


Article

# Robotic Nursing Assistant Applications and Human Subject Tests through Patient Sitter and Patient Walker Tasks <sup>†</sup>

Cody Lee Lundberg <sup>1</sup>, Hakki Erhan Sevil <sup>2,\*</sup> , Deborah Behan <sup>3</sup> and Dan O. Popa <sup>4</sup>

<sup>1</sup> Automation and Intelligent Systems Division, University of Texas at Arlington Research Institute (UTARI), Fort Worth, TX 76118, USA; codyl@uta.edu

<sup>2</sup> Intelligent Systems & Robotics, University of West Florida, Pensacola, FL 32514, USA

<sup>3</sup> Nursing and Health Innovation, University of Texas at Arlington, Arlington, TX 76019, USA; dgreen@uta.edu

<sup>4</sup> Electrical & Computer Engineering, University of Louisville, Louisville, KY 40292, USA; dopopa01@louisville.edu

\* Correspondence: hsevil@uwf.edu

<sup>†</sup> This paper is an extended version of our paper published in Dalal, A.V.; Ghadge, A.M.; Lundberg, C.L.; Shin, J.; Sevil, H.E.; Behan, D.; Popa, D.O. Implementation of Object Fetching Task and Human Subject Tests Using an Assistive Robot. In Proceedings of ASME 2018 Dynamic Systems and Control Conference (DSCC 2018), Atlanta, GA, USA, 30 September–3 October 2018, DSCC2018-9248; Fina, L.; Lundberg, C.L.; Sevil, H.E.; Behan, D.; Popa, D.O. Patient Walker Application and Human Subject Tests with an Assistive Robot. In Proceedings of Florida Conference on Recent Advances in Robotics (FCRAR 2020), Melbourne, FL, USA, 14–16 May 2020; pp. 75–78.

**Abstract:** This study presents the implementation of basic nursing tasks and human subject tests with a mobile robotic platform (PR2) for hospital patients. The primary goal of this study is to define the requirements for a robotic nursing assistant platform. The overall designed application scenario consists of a PR2 robotic platform, a human subject as the patient, and a tablet for patient–robot communication. The PR2 robot understands the patient’s request and performs the requested task by performing automated action steps. Two categories and three tasks are defined as: patient sitter tasks, include object fetching and temperature measurement, and patient walker tasks, including supporting the patient while they are using the walker. For this designed scenario and these tasks, human subject tests are performed with 27 volunteers in the Assistive Robotics Laboratory at the University of Texas at Arlington Research Institute (UTARI). Results and observations from human subject tests are provided. These activities are part of a larger effort to establish adaptive robotic nursing assistants (ARNA) for physical tasks in hospital environments.

**Keywords:** assistive robotics; human subject tests; human–robot interaction; robotics in healthcare



**Citation:** Lundberg, C.L.; Sevil, H.E.; Behan, D.; Popa, D.O. Robotic Nursing Assistant Applications and Human Subject Tests through Patient Sitter and Patient Walker Tasks. *Robotics* **2022**, *11*, 63. <https://doi.org/10.3390/robotics11030063>

Academic Editors: Angel P. Del Pobil and Ester Martinez-Martin

Received: 31 March 2022

Accepted: 12 May 2022

Published: 16 May 2022

**Publisher’s Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**Copyright:** © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

Patients with disabilities and with less mobility often require one-to-one assistance to manage their daily activities. Due to the increasing number of patients, nurses are not able to offer enough care and attention to patients [1]. By using robotic assistants for nursing tasks, we can free up some of the time of nurses so that they can prioritize their tasks with patients who have severe health conditions. In the literature, there are various robotic systems that have been developed to help patients with activities of their daily living without needing much help from others.

Devices such as wheelchairs are specifically designed for mobility, and they offer limited support in performing everyday tasks. Being confined to a wheelchair most of the time, people with disabilities often face difficulty in performing their everyday tasks. There are various studies in the literature in which manually controlled robotic manipulators are used to help disabled people with their everyday activities. Some of these studies are worth mentioning here. A joystick-controlled manipulator robot [2] is presented in the literature that is specifically designed to help people with eating. “Handy 1” [3], a rehabilitation robot,

is introduced to help severely disabled people with tasks such as eating, drinking, applying makeup, washing, and shaving. The “MANUS” robot [4,5] is developed to assist people with navigation in an unstructured environment by using head/neck movement tracking, voice recognition, and a wrist/arm- and finger-controlled joystick. A pressure-sensitive multi-finger skin care robot [6] applies ointment to the patient’s skin autonomously. “WAO-1” [7], a massaging robot, massages patients and helps them to relax for better skin and oral rehabilitation.

Another important aspect is that people with social deficits and cognition impairments require psychological help to facilitate themselves with better cognition, communication, and motivation skills. Several studies have shown that using social robotic systems can provide better results rather than using physical robots for the aforementioned psychological requirements. For instance, the social robot “Aiba” [8] helps patients to lose weight by building a social relationship with them through daily exercise routines. “Clara” [9] monitors and guides patients through spirometry exercises using pre-recorded voices. An animal robot, “Paro” [10], and a companion robot, “Parlo” [11], simulate care and affection to treat elderly people. To build even better relationships with users, toy animal robots have been introduced in the literature, such as “Pleo” [12], which can express clear emotions. A humanoid robot “Bandit II” [13,14] that has been developed to help to build cognition and social skills using interactive exercises demonstrates promising results for the treatment of autism in kids and dementia in elders.

In places such as elderly homes and hospitals, robots need to coordinate and assist multiple people at once. A smart-home-based concept is introduced in [5], with robotic arms mounted to the ceiling that can assist people with eating. It also has the capability of transferring people to pre-defined destinations, it has robotic hoists that transfer people to and from wheelchairs, and it has a smart bed that can position users on the bed and can be used to provide healthcare to people in groups [5]. We use similar ideology to design a robotic system that can help several people with better coordination at the same time.

The robotic platforms discussed above are designed to only provide assistance in specific applications for patients. In a real-world hospital environment, robots need to assist patients with different tasks. Hence, using such robots, which are not designed for a scattered and unstructured environment, to provide general healthcare in hospitals may become impractical. In our study, we explore a novel autonomous solution, Adaptive Nursing Assistant System (ARNA), that can provide general healthcare to hospitalized people. The tasks used in our study include fetching objects such as water bottles, medicines, and food, assisting patients with their everyday activities using a walker, and measuring their vitals.

Robotic platforms that fetch objects can be very useful in hospitals to assist with several tasks, such as delivering medicines, refreshments, and food to patients when needed. Using robots to fetch objects can dramatically improve the quality of life of patients who are confined to bed. Fetching objects autonomously is a challenging task since real-world objects dramatically vary in size and shape. Several studies have proposed alternate solutions to make the fetching task simpler. One of the approaches introduced in the literature is manual control by the user, such as mobile robot “SUPER-PLUS” [15] and helper mobile robot “Robonaut” [16,17]. Another example is the service robot “MARY” [18], which navigates to a destination by taking voice commands such as ‘move to left’, ‘move to right’, and ‘go forward’ from a user to guide the robot to fetch objects for the user. Although these techniques make the object fetching task simpler, issuing commands in every step can be very tedious to the user.

The object manipulator mobile robot El-E [19] alleviates this problem by using an autonomous algorithm instead of voice commands from the user. The robot fetches household objects, such as a water bottle, placed on a surface by using a laser-guided path pointed by the user without using any additional commands from the user. Finding a good gripping point to hold objects during fetching is one of the most challenging problems in autonomous object fetching. The study in [20] uses tactile feedback from the sensors in

grippers to find a proper gripping point to hold objects such as plastic bottles and cups. Another study conducted in [21] uses markings on objects to fetch household items such as mugs, books, pencils, and toothbrushes. A similar technique is used in our current project to fetch objects. The objects used in our project are marked with unique AR tags. The “Personal Robot 2” (PR2) uses the AR tags to identify and fetch the objects in this study.

Fetching large objects can also be challenging and would require additional help to fetch objects since the objects cannot be held in a single gripper. Such a problem can be overcome by using additional robots to fetch objects instead of just using one. For instance, Pettinaro et al. introduced a study in which several tiny “S-Bots” [22] are used to fetch large objects that cannot be held by a single gripper. Although it is a good solution, using several robots is not a feasible solution in a hospital environment. In our project, we use two grippers to hold objects if they cannot be held by using one, such as fetching a patient walker, as discussed in the following sections.

Patient walkers are widely used in hospitals to support walking of patients. They increase mobility and allow patients to move freely, yet in some cases the patient requires additional assistance from nurses or caregivers when using a walker. Thus, our motivation is to develop algorithms for the robotic platforms to assist the patient with the walker equipment. “XR4000”, a walker robot [23] with an inbuilt walker, assists elderly people in walking to a destination autonomously using a pre-defined map template. The PAMM (Personal Aid for Mobility and Monitoring) robot discussed in [24] adds additional functionalities such as obstacle avoidance and navigation guidance to existing walkers. When using the walker, the robot can monitor the health condition of the users and informs caregivers if any emergency situation is detected.

