PyData Global Impact Community Get-Together

ML and DL applied for Autonomous Vehicles

Applied Projects and Development

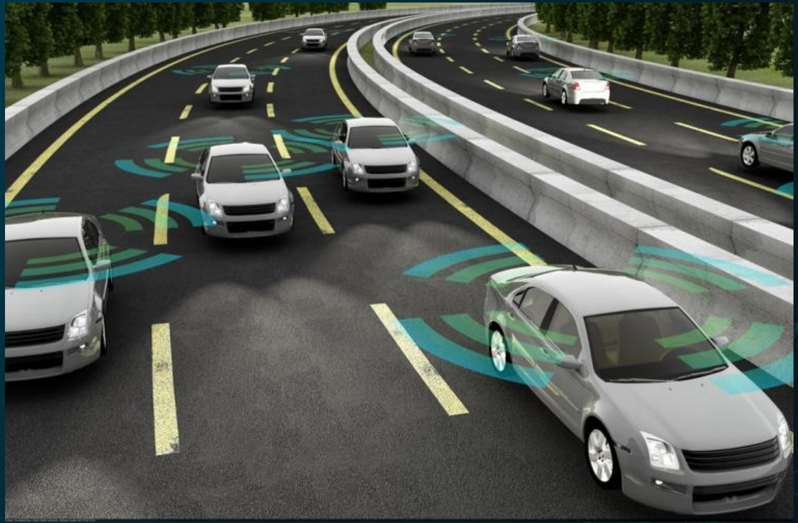
Marcelo Fernando Condori Mendoza

marcelof.condor@gmail.com

LinkedIn: https://www.linkedin.com/in/marcelo-fernando-condori-mendoza-46131645/

Background

Autonomous Driving has become one of the most popular research points in the field of artificial intelligence (AI). In the case of autonomous vehicles, it is based on **reinforcement learning algorithms** and **heuristic planning**. The proposed decision engine focuses on keeping autonomous vehicles running safely and efficiently.



Objectives

General

Implement a prototype that allows demonstrating driving in autonomous cars using the Carla Simulation tool in addition to demonstrating with a physical prototype the operation of some sensors that are applied in autonomous cars.

Secondary

Investigate the technology implemented in autonomous cars.

Analyze and understand the tool to be used for the simulation.

Install the tool and configure for simulation.

Test the respective system.

Secondary Objetives

Design an autonomous system based on the principles and fundamentals of mobile robotics, whose functionality, under the acquisition of a signal, determines its physical behavior, moving freely in an environment

Implement a physical structure of a mobile robot as an autonomous system, together with the systems that compose it, such as the movement system, the control system and the sensory system.

