Development and Practice of Intelligent Unmanned Cluster System Full Stack Development Case Based on RflySim Toolchain

Lesson 3 3D Scene Modeling and Simulation

飞思实验室

Outline

1. Experimental platform configuration
2. Introduction to key interfaces
3. Basic Experimental Case (Free Version)
4. Advanced Interface Lab (Personal Edition)
5. Advanced Case Lab (Collection)
6. Extended Case (Full Version)
7. Brief summary

1. Installation method

1.1 Components to be installed

- 3Ds Max 2020 (or other version, install it yourself)
- Unreal Engine 4.27 (both demo and full versions require installation)
- Unreal Engine 5.1. * (Advanced Full Installation)

Note: For the full version, if you need to use RflySimUE5 for scenario development, you need to install the latest UE5 engine. The use and import methods are exactly the same as those described in 4.27 in this section.

The following describes the installation method of UE4.27 engine (networking is required):

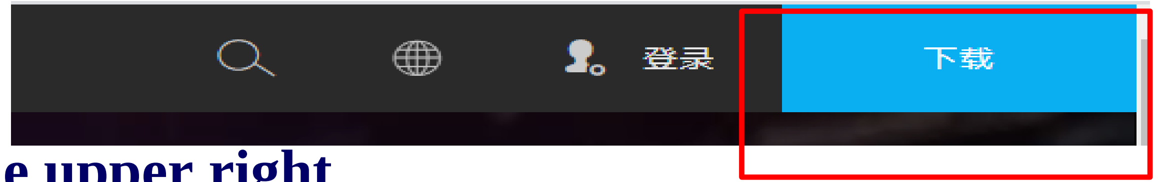
Open the EPIC UE Engine official website:

https://www.unrealengine.com/zh-CN/?lang=zh-CN

If you encounter network problems and cannot download the installation package online, you can also use the offline installation package provided by us: http://rflysim.com/res/EpicInstaller.msi

Sign up for an EPIC account and log in

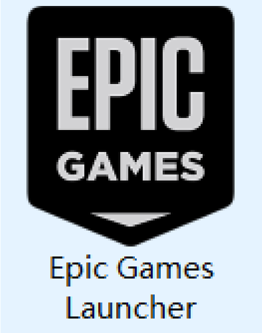
Then, click the “Download” button on the upper right



1. Installation method

1.2 How to install Epic Games

- After clicking the “Download” button, the web page will jump to the download guide.



- In the first step of the download guide, click Download Startup Program to download the latest installation package.
- Click on the installation package to install, you can get the “Epic Games Launcher” “shortcut.
- After clicking the shortcut and logging in to the Epic account, you can enter the Epic Games management page.

飞思实验室

1. Installation method

1.2 Installation method of UE 4.27

- In the Epic Games program, click on the “Unreal Engine” in the left column, and then click on the “Library” page in the right figure. The ” +” plus icon after “Engine Version” .
- Select the UE4 version to be installed in the drop-down box (Note: please install the 4.27. * version engine)
- Wait patiently for the installation to finish (note: you need to download 10 + G of resources here, and the whole installation process will take about an hour)

1. Installation method

1.3 Installation method of UE5

- In the Epic Games program, click on the “Unreal Engine” in the left column, and then click on the “Library” page in the right figure. The ”+” plus icon after “Engine Version”
- Select the UE5 version to be installed in the drop-down box (Note: please install the 5.1. * version engine and select the latest version)

