

# Autonomous driving agent in



# unity

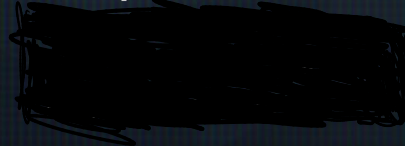
**Course:**

Special Topics in AI - Autonomous vehicles: Simulation framework to train intelligent agents (SS22)

**Lecturer:**

Walter Morales-Alvarez

**Group members:**



Retrieved from:

<https://www.filepicker.io/api/file/t84NGUdrRy69bwIK12MI>



# Table of contents

1. General information about state of research
2. Goal of our research
3. Testing environment
4. Driving Agent
5. Training process
6. Results
7. References



# Introduction

- Autonomous driving is very important today
- More and more attention as time passes
- Major improvements over the past years
- Still far from perfect



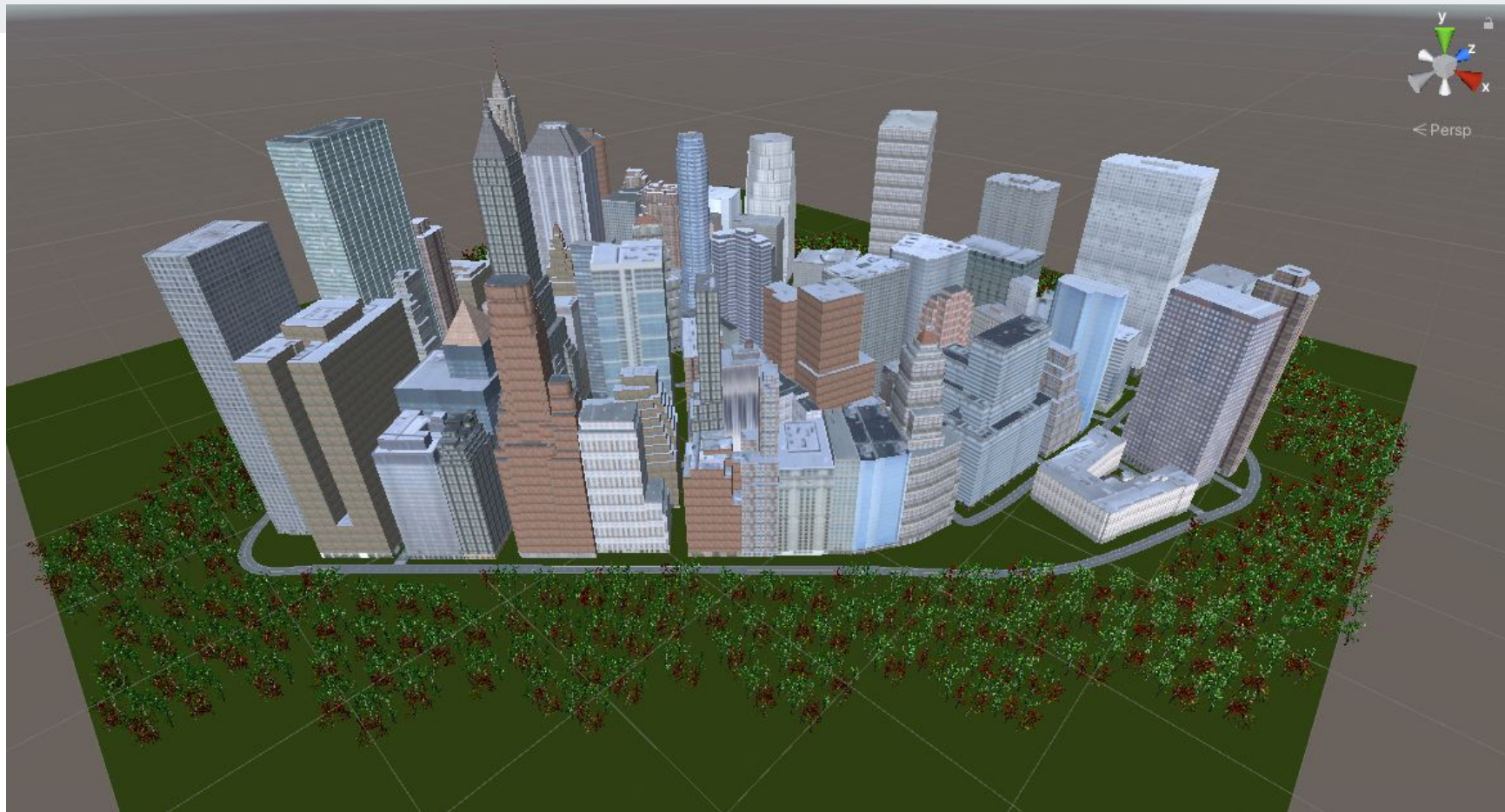
## Goal of the work

- Create a city to drive in
  - Traffic
  - Traffic-Lights
- Create an autonomous driving agent
- Agent should drive from point A to point B



