Autonomous UGV

Introduction

Automation

Unmanned Ground Vehicle (UGV)

Autonomous waypoint Guidance

Path planning

Obstacle avoidance

Problem

- Space missions
- Nuclear power plants
- Military
- Agriculture

Objectives

01 Waypoint navigation

02 Obstacle avoidance

03 Obstacle detection

04 GUI Designing

Development Phases

Phase 1
Research on autonomous vehicle technology.

Phase 2
State estimation and localization.

Phase 3
Waypoint navigation and controller design.

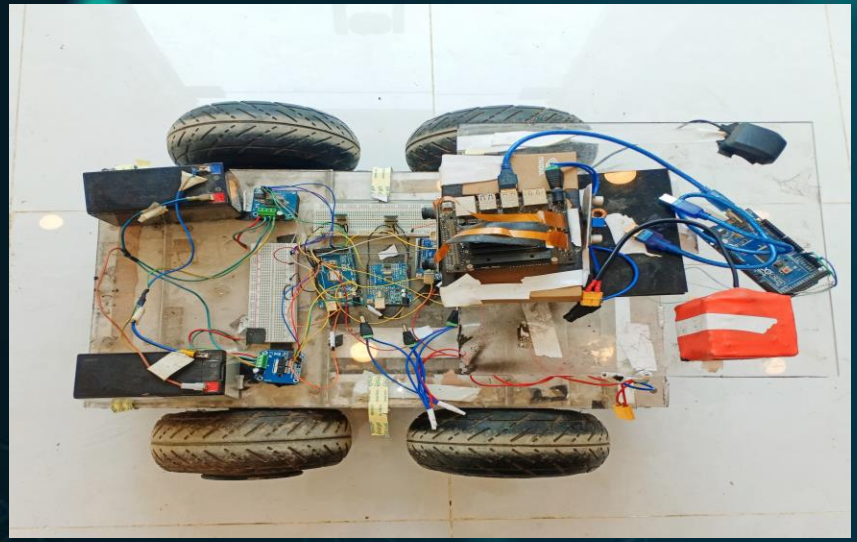
Phase 4
GUI design and ROS integration.

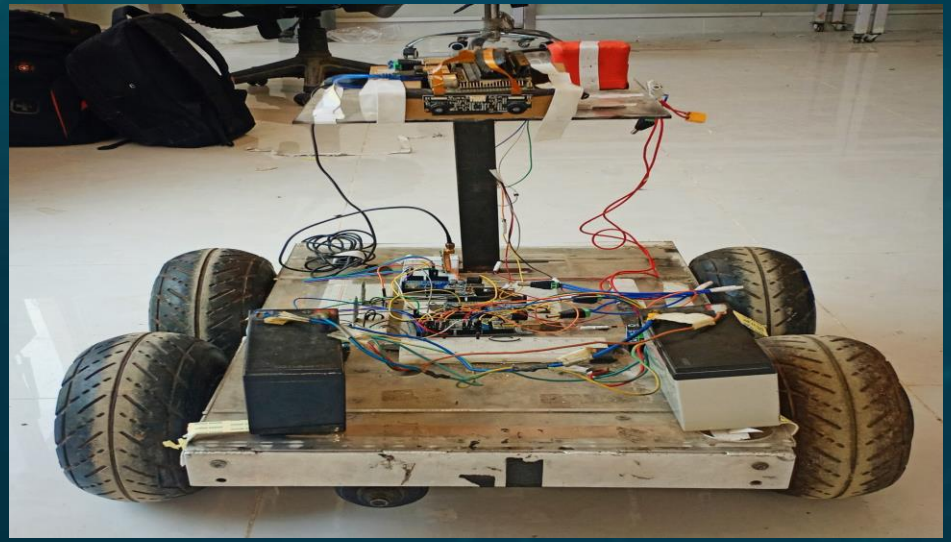
Phase 5
Object detection using deep learning.

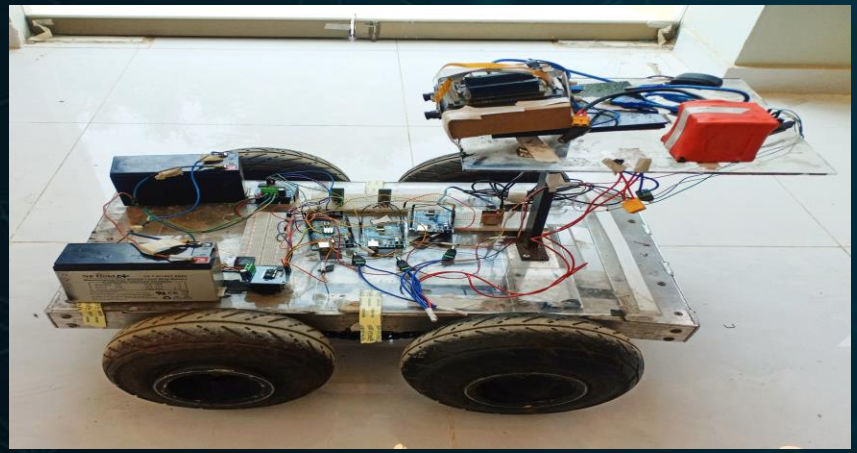
Phase 6
Optimization for real time inference on GPU.