飞思实验室

Outline

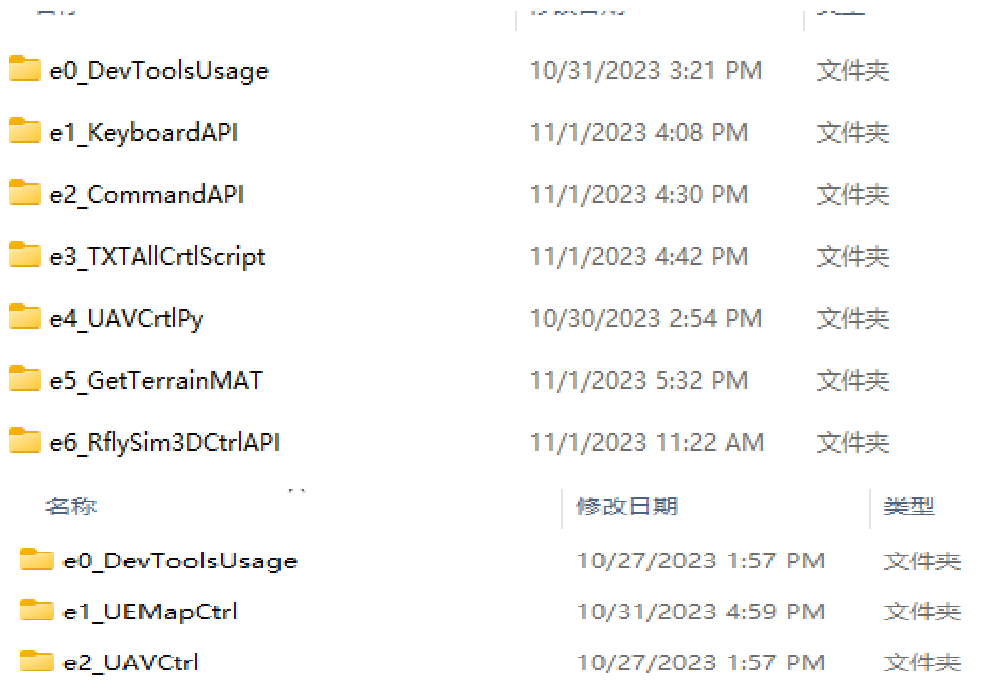
1. Experimental platform configuration
2. Introduction to key interfaces
3. Basic Experimental Case (Free Version)
4. Advanced Interface Lab (Personal Edition)
5. Advanced Case Lab (Collection)
6. Extended Case (Full Version)
7. Brief summary

2. Introduction to key interfaces

2.0 Interface Lab Overview

Comprises a basic function interface’RflySimAPIs \ 3.RflySim3DUE \ 0.ApiExps’ and an advanced function interface’RflySimAPIs \ 3.RflySim3DUE \ 2.AdvExps \ E0 ’_ AdvApiExps ”.

See the PX4PSP \ RflySimAPIs \ 3. RflySim3DUE \ API. PDF and the PX4PSP \ RflySimAPIs \ 3.RFlySim3DUE \ Readme. PDF for details.



2. Introduction to key interfaces

2.1 RflySim3D Console Command Interface Overview

This experiment provides a set of RflySim3D built-in global commands (including the built-in commands of the RflySim3D platform and the built-in commands of the UE), which can complete most of the RflySim3D-related functions, mainly including operating simulation scene objects and optimizing scene rendering performance. See PX4PSP\RflySimAPIs\3.RflySim3DUE\0.ApiExps\e2_CommandAPI\Readme.pdf for detailed operation and experimental results.

Use the command “RflySetActuatorPWMs 1000 10 00”

Enter the command: RflyDelVehicles “1000, 1001”