## Testing environment: City

- City used as environment for testing
- Arcgis City engine is used
- 3D object of Manhattan
- Export to Unity



*Fig. 1: New York City. Part of Manhattan extracted with help of City Engine. Imported into Unity.*



## Testing environment: Traffic

- Needed for more realistic testing
- Simple-AI-Traffic package for Unity
- Define paths for self-driving vehicles with waypoints
- Implement traffic lights
- Some issues



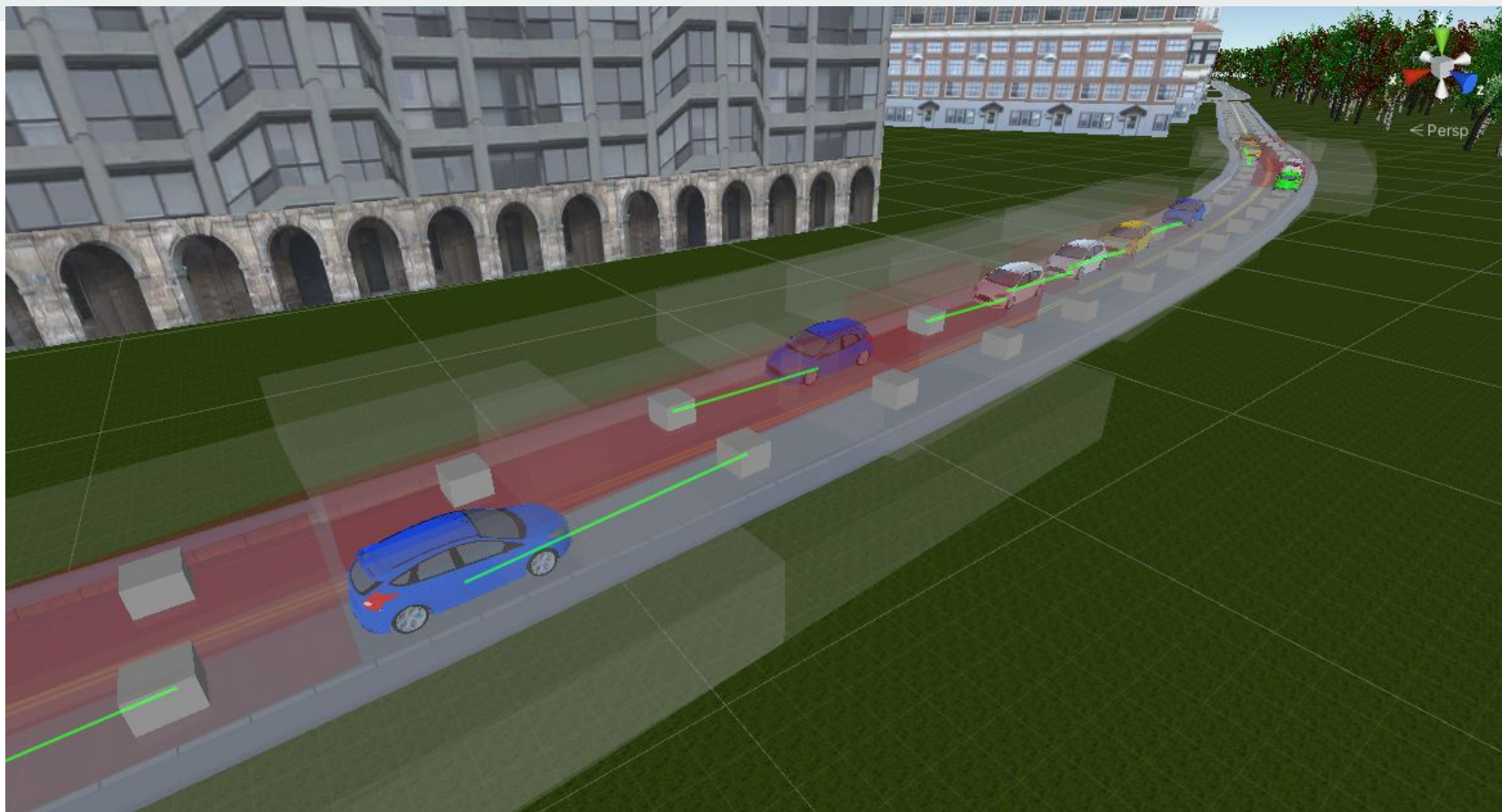


Fig. 2: Manhattan with from Simple-AI-Traffic package imported NPCs and waypoints.