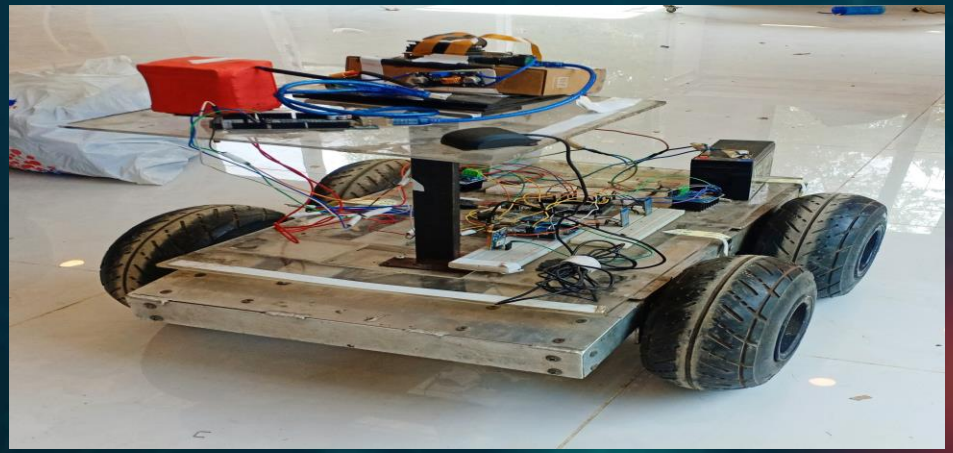
Block Diagram

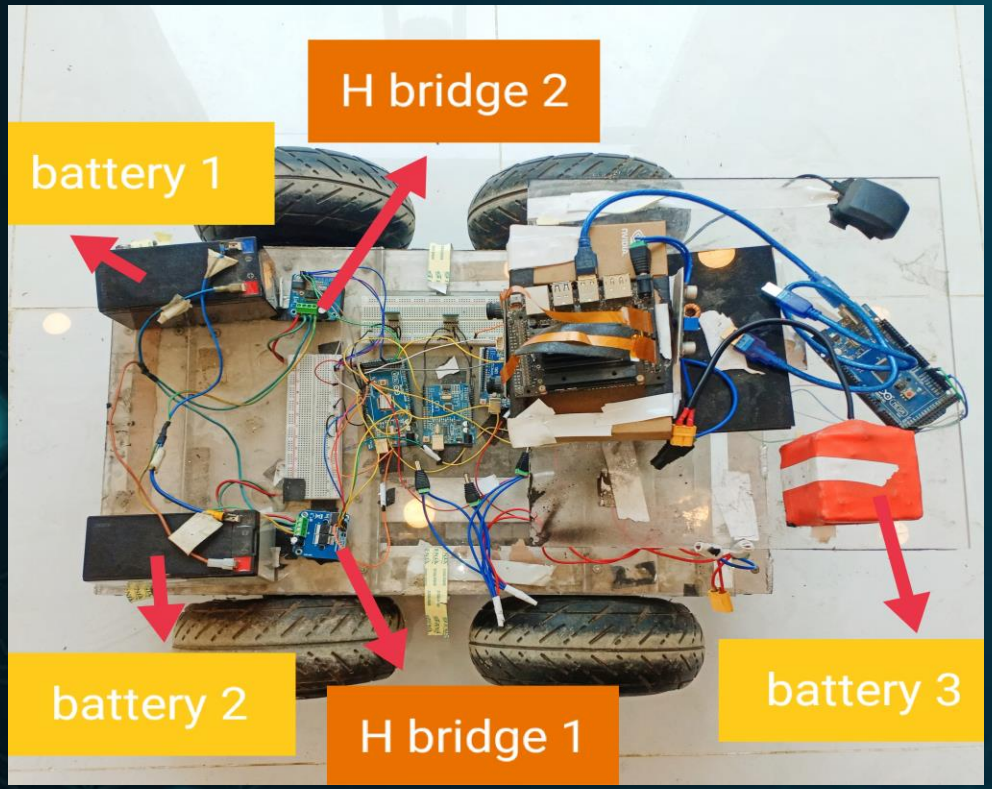
Hardware

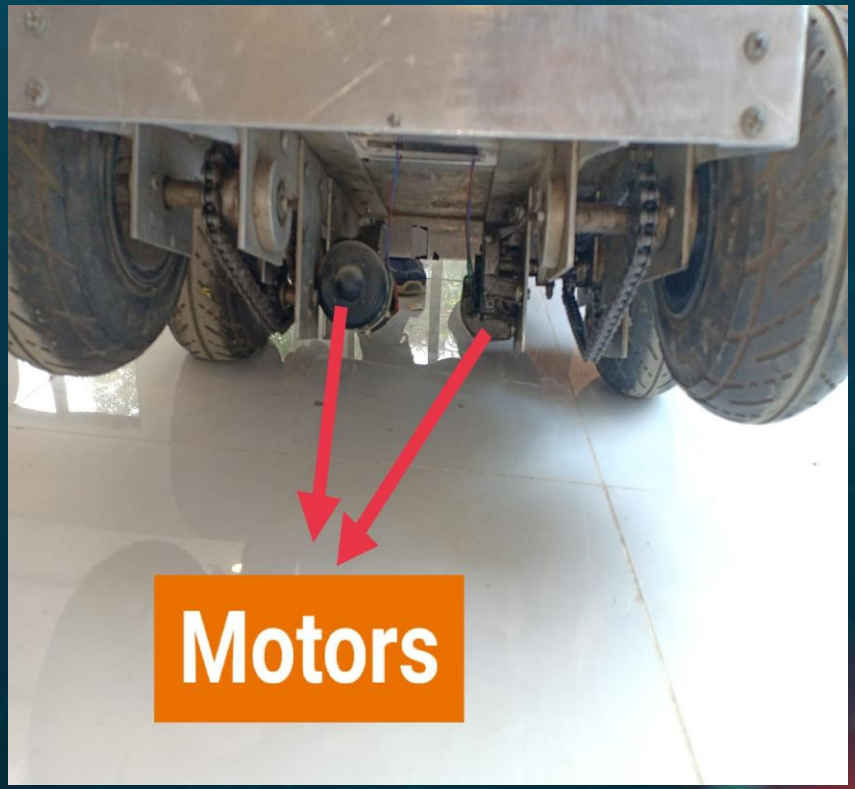


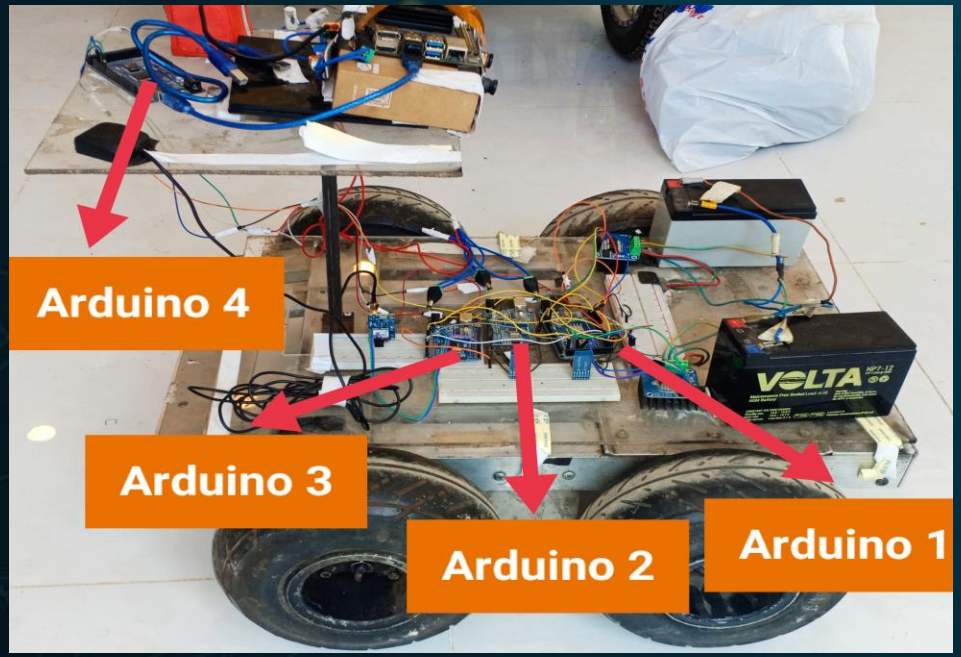


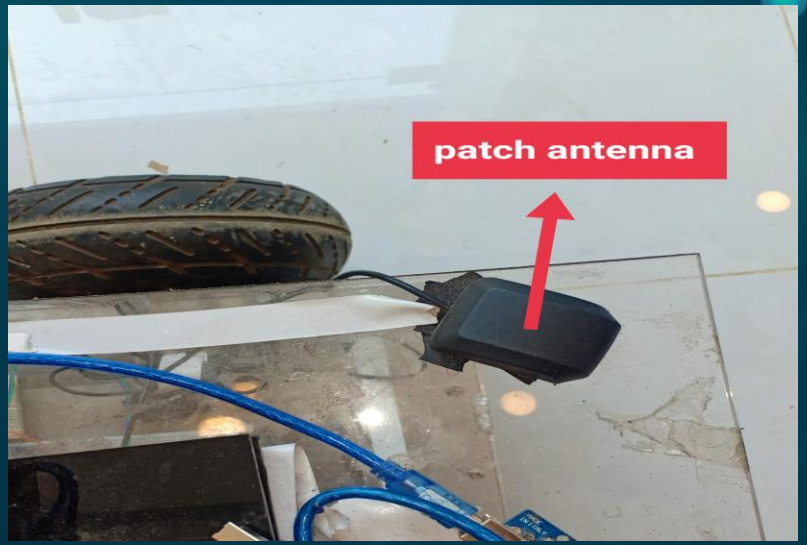




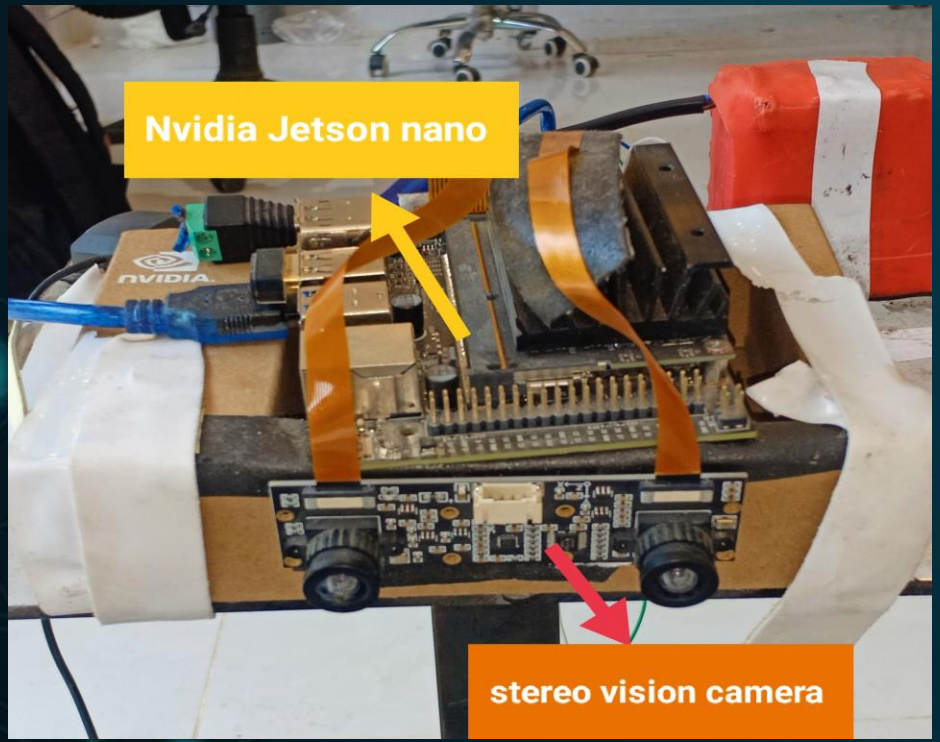

H bridge 2
battery 1
battery 2
H bridge 1
battery 3

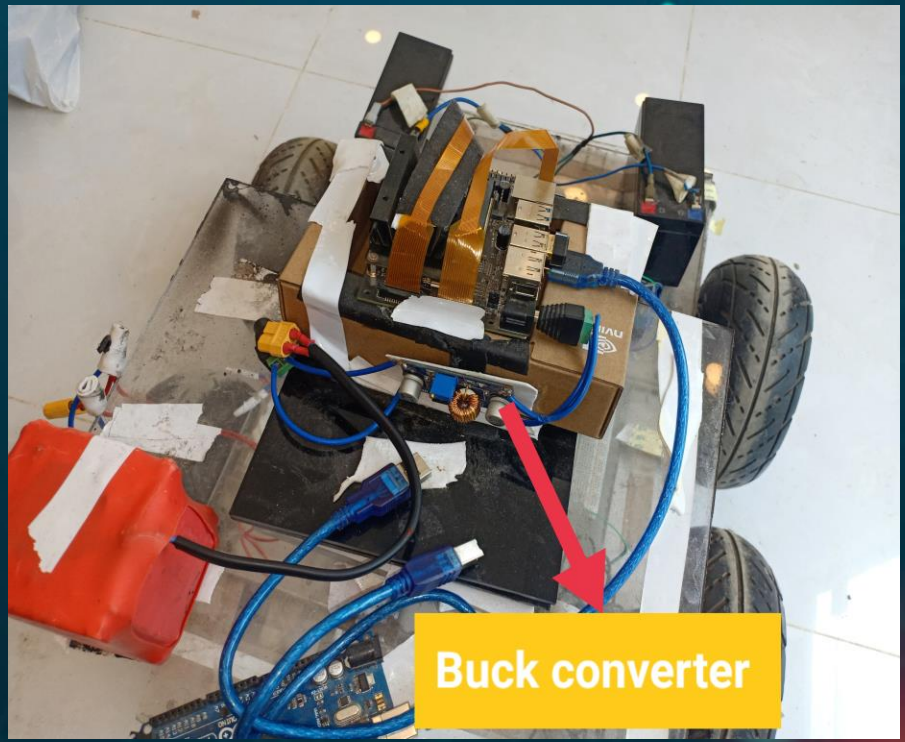

Motors


Arduino 4
Arduino 3
Arduino 2
Arduino 1
VOLTA

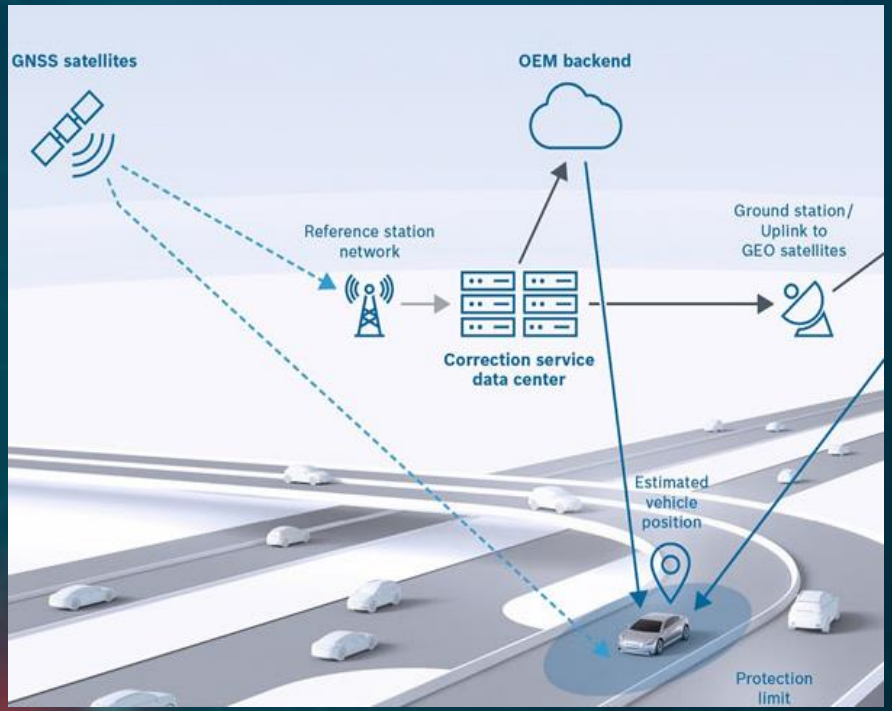

patch antenna


HC-05
NEOM8n


Nvidia Jetson nano
stereo vision camera


Buck converter

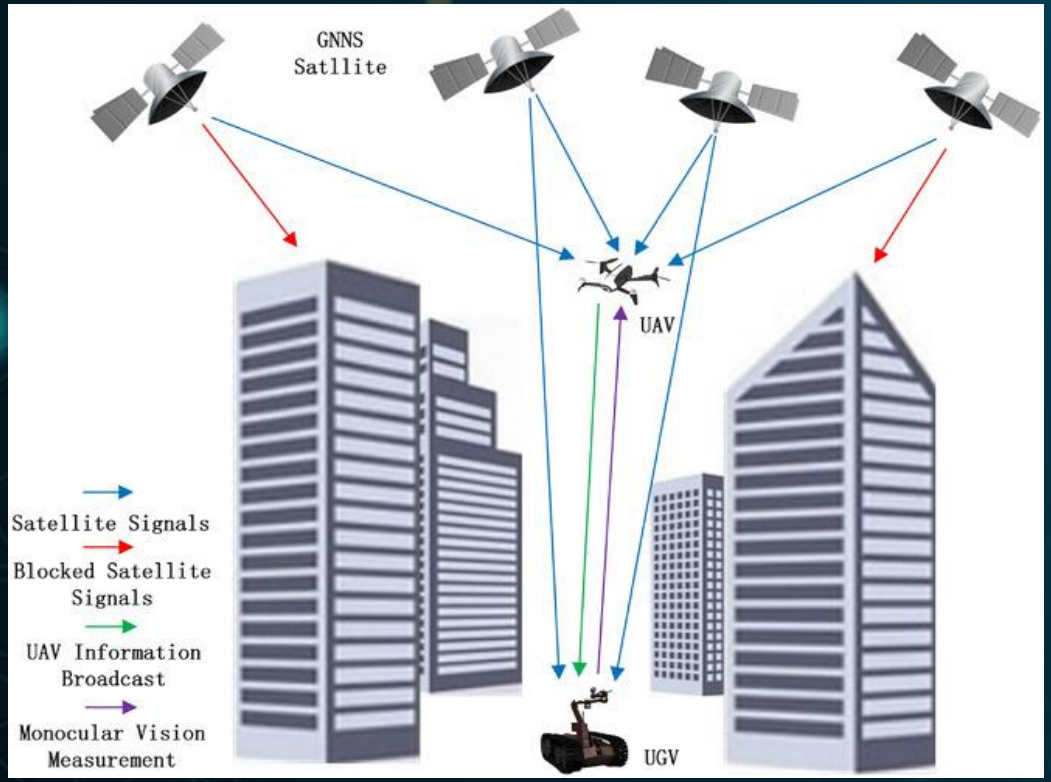
Software

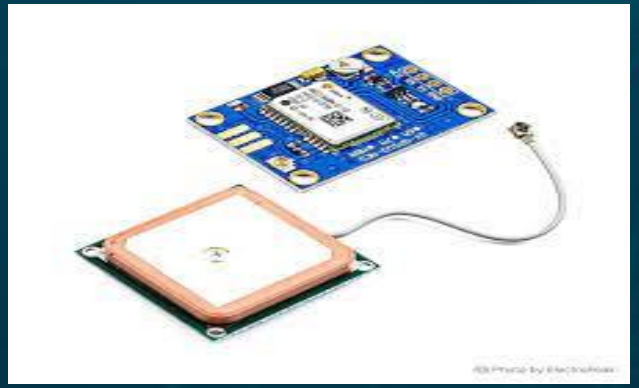


- Positioning System
- Heading Angle

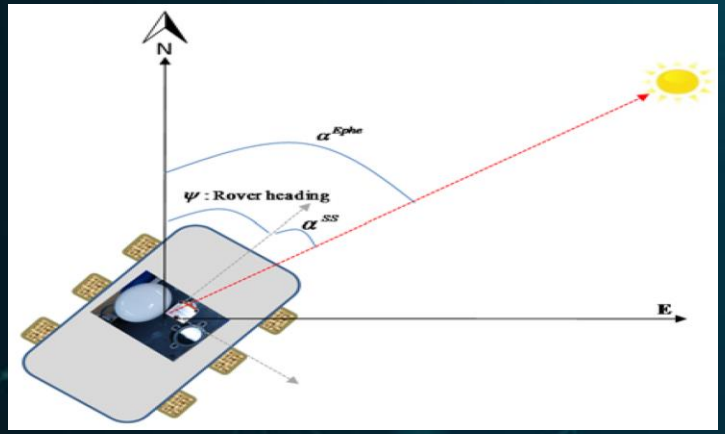
State Estimation and localization

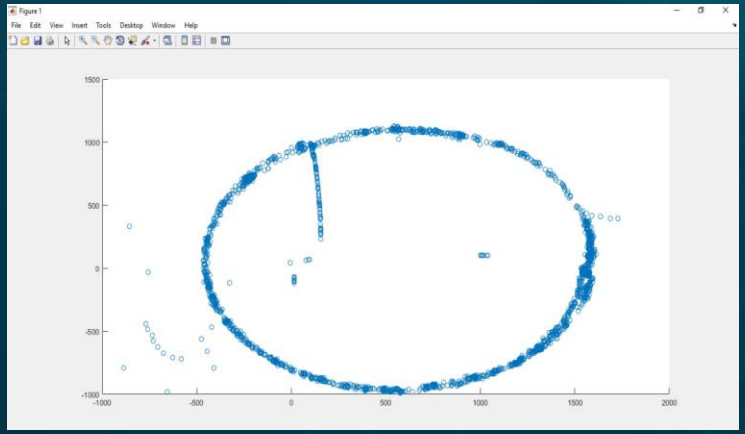
