

Robot Navigation: Obstacle Avoidance and Object Interaction on the Moon



TESP '25 – Robotics



Space Robotics Lab
Prof. K. Yoshida



TOHOKU
UNIVERSITY

Andre Khoo

Alessio Borgi

Kristjan Tarantelli

Rasmus Börjesson Dahlstedt

Who are We?



Kristjan Jurij Tarantelli 

Sapienza University of Rome
Master in AI&Robotics



Rasmus Börjesson Dahlstedt 

Chalmers University of Technology
Complex Adaptive Systems, MSc



Alessio Borgi 

Sapienza University of Rome
Master in AI & Robotics



Andre Khoo 

Nanyang Technological University
Bachelors in Mechanical Engineering



Introduction

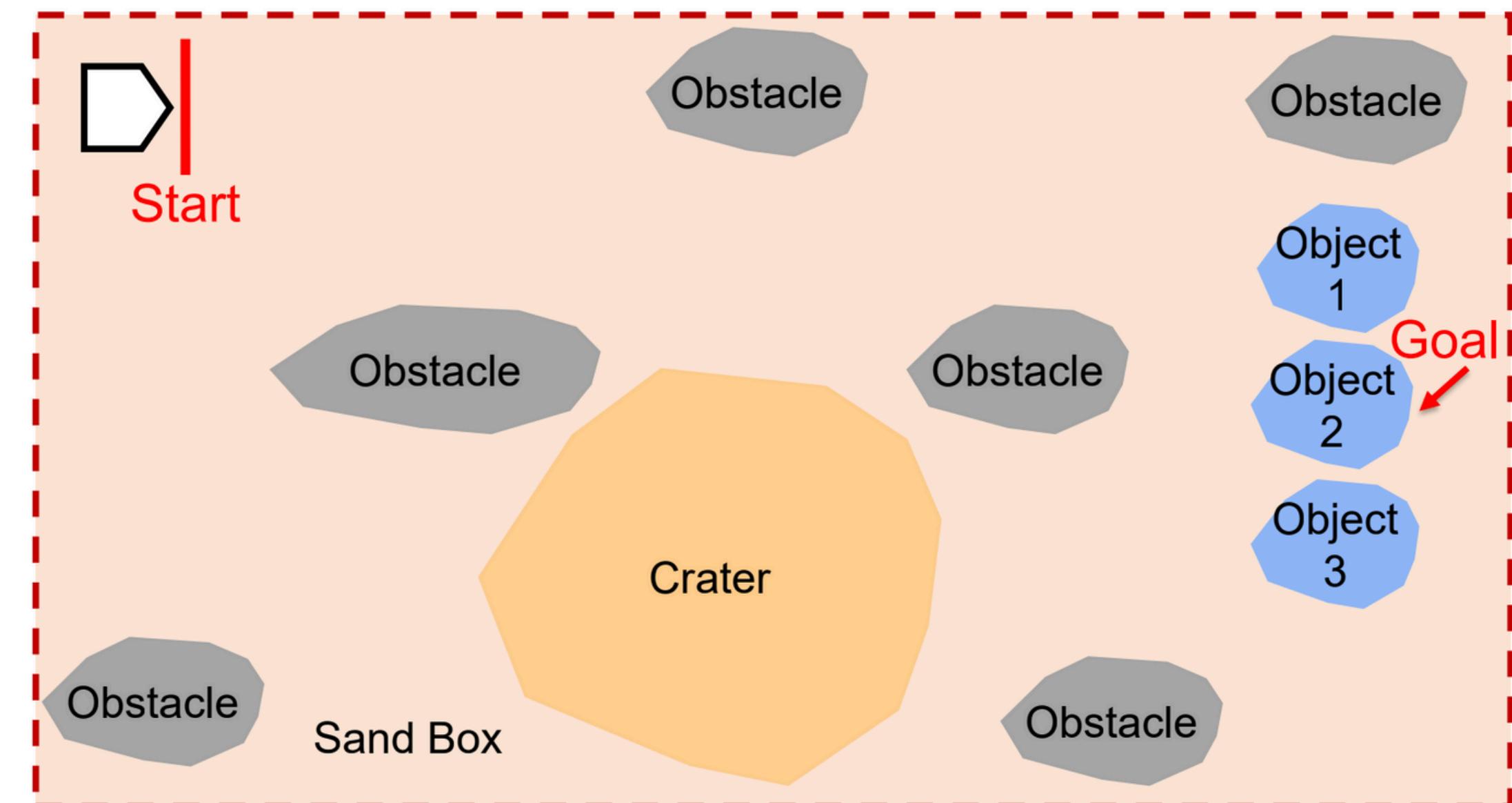


TASK

Make a **mobile robot** navigate through a **static sand field**.

Challenges

- **Build** the robot
- Uneven sandy **terrain**
- **Path** finding
- Object **interaction**



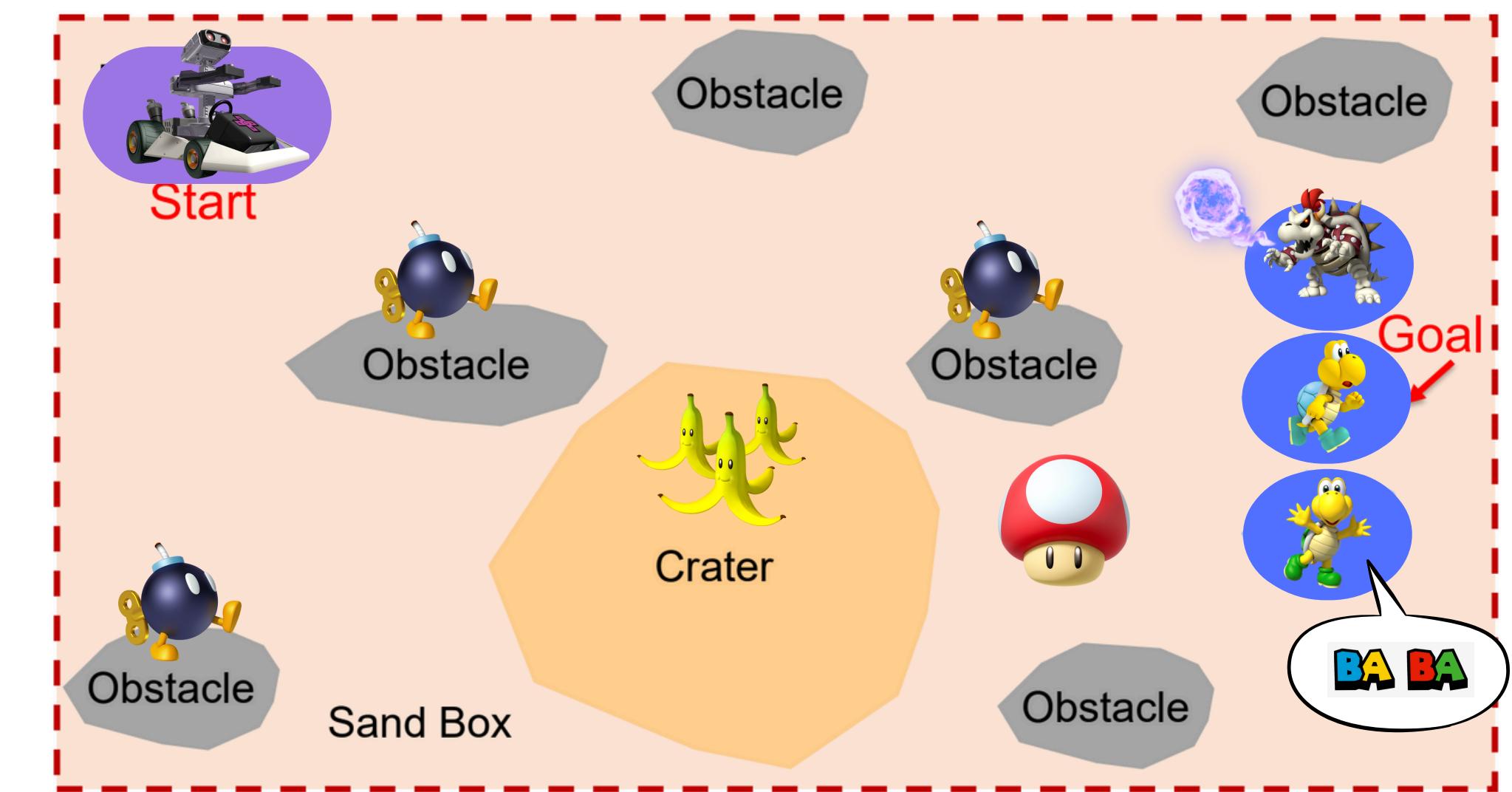
Our Story



The Real Task

- **Koopas** were in an **honeyMOON**
- Entered the reign of **Dry-Bowser**
- He wants to take them hostage
- Mario is busy with Peach

Perfect task for R.O.B.!