Based on their surroundings, people tend to walk with different speeds. To assist the user in such situations, robotic platforms have to change their speed with respect to the user actions. An omnidirectional moving robot discussed in [25,26] uses information from various sensors, such as force, laser, and tilt sensors, to predict user’s actions. Based on the prediction, the robot adjusts its speed and facilitates users to move at a variable speed depending on the situation. The PAM-AID (Personal Adaptive Mobility) robot discussed in [27] detects surroundings and delivers the information to the users, helping blind people to navigate and interact with their surroundings. A similar technique is used in our project to assist patients with a walker. The PR2 platform used in this study supports users and prevents them from falling, similar to the robot discussed in [28].

Vital signs such as heartbeat, temperature, and blood pressure help doctors to understand the patient’s health condition and to decide the treatments that should be given to the patients. Hence, a reliable and error-free measurement recording is crucial, especially in situations such as measuring the patient’s heartbeat [29] during surgeries and measuring the activity [30] of the patient during rehabilitation. Among the above discussed vital signs, temperature measurement is a widely and commonly used method to monitor the patient’s health condition. Several temperature measurement techniques have been proposed in earlier studies to monitor the patient’s body temperature with precision. For instance, the mobile robot “Rollo” [30] uses an IR sensor (infrared sensor) to measure the temperature of the patient, and a robotic system introduced in [31] uses a temperature sensor to do so. However, temperature measurement alone is not enough to understand the patient’s health condition, especially when they are confined to bed. In order to understand the health condition of such patients better, we require measuring more vital signs in addition to temperature. A “SleepSmart” [32] multi-vitals monitoring bed measures the person’s blood pressure, oxygen levels, breathing inhale/exhale rate, heartbeat and the temperature of the patient to monitor their health condition. Another study monitors blood pressure, blood oxygen level, body temperature, pulse rate and galvanic skin response by using a modular health assistant robot called “Charles” [33]. The robot can also measure other vital signs such as blood glucose levels by interfacing with additional equipment, a blood glucose monitoring system, to understand the patient’s health condition better.

Although these devices can measure the vital signs accurately, they require additional sensors, restricting them to performing only certain task. On the other hand, the PR2 robot used in our project measures the temperature of the patient by using a contactless home IR thermometer without any additional sensors. We utilize computer vision techniques to read the temperature from the thermometer's screen, and that information can be sent to nurses or caregivers for further analysis.

Toward to our larger goal of developing ARNA platforms, our main focus in this paper is to study three specific applications: "object fetching" and "temperature measurement" as patient sitter tasks, and a "patient walker" task. The results obtained from this study will be part of development efforts for ARNA platforms. In this paper, we present developed algorithms, parameter analysis, and observations from human subject tests. We build these efforts upon our previous studies, and further details about previous research on ARNA can be found in [34–38]. The original contributions of this paper include (i) identifying basic nursing tasks and designing an application pipeline of those tasks in order to implement them with a robotic platform, (ii) proposing solutions to the integration of the physical environment/objects and robotic platform in a hospital-like setup, (iii) performing parameter analysis to emphasize different effects of variables on the designed nursing task applications, (iv) conducting human subject tests to demonstrate practical aspects of designed nursing implementations, and (v) a general feasibility assessment of developed algorithms for basic nursing tasks with providing human subject test results and feedback and comments from human subjects.

The remainder of the paper is organized as follows. The next section describes the developed algorithms in this study. The hardware and workspace used and parameter analysis are presented in Section 3 and Section 4, respectively. Section 5 provides information about human subject test design, scenario details, results, and observations from participants. In the final section, conclusions are presented.

## 2. Description of Algorithms

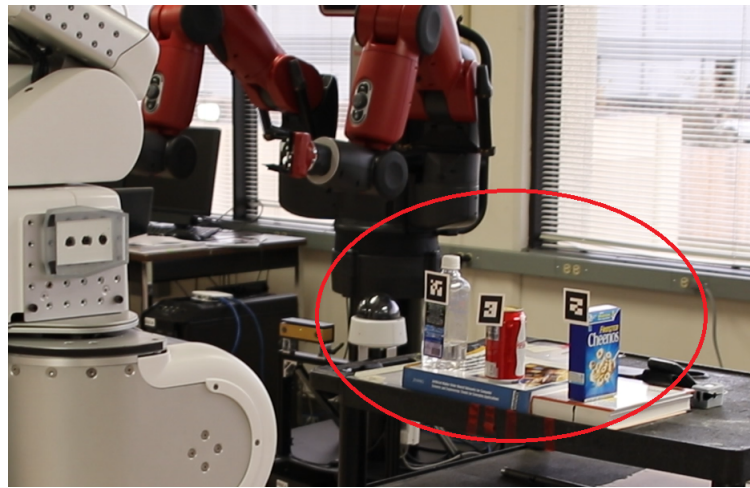
### 2.1. Navigation Algorithm

Navigation is one of the crucial tasks in this study. Since the hospital environment is unstructured and cluttered, the robotic platform operating in such an environment can face several challenges. It needs to know its environment accurately to avoid obstacles and reach the goal position precisely. The major objective of this task is to construct a safe and collision-free navigation for PR2 that can fulfil the above-stated challenges. We adopt 'ROS 2D navigation stack' [39] for this purpose. The modular software package constructs a 3D map of the surroundings and localizes PR2 on the map. It combines data from PR2's base LiDAR and torso LiDAR to construct a 3D occupancy grid, which is flattened to a 2D occupancy cost map of the surrounding obstacles. The cost map of the surroundings is fused with its odometry sensors by a probabilistic localization library, AMCL, to localize PR2 on the map [40]. The AMCL library implements an adaptive Monte Carlo localization algorithm to predict PR2's location and to track its position during the navigation. Using the 2D cost map, location, and position of PR2, we construct a navigation map of the environment. The map is updated with new obstacles in real-time and a new navigation plan is prepared using that information. In this study, several pre-defined waypoints are used for the patient's bed location, start position, and goal position. When a task is requested by the user in the experiments, PR2 uses its base LiDAR and torso LiDAR to estimate its location. Using initial and destination points, PR2 prepares a navigation plan, which is translated into velocity commands and sent to the base controller for navigation. For all our experiments discussed in this study, PR2 navigates to the patient's bed and waits for a request from the user at the beginning of the experiment. When requested, the PR2 robot navigates to the goal position, performs the task (for instance, fetch an item), and returns to the patient's location to hand over the object, or to complete some other task.



## 2.2. Object Position Detection Algorithm

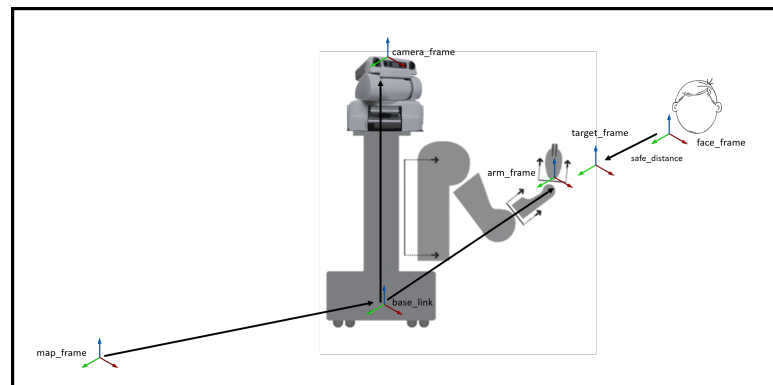
To detect the position of the objects in this study, we use an open-source ROS AR tag tracking library, “ar\_track\_alvar”, which detects AR tags in real-time [41]. The library detects the position and pose of the objects using AR tags. The reason for choosing this library is that it performs tag detection with high accuracy, even in poor lighting conditions. Additionally, it can detect multiple AR tags at the same time. AR tags with a fixed size and resolution are generated using this library. Objects used in experiments are labelled with the generated AR tags and are placed on a table, as shown in Figure 1, for the robot to pick up and fetch them. The idea of adding AR tags to objects is intended to increase detection performance for corresponding objects as these tags have unique patterns to help the developed algorithm with detection. PR2 uses its stereo camera to identify objects placed on the table. The library uses AR tags on objects to estimate information such as position, orientation, and distance from camera in order to plan the arm motion to fetch the objects.



**Figure 1.** Objects used for fetching task.

## 2.3. Human Face Detection Algorithm

The face detection technique in this study is used to find the forehead location on a patient’s face in a temperature measurement task. After reaching a goal position, PR2 uses its stereo camera to look for the “face” of the patient. The images are then processed by the “face\_detector” [42] ROS library to find faces and their orientations. The library implements the Haar–Cascades technique to detect faces in real-time. The Haar–Cascades [43] technique uses pre-compiled model templates that can recognize face features, such as eyes, nose, and mouth, in images. Other facial features such as distance between eyes, depth of the eye sockets, and size of the nose [44] are used to generate unique fingerprints of a face. The images taken are then compared with pre-compiled fingerprints to detect faces. Any false positives in images are removed by using depth data of objects from the stereo camera. In addition to removing false positives, the stereo camera’s depth data are also used to calculate the position ( $x, y, z$ ) and orientation (Roll, Pitch and Yaw) of the patient’s face with reference to the stereo camera’s frame. The ROS “tf” library [45] provides several functions to keep track of coordinate frames and to transform of the coordinate frames without tracking them manually. The calculated coordinate frame is tracked with respect to various other coordinate frames (base, arms, head) in a tree structure by the “tf” library. Figure 2 shows details of several coordinate frames associated with PR2. Some of these frames are generated in real time using various PR2 sensors, while the others are hard-coded. PR2 keeps track of the patient’s face with respect to camera frame and re-calculates face coordinates when the face moves. In our experiment, PR2 is able to track patients even when they are standing, sitting, or lying on bed. Even when the user is moving away from the robot, the technique can efficiently keep track of the patient’s face from a long distance.



