OzoBot tracking & map reconstruction

Antonín Jareš

Goals

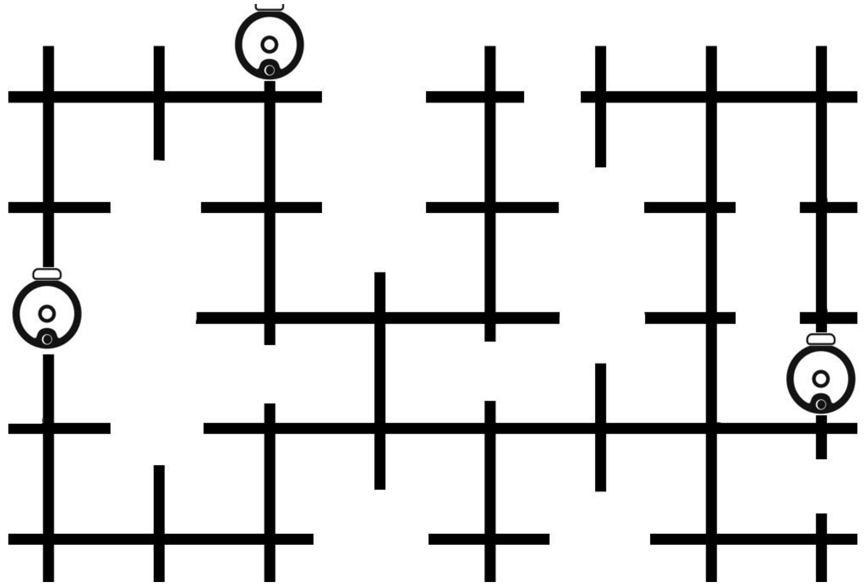
- Reconstruct map from provided frames
    - Webcam (potentially realtime), footage from file
- Track movement of ozobots on the map
- Track collisions/other parameters
- Possibility to save the whole video to a file/replay
- Ideally make custom user evaluation possible

Steps needed

1) Load the map and parse it to the system
2) Classify robots in frame 0
3) Process video stream
   a) Track the robots in each iteration
4) Evaluate changes dynamically, e.g. track collisions
5) Visualize the map & robot behaviours
   a) Play the reconstructed video back
   b) E.g. https://robot-simulator.herokuapp.com

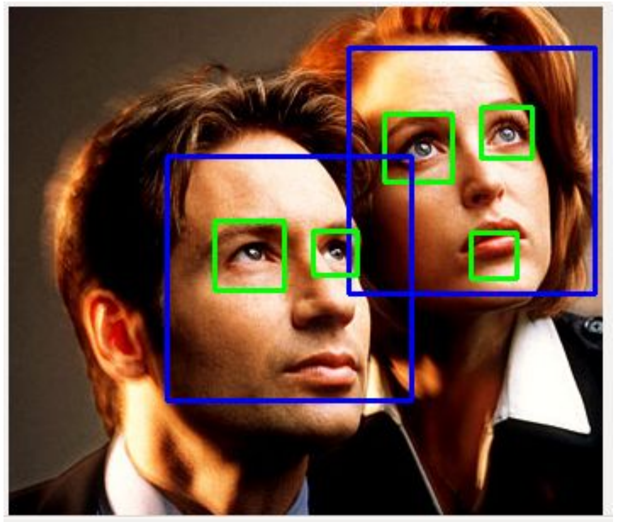
Map loading and visualization

- Web based - React + HTML5 Canvas
- Ease of implementation, easy to use, available on virtually any device
    - Possibility to go back and forth in the video with ease



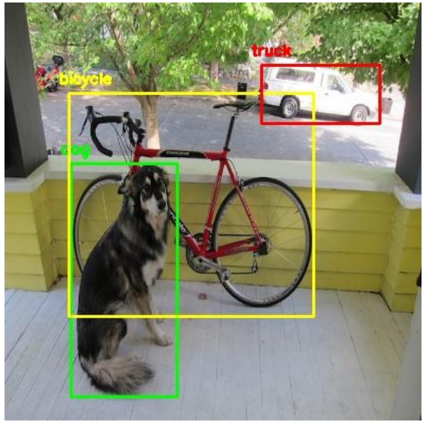
Ozobot tracking

- OpenCV
  - Established library with extensive documentation
  - Basic features out of the box
  - Multiple tracking algorithms - BOOSTING, MIL, TLD, MEDIANFLOW, MOSSE, GOTURN



← face detection on ~ 10 lines of code

Object labeling →



Tracking =/= repetitive detection!

- Repeating detection in each step might fail and is slower than object tracking
- Tracking utilizes prior information regarding the object, its speed etc
- Tracking preserves identity

Tracking preserves identity

Video parsing steps

1) Process stream on the server
2) In frame 0 define bounding boxes for ozobots using classification methods
3) Use these boxes and utilize OpenCV to track ozobots
4) Process changes between states, save new state and write down anything important
    a) Some changes have to be taken into account over a larger timespan
5) Reconstruct video from the information gathered (ideally in real time) & animate
    a) Example https://robot-simulator.herokuapp.com