Our Story



First Prototype Model

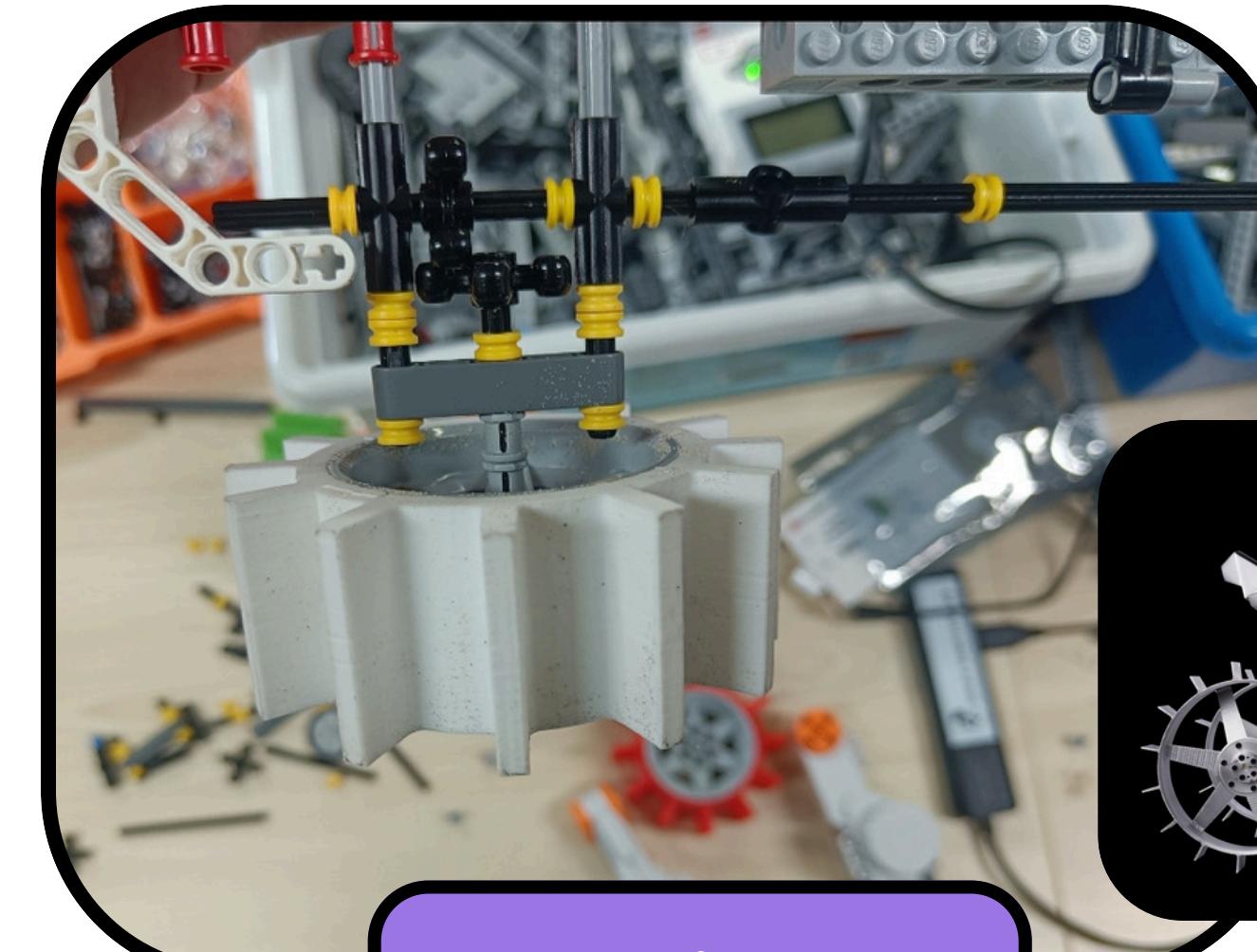


PROs

- Only 2 motors (**lightweight**)
- **Easy** to program
- Best wheels

CONs

- **Large steering** radius
- **Unstable** design



Tsukikage



Second Prototype Model

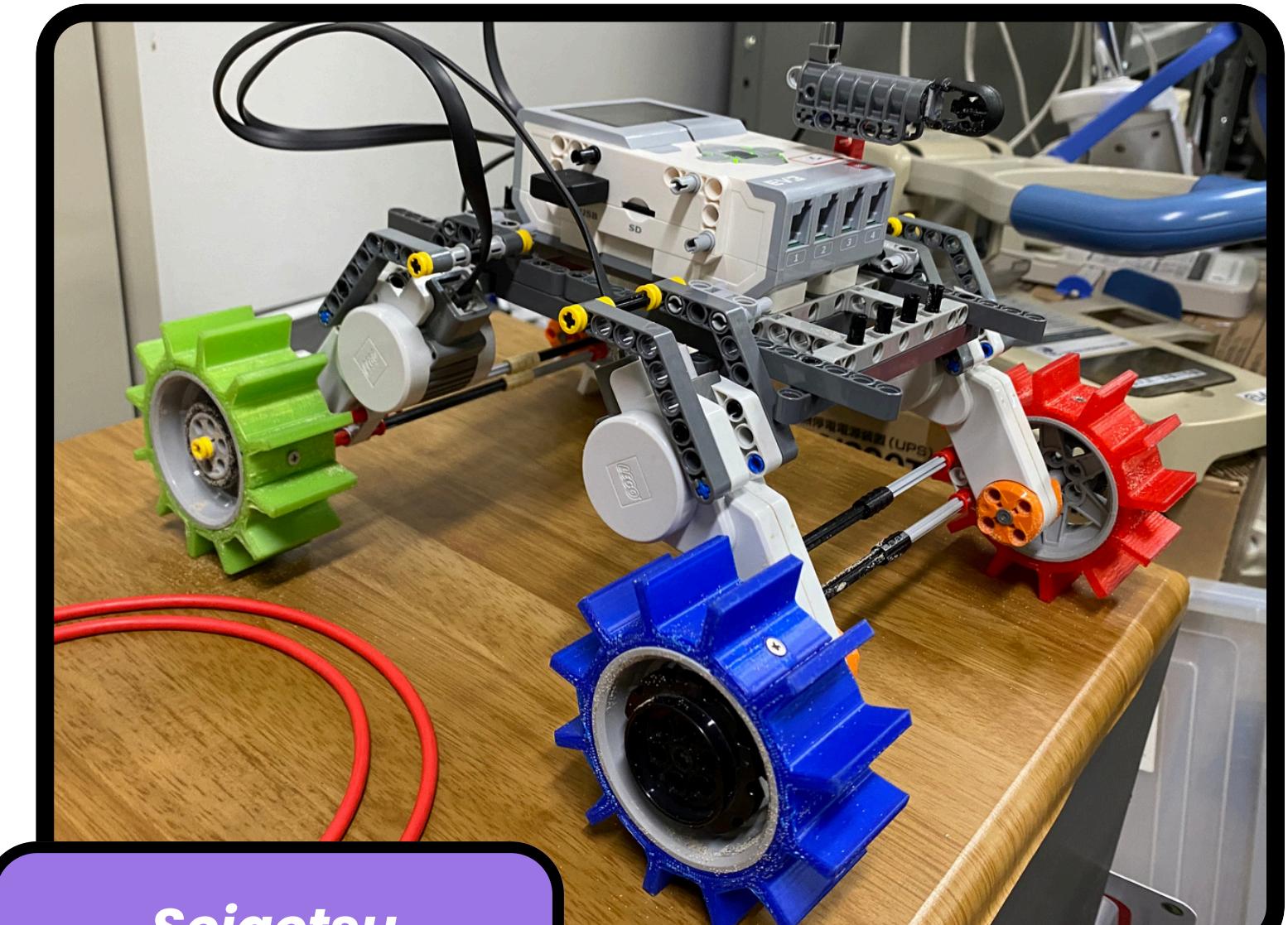


PROs

- Fast and **Powerful** (4 motors)
- **Robust** Design

CONS

- Large Steering Radius
- **Heavy** and Hard to Fit Gripper