**Figure 2.** Illustrative representation of coordinate frames in the system consisting of the robot and its environment.

#### 2.4. Motion Planning Algorithm for the Robot Arm

The patient's face is used as a virtual target frame to plan motion for the robotic arm. In our study for human subject tests, a safety offset called "safe distance" (Figure 2) is added to the virtual target frame to increase the patient's comfort and to prevent the robotic arm from getting too close to the patient. The offset parameter can be adjusted based on the user's comfort. The motion trajectory planning system uses a virtual target frame as the target frame. The coordinates of the target frame are checked to verify whether they lie in a currently defined workspace or not. After verifying the coordinates, the target frame is compared to check if any further movement should be performed to reach patient. If movement is needed, i.e., the PR2 arm cannot reach the target frame, the system calculates the necessary distance to move for the robot to reach desired target frame position. We use inverse kinematics to calculate the parameters for each joint (seven joints for PR2) of the PR2's arm. Since there can be several possible solutions, different constraints, such as trajectory time, effort required to perform the motion, and power consumption, are imposed on the possible solutions to select a feasible solution. After calculating the required parameters, the OMPL (Open Motion Planning Library) planner [46] from the "MoveIt" [47] ROS library is used to plan motion for the robotic arm. The library allows the user to configure virtual joints, collision matrix and some other motion parameters. The GUI also allows the user to tune optimization parameters such as search timeout by selecting the suitable kinematics solver. The various parameters such as target frame, joint parameters, and solver are used by the KDL (Kinematics and Dynamics Library) to calculate translation and rotation parameters that the robot should take to reach the desired goal, as shown in Figure 3. The values are then used by the arm controller to perform collision-free arm motion.

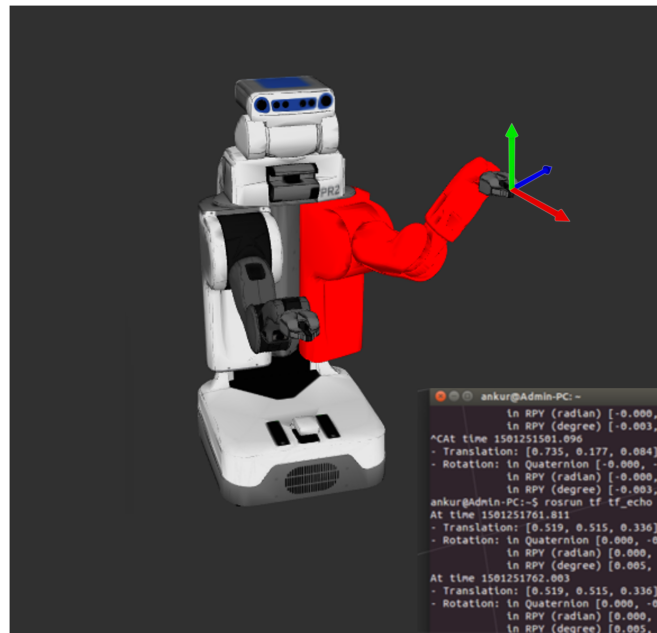


Figure 3. Motion planning for robotic arm using target frame.

### 2.5. Thermometer Digit Detection Algorithm using OCR

For the temperature-measurement task, a high-resolution camera is mounted on the PR2’s shoulder to record images of the thermometer’s screen. Using the robot’s odometry sensors, we estimate the orientation of the thermometer and use the perspective geometry to perform image tilt correction. The captured image is then cropped to show only the thermometer screen region and an additional buffer for better contour detection. An ROI (region of interest) is extracted from the captured image. A black hat morphological operation is performed on the image to separate dark (digits region) and light regions (backlit screen) of the image. The digits are joined together to create a continuous blob for each character using the fill technique. The ROI is further processed to extract the contours of the digits. A threshold is applied to the resultant image to extract larger regions in the image to filter out any noise. The image is then cropped using the contour area information to just show the region of the digits. A template-matching OCR technique is applied to the final cropped image. This technique matches the input image to a reference image to recognize digits. A seven-segment royalty-free image (Figure 4) is used as the reference image in this algorithm. An additional fill operation is applied to this image to make the digits continuous, the same as the input image. A distance function is used to calculate scores for the pre-processed contours by using the reference image. The digit with the highest score is selected to estimate the temperature reading.



Figure 4. Reference template used for thermometer digit detection.

The OCR algorithm can be described as follows. Let us call the input image  $I(x, y)$  and the reference image (template)  $T(x, y)$ . The goal of the template-matching OCR technique is to find the highest matching pair using the function  $S(I, T)$ . The ‘correlation coefficient matching’ technique is used to calculate scores for the input image using the equations below [48,49].

$$S(I, T) = \sum_{x', y'} [T'(x', y') \cdot I'(x + x', y + y')]^2 \tag{1}$$

where  $x' = 0 \dots w - 1$ ,  $y' = 0 \dots h - 1$ ,  $w$  and  $h$  are width and height of the template image, and  $T'$  and  $I'$  are defined as

$$T'(x', y') = T(x', y') - \frac{1}{(wh)} \sum_{x', y'} T(x', y') \quad (2)$$

$$I'(x + x', y + y') = I(x + x', y + y') - \frac{1}{(wh)} \sum_{x', y'} I(x + x', y + y') \quad (3)$$

The ROS Tesseract library is used for this OCR recognition task in our study [50]. The library creates a bounding box of the recognized region and displays temperature reading on the image. The reading can be sent to the nurses for monitoring the patient's health condition. Further, the PR2 can be programmed to take multiple temperature readings for better accuracy and to take frequent (bihourly, trihourly, hourly, etc.) to monitor the patient's health.

### 2.6. Patient Walker Algorithm

The patient walker task involves multiple forms of autonomous navigation. The robot makes use of the ROS navigation stack and 2DNav (two-dimensional navigation) method for navigating in dynamic cluttered environments full of obstacles. In addition, the robot uses a modified 2DNav and another simpler base controller for patient walker task. ROS 2DNav is designed to flatten the robot and environment geometry into a two-dimensional plane for path planning and obstacle avoidance. This works well with small objects being carried by the robot's grippers, but will fail if the robot needs to move a larger object. In a hospital-like environment, the robot can be programmed to move a cart, a walker, and an IV pole (intravenous pole), which affects the algorithm's ability to flatten and separate the carried items from the robot and the dynamic environment. The flattened robot footprint was expanded to include the area occupied by either the IV pole or walker. This helps both define the object as being rigidly attached to the robot and avoid collisions between the carried object and environment.

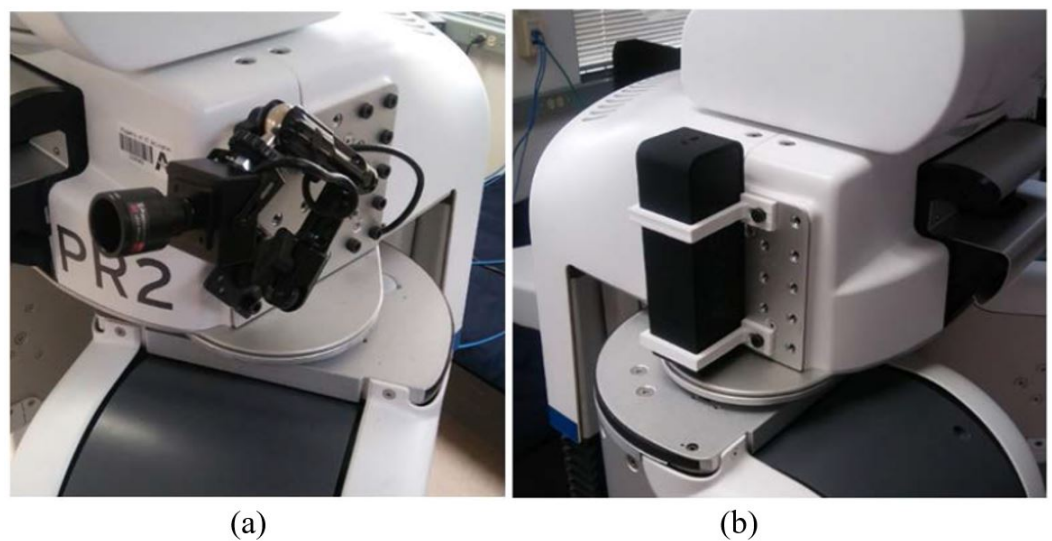
Since the robot follows and supports the user in this task, the robot's motion with the walker should be smooth and easy to operate. The user pushes the walker, therefore applying force on the grippers and leading the robot to a desired location, and the robot understands the user's intentions to walk in a corresponding direction. The PID controller uses traditional force-based logic tries to maintain a desired force all the time during the motion. Since our experiment requires operating at variable speeds, using a PID controller is not suitable for this task. Instead, we adopted a custom controller, called 'Stiffness controller', for this task [51]. When the user selects the 'Start Walker' function on the android tablet, this controller is initialized by PR2. Two parameters, a task position that is in front of the PR2 and a stiffness force parameter, are set before starting the experiment for the controller. When the user applies force greater than that of the stiffness parameter, PR2 grippers move freely to a new position and change the coordinates of the grippers. This motion creates an error in the task space, and to minimize this error, PR2 drives its base and grippers close to the home pose. This technique is used by the PR2 to coordinate and move along with the patient walker. This motion is continued until the patient selects the "Stop Walker" functionality on the android tablet. The controller allows the robot to follow the walker while applying a directionally adjustable level of stiffness to the walker for stability. The walking mode could allow the patient to adjust the stiffness their arms used to hold the walker into position, which then could allow different patients to use the walking mode more comfortably with different settings.

