



LUND  
UNIVERSITY

# Functional Safety of Autonomous Driving Robot

---

AN ANALYSIS OF SAFETY IN SIMULATION VS REALITY  
DONE BY: AHMED ABDALLA SUPERVISOR: MOMINA RIZWAN



# INTRODUCTION

Functional safety is part of the overall safety of a system or equipment that depends on automatic protection



# INTRODUCTION

Functional safety is part of the overall safety of a system or equipment that depends on automatic protection

A system is considered to be functionally safe if it operates correctly in response to its inputs and if it can't, then fail in a predictable way.



- Autonomous function being executed
- Sensors record some values that indicate a safety hazard
- We take actions to ensure safety

# Example of Autonomous functions

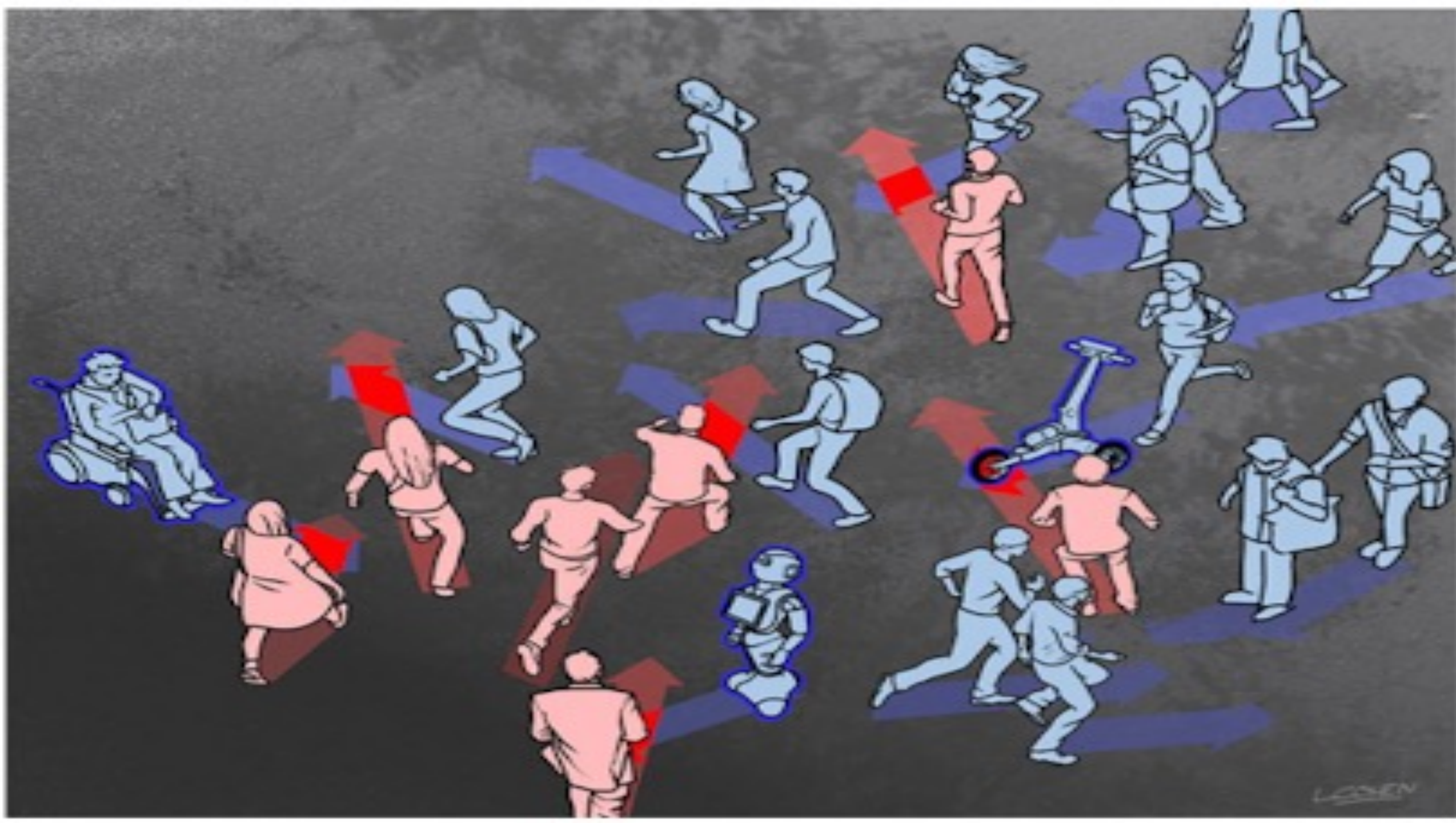
- -Navigation: Mobile robot moving back and forth between start and goal
- Pick and place

# Safety Hazards

- Physical Contact between
  - Robot and Human R2H
  - Human and Robot H2R (initiated by person)
  - Cooperatively-initiated contact (Exchange of object)
- These could be intentional or unintentional (Accidents)
  - Dynamic
  - Static

# Hazards Types

- Serious damage to Human or Robot
  - Crushing
  - Collision
  - Pushing
  - Drag
  - Touch Contact
  - Swipe