Software Tools

01 Carla Simulator

02 Unreal Engine

03 Docker

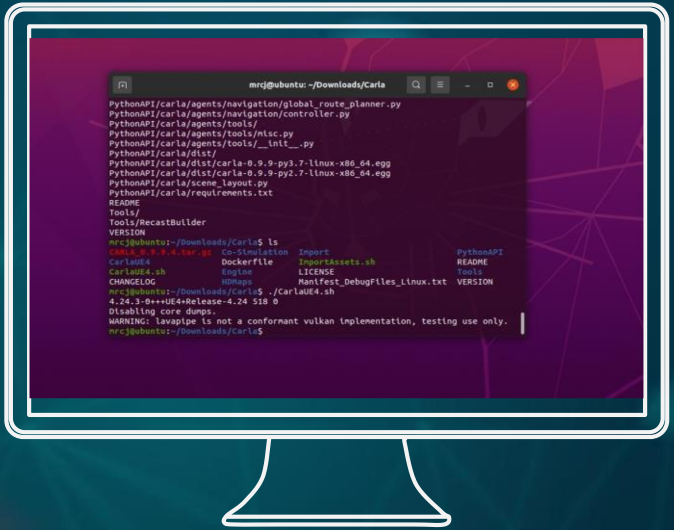
04 Arduino

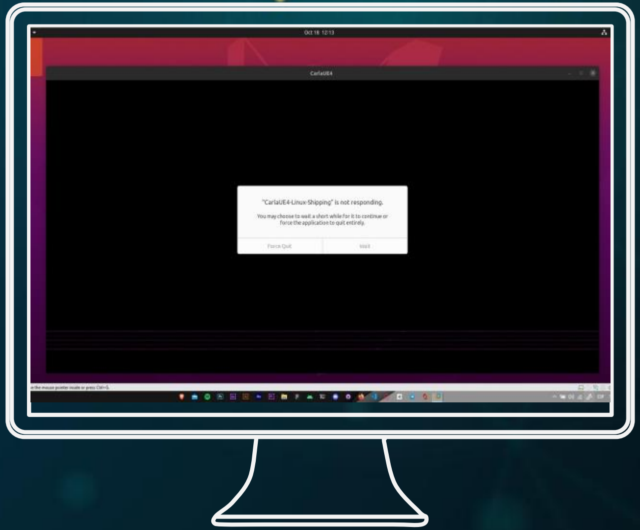
APPLIED DEVELOPMENT

Execution and Testing Carla Simulator in Ubuntu Operative System

Testing and Downloading Different Versions

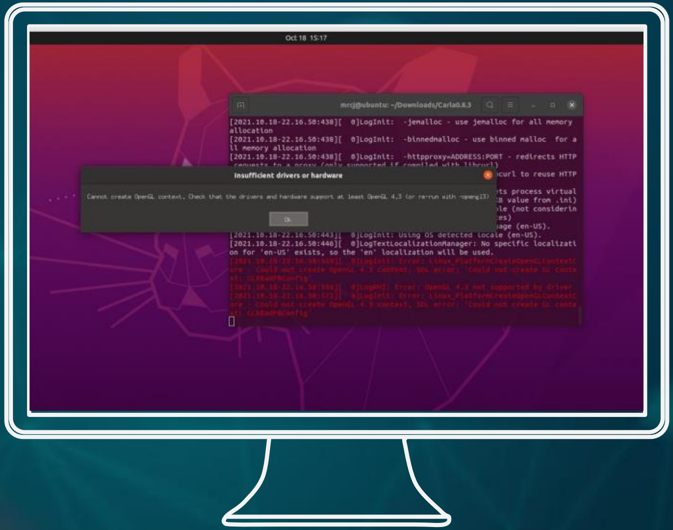
Executing Carla 0.9.9.4





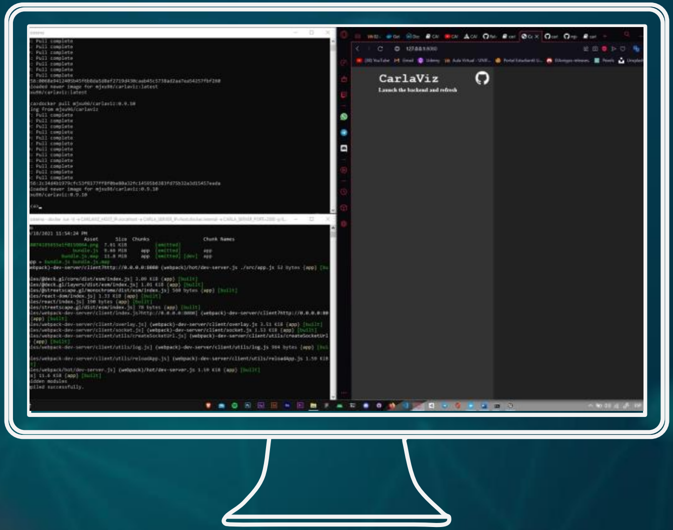
Visualizing the Trial of Carla 0.9.9.4

Executing Carla 0.6.0 - 0.8.3



Associated bug on OpenGL due to old graphics card version, element which helps 3D rendering.

Simulation Test using Docker Tool

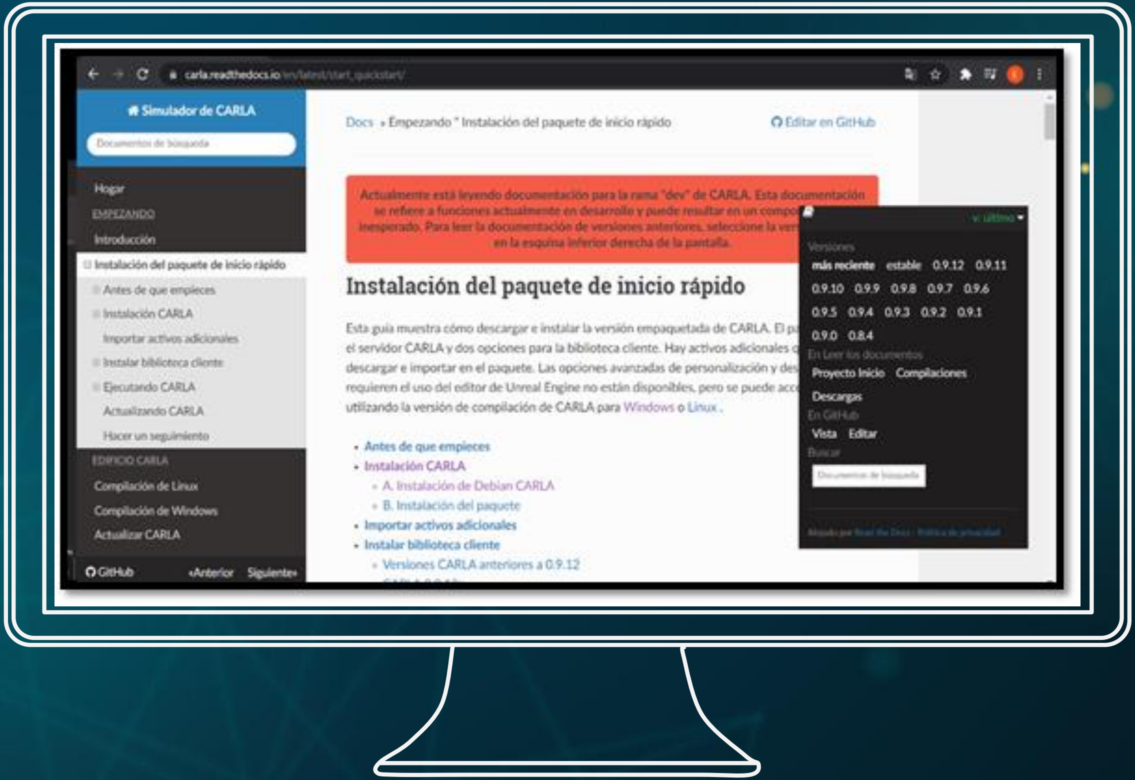


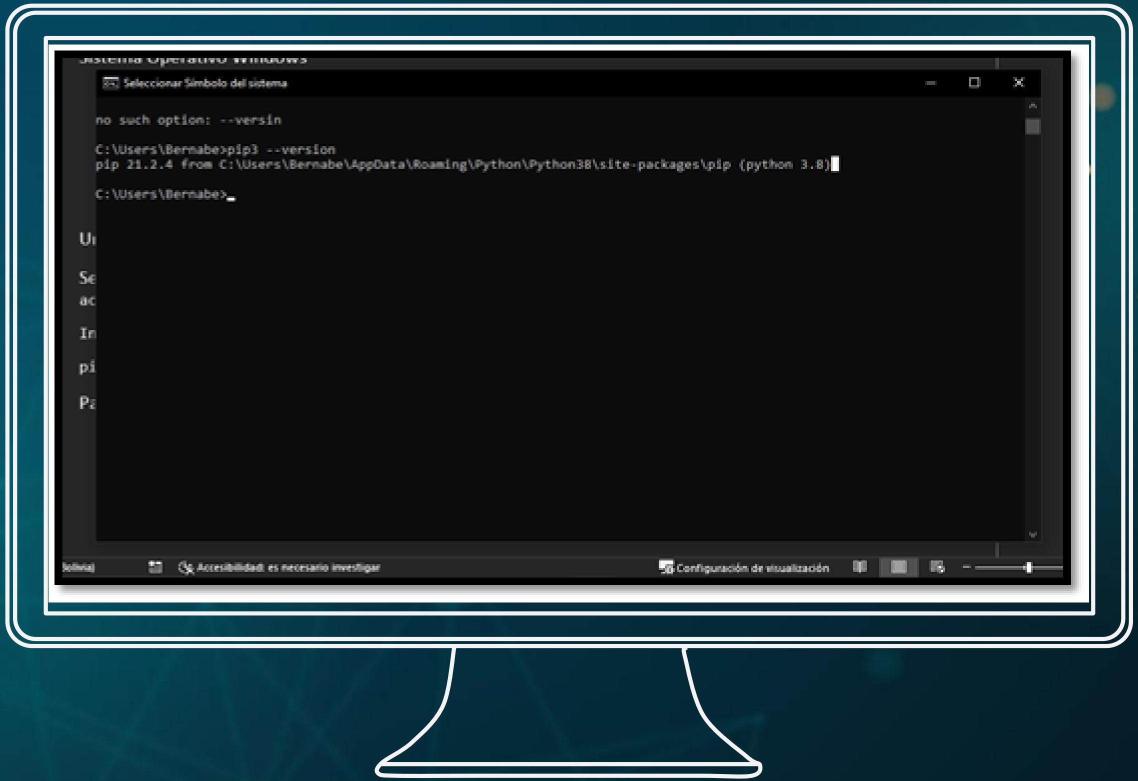
Carla Simulator database execution problem

Expected Result Using Docker

This option uses CarlaViz which is a plugin which allows the simulation to be viewed in a web browser. Next, it performs a basic representation of the scene based on lines and polylines.