## 3. Hardware and Workspace Description

### 3.1. PR2 Robotic Platform

PR2 is equipped with two onboard computers that run on quad-core Nehalem processors [52]. PR2 has a 1.3 kWh Lion battery pack, which provides an average runtime

of 2 h. The computers can be accessed remotely from a base station to operate PR2 functions [53]. A wide-angle stereo camera and a narrow-angle stereo camera are mounted to the PR2's head. The wide-angle camera is used for face detection and object detection in the experiments. In addition to these, there is a 5 MP (Mega Pixel) camera and a projector mounted to the head. Further, a high-definition camera with optical zoom capability is mounted to the PR2 shoulder as shown in Figure 5a. The camera is angled in such a way to record objects held in PR2's grippers. In this study, this camera is used in thermometer digit detection. The grippers are equipped with pressure-sensor arrays to detect objects held in them. A BLE (Bluetooth Low-Energy) speaker, as shown in Figure 5b, is mounted to the PR2's shoulder to repeat received commands out loud. Two LiDAR scanners are present in the PR2. One is mounted on its torso and the other is on its base. The PR2's base is omni-directional. The motion of the PR2 can also be controlled with a joystick and/or by a keyboard from the base station.



**Figure 5.** (a) High-resolution camera (b) Bluetooth speaker.

### 3.2. Experiment Workspace

Experiments for the project were conducted in Assistive Robotics Laboratory at UTARI. In the laboratory, a hospital setup is created to mimic the real-world environment. Several obstacles such as chairs and tables are added to create a cluttered space. The setup consists of a hospital bed for patients and a table to place objects on for the PR2 to pick up and fetch them. The hospital bed and table are placed 20' (6.1 m) apart and the PR2 start point is placed 9' (2.7 m) away from the bed for our experiments. The PR2 start point and the table are also placed 20' (6.1 m) apart, as shown in Figure 6. We use 'Hill-Rom Hospital bed Model: P3200J000118' for our experiments. The bed dimensions are 83'' (2.1 m) in length and 35'' (0.9 m) in width. The height of the bed can be adjusted from a minimum height of 20'' (0.5 m) to a maximum height of 39'' (1 m). The table used in our experiments has dimensions of 32'' × 24'' × 35'' (0.8 m × 0.6 m × 0.9 m, Length × Width × Height).

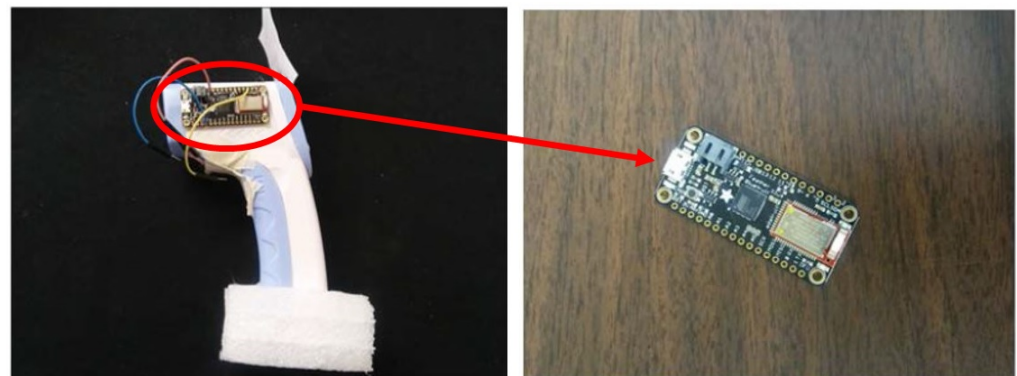




**Figure 6.** Robot's workspace used for human subject testing.

### 3.3. Thermometer

A contactless body thermometer (SinoPie Forehead thermometer) is used to measure the temperature of the patient in this study. A foam base is mounted to the thermometer to stand it upright. A glare filter is added to the thermometer's screen to reduce the effect of surrounding lighting to record the temperature. A Bluetooth microcontroller, 'Adafruit Feather 32u4 Bluefruit LE' (Figure 7) is attached to the thermometer in order to trigger it remotely (Figure 7). The PR2 connects to this module and triggers the thermometer during the temperature measurement task.



**Figure 7.** Non-contact thermometer with Bluetooth trigger.

### 3.4. Patient Walker

In this study, we use 'Drive Medical Walker HX5 9JP' model no. 10226-1 for patient walker experiments. The four-wheeled walker provides easy steering, and the aluminum build makes the walker lightweight, so it requires less effort to walk with. The walker can hold up to 350 lbs (158.8 kg) and has dimensions of 16.75" × 25" (0.4 m × 0.6 m, Length × Width), and comes with 5" (0.1 m) wheels. It provides easy mobility for people with disabilities and elderly people. The walker is modified with a handle to support PR2 robot grippers to hold it, and a shelf is added to place the tablet on during experiments. The final design of the walker is shown in Figure 8.

In order for the patient to be able to rotate relatively easier, the walker was modified to have four caster wheels. In a traditional setting, the extra caster wheels could reduce the

stability granted by the walker, but in this case the robot is used to increase stability for the patient. The casters allow the robot to make use of its dexterous holonomic base and allow the patient to choose between multiple paths to reach the same goal position.



Figure 8. Modified patient walker.

### 3.5. Tablet and Android App User Interface

In order to provide a remotely controlled interface, an Android application software (running Android 5.1 or higher) is developed. The application software (app), named ARNA, includes a custom graphical user interface (GUI) for interacting with the PR2 (running on ROS). The application is developed to communicate and send instructions/information between the tablet and PR2. For this study, we use the Indigo version of ROS on an Ubuntu 14.04 computer. Since android and ROS are not directly compatible, we use ROSJAVA for Android to develop the app. ROSJAVA enables ROS nodes to run on Android devices. The Android tablet acts as a client, which requests items, information and actions to be performed by the robot (PR2). The robot acts as the server, which receives the client requests and processes them. It also sends information over the network to the tablet. Figure 9 shows a screen layout of the user interface. The app is intended to provide two main features for the users: (1) sending commands to the robot and (2) displaying the camera view that the robot sees. In order to send commands to the PR2 robot, the app is intended to allow participants to use either buttons or voice. In order to implement voice commands, Google Android Speech recognition is adopted to process audio from participants. After processing the audio, the app receives text sentences and extracts key words that match commands of interest. The display for the camera view delivers a live video stream from the PR2 cameras. This will be useful when the robot performs tasks away from the user’s view.

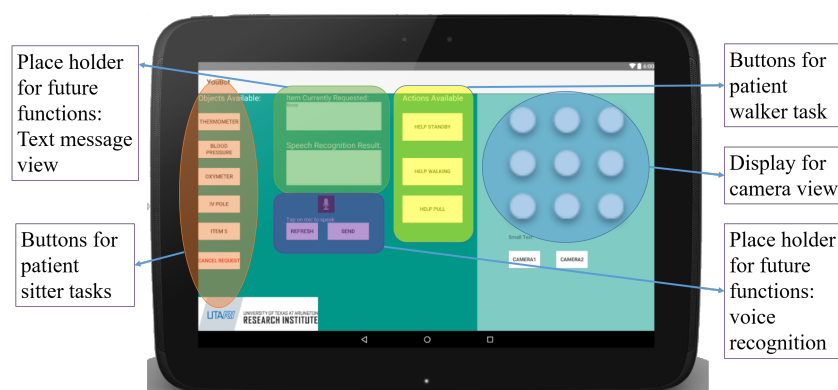


Figure 9. Android app user interface.