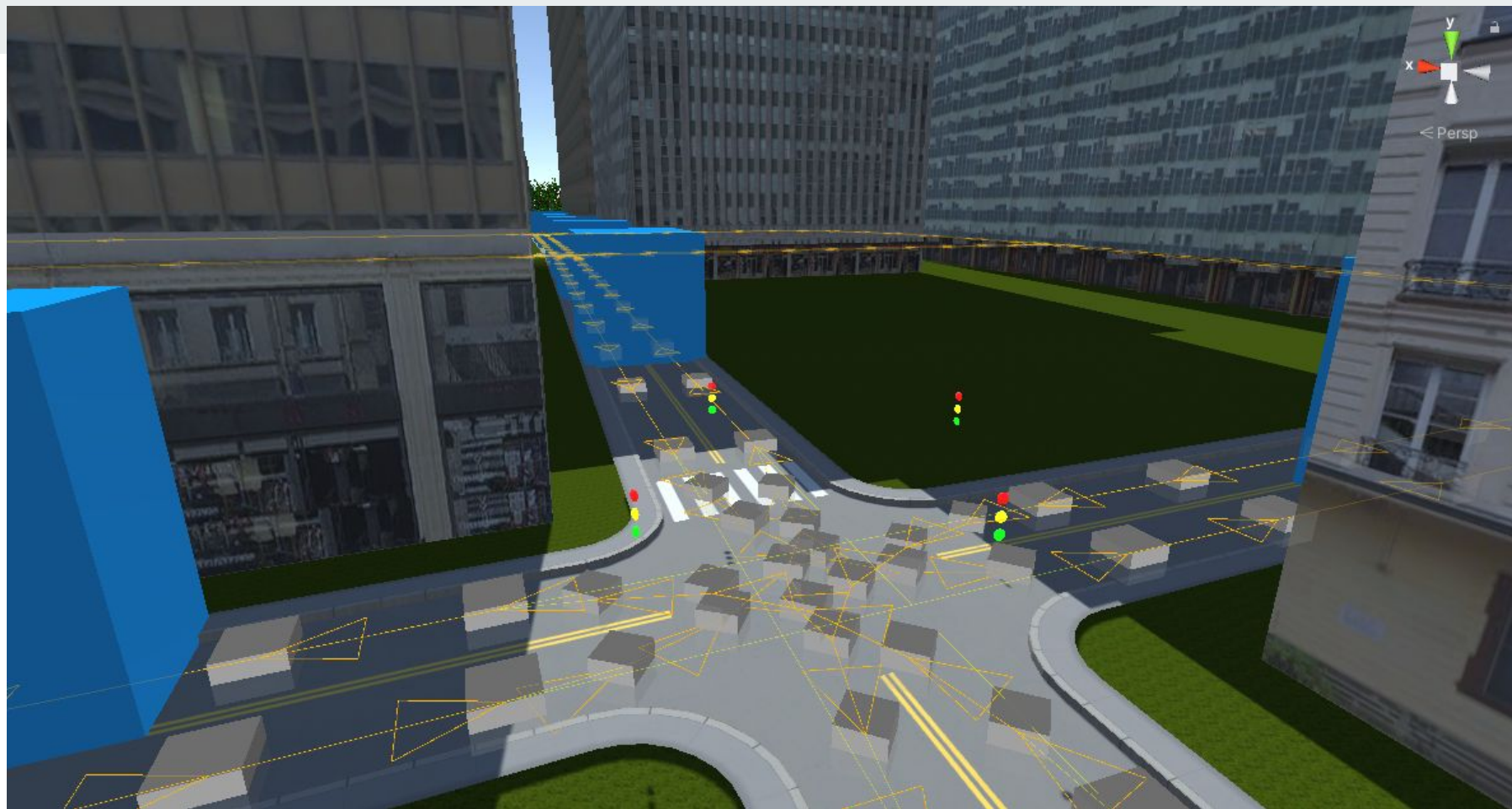


Fig. 3: Manhattan with implemented waypoints, traffic lights and spawn points for cars.