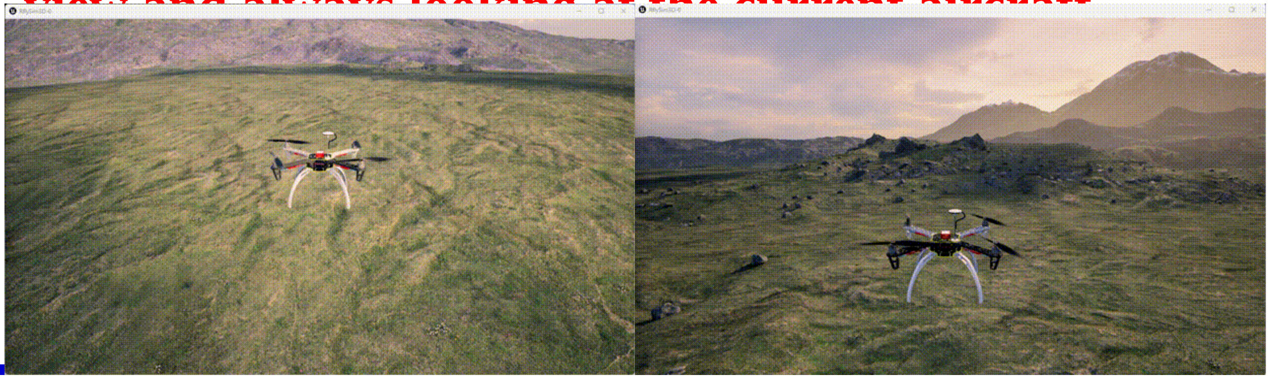
2. Introduction to key int

During ring simulation, some models in RflySim3D cannot be cleared because UDP messages are constantly received and refreshed.

2.2 Overview of RflySim3D shortcut key interface

On the basis of RflySim3D built-in global commands, this experiment also provides a set of simple interaction methods with different objects in RflySim3D scene through shortcut keys. Include popping up help menus, clearing objects in the scene, showing or hiding information, switching maps and planes, activating collision engines, showing and hiding small maps, and so on. In the ring simulation (open CopterSim), you can also switch the visual angle control, aircraft track recording and collision communication mode through the shortcut key. See PX4PSP \ RflySimAPIs\3.RflySim3DUE\0.ApiExps\e1_KeyboardAPI\Readme.pdf for detailed operation and experimental results.

The view of God 0 is different from that of God 1. The view of God 0 will move with the aircraft and will not rotate with the aircraft. View 1 Fixed ground view and always looking at the current aircraft



2. Introduction to key interfaces

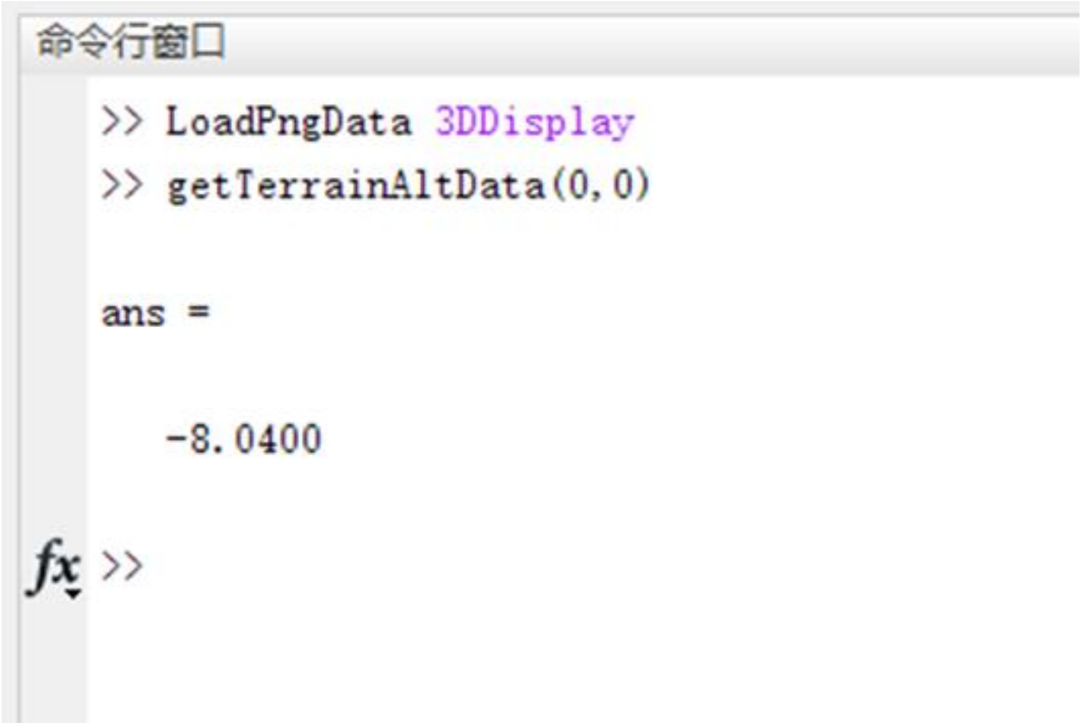
2.3 RflySim3D startup txt control script experiment

