Course Code: MCT433s

Course Name: Design of Autonomous Systems

# Final SUBMISSION

TEAM (6)

SUBMITTED TO:

Dr. Omar Shehata

Eng. Mazen Talaat

# TEAM MEMBERS INFO:

| Name: | ID: | Work Percentage: |
|---|---|---|
| Shehab Mohamed Abdo | 2001558 | 25% |
| Ahmmad Anany Abdel Latif | 2001893 | 25% |
| Abdelrahman Osama Mohamed | 2000979 | 25% |
| Ahmed Mohamed Abdallah | 2001384 | 25% |

# Agenda

01

INTRODUCTION

04

GAZEBO SNABSHOTS

02

OBJECTIVES

03

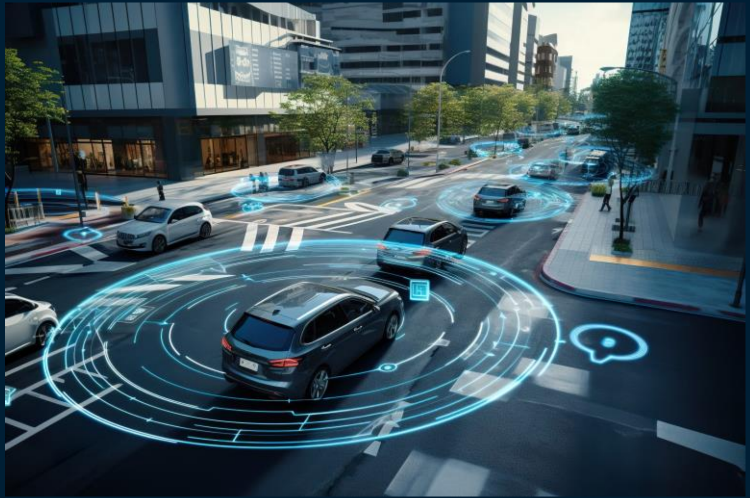
LANE KEEPING &

LANE TRANSITION

05

CONTROL LOGIC

# 1-0 INTRODUCTION:



Autonomous systems are self-operating machines or software that can perform tasks without direct human intervention. They use advanced technologies like artificial intelligence, machine learning, sensors, and data processing to perceive their environment, make decisions and execute actions.
Examples include self-driving cars, drones, industrial robots, and intelligent software agents.

## Key Features:

- **Perception:** Collect and process real-time data from surroundings.
- **Decision-Making**: Analyze information to determine the best actions.
- **Action**: Execute tasks accurately and efficiently.

# 2-0 OBJECTIVES

**MAINLY:**

The objective of this project is:

- ❑ Design and implement a fully functional autonomous vehicle system in simulation environment.
- ❑ Integrating control systems for lane-keeping and obstacle avoidance using a differential drive robot.
- ❑ Develop and validate our control algorithms in simulation tools like ROS and Gazebo.

# 2-1 Objectives

**FOR PHASE (2):**

- ❑ Define and develop the control strategy for lane-keeping, focusing on feedback control using sensors (P-controller).
- ❑ Develop a strategy for obstacle detection and avoidance, specifying how the robot will execute lane transitions

**FOR PHASE (3):**

- ❑ Implement lane-keeping.
- ❑ Implement obstacle avoidance and lane transitions.