# Problem Definition

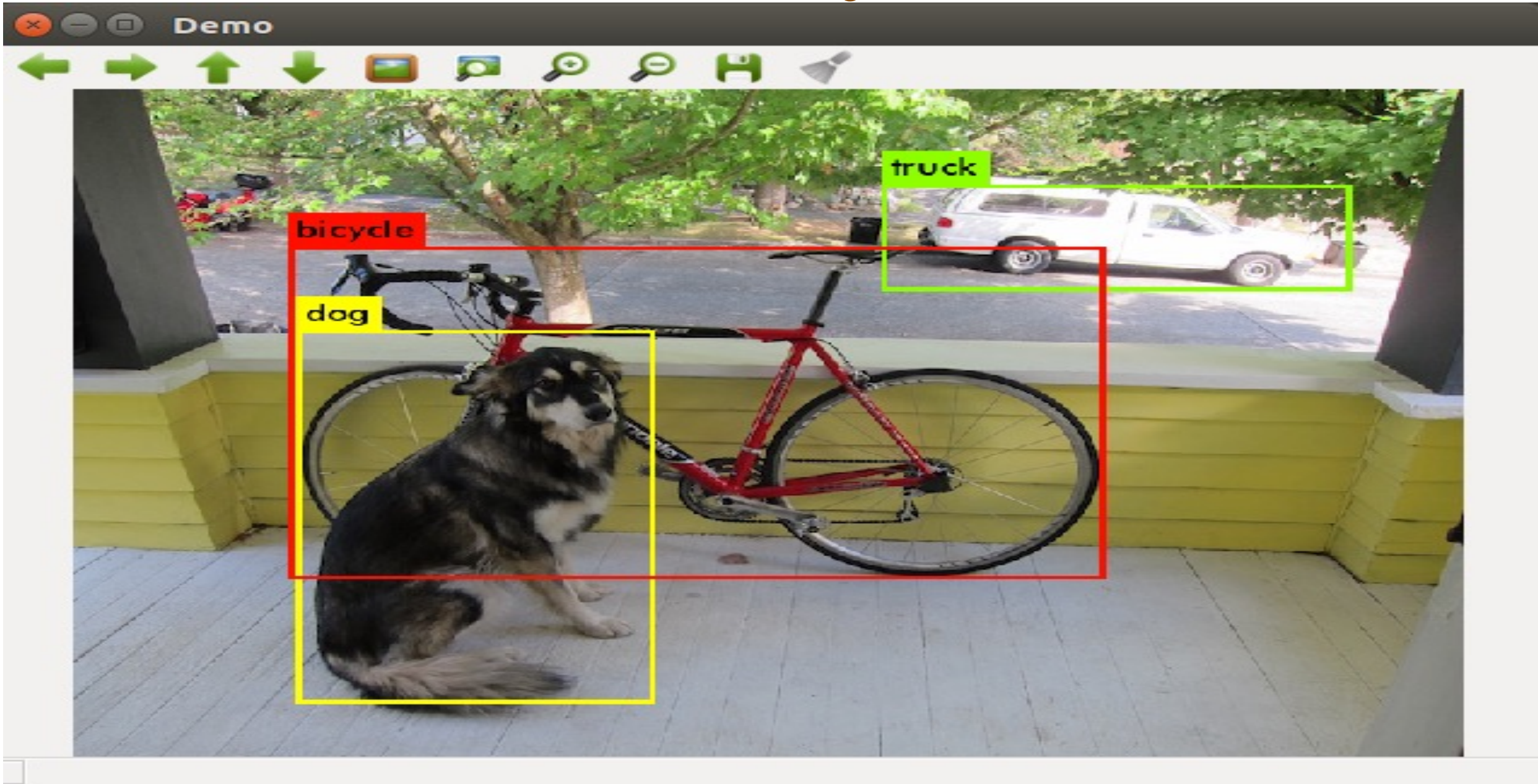
- A robot is carrying out some autonomous function and an unexpected safety hazard is detected. That hazard can be due to a dynamic or static object (person, cat, table,...) raises a potential safety hazard.
- How can we detect this particular hazard?
- How can we assess it's danger level?
- How do we react appropriately to said hazards?

# Solution Overview

- A robot is carrying out some autonomous function and an unexpected safety hazard is detected. That hazard can be due to a dynamic or static object (person, cat, table,...) raises a potential safety hazard.
  - Robot is Navigating from start to Goal as shown in Simulation
- How can we detect this particular hazard?
  - We use Ros and YOLO: You Only Look Once (ros\_yolo package)
- How can we assess it's danger level?
  - How close is the person or object to the robot? Using Bounding Box size.
- How do we react appropriately to said hazards?
  - Static Object: Table
  - Dynamic Object: Person