Seigetsu



Third Prototype Model

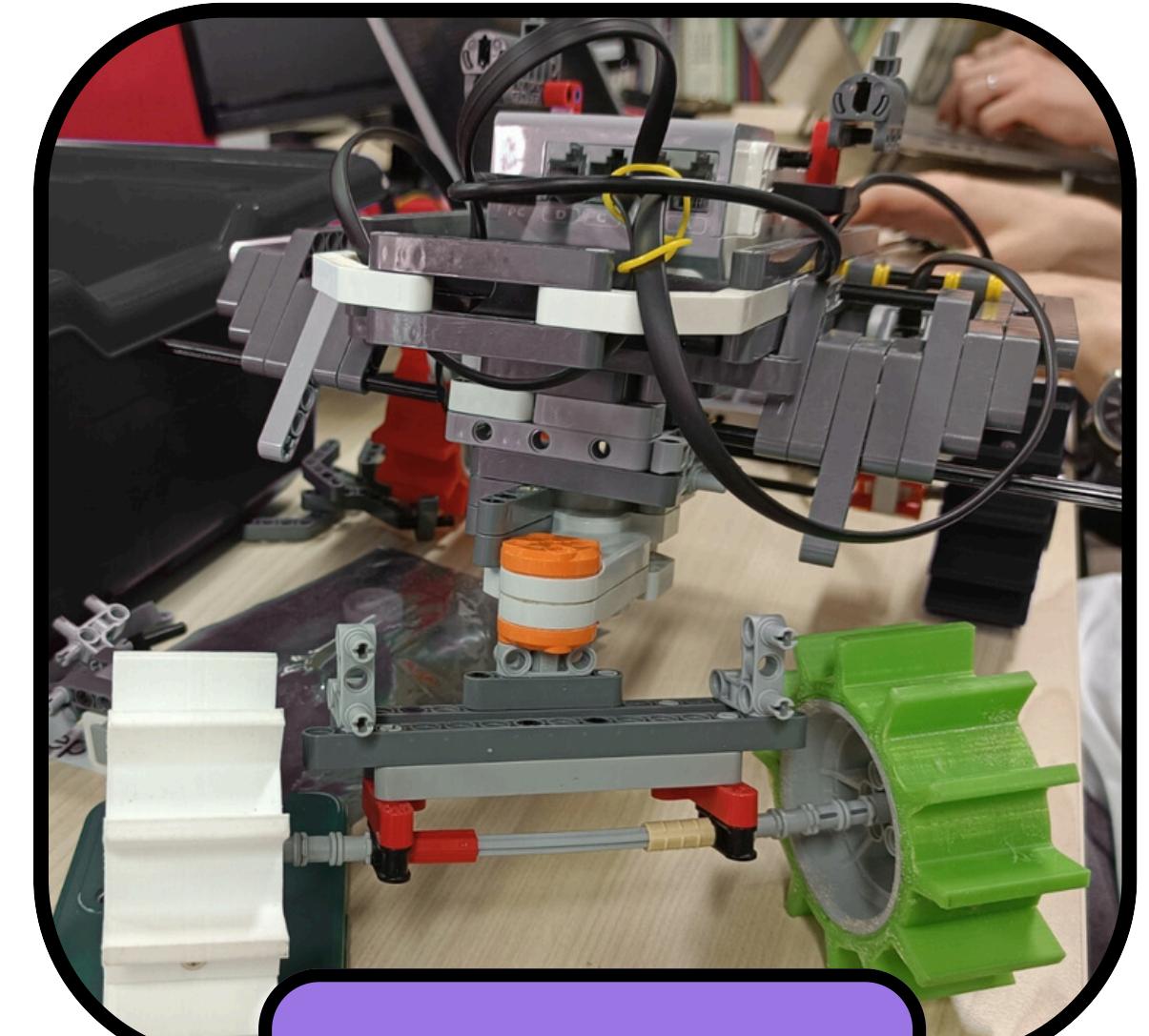


PROs

- Small Turning Radius
- Precise Turning

CONS

- Less Power and Grip
- Unreliable Front End



Mikazuki



Final Prototype Model

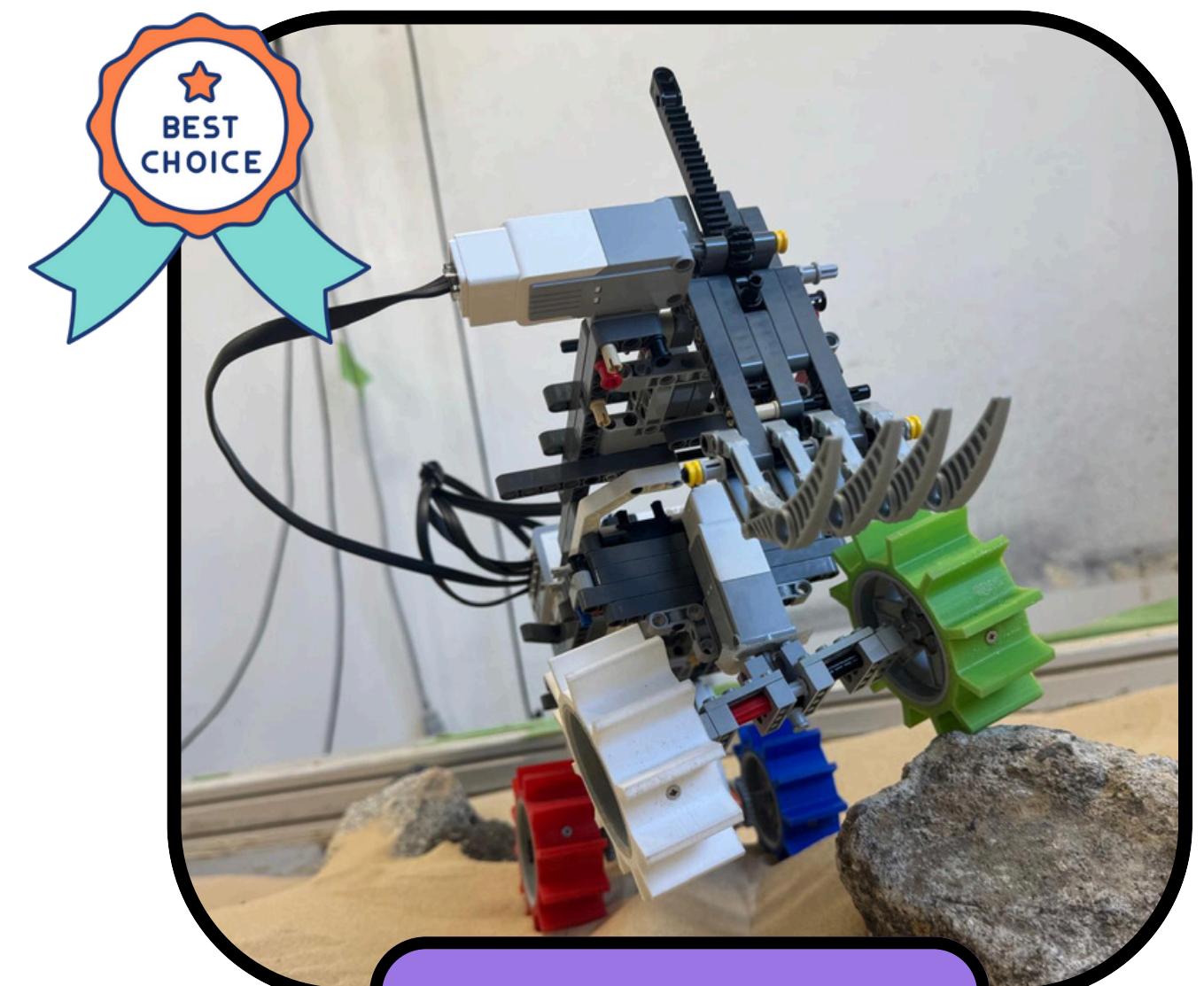


PROs

- Lighter Design
- Sturdier Front End

CONS

- Less Power



Tenshiko

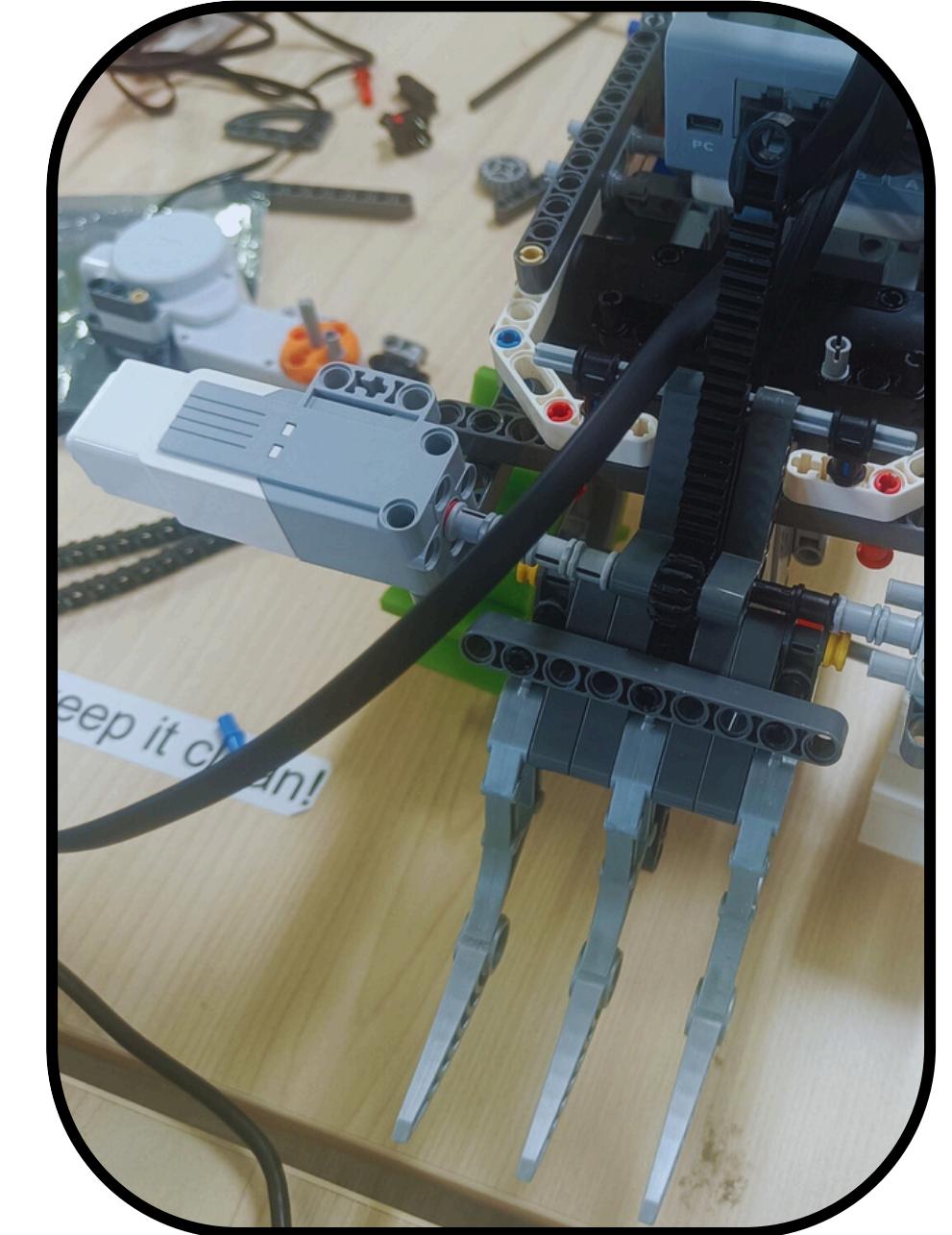
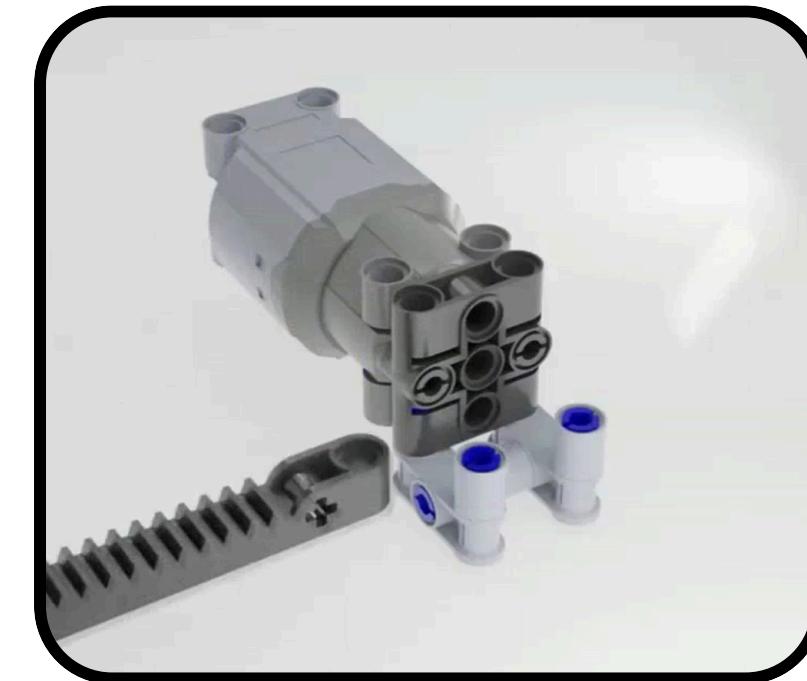


