10th ICMRE 2024

2024 10th International Conference on MECHATRONICS AND ROBOTICS ENGINEERING

Milan, Italy

February 27-29, 2024

Sponsored by

IEEE

IEEE Robotics & Automation Society

BICRI

Patrons

FUDAN UNIVERSITY 1905

YORK UNIVERSITÉ UNIVERSITY

CLERMONT AUVERGNE UNIVERSITÉ 1889

UNIVERSITÉ LE HAVRE

INSTITUT CARNOT ARTS

LAMIH UMR CNRS 8201

Design and Implementation of an Autonomous Mobile Robot for Slug Detection and Safe Collection to Prevent Agricultural Damage

Nabil Miri, Jana Marzouk, Abdulrahim El Mohamad, Yahya Aljamal, and Hassan Hariri

Fighting Slugs in an ethical way

Rafik Hariri University

x

SPEXAL SARL

Problem:

Plants Pests:

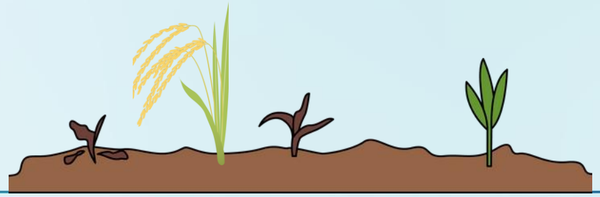
- Decimate emerging crops
- Significant impact on yield

Harmful Types:

- Black keeled slugs - Reticulated slugs
- Costing UK farming about £43.5m a year

Spot:

- Heavy soils and wet areas in the high rainfall area
- Active after rain
- Poorly drained soils



Traditional Solutions:

Coffee Grounds

Aroma

Hand Picking

Manual and tiring

Slug Pellets

Detriment effect on environment

Weather dependent

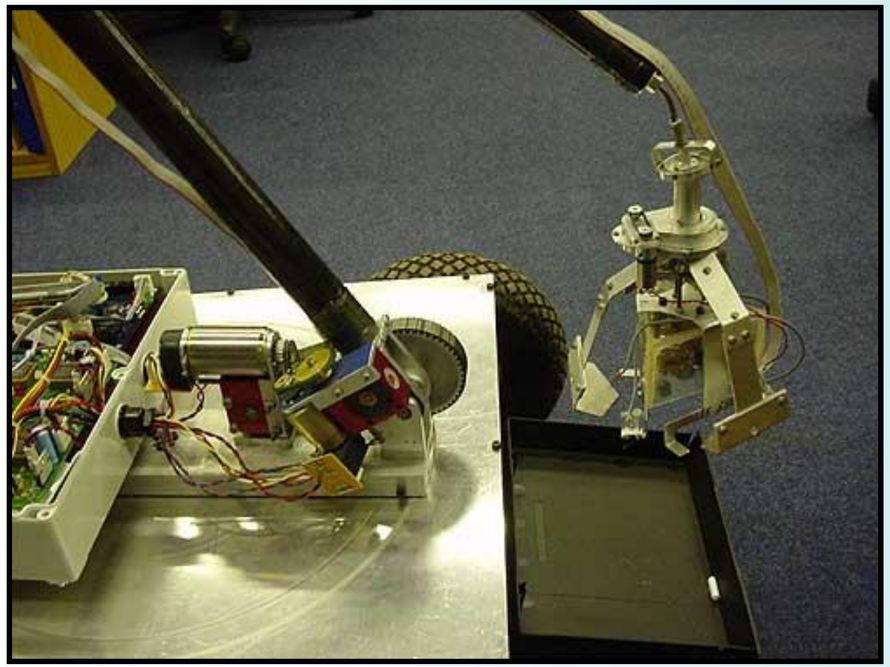
[metaldehyde]

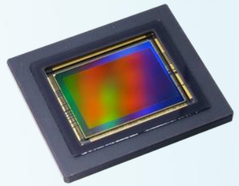
Traps

Ineffective

Smart Solutions:

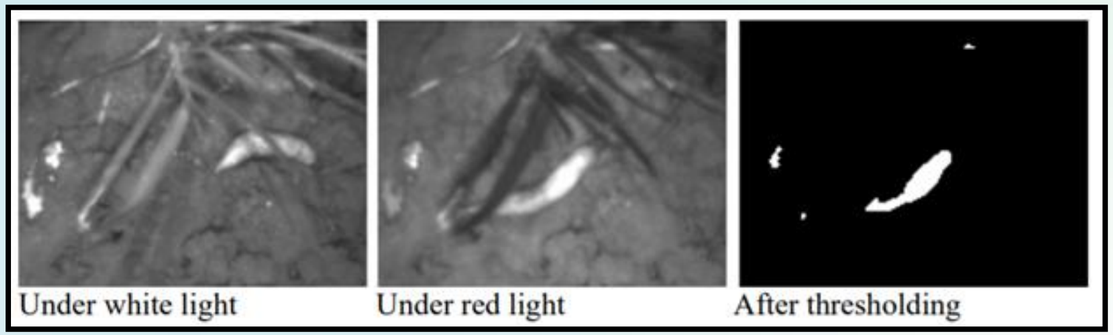
- Slugbot 2001 (University of West England)





CMOS image sensors

- Last public article in 2013
- University of West England's Intelligent Autonomous Systems Laboratory
- TIME 2001 invention of the year
- Use rotting slugs bodies to generate electricity

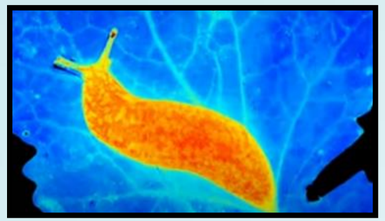


Smart Solutions:

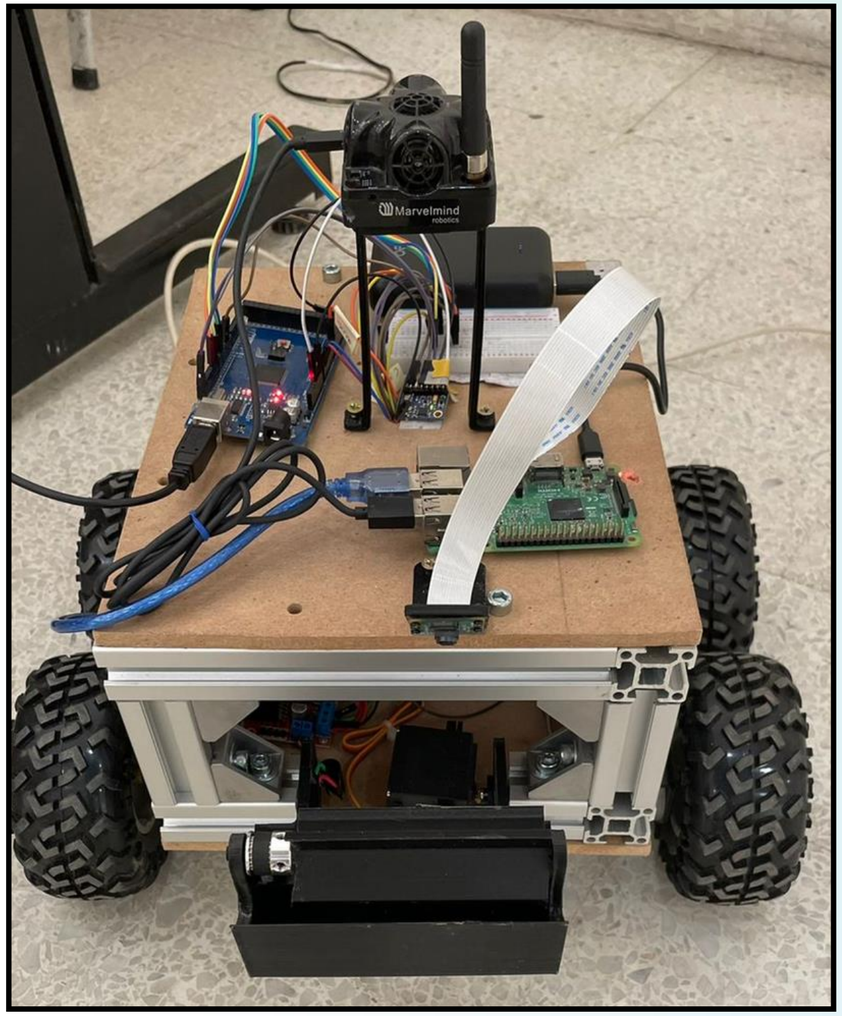
- Slugbot (Small Robot Company, CHAP, COSMONiO and AV and N Lee)

  Still under development

- Detection based on morphological features using Multi-spectral imaging
- Spraying system
- Used for big arable farms

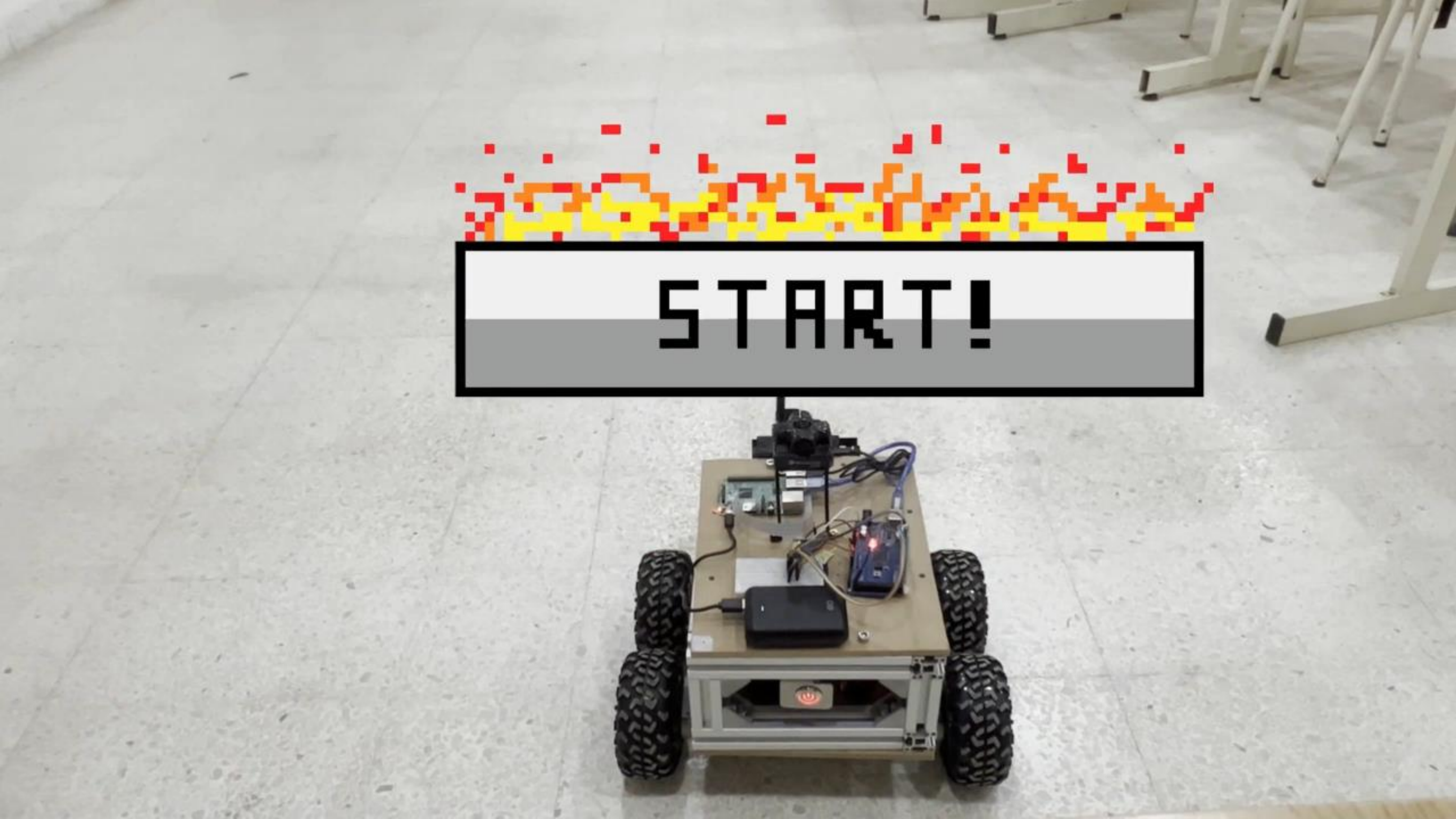


Our Solution:

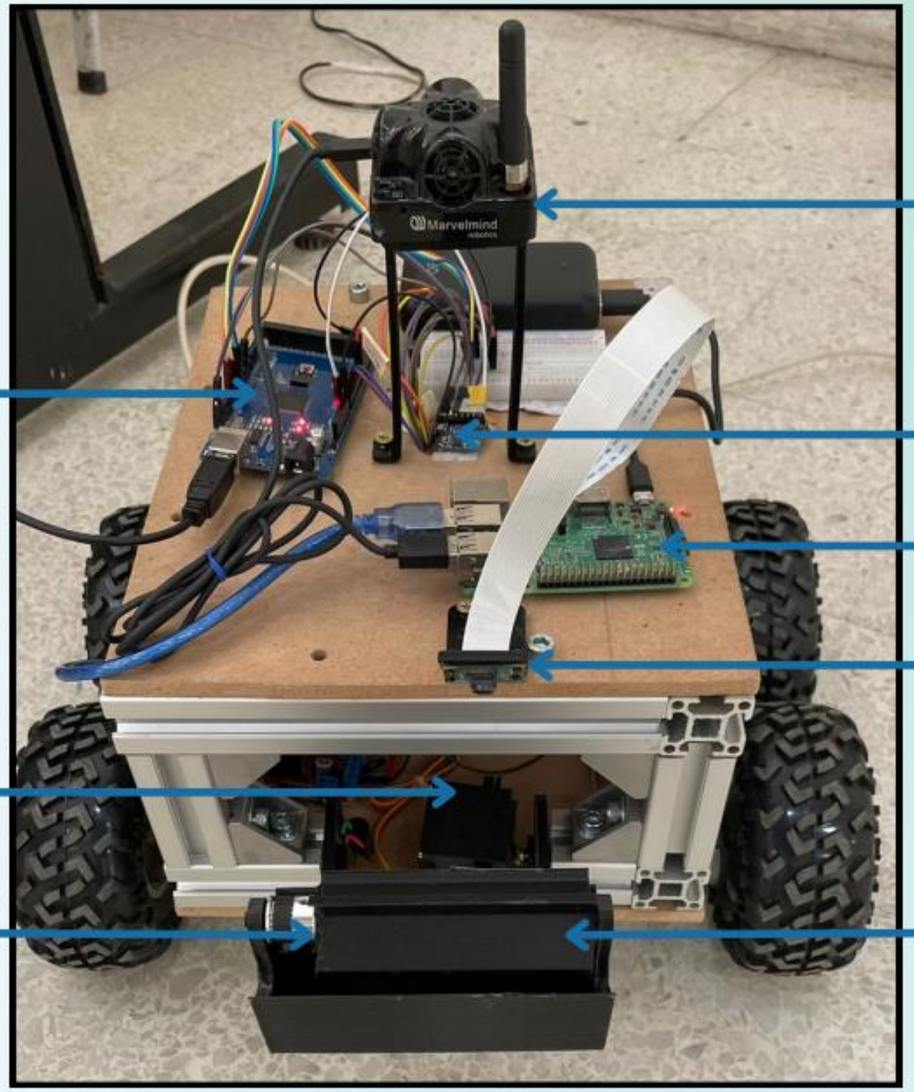


SLUGYBOT

FIGHTING SLUGS IN AN ETHICAL WAY


START!

Our Solution:



PLAN:

Design

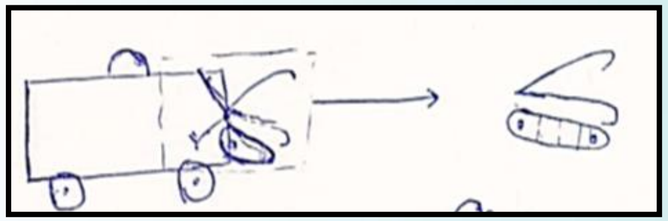
Vision

Navigation

Constraints & Future Work

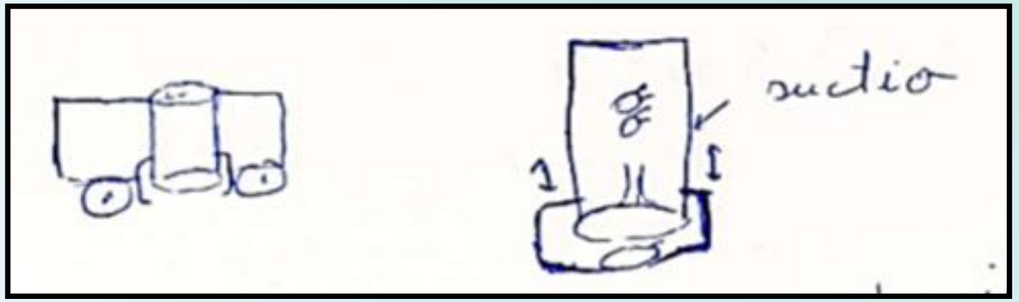
Collection Mechanism Design:

Initial Designs ⇢ **CAD Design** ⇢ **Testing** ⇢ **Implementation**

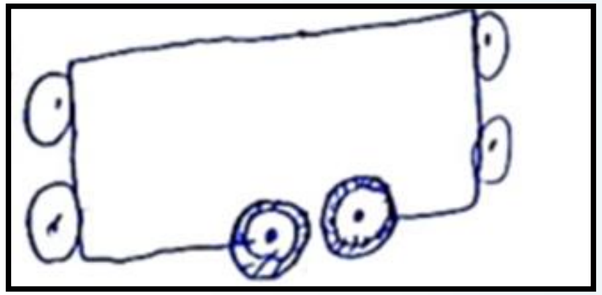


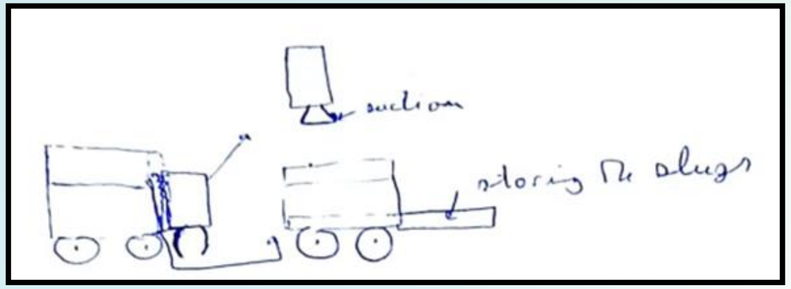
BULLDOZERS

ROLLERS



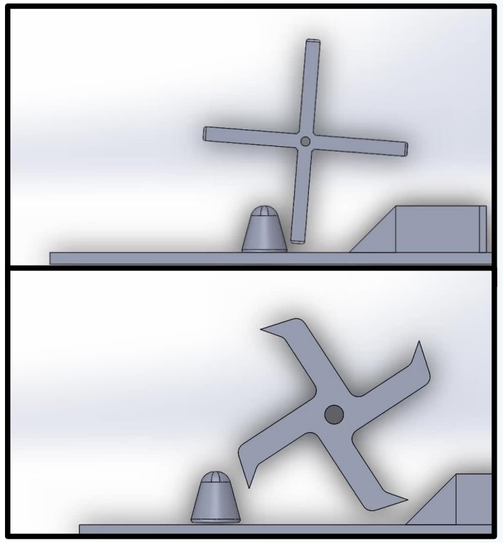
GRIPPERS

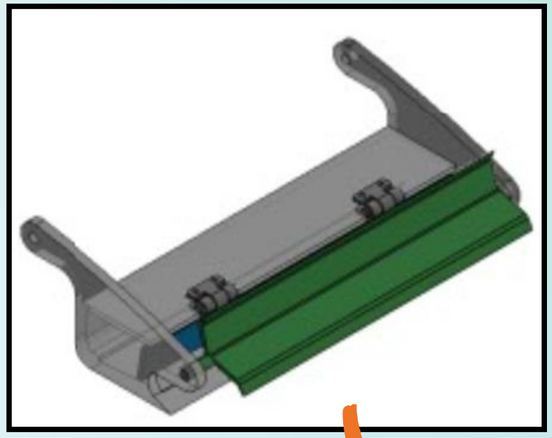


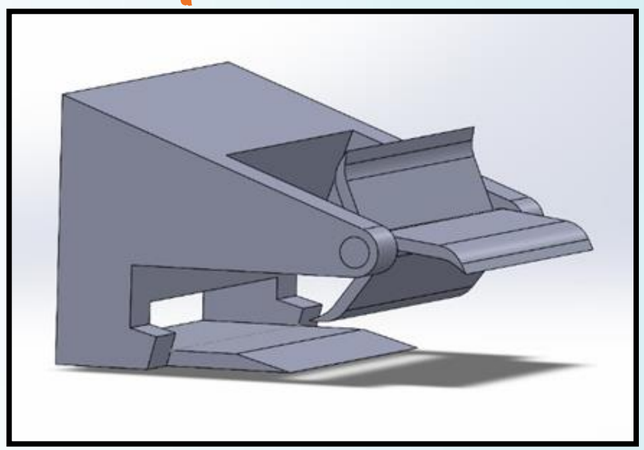


Collection Mechanism Design:

Initial Designs → **CAD Design** → **Testing** → **Implementation**







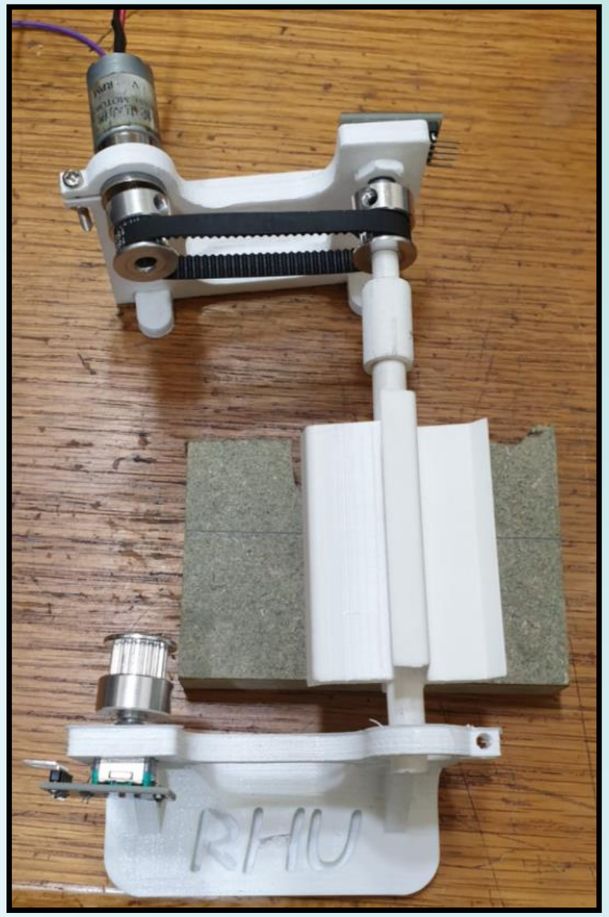
Test

Collection Mechanism Design:

Initial Designs → **CAD Design** → **Testing** → **Implementation**

COLLECTION MECHANISM TESTS

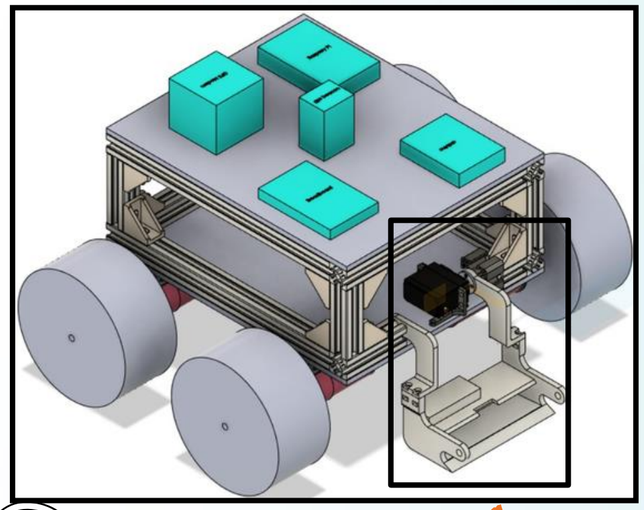
Distance	Motor Speed	Slug Size	Slug Orientation	Test #	Result
1.5 cm	35 RPM	M	1	1	✓
				2	✓
				3	✓
			2	4	✓
				5	✓
				6	✓
		S	1	7	X
				8	X
				9	X
			2	10	✓
				11	X
				12	X
	60 RPM	M	1	13	✓
				14	✓
				15	✓
			2	16	✓
				17	✓
				18	✓
		S	1	19	✓
				20	✓
				21	X
			2	22	X
				23	✓
				24	✓