# ROS for programming

# YOLO ROS: Real-Time Object Detection for ROS



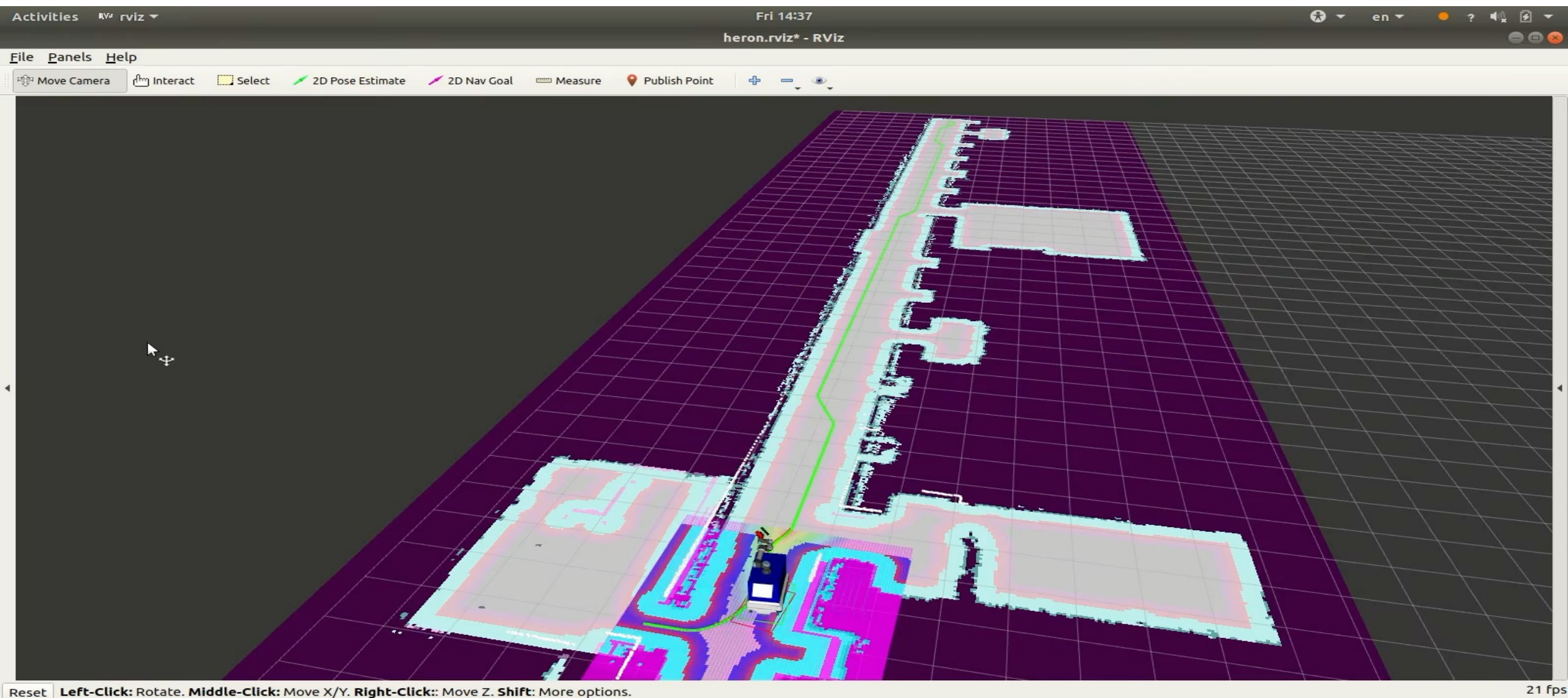
# Why ROS?

- ROS is a powerful open-source framework for robot programming that makes reusing packages, sensor drivers and more easily available.
- We use the `ros_yolo` package because of it performs online object detection using a trained neural network
- ROS allows us to create nodes that subscribe to certain topics and publish commands according to the safety criteria we set.