Positioning system

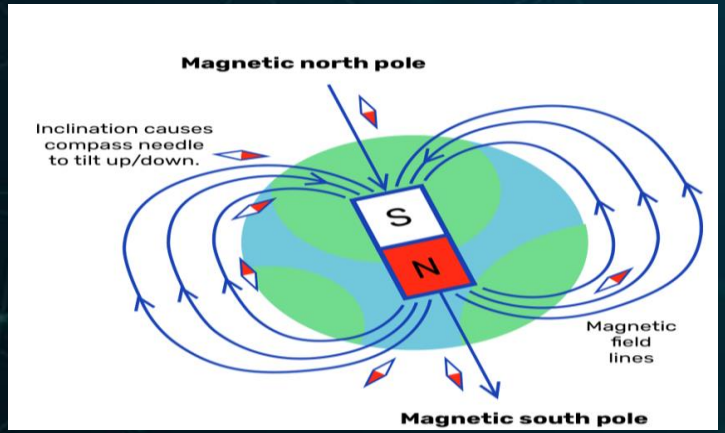


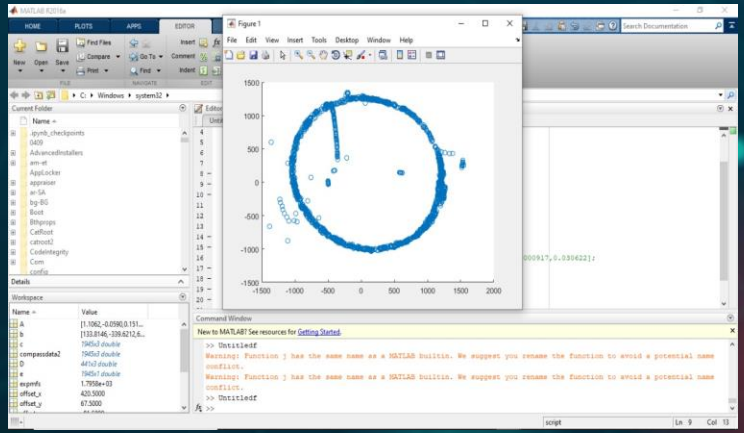


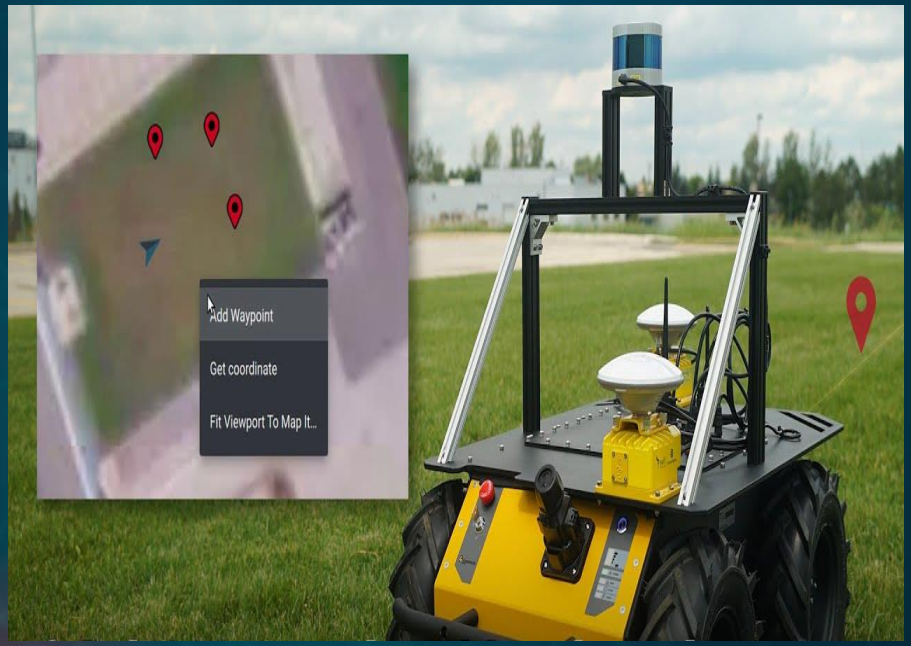
Heading Angle











- Distance
- Bearing Angle

Waypoint Navigation

Distance

$$\operatorname{haversin}\left(\frac{d}{r}\right) = \operatorname{haversin}(\phi_2 - \phi_1) + \cos(\phi_1)\cos(\phi_2)\operatorname{haversin}(\lambda_2 - \lambda_1)$$

a = sin²(ΔlatDifference/2) + cos(lat1).cos(lt2).sin²(ΔlonDifference/2)
c = 2.atan2(√a, √(1−a))
d = R.c

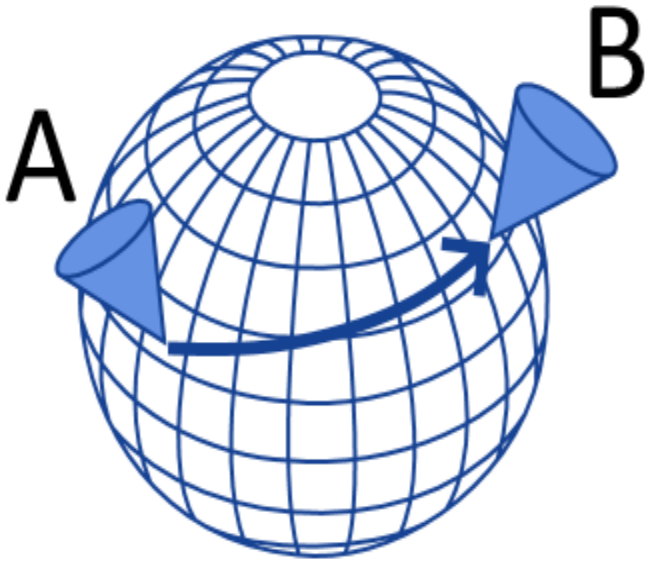
where,

ΔlatDifference = lat1 − lat2 (difference of latitude)

ΔlonDifference = lon1 − lon2 (difference of longitude)

R is radius of earth i.e **6371 KM or 3961 miles**

and **d** is the distance computed between two points.



