

571A Project Proposal

Nansong Yi, Thomas Wang, Ollin Boer Bohan

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 in vision only using free-space detection and projection as with the person-detection approach above, but will likely be supplemented or replaced by use of the depth camera and laser scanner (depending on what is most empirically reliable).

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

Human-following is a fun application of modern tracking algorithms to a mobile robot platform, and, as such, there have been several implementations of this concept on various platforms. For the most part, however, these implementations have had minimal obstacle avoidance capability and have used classical computer vision approaches rather than CNNs for the tracking component.

- 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://github.com/pusnik/robot-human-follower>

The following approach does use CNNs (online, rather than pre-trained) for person tracking, but is closed-source:

- <https://www.semanticscholar.org/paper/Integrating-Stereo-Vision-with-a-CNN-Tracker-for-a-Chen-Sahdev/2067a5a4e8851bf8083285a33f54229792f1826>

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's ROS API as well. We may need to modify the robot slightly (e.g. changing the camera angle) in order to make tracking more accurate.
- **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). **Milestone:** At this point we should be able to record a video of person-following under ideal environmental conditions.
- **Week 5:** Perform camera calibration and begin incorporating 3d information into the reactive controller. We may also test a running SLAM pipeline like RTAB-Map or ORB-SLAM2.
- **Week 5:** Switch over entirely to 3d reactive control, which will (hopefully) improve reliability and accuracy of tracking. **Milestone:** at this point we should be able to record a video of smoother and more accurate person-following under ideal environmental conditions.
- **Week 6:** Get a simple reactive obstacle-avoidance mechanism working (e.g. texture-similarity-based flood fill, as in the Stanford DARPA entry, or simple rules on top of the depth-camera input to avoid driving into obstacles, e.g. targeting the on-screen region whose nearest point is furthest away) **Milestone:** at this point, we should be able to record a video of indoor person-following amidst basic obstacles.
- **Week 6:** Test further improvements (switching to full path-planning in a 2D occupancy grid, cross-frame tracking to predict the human's movement despite occlusions, etc.)
- **Week 7+:** Improve reliability / smoothness, record final demo, and prepare report.