## 4. Parameter Selection and Analysis for Defined Nursing Tasks

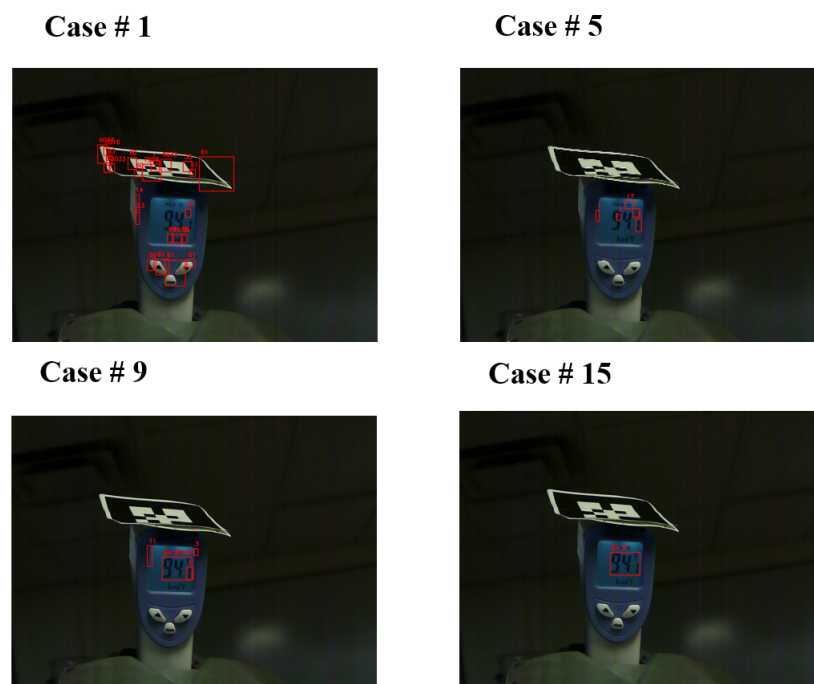
### 4.1. Temperature Measurement Task

A parameter analysis is performed to determine the best set of parameters for thermometer screen digit detection in 15 cases varying the following parameters: threshold to be applied to average score (Th), aspect ratio (AR) for detected contours, size limits for detected contours (Cntr Limits–Width and Height), size of the structural element: rectangle (Rect) and square (Sq), morphological operation to fill the gaps (Fill), full image or cropped image (Crop). The list of the 15 cases with the values of these parameters is given in Table 1. The results of the analysis are evaluated considering three values: detection rate (DR), number of detected contours (#Cntr), and average matching score (AS). The detection rate equals the number of true digits that the algorithm detects over the total number of actual digits. The number of contours gives the total of the contours detected, which may include false positive detections. The results are given in the last three columns of Table 1. According to the results, it can be interpreted that contour-limiting parameters (width and height) and aspect ratio have an effect on eliminating contours other than the digits of interest. In addition to this, adding a threshold to average score is very effective in eliminating false positives. On the other hand, morphological operations (structural size), fill and crop parameter/cases affect the detection of the digits correctly.

**Table 1.** Parameter analysis cases—temperature measurement task.

	Th	AR	Cntr Limits		Struct Size			Crop	DR	#Cntr	AS
			Width	Height	Rect	Sq	Fill				
Case 1	No	0,1.5	0,150	10,200	5,5	5,5	1,1	No	33.33%	29	21345722.64
Case 2	No	0,1.5	0,150	10,200	5,5	5,5	1,1	Yes	33.33%	5	20020376.20
Case 3	No	0,1.5	0,150	10,200	5,5	5,5	25,25	Yes	0.00%	5	19322476.00
Case 4	No	0,1.5	0,150	10,200	5,5	5,5	50,50	Yes	33.33%	5	23890805.00
Case 5	No	0,1.5	0,150	10,200	5,5	5,5	75,75	Yes	33.33%	5	25687678.40
Case 6	No	0,1.5	0,150	10,200	5,5	5,5	100,100	Yes	33.33%	5	24408772.20
Case 7	No	0,1.5	0,150	10,200	10,10	5,5	75,75	Yes	100.00%	8	27724953.75
Case 8	No	0,1.5	0,150	10,200	10,10	10,10	75,75	Yes	100.00%	8	27822773.50
Case 9	No	0,1.5	0,150	10,200	15,15	10,10	75,75	Yes	100.00%	9	29880588.00
Case 10	No	0,1.5	10,150	20,200	15,15	10,10	75,75	Yes	100.00%	7	29329435.71
Case 11	No	0,1.5	20,150	30,200	15,15	10,10	75,75	Yes	100.00%	6	27701941.00
Case 12	No	0,1.5	30,150	40,200	15,15	10,10	75,75	Yes	100.00%	6	27701941.00
Case 13	No	0.5,2	30,150	40,200	15,15	10,10	75,75	Yes	100.00%	6	27701941.00
Case 14	No	0.5,3	30,150	40,200	15,15	10,10	75,75	Yes	100.00%	6	27701941.00
Case 15	Yes	0.5,3	30,150	40,200	15,15	10,10	75,75	Yes	100.00%	3	37202478.67

Figure 10 shows sample case outputs from the analysis. As seen from Table 1 and Figure 10, the cases are given in an increasing performance manner. The performance of the detection algorithm increases with a higher detection rate, and contour number equals the number of digits on the screen. A contour number greater than the actual digit number indicates false positives. The best case desired is when the detection rate is 100% and the contour number is 3, because the actual temperature reads 94.1°F in the parameter analysis (Figure 10). In many cases, the detection rate is 100%, but the contour number is higher than 3. The last case, Case 15, has the parameters that give the best results: 100% detection rate and no false positives. These parameters are used for human subject tests.



**Figure 10.** Examples of parameter analysis results—selected cases #1, #5, #9, and #15.

#### 4.2. Patient Walker Task

A parameter analysis is performed with 11 cases to optimize the robot navigation while retrieving the walker. The analysis is performed along the preferred path the robot has access to during the human subject testing. The following parameters are varied: the maximum linear velocity ( $V_{limit}$ ), the maximum angular velocity ( $W_{limit}$ ), the forward proportional gain ( $P_x$ ), the sideward proportional gain ( $P_y$ ), and the angular proportional gain ( $P_w$ ). These cases are listed with the values of the parameters in Table 2.

The analysis is evaluated by considering four force values and three velocity values. These values are: the maximum recorded force ( $F_{max}$ ), the minimum recorded force ( $F_{min}$ ), the mean of the recorded force ( $F_{mean}$ ), variance in the recorded force ( $F_{var}$ ), the maximum recorded velocity ( $V_{max}$ ), the mean of the recorded velocity ( $V_{mean}$ ), and the variance in the recorded velocity ( $V_{var}$ ). The force values can be separated between the left gripper and right gripper with  $l$  and  $r$  subscripts, such as  $F_{lmax}$  and  $F_{rmax}$ . The results show that the increase in proportional gains leads to an increase in output force values in corresponding direction. Similarly, increasing velocity limits results in higher force output values.

The input parameters in Case 11 are used during the human subject testing. These parameters are chosen in order to reduce the maximum force measured in both grippers to contact the walker handle without pushing it out of the open grippers before grasping, to complete the experiment in a timely manner, and to move the robot without aggressive maneuvers. Case 11 does not have the lowest force for either gripper but keeps the force in both grippers low without raising the force of the opposing gripper, while not increasing the angular speed of the robot. The values in Case 11 are more likely to be able to have the grippers contact the walker handle to grasp it without pushing the handle out of the grippers.



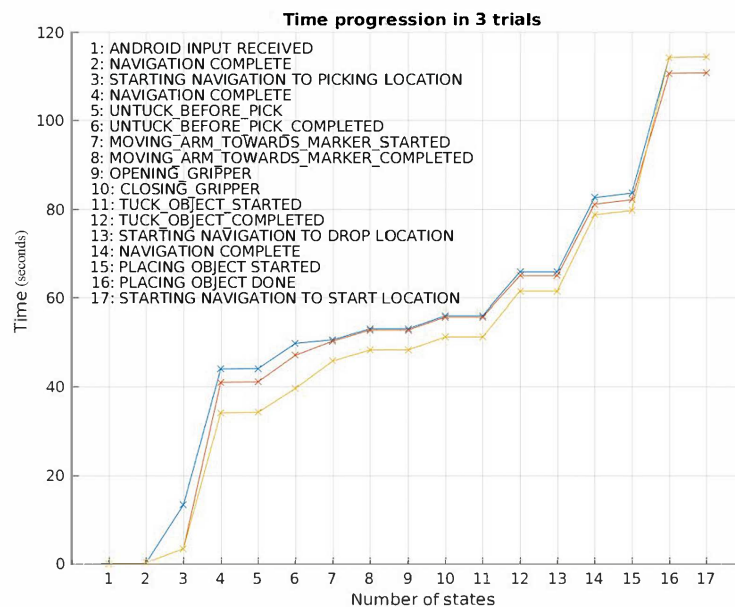
**Table 2.** Parameter analysis cases—patient walker task.