Bearing Angle

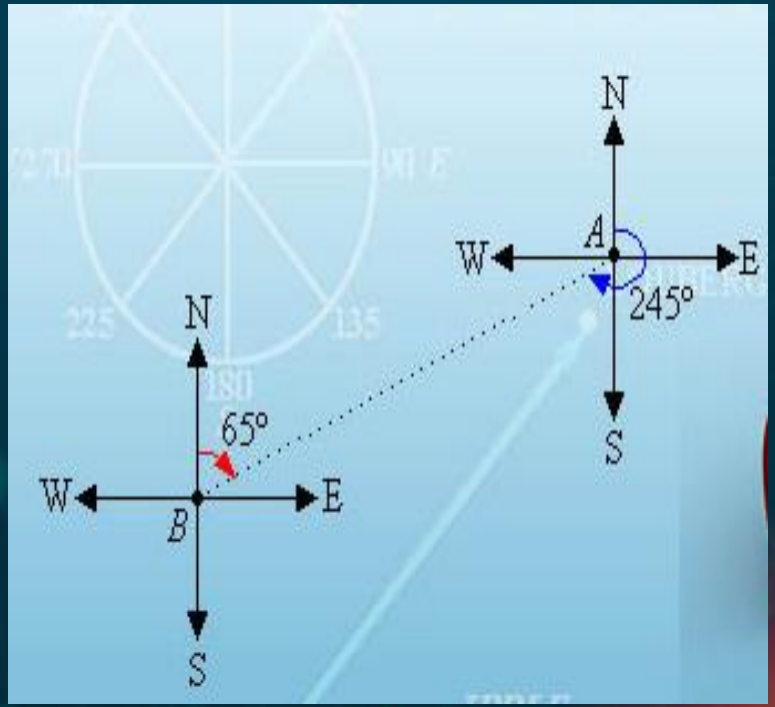
Bearing from point A to B, can be calculated as,

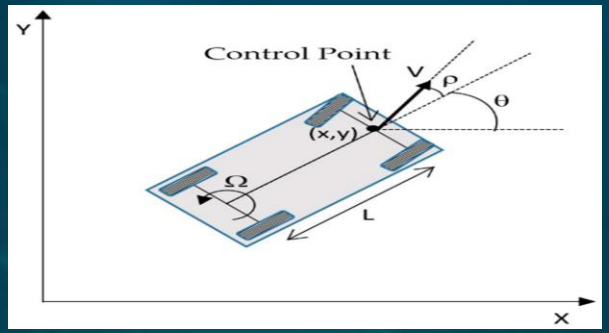
$$\beta = \text{atan2}(X,Y),$$

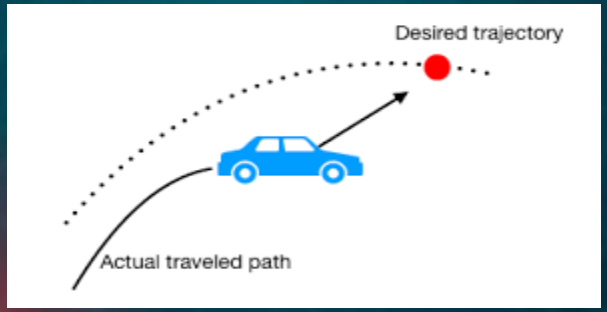
where, X and Y are two quantities and can be calculated as:

$$X = \cos \theta b * \sin \Delta L$$

$$Y = \cos \theta a * \sin \theta b - \sin \theta a * \cos \theta b * \cos \Delta L$$







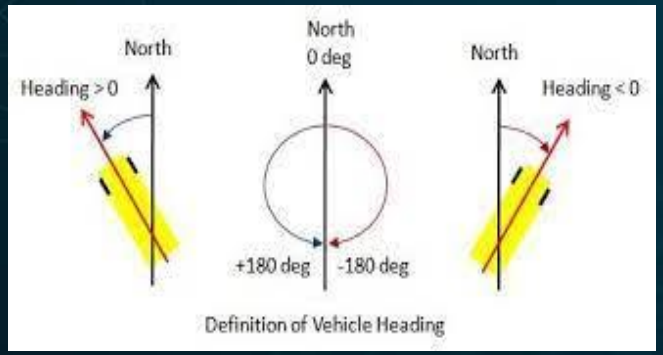
- Pose Correction
- Path Correction
- Obstacle Avoidance