# Results

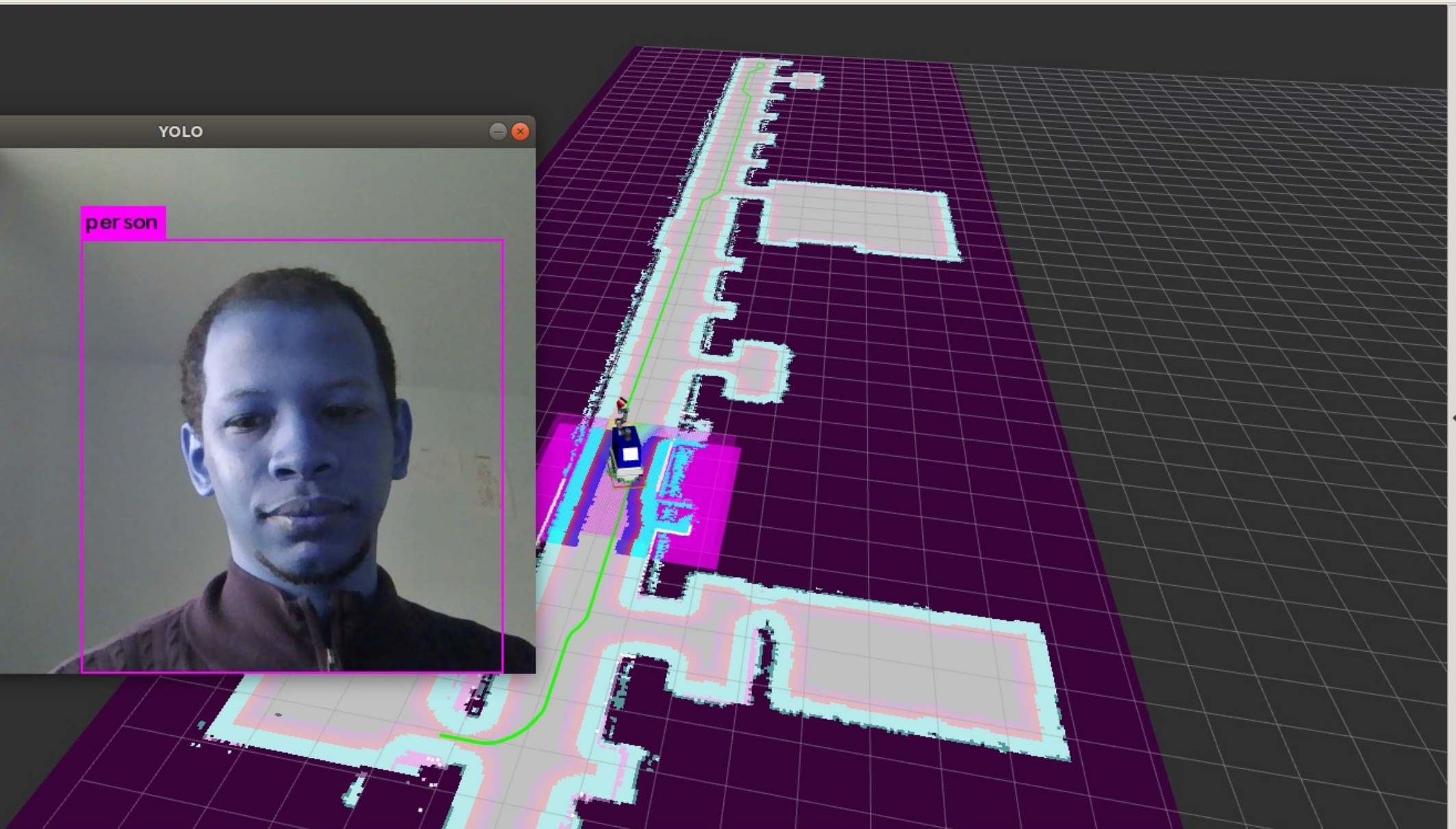
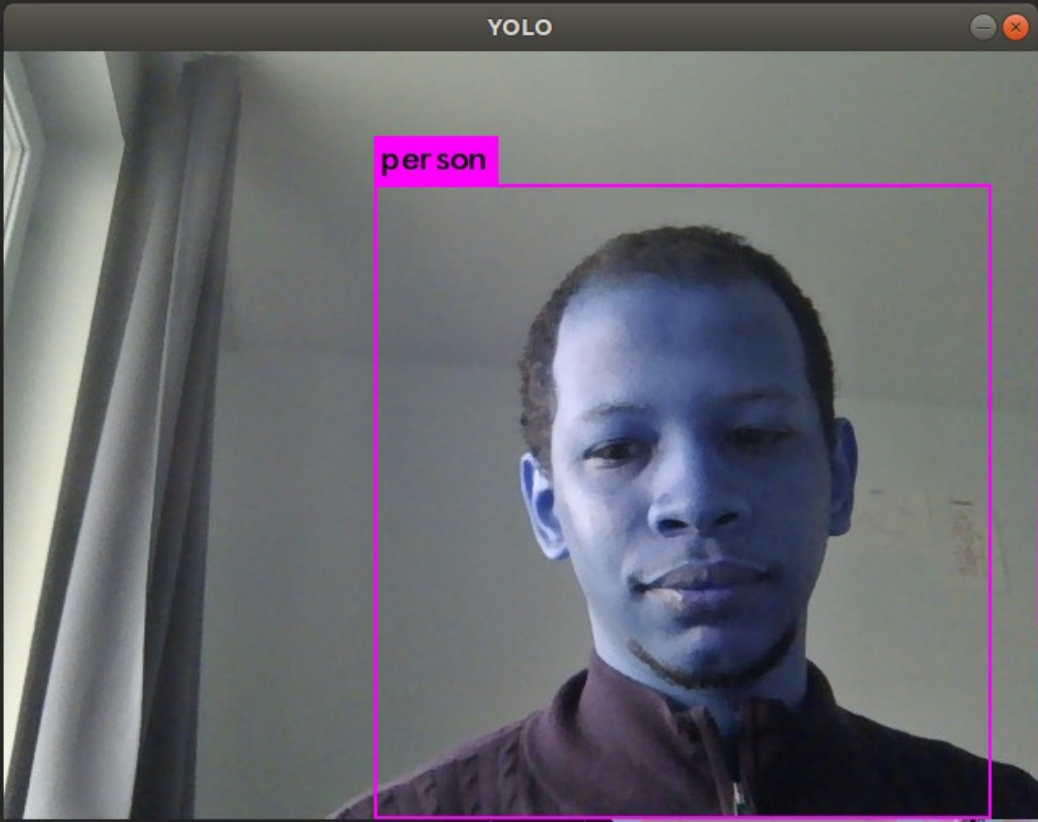
# Results