Gripper Design

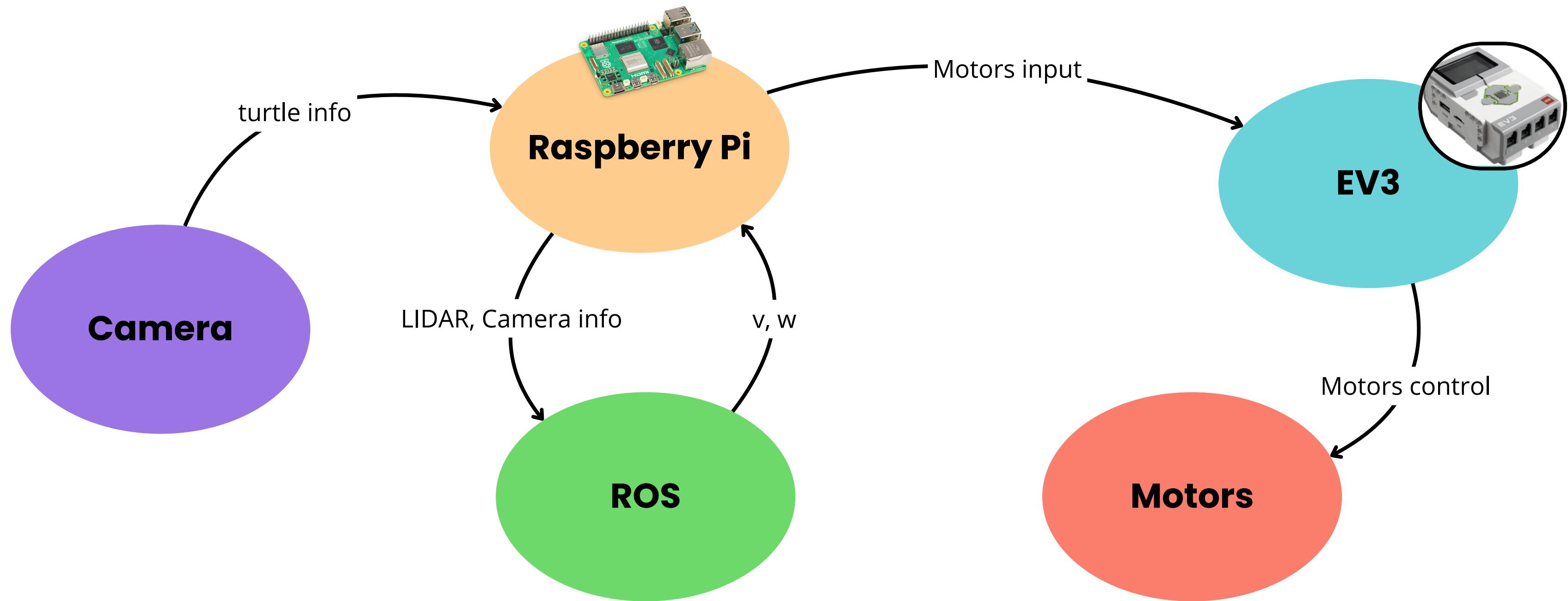


Gripper

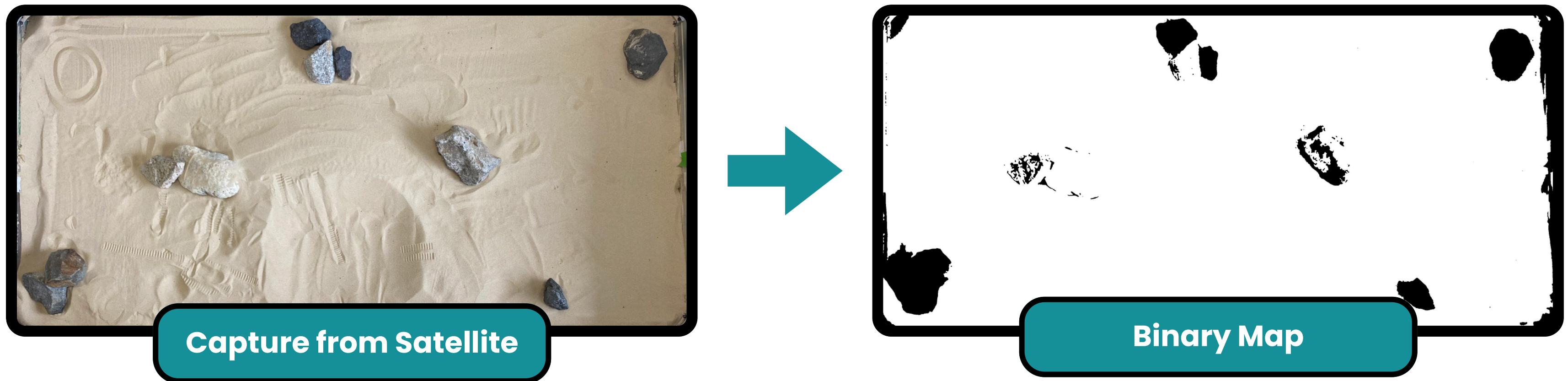
- Design based on **Loaders**
- **One motor** only
- From rotational to **linear**
- Must **not touch** the ground when navigating **until goal**