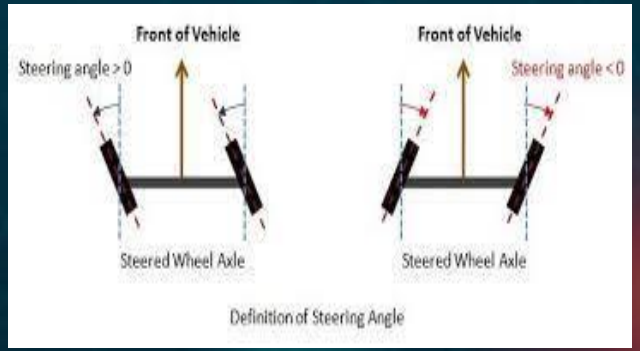
Controllers

Pose Correction

Steering Angle

```
double steeringAngle = bearing - heading;
```

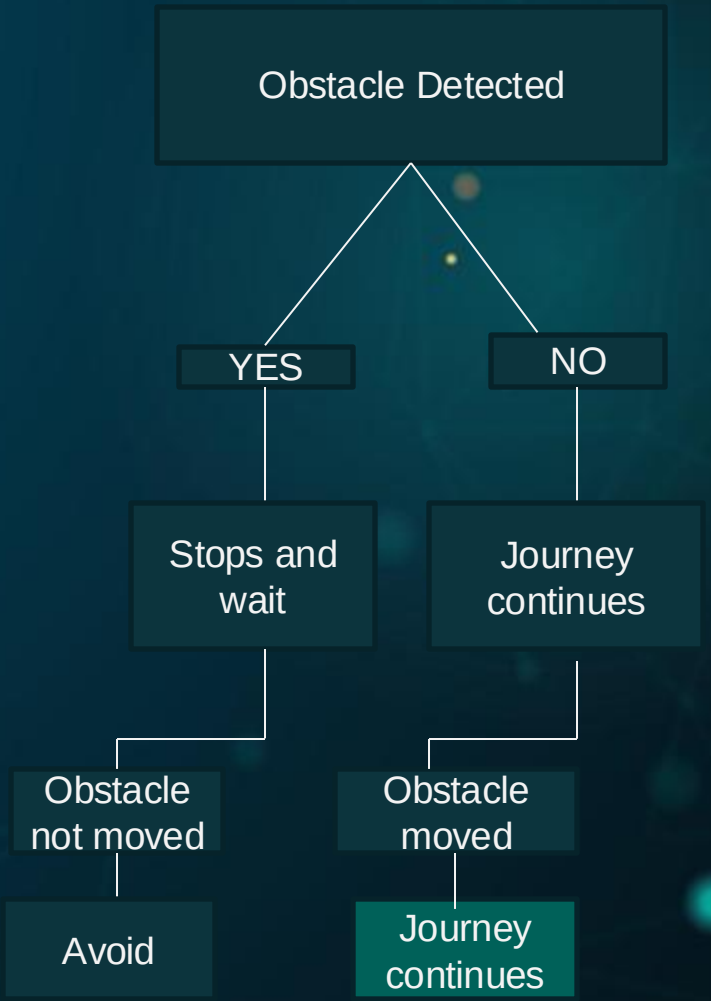




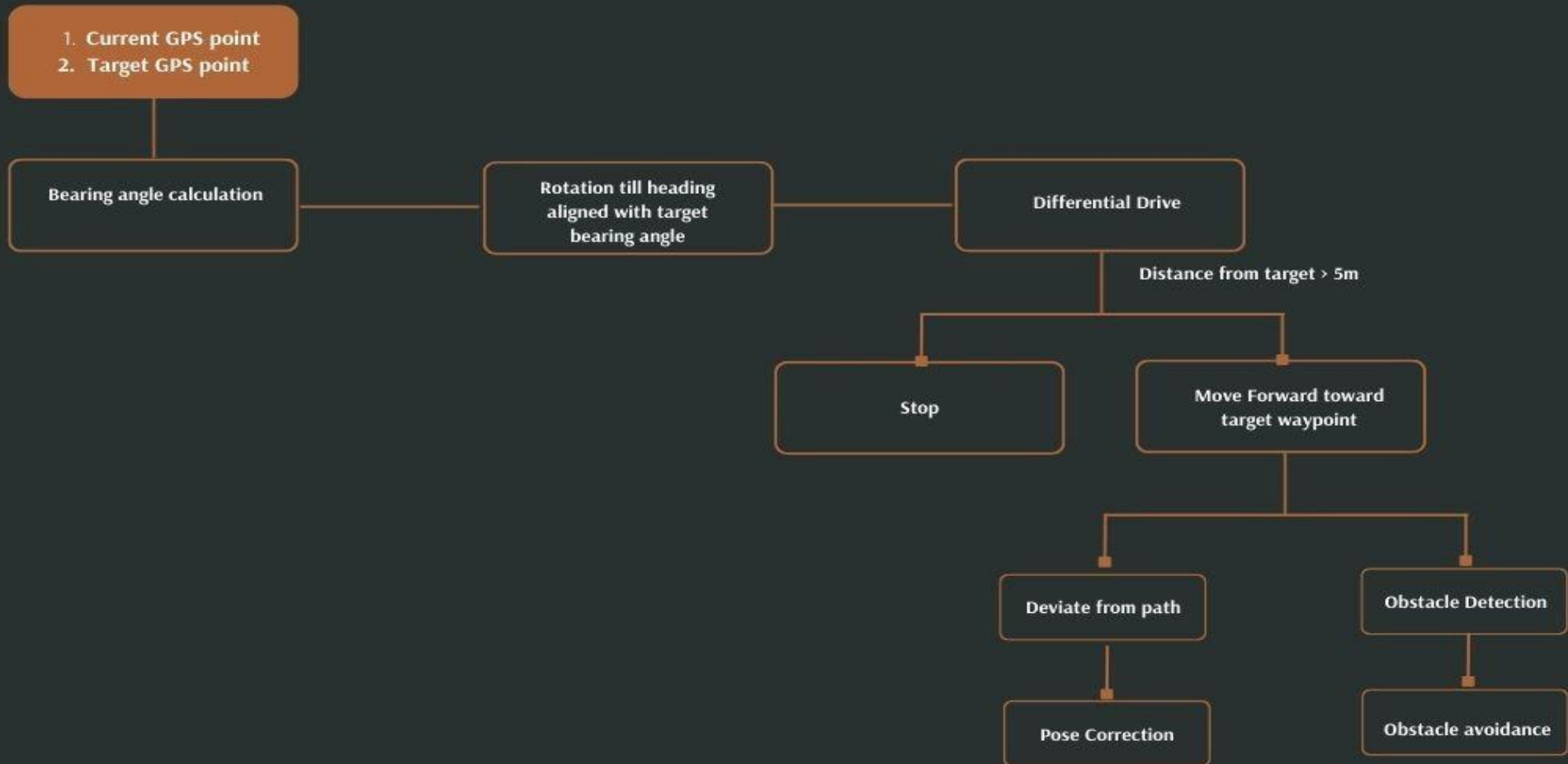
Path Correction

- Controller Activation Criteria
- Controller Parameters

Obstacle Avoidance



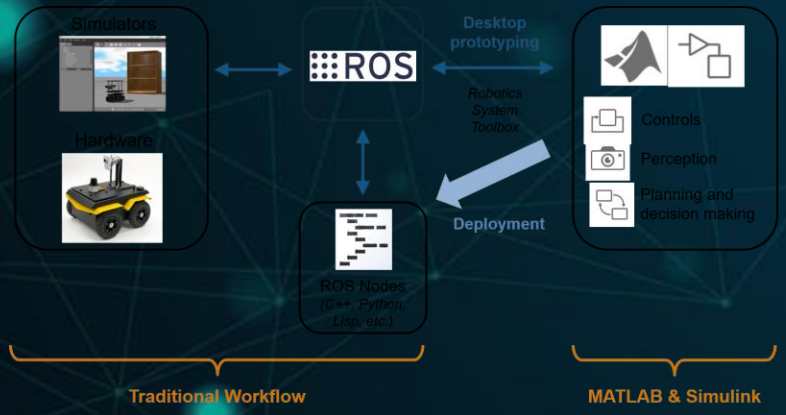
Waypoint Guidance



System Integration

- Why System integration is important?
- System integration Tools

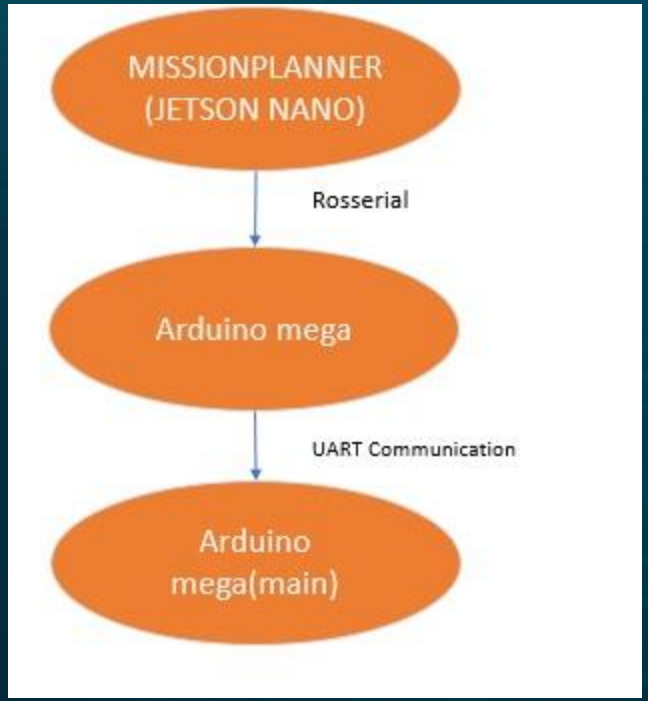
ROS: Robot operating system



ROS TOPICS

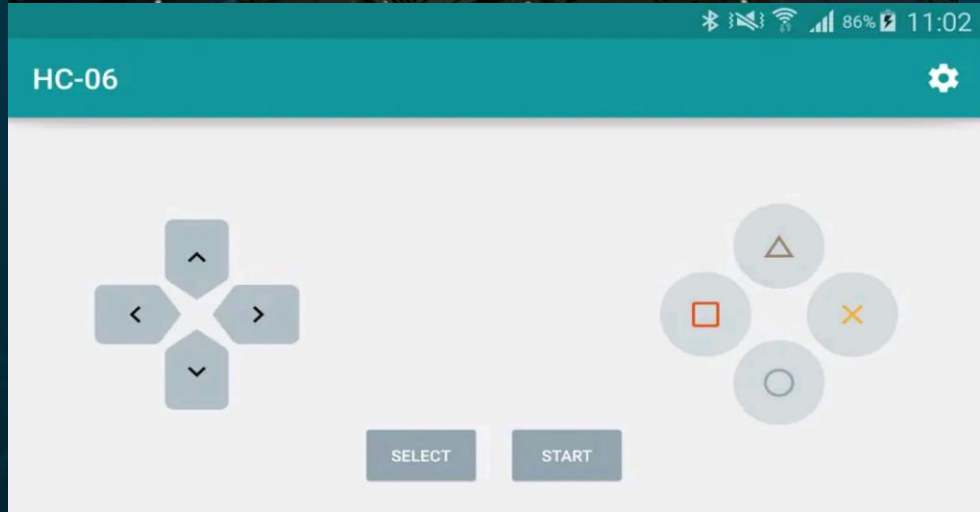
- ❑ GUI
- ❑ PLANNER
- ❑ CAMERA

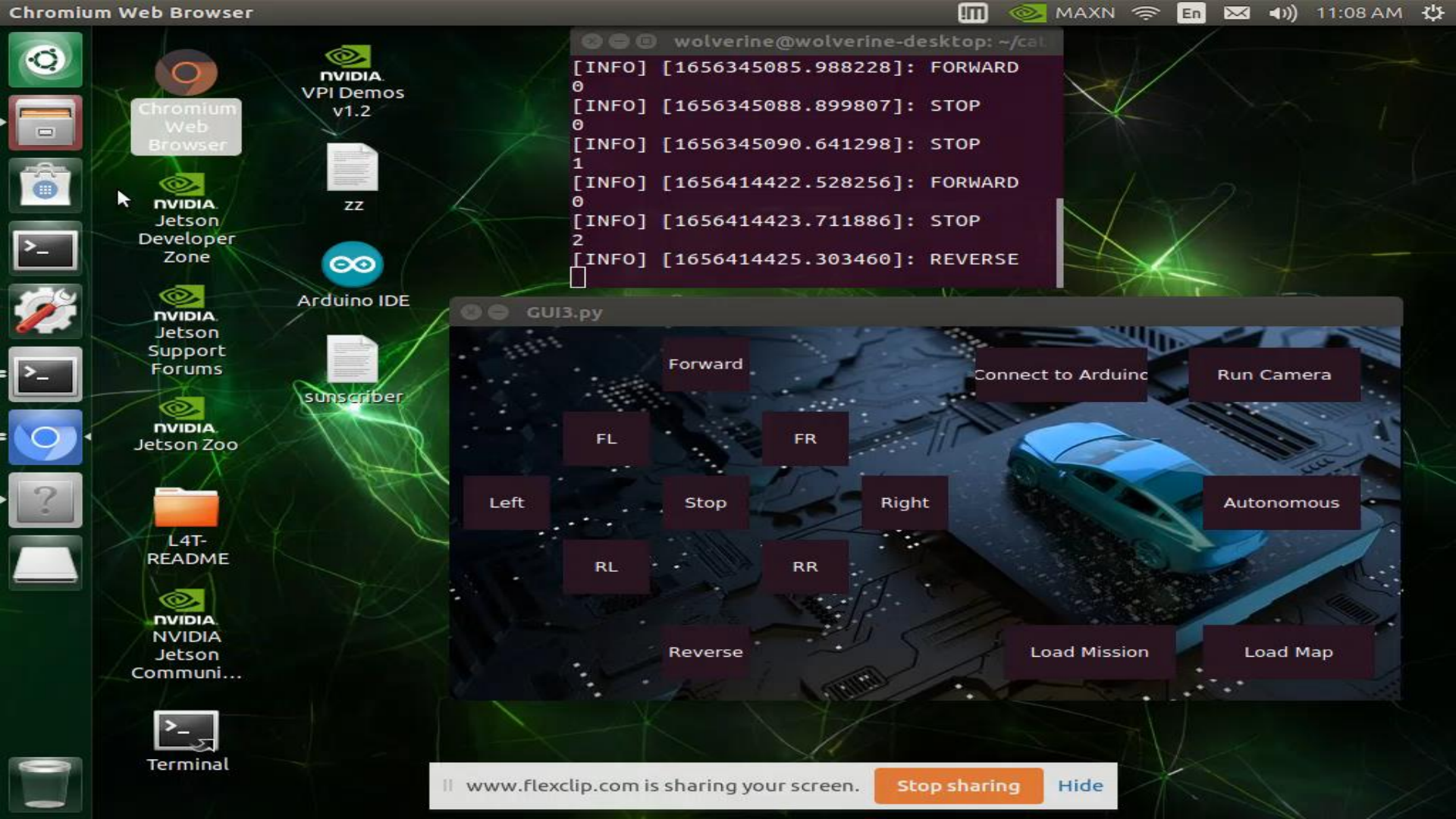
Information Flow

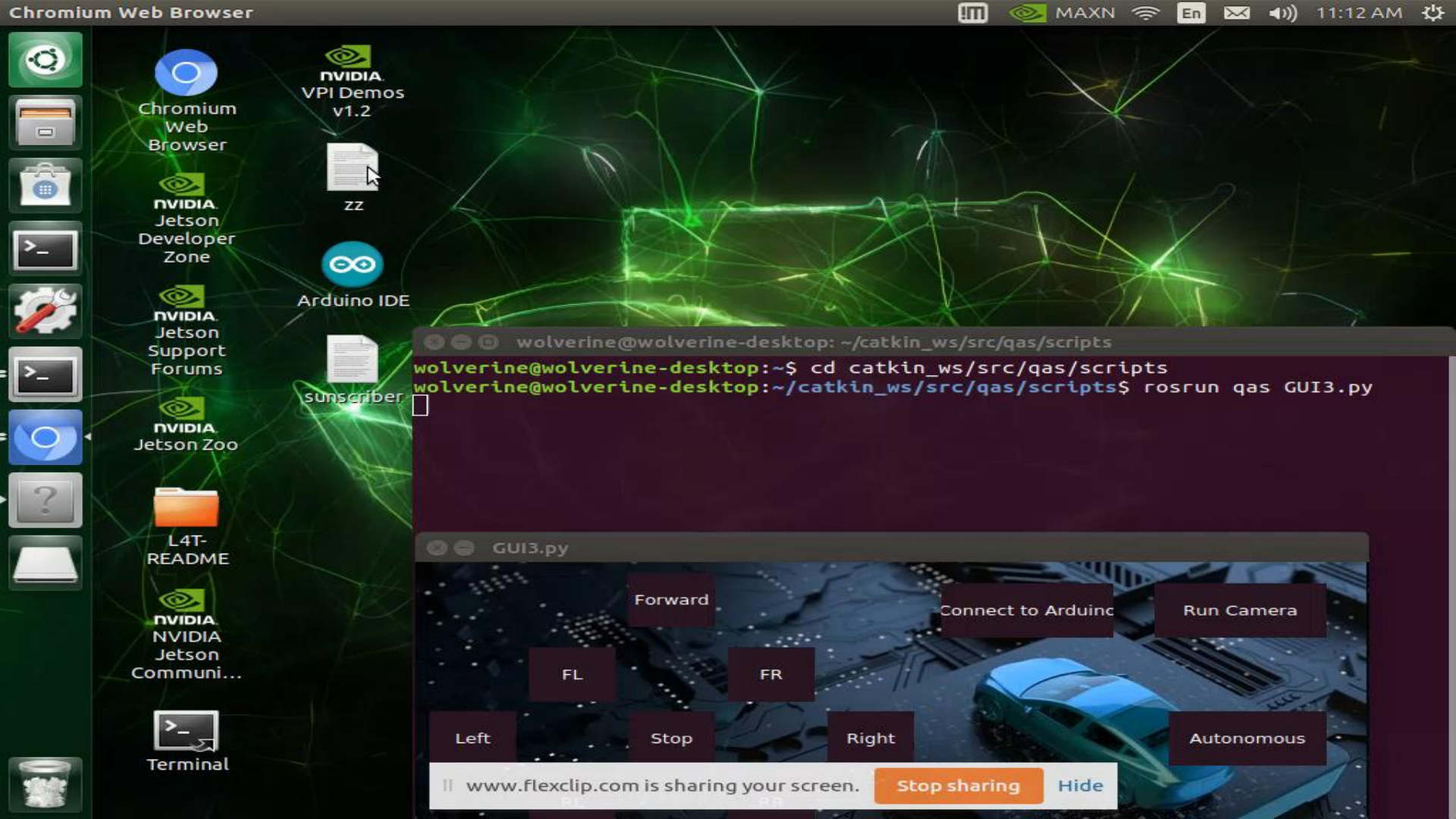


Topic: GUI

- Two-way control
- Teleoperations

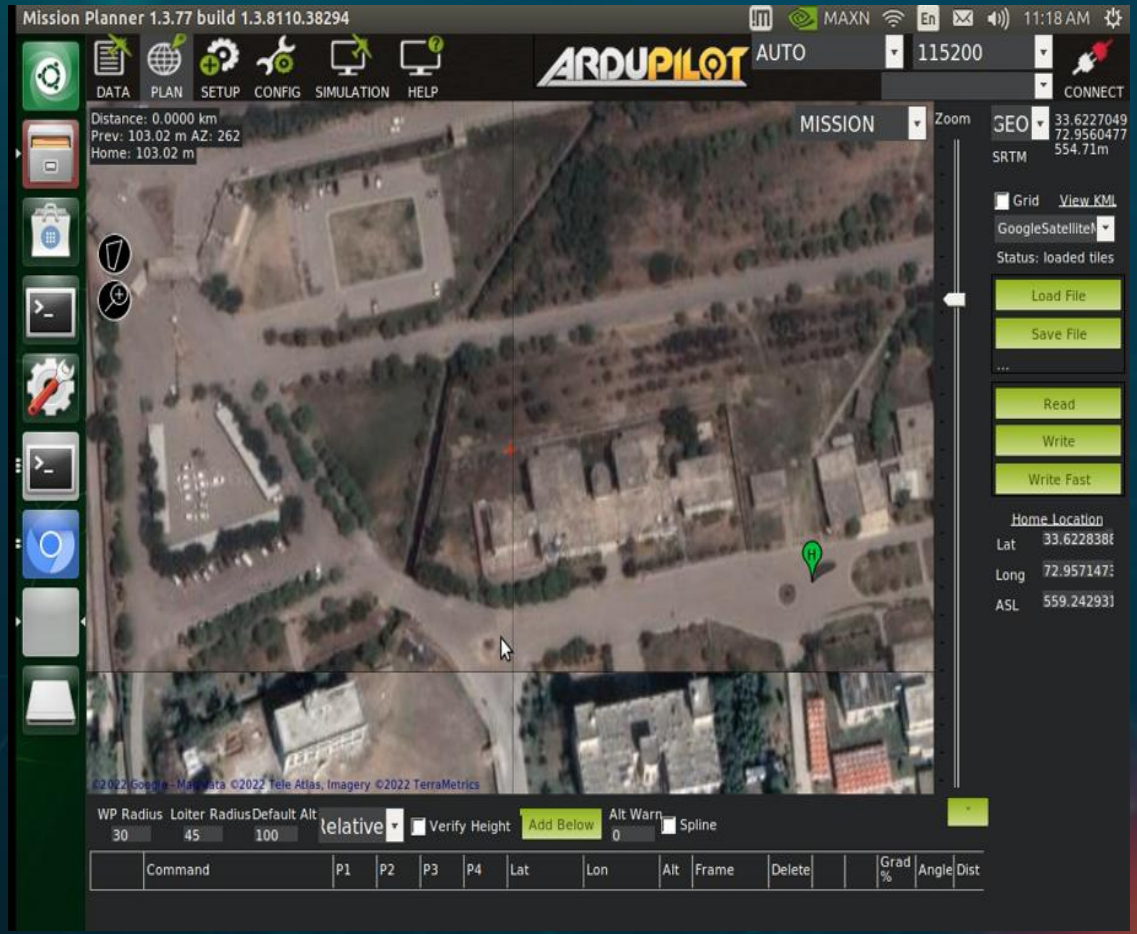



Chromium Web Browser
MAXN
En
11:08 AM
Chromium Web Browser
NVIDIA VPI Demos v1.2
NVIDIA Jetson Developer Zone
zz
Arduino IDE
NVIDIA Jetson Support Forums
sunscriber
NVIDIA Jetson Zoo
L4T-README