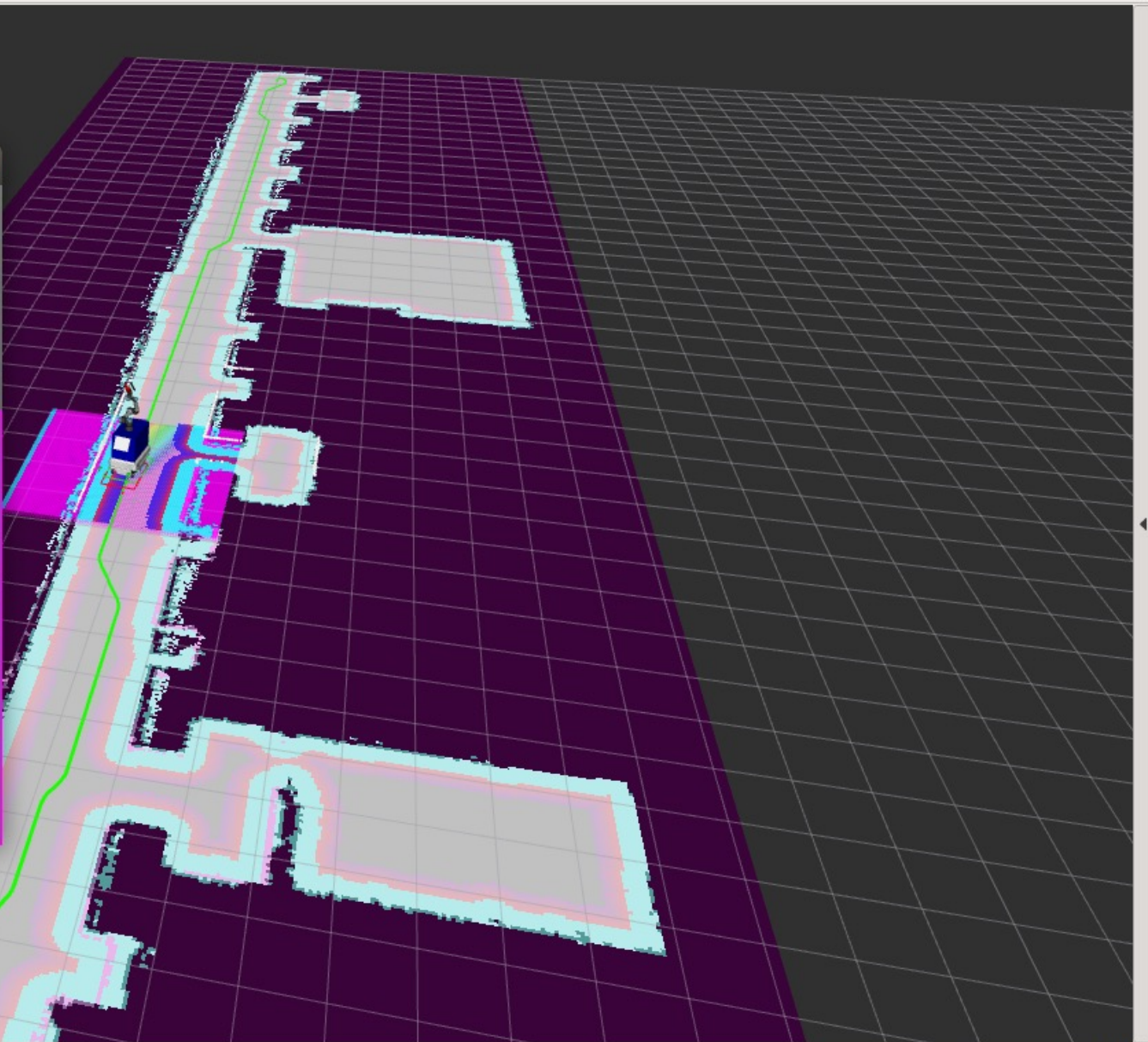
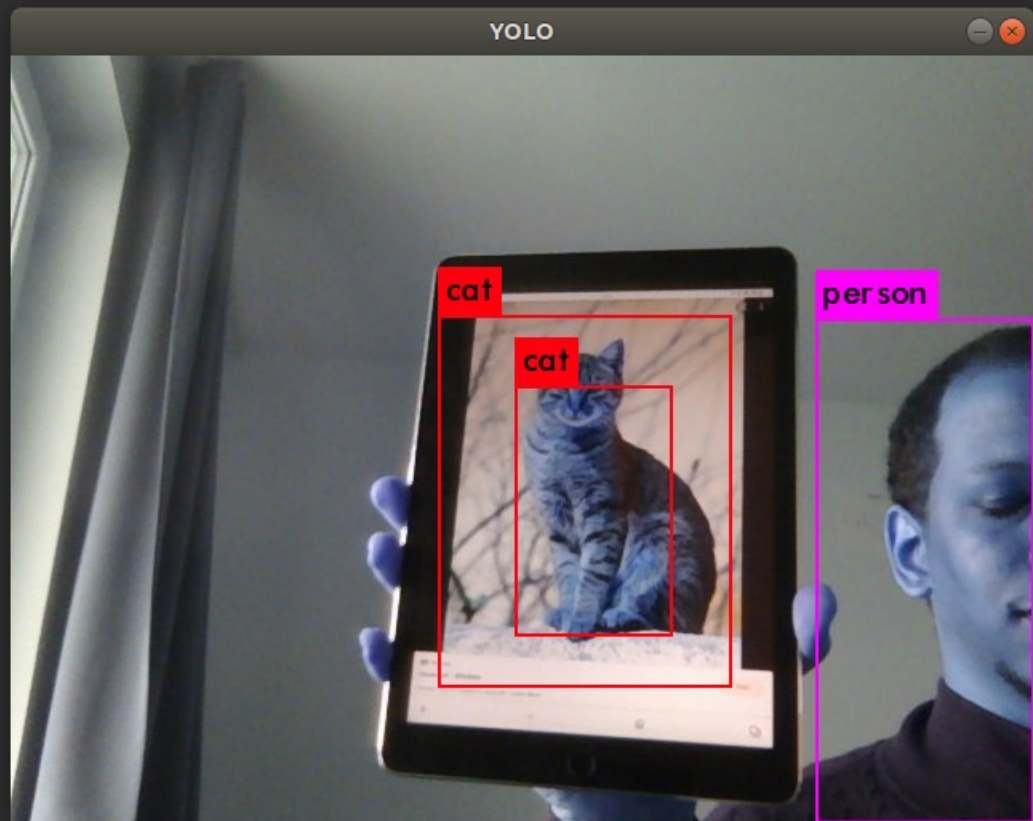
## Navigating to goal