What we wanted to do



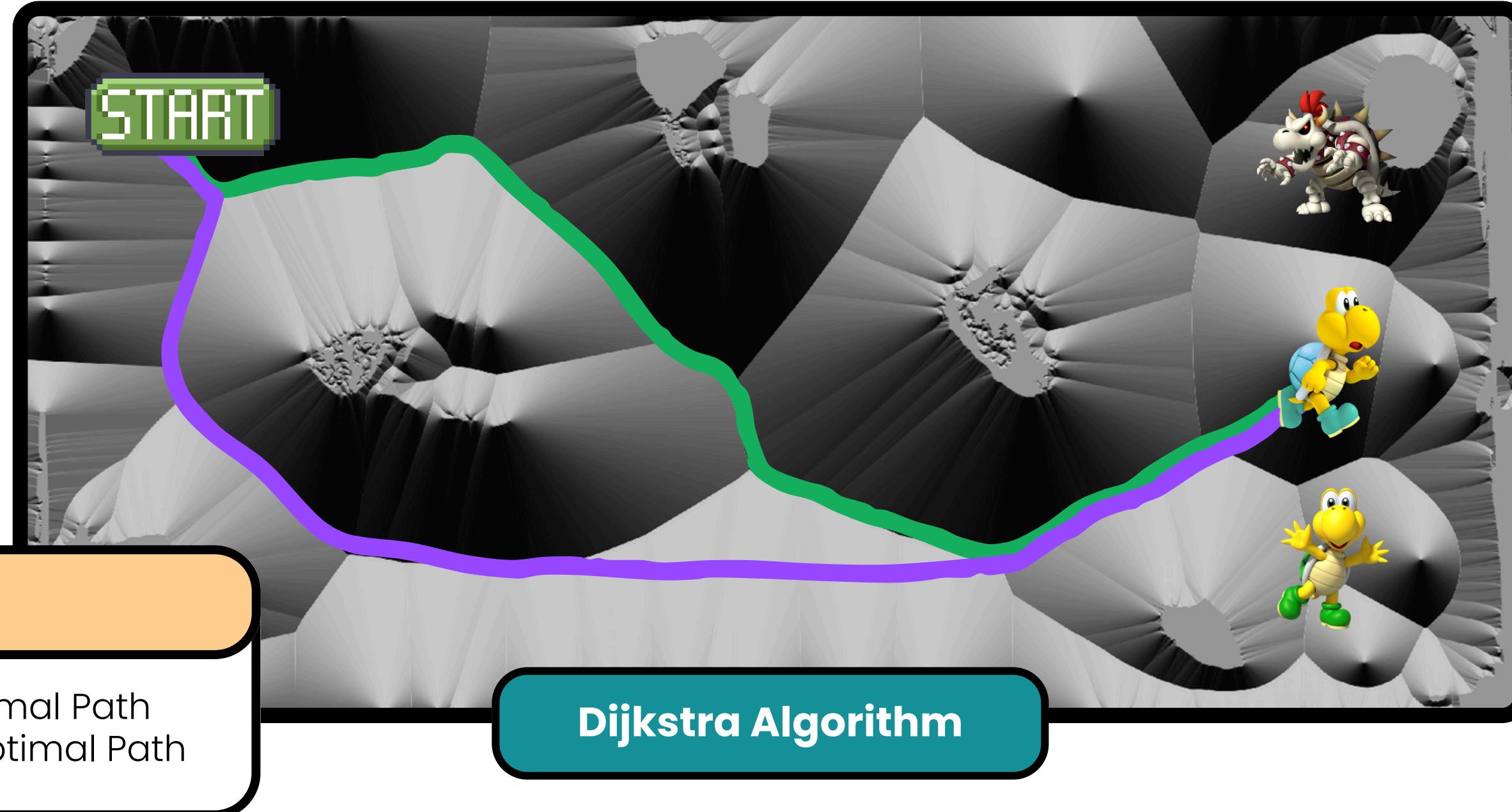
Planning



Planning



Planning



ROS2 Simulator



RViz

- **Draws** the map, the path and the TFs

Simulator

- 1 KHz refresh rate
- **Updates** the TFs

Planner

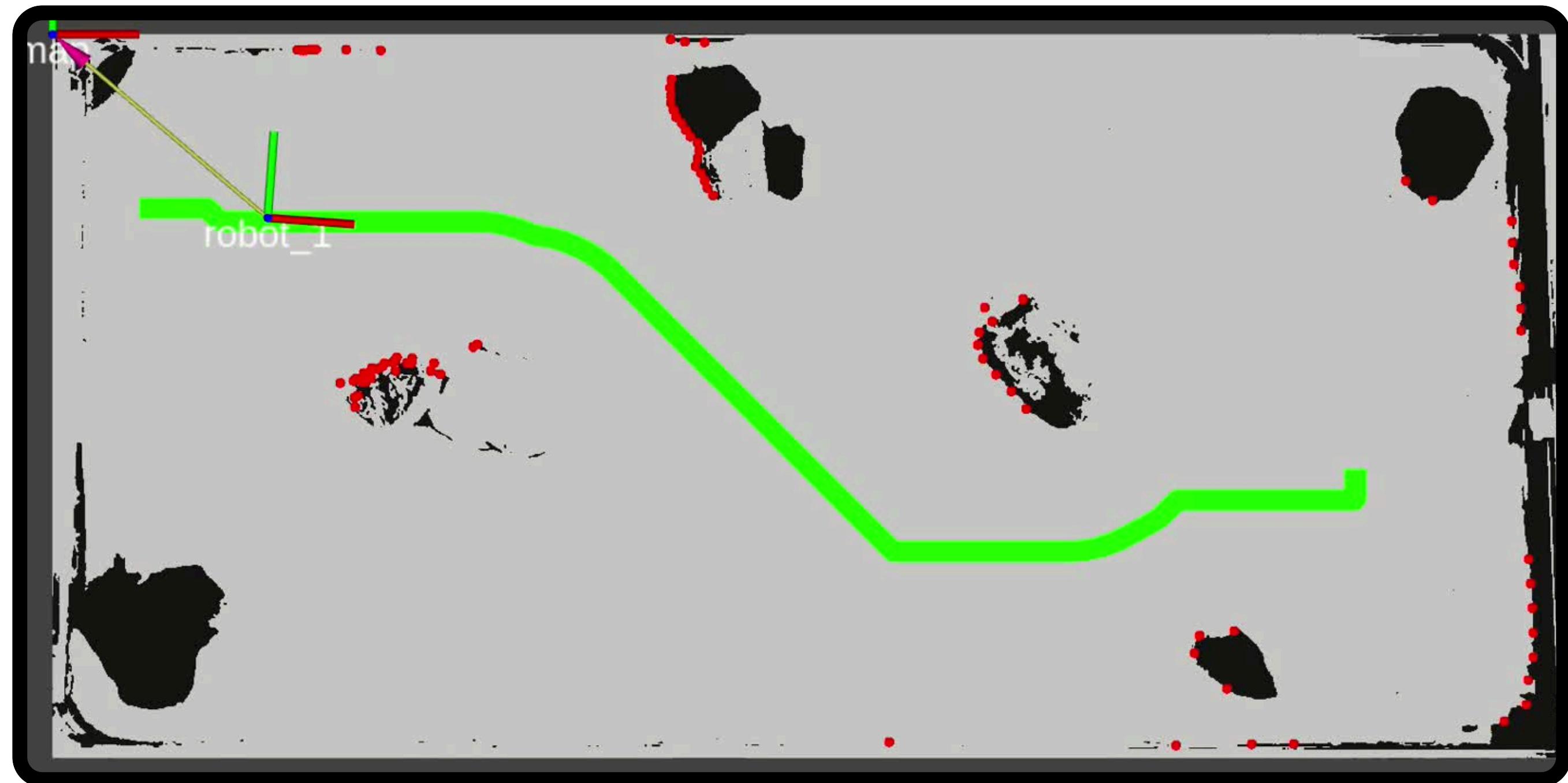
- Takes the Binary Map and runs **Dijkstra** Algorithm
- Publishes the **waypoints**

Controller

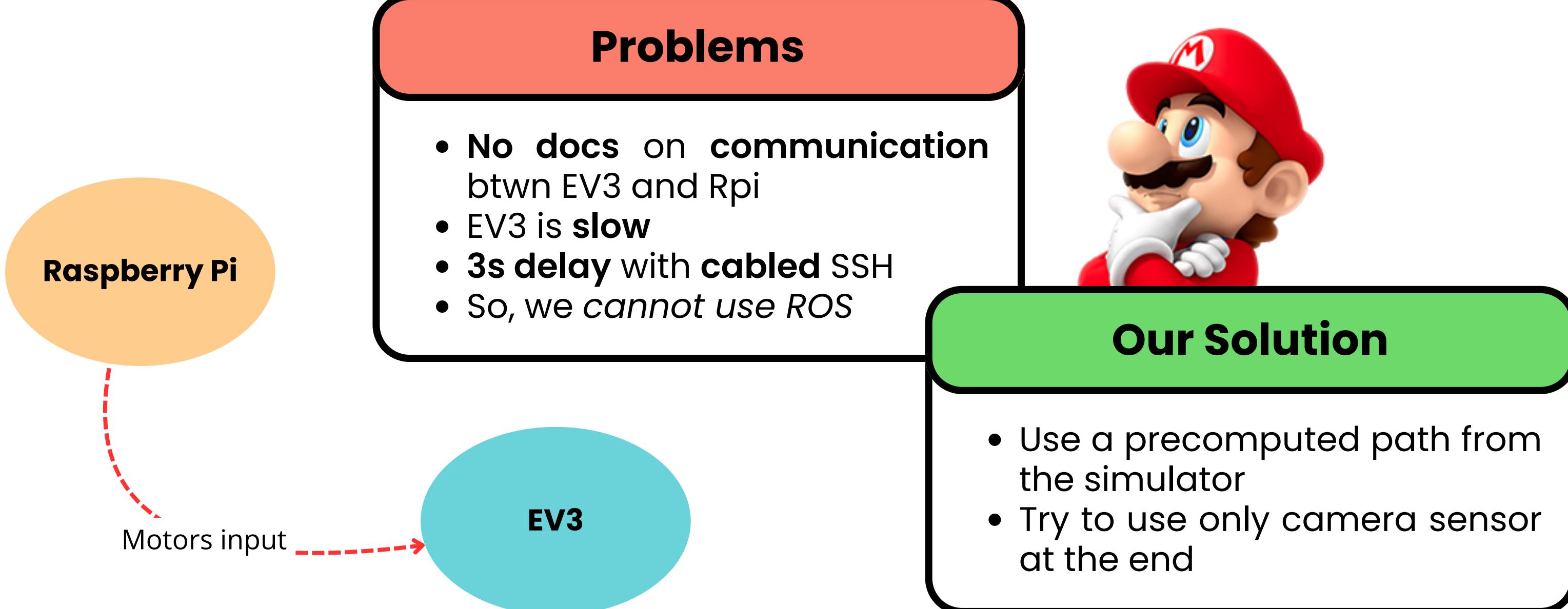
- Proportional and **Derivative**
- Follows the path **waypoints**
- Outputs vel **v** and angular vel **w**