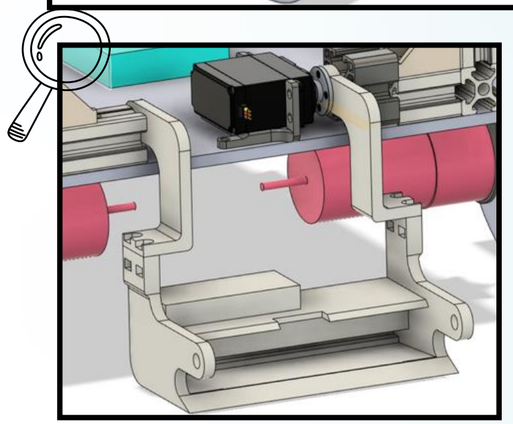


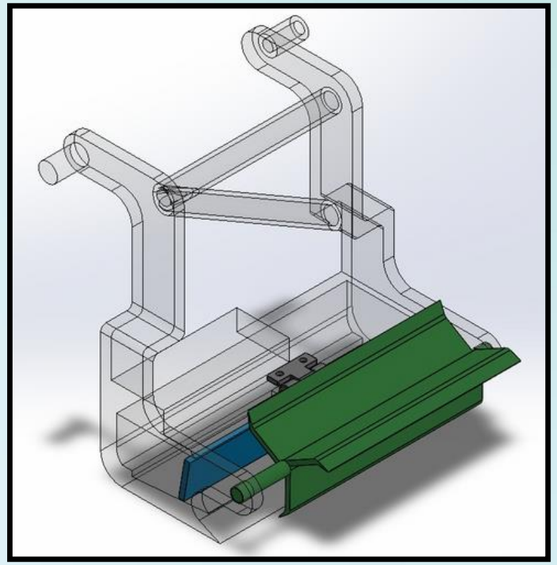
Distance	Motor Speed	Slug Size	Slug Orientation	Test #	Result
0.8 cm	35 RPM	M	1	1	✓
				2	✓
				3	✓
			2	4	✓
				5	✓
				6	✓
		S	1	7	✓
				8	✓
				9	✓
			2	10	✓
				11	✓
				12	✓
	60 RPM	M	1	13	✓
				14	✓
				15	✓
			2	16	✓
				17	✓
				18	X
		S	1	19	✓
				20	✓
				21	✓
			2	22	✓
				23	✓
				24	✓

Collection Mechanism Design:

Initial Designs → **CAD Design** → **Testing** → **Implementation**





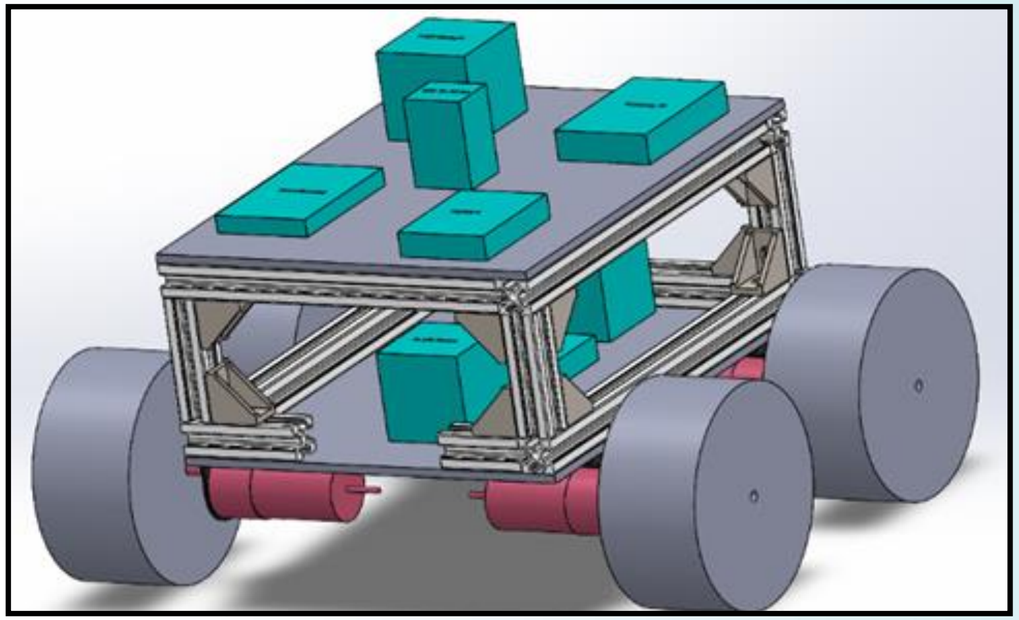


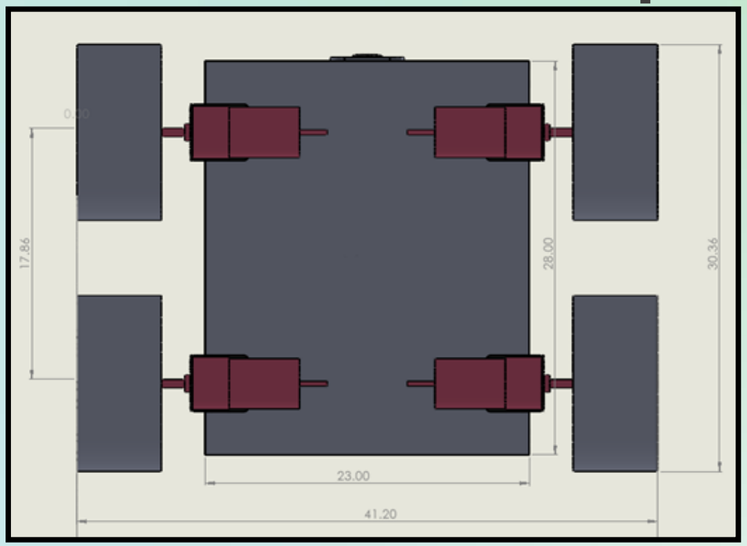
Drivetrain:

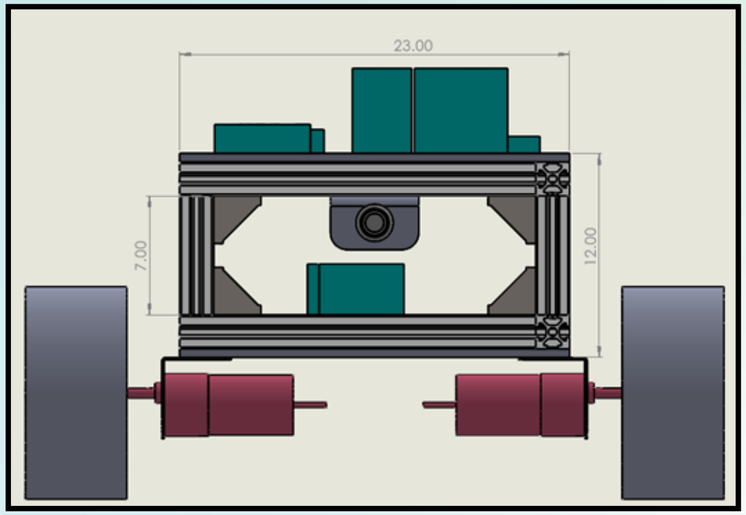
CAD Design

Implementation

4 WHEELED ROBOT STRUCTURE

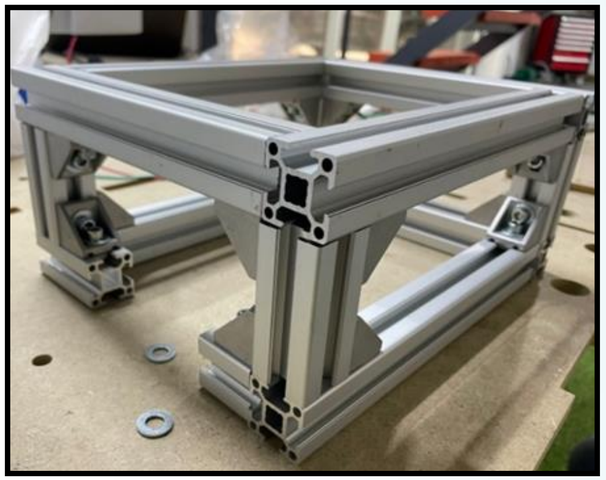




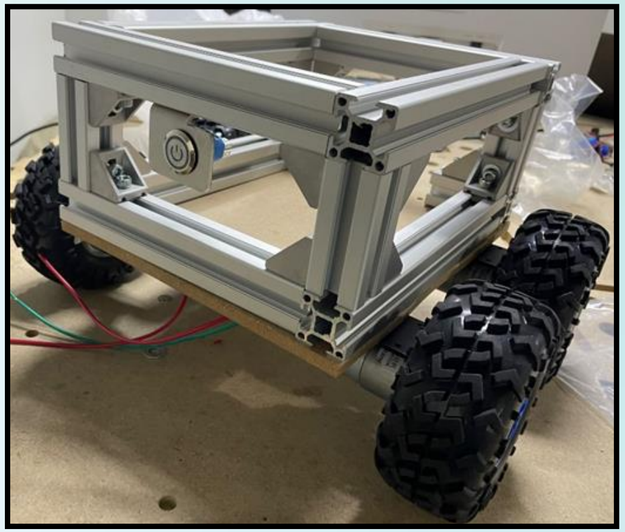


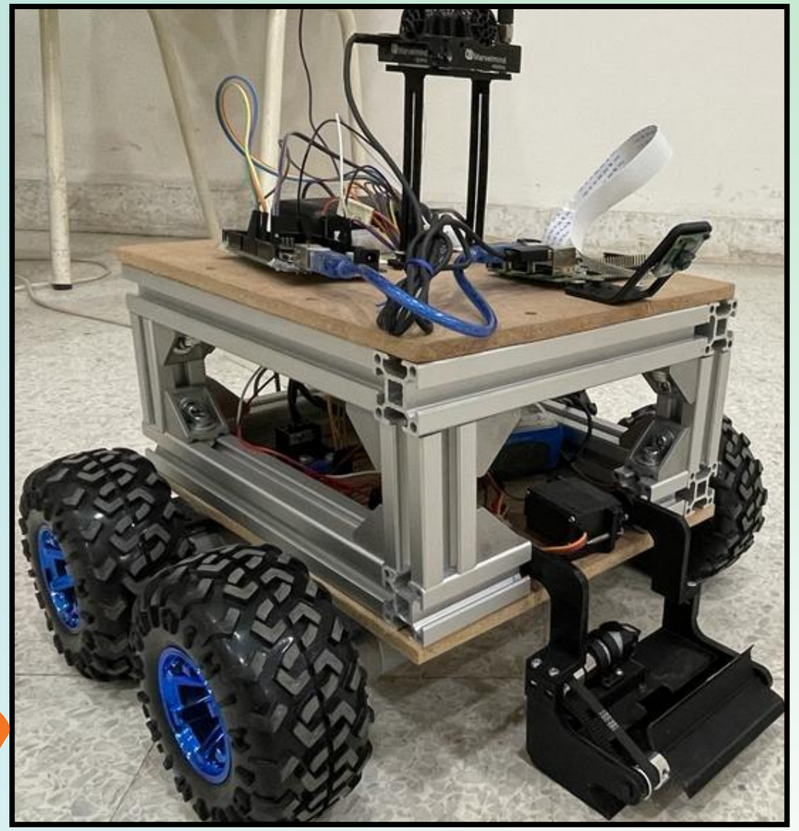
Drivetrain:

CAD Design ---------- **Implementation**



Initial Structure

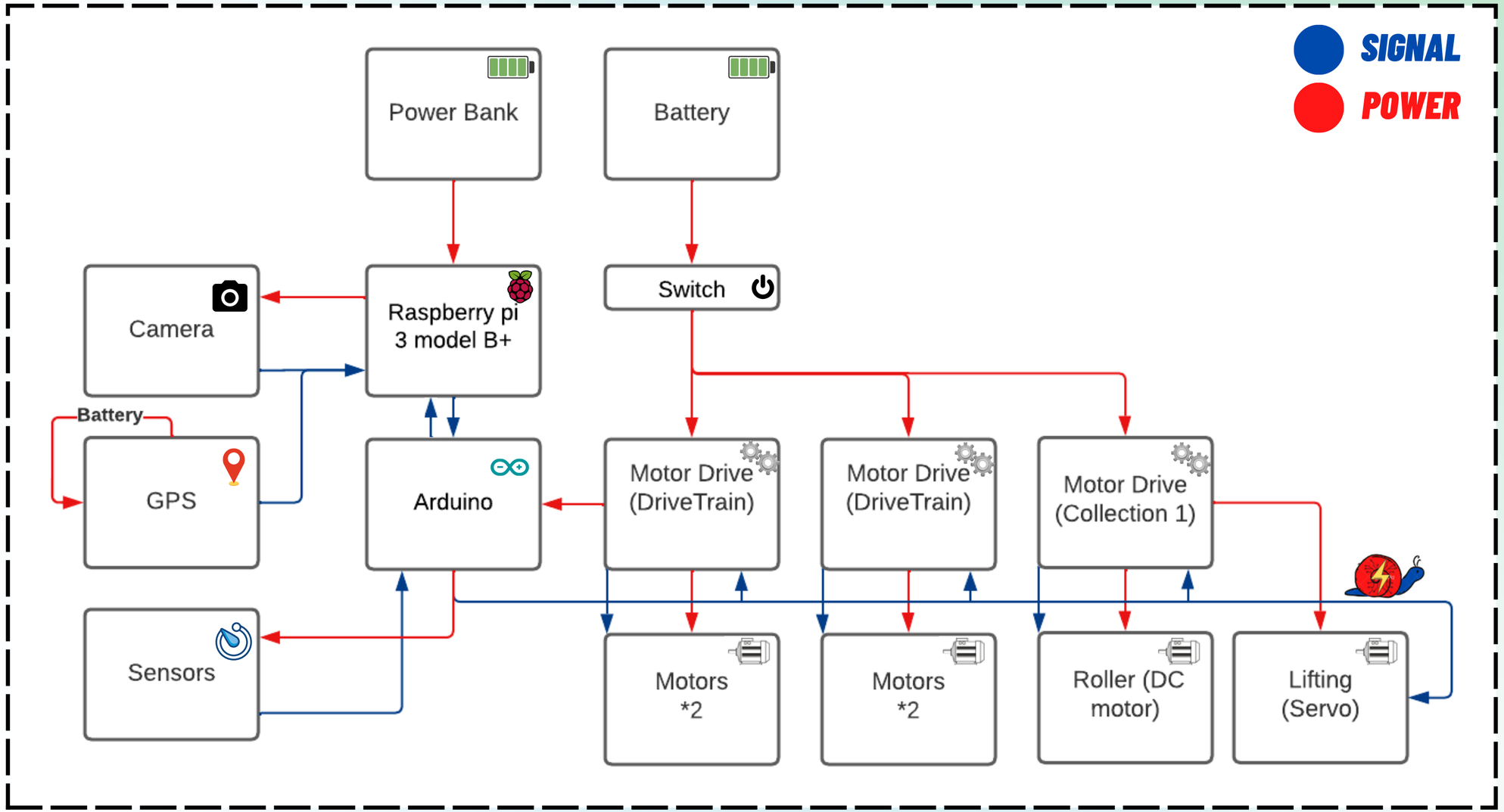




Final Structure

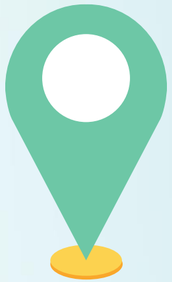
Drivetrain:

ELECTRONICS BLOCK DIAGRAM



PLAN:

Design



Vision

Navigation

Constraints & Future Work

Vision:

Data → **Training** → **Inference** → **Commands**

Slugs Dataset on Kaggle

989 Photos

Our Own Dataset

Collected Slugs

Took images around the RHU Campus

200 Photos

Labelling using LabelImg Program:

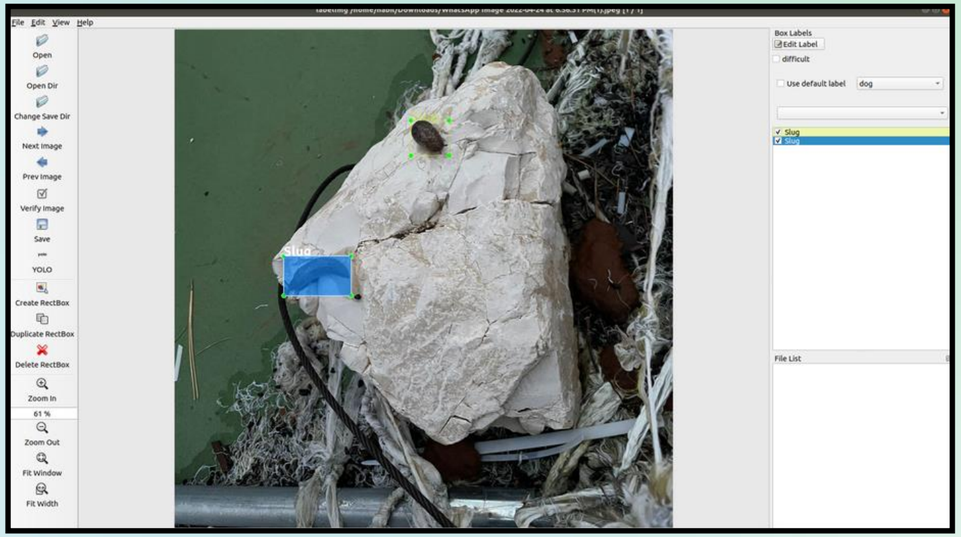
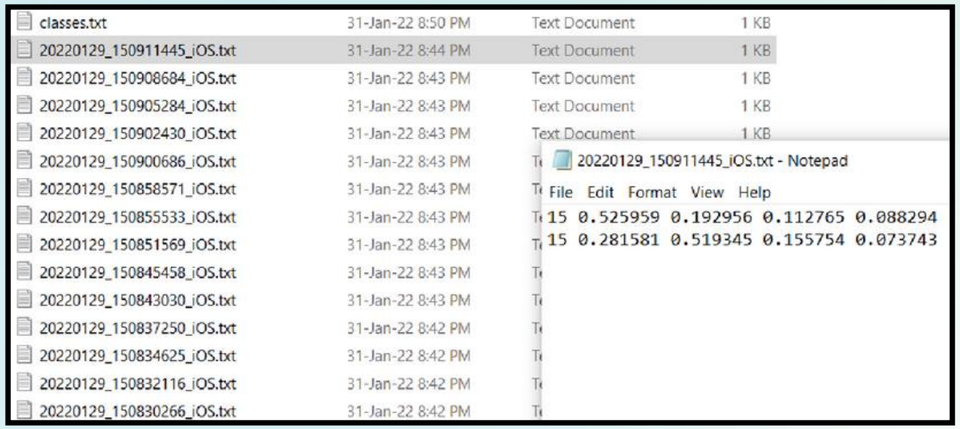


Image Labelling



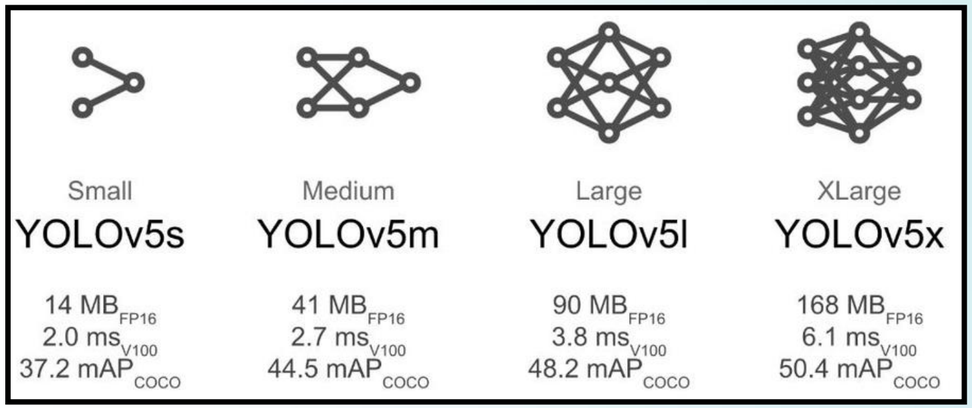
Output Files

Vision:

Data → **Training** → Inference → Commands

MODEL: YOLOV5 TRAINED ON CUSTOM DATASET

YOLOv5 is a model in the You Only Look Once (YOLO) family of computer vision models.
YOLOv5 is commonly used for object detection.

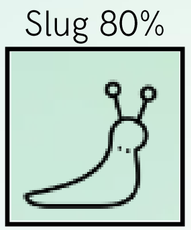


ultralytics
YOLOv5

Train-Validation-Test Split:
[Images]
Train: 134
Validate: 50
Test: 16

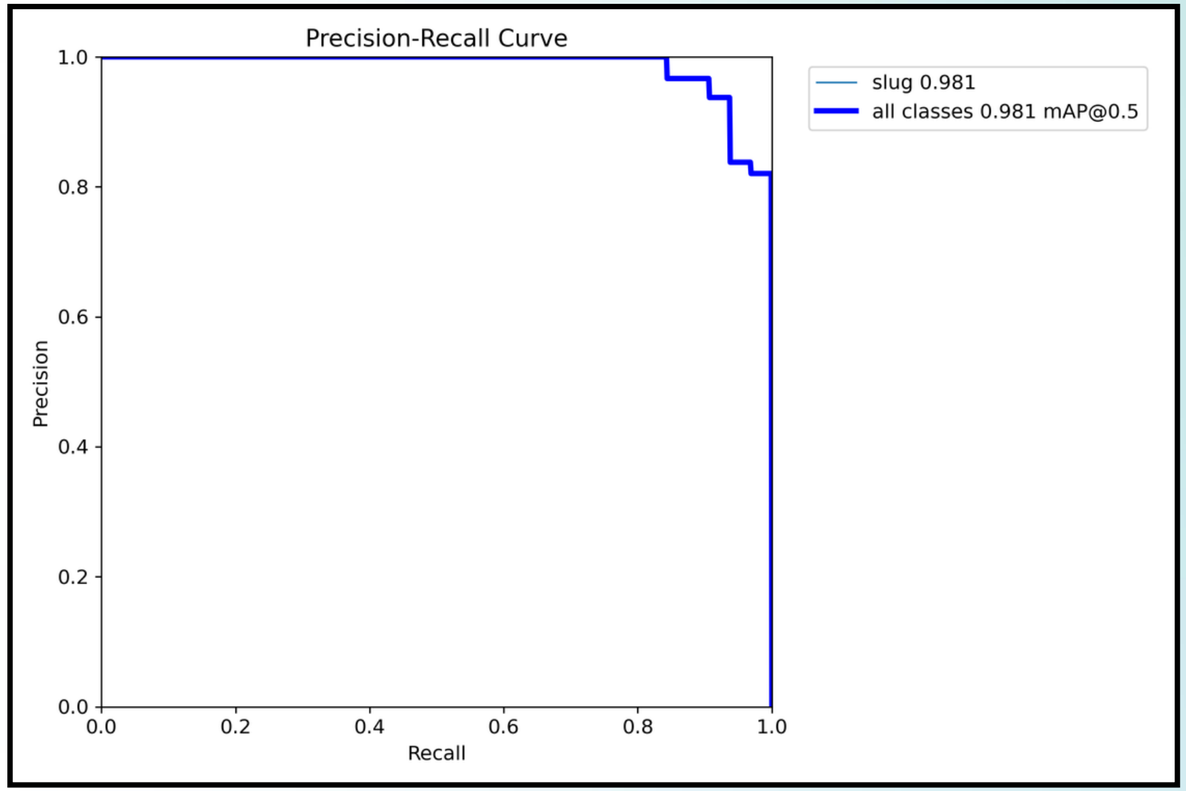
Model Params:
--epochs 270
--img 416
--weights yolov5s.pt

Output:



Vision:

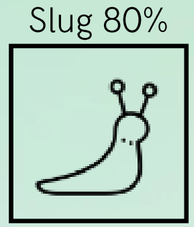
Data → **Training** → **Inference** → **Commands**



Dataset	Precision	Recall	mAP50	mAP50-95
Training	0.938	0.969	0.976	0.576
Validation	0.936	0.969	0.978	0.579

Model Metrics

Output:



Yolov5 Detection Sample:

Vision:

Data → **Training** → **Inference** → **Commands**

Raspberry Pi Camera Module V2.1

Raspberry Pi 3

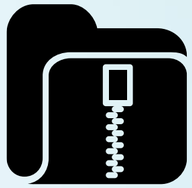
ROS

Raspicam_node Package

```
ros@ros-desktop:~$ roslaunch turtlebot3_bringup
turtlebot3_rpicamera.launch
```

Laptop

HOW TO INTEGRATE THE MODEL INTO THE ROBOTIC SYSTEM?



image_transport Command

Inference

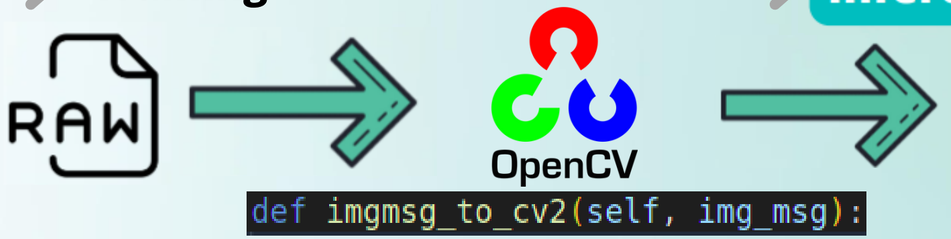
Model Inference

```
nabil@nabil-Ideapad:~$ rosrun image_transport
republish compressed in:=raspicam_node/image
raw out:=image_raw_
```

```
nabil@nabil-Ideapad:~/vision_ws/src/pipelinev5/
src/yolov5$ python3 cam_inference_ROS_Servo.py
```

Vision:

Data → **Training** → **Inference** → **Commands**



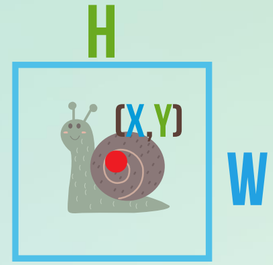

```
def imgmsg_to_cv2(self, img_msg):
```

