

**УДК 004.94**

*Мужайло Михаил Константинович,*

**Студент**

**Государственный Университет Аэрокосмического Приборостроения**

**Г. Санкт-Петербург**

*Научный руководитель: Савельев А.И., к.т.н., доцент*

**Государственный Университет Аэрокосмического Приборостроения**

**Г. Санкт-Петербург**

## **СОЗДАНИЕ МОДЕЛИРУЕМОГО ПРОСТРАНСТВА ДЛЯ НАЗЕМНОГО ТРАНСПОРТНОГО СРЕДСТВА**

*Аннотация:* В статье представлен способ создания моделируемого пространства, посредством использования двух программ. Цель работы состоит в том, чтобы создать базовое моделируемое пространство для тестирования и отладки алгоритмов управления робототехническим средством.

*Ключевые слова:* моделируемое пространство, виртуальное моделирование, киберфизические пространства, пространство для тестирования алгоритмов

**Muzhailo Mikhail Konstantinovich,**

**Student**

**State University of Aerospace Instrumentation**

**St. Petersburg**

*Scientific supervisor: Savelyev A.I., Candidate of Technical Sciences,*

**Associate Professor**

## **CREATING A SIMULATED SPACE FOR A GROUND VEHICLE**

*Abstract:* The article presents a way to create a simulated space by using two programs. The purpose of the work is to create a basic simulated space for testing and debugging algorithms for controlling a robotic vehicle.

*Keywords:* simulated space, virtual modeling, cyberphysical spaces, space for testing algorithms

Сегодня транспортные средства настолько плотно интегрированы в жизнь человека, что представить жизнь без них кажется невозможным.

Одной из наиболее актуальных проблем является задача автоматического определения состояния дорожной ситуации и своевременного оповещения водителя и реагирования на них. Крайне важным аспектом является разработка системы распознавания дорожных знаков и разметки дорожного полотна.

Но, такую систему все равно необходимо где-либо тестировать и разрабатывать, прежде чем внедрять в реальный транспорт. Поэтому, в области автомобильных систем беспилотной езды используются симуляции реального пространства. Симуляции крайне необходимы для ускорения способов проектирования во многих областях промышленности, автомобильная не является исключением.

Проведение исследований на виртуальных объектах еще на стадии тестирования обеспечит выявление алгоритмических ошибок модели, что в разы сокращает затраты времени и финансов.

На данный момент есть уже несколько алгоритмов для решения задач симуляции и автономного управления транспортом, но все они имеют свои недостатки. Такие как: скорость распознавания, точность, качество симуляции, оптимизацию и т.д. Следовательно, можно выделить проблему – на данный момент все также есть необходимость в создании качественного модельного пространства и алгоритмов управления робототехническим средством в условиях дорожного движения.

Актуальность работы заключается в том, что на текущий момент просто необходимо совершенствовать систему безопасности дорожного движения из-за колоссального количества ДТП и несчастных случаев по причине неверно или невовремя принятых решений человеком.

Для того, чтобы создать систему управления робототехническим средством в городских условиях сначала необходимо воссоздать эти условия в 3D-среде.

В качестве моделируемого пространства будет использована дорожная карта, что представляет собой дорогу, имеющую разную геометрию и разметку.

Этапы генерации сцены следующие:

1. Создание дорог в программе RoadRunner
2. Создание дополнительных объектов визуализации (при необходимости)

3. Задание трафика и зон навигации для прочих автомобилей и пешеходов (при необходимости)

4. Импорт созданного проекта в симулятор CARLA при помощи Docker-образа

Для создания моделируемого пространства воспользуемся симулятором программным обеспечением RoadRunner.

RoadRunner – это интерактивный редактор, который позволяет создавать 3D-сцены для моделирования и тестирования автоматизированных систем вождения. Его можно использовать для создания дорожных макетов и сопутствующей информации об открытом движении и геометрии.