This lab provides a way to pre-write console commands in a txt file to automatically perform certain actions when RflySim3D starts, including switching maps, limiting the frame rate, and so on. Some operation effects are shown in the right figure. See PX4PSP \ RflySimAPIs \ 3.RflySim3DUE \0.ApiExps\e3_TXTAllCrtlScript\Readme.pdf-for specific experimental operations and effects.

2. Introduction to key interfaces

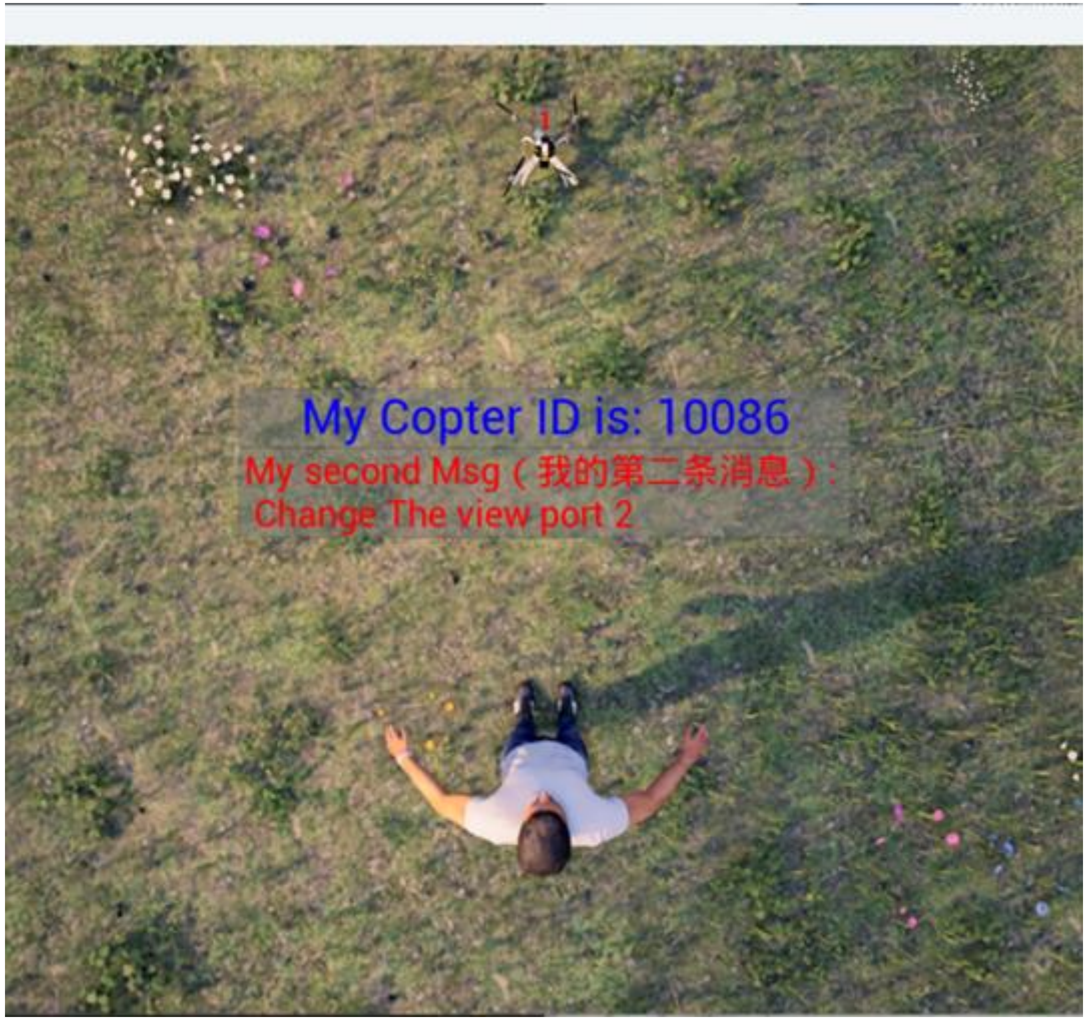
2.4 Interface experiment of obtaining terrain height matrix in MATLAB