NVIDIA NVIDIA Jetson Communi...
Terminal
wolverine@wolverine-desktop: ~/cat
[INFO] [1656345085.988228]: FORWARD
0
[INFO] [1656345088.899807]: STOP
0
[INFO] [1656345090.641298]: STOP
1
[INFO] [1656414422.528256]: FORWARD
0
[INFO] [1656414423.711886]: STOP
2
[INFO] [1656414425.303460]: REVERSE
GUI3.py
Forward
Connect to Arduinc
Run Camera
FL
FR
Left
Stop
Right
Autonomous
RL
RR
Reverse
Load Mission
Load Map
www.flexclip.com is sharing your screen.
Stop sharing
Hide


Chromium Web Browser
MAXN
En
11:12 AM
Chromium Web Browser
NVIDIA VPI Demos v1.2
zz
NVIDIA Jetson Developer Zone
Arduino IDE
NVIDIA Jetson Support Forums
sunscriber
NVIDIA Jetson Zoo
L4T-README
NVIDIA NVIDIA Jetson Communi...
Terminal
wolverine@wolverine-desktop: ~/catkin_ws/src/qas/scripts
wolverine@wolverine-desktop:~$ cd catkin_ws/src/qas/scripts
wolverine@wolverine-desktop:~/catkin_ws/src/qas/scripts$ rosrun qas GUI3.py
GUI3.py
Forward
Connect to Arduinc
Run Camera
FL
FR
Left
Stop
Right
Autonomous
www.flexclip.com is sharing your screen.
Stop sharing
Hide

