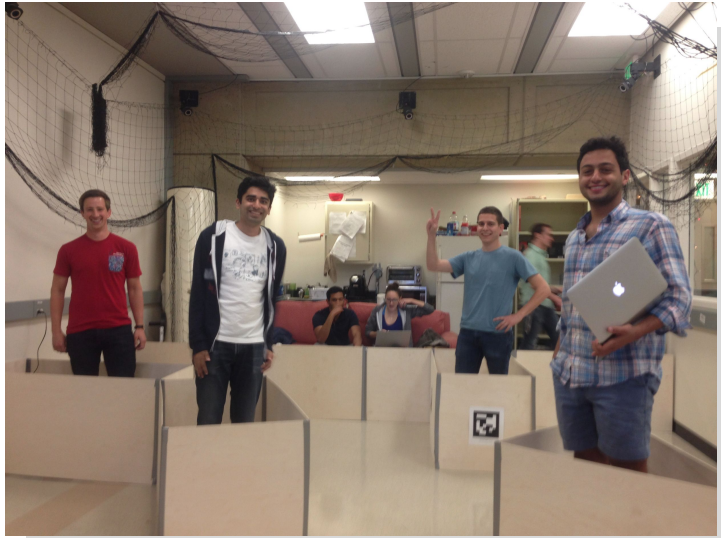
# AA274A Final Project

Demo Day: Friday, Nov 20th

## 2017: Exploration on Mars



Explore and excavate points of interests in a particular order.