This experiment provides the process of using MATLAB function to call the command interface of RflySim3D and to obtain and analyze the terrain data of the scene when simulating. The terrain data is mainly obtained by parsing the scene terrain configuration file (txt calibration data and PNG height map matrix). Some experimental results are shown in the right figure. For detailed operation and experimental results, see PX4PSP \ RflySimAPIs \ 3.RflySim3DUE\0.ApiExps\e5_GetTerrainMAT\Readme.pdf



2.5 Visual label display interface call (this function is only supported above the personal version)

This lab provides a way to create a target and set the tag properties of the target by calling the python interface. See PX4PSP \ RflySimAPIs \ 3.RflySim3DUE \0.ApiExps\e7_UEMapCtrl\8.TXTMapCrtlScript\Readme.pdf for specific experimental operations.



2.6 Use of global large scenes based on Cesium (this function is only supported above the personal version)

This experiment provides a method to import a large scene with high precision and a three-dimensional simulation of arbitrarily specified aircraft GPS starting point coordinates. See PX4PSP \ RflySimAPIs \ 3.RflySim3DUE \0.ApiExps\e7_UEMapCtrl\9.CesiumPlugin\Readme.pdf for specific experimental operations.

Outline

1. Experimental platform configuration
2. Introduction to key interfaces
3. Basic Experimental Case (Free Version)
4. Advanced Interface Lab (Personal Edition)
5. Advanced Case Lab (Collection)
6. Extended Case (Full Version)
7. Brief summary

3. Basic experimental case

3.1 3D scene production and import experiment

This lab provides the process of importing UE4' s own scenes into RflySim platform, getting familiar with baking scenes from UE4 and importing RflySim3D and CopterSim. In addition to the baked 3D scene file (.umap), terrain configuration files (.txt calibration data and.png height map matrix) are needed to help CopterSim recognize the map, and subsequent scene terrain services rely on these two configuration files. See the file PX4PSP \ RflySimAPIs \ 3.RflySim3DUE \1.BasicExps\e0_StarterContent\Readme.pdf-for the specific experimental operation. The import effect is as shown in the figure:

3. Basic experimental case

3.2 3D model making and importing experiment

In this experiment, the custom quadrotor model is adjusted in 3ds Max, and the wing and fuselage are imported into UE as static grids to add materials and bake, and finally imported into RflySim3D together with the supporting model configuration file XML to show the effect. This experiment takes the Droneyee X680 aircraft as an example. See the file PX4PSP \ RflySimAPIs \ 3.RflySim3DUE \1.BasicExps\e1_CusLoadDroneyeeX680\Readme.pdf-for the specific experimental operation.