Рисунок 1 – Интерфейс программы RoadRunner

Программа имеет огромное количество встроенных пресетов дорог и материалов, а также 3D-Объектов. Используя их необходимо создать дорогу. Процесс создания показан на рисунке 2.

Карту необходимо создавать с исходными координатами (0,0). В противном случае это может повлечь за собой сбой в работе CARLA Simulator при попытке создания трафика или иных объектов.

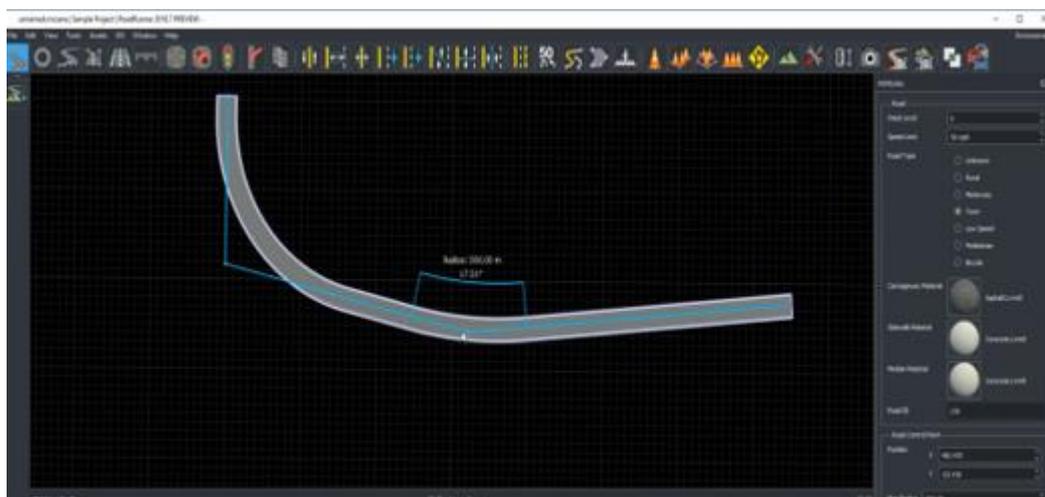


Рисунок 2 – Создание дороги

Созданная дорога имеет встроенную и нанесенную разметку. При необходимости её можно заменить.

Необходимо следить за всей геометрией и соединениями, потому что иногда объекты могут состыковываться не корректно.



Рисунок 3 – Пример правильной геометрии(слева) и неверной(справа)

Когда карта будет создана в программе RoadRunner необходимо провести визуализацию OpenDRIVE для предпросмотра дорожного движения.



Рисунок 4 – Визуализация дорожного движения в пределах перекрестка

В случае успешной визуализации и отсутствия ошибок карту необходимо сохранить и в последствии экспортировать в поддерживаемые симулятором CARLA формат (.fbx, .xml, .xodr).

Для экспорта необходимо перейти в ветку File/Export/CARLA и нажать на соответствующую кнопку.

В результате работы в программе RoadRunner была создана многоуровневая дорожная трасса, что отражена на рисунках 5-7.