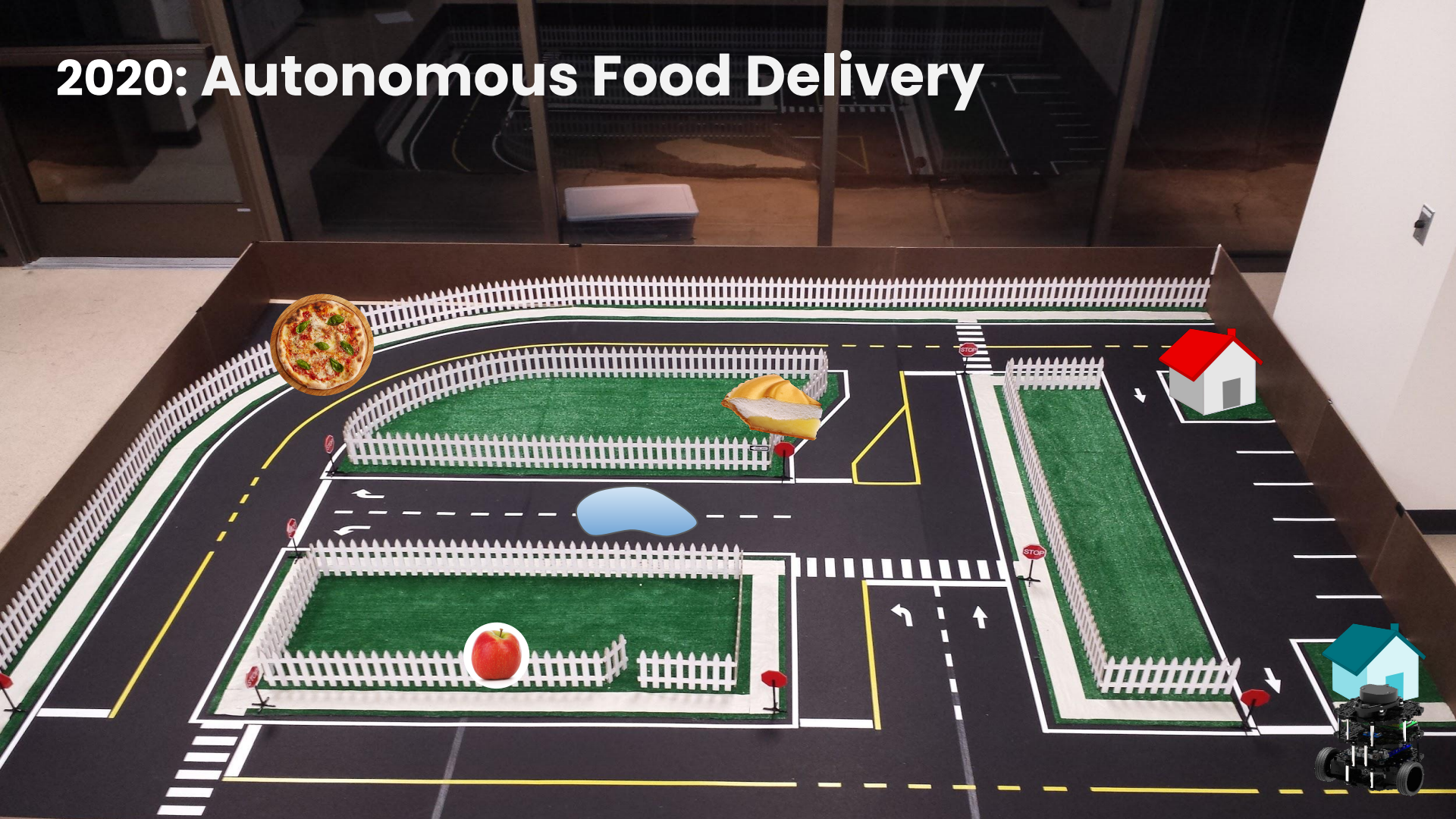
## 2018: Animal Rescue

Finding lost animals in a city, and rescuing as