	Input					Output										
	$V_{limit}$	$W_{limit}$	$P_x$	$P_y$	$P_w$	$F_{lmax}$	$F_{rmax}$	$F_{lmin}$	$F_{rmin}$	$F_{lmean}$	$F_{rmean}$	$F_{lvar}$	$F_{rvar}$	$V_{max}$	$V_{mean}$	$V_{var}$
1	0.35	0.45	0.30	0.30	2	30.17	51.56	10.40	11.33	13.70	27.29	1.20	5.59	0.35	0.14	0.02
2	0.30	0.45	0.30	0.30	2	23.88	42.20	12.27	16.18	16.04	25.79	0.98	4.14	0.30	0.14	0.01
3	0.25	0.60	0.30	0.30	2	23.17	45.23	10.55	17.25	13.77	27.63	0.83	4.42	0.24	0.13	0.01
4	0.25	0.52	0.30	0.30	2	21.72	41.99	11.94	20.01	15.20	26.83	0.76	6.66	0.24	0.13	0.01
5	0.25	0.45	0.60	0.30	2	28.16	42.50	10.89	16.31	13.77	27.57	0.60	4.01	0.24	0.17	0.01
6	0.25	0.45	0.45	0.30	2	23.58	38.46	11.49	13.90	15.86	24.99	0.98	3.09	0.24	0.15	0.01
7	0.25	0.45	0.30	0.60	2	22.77	45.35	10.24	16.26	13.37	27.81	0.83	4.50	0.24	0.13	0.01
8	0.25	0.45	0.30	0.42	2	22.83	45.60	10.25	17.69	14.53	26.83	1.17	8.31	0.24	0.13	0.01
9	0.25	0.45	0.30	0.30	2.50	22.95	44.36	10.58	18.96	13.37	28.12	0.83	5	0.24	0.13	0.01
10	0.25	0.45	0.30	0.30	2.25	22.78	41.18	11.87	17.32	15.76	25.07	1.10	3.94	0.24	0.13	0.01
11	0.25	0.45	0.30	0.30	2	23.18	43.94	9.92	14.74	13.75	27.89	1.19	5.42	0.24	0.13	0.01

### 5. Human Subject Tests and Results

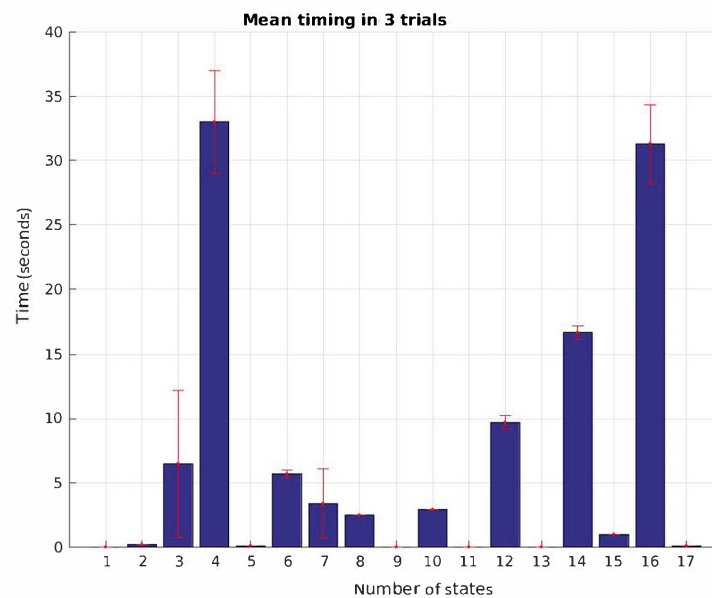
#### 5.1. Object Fetching Task

Object fetching task experiments with human subjects are conducted at UTARI with a total of 11 volunteer participants (10 nursing students and 1 engineering student). The purpose of the experiments is to investigate how people interact with the robot and how the robot detects and responds to this interaction. The tablet with the developed app is used to request fetching three different objects. Subjects either sit or lie on the bed and interact with the robot following the experiment scenario described below. Each subject requests the robot to fetch three different objects. The time to complete each task is recorded and plotted for three trials (three objects are fetched) to show the required average time for this task (Figure 11). The overall fetching task is also broken down to 17 individual smaller tasks and the time to complete each of these tasks is depicted in Figure 12.



**Figure 11.** Time progression for 3 trials.





**Figure 12.** Mean time of 3 trials.

The scenario below describes the fetching task. The fetching task takes about 2 minutes between taking the command and releasing the fetched item to the user.

Scenario:

- A human subject is asked to sit or lie on a hospital bed (pretending to be a patient in a hospital). The subject is asked to use buttons on the tablet to interact with the PR2 during the experiment.
- The PR2 robot's starting position is nearby the patient, about 6 feet (1.8 m) away.
- The PR2 robot detects a human face and start tracking the subject's face position.
- The PR2 robot says "Please interact with the tablet".
- The subject pushes a button on the tablet to request a fetch task. Objects that can be fetched are a soda bottle, water, or cereal box. Once the PR2 receives the tablet input, first, it moves to its starting pose to start the experiment (step 2 in Figure 11).
- The PR2 robot acknowledges the subject's command from the tablet and starts moving toward a table located about 20 feet (6.1 m) away from the bed.
- The PR2 robot stops near the table and picks up the requested object on the table (Figure 13).
- The PR2 robot brings the object near to the bed, about 3 to 4 feet (0.9–1.2 m) away from the subject.
- The subject is asked to take the object from the robot.
- The robot releases the object (Figure 14).
- This task is repeated a total of three times for each subject.

Observations:

- The robot's navigation velocity is programmed to a max limit of 0.3 m/s forward and 0.1 m/s backward. The average time to fetch objects from a travel distance of 29 feet (8.8 m) is in the range of 120–160 s (average 136.66 s with a standard deviation of 17.98 s).
- Considering that the time for a person to complete the same fetching task is a few seconds, the robot's speed needs to be improved for better efficiency.
- The fetching tasks are completed with a success rate of 94.12% out of 34 trials (11 subjects  $\times$  3 trials + 1 additional trial for one subject). This rate is based on the robot returning the correct object directly from the tablet input. The failures (only to occurrences) include both the robot returning the wrong object due to wrong detection (computer vision) and the robot returning with nothing due to a bad grasp.

- The robot was stuck two times during navigation due to moving over the bed sheet. The robot is sensitive to obstacles under the wheels. When the wheels pass over the cloth, they pull the cloth closer to the robot, blocking some of the sensors and this impedes the path planning.
- In one trial, the subject pushes multiple buttons unknowingly. Multiple item retrieval messages are sent to the robot. Each additional input is seen as a correction or change of command and overwrites the prior item message.
- The robot's arm hits the table two times when reaching out for objects on two separate trials. The path planning for arm manipulation is not appropriate with a reduced distance between the robot and table.
- Comments are collected from the human subjects. Some examples of those comments are as follows:
  - “The fetching speed is slow.”
  - “Face tracking is a good feature making the robot more human like in interaction, however the constant tracking and searching can cause negative effects. Depending on the requirements of the patient profile the face tracking behavior should vary.”



**Figure 13.** PR2 robot picks up an object during fetching task.



**Figure 14.** Snapshot of a fetching task example.

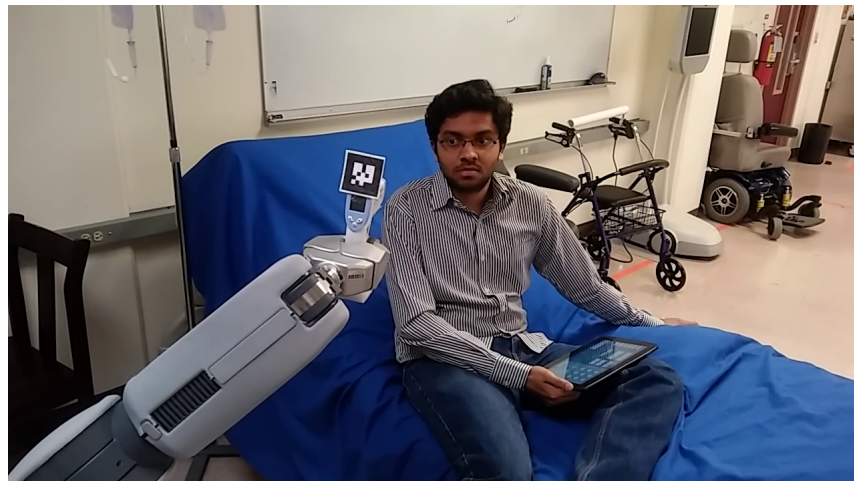
### 5.2. Temperature Measurement Task

Human subject tests are performed with eight volunteers over 2 days for the temperature measurement task. The designed test scenario is as follows: A human subject is asked to lie on the bed, and once the PR2 receives the temperature measurement task request, it navigates next to the table to pick up the thermometer (Figure 15), navigates back next

to the patient, finds the patient's face in order to direct the thermometer, and move its arm with thermometer to the calculated position (Figure 16). Then, the thermometer is triggered by a Bluetooth module. Finally, the PR2 moves its arm with thermometer close to the high-definition camera and a single image is saved for detection purposes.



**Figure 15.** PR2 robot picks up the thermometer.



**Figure 16.** Snapshot of a human temperature measurement.

The list of observations during human subject experiments are given below:

- Two times, the patients lay down quite low on the bed. It takes longer for the PR2 to find the subject's face.
- Three times, subjects pushed the button twice.
- One time, the PR2 hit the table when lifting the arm during the thermometer pick-up phase.
- One subject removed glasses while the PR2 pointed the thermometer.
- Some examples of human subjects' comments are:
  - "It looks like the robot from the Jetsons".
  - "The speed of the robot is too slow and that the tablet interface can be improved".
  - "Can the supplies be put on the robot?"

The thermometer digit detection results from human subject tests are given in Table 3. In two out of eight human subject cases, the system reads the thermometer screen 100% correctly with no false positive contours. The system also has 100% for two more cases; however, there are 1 and 3 false positive detections in those cases, respectively. Three out of the remaining four cases ends up with a 33% detection rate, and there is one case with

a 66% detection rate. Some examples of resulting images from human subject tests are shown in Figure 17. When the parameter analysis is performed, we defined an ROI in the image using known locations of the arm of PR2, the camera, and the thermometer. During human subject tests, we realize that, depending on how the PR2 picks up the thermometer, the orientation of the thermometer in the gripper may change. Even though the orientation difference is very small, it highly affects the performance of the detection algorithm. Additionally, lighting conditions may contribute to the high false positive rate. The possible solutions to improve detection include (i) modifying the thermometer to allow the PR2 to pick it up the exact same way every time, (ii) adding LED lights around the camera to improve visibility of the digits, and (iii) defining dynamic and adaptive ROI using visual markers around thermometer screen.



