Auxilio Robotics

Propelling Assistive Healthcare into the Future

Spring Project Test Plan
Team F

February 13, 2023

Team Members
Abhinav Gupta
Atharva Pusalkar
Praveen Venkatesh
Shaolin Kataria
Shivam Tripathy

Sponsors: Prof Yonatan Bisk (LTI), Prof Zackory Erickson (RI)

Contents

1 Introduction

This document details the tests that team Auxilio Robotics will perform during the Spring Semester of 2023. For each test, the objective, associated requirements, equipment required, elements involved, personnel in charge, test location, and verification criteria have been listed. The purpose of these tests is to verify that our system meets the performance requirements listed in Appendix A. This document also contains a schedule that highlights what milestones we plan to achieve by a set PR date, the tests it'll involve, and the requirements that will be covered.

2 Logistics

All tests will be performed in our development and testing space - AI Makerspace, Tepper School of Business. A single team member will be the primary person in charge of the tests. However, an assistant will help the lead tester with required tasks, such as keeping an eye on the robot in case of unpredictable behavior.

3 Schedule

PR #	Date	Capability Milestone	Test	Requirements
PR#1	**16 Feb**	- HRI: Accurate speech parsing on the robot - Accurate object detection for 1 object	1, 2	M.P.1.2, M.P.6
PR#2	**2 Mar**	Robot can localize, plan & navigate environment while avoiding obstacles	3, 4, and 6	M.P.3, M.P.4.1, M.P.4.2, M.P.5
PR#3	**23 Mar**	- The manipulation pipeline is functional and can pick/place 1 object. - Robot can understand 5 different template based speech commands	5, 7, 8, 9, 10	M.P.8, M.P.9, M.P.1.1, M.P.7
PR#4	**6 Apr**	Entire robot is tested	11	Dry Run (as in table 1, section 4.11)
PR#5 (SVD)	**19 Apr**	Entire robot is tested	11	As in table 1, section 4.11
PR#6 (SVD Encore)	**26 Apr**	Entire robot is demonstrated + late breaking results	11	-

4 Tests

4.1 Test 1: Speech Recognition Test

Test 1: Speech Recognition Test | Objective	
Assess the latency of the speech recognition pipeline on the robot.	
Associated Requirements	M.P.1.2 (M.F.1)
Equipment	Stretch RE1
Elements	HRI - Speech Recognition
Personnel	Lead: Atharva | Assistant: Praveen
Location	AI Makerspace (Tepper School of Business)
Procedure	
1. Turn the robot on and home it. 2. Activate the speech recognition pipeline. 3. Activate the test script that measures the time between completion of voice command input and transcription. 4. Praveen (patient) will speak (in a normal voice) a random sentence (from https://randomwordgenerator.com/sentence.php). 5. The recognized voice will be transcribed. The time duration for this is measured using the test script. 6. Steps 3-5 are repeated 10 times. The mean time for transcription is then calculated. 7. Kill the script and turn the robot off.	
Verification Criteria	
Average transcription time should be less than 3.5 seconds. Although our performance requirements have a desired latency of within 5 seconds, we are only partially testing the robot's speech system as a transcription to control module is to be implemented as well, and thus we require the robot to transcribe test within 2.5 seconds of completing the command. We will test this at a later point of time.	