many of them as possible.

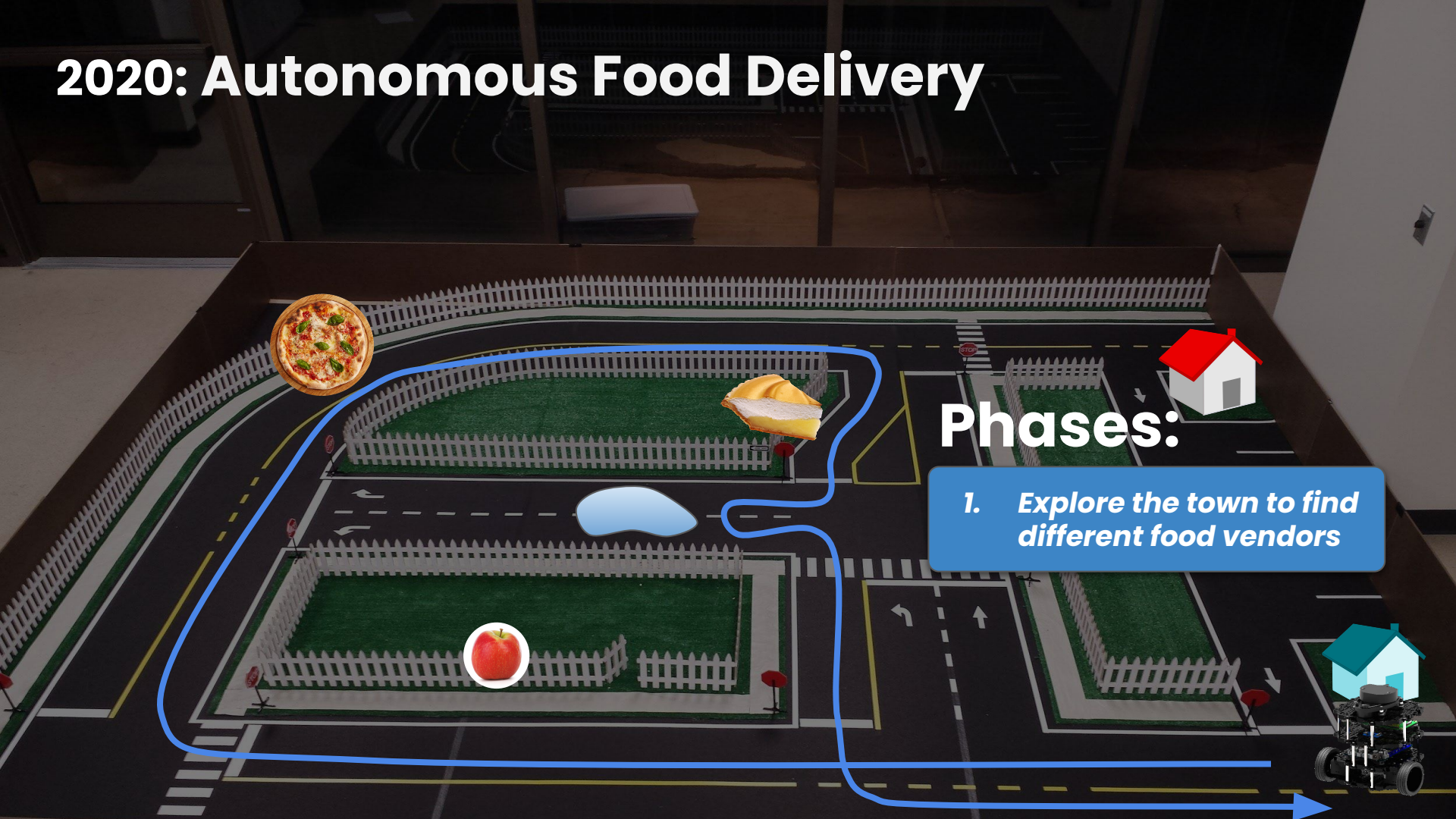
## 2019 and 2020: Autonomous Food Delivery