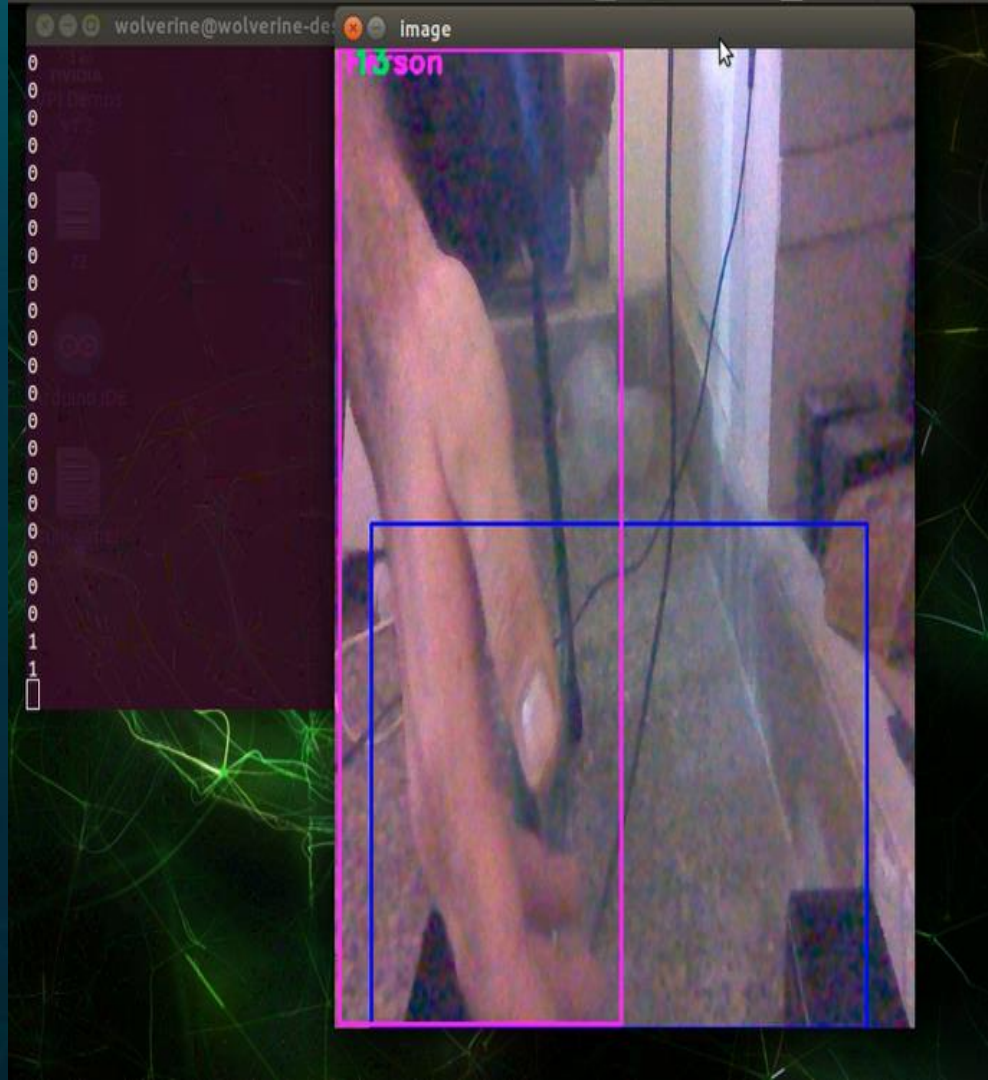
Topic: Planner

- Waypoint Selection
- ArduPilot MissionPlanner



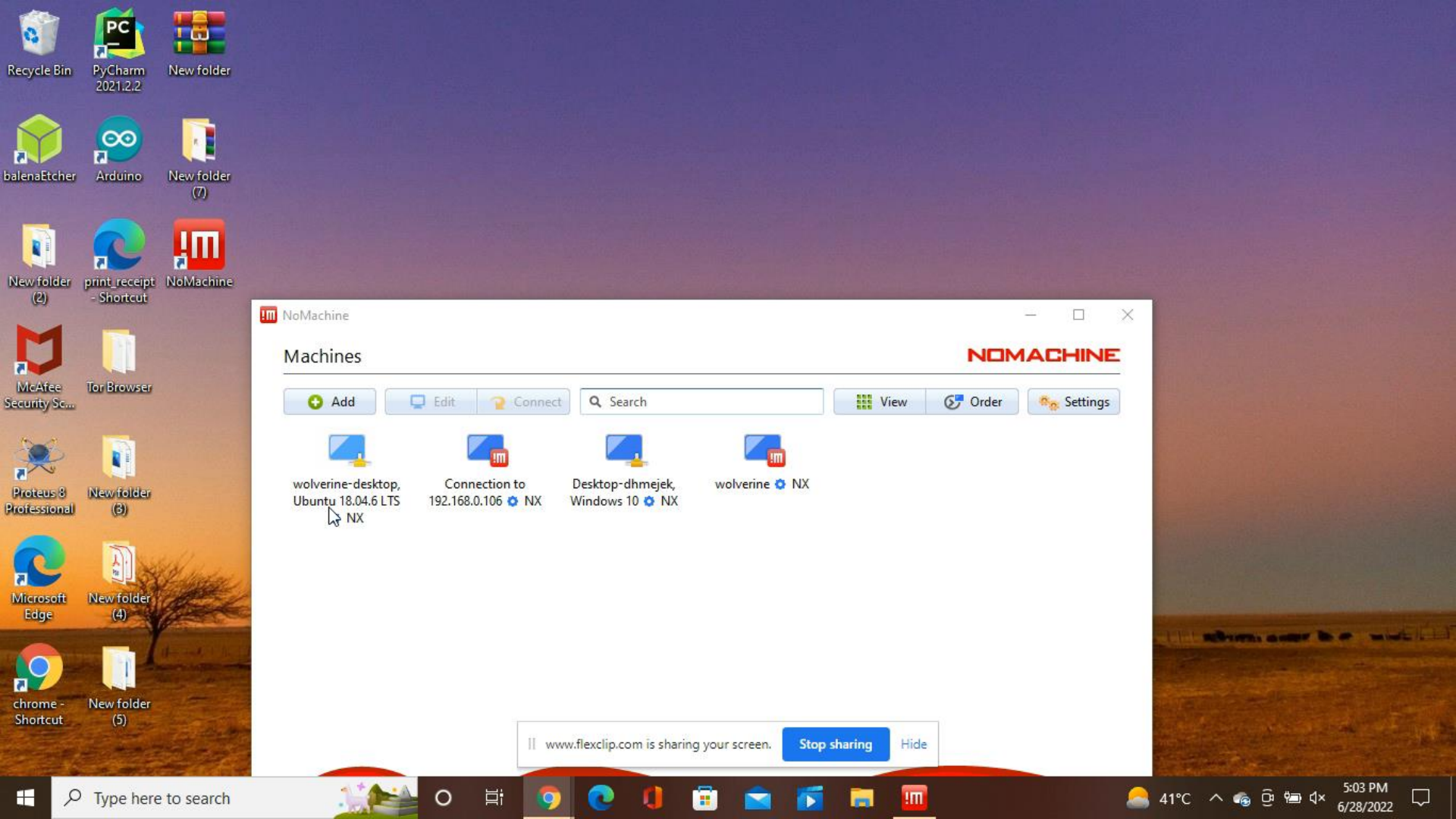
Topic: CAMERA

- Publishing 1 or 0



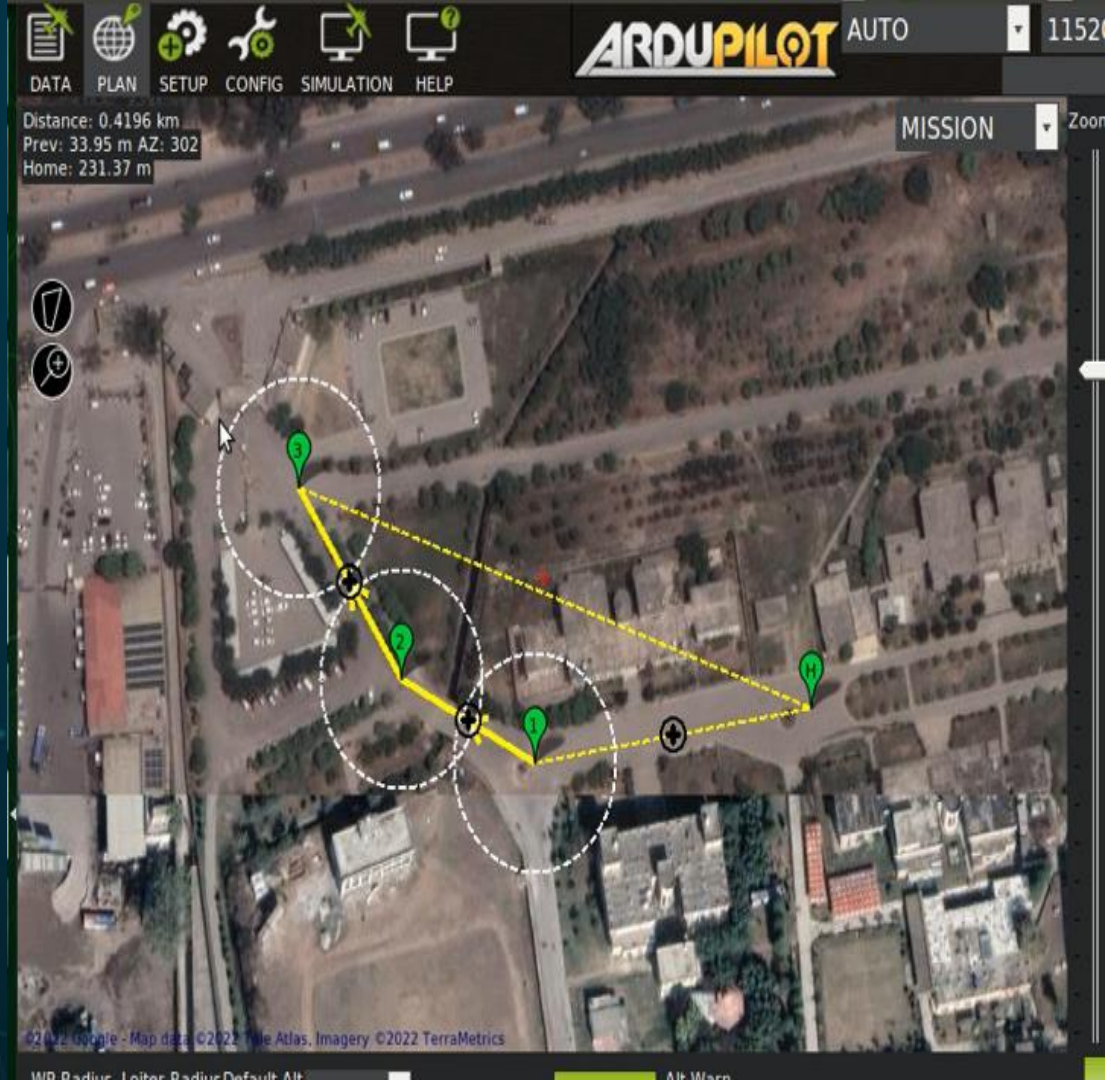
WLAN Connection

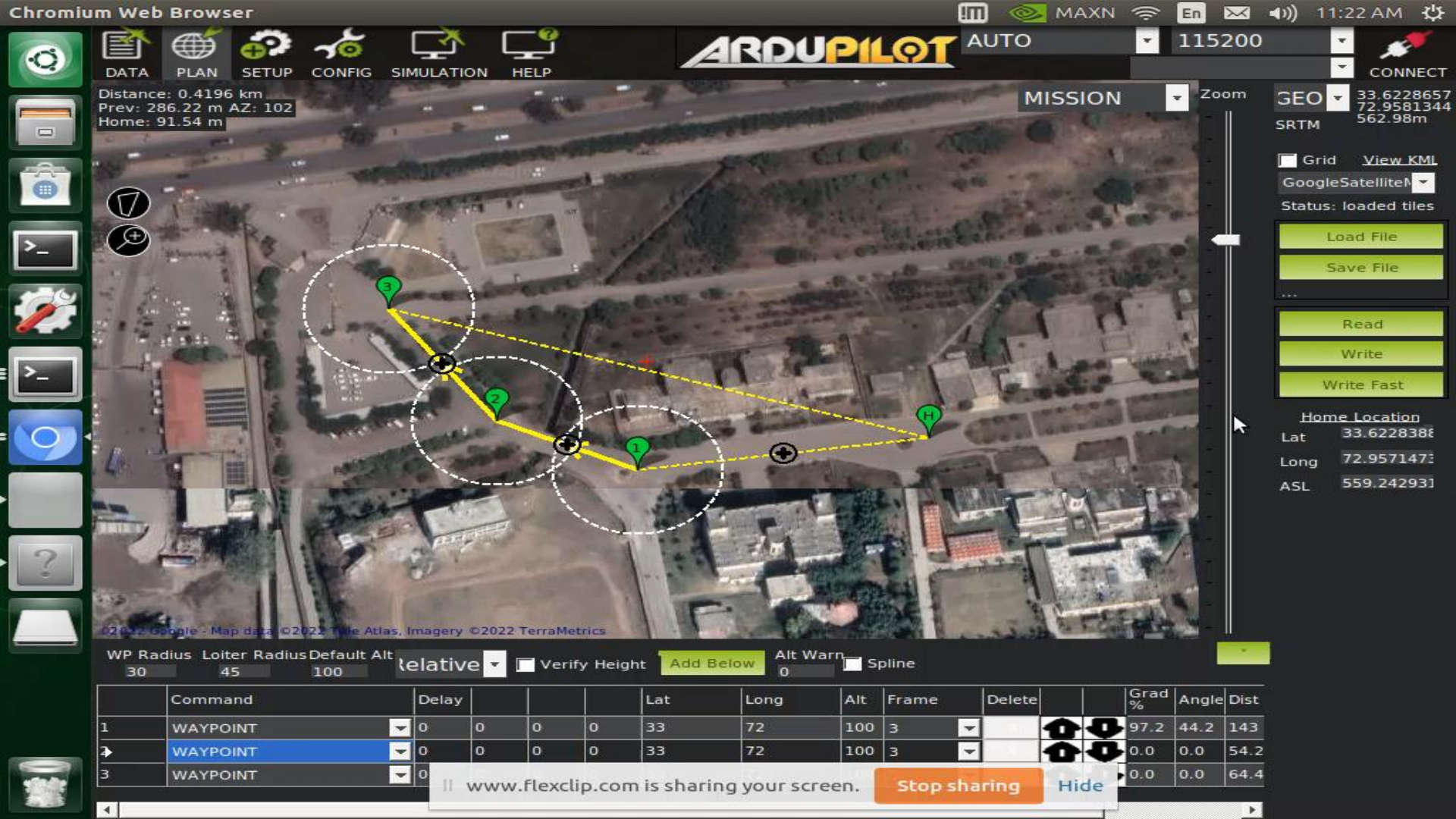
- Mobile Wi-Fi Hotspot
- Remote Desktop Based Connection


Recycle Bin
PyCharm 2021.2.2
New folder
balenaEtcher
Arduino
New folder (7)
New folder (2)
print_receipt - Shortcut
NoMachine
McAfee Security Sc...
Tor Browser
Proteus 8 Professional
New folder (3)
Microsoft Edge
New folder (4)
chrome - Shortcut
New folder (5)
NoMachine
Machines
NOMACHINE
Add
Edit
Connect
Search
View
Order
Settings
wolverine-desktop, Ubuntu 18.04.6 LTS NX
Connection to 192.168.0.106 NX
Desktop-dhmejek, Windows 10 NX
wolverine NX
www.flexclip.com is sharing your screen.
Stop sharing
Hide
Type here to search
41°C
5:03 PM
6/28/2022

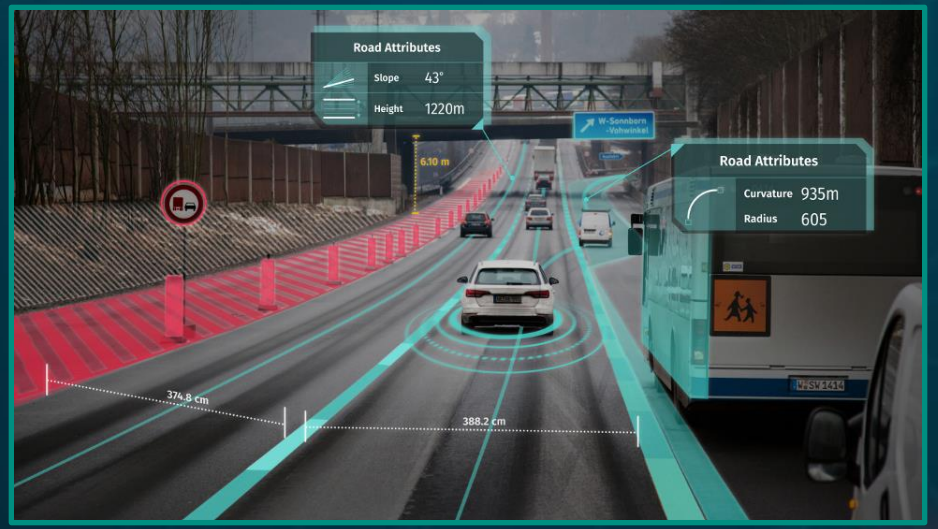
Path Planning

waypoint selection



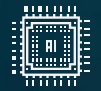

Chromium Web Browser
MAXN
En
11:22 AM
DATA
PLAN
SETUP
CONFIG
SIMULATION
HELP
ARDUPILOT
AUTO
115200
CONNECT
Distance: 0.4196 km
Prev: 286.22 m AZ: 102
Home: 91.54 m
MISSION
Zoom
GEO
33.6228657
72.9581344
562.98m
SRTM
Grid
View KML
GoogleSatelliteM
Status: loaded tiles
Load File
Save File
...
Read
Write
Write Fast
Home Location
Lat 33.6228388
Long 72.9571473
ASL 559.242931
©2022 Google - Map data ©2022 Tele Atlas, Imagery ©2022 TerraMetrics
WP Radius 30
Loiter Radius 45
Default Alt 100
Relative
Verify Height
Add Below
Alt Warn 0
Spline
Command Delay Lat Long Alt Frame Delete Grad % Angle Dist