4.2 Test 2: Object Detection Preliminary Test

Test 2: Object Detection Preliminary Test | Objective	
Assess the performance of the object detection pipeline for a single object category (bottles) under controlled conditions.	
Associated Requirements	M.P.6 (M.F.6)
Equipment	Stretch RE1, remote compute server, and coloured cellophane tape
Elements	Manipulation Subsystem - Perception
Personnel	Lead: Praveen | Assitant: Abhinav
Location	AI Makerspace (Tepper School of Business)
Procedure	
1. Setup the test environment as follows: a) Place the robot at a fixed location in front of the table. b) Mark 5 locations on the table (using coloured tape) where a given object can be placed. Ensure that there is sufficient lighting on the table. c) Place 5 different bottles of varying shapes and sizes on each of the locations of interest. 2. Turn on the robot, home it and enable the camera node. 3. Using the robot, collect a dataset of images using the "rosbag" tool that is supplied with ROS. The dataset should be collected as follows: a) With a given setup of 5 bottles in their locations, run the rosbag for 5 seconds. b) Interchange the positions of the bottles, and repeat "3. a)". This is done 3 times. 4. Using the collected dataset, mark the bounding boxes on the actual location of the object in the image on the dataset. This will serve as the ground truth. 5. Run the object detection algorithm on the collected rosbag, and compute the mAP with respect to the annotated ground truth collected in the previous step. 6. Kill the script and turn the robot off.	
Verification Criteria	
The computed mAP is >= 80% for different kinds of bottles in each 5 second interval.	

4.3 Test 3: Localization Test

Test 3: Localization Test | Objective	
Assess the performance of the localization algorithm.	
Associated Requirements	M.P.3 (M.F.3)
Equipment	Stretch RE1, coloured cellophane tape
Elements	Navigation Subsystem
Personnel	Lead: Abhinav | Assistant: Shivam
Location	AI Makerspace (Tepper School of Business)
Procedure	
1. Ensure that the room in which the robot is navigating has been mapped and that the map has been loaded on the robot. 2. Select 10 different locations of interest in the testing environment. Obtain their locations (x, y) and mark the ground using coloured cellophane tape. This location is obtained using a traditional tape measure with respect to other local objects near the location of interest. 3. Start the robot and home it. 4. Enable the localization algorithm node on the robot. 5. From an arbitrary location (that is not the location of interest), teleoperate the robot to each of the locations of interest sequentially, pausing at each location for a duration of 10 seconds and noting down the estimated localization coordinates. 6. Kill the script and turn the robot off.	
Verification Criteria	
The average L2 norm between the true and estimated localization coordinates should be less than 25cms.	

4.4 Test 4: Path Planning Speed Test

Test 4: Path Planning Speed Test | Objective	
Assess the generation speed of the path planning algorithm.	
Associated Requirements	M.P.4.1 (M.F.4)
Equipment	Stretch RE1
Elements	Navigation Subsystem
Personnel	Shivam
Location	AI Makerspace (Tepper School of Business)
Procedure	
1. Start the robot and home it. 2. Using the test script, load the map of the environment on the robot. 3. Generate 10,000 random start and end points within the given map algorithmically. The points are sampled using a uniform distribution. 4. Run the path planning algorithm on the robot using each of the above start and end locations. The time taken to run the path planning algorithm on each of these paths is saved down in a file. 5. Kill the script and turn the robot off.	
Verification Criteria	
The average time taken for path planning is less than 2 minutes.	

4.5 Test 5: Manipulation Speed Test

Test 5: Manipulation Speed Test | Objective	
Assess the speed of execution of manipulation tasks.	
Associated Requirements	M.P.7.2 (M.F.7)
Equipment	Stretch RE1, objects for grasping
Elements	Manipulation Subsystem
Personnel	Lead: Shaolin | Assistant: Praveen
Location	AI Makerspace (Tepper School of Business)
Procedure	
1. Setup the test environment as follows: a) Place the robot in front of a table. b) Ensure adequate lighting is present. c) Keep 10 objects (of different categories) ready for testing. 2. Pick an object for manipulation and place it on the table within the field of view of the robot. 3. Start the robot and home it. 4. Issue a command to pick up the object of interest. Note down the start time (programmatically). 5. Once the object has been picked up, note down the time (programmatically) it took to execute said task. 6. Issue a command to place the object of interest. Note down the start time (programmatically). 7. Once the object has been placed down, note down the time (programmatically) it took to execute said task. 8. Repeat steps 4-7 for all 10 different objects. 9. Kill the script and turn the robot off.	
Verification Criteria	
Each manipulation task (pick-and-place individually) complete within 8 minutes of issuing the command.	