# 2020: Autonomous Food Delivery

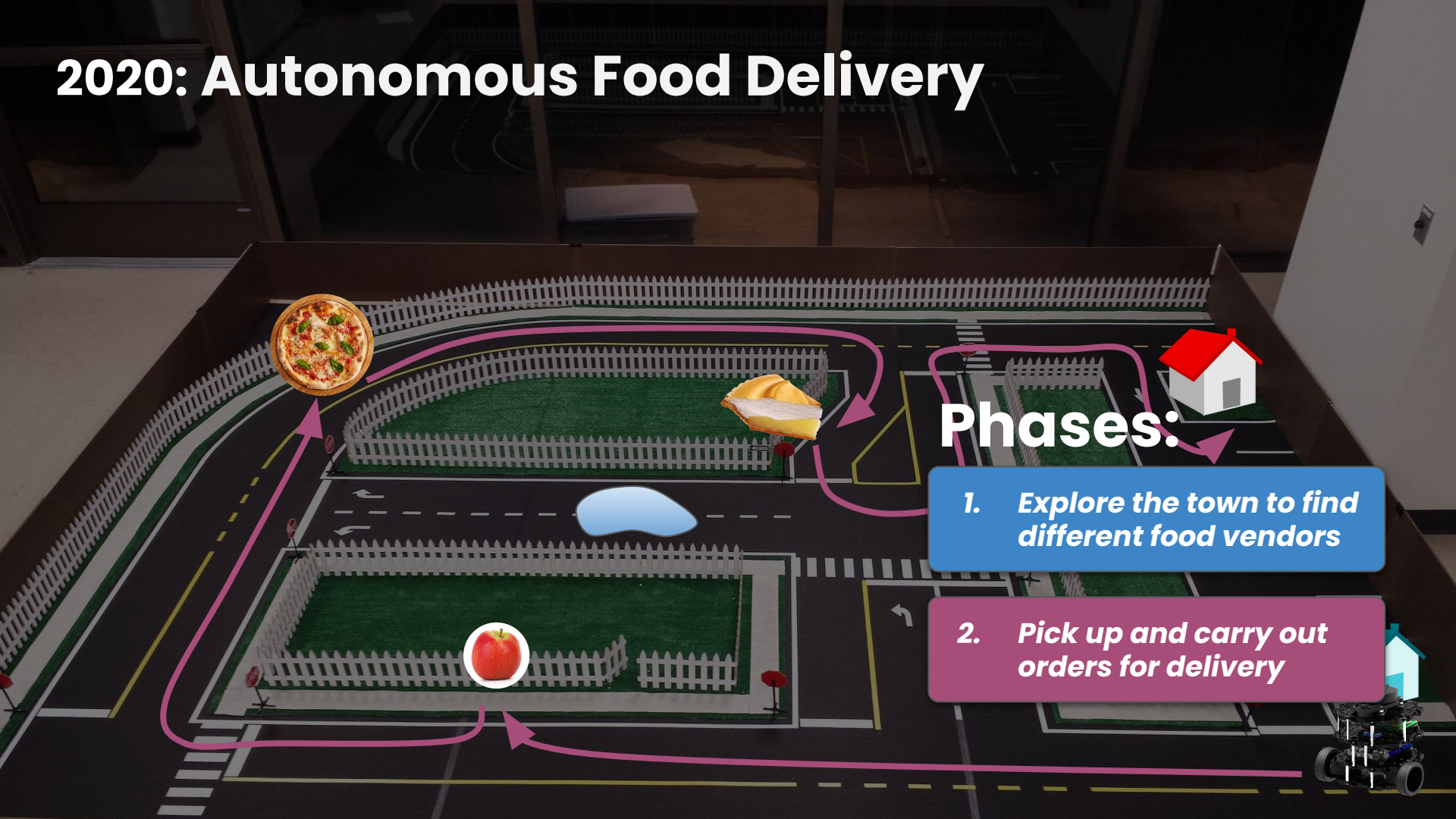
# 2020: Autonomous Food Delivery

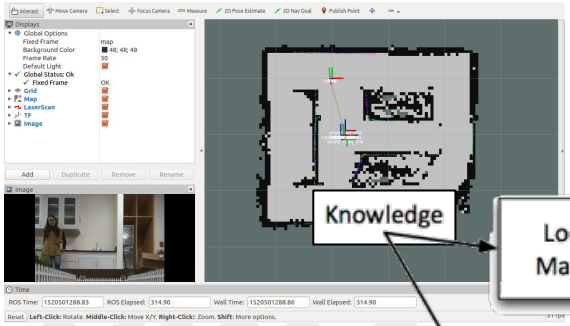


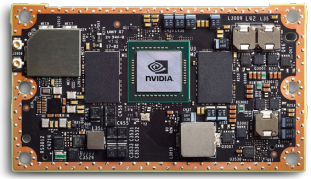
# 2020: Autonomous Food Delivery

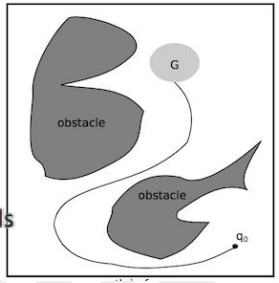


# 2020: Autonomous Food Delivery








Localization
Map Building

position
global map

Decision making
Motion planning

Mission goals

environmental model
local map

trajectory

Information
extraction

raw data

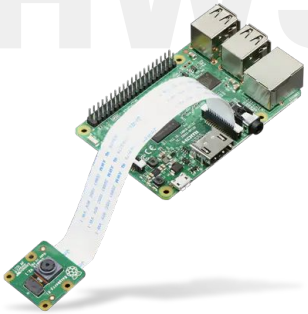
Sensing

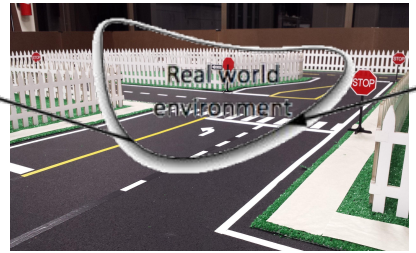