Рисунок 5 – Карта общим планом



Рисунок 6 – Один из перекрестков, крупным планом

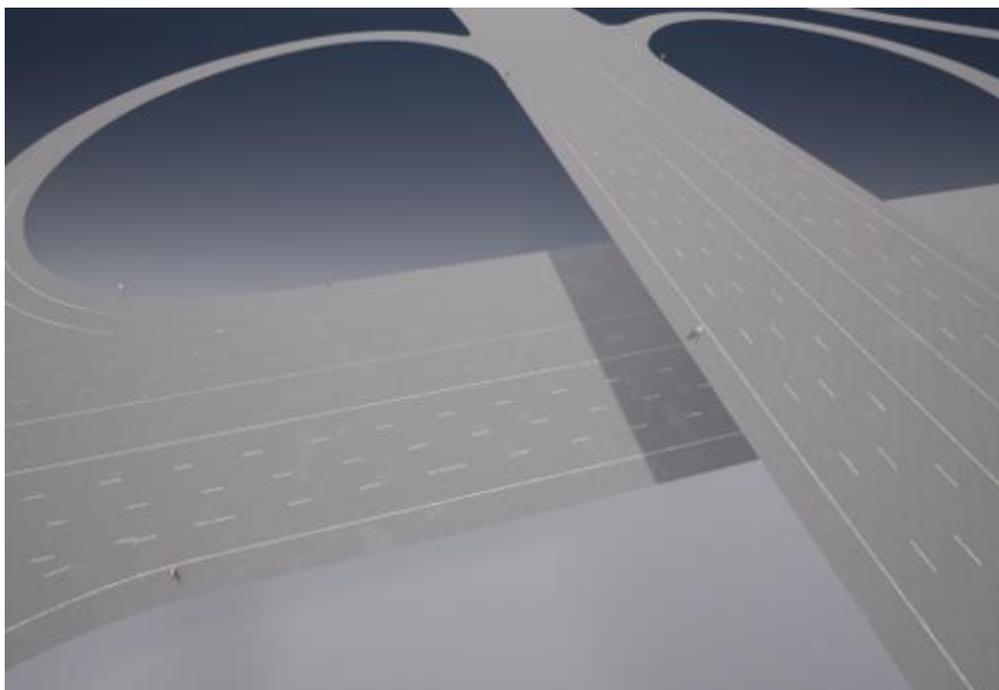


Рисунок 7 – Разноуровневая транспортная развязка

Полученные файлы необходимо импортировать в пакет симулятора CARLA. Для этого разработчиками данного симулятора был создан Docker-образ с редактором Unreal Engine, с открытым исходным кодом.

Запустить данный образ можно при помощи следующей команды:

Листинг №1 – Запуск Docker-образа

```
python docker_tools.py --input ~/ C:\Users\Mikhail\Desktop\carla\Map --  
output ~/ C:\Users\Mikhail\Desktop\carla\Map_new --packages map_package
```

В результате успешного выполнения данной программы будет создан архив `map_package.tar.gz` по директории, указанной в коде.

В конечном итоге, необходимо распаковать данный архив в папку с картами симулятора.

Для начала необходимо запустить программу CARLA.

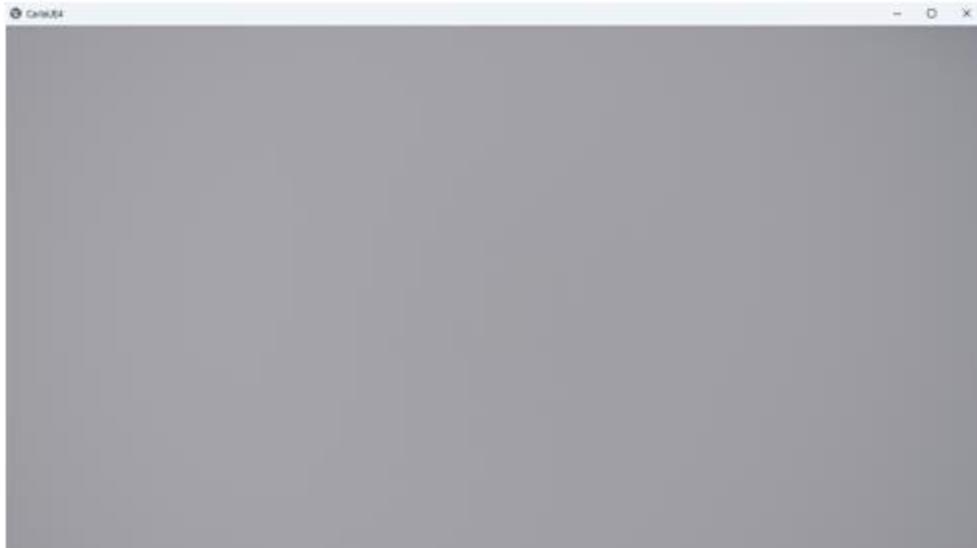


Рисунок 8 – Интерфейс симулятора

На данном этапе в симулятор не подгружено ни одной карты, поэтому имеется лишь пустая сцена.