4.6 Test 6: Obstacle Avoidance Test

Test 6: Obstacle Avoidance Test | Objective	
Assess obstacle avoidance against 10 static obstacles.	
Associated Requirements	M.P.5 (M.F.5)
Equipment	Stretch RE1, 10 static obstacles
Elements	Navigation Subsystem
Personnel	Lead: Shivam | Assitant: Abhinav
Location	AI Makerspace (Tepper School of Business)
Procedure	
1. Turn the robot on and home it. Check if the robotic arm is stowed. 2. If the arm is not stowed, run the script to stow the arm. 3. Set up a test environment with 10 different obstacles (with varying shapes and sizes) on a path from the robot's starting location to the robot's goal location. 4. Issue a navigation command for the robot from the start to the end location. 5. Abhinav to stay on-alert within the vicinity of the robot to be able to press the kill switch button in case the robot starts behaving undesirably. 6. Make a note of success/ failure as the robot passes a pre-defined list of 10 objects. 7. Kill the script and turn the robot off.	
Verification Criteria	
The robot was able to navigate in the desired path while avoiding all 10 obstacles.	

4.7 Test 7: Teleoperation Command Latency Test

Test 7: Teleoperation Command Latency Test | Objective	
Assess latency of teleoperation commands on the robot for 3 minutes.	
Associated Requirements	M.P.8 (M.F.8)
Equipment	Stretch RE1, Tablet
Elements	HRI Subsystem - Teleoperation
Personnel	Lead: Atharva | Assistant: Praveen
Location	AI Makerspace (Tepper School of Business)
Procedure	
1. Turn the robot on and home it. 2. Atharva to turn on the mobile application on the tablet and request to connect to the robot. 3. Praveen to play the role of a patient and accept Atharva's request to teleoperate the robot. 4. Atharva gives commands on the application on the tablet to maneuever the robot. This is noted as timestamp 1 in our test verification script. 5. Praveen now plays the role of emergency rescue and stays on standby near the robot to be able to press the kill switch button in case the robot starts behaving undesirably. 6. Our test verification script notes the second timestamp which depicts the time it takes for the robot to execute the command that Atharva passed as the instruction on the application on the tablet. This can be either change of coordinates in navigation, or change of pose of arm in manipulation. 7. Repeat steps 4-6 for 3 minutes and store the timestamp differences as an array. 8. Once step 7 is finished, disconnect the user from the teleoperation mode, kill the script, and turn the robot off.	
Verification Criteria	
The loved one (Atharva) was able to teleoperate the robot with the time between his commands and the resulting robot operation (latency) not once exceeding 5 seconds. The proof of this will be visible in the timestamp difference array for a period of 3 minutes.	

4.8 Test 8: Teleoperation Robot Feedback for Mobile Application Test

Test 8: Teleoperation Robot Feedback for Mobile Application Test | Objective	
Assess the latency of robot feedback on the mobile application for 3 minutes.	
Associated Requirements	M.P.9 (M.F.9)
Equipment	Stretch RE1, Tablet
Elements	HRI Subsystem - Robot Metrics and Video Feed
Personnel	Lead: Atharva | Assistant: Praveen
Location	AI Makerspace (Tepper School of Business)
Procedure	
1. Turn the robot on and home it. Ensure that the clocks on the tablet and the robot are synced to the global clock. 2. Praveen to play the role of a patient and give a teleoperation command to the robot. 3. The robot acts on the teleoperation command. The image sent packet here is timestamp 1 for the test verification script. 4. Atharva plays the role of emergency rescue and stays on standby near the robot to be able to press the kill switch button in case the robot starts behaving undesirably. 5. Praveen checks the robot metrics and the video feed he is seeing on the tablet. This is image received packet and serves as timestamp 2 for the test verification script. 6. The test verification script augments these timestamp differences (between timestamp 1 and timestamp 2) in an array. 7. Repeat steps 2-6 for 3 minutes. 8. Once step 7 is finished, kill the script, and turn the robot off.	
Verification Criteria	
The patient (Praveen) was able to retrieve robot's metrics and video feed with the time between the robot's real-time status and the one reflected on the mobile application (latency) not exceeding 2 seconds. The proof of this will be visible in the timestamp difference array for a period of 3 minutes.	

