Ollin Boer Bohan ollin@cs.uw.edu

571A Project Proposal

Research Problem & Proposed Approach

We propose to implement a person-following + (optional) obstacle avoidance system on the MIT RACECAR robot platform.

To be more precise, the robot should be able to:

1. Determine locations of nearby people (this can be done monocularly with proper camera calibration, bounding-box detection via YOLO or SSD, and projection using a flat-floor assumption)
2. Plan navigation goals to move the robot closer to those people, stopping at a safe margin
3. Execute those plans to smoothly follow the nearest person.
4. (Optionally) avoid collisions with nearby obstacles. This can be implemented using free-space detection and projection as with the person-detection approach above, along with modification to the planning code. Or, if the RACECAR has a LIDAR sensor, just use that.

Evaluation Component

We will evaluate our system by measuring:

1. How long it can follow a person in an indoor environment without losing track of them
2. How long it can follow a person in an indoor environment containing obstacles, without getting stuck and requiring manual recovery

Prior Work

- https://www.researchgate.net/publication/326019139_HUMAN_FOLLOWING_ON_ROS_FRAMEWORK_A_MOBILE_ROBOT
- https://medium.com/@waleedmansoor/make-human-following-robot-using-realsense-camera-3a67b29921fd
- https://www.semanticscholar.org/paper/Integrating-Stereo-Vision-with-a-CNN-Tracker-for-a-Chen-Sahdev/2067a5a4e8851bf8083285a33f542229792f1826
- https://github.com/pusnik/robot-human-follower

Milestones & Timeline

- **Week 3:** Get software set up on the robot and drive-by-wire working (using smoothed random steering inputs), and make sure we can access the sensors from ROS.

- **Week 4:** Get a person-detector (e.g. `https://github.com/philipperemy/yolo-9000`) running on the robot, at some sort of reasonable frame rate. We may be able to do this using the RealSense camera.
- **Week 4:** Implement screen-space reactive control (changing steering angle to center the largest person-box, and moving forward/backward to move the person-box to a target size)
- **Week 5:** Perform camera calibration and get 3d projection to a ground-plane working. Or, if the RACECAR has a LIDAR or other non-camera sensor we can use, figure out how to connect to that.
- **Week 5:** Switch to 3d-projected reactive control.
- **Week 6:** Get a simple free-space detection algorithm working (e.g. texture-similar flood fill, as in the Stanford DARPA bot), or obstacle avoidance using the LIDAR.
- **Week 6:** Improve reactive control to target free-space (still using greedy path-finding)
- **Week 7+:** Improve reliability / smoothness and record demo / prepare report.