Figure 17. Examples of human subject test results—selected tests #1, #4, #5, and #8.

Table 3. Human subject test results—temperature measurement task.

	Actual Temp. (°F)	System Output	Detection %	Correct Digit %	# of False Positives
Subject 1	76.5	215.151	33%	0%	5
Subject 2	73.2	2	33%	0%	0
Subject 3	72	72.1	66%	66%	1
Subject 4	79.2	79.2	100%	100%	0
Subject 5	77.7	77.7	100%	100%	0
Subject 6	76.3	43.7631	100%	0%	3
Subject 7	77.9	77.191	100%	66%	2
Subject 8	76.3	7	33%	0%	0



### 5.3. Patient Walker Task

Human subject tests are performed for the patient walker task with a total of eight volunteers. The patient walker task begins with the patient in a bed and having access to the tablet to communicate with the robot. A customized walker is stored in a separate location.

When the patient selects the walker task on the tablet, the robot will navigate to retrieve the walker using the ROS 2DNav algorithm [39]. Once the robot is positioned in front of the walker, it places its arms into the gripping position. The multimodal proportional controller is used to contact the walker. The robot closes its grippers and uses the controller to gently push the walker to the patient's bed. The patient can then stand up, place the tablet onto the walker, and use the tablet to turn the walking mode on the robot. The patient can then push and pull the walker in any direction. The robot will sense the motion of the walker and follow it, while limiting the walker's speed for stability. When the patient arrives at his/her desired location, he/she can turn off the walking mode and the robot will hold the walker rigidly in place. A snapshot from a test run is depicted in Figure 18.

The comments from volunteers and observations during the human subject tests are given below, which are provided as recommendations for the development of custom ARNA platforms.

Observations:

- Patient cannot be sure when to press the button (Test 1).
- PR2 has a hard time navigating to the walker (Test 1).
- PR2 has a hard time finding the walker (Test 1).
- One of the grippers misses the walker handle (Test 1).
- Patient says turning is tricky (Test 1).
- Patient forgets to turn off the walker mode (Test 4).
- Initialization is failed, and the experiment is started over (Test 5).
- During navigation to the walker, the PR2 failed. The experiment is restarted (Test 5).
- During navigation to the walker, the PR2 failed again. Experiment is restarted (Test 5).
- Patient says that rotation is hard and tricky (Test 6).



Figure 18. Snapshot of a test for patient walker task.

## 6. Conclusions

In this study, we present outcomes of nursing assistant task design, analysis, and human subject test results using an assistive robot (PR2). Our main focus is to implement three tasks: object fetching and temperature measurement (patient sitter), and a patient walker task for assisting patients with basic tasks. Parameter analysis is performed and the parameters with the best results are selected to be used in the human subject tests. Human subject tests are performed with 27 volunteers in total. In the experiments with human subjects, in all cases the algorithm works successfully in assisting volunteers with



the corresponding task. This study is part of a larger research effort in which the system is aimed to be integrated on an adaptive robotic nursing assistant (ARNA) platform.

**Author Contributions:** Conceptualization, methodology, investigation, writing—original draft preparation, C.L.L., H.E.S.; supervision, project administration, writing—review and editing, H.E.S., D.B., D.O.P. All authors have read and agreed to the published version of the manuscript.

**Funding:** This work was supported by the National Science Foundation (NSF) Partnerships for Innovation: Building Innovation Capacity (PFI: BIC) grant (Award Number: 1643989).

**Institutional Review Board Statement:** This study was approved by the Institutional Review Board (IRB) of the University of Texas at Arlington (IRB Protocol Number: 2015-0780).

**Informed Consent Statement:** Informed consent was obtained from all subjects involved in the study.

**Data Availability Statement:** Not applicable.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Landro, L. Nurses Shift, Aiming for More Time with Patients. Available online: <https://www.wsj.com/articles/nurses-shift-aiming-for-more-time-with-patients-1405984193> (accessed on 3 July 2018).
2. Hillman, M.; Hagan, K.; Hagan, S.; Jepson, J.; Orpwood, R. A wheelchair mounted assistive robot. *Proc. ICORR* **1999**, *99*, 86–91.
3. Park, K.H.; Bien, Z.; Lee, J.J.; Kim, B.K.; Lim, J.T.; Kim, J.O.; Lee, H.; Stefanov, D.H.; Kim, D.J.; Jung, J.W.; et al. Robotic smart house to assist people with movement disabilities. *Auton. Robot.* **2007**, *22*, 183–198. [[CrossRef](#)]
4. Driessen, B.; Evers, H.; Woerden, J. MANUS—A wheelchair-mounted rehabilitation robot. *Proc. Inst. Mech. Eng. Part J. Eng. Med.* **2001**, *215*, 285–290. [[CrossRef](#)] [[PubMed](#)]
5. Kim, D.J.; Lovelett, R.; Behal, A. An empirical study with simulated ADL tasks using a vision-guided assistive robot arm. In Proceedings of the 2009 IEEE International Conference on Rehabilitation Robotics, Kyoto, Japan, 23–26 June 2009; pp. 504–509.
6. Tsumaki, Y.; Kon, T.; Suginuma, A.; Imada, K.; Sekiguchi, A.; Nenchev, D.N.; Nakano, H.; Hanada, K. Development of a skincare robot. In Proceedings of the 2008 IEEE International Conference on Robotics and Automation, Pasadena, CA, USA, 19–23 May 2008; pp. 2963–2968.
7. Koga, H.; Usuda, Y.; Matsuno, M.; Ogura, Y.; Ishii, H.; Solis, J.; Takanishi, A.; Katsumata, A. Development of oral rehabilitation robot for massage therapy. In Proceedings of the 2007 6th International Special Topic Conference on Information Technology Applications in Biomedicine, Tokyo, Japan, 8–11 November 2007; pp. 111–114.
8. Kidd, C.D.; Breazeal, C. Designing a sociable robot system for weight maintenance. In Proceedings of the IEEE Consumer Communications and Networking Conference, Las Vegas, NV, USA, 8–10 January 2006; pp. 253–257.
9. Kang, K.I.; Freedman, S.; Mataric, M.J.; Cunningham, M.J.; Lopez, B. A hands-off physical therapy assistance robot for cardiac patients. In Proceedings of the 9th International Conference on Rehabilitation Robotics, Chicago, IL, USA, 28 June–1 July 2005; pp. 337–340.
10. Wada, K.; Shibata, T.; Saito, T.; Tanie, K. Robot assisted activity for elderly people and nurses at a day service center. In Proceedings of the 2002 IEEE International Conference on Robotics and Automation (Cat. No. 02CH37292), Washington, DC, USA, 11–15 May 2002; Volume 2, pp. 1416–1421.
11. Obayashi, K.; Kodate, N.; Masuyama, S. Socially assistive robots and their potential in enhancing older people’s activity and social participation. *J. Am. Med. Dir. Assoc.* **2018**, *19*, 462–463. [[CrossRef](#)] [[PubMed](#)]
12. Kim, E.S.; Berkovits, L.D.; Bernier, E.P.; Leyzberg, D.; Shic, F.; Paul, R.; Scassellati, B. Social robots as embedded reinforcers of social behavior in children with autism. *J. Autism Dev. Disord.* **2013**, *43*, 1038–1049. [[CrossRef](#)] [[PubMed](#)]
13. Tapus, A.; Fasola, J.; Mataric, M.J. Socially assistive robots for individuals suffering from dementia. In Proceedings of the ACM/IEEE 3rd Human-Robot Interaction International Conference, Workshop on Robotic Helpers: User Interaction, Interfaces and Companions in Assistive and Therapy Robotics, Amsterdam, The Netherlands, 12–15 March 2008.
14. Tapus, A. Improving the quality of life of people with dementia through the use of socially assistive robots. In Proceedings of the 2009 Advanced Technologies for Enhanced Quality of Life, Iasi, Romania, 22–26 July 2009; pp. 81–86.
15. Zemg, J.-J.; Yang, R.Q.; Zhang, W.-J.; Weng, X.-H.; Qian, J. Research on semi-automatic bomb fetching for an EOD robot. *Int. J. Adv. Robot. Syst.* **2007**, *4*, 27.
16. Bluethmann, W.; Ambrose, R.; Diftler, M.; Askew, S.; Huber, E.; Goza, M.; Rehnmark, F.; Lovchik, C.; Magruder, D. Robonaut: A robot designed to work with humans in space. *Auton. Robot.* **2003**, *14*, 179–197. [[CrossRef](#)] [[PubMed](#)]
17. Diftler, M.A.; Ambrose, R.O.; Tyree, K.S.; Goza, S.; Huber, E. A mobile autonomous humanoid assistant. In Proceedings of the 4th IEEE/RAS International Conference on Humanoid Robots, Santa Monica, CA, USA, 10–12 November 2004; Volume 1, pp. 133–148.