4.9 Test 9: HRI Task Interpretation Accuracy Test

Test 9: HRI Task Interpretation Accuracy Test | Objective	
Assess the accuracy of robot's understanding of a voice command and its interpretation as a task.	
Associated Requirements	M.P.1.1 (M.F.1)
Equipment	Stretch RE1, Tablet
Elements	HRI Subsystem - Speech Recognition and Task Identification
Personnel	Lead: Atharva | Assistant: Praveen
Location	AI Makerspace (Tepper School of Business)
Procedure	
1. Turn the robot on and home it. 2. Activate the speech recognition pipeline. 3. Praveen to play the role of a patient and will speak (in a normal voice) a pre-defined sentence from our test verification script (ground truth). This is a task that can either be pick-and-place, navigating or just a voice reply. 4. The robot acts on the voice command. 5. The robot metric data highlights the task being performed. This is noted as the experimental data. 6. The test verification script compares the task being performed (experimental data) and the desired task in the script that Praveen implied (ground truth). If the two things don't match, a flag is raised. 7. Steps 3-6 are repeated 15 times. 8. Once step 7 is finished, kill the script, and turn the robot off.	
Verification Criteria	
The robot was successfully able to understand the tasks that the patient (Praveen) delegated. Out of 15 attempts, the flags were less than 6, meaning that the robot was successfully able to interpret 10 speech templates as task.	

4.10 Test 10: Manipulation Pick-and-Place Success Rate Test

Test 10: Manipulation Pick-and-Place Success Rate Test | Objective	
Assess the success rate of robot's ability to pick-and-place objects.	
Associated Requirements	M.P.7.1 (M.F.7)
Equipment	Stretch RE1, Tablet, objects for grasping
Elements	Manipulation Subsystem
Personnel	Lead: Shaolin | Assistant: Praveen
Location	AI Makerspace (Tepper School of Business)
Procedure	
1. Turn the robot on and home it. 2. Praveen to play the role of a patient and give a voice command that is a pick-and-place task. 3. The robot acts on the voice command. 4. Shaolin plays the role of emergency rescue and stays on standby near the robot to be able to press the kill switch button in case the robot starts behaving undesirably. 5. Make a note of success/ failure of the pick-and-place task. 6. Repeat steps 2-5 for 10 times. 7. Once step 6 is finished, kill the script, and turn the robot off.	
Verification Criteria	
The robot was able to pick-and-place objects based on Praveen's (patient) input at least 7 times.	

4.11 Test 11: Spring Validation Demo (Experiment)

- **Associated Requirements**: As listed on table 1
- **Equipment**: Stretch RE1, Objects for Manipulation
- **Elements**: Manipulation, Navigation, Human-Robot Interaction
- **Personnel**: Abhinav, Atharva, Praveen, Shaolin, Shivam
- **Location**: AI Makerspace (Tepper School of Business)
- **Environment setup**: We set up the environment in the following way:
    1. Three different objects are placed on a table that is placed in a location that is not near the start location of the robot.
    2. Place obstacles in the path between the object and the user's location.

Table 1: Spring Demo Procedure

Steps	Procedure	Success Criteria	Requirements
1	Pam (senior citizen) asks Alfred to set up a video call with his/her family	Successful command interpretation and video calling	**- M.P.1 (partial)** **- M.N.3**
2	User (Pam's family members) teleoperates the robot using the app in handheld device to pick up an object for Pam.	Communication Latency <5s (visual verification)	**M.P.8**
3	Robot prevents obstacles while moving in teleoperation and performs manipulation.	Avoids 100% of obstacles	**M.P.5**
4	Pam gives speech commands to Alfred to pick up an object.	Correctly interprets speech with communication latency <5s (visual verification)	**M.P.1 (partial)**
5	Robot localizes itself in the premapped environment	Localization with error threshold<25 cms	**M.P.3**
6	Robot interprets the command and plans a global path using predefined heuristics	- Correct interpretation of the task - Plans global path within 2 minutes	**M.P.1.1, M.P.4.1**
7	Robot navigates and reaches the goal location while avoiding static and dynamic obstacles.	- Avoids 100% of obstacles - Navigates at an average speed of 0.4 m/s	**M.P.5, M.P.4.2**