See-think-act

Trajectory
execution

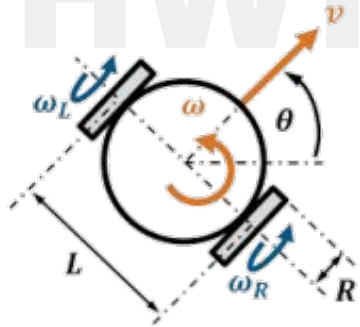
actuator
commands

Actuation




Mock environment




HW4

HW3

HW2

HW1

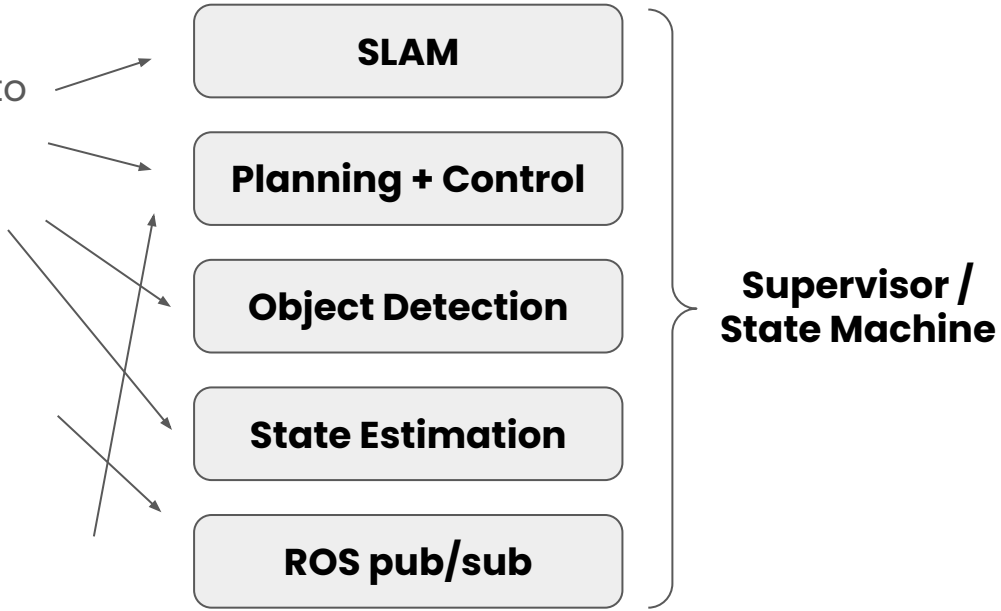
# Mission Outline

## Explore

- Navigate the city without colliding into obstacles.
- Record locations of food vendors.

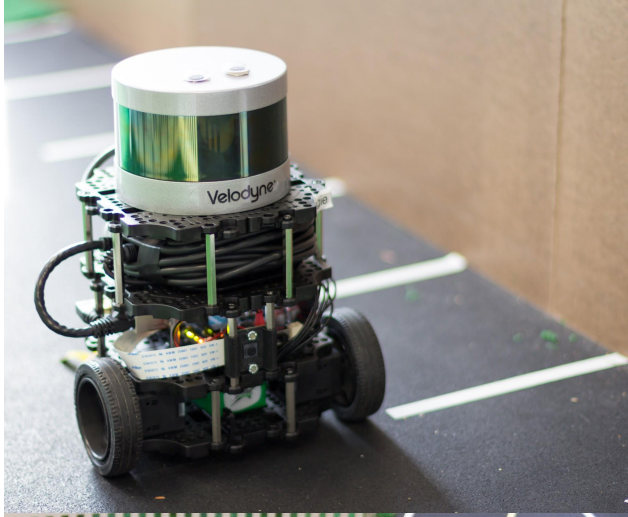
## Deliver

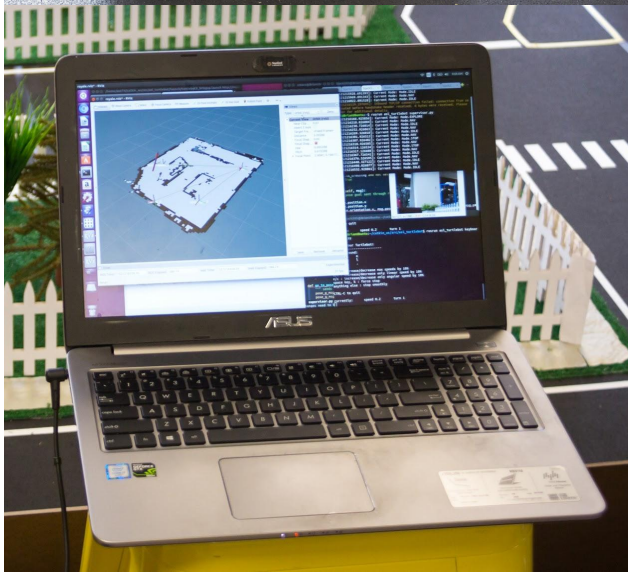
- Receive and process requests for delivery.
- Autonomously drive through the city to pick up desired items and deliver them to the goal.