1 WAYPOINT 0 0 0 0 33 72 100 3 97.2 44.2 143
2 WAYPOINT 0 0 0 0 33 72 100 3 0.0 0.0 54.2
3 WAYPOINT 0 0.0 0.0 64.4
www.flexclip.com is sharing your screen.
Stop sharing
Hide

Perception


Road Attributes
Slope 43°
Height 1220m
6.10 m
W-Sonnborn
-Vohwinkel
Road Attributes
Curvature 935m
Radius 605
374.8 cm
388.2 cm

WHAT IS PERCEPTION?

Comprehension of envirnoment around the vehicle.



Processing of data from cameras and other high tech sensors.

Perception is crucial for safe and reliable operation.

It fuels the core decision-making i.e, how the vehicle should move next

CAMERA CALIBRATION

Why we need camera calibration?

Intrinsic and extrinsic parameters of the camera.

Camera Calibration methodology.

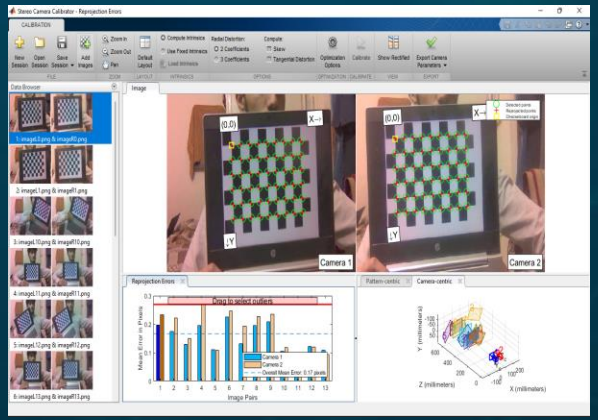


Image acquisition from different view points.

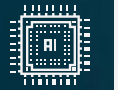
Defining real world coordinate of known 3D points

Find 2D image coordinate of each 3D point using corner detection

Estimate camera parameter using 3D points and their pixel coordinates.

Deep Learning framework for object detection

Object Detection



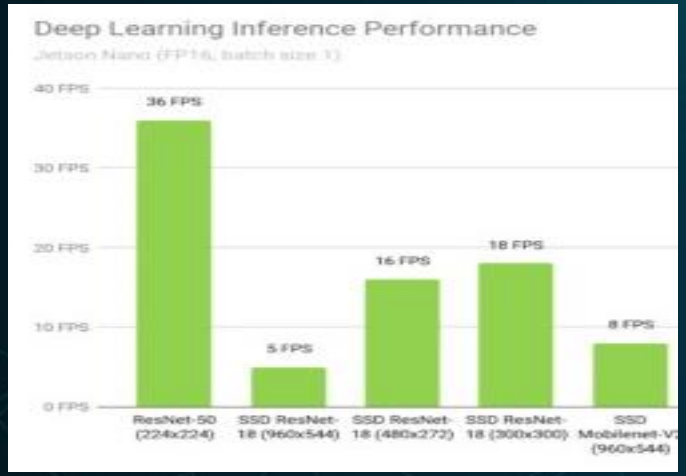
Trade off between inference speed and accuracy.

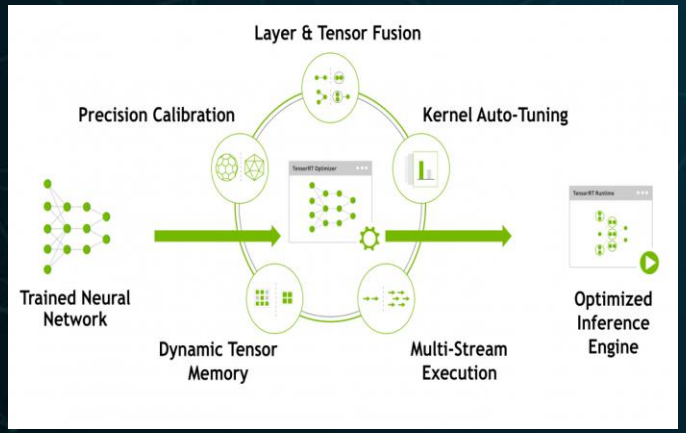
Deep Learning models for object detection.

Depth estimation vs ROI based approach.

- REAL TIME OBSTACLE DETECTION AND AVOIDANCE

TensorRT Optimization





- Software development kit for high inference of deep learning models.
- INT8 and FP-16 optimization for Low latency and high throughput.
- 36x faster than cpu only platform.

conclusion

Thank you