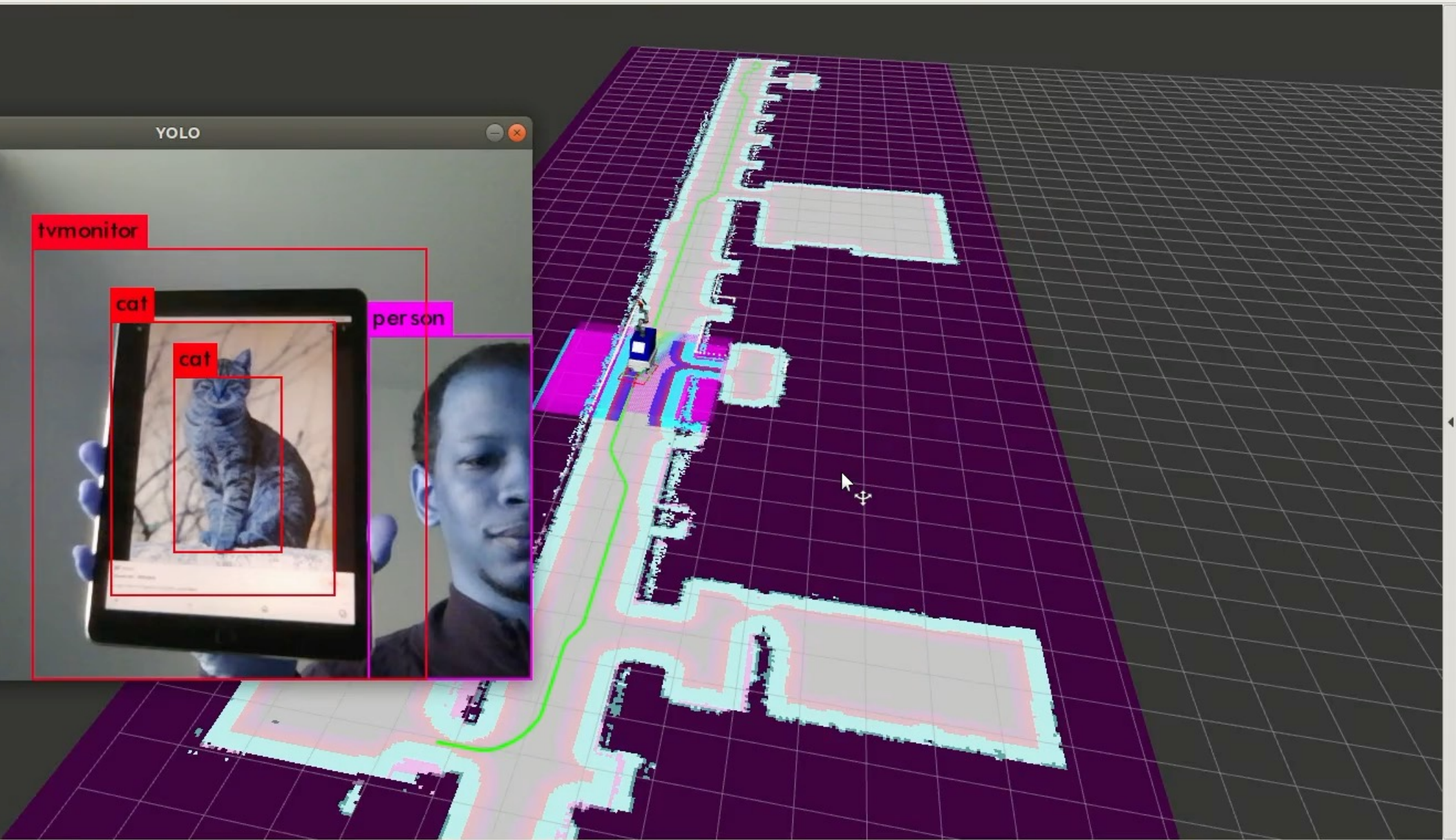
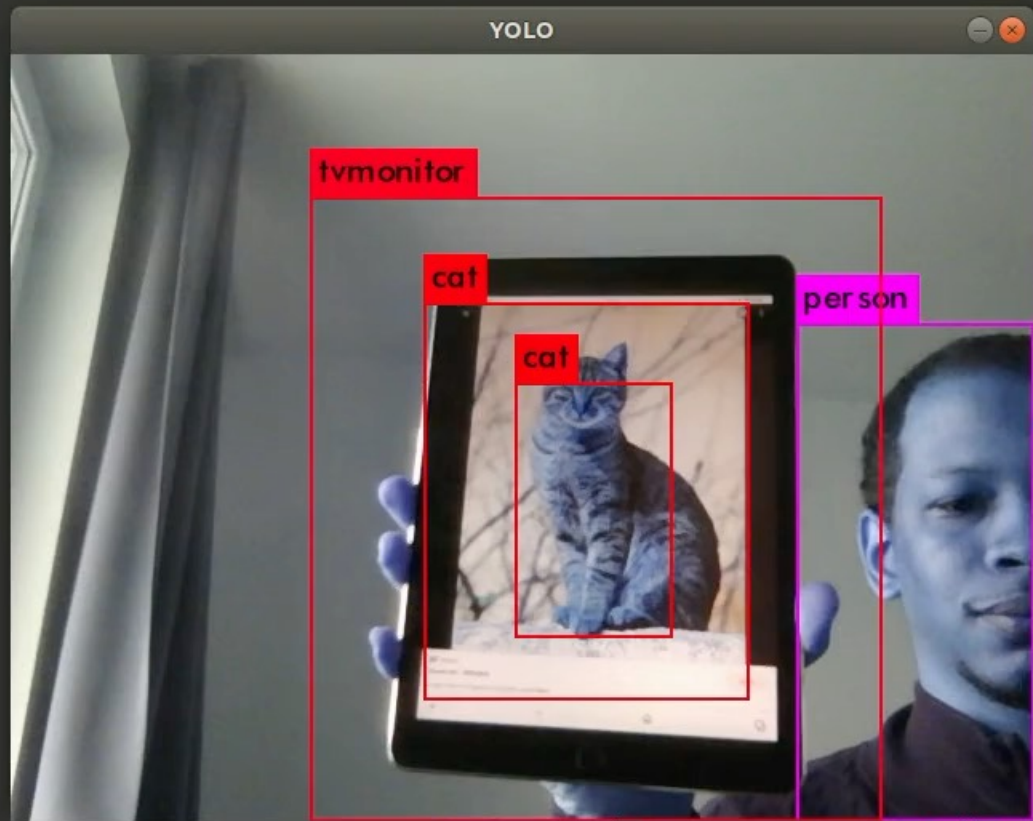