ROS2 Simulator



Encountered Problems

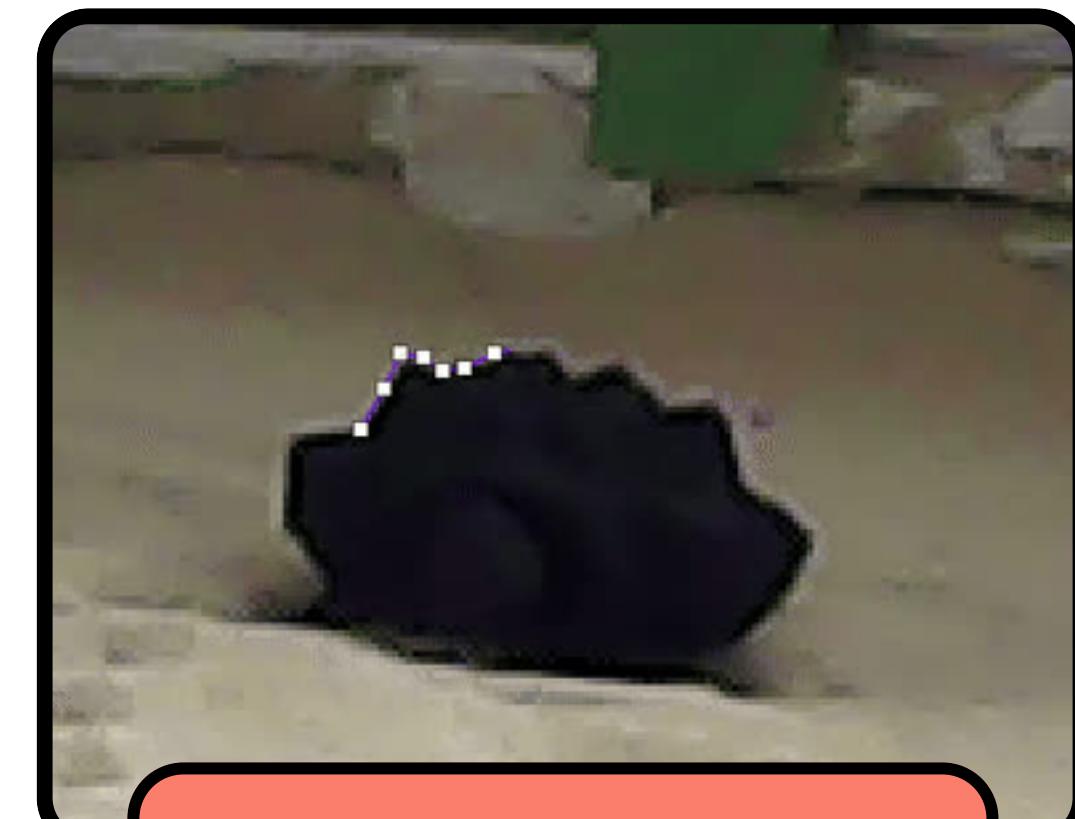


Object Detection Model and Training



Camera

- GC0308 CMOS Sensor Color
- 2MP, 480p, 30 FPS

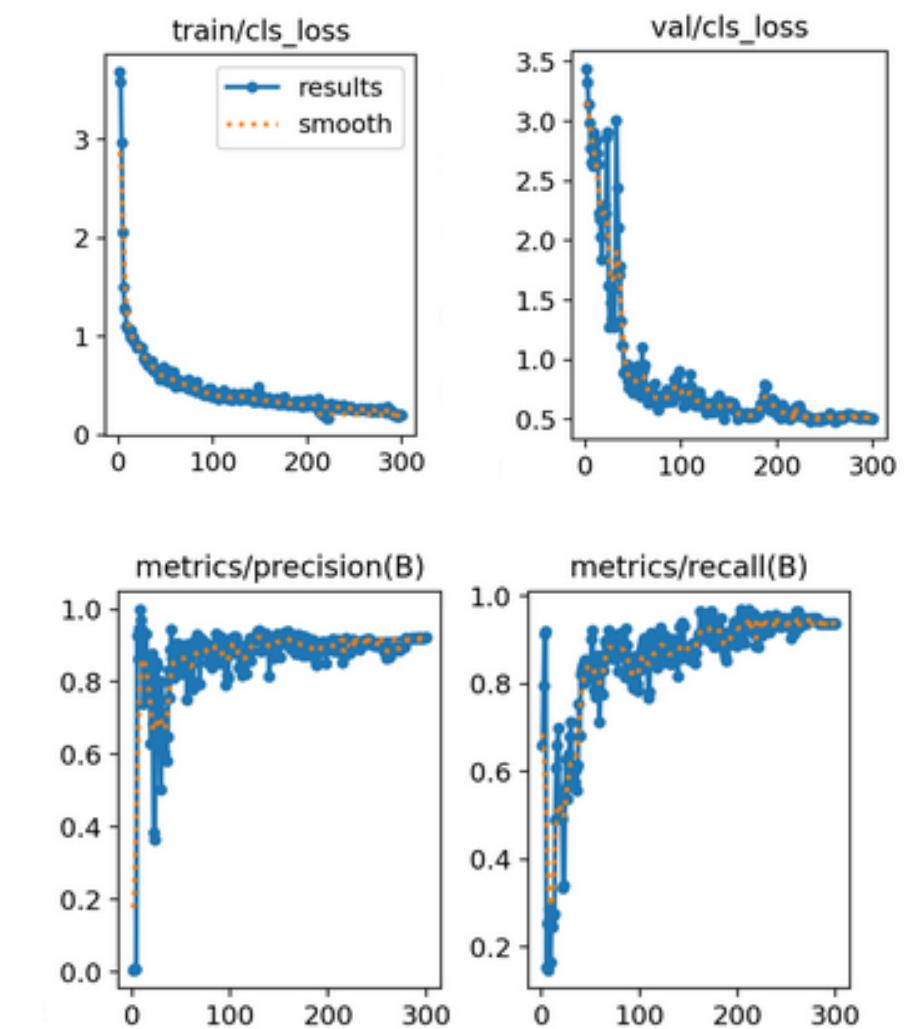


ML Model

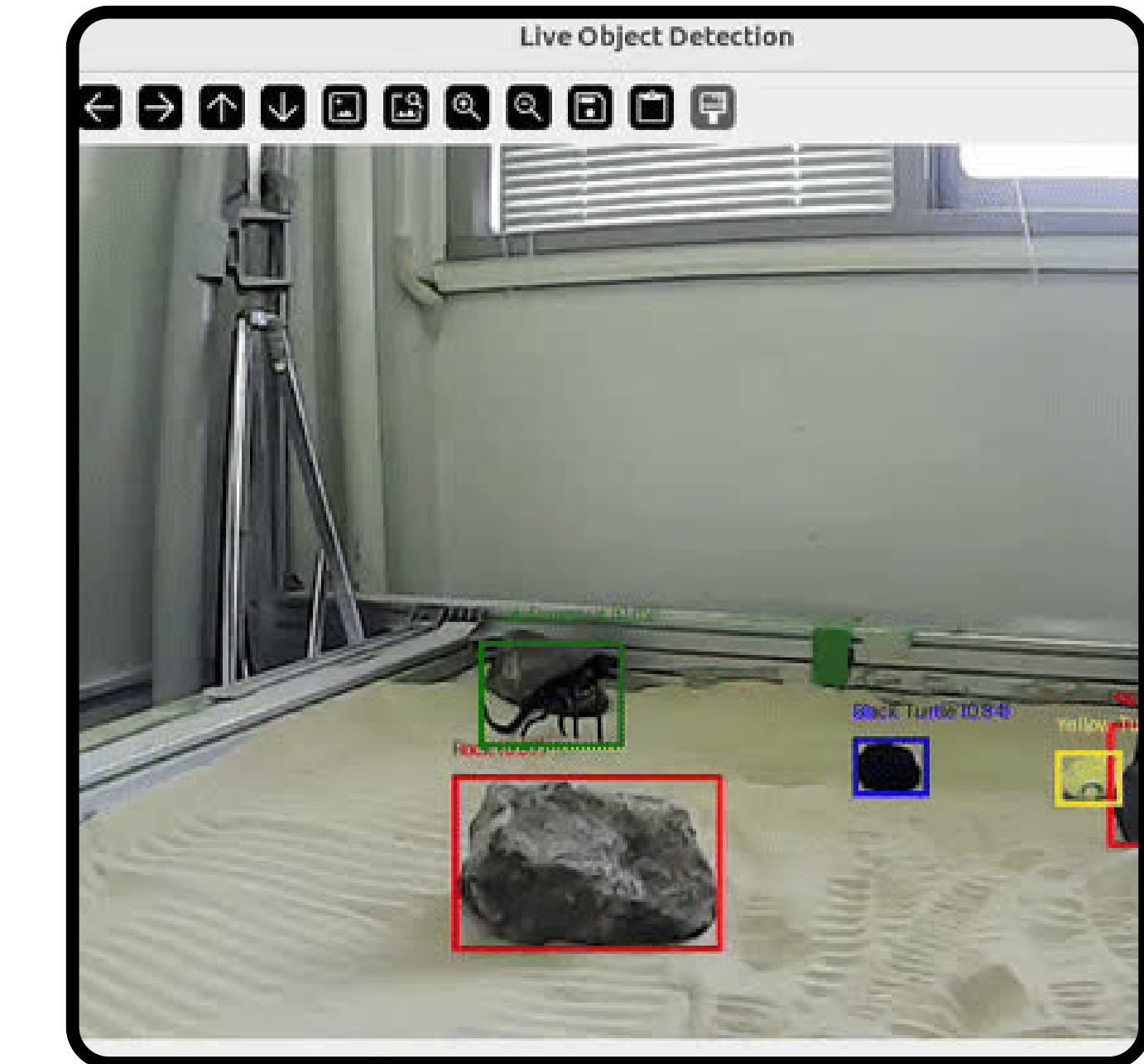
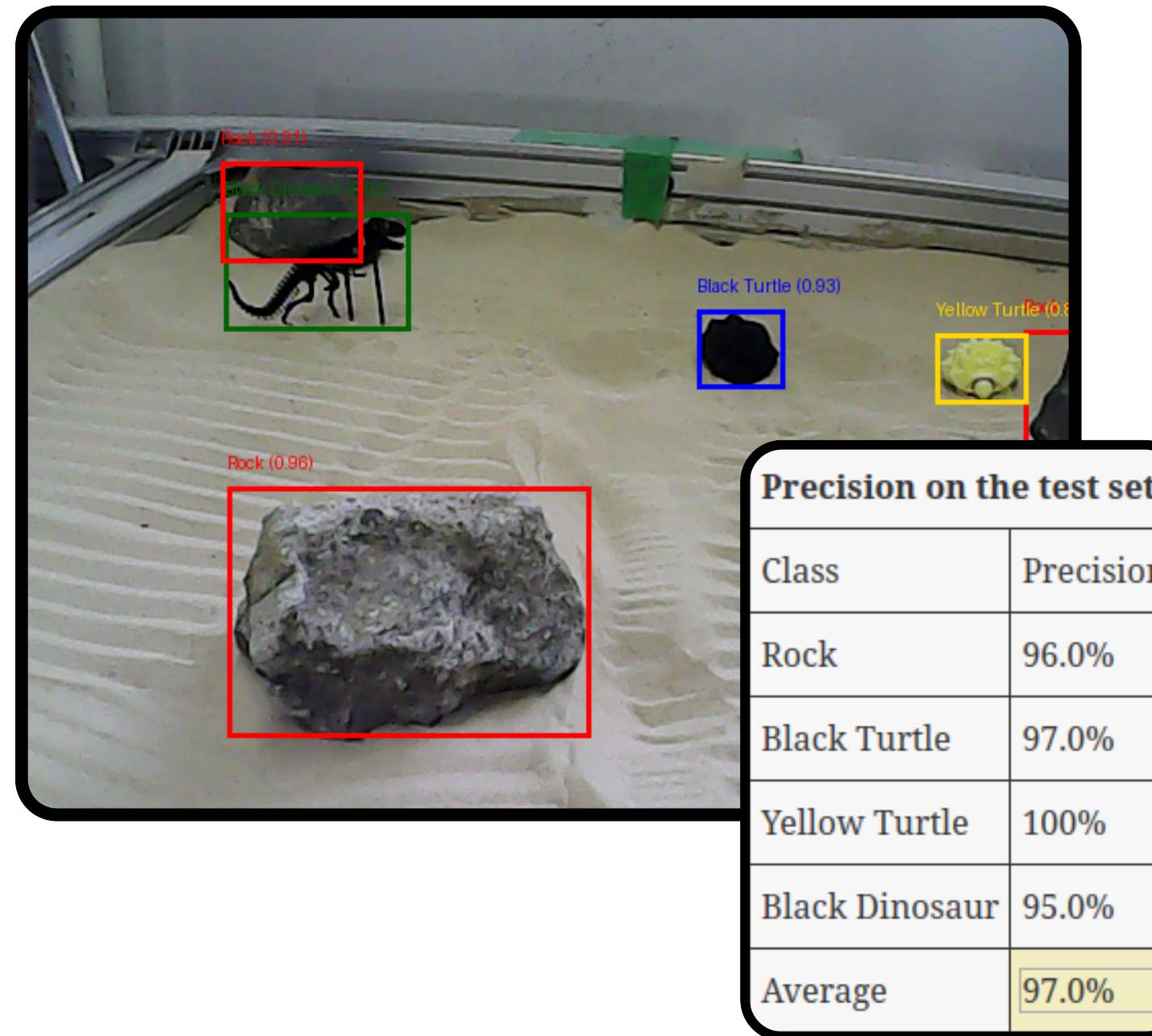
- Roboflow Object Detection
- 240 Images as Training Data
- Data Augmentation