Для того, чтобы как-либо взаимодействовать с симулятором – необходимо использовать python-скрипты.

На данном этапе необходимо написать скрипт, который обеспечит загрузку карты в программу. Скрипт представлен в листинге №2.

Листинг №2 – Скрипт для загрузки карты в симулятор

```
import glob
import os
import sys
import carla

try:
    client = carla.Client('localhost', 2000)
    client.set_timeout(5.0)
    world = client.load_world('EmptyTown')
```

В результате выполнения скрипта создается сервер CARLA с созданной ранее картой “EmptyTown”.

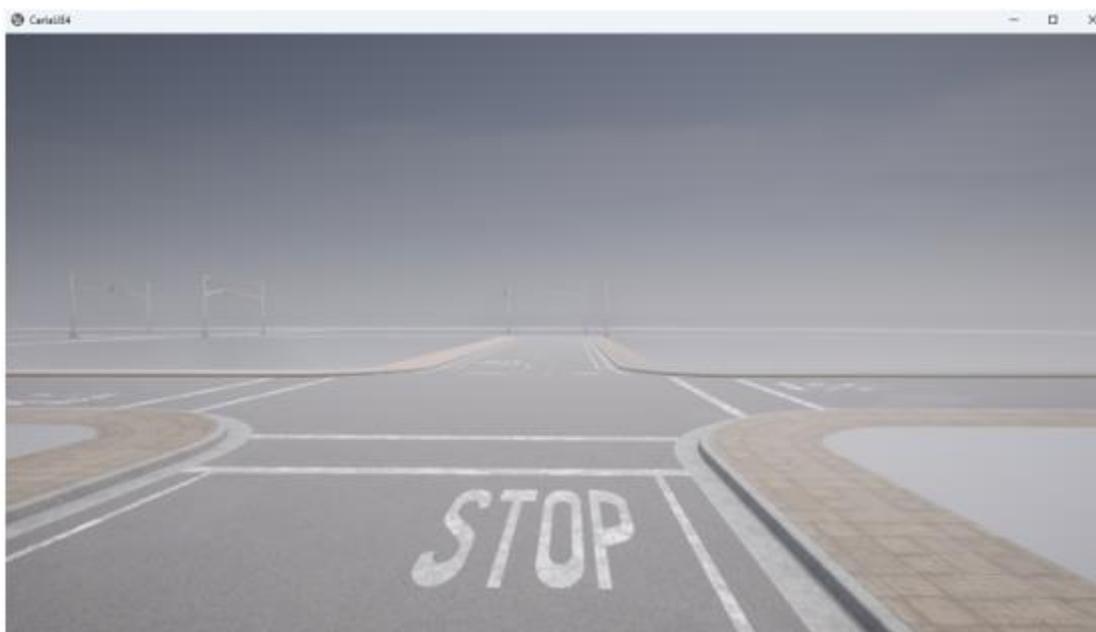


Рисунок 9 – Успешно загруженная карта в симулятор

В следствие всех выполненных действий в данной работе было успешно создано и импортировано в симулятор моделируемое пространство, которое представляет собой дорожное полотно с разметкой и сложной геометрией.

В последствие данное пространство можно использовать для тестирования автопилотируемых транспортных средств.

## **Использованные источники**

1. Обзор 10 лучших программ для 3D-моделирования [Электронный ресурс]. Режим доступа: <https://3dclub.com/blog/top-programm-dlya-3d-modelirovaniya> (дата обращения 26.03.21).
2. Сравнительный анализ свободного программного обеспечения для 3D-моделирования [Электронный ресурс]. Режим доступа: <http://www.nsktvs.ru/node/49> (дата обращения 26.03.21).
3. Доусон М. «Программируем на Python» / Доусон М. -М.: Питер, 2014. 152-159 с.
4. Бейдер Д. «Чистый Python. Тонкости программирования для профи» / Бейдер Д. -М.: Питер, 2018. 155-165 с.