18. Taipalus, T.; Kosuge, K. Development of service robot for fetching objects in home environment. In Proceedings of the 2005 International Symposium on Computational Intelligence in Robotics and Automation, Espoo, Finland, 27–30 June 2005; pp. 451–456.
19. Nguyen, H.; Anderson, C.; Trevor, A.; Jain, A.; Xu, Z.; Kemp, C.C. El-e: An assistive robot that fetches objects from flat surfaces. In Proceedings of the Robotic Helpers, International Conference on Human-Robot Interaction, Amsterdam, The Netherlands, 12 March 2008.
20. Natale, L.; Torres-Jara, E. A sensitive approach to grasping. In *Proceedings of the Sixth International Workshop on Epigenetic Robotics*, Paris, France, 20–22 September, 2006; pp. 87–94.
21. Saxena, A.; Driemeyer, J.; Ng, A.Y. Robotic grasping of novel objects using vision. *Int. J. Robot. Res.* **2008**, *27*, 157–173. [[CrossRef](#)]
22. Pettinaro, G.C.; Gambardella, L.M.; Ramirez-Serrano, A. Adaptive distributed fetching and retrieval of goods by a swarm-bot. In Proceedings of the ICAR'05, 12th International Conference on Advanced Robotics, Seattle, WA, USA, 18–20 July 2005; pp. 825–832.
23. Morris, A.; Donamukkala, R.; Kapuria, A.; Steinfeld, A.; Matthews, J.T.; Dunbar-Jacob, J.; Thrun, S. A robotic walker that provides guidance. In Proceedings of the 2003 IEEE International Conference on Robotics and Automation (Cat. No. 03CH37422), Taipei, Taiwan, 14–19 September 2003; Volume 1, pp. 25–30.
24. Dubowsky, S.; Genot, F.; Godding, S.; Kozono, H.; Skwersky, A.; Yu, H.; Yu, L.S. PAMM-A robotic aid to the elderly for mobility assistance and monitoring: A “helping-hand” for the elderly. In Proceedings of the 2000 ICRA. Millennium Conference. IEEE International Conference on Robotics and Automation. Symposia Proceedings (Cat. No. 00CH37065), San Francisco, CA, USA, 24–28 April 2000; Volume 1, pp. 570–576.
25. Wakita, K.; Huang, J.; Di, P.; Sekiyama, K.; Fukuda, T. Human-walking-intention-based motion control of an omnidirectional-type cane robot. *IEEE/ASME Trans. Mechatron.* **2011**, *18*, 285–296. [[CrossRef](#)]
26. Wang, H.; Sun, B.; Wu, X.; Wang, H.; Tang, Z. An intelligent cane walker robot based on force control. In Proceedings of the 2015 IEEE International Conference on Cyber Technology in Automation, Control, and Intelligent Systems (CYBER), Shenyang, China, 8–12 June 2015; pp. 1333–1337.
27. Lacey, G.; Dawson-Howe, K.M. The application of robotics to a mobility aid for the elderly blind. *Robot. Auton. Syst.* **1998**, *23*, 245–252. [[CrossRef](#)]
28. Huang, J.; Di, P.; Fukuda, T.; Matsuno, T. Motion control of omni-directional type cane robot based on human intention. In Proceedings of the 2008 IEEE/RSJ International Conference on Intelligent Robots and Systems, Nice, France, 22–26 September 2008; pp. 273–278.
29. Yuen, S.G.; Novotny, P.M.; Howe, R.D. Quasiperiodic predictive filtering for robot-assisted beating heart surgery. In Proceedings of the 2008 IEEE International Conference on Robotics and Automation, Pasadena, CA, USA, 19–23 May 2008; pp. 3875–3880.
30. Harjo, P.; Knuutila, J.; Taipalus, T.; Vallet, J.; Halme, A. Automation and telematics for assisting people living at home. *Ifac Proc. Vol.* **2005**, *38*, 13–18. [[CrossRef](#)]
31. Abdullah, M.F.L.; Poh, L.M. Mobile robot temperature sensing application via bluetooth. *Int. J. Smart Home* **2011**, *5*, 39–48.
32. Der Loos, V.; Machiel, H.; Ullrich, N.; Kobayashi, H. Development of sensate and robotic bed technologies for vital signs monitoring and sleep quality improvement. *Auton. Robot.* **2003**, *15*, 67–79. [[CrossRef](#)]
33. Kuo, I.H.; Broadbent, E.; MacDonald, B. Designing a robotic assistant for healthcare applications. In Proceedings of the 7th Conference of Health Informatics, Rotorua, New Zealand, 15–17 October 2008.
34. Cremer, S.; Doelling, K.; Lundberg, C.L.; McNair, M.; Shin, J.; Popa, D. Application requirements for Robotic Nursing Assistants in hospital environments. *Sensors -Next-Gener. Robot. III* **2016**, *9859*, 98590E.
35. Das, S.K.; Sahu, A.; Popa, D.O. Mobile app for human-interaction with sitter robots. In *Smart Biomedical and Physiological Sensor Technology XIV*; International Society for Optics and Photonics, Anaheim, CA, USA 2017; Volume 10216, p. 102160D.
36. Dalal, A.V.; Ghadge, A.M.; Lundberg, C.L.; Shin, J.; Sevil, H.E.; Behan, D.; Popa, D.O. Implementation of Object Fetching Task and Human Subject Tests Using an Assistive Robot. In the Proceedings of the ASME 2018 Dynamic Systems and Control Conference (DSCC 2018), Atlanta, GA, USA, 30 September–3 October 2018; DSCC2018-9248.
37. Ghadge, A.M.; Dalal, A.V.; Lundberg, C.L.; Sevil, H.E.; Behan, D.; Popa, D.O. *Robotic Nursing Assistants: Human Temperature Measurement Case Study*. In the Proceedings of the Florida Conference for Recent Advances in Robotics (FCRAR 2019), Lakeland, FL, USA, 9–10 May 2019.
38. Fina, L.; Lundberg, C.L.; Sevil, H.E.; Behan, D.; Popa, D.O. *Patient Walker Application and Human Subject Tests with an Assistive Robot*. In the Proceedings of the Florida Conference for Recent Advances in Robotics (FCRAR 2020), Melbourne, FL, USA, 14–16 May 2020.
39. ROS Wiki. Navigation Package Summary. Available online: <http://wiki.ros.org/navigation> (accessed on 10 April 2018).
40. ROS Wiki. Amcl Package Summary. Available online: <http://wiki.ros.org/amcl> (accessed on 10 April 2018).
41. ROS Wiki. ar\_track\_alvar Package Summary. Available online: [http://wiki.ros.org/ar\\_track\\_alvar](http://wiki.ros.org/ar_track_alvar) (accessed on 10 April 2018).
42. ROS Wiki. Face\_detector Package Summary. Available online: [http://wiki.ros.org/face\\_detector](http://wiki.ros.org/face_detector) (accessed on 10 April 2018).
43. OpenCV. Face Detection using Haar Cascades. Available online: [https://docs.opencv.org/trunk/d7/d8b/tutorial\\_py\\_face\\_detection.html](https://docs.opencv.org/trunk/d7/d8b/tutorial_py_face_detection.html) (accessed on 10 April 2018).
44. Robots and Androids. Robot Face Recognition. Available online: <http://www.robots-and-androids.com/robot-face-recognition.html> (accessed on 10 April 2018).
45. ROS Wiki. tf Library Package Summary. Available online: <http://wiki.ros.org/tf> (accessed on 10 April 2018).

46. OMPL. The Open Motion Planning Library. Available online: <http://ompl.kavrakilab.org/> (accessed on 10 April 2018).
47. MoveIt! Website Blog. Moveit! Setup Assistant. Available online: [http://docs.ros.org/indigo/api/moveit\\_tutorials/html/doc/setup\\_assistant/setup\\_assistant\\_tutorial.html](http://docs.ros.org/indigo/api/moveit_tutorials/html/doc/setup_assistant/setup_assistant_tutorial.html) (accessed on 10 April 2018).
48. Muda, N.; Ismail, N.K.N.; Bakar, S.A.A.; Zain, J.M. Optical character recognition by using template matching (alphabet). In Proceedings of the National Conference on Software Engineering & Computer Systems 2007 (NACES 2007), Kuantan, Malaysia, 20–21 August 2007.
49. Bradski, G.; Kaehler, A. *Learning OpenCV: Computer Vision with the OpenCV Library*; O'Reilly Media, Inc.: Newton, MA, USA, 2008.
50. GitHub. Tesseract Open Source OCR Engine. Available online: <https://github.com/tesseract-ocr/tesseract> (accessed on 10 April 2018).
51. Cremer, S.; Ranatunga, I.; Popa, D.O. Robotic waiter with physical co-manipulation capabilities. In Proceedings of the 2014 IEEE International Conference on Automation Science and Engineering (CASE), New Taipei, Taiwan, 18–22 August 2014; pp. 1153–1158.
52. Rockel, S.; Klimentjew, D. ROS and PR2 Introduction. Available online: [https://tams.informatik.uni-hamburg.de/people/rockel/lectures/ROS\\_PR2\\_Introduction.pdf](https://tams.informatik.uni-hamburg.de/people/rockel/lectures/ROS_PR2_Introduction.pdf) (accessed on 10 April 2018).
53. Willow Garage. PR2 Overview. Available online: <http://www.willowgarage.com/pages/pr2/overview> (accessed on 10 April 2018).