# 2-1 Objectives

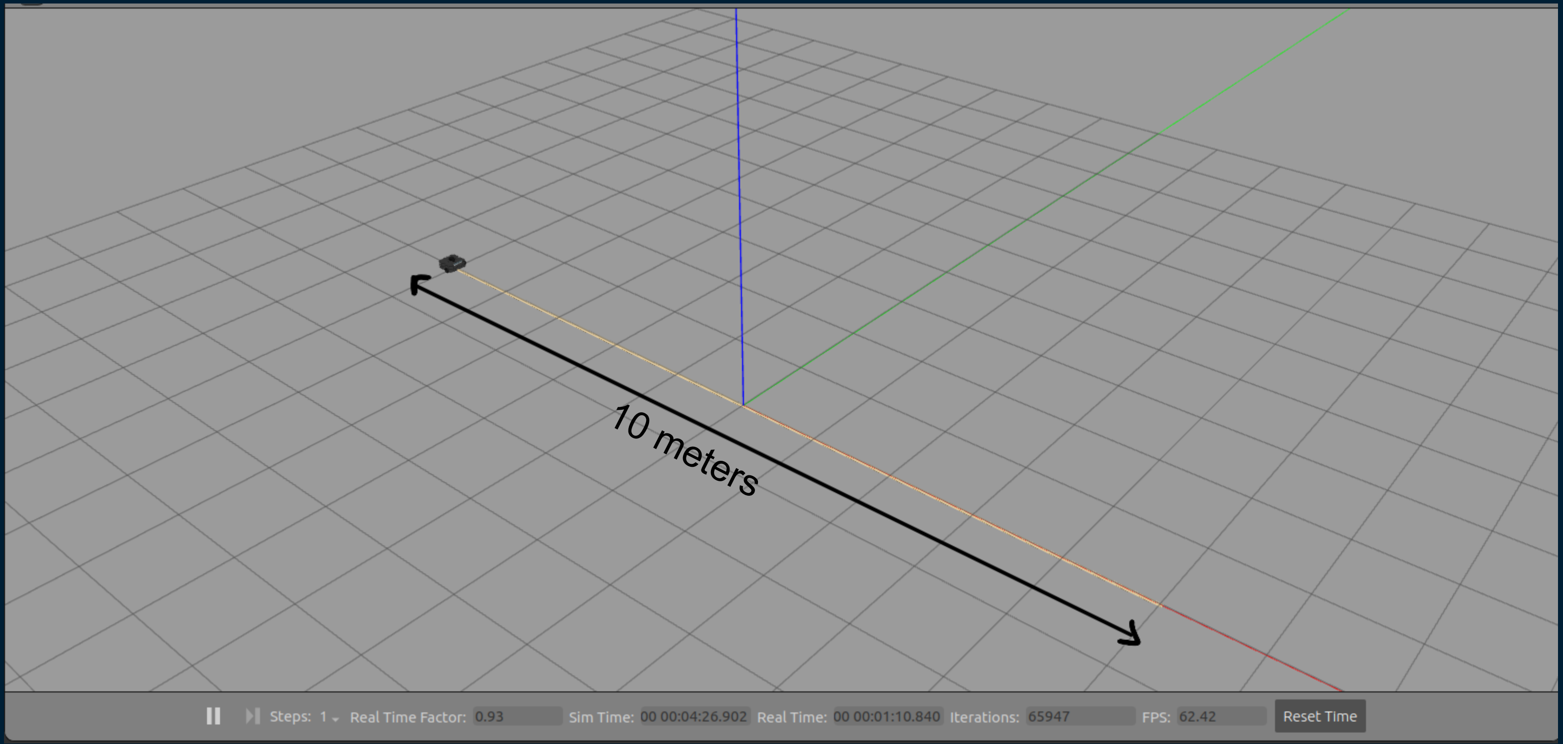
**FOR PHASE (2):**

- ❑ Define and develop the control strategy for lane-keeping, focusing on feedback control using sensors (P-controller).
- ❑ Develop a strategy for obstacle detection and avoidance, specifying how the robot will execute lane transitions

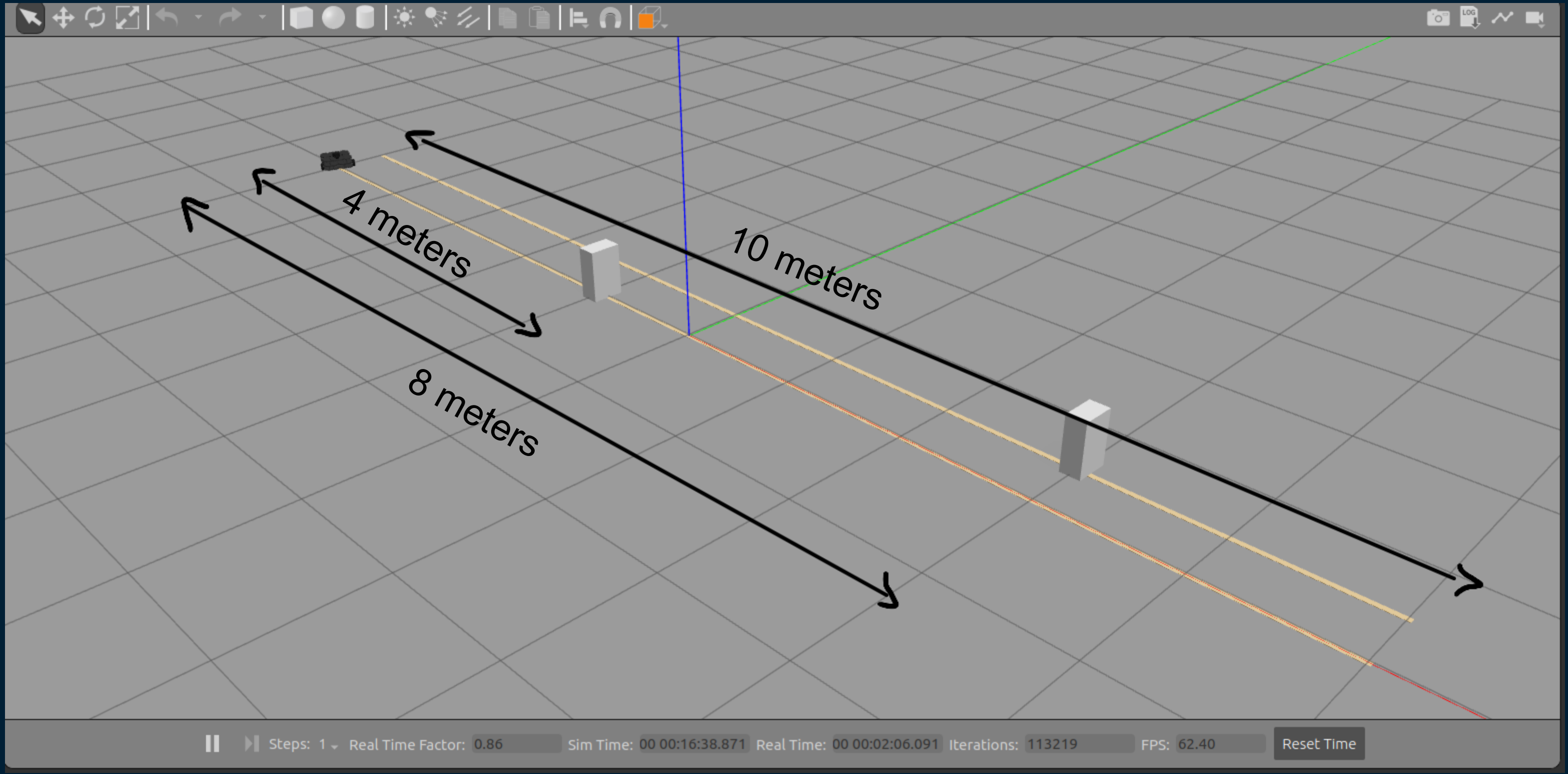
**FOR PHASE (3):**

- ❑ Implement lane-keeping.
- ❑ Implement obstacle avoidance and lane transitions.

# 3-0 LANE KEEPING

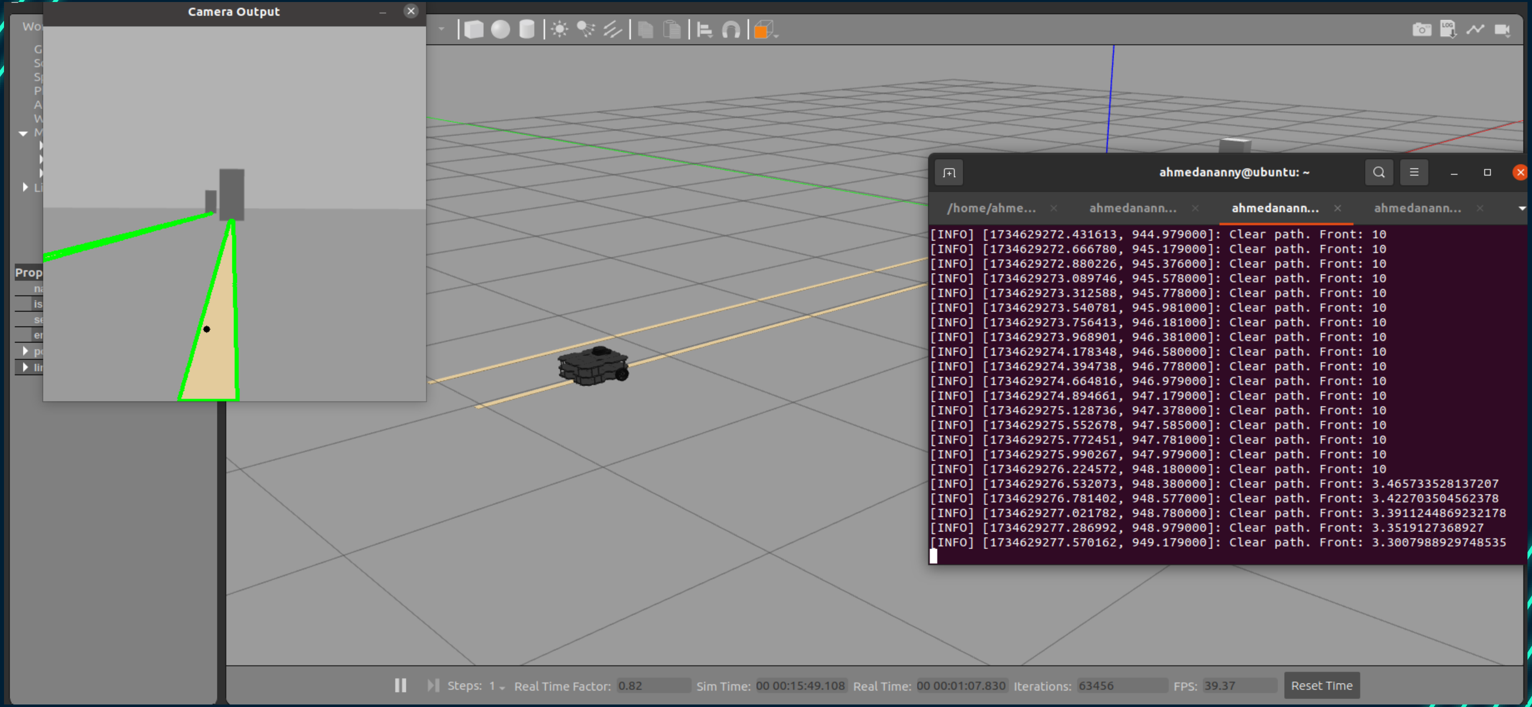


# 3-1 LANE TRANSITION



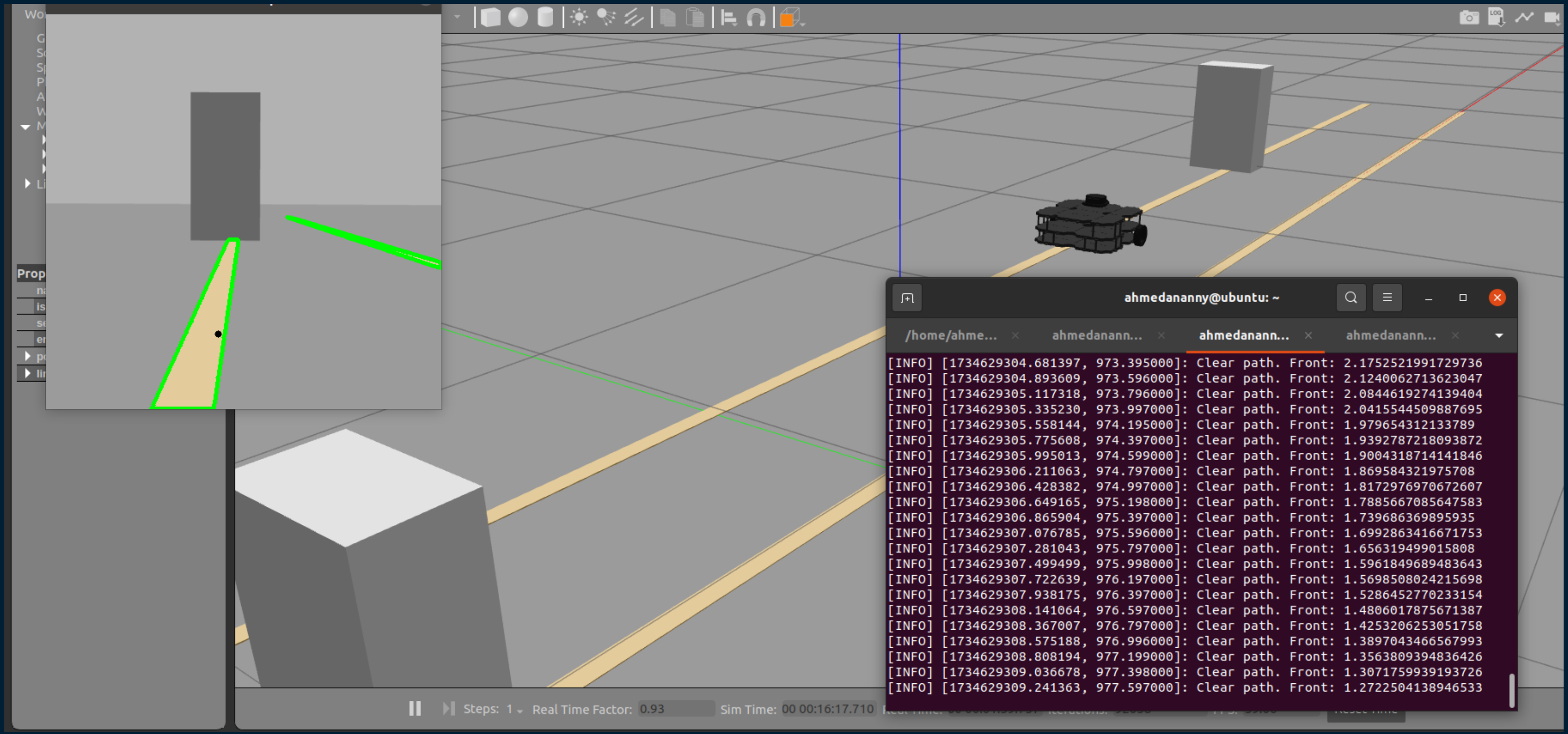
# 3-0 GAZEBO ENVIRONMENT SNAPSHOTS

❑Clear Path 1:



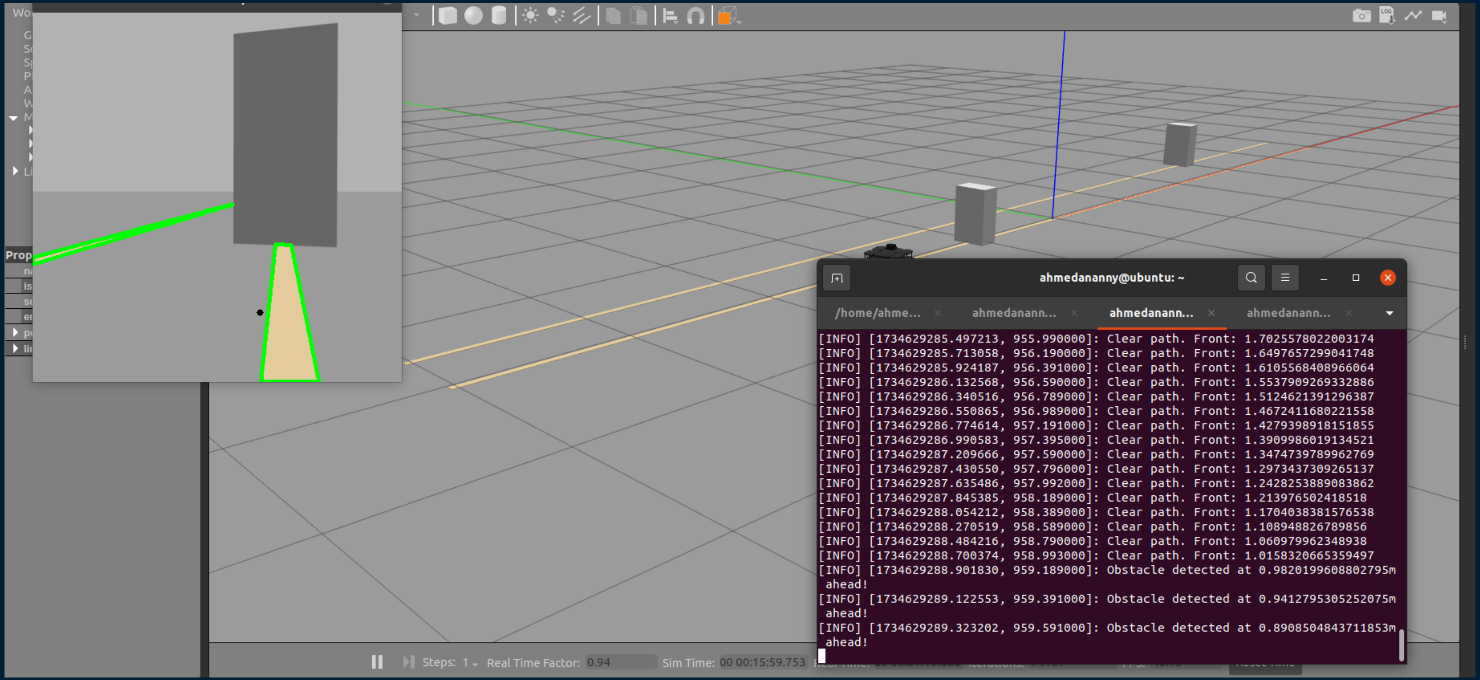
# 3-1 GAZEBO ENVIRONMENT SNAPSHOTS

❑ Clear Path 2 :



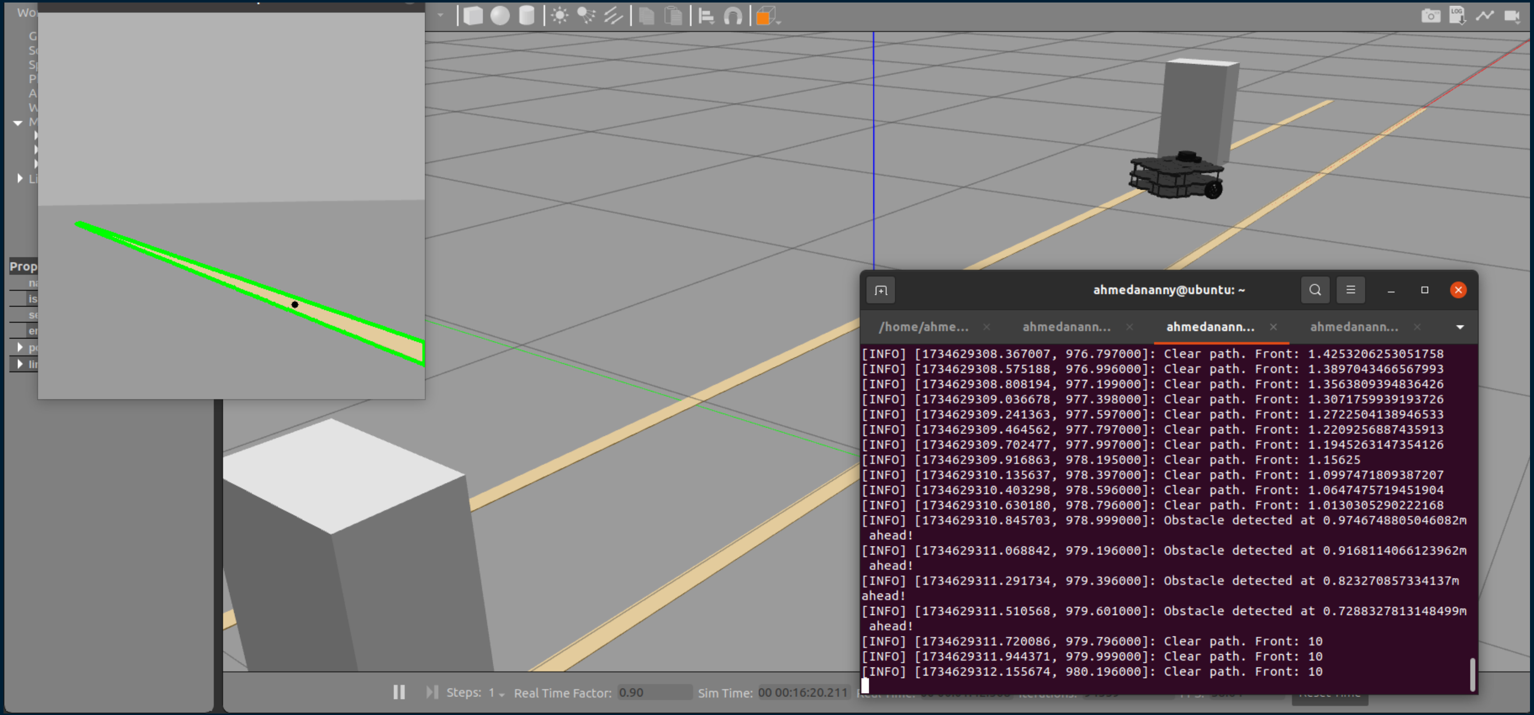
# 3-2 GAZEBO ENVIRONMENT SNAPSHOTS

- First object detection:



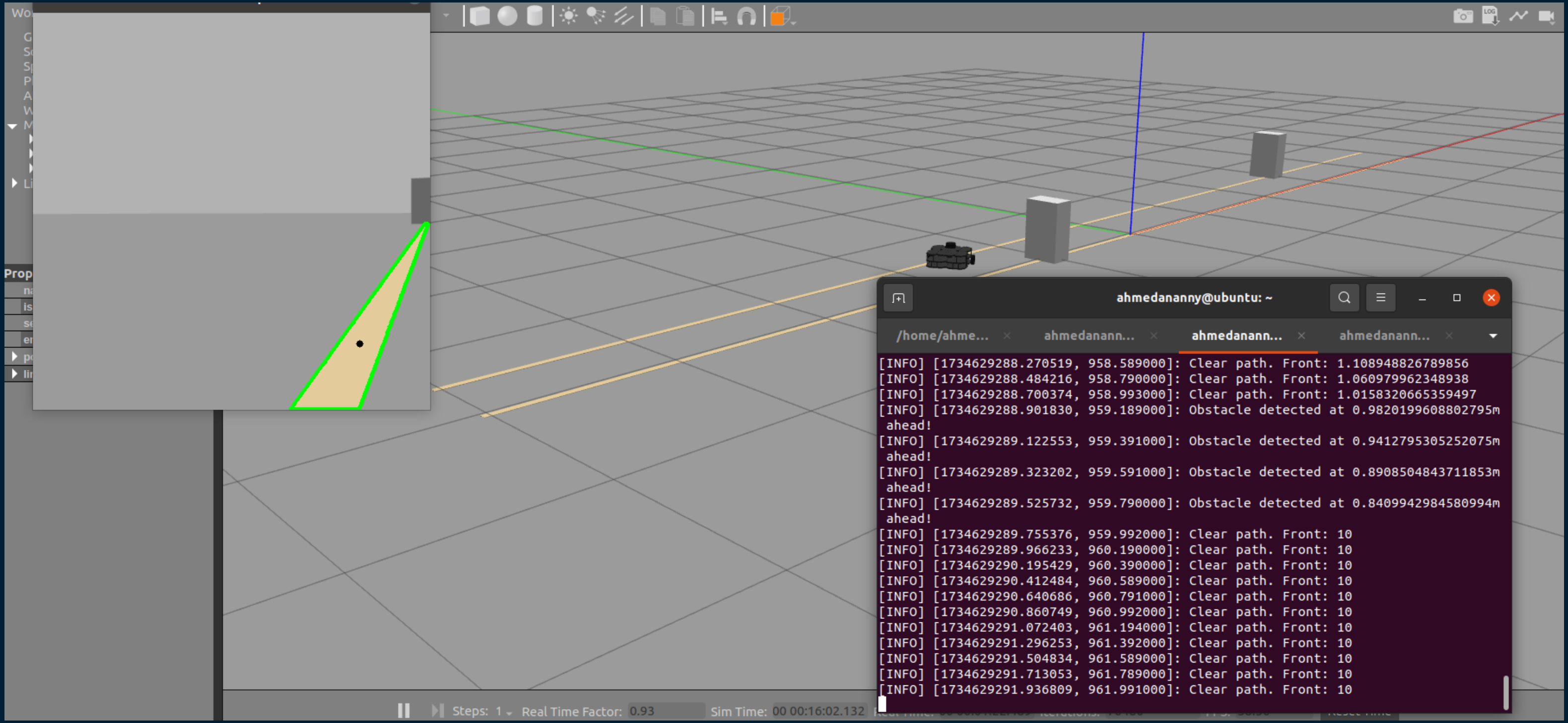
# 3-3 GAZEBO ENVIRONMENT SNAPSHOTS

- ❑ Second object detection:



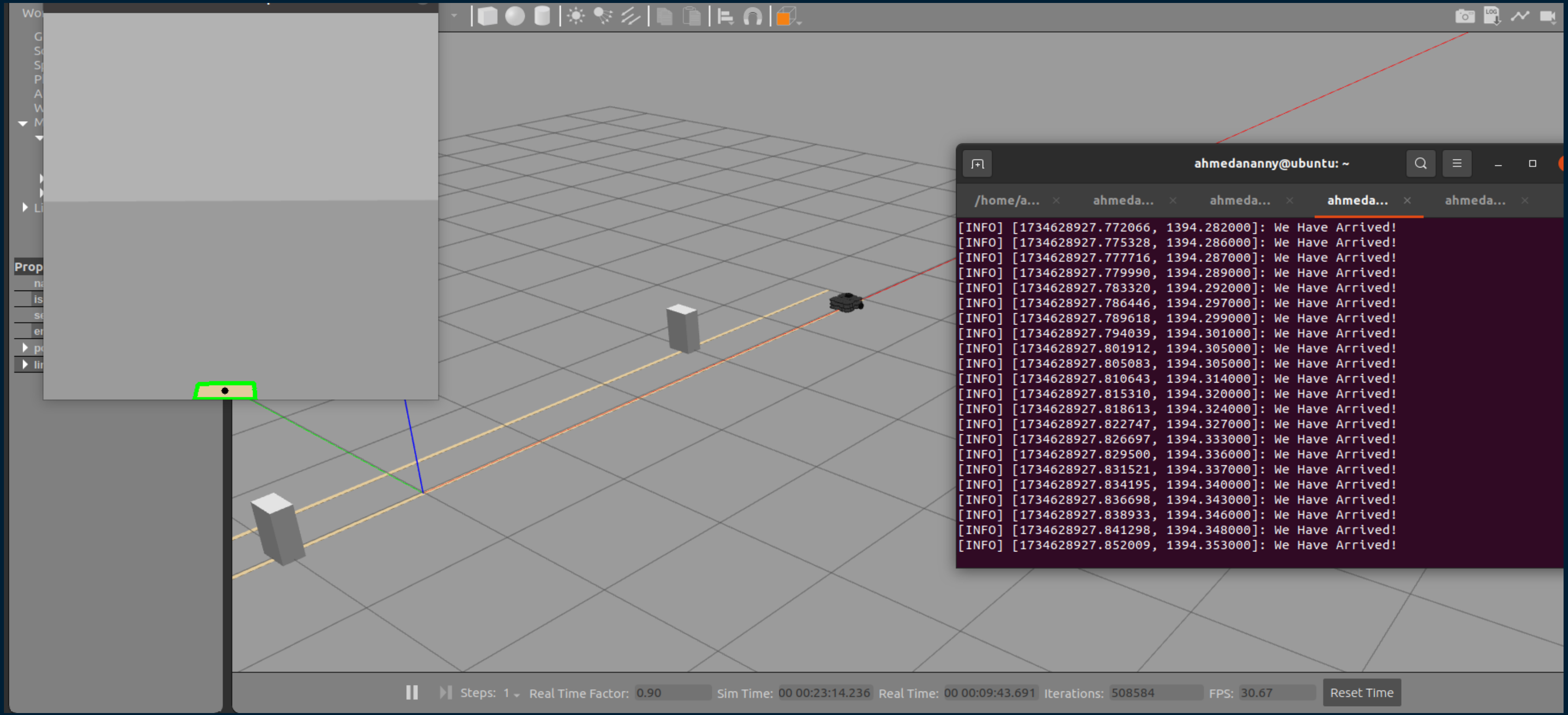
# 3-4 GAZEBO ENVIRONMENT SNAPSHOTS

- ❑ Lane switching:



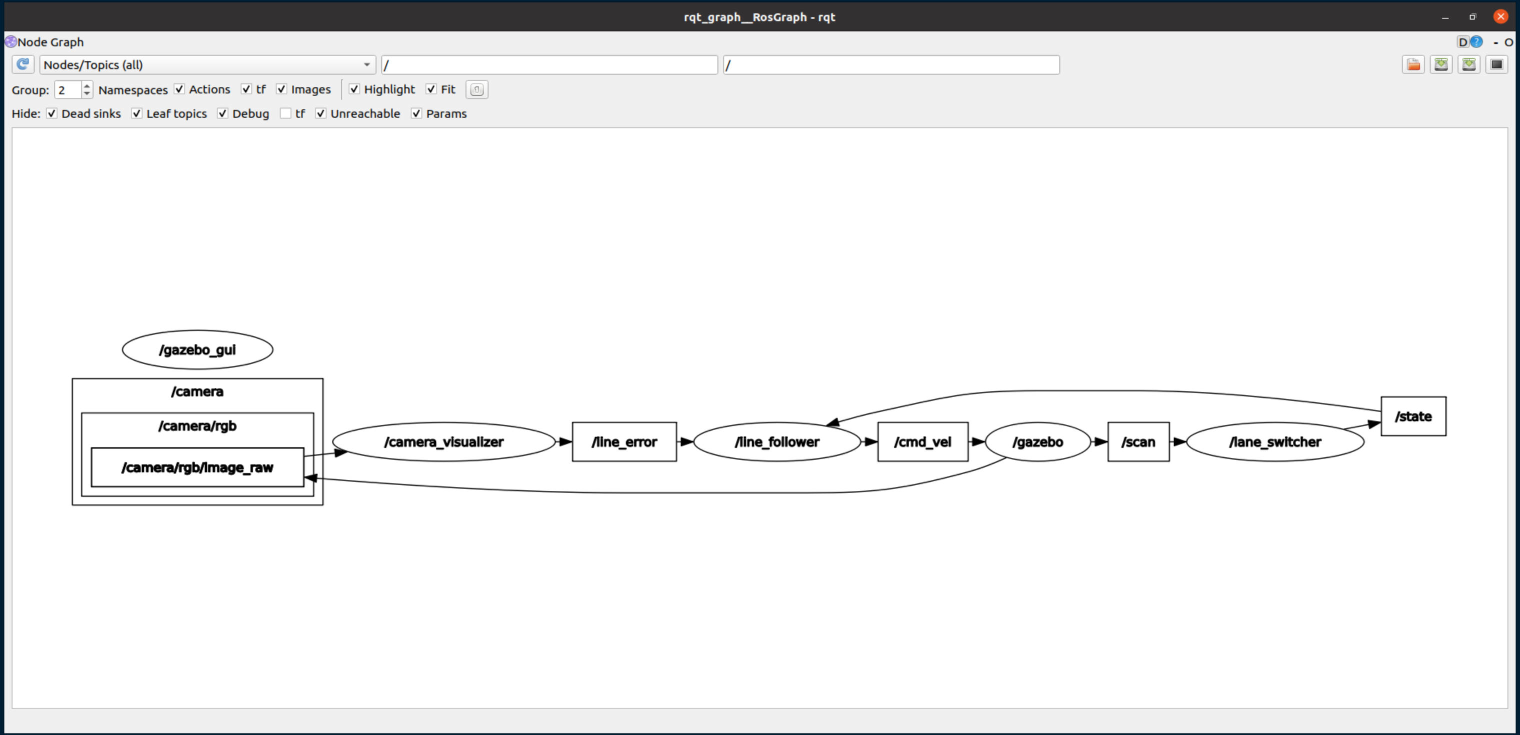
# 3-5 GAZEBO ENVIRONMENT SNAPSHOTS

- Goal reached!



# 3-5 GAZEBO ENVIRONMENT SNAPSHOTS

- Nodes graph:



# 4-CONTROL LOGIC OVERVIEW

**Line Follower Control Logic:**

- **Objective**: Guide the robot to follow a line using proportional control.
- **Control Scenarios**:

    1. **Non-Zero Error**:
        - Adjust angular velocity: Control Output= Kp×Error.
        - **Linear Velocity**: 0.2, **Angular Velocity**: -Control Output.
    2. **Zero Error**:
        - Move straight faster.
        - **Linear Velocity**: 0.5, **Angular Velocity**: 0.0.
    3. **Arrival Condition**:
        - Stop movement.
        - **Linear Velocity**: 0.0, **Angular Velocity**: 0.0.

- **Key Features**:
    - Input: /line_error topic for error feedback.
    - Output: /cmd_vel for velocity commands.
    - Logs real-time behavior for monitoring.

***Simple, effective, and reactive control for real-time line tracking!***

# Thanks!

Do you have any questions?