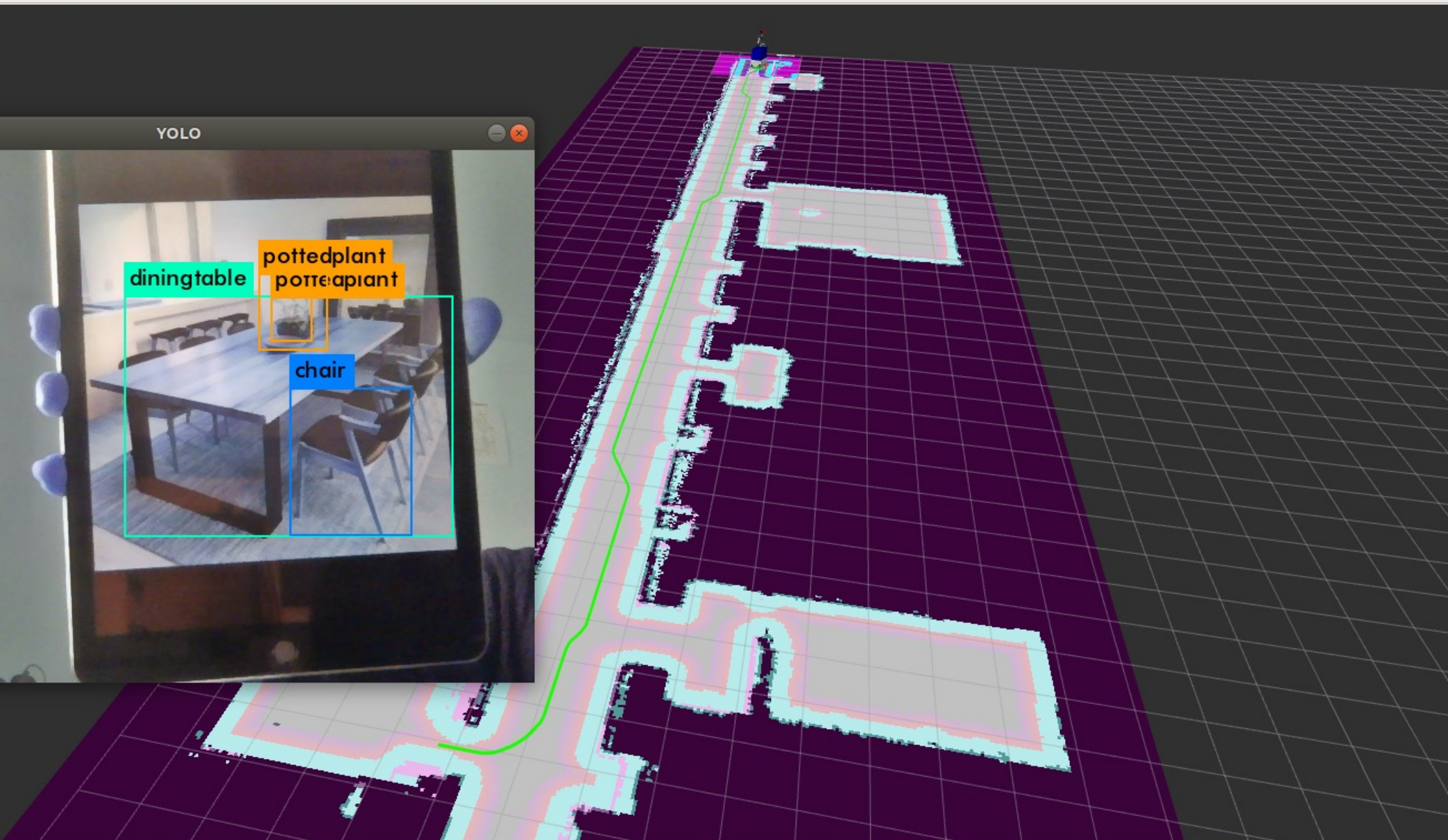
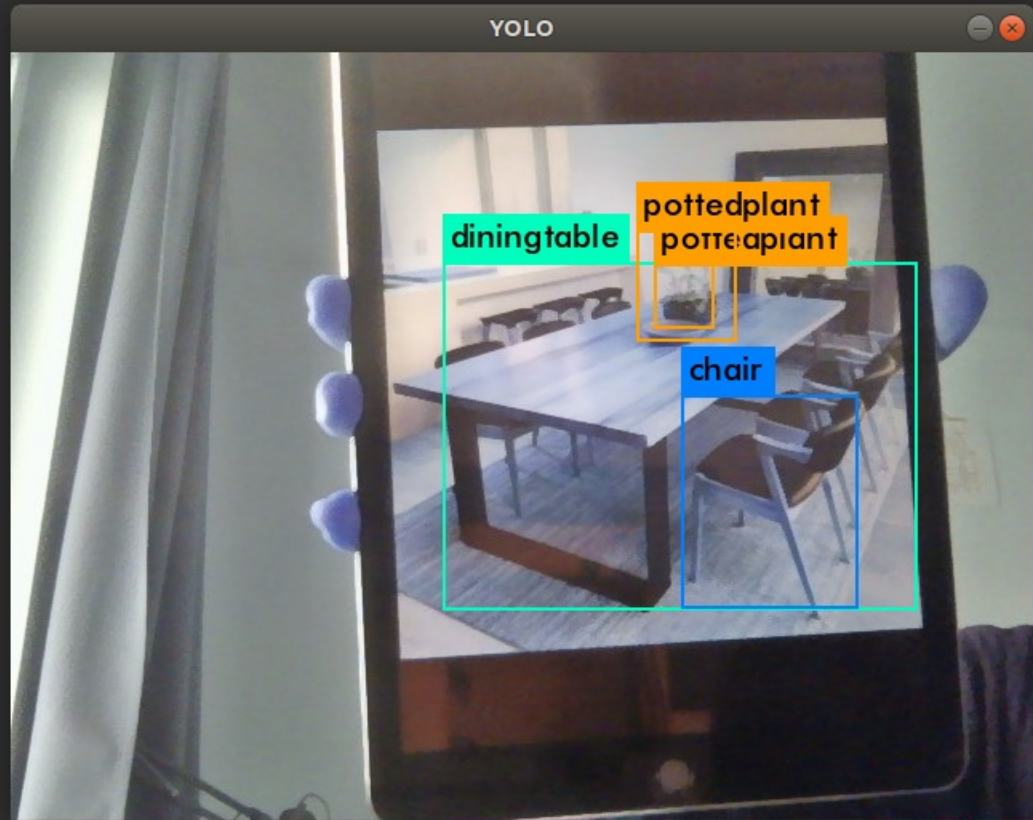
When Detection works well