Executing and Testing Carla Simulator in Windows Operating System


carla.readthedocs.io
Simulador de CARLA
Hogar
EMPEZANDO
Introducción
Instalación del paquete de inicio rápido
Antes de que empieces
Instalación CARLA
Importar activos adicionales
Instalar biblioteca cliente
Ejecutando CARLA
Actualizando CARLA
Hacer un seguimiento
EDIFICIO CARLA
Compilación de Linux
Compilación de Windows
Actualizar CARLA
GitHub
«Anterior
Siguiente»
Docs » Empezando » Instalación del paquete de inicio rápido
Editar en GitHub
Instalación del paquete de inicio rápido
Antes de que empieces
Instalación CARLA
A. Instalación de Debian CARLA
B. Instalación del paquete
Importar activos adicionales
Instalar biblioteca cliente
Versiones CARLA anteriores a 0.9.12
Versiones
más reciente
estable
0.9.12
0.9.11
0.9.10
0.9.9
0.9.8
0.9.7
0.9.6
0.9.5
0.9.4
0.9.3
0.9.2
0.9.1
0.9.0
0.8.4
Proyecto Inicio
Compilaciones
Descargas
Vista
Editar

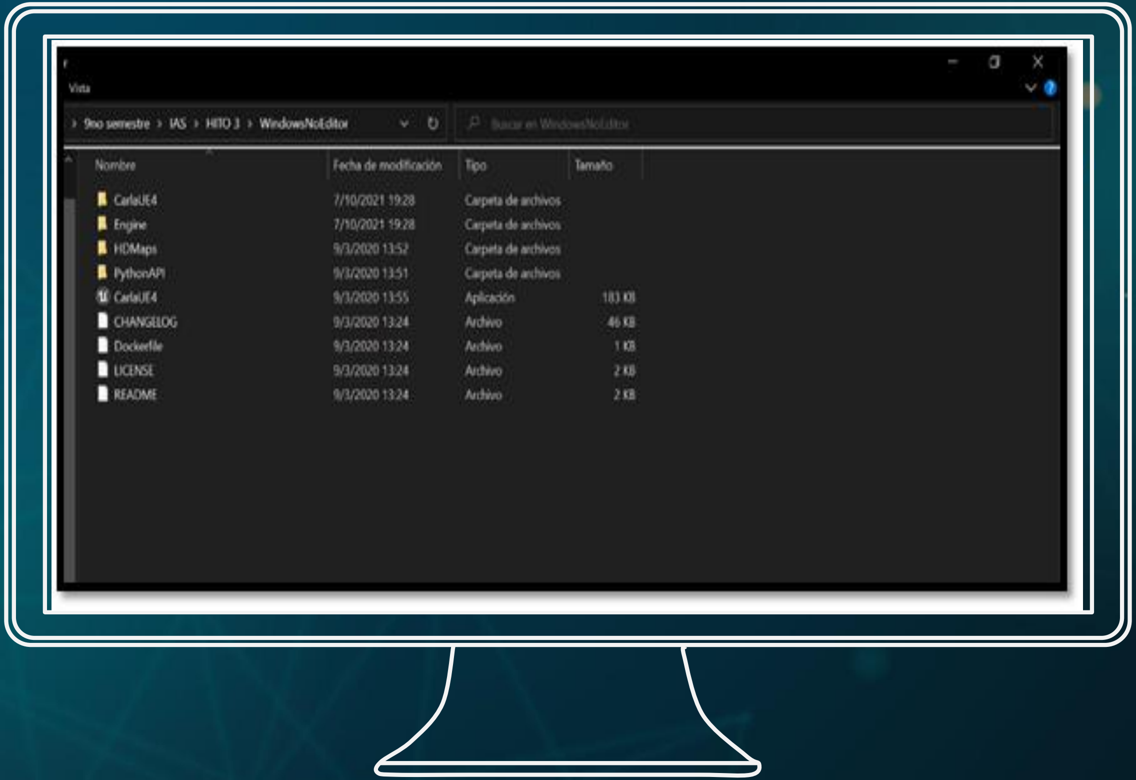

Seleccionar Símbolo del sistema
no such option: --versin
C:\Users\Bernabe>pip3 --version
pip 21.2.4 from C:\Users\Bernabe\AppData\Roaming\Python\Python38\site-packages\pip (python 3.8)
C:\Users\Bernabe>
Accesibilidad: es necesario investigar
Configuración de visualización

Instalación CARLA

Hay dos métodos para descargar e instalar CARLA como paquete:

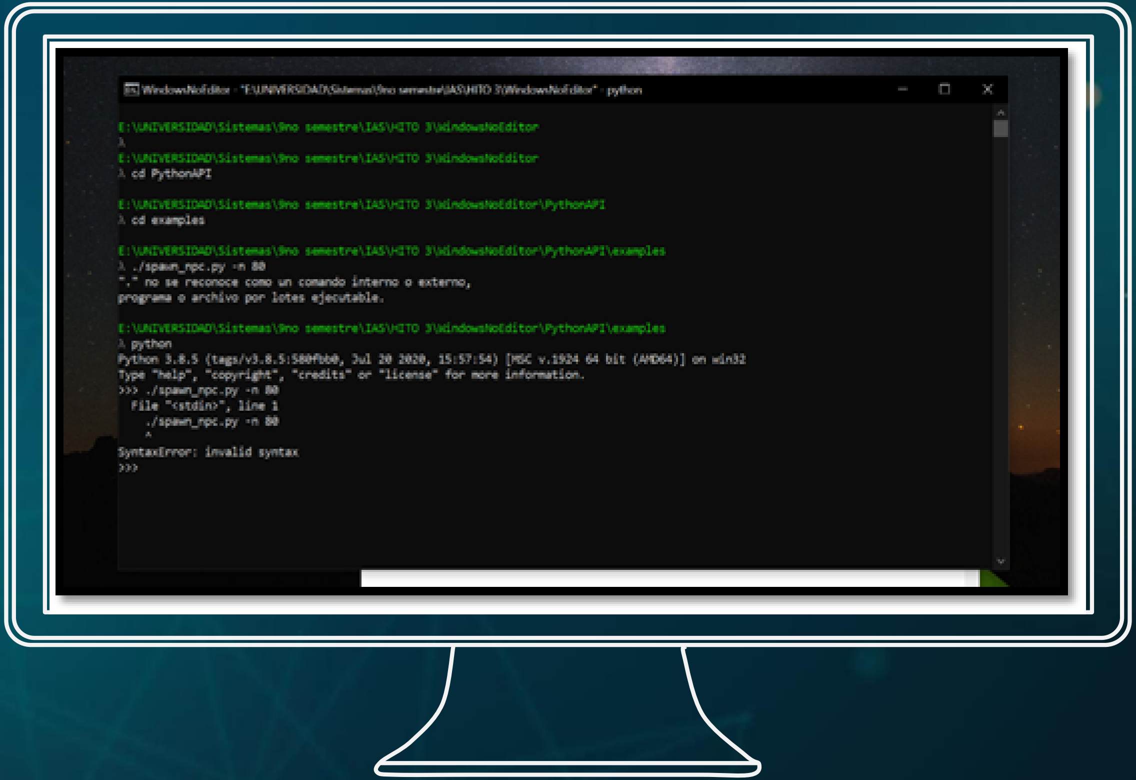
A) Descargue el paquete Debian.

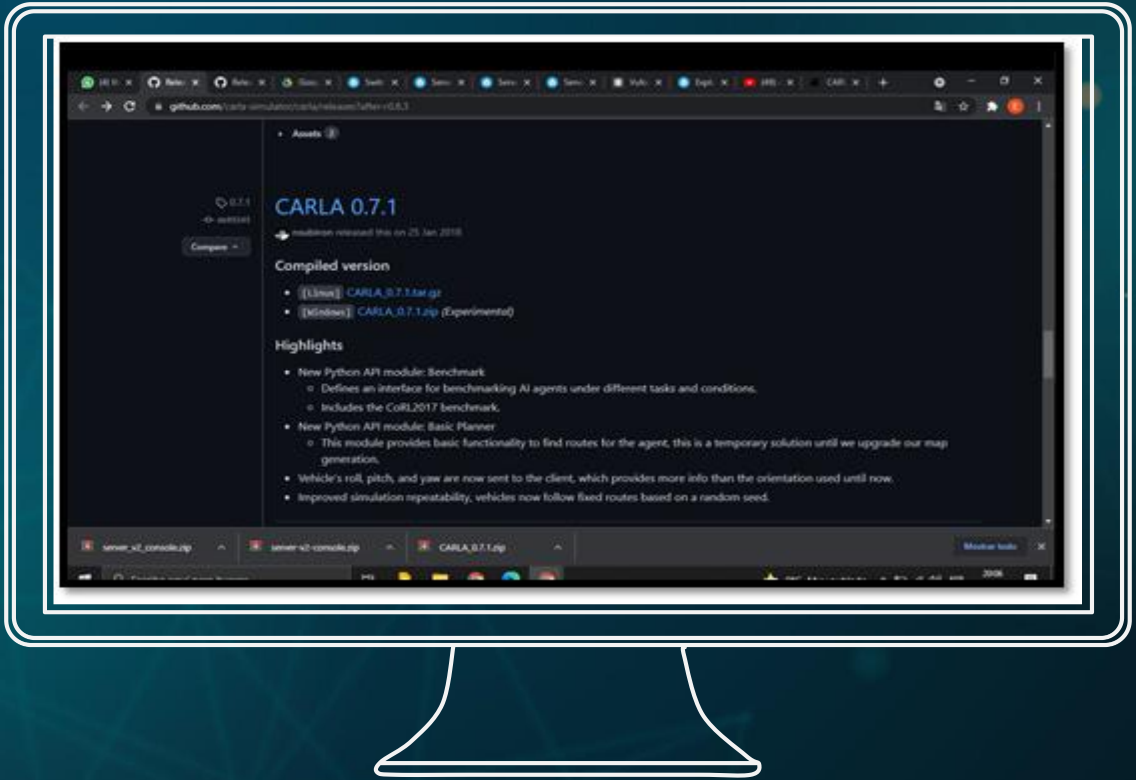
B) Descargue el paquete de GitHub.