# Deliverables

1. Software Demo on Genbu
2. “Command Center” Visualization Tool in RViz
3. Three Minute Pitch Presentation



# 1. Software Demo

**Baseline:**

Phase 1: Exploration

- Human operated waypoint following for exploration.
- Detecting and Logging food locations through TF frames / broadcasting on a topic.

Phase 2: Delivery

- Autonomous navigation to revisit vendors and drive to delivery location without running into fences. Stop at a vendor location for 3-5 s to “pick it up”.

The baseline implementation is **not** sufficient for full credit. A baseline implementation will result in 70% of total credit. Each extension will net roughly 10% or more in additional credit.

**Possible Extensions:**

- Autonomous exploration.
- Accurate vendor pose estimation via filtering.
- Shortest path / user-defined order for picking up deliveries.
- Stopping at stop signs
- ~~Detecting and avoiding driving over puddles.~~(Not recommended)
- Collision avoidance
- Detecting a cat/dog and doing something fun like publishing “meow”!
- Calibration of velodyne/camera offsets.
- **Your own ideas!**

# 1. Software Demo

**Breaking down the baseline:**

Phase 1: Exploration

- Human operated waypoint following for exploration.
  - Use SLAM node gmapping (no need for EKF you wrote) for robot pose and occupancy grid.
  - Navigate using rviz clicks or hard-coded waypoints. This will use A* for path planning, controllers for tracking, pose controller, etc. that already exist in navigator.py.
  - Objective is for robot to discover all the walls and explore the map fully, then return to starting point.
- Detecting and Logging food locations through TF frames / broadcasting on a topic.
  - While robot is exploring, have the detector running. Use CNN detector with use_tf=True.
  - When an object is detected, keep track of its position to revisit it later.

Phase 2: Delivery

- Autonomous navigation to revisit vendors and drive to delivery location without running into fences.
  - TAs will ask which objects you've detected and ask you to revisit 2-3 of those objects.
  - Robot must autonomously navigate back to the objects, in any order. Stop at a vendor location for 3-5 s to "pick it up". No gripper needed.
  - Return to the starting point to complete "delivery".

# 1. Software Demo FAQs

Many answers can be found in the https://github.com/StanfordASL/asl_turtlebot README.

More FAQs are answered in this document.

Here are some common clarifications:

- You can create the food/vendors that you would like to detect for the final project!
  - They can be simple spheres or you can get as creative as you'd like. You can display 2D images in Gazebo! The one requirement we have is that you use the CNN detector rather than just color-coded stop signs, i.e. set use_tf = True.
  - The diagram on slide 6 of this deck is only an example, you are in charge of making your map.
- Your robot should start at "home base" which is just the starting pose of your robot. Once given given orders, it should go to vendors and then return to "home base"
- Orders for deliveries will be given by the TAs verbally after which you use the request_publisher.py script to send out our desired objects.
  - We'll verbally say which objects we want picked up/delivered and you'd enter that in the script.
- You may get orders which have multiple objects in them, the request publisher script can handle this with commas.

# 1. Software Demo

**Baseline:**

- Human operated waypoint following for exploration.
- Logging food locations through TF frames / broadcasting as a topic.
- Autonomous navigation to revisit vendors and drive to delivery location without running into fences.

**Extensions:**

- Autonomous exploration.
- Pose estimation via filtering.
- Shortest path / user-defined order for making the deliveries.
- Detecting and avoiding driving over puddles.
- Calibration of velodyne/camera offsets.
- **More ideas in HW3, or your own ideas!**