Reactions to detected Objects  
Is done through python ROS nodes

```
person
25230
0.08212890625
person
35224
0.114661458333
person
25029
0.081474609375
person
24786
0.08068359375
chair
14630
0.0476236979167
bowl
1073
0.00349283854167
bottle
459
0.001494140625
bowl
680
0.00221354166667
person
61440
0.2
person
129019
0.419983723958
person
145376
0.473229166667
person
141246
0.45978515625
person
124800
0.40625
person
130560
0.425
person
131520
```

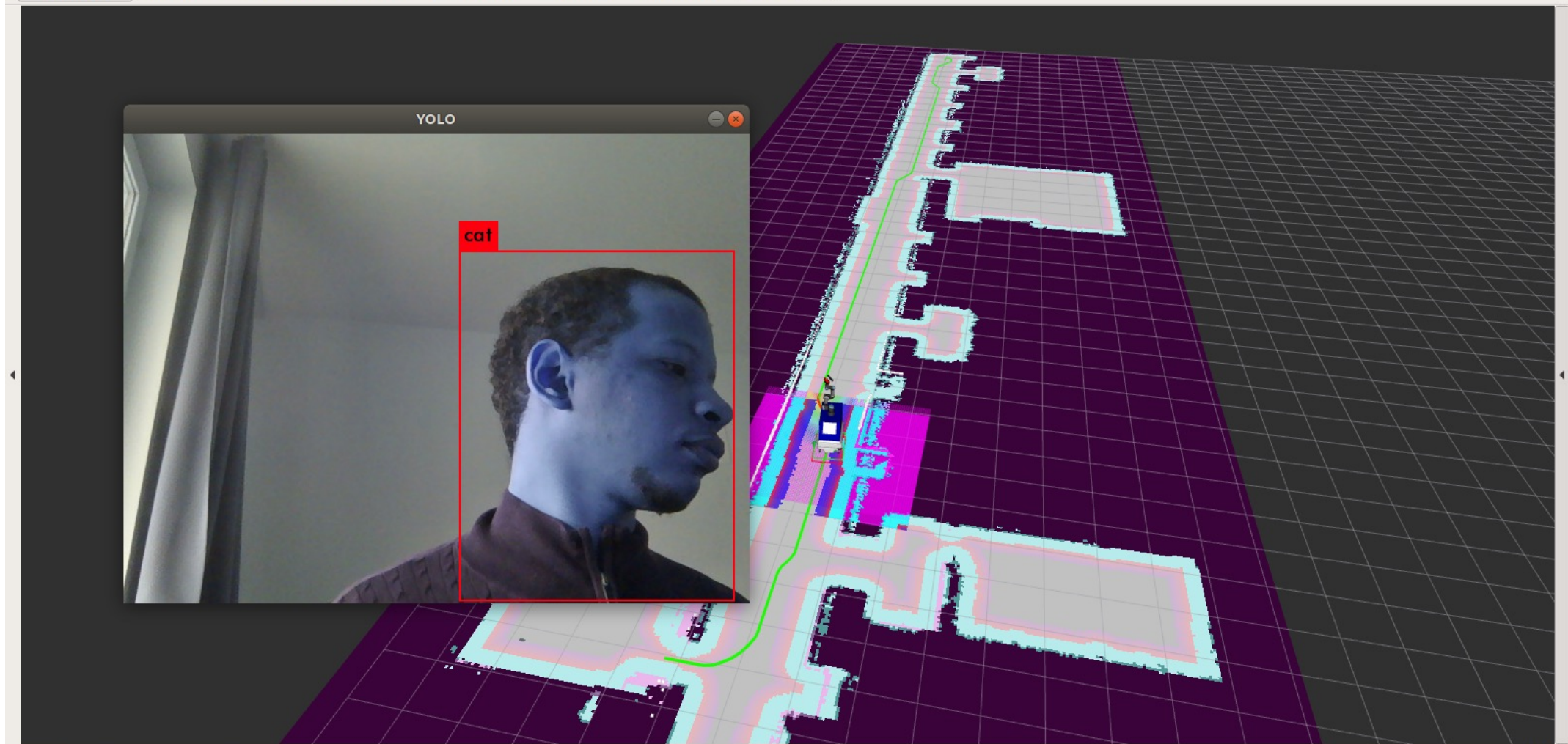
## Reactions to detected Objects

Is done through python ROS nodes

- One node detects the object and its "closeness" by size of the bounding box
- We stop the robot's navigation once a person is detected by launching node that publishes to the a Twist message on the /cmd\_vel topic

When Detection doesn't work well





Reset Left-Click: Rotate. Middle-Click: Move X/Y. Right-Click: Move Z. Shift: More options.

14 fps

Thank you

## Future Work

- Have more reactions to scenarios ( objects, and safety hazards)
- Implement on the Robot
- Train model on more objects



LUND  
UNIVERSITY