Model Inference

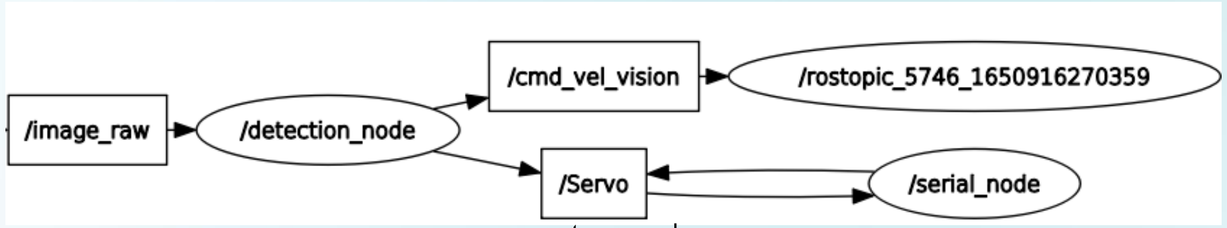
Prediction Result:

X_CENTER	Y_CENTER	W	H
[[461	353	508	381
0.62211	15]]		

PREDICTION CLASS



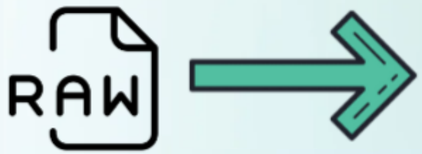
Sample



rqt_graph

Vision:

Data → **Training** → **Inference** → **Commands**




```
def imgmsg_to_cv2(self, img_msg):
```

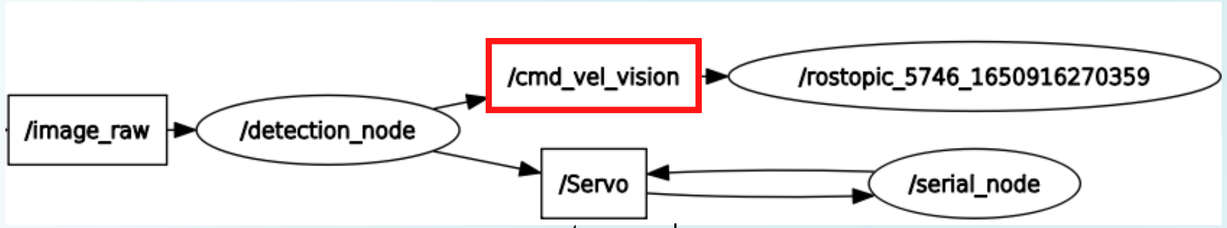
Model Inference

Prediction Result:

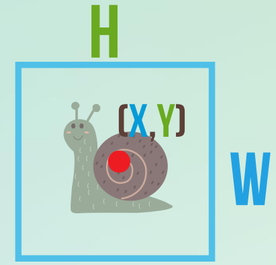
X_CENTER	Y_CENTER	W	H
[[461	353	508	381
0.62211	15]]		

PREDICTION CLASS

DETECTION TO VELOCITY ?



rqt_graph

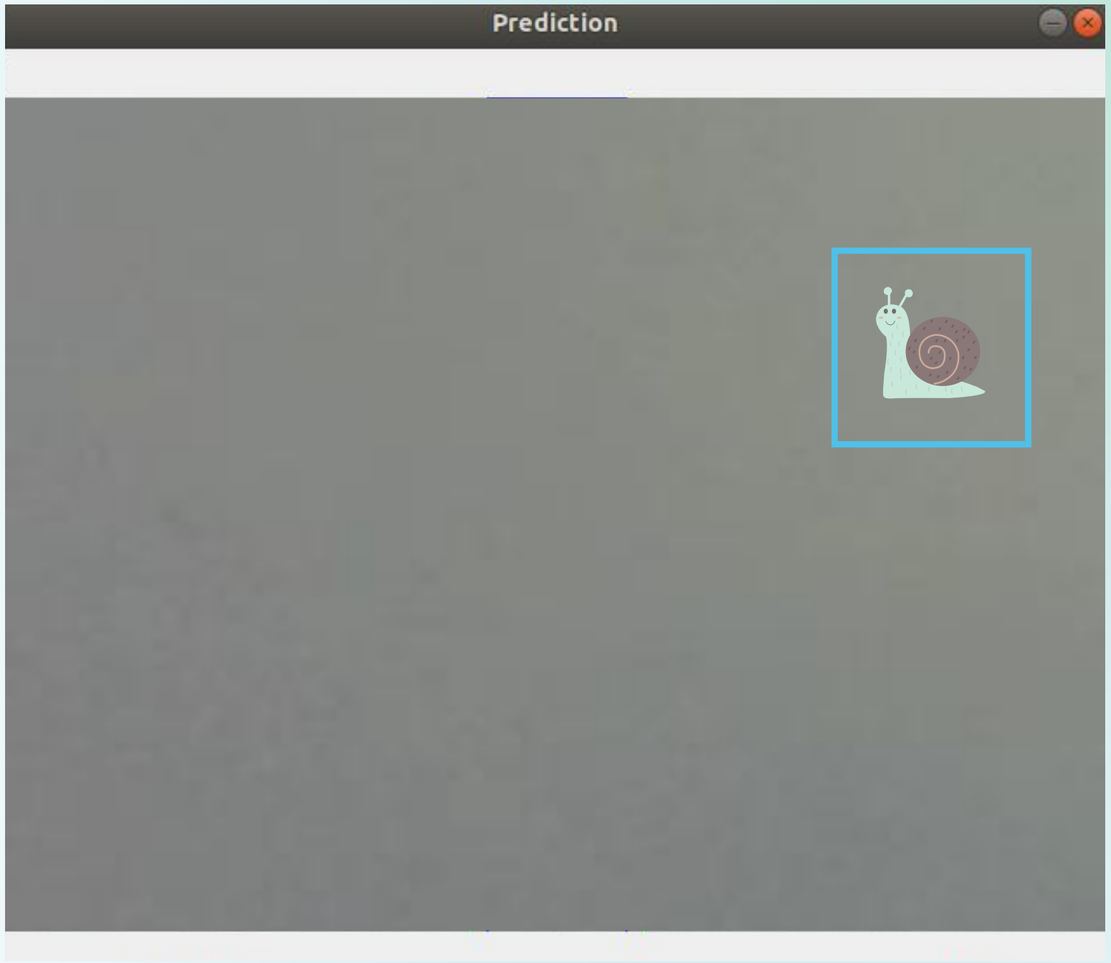


Sample

Vision:

Data → **Training** → **Inference** → **Commands**

DEPTH INFO ?

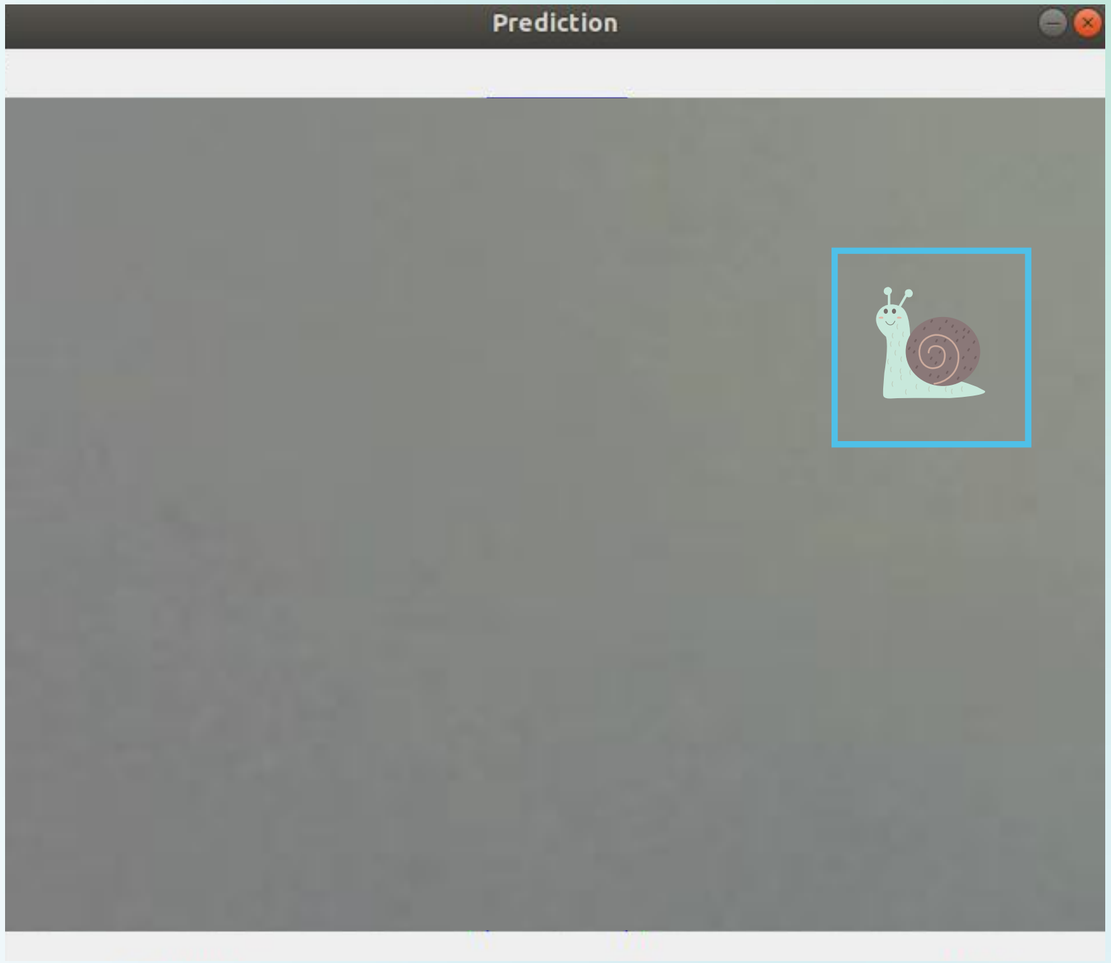


Vision:

Data → **Training** → **Inference** → **Commands**

DEPTH INFO ?

ROI

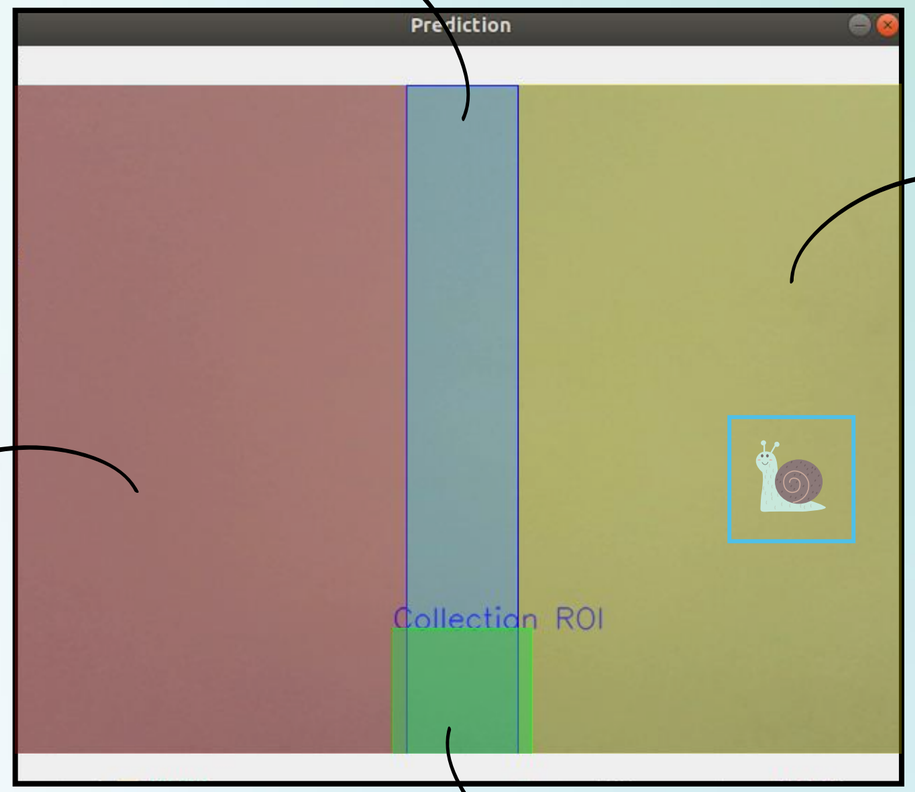


Vision:

Data → **Training** → **Inference** → **Commands**

Move Forward
twist_object.linear.x = 1

Forward



Right

Turn Right
twist_object.angular.z = -0.3

Left

Turn Left
twist_object.angular.z = 0.3

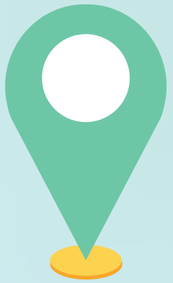
Collect

Just Stop
twist_object.linear.x = 0
twist_object.angular.z = 0
servo_pub.publish(true)

PLAN:

Design

Vision



Navigation

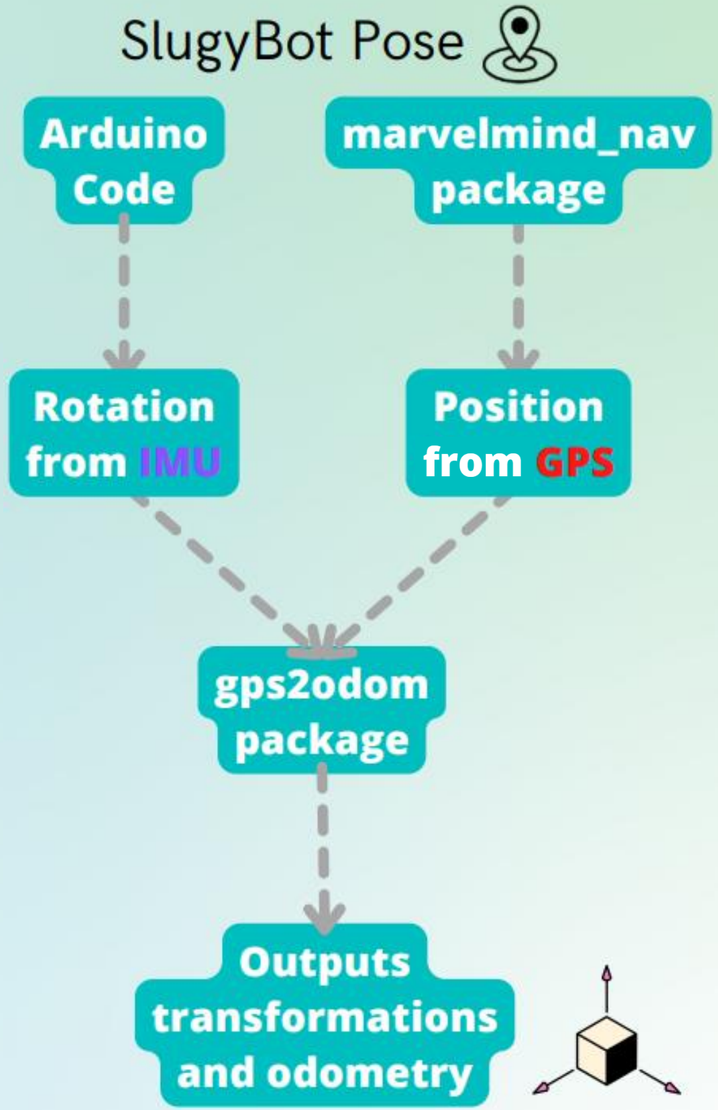
Constraints & Future Work

Navigation:

Marvelmind Indoor GPS Kit

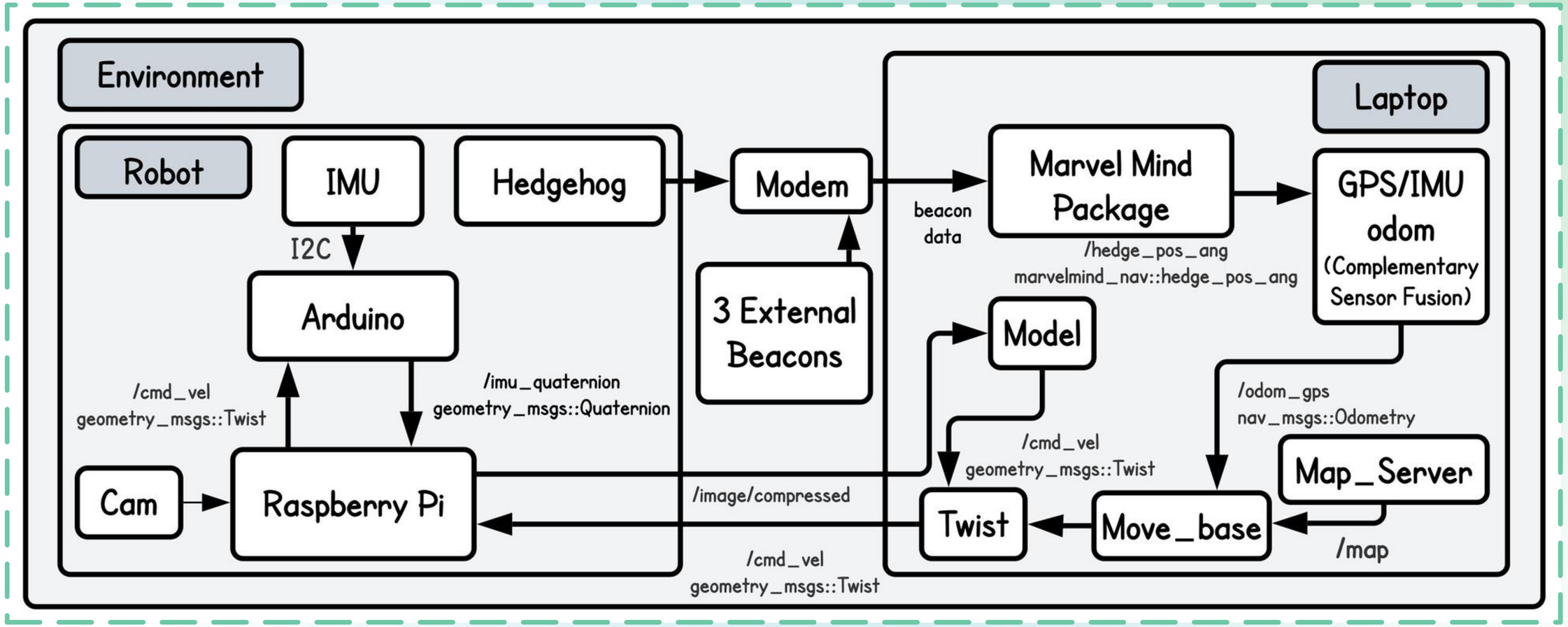
3 External Beacons
1 Hedgehog
1 Modem

MPU-6050



Navigation:

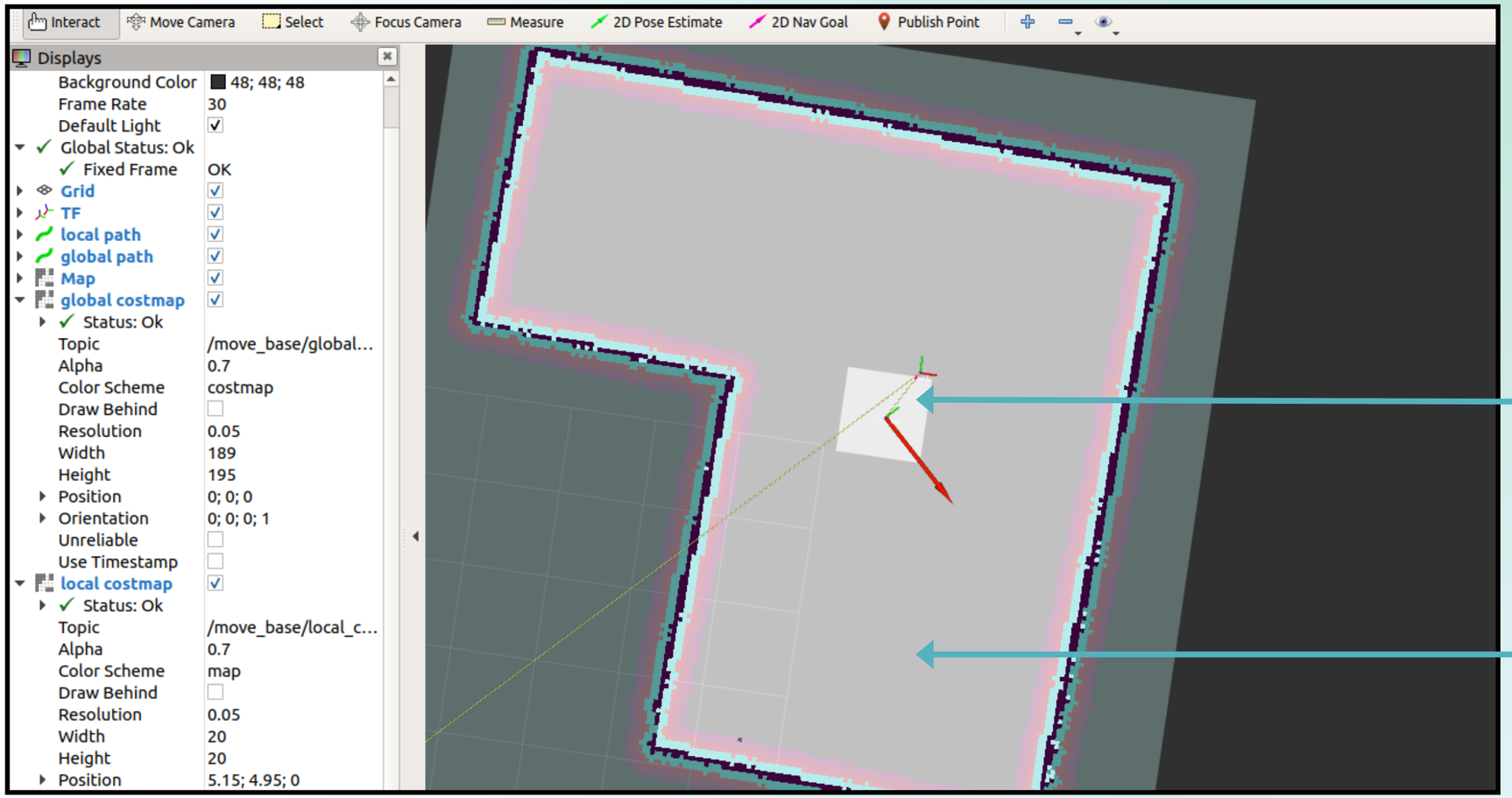
ROBOT NAVIGATION SCHEMATIC:



Navigation:

SPR_NAVIGATION PACKAGE OUTPUT:

Global and Local Costmaps



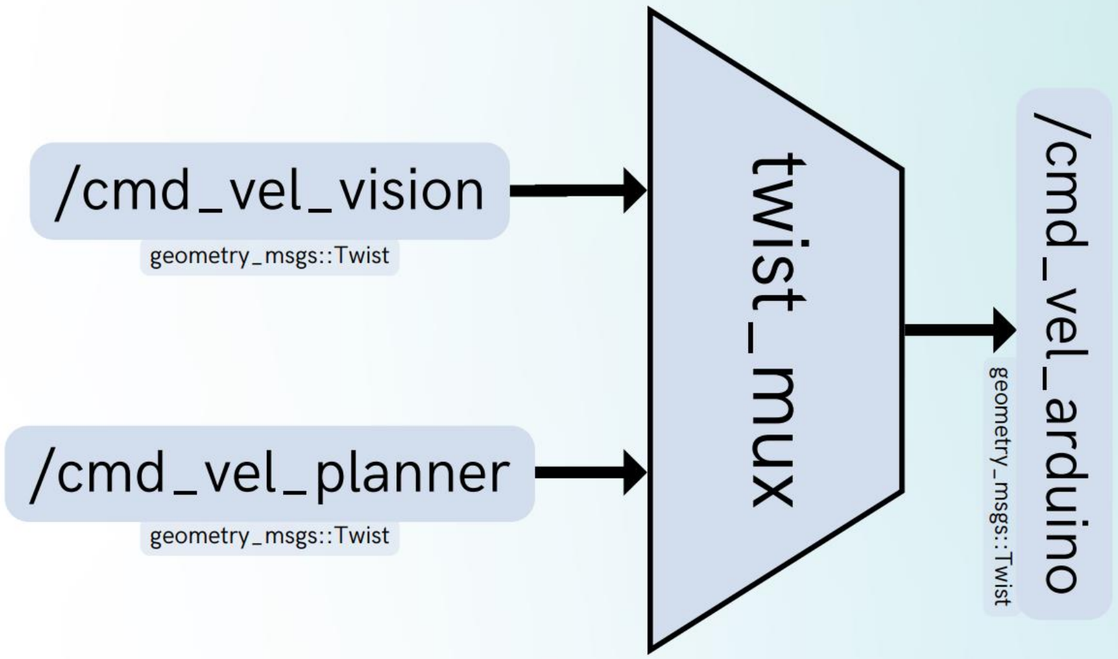
Local Costmap
[DWA Planner]

Global Costmap
[Navfn Planner]

Navigation:

TWIST_MUX PACKAGE:

twist_mux working principle:



Command velocities' priorities:

```
topics:
-
 name      :planner
 topic     :cmd_vel_planner
 timeout   :0.5
 priority  :1
-
 name      :vision
 topic     :cmd_vel_vision
 timeout   :0.5
 priority  :2
```

cmd_vel_vision is of higher priority

PLAN:

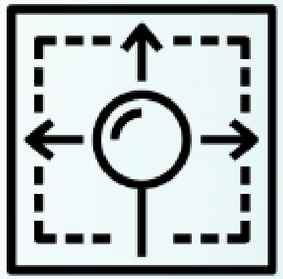
Design

Vision

Navigation

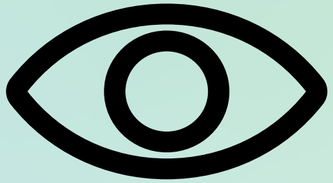
Constraints & Future Work

Constraints:



Operational area:

- **Flat Surface**
- **Few obstacles**

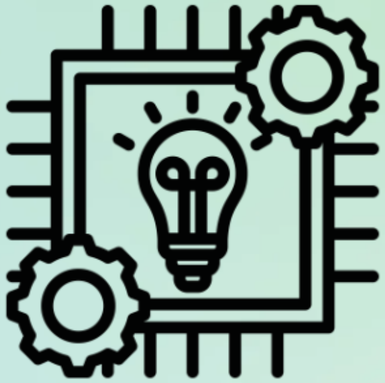


Vision System:

- **Camera Type**
- **Model training**
- **Data**

- **Material Used**
- **Size of robot**

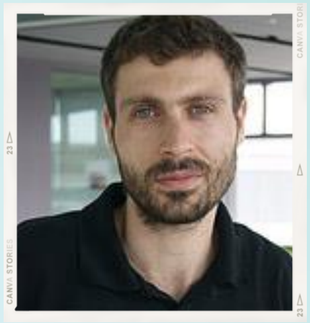
Future Work:



- **Centralized computation with a Friendly UI**
- **Develop algorithm to send goals without Rviz**
- **Better Camera, more data, advanced models**
- **Integrate Dynamic Obstacle avoidance**
- **Collection Mechanism Material**
- **Use Outdoor GPS Toolkit**

OUR TEAM

SUPERVISOR

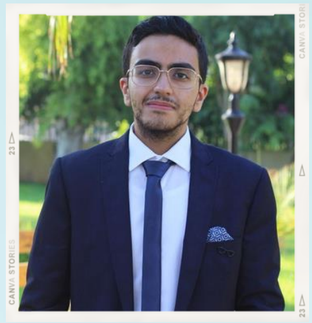


Prof. HASSAN HARIRI

STUDENTS

NABIL MIRI

JANA MARZOUK



ABDULRAHIM EL MOHAMAD

YAHYA AL JAMAL

Thank you

Navigation:

TWIST_MUX PACKAGE:

```
nabil@nabil-Ideapad:~/vision_ws/src/pipelinev5/src/yolov5$ rostopic
 echo /cmd_vel_vision
linear:
  x: 0.0
  y: 0.0
  z: 0.0
angular:
  x: 0.0
  y: 0.0
  z: 0.0
---
```

```
nabil@nabil-Ideapad:~/vision_ws/src/pipelinev5/src/yolov
5$ rostopic echo /cmd_vel_arduino
linear:
  x: 0.057
  y: 0.0
  z: 0.0
angular:
  x: 0.0
  y: 0.0
  z: 0.0
---
```

```
nabil@nabil-Ideapad:~/vision_ws/src/pipelinev5/src/yolov5$ rostopic echo /c
md_vel_planner
linear:
  x: 0.057
  y: 0.0
  z: 0.0
angular:
  x: 0.0
  y: 0.0
  z: 0.0
---
```

Vision ⟶ Twist ⟵ Planner

```
abil@nabil-Ideapad:~/vision_ws/src/pipelinev5/src/yolov5$ rostopic
 echo /cmd_vel_vision
linear:
  x: 0.0
  y: 0.0
  z: 0.0
angular:
  x: 0.0
  y: 0.0
  z: -0.2
---
```

```
nabil@nabil-Ideapad:~/vision_ws/src/pipelinev5/src/yolov
5$ rostopic echo /cmd_vel_arduino
linear:
  x: 0.0
  y: 0.0
  z: 0.0
angular:
  x: 0.0
  y: 0.0
  z: -0.2
---
```

```
nabil@nabil-Ideapad:~/vision_ws/src/pipelinev5/src/yolov5$ rostopic echo /c
md_vel_planner
linear:
  x: 0.057
  y: 0.0
  z: 0.0
angular:
  x: 0.0
  y: 0.0
  z: 0.0
---
```

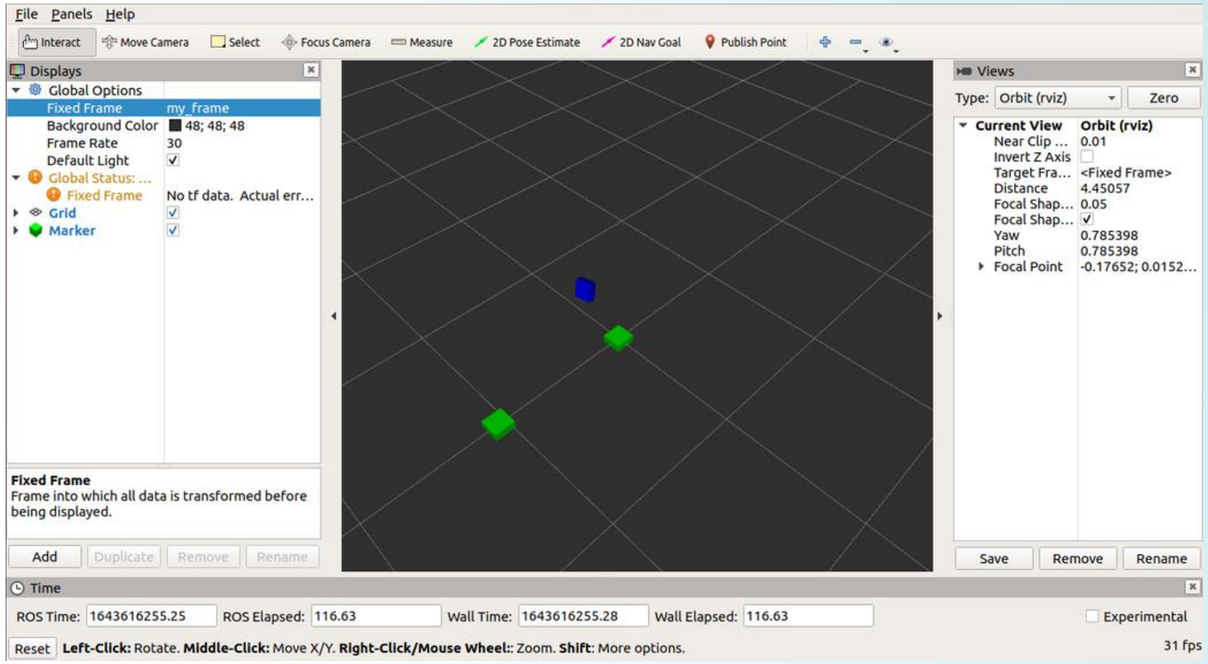
Extras

Navigation:

GPS PACKAGE: marvelmind_nav package

```
jana@jana-ThinkPad-T440p:~/gps_ws$ rosrun marvelmind_nav hedge_rcv_bin /dev/ttyACM0
Opened serial port /dev/ttyACM0 with baudrate 9600
[ INFO] [1643616045.799194770]: IMU fusion: Timestamp: 00169596, X=-0.078  Y= -0.554  Z=0.000  q=0.249,0.017,-0.007,-0.968 v=0.039,0.003,-0.00
8  a=-0.321,0.226,0.567
[ INFO] [1643616045.803776030]: Address: 4, timestamp: 169721, 169721, X=-0.078  Y= -0.554  Z=0.000  Angle: 0.0  flags=2
[ INFO] [1643616045.810574407]: IMU fusion: Timestamp: 00169721, X=-0.077  Y= -0.554  Z=0.000  q=0.249,0.021,-0.009,-0.968 v=0.038,0.003,-0.00
9  a=-0.016,0.011,0.597
[ INFO] [1643616045.819663004]: IMU fusion: Timestamp: 00169721, X=-0.077  Y= -0.554  Z=0.000  q=0.249,0.022,-0.005,-0.968 v=0.035,0.003,-0.00
9  a=-0.013,-0.059,0.537
[ INFO] [1643616045.828650927]: IMU fusion: Timestamp: 00169721, X=-0.077  Y= -0.554  Z=0.000  q=0.249,0.019,-0.006,-0.968 v=0.031,0.002,-0.00
9  a=-0.024,-0.013,0.534
```

Position received from GPS



Rviz showing 2 green boxes as the stationary beacons and 1 blue box as the hedgehog beacon on the robot

Navigation:

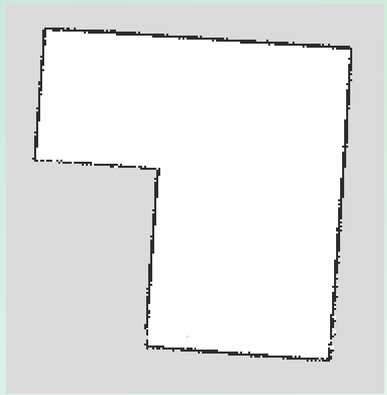
SPR_NAVIGATION PACKAGE:

spr_navigation launch file:

```
<launch>
  <arg name="map_file" default="$(find spr_navigation)/maps/room.yaml"/>

  <!-- Map server -->
  <node pkg="map_server" name="map_server" type="map_server" args="$(arg map_file)"/>

  <!-- move_base -->
  <include file="$(find spr_navigation)/launch/move_base.launch">
  </include>

</launch>
```

map_server

move_base

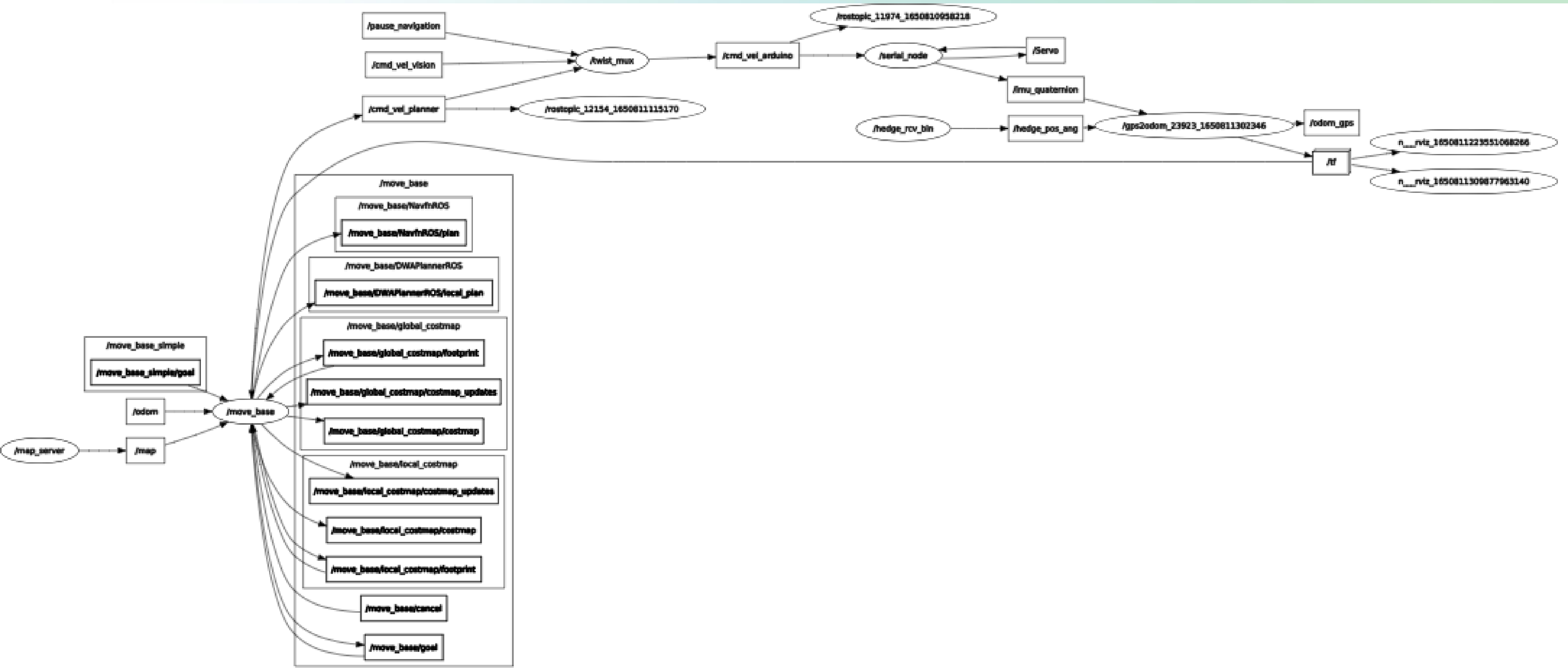


spr_navigation package DWA Planner:

```
DWAPlannerROS:
# Robot Configuration Parameters
  max_vel_x: 0.5
  min_vel_x: -0.5
  max_vel_y: 0.0
  min_vel_y: 0.0

# The velocity when robot is moving in a straight line
  max_vel_trans:  0.5
  min_vel_trans:  0.25
  max_vel_theta: 3
  min_vel_theta: 1.5
  acc_lim_x: 0.5
  acc_lim_y: 0.0
  acc_lim_theta: 3
```

```
# Goal Tolerance Parameters
  xy_goal_tolerance: 0.1
  yaw_goal_tolerance: 0.2
  latch_xy_goal_tolerance: false

# Forward Simulation Parameters
  sim_time: 1.5
  vx_samples: 20
  vy_samples: 0
  vth_samples: 40
  controller_frequency: 10.0
```

```
# Trajectory Scoring Parameters
  path_distance_bias: 5.0
  goal_distance_bias: 1.0
  occdist_scale: 0.02
  forward_point_distance: 0.325
  stop_time_buffer: 0.2
  scaling_speed: 0.25
  max_scaling_factor: 0.2

# Oscillation Prevention Parameters
  oscillation_reset_dist: 0.05
# Debugging
  publish_traj_pc : true
  publish_cost_grid_pc: true
```