8	Robot detects the object and estimates the grasping location	Successful object detection for chosen object category.	**M.P.6 (partial)**
9	The robot picks up the object.	- Successfully picks up the object within 3 tries. - Time to completion <=8min	**M.P.7.1, M.P.7.2**
10	Robot navigates back to starting location and places the object on the user's table.	- Avoids 100% of obstacles - Navigates at an average speed of 0.4 m/s - Greater than 70% successful placements	**M.P.5, M.P.4.2**
11	Robot metrics and video feed of the robot are displayed on device of the user throughout the mission	Communication Latency <5s (visual verification)	**M.P.9**

A Appendix

Table 2: Mandatory Functional and Performance Requirements

Functional	Performance	Description
M.F.1 Receive commands from the user: preset speech primitives/handheld interface	**M.P.1.1** Interpret 10 speech templates as tasks. **M.P.1.2** Latency for control commands <5s	The robot will primarily operate using speech inputs. This should be seamless for the user.
M.F.2 Perform basic (pre-defined) social engagement with user	**M.P.2** Fallback rate: <20%	Robot provides feedback upon receiving commands, and should be able to automatically execute pre-defined tasks.
M.F.3 Localize itself in the environment	**M.P.3** Average error <25 cms	
M.F.4 Plan and navigate through the pre-mapped environment	**M.P.4.1** Plan global path to desired location within 2 minutes. **M.P.4.2** Navigate at a speed of 0.4 m/s	
M.F.5 Autonomously avoid obstacles in the environment	**M.P.5** Avoid 100% of static obstacles in range	The robot should not collide with any object as it may render the environment unsafe for users.
M.F.6 Detect objects for grasping	**M.P.6** mAP >= 80% for 10 object categories (e.g bottle, remote, medicines etc) under the following conditions: 1) Object is within 1m of body camera 2) Not kept on a white/transparent surface 3) Adequate Indoor lighting conditions	The conditions have been pre-specified keeping in mind that a robust system in a structured environment is more preferred than an unreliable system in a general environment
M.F.7 Manipulate predefined objects to/from planar surfaces at known locations in the environment	**M.P.7.1** Greater than 70% successful picks and places **M.P.7.2** Manipulation tasks should be completed within 8 min.	The objects will be picked up and placed on flat planar surfaces such as tables only. i.e, allowing "the user to grasp and remove the object from the end-effector" is out of scope.
M.F.8 Allow approved operators to teleoperate the robot	**M.P.8** Communication latency <5s	
M.F.9 Provide user with robot metrics and video feed of the robot on a handheld interface	**M.P.9** Communication latency: <2s	

Table 3: Mandatory Non Functional Requirements

Requirements	
M.N.1	Appear non-threatening to the user
M.N.2	Be physically compliant to human interaction/contact
M.N.3	Have a simple UI/UX for the handheld interface
M.N.4	Have a modular software architecture for further development
M.N.5	Allow users to pre-schedule tasks/assistance

Table 4: Desirable Functional Requirements

Functional	Performance
D.F.1 Perform all required computation locally	–
D.F.2 Understand and interpret non-template natural language commands	**D.P.2** Fallback rate $< 20\%$
D.F.3 Detect and alert caregivers if the primary user suffers a fall.	**D.P.3** <1 in 100 false positives

Table 5: Desirable Non Functional Requirements

Requirements	
D.N.1	Appear aesthetic to the user
D.N.2	Reasonable cost for the user