3. Basic experimental case

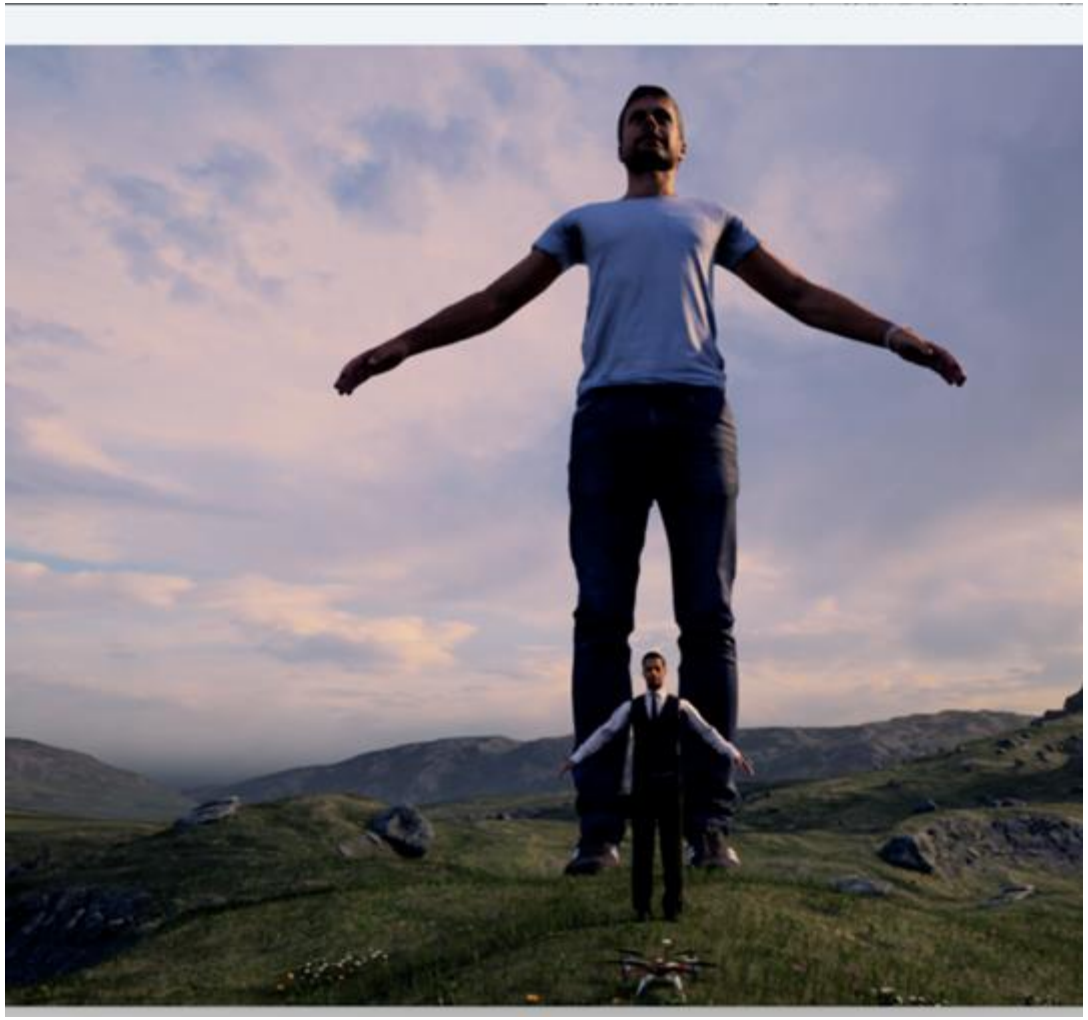
3.3 Overview of simple scene control interface

This series of experiments provides a set of methods for routine adjustment of simulation scenarios and models by calling the python interface, including loading of model scenarios, verification of communication ports, etc.

Load the model	0.ApiExps\e6_RflySim3DCtrlAPI\2.LoadModelsOnBat\Readme.pdf
Load txt file to initialize RFlySim 3D scene	0.ApiExps\e6_RflySim3DCtrlAPI\3.LoadModelsByTxt\Readme.pdf
UDP communication authentication	0.ApiExps\e6_RflySim3DCtrlAPI\4.PX4RecUE4APITest\Readme.pdf
Move object creation	0.ApiExps\e6_RflySim3DCtrlAPI\5.RflySim3DMapTerrainDemo\Readme.pdf
Angle adjustment	0.ApiExps\e6_RflySim3DCtrlAPI\6.RflySim3DViewPortDemo\Readme.pdf