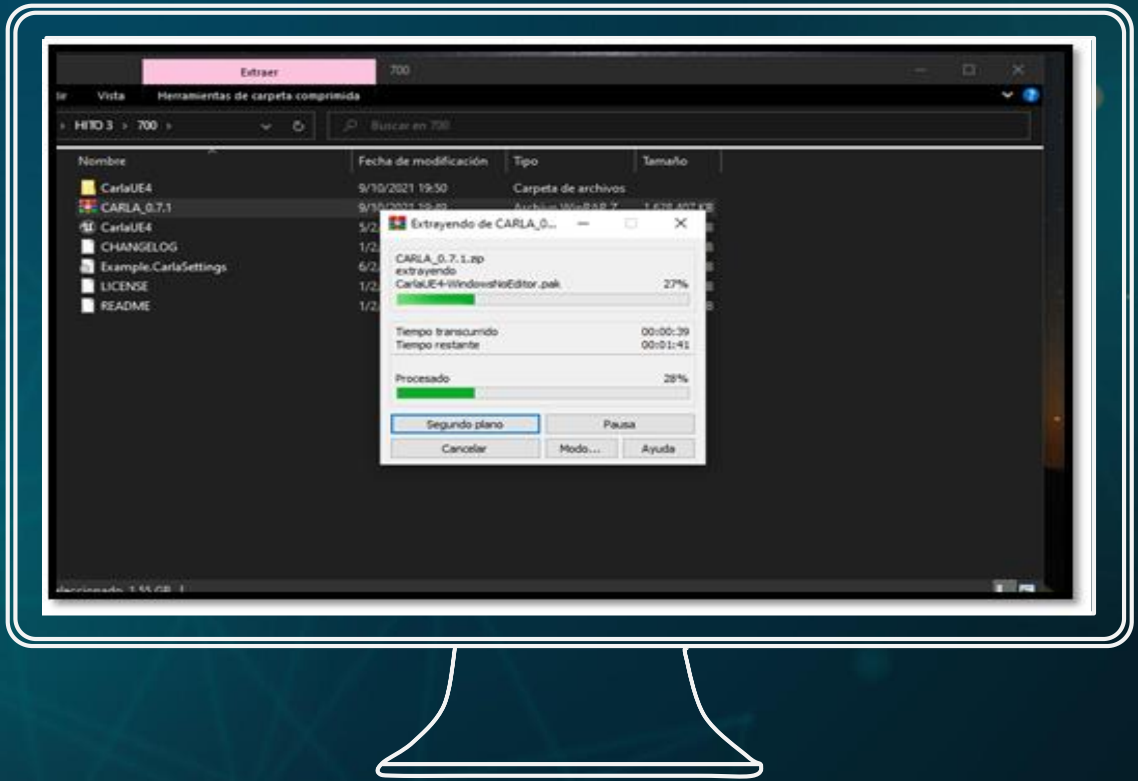


Vista
9no semestre › IAS › HITO 3 › WindowsNoEditor
Nombre
Fecha de modificación
Tipo
Tamaño
CarlaUE4 7/10/2021 19:28 Carpeta de archivos
Engine 7/10/2021 19:28 Carpeta de archivos
HDMaps 9/3/2020 13:52 Carpeta de archivos
PythonAPI 9/3/2020 13:51 Carpeta de archivos
CarlaUE4 9/3/2020 13:55 Aplicación 183 KB
CHANGELOG 9/3/2020 13:24 Archivo 46 KB
Dockerfile 9/3/2020 13:24 Archivo 1 KB
LICENSE 9/3/2020 13:24 Archivo 2 KB
README 9/3/2020 13:24 Archivo 2 KB


Escribe aquí para buscar


WindowsNoEditor - "E:\UNIVERSIDAD\Sistemas\9no semestre\IAS\HITO 3\WindowsNoEditor" - python
E:\UNIVERSIDAD\Sistemas\9no semestre\IAS\HITO 3\WindowsNoEditor
λ
E:\UNIVERSIDAD\Sistemas\9no semestre\IAS\HITO 3\WindowsNoEditor
λ cd PythonAPI
E:\UNIVERSIDAD\Sistemas\9no semestre\IAS\HITO 3\WindowsNoEditor\PythonAPI
λ cd examples
E:\UNIVERSIDAD\Sistemas\9no semestre\IAS\HITO 3\WindowsNoEditor\PythonAPI\examples
λ ./spawn_npc.py -n 80
"." no se reconoce como un comando interno o externo,
programa o archivo por lotes ejecutable.
E:\UNIVERSIDAD\Sistemas\9no semestre\IAS\HITO 3\WindowsNoEditor\PythonAPI\examples
λ python
Python 3.8.5 (tags/v3.8.5:580fbb0, Jul 20 2020, 15:57:54) [MSC v.1924 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> ./spawn_npc.py -n 80
File "<stdin>", line 1
./spawn_npc.py -n 80
^
SyntaxError: invalid syntax
>>>

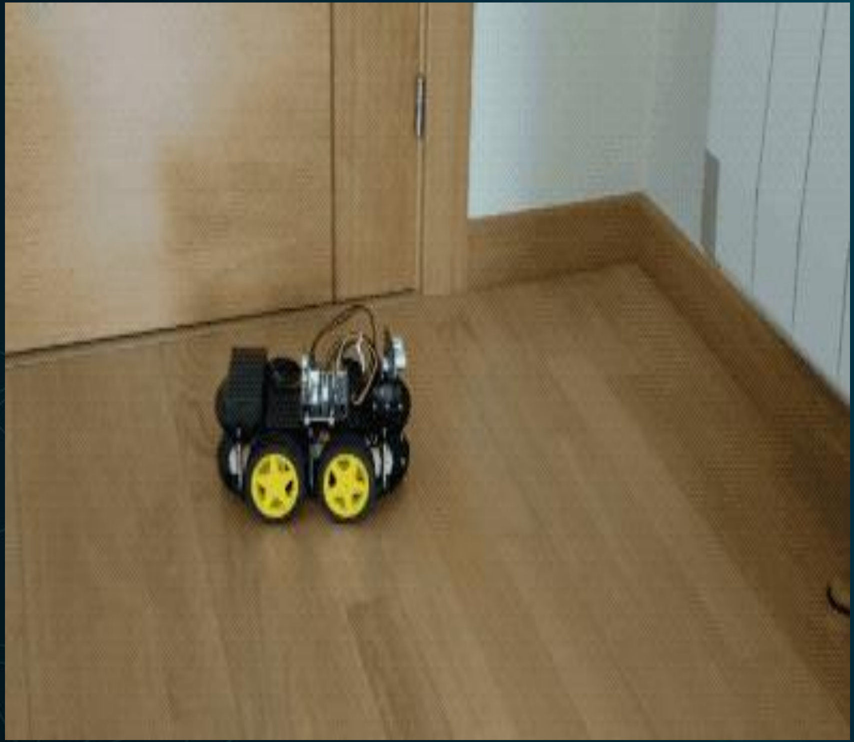

CARLA 0.7.1
Compiled version
[Linux] CARLA_0.7.1.tar.gz
[Windows] CARLA_0.7.1.zip (Experimental)
Highlights
New Python API module: Benchmark
Defines an interface for benchmarking AI agents under different tasks and conditions.
Includes the CoRL2017 benchmark.
New Python API module: Basic Planner
This module provides basic functionality to find routes for the agent, this is a temporary solution until we upgrade our map generation.
Vehicle's roll, pitch, and yaw are now sent to the client, which provides more info than the orientation used until now.
Improved simulation repeatability, vehicles now follow fixed routes based on a random seed.
CARLA_0.7.1.zip


Extraer
700
Vista
Herramientas de carpeta comprimida
HITO 3 › 700 ›
Nombre
Fecha de modificación
Tipo
Tamaño
CarlaUE4
9/10/2021 19:50
Carpeta de archivos
CARLA_0.7.1
CarlaUE4
CHANGELOG
Example.CarlaSettings
LICENSE
README
Extrayendo de CARLA_0...
CARLA_0.7.1.zip
extrayendo
CarlaUE4-WindowsNoEditor.pak
27%
Tiempo transcurrido
00:00:39
Tiempo restante
00:01:41
Procesado
28%
Segundo plano
Pausa
Cancelar
Modo...
Ayuda