Manual Masks

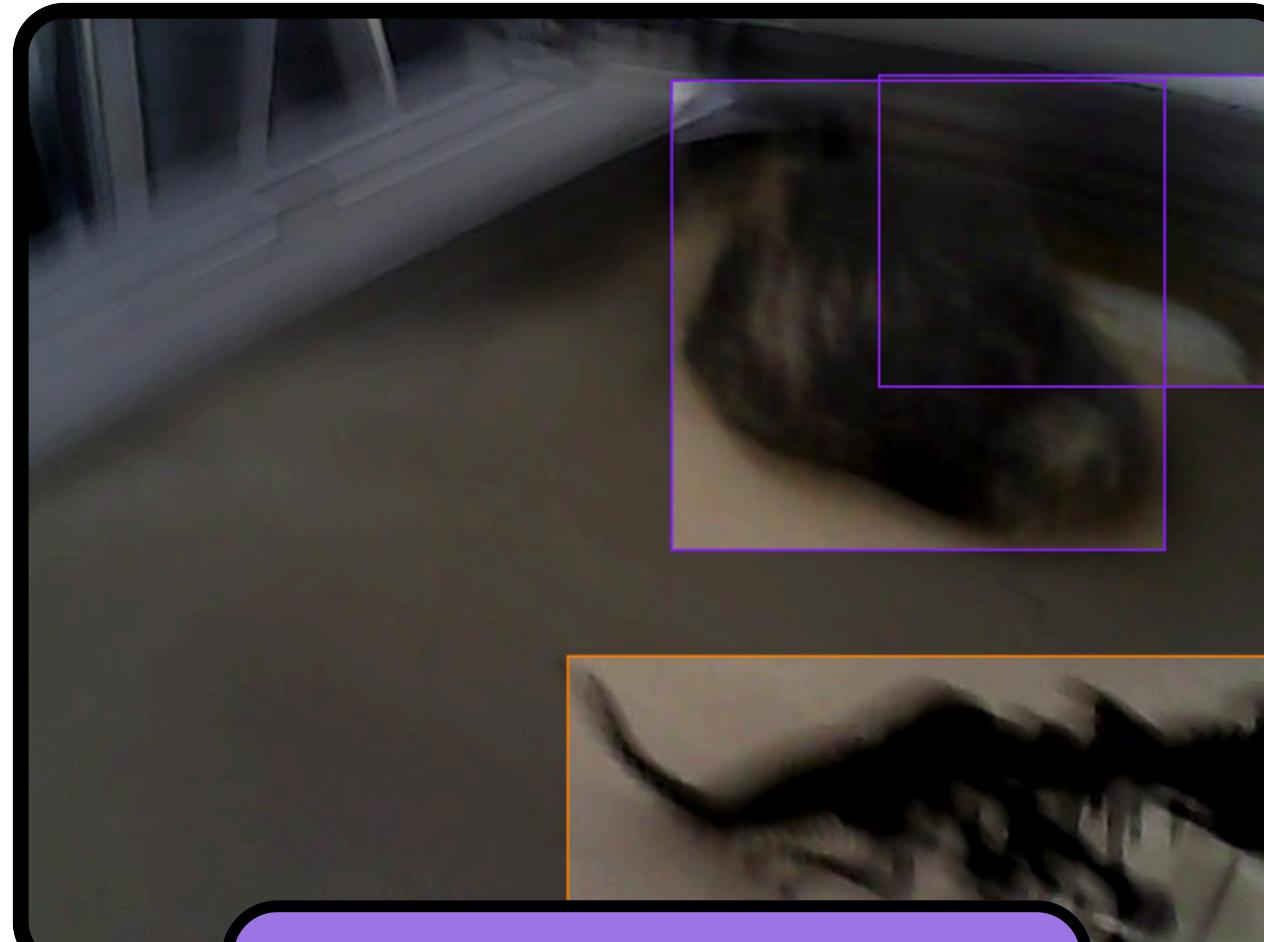
Training Progress



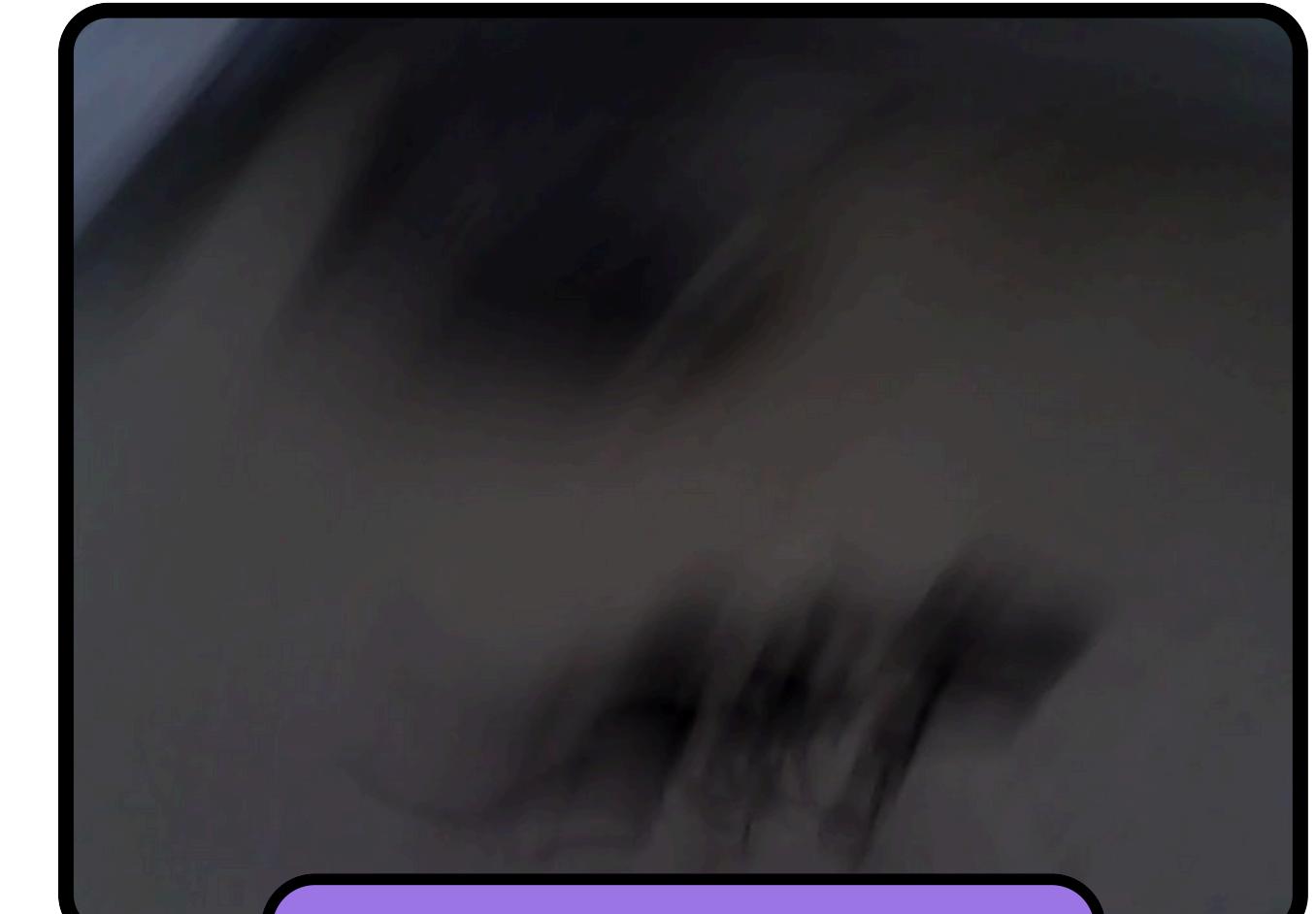
Object Detection Results



Failed Detection Examples



Blurry



Very Blurry



Close range steering using camera

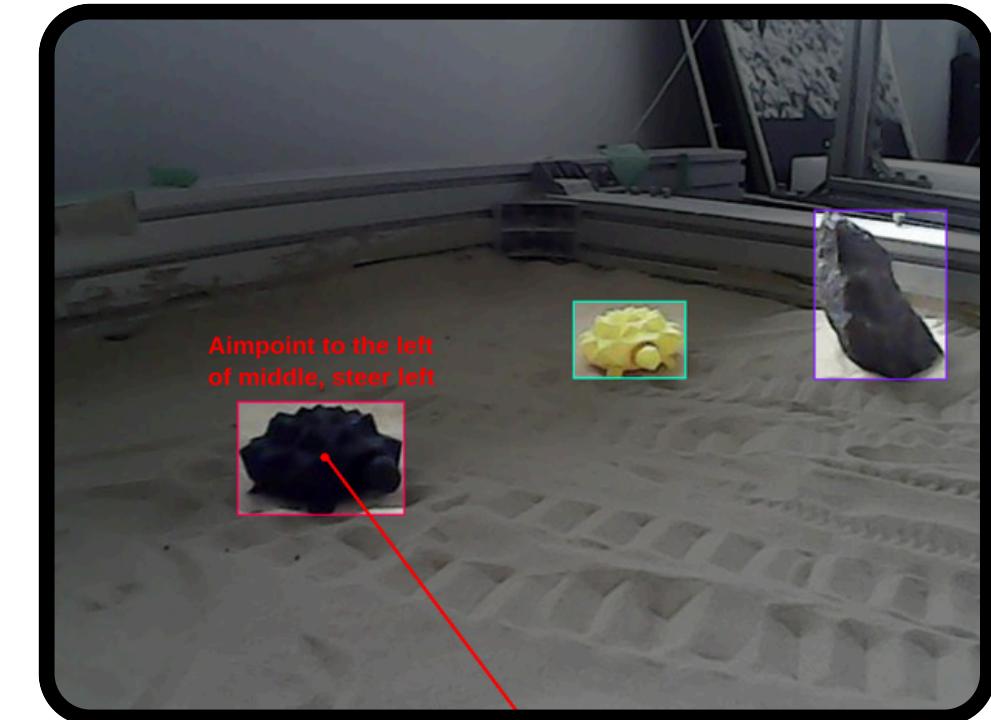
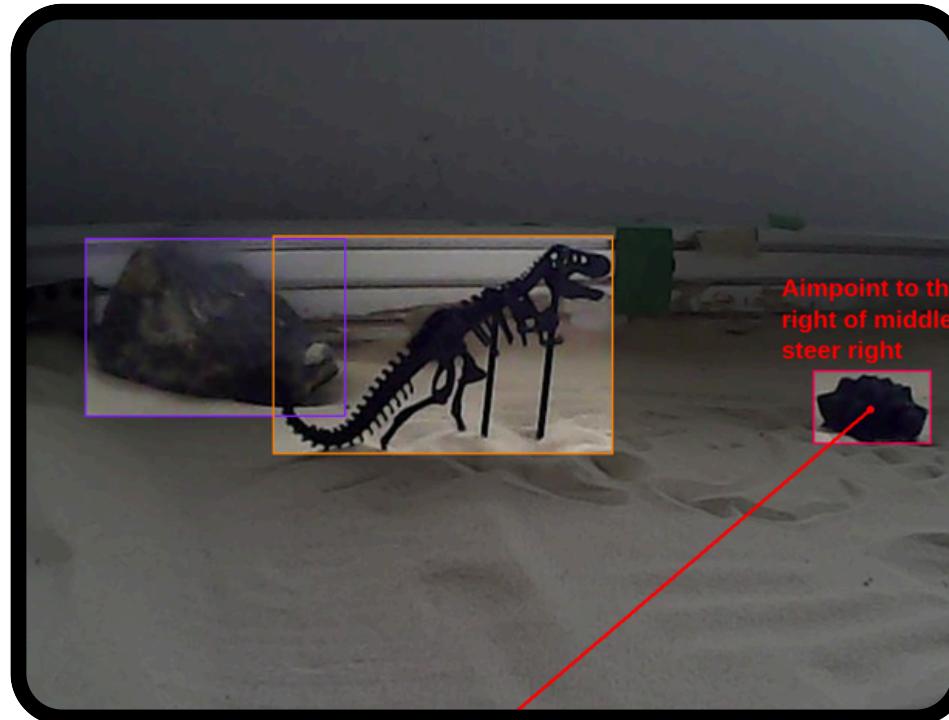


Problem

- Nondeterministic error such as slipping, gliding etc.
- Robot does not end up at turtle
- Makes gripping hard

Solution

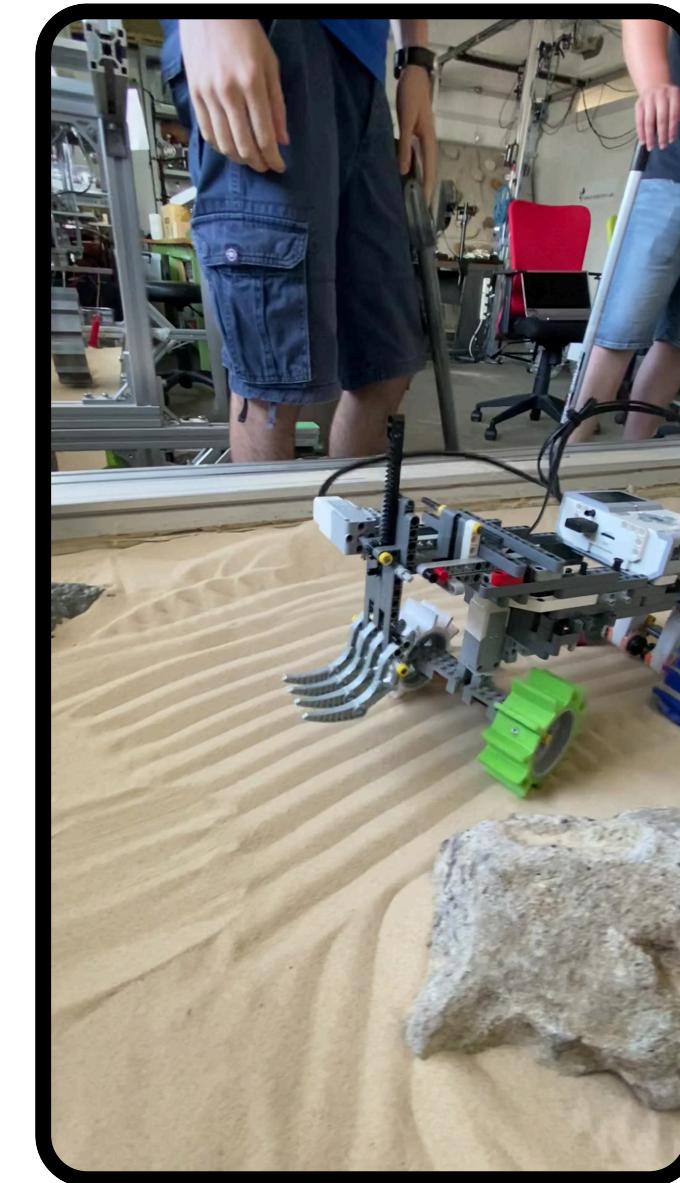