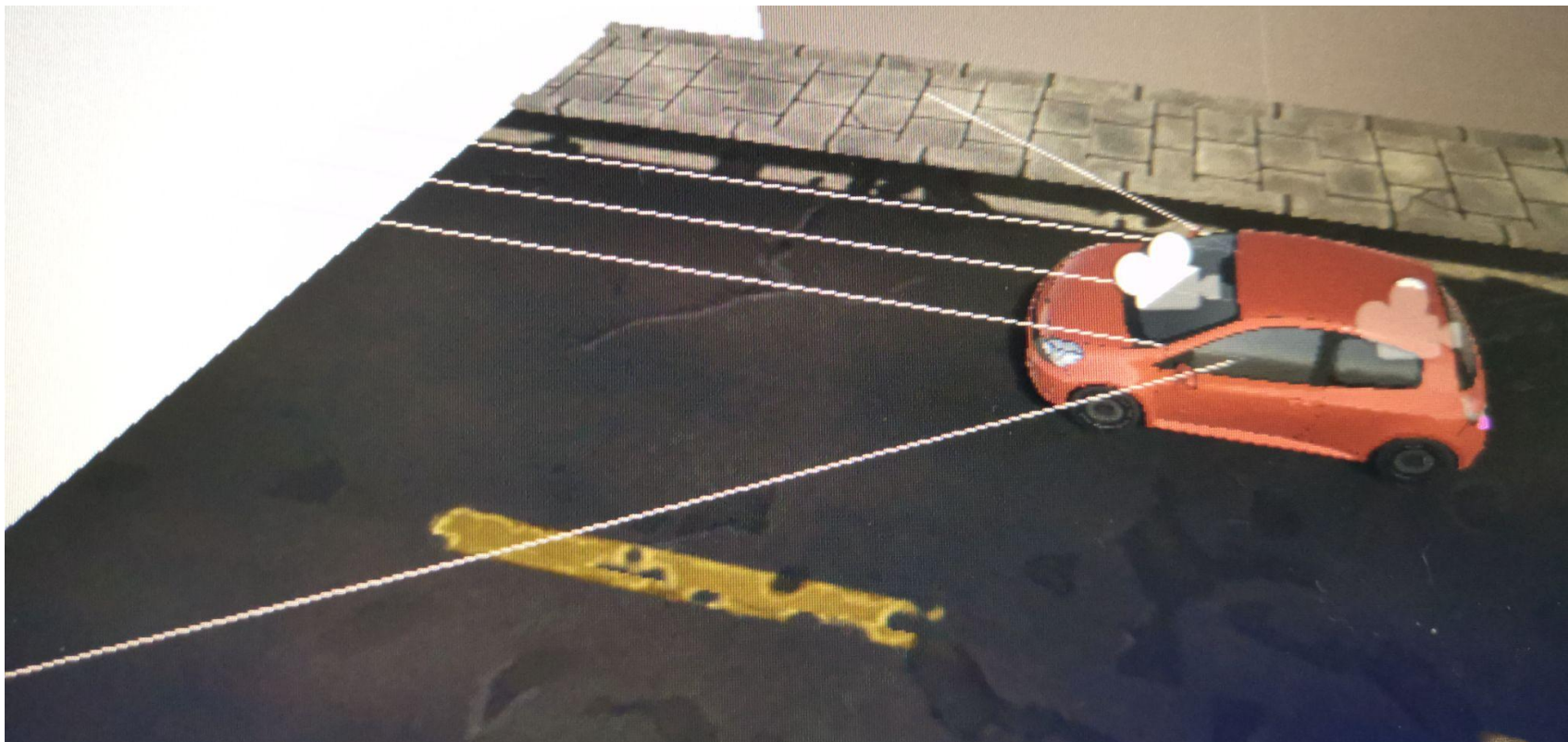
## Driving Agent

- Install ML-Agents package for Unity to make use of Reinforcement Learning
- Car design from the “carDemo19” package
- Behavioural script “CarAgent.cs” already given by the package but further modified
  - Reward/Penalty system
  - changed variable for breaking, as agent would start driving backwards



# Driving Agent: Script content

- Basics like steering and driving given through the lecture content.
- Sensor LIDAR:
  - measure distance between car agent and object
  - five sensor lines were implemented at the front



*Fig. 4: White LIDAR lines for measuring the distance between car agent and white wall/curb.*



# Driving Agent: Script content

## Rewards:

- for reaching “AI Traffic Waypoints” from the “AI Traffic” package. These objects got the tag “check” (reward=1 for each waypoint).
- Last waypoint of a route gets the tag “goal” such that the agent successfully ends the episode and gets a reward of 10.

## Penalties:

- to avoid action “do nothing”, we give a penalty of -10/MaxStep for each passing time step.
- when agent touches objects with tag=wall/Player/Untagged, reward is set to -1 and the episode ends.



## Driving Agent: Script content

- Penalties for crashing into objects
- Sidewalk
- Line in the middle of the road
- Other cars



## Driving Agent: Script content

For more details you can have a look at the “CarAgent.cs” script which is well provided with explanatory comments.





## Driving Agent: Sensors

- **LIDAR** as explained before
- **Two camera sensors**
  - at the front and back



## Training process

- Record demo using manual input
- Train agent based on demo
- No traffic to make it less complicated



## Results without traffic

- After training for roughly 4 hours
- Very wobbly (constantly turning left and right)
- Does not like to turn corners
- Sometimes stops at traffic lights, but for a very short time



## Results with traffic

- Traffic makes things more complicated
- Ignores traffic
- Same behavior like before



## Results with demo to turn left/right

- For this we recorded a demo where we just turn left/right
- Still fails to turn left
- Same behavior like before



## Driving agent in unity



**Thank you for your attention!**