Mobile manipulator for pushing buttons.

Project Description

This project aimed at using a mobile manipulator to push buttons and toggle switches instead of humans. The robot will be interact with targets (buttons and swtiches) in a semi-autonomous fashion. Robot will provide user with an visual representation of the surrounding, user will select the location of the the buttons and switchs to operate, along with necessary instruction on how to operate it, and robot will complete the rest of the operations.

The insperation of the project:

There are planty of lights and buttons being pushed everyday. Some of them, like light switches, have connected alternative that's controable through software, however, majorities of them are not trivel to be converted to connected version. For example, the start button of a washing machine. In order to automate more everyday appliances without needing a "smart" version of all of them, a robot that can push button will bring great convenience to everyday life.

What this project is not:

This project is not to design a complete human machine interface for teleoperating swtiching with minimal human input. Thus the final project might have a clumzy interface for operation.

This project is also not a "pure teleoperate" project. User are not expected to directly control any joints of the robot, but rather, give high level instruction on end-effector motions.

Technical objectives

Fallback goal

Targets are choosen with specific type for easy manipulation and are placed at easy to reach locations. For example some light switches firmly moutned on the side of tables to meet the reach-envolope for the robot.

Navigation environment will be greatly simplified with complex geometries coverted off with rigid surfaces. More of a low poly 3D cubed maze.

core goal

- Robot able to reach and operate existing light swtiches and buttons installed for human interaction.
- Targets could be marked with additional visual tags to help locate them and simplify human interface design (as that's not part of the project's goal).

- Manipulator have a set of pre-defined motions for different buttons and swtiches that operator can choose from.

Reach goal

- Manipulator able to push some buttons on not perminently fixed panels (like a keyboard on table)
- End effector force of the manipulator is tracked over time. The tack-tile feedback in most buttons and swtiches will be captured and detected.
- Robot able to identify and locate certain kinds of switches (pre-defined) visually, which allow operator to command operating certain switch in one click.

Learning objectives

From core goal * Mobile manipulation taks and motion planning in 3d space (with octomap or similar) * (Potential) Manipulator mounting on mobile platform, specifically z axis elevator for increasing reach of short robots

From reach goal * Force feedback from task * CV object detection with machine learning (YOLO)

Tasks

1. Selecting and building of the mobile manipulator.
2. 3D slam using a combination of sensors. Contructing and using octomap
3. Motion planning the mobile base to place target in reach of the arm with high manipulability.
4. Motion planning of robot arm to manipulate switches and buttons
5. Basic interface for operator to see surranding of robot. Mark position, type and oriantation of a switch and get visual feedback to confirm the selection.
6. End effector force reading and detect the ticktile feedback
7. CV for auto detecting targets

Risks

Hardware selection and custom hardware design.

Selecting a "best fit" mobile manipulator might result in some challange. Some existing options might be too small/short, while others might be too big to fit in thigher environments. If custom modification were to be made, it might take signifigent amount of time to debug the hardware (like adding a actuator on Z axis).

Could be solved by having backup robot options that's fully integrated and ready to use. Time box the hardware custimization and drop back to fallback option early on.

Having good hardware abstraction would also help the potentional of swtiching hardware if certain route are proven to be not possible within the time-constrain.

SLAM with 3d mapping

This is something completely new to me. Planty of time need to be spent in learning the tools, reading publications, and expermenting on actual hardware.

The plan is to use complete solution packages as much as possible.

Obstical avoidance during motion

Since the project target at human enviornment, which is very complex, it might be difficult to generate a clear 3d mapping and avoid all obstical during motion.

Could try to simplify the environment by removing objects or cover them with sensor friendly materials.

HMI design

Designing of UI could be a huge time-sink.

Push this task to later and timebox it, and it's less of a critical part of the system.

Tools and References

Hardware

Mobile manipulator:

The manipulator need to be able to reach light switches height (125mm) while reaching outside the base-radius and still have some manipulability (would with the help of extended end effector)

- Option 1: Ridgeback + Sawyer arm. But this is really huge. might makes it more difficult to motion plan the arm for collision free motion
- Option 2: Jackal Robot + Widow 200X robot arm (with girpper swapped out for extra dof)
  - This option require an additional moutning design to reach enough height, and/or a custom built z-lift to make up for arm's reach

Force sensor for each goal

Decently powerful computer with cabilities of live 3d SLAM onboard of the mobile base. Likely require a recent GPU if octomap-rt were to be used.

Sensors:

- 3d Lidar or depth camera

Reference materials

- octomap and octomap RT for 3d mapping
  - https://github.com/OctoMap/octomap
  - http://graphics.ewha.ac.kr/OctoMap-RT/