testing as it provides the most basic physical attributes of the service dog. In addition to being mobile, it has auditory and 3-D visual sensing ability through the Kinect camera. These functions allow it to be programmed to navigate autonomously and avoid unanticipated obstacles. The Turtlebot system consists of three parts: the human computer interface, automatic navigation, and the mobile robot control. The details of each part are described below.

A. Human-Computer Interface

Three control modalities were applied: keyboard, speech recognition based interface, and gesture recognition based interface. Five commands are applied to navigate the mobile robot: forward, backward, left turn, right turn, and stop.

For the keyboard control, specific keys are assigned corresponding to each command. When a directional key is pressed repetitively, the mobile robot speeds up in that direction. The speech recognition based interface was implemented using an open source software CMU Sphinx. The software has an inner trained language model that was specified to a five-word-dictionary (corresponds to the five navigational commands). The gesture recognition based interface uses a Kinect camera to recognize the users’ hands and face. Three steps were applied: foreground segmentation, hand detection and tracking, and trajectory recognition. A detailed description can be referred to [7]. A five-gesture lexicon was adopted: the vertical hand movements representing four directions and a clockwise hand trajectory indicating the stop command (Fig. 1 A).

B. Autonomous Navigation

Simultaneous Localization and Mapping (SLAM) technology is applied for autonomous navigation. A map of the environment is built using the depth information from a Kinect camera when the mobile robot is actively exploring the environment. With this map, the mobile robot can avoid obstacles and navigate itself to a desired position using the global and local path planning algorithm [13] (Fig. 1 B).

As only 2D map is used, the robot’s pose (Eq. 1) is described by two-dimensional planar coordinates, and the angular orientation.

$$\begin{align*}
(x, y, \theta)
\end{align*}$$

(1)

For robot movement control, the Odometry Moving Model [14] was used, each control input $u_t$ that move the robot from pose $\mathbf{L}$ to pose $\Omega$ can be described as Eq. 2.

$$\begin{align*}
\mathbf{u}_t = \left( \begin{array}{c}
\mathbf{L} \\
\Omega
\end{array} \right)
\end{align*}$$

(2)

The control input $u_t$ can be accomplished by 3 steps. The first step is a rotation, followed by a straight line motion and finally another rotation.

C. Assistive Robotic Control

The Turtlebot mobile robot is programmed using the Robotic Operation System (ROS). The program is executed on the robot’s workstation (client) and waits for commands from the server sent through socket communications. Robot odometry was calculated and saved for positional feedback to notify the robot’s current location.
By integrating both local visual information and navigational information from a map, an inner developed algorithm is applied to decide whether the robot has met its proper destination (Fig. 1 C).

IV. EXPERIMENTAL RESULTS

There were two objectives of the preliminary experiments: 1) Compare different HRI control modalities: keyboard, voice, gesture and auto-navigation. 2) Investigate the usability of this AR system in accomplishing essential service dog functions. Four scenarios were adopted to validate the efficiency of the presented AR system. Experiments were conducted in a simulated home environment where the known obstacles were a desk, chairs, and file cabinet (Fig. 2).

Each control modality was tested in four scenarios (with four subjects and each subject performed three trials). Four subjects (2 males and 2 females, aging from 23 to 30) were involved in the experiments, during which the subjects were required to control the Turtlebot from a set starting point to a specific location using the keyboard, voice, or gesture commands. For each scenario, an ANOVA test was conducted to determine statistical significance in performance among the four different control modalities; keyboard, voice, gesture, and auto-navigation.

A. Average Task Completion Time

Average task completion time can reflect the efficiency of the operation. According to ANOVA test results, there was a significant difference among different control modalities ($P=0.015, 4.62E-5, 1.02E-5, 4.97E-8$). Keyboard operation and auto-navigation were similarly the quickest control modalities for Route 1 without obstacles (Fig. 3). For Route 1 with obstacles, auto-navigation almost doubled the completion time (Fig. 3).

Voice control took the most time in three of the four scenarios primarily due to the delay in reaction time to voice recognition. In scenario 4 with the longer route and obstacles, gesture recognition became the most time-consuming control modality (Fig. 3).

B. Error Rate

During a certain trial, an error would be counted if the Turtlebot hit obstacles or got stuck. Average numbers of errors per trial was regarded as error rate. According to ANOVA results, only in scenario 1 and 2 (the same route with or without obstacles) were significantly different ($P=0.045$ and $P=0.002$, respectively) (Fig. 3). There were no errors using the keyboard. As expected, more errors were evident when obstacles were introduced. This was particularly evident during auto-navigation. However, statistical significance among inputs was only apparent during Route 1 (scenarios 1 and 2), likely because recording errors is more sensitive than the much longer Route 2 (scenarios 3 and 4) (Fig. 3).

C. Number of Total Commands

Since no commands were issued during auto-navigation, only keyboard, voice, and gesture control were compared using ANOVA tests. Only Scenario 2 showed a significant difference for number of commands among input modalities ($P=0.006$). In this case, where the route is shortest and has...
obstacles, voice recognition required more commands than the other inputs (Fig. 4). In the other scenarios, all three input modalities required a similar number of commands.

![Graph showing number of total commands for scenarios 1 to 4.](image)

**Figure 4. Experiment results of total commands**

V. DISCUSSION

Although there are many factors involved in the implementation of new AT, including cost, size, usefulness, PWD usability, dependability, social acceptance, and payer source [1]. It is critical to identify those essential design features that would result in successful adoption of mobile ARs in the homes of quadriplegics due to SCI. Thus, a user-centered design strategy based on the needs of operators with disabilities is critical. Unfortunately, it is not always possible to provide widespread user testing when there is only a single or few prototypes available. In order to make general design recommendations for a low incidence population it is useful to adopt a known bio-inspired design model [15].

Service dogs have proven to be invaluable assistants for persons that are quadriplegic. Service dogs respond to verbal and gestural commands as well as behavioral and perhaps physiological cues [16]. Therefore, it is reasonable to assume a multimodal HRI would be more practical for users with quadriplegia to control an AR.

Route 1 was more sensitive to detecting task completion time, error rate, and number of commands than the longer Route 2. This is likely due to the more intricate maneuvering that is required for this obstacle course. Consistently, keyboard operation was quicker and resulted in a lower error rate than voice and gesture recognition. Scenario 2 that included obstacles was the most sensitive test course. Voice control required more commands because of its higher error rate and completion time compared to keyboard and gesture control. Voice control had a delay in sending commands, which likely made maneuvering the Turtlebot more difficult.

Gesture recognition required approximately the same number of commands as keyboard operation during Scenarios 2 and 4 when obstacles were present. However, task completion was longer with gesture recognition and there were more errors. Processing time likely was also a factor in the greater delay of gesture recognition. During keyboard operation many commands are executed by the user to control turning as well as speed. To increase speed keys are pressed consecutively, which resulted in faster completion times but more issued commands. It is unclear whether one input modality is more fatiguing for quadriplegic users to perform than another.

Among all the input modalities, longer completion time corresponded to higher error rate. The exception was auto-navigation which created a lot of errors but had a quick completion time due to a rapid reaction time without user input. Though completion time is important, quadriplegic users may still prefer using voice and gesture recognition due to greater usability, particularly in situations, like lying in bed where users may not be able to physically manipulate a keyboard or other computing input device.

More testing is needed by quadriplegic participants. Technical challenges include reducing processing time of voice and gesture control. In addition, better object recognition from visual sensors should enhance navigation.

REFERENCES