Carla Application

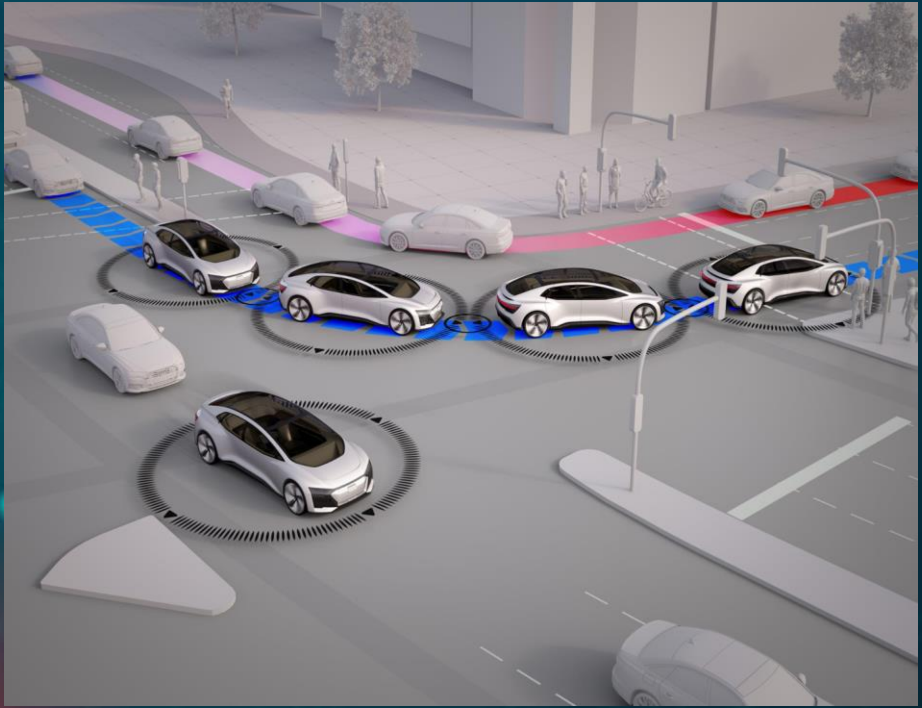
Objects Detection

Traffic Signals Compliance



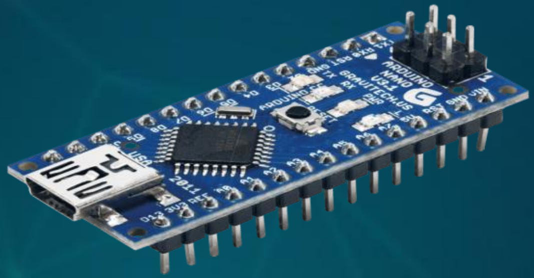
Small Auto that avoids obstacles.

Application of the Small Auto that avoids obstacles.

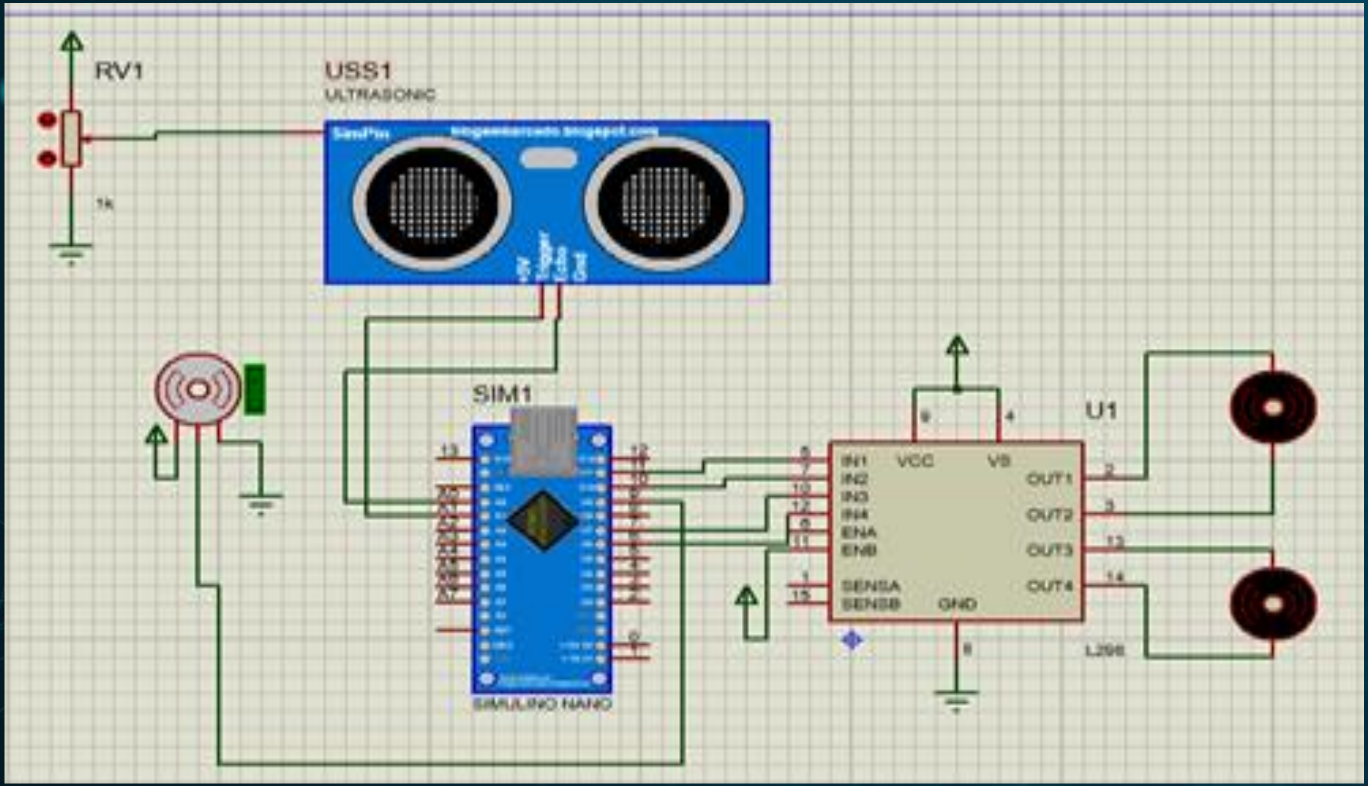


Through the Avoid Obstacles function, the robot must be able to travel a circuit avoiding all the obstacles placed on the route. For this purpose, the robot will have an ultrasonic sensor HCSR04