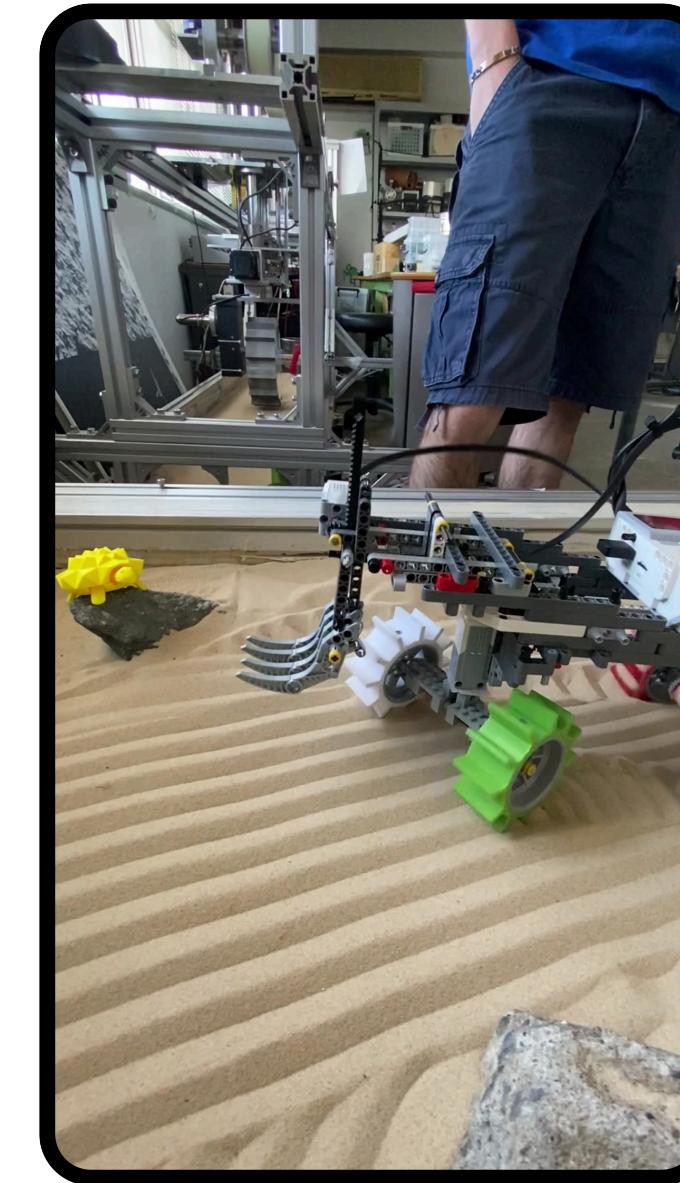
- Simple aimpoint model
- Steer to keep turtle centered
- Used when in vicinity of the target



Results



Fails



Future Works



Hardware

- Use different motors
- Do not use EV3, only Rpi
- Add distance sensors

Software

- Connect Ros with motors
- Speed up the ML model





SCAN ME

Thank You!

Questions?



TESP '25 – Robotics



Space Robotics Lab
Prof. K. Yoshida

TOHOKU
UNIVERSITY



Andre Khoo

Alessio Borgi

Kristjan Tarantelli

Rasmus Börjesson Dahlstedt