**This is very open ended!**

**Talk to us about your ideas. You're free to tweak the setup, add objects, etc. to help you demo your extensions!**

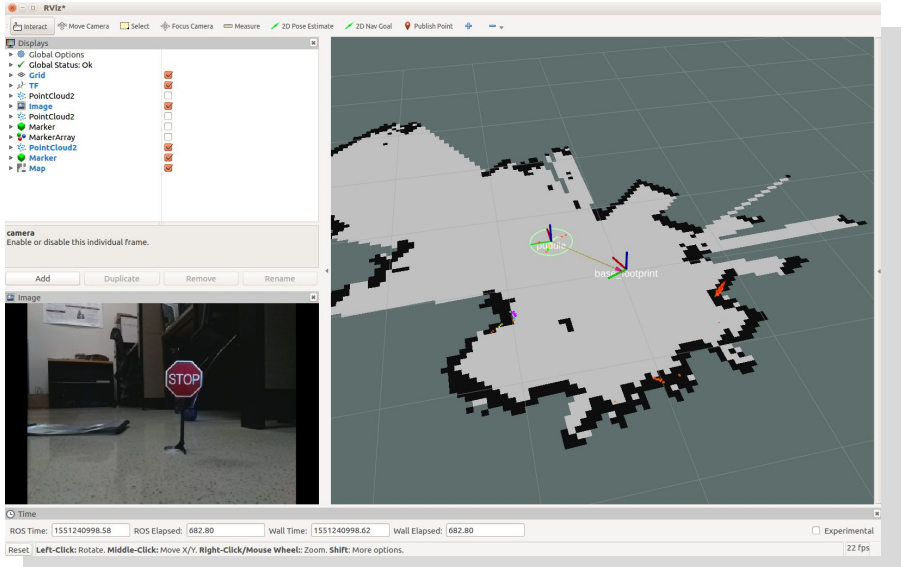
# 2. Command Center Visualizations

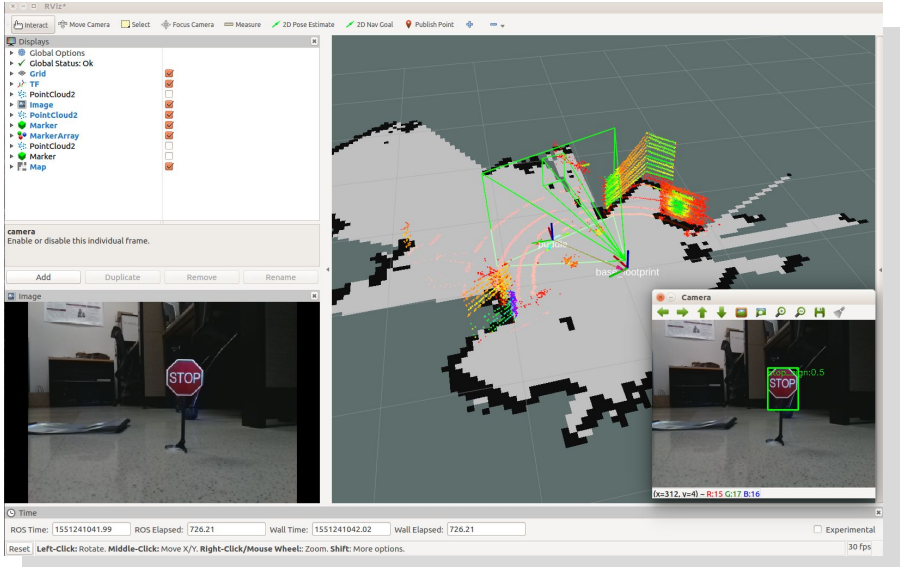
## Baseline:

- Visualize the TurtleBot and desired poses with markers

## Possible Extensions:

- Mark/label objects on map
- Visualize camera FOV



# 3. Pitch Presentation

Three minute presentation:

- Describe your overall software stack.
- Explain your design decisions.
- Highlight what makes your robot unique!
- Include extra slides to tell us about details of your robot.

# Rubric

1. Software Demo on Genbu (~70%):
   baseline is 70% of this, each extension would result in 10% more
2. Command Center Visualization in RViz (~20%)
3. Three Minute Pitch Presentation (~10%)

# Demo Day Logistics

- Demos will be held on **Friday, Nov. 20th** over Zoom from 9am-12pm PST.
- Groups will present and demo in **12-minute slots**.
  - Simultaneously presenting while the robot is moving, to maximize time efficiency.
- We will have **3 groups** doing their demo on Genbu **at the same time** to get through the final project evaluations quicker. You will be in separate Zoom breakout rooms.
- A Google sheet with time slot sign-ups is here. A minimum of 1 person from each group is required to be in the zoom room and present on Genbu during the time slot.