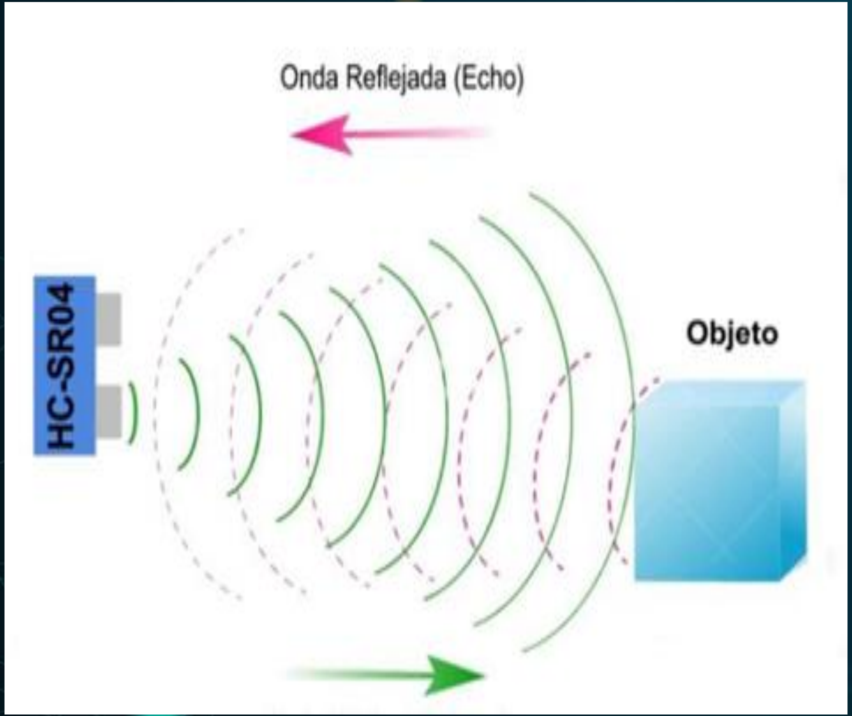
Implemented Hardware Tools



Proteus Simulation



What is the Main Functioning Principle?



1. Send a pulse of approximately 10 uS through the Trigger pin

2. The sensor will send eight 40KHz pulses and set the Echo pin in HIGH. This is necessary to trigger the timer.

3. When the sensor receives the reflected pulse, it will set the Echo pin in LOW and thus end the time counting.

4. The distance can be obtained using the speed of sound (340 m/s) multiplying it by the time obtained in us x 0.017

Formula for the Ultrasonic Sensor

$$\text{Velocidad} = \frac{\text{Espacio}}{\text{Tiempo}} \longrightarrow \text{Espacio} = \text{Velocidad x Tiempo}$$

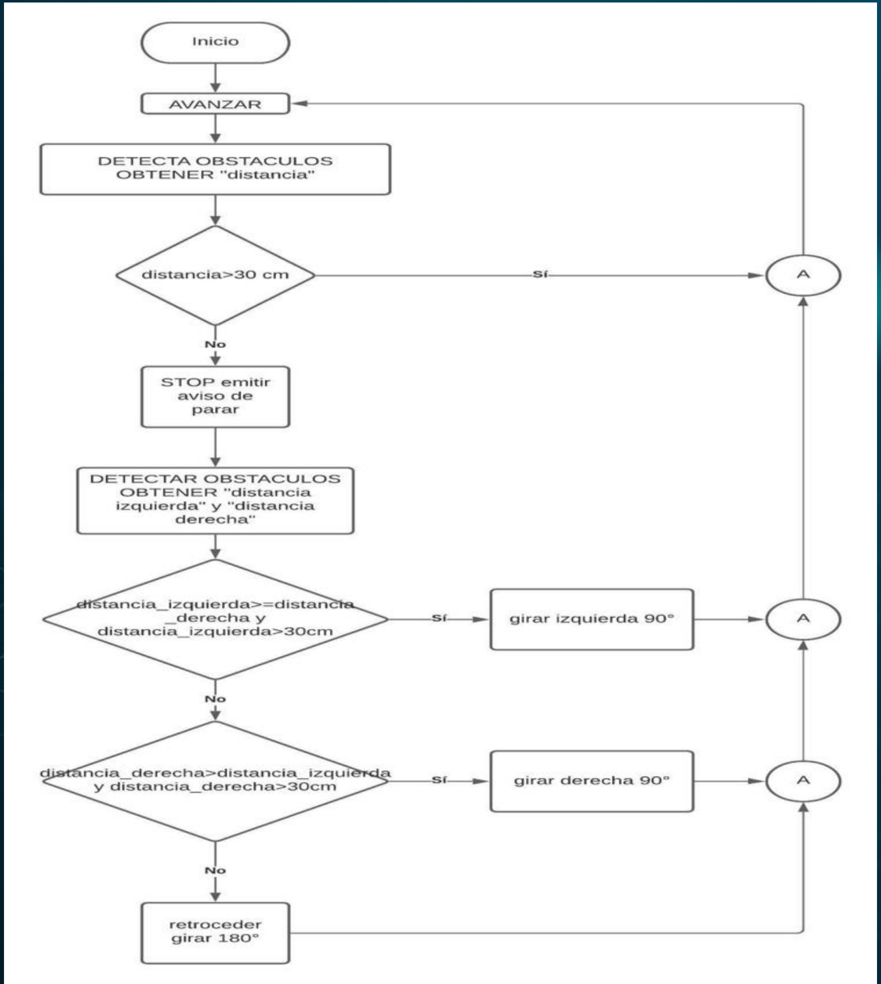
Velocidad del sonido = 343 m/s = 0.0343 cm/μs

$$\text{Espacio} = 0.0343 \text{ x Tiempo}$$

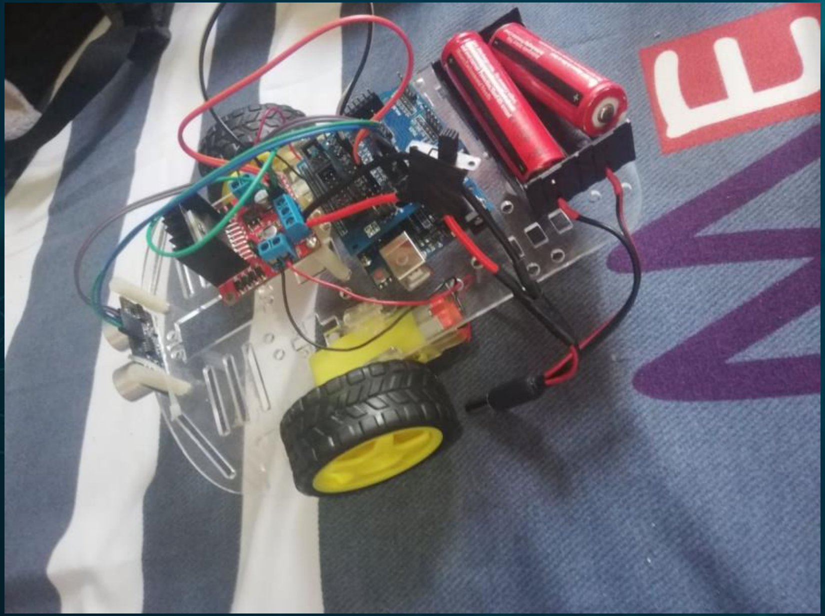
*Pero como la onda ha recorrido el camino dos veces (ida y vuelta)hay que dividir entre dos para conocer la distancia a la que se encuentra el objeto.