/pause_navigation
/cmd_vel_vision
/cmd_vel_planner
/twist_mux
/rostopic_11974_1650810058218
/cmd_vel_arduino
/serial_node
/Servo
/imu_quaternion
/rostopic_12154_1650811115170
/hedge_rcv_bin
/hedge_pos_ang
/gps2odom_23923_1650811302346
/odom_gps
/tf
n__rviz_1650811223551068266
n__rviz_1650811309877963140
/move_base
/move_base/NavfnROS
/move_base/NavfnROS/plan
/move_base/DWAPlannerROS
/move_base/DWAPlannerROS/local_plan
/move_base/global_costmap
/move_base/global_costmap/footprint
/move_base/global_costmap/costmap_updates
/move_base/global_costmap/costmap
/move_base/local_costmap
/move_base/local_costmap/costmap_updates
/move_base/local_costmap/costmap
/move_base/local_costmap/footprint
/move_base/cancel
/move_base/goal
/move_base_simple
/move_base_simple/goal
/odom
/move_base
/map_server
/map

Our Solution

Specifications	Metric (Final Product)	Metric (POC)	Unit
Operation time (battery)	70	45	Minutes
Area of backyard (m^2)	3x4	3x4	m2
Weight without slugs	2	6	KG
L x W x H	15 x 22 x 10	28 x 23 x 31	inches /cm
Storage capacity	10 to 15	10	Slugs
Drivetrain	can navigate on muddy and rough terrains	can navigate in indoor spaces	
Automatic Charging	goes to charging station when the battery is low	-	

Low Noise	60 (daytime) and 45 (nighttime)	-	dB(A)
Weather resistance	IP68 - IK10	-	N/A
Types of Crops in the targeted backyard	Salads (Cabbage, Tomatoes, Pumpkins...etc.)	-	N/A
Navigation	100% of the field covered	-	%
Disposal	0% Slug Harm	0% Slug Harm	%

Weight and Motor Selection

INPUTS:

Without SF

With SF

Nominal Robot Speed: 0.75 [m/s]
Nominal Robot Acceleration: 0.25 [m/s^2]

Drive Wheel Diameter: 0.125 [m]
Friction Coefficient: 0.158

OUTPUTS:

Motor Torque: 3.59 [Kg.cm]
Motor Speed: 114.6 [RPM]

Motor Used:

- 120 rpm, torque 18kgcm, stall current 7A

- 100 rpm, torque 12kgcm, stall current 2A

Serial No.	Model	Rated Voltage	No Load		At Maximum Efficiency				Stall	
			Speed	Current	Speed	Torque	Current	Output	Torque	Current
		V	rpm	mA	rpm	Kgcm	A	W	Kgcm	A
37GB520	13490-50	24	180	50	145	2	0.22	3	10	1
	11750-490	24	12	45	11	10.0	0.1	1.2	40	0.7
	11750-270	12	11	25	9	6	0.1	0.5	30	0.4
	13490-70	12	60	50	48	2	0.15	1	10	0.6
	17320-70	12	96	100	80	2	0.32	1.6	12	2
	20260-30	12	280	100	230	1.5	0.6	3.5	8	2.6
	25150-50	6	152	170	125	1.5	0.85	1.9	8	3.5
	17410-90	12	60	50	48	4	0.32	2	18	1.5
	11750-50	24	125	45	100	2	0.15	2.1	10	0.64

Battery Sizing

45 min one cycle

Battery Sizing (50% of Maximum Current)		
Component	**Current in A**	**Explanation**
4 DC Motors	4	50% of Stall Current (2*4*50%)
Servo Motor (Lifting)	0.4	50% of Current at Max Load (0.8*50%)
DC Motor (Roller)	0.097	Continuous Current (A): 97mA
4 Motor Driver	0.06	
Capacity	3.4178	Ah: Sum of the Currents * 45 min (1 Cycle Operational Time)

Worst-Case

Power Bank Sizing		
Component	**Current in A**	**Explanation**
Raspberry Pi	2	
Arduino Mega	0.0732	
Capacity	1.5549	Ah: Sum of the Currents * 45 min (1 Cycle Operational Time)

12V 4400mAh 18650 Lithium-ion Battery

Battery Sizing (50% of Max Current)		
Component	**Current in A**	**Explanation**
4 DC Motors	3.6	Operates for 90% of the time
Servo Motor (Lifting)	0.04	Operates for 10% of the time
DC Motor (Roller)	0.034	Operates for 35% of the time
4 Motor Driver	0.06	
Capacity	2.8005	Ah: Sum of the Currents * 45 min (1 Cycle Operational Time)

Operational Time

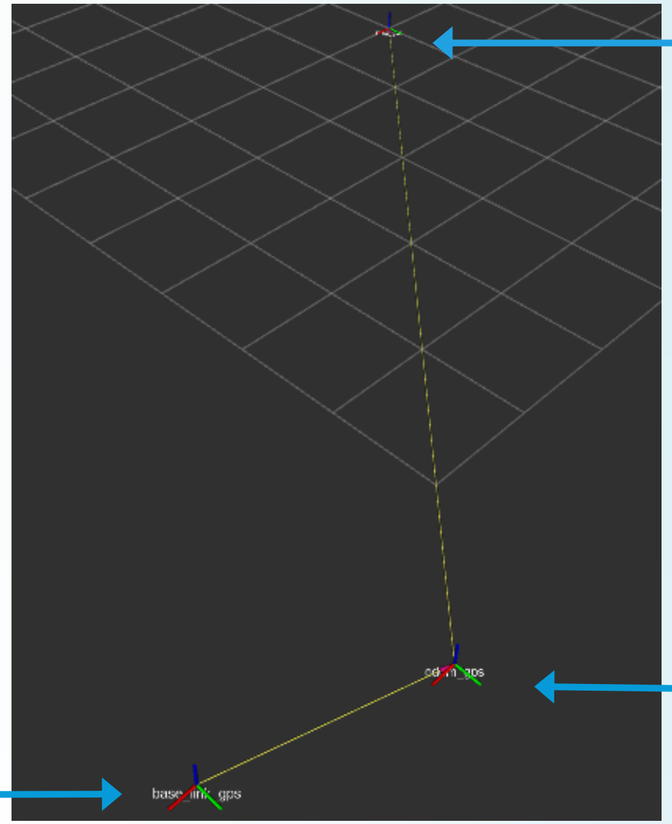
Navigation:

GPS2ODOM PACKAGE:

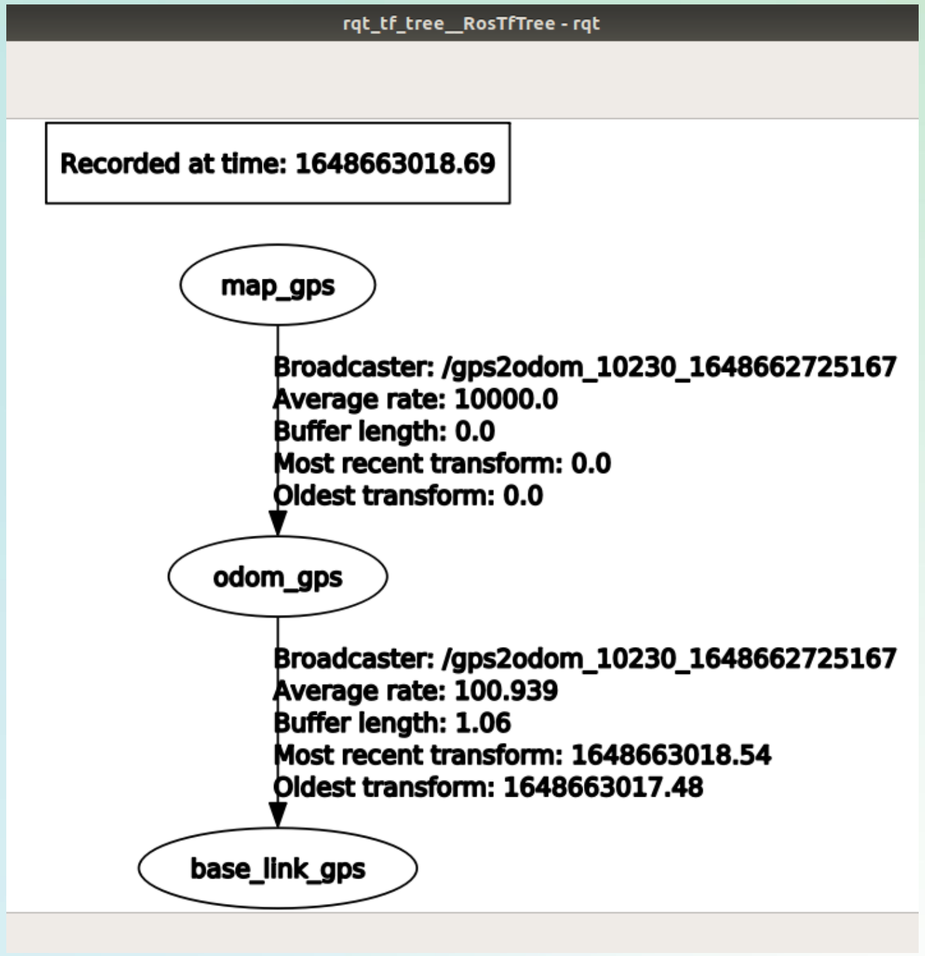
running the package

```
jana@jana-ThinkPad-T440p:~$ cd gps_ws
jana@jana-ThinkPad-T440p:~/gps_ws$ source devel/setup.bash
jana@jana-ThinkPad-T440p:~/gps_ws$ rosrun gps2odom gps_to_odom.py
```

Rviz showing frame-links



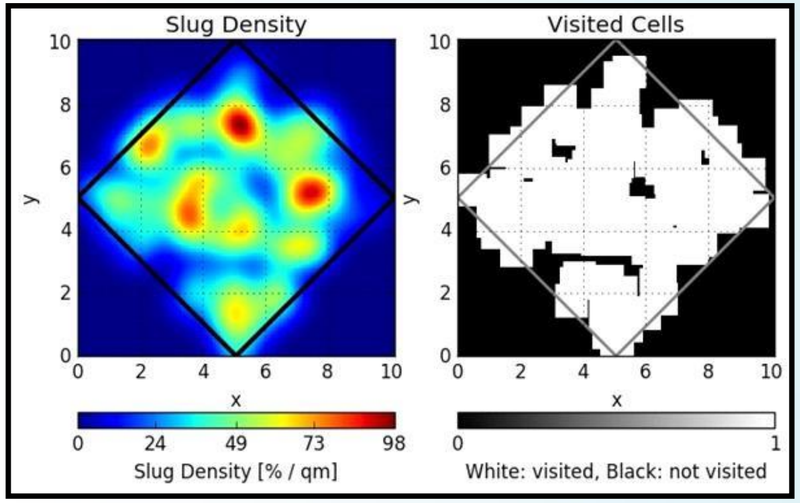
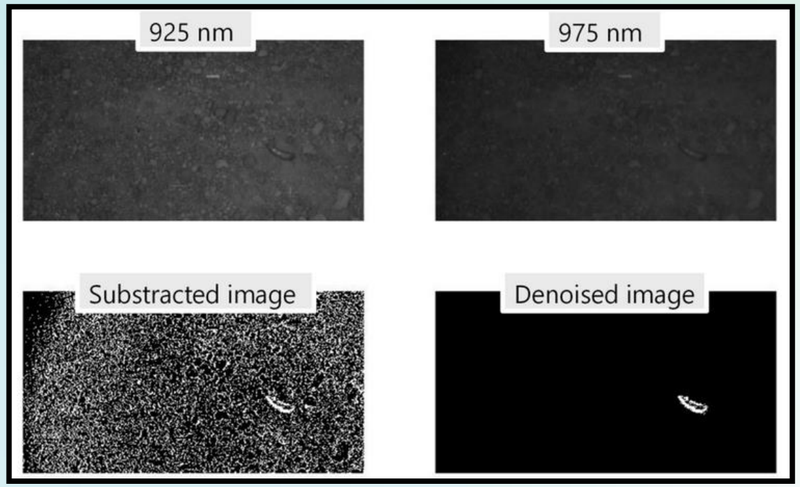
obtained frame-links



Smart Solutions:

UNIKASSEL VERSITÄT

- MSR-BOT-PROJECT (2016-2020)



- 2 Companies worked on this project
- Mainly for arable farming
- Uses Spectral Imaging
- Kills the detected slugs using nails
- Recognize slugs "Hotspots"