$$\boxed{\text{Espacio} = 0.01715 \text{ x Tiempo}}$$

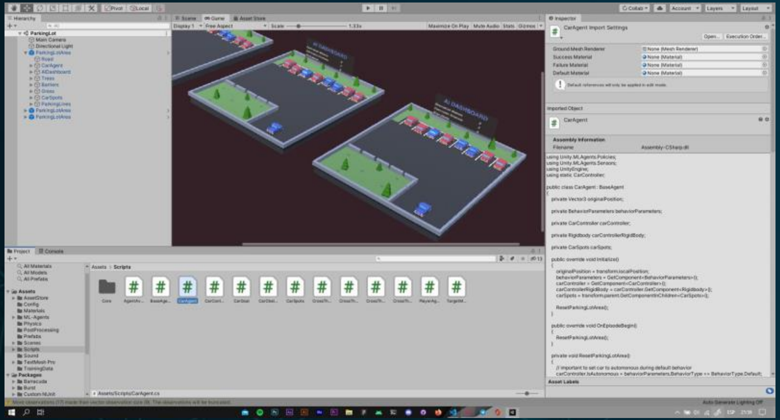
Workflow for the Robot that avoids Obstacles

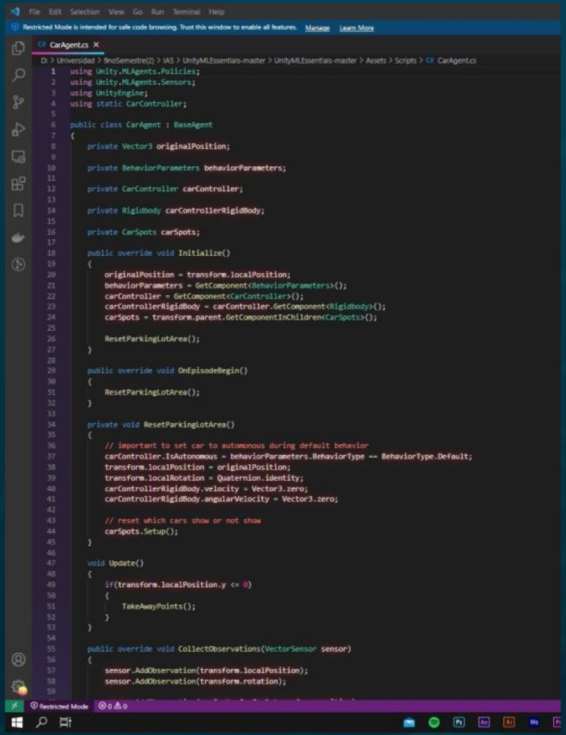


Implemented Hardware Circuitry

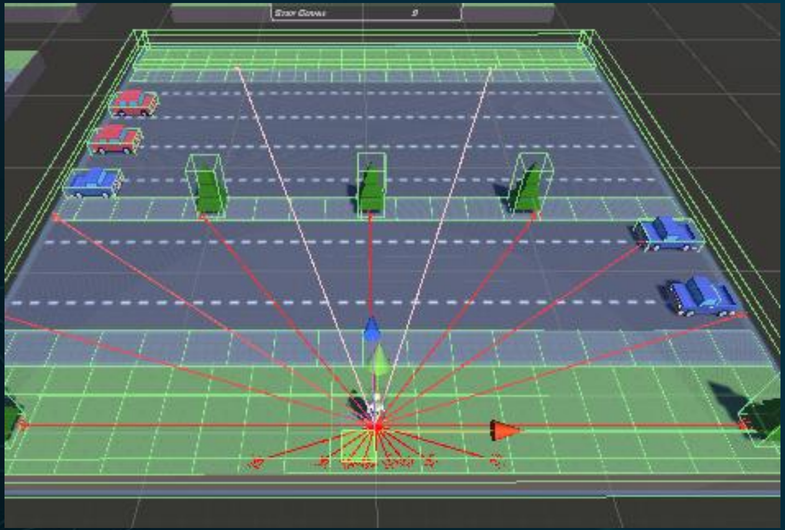


Other Applications
Intelligent Agents for Automobile Parking





Funcionamiento
Automóviles Inteligentes



```
public override void OnActionReceived(float[] vectorAction)
{
    if (moveInProgress)
        return;

    direction = Mathf.FloorToInt(vectorAction[0]);

    switch (direction)
    {
        case 0: // idle
            moveTo = transform.localPosition;
            moveToDirection = MoveToDirection.Idle;
            break;
        case 1: // left
            moveTo = new Vector3(transform.localPosition.x - s
            moveToDirection = MoveToDirection.Left;
            moveInProgress = true;
            break;
        case 2: // right
            moveTo = new Vector3(transform.localPosition.x + s
            moveToDirection = MoveToDirection.Right;
            moveInProgress = true;
            break;
        case 3: // forward
            moveTo = new Vector3(transform.localPosition.x, tr
            moveToDirection = MoveToDirection.Forward;
            moveInProgress = true;
            break;
    }
}
```

Conclusions

It was possible to install the CARLA environment for windows in its first version for the operating system, it was also possible to observe the fluid traffic that exists in carla to be able to study the movements of autonomous cars, which comply with what was proposed, such as: Autonomous cars that respect the lines of the driving lanes by not leaving their lanes, report the traffic if there are cars ahead the car stops and finally respect the rules of the traffic light if it is red it stops and if it is green it advances.

Based on the tests carried out on the Linux-Ubuntu system, it is observed that, due to some installation omission of some dependency, or trying to apply new versions, a lot of performance is necessary on the part of the computer for its execution, and based on Docker is considered to be a very stable solution for low to medium performance computers.

Conclusions —

The implementation of mobile robots can make very viable solutions in technological developments with low costs and high benefits, referring to industrial needs, as well as with customized prototypes for autonomous cars. We can understand and simulate their operation to see how autonomous cars react in an environment that is controlled and scaled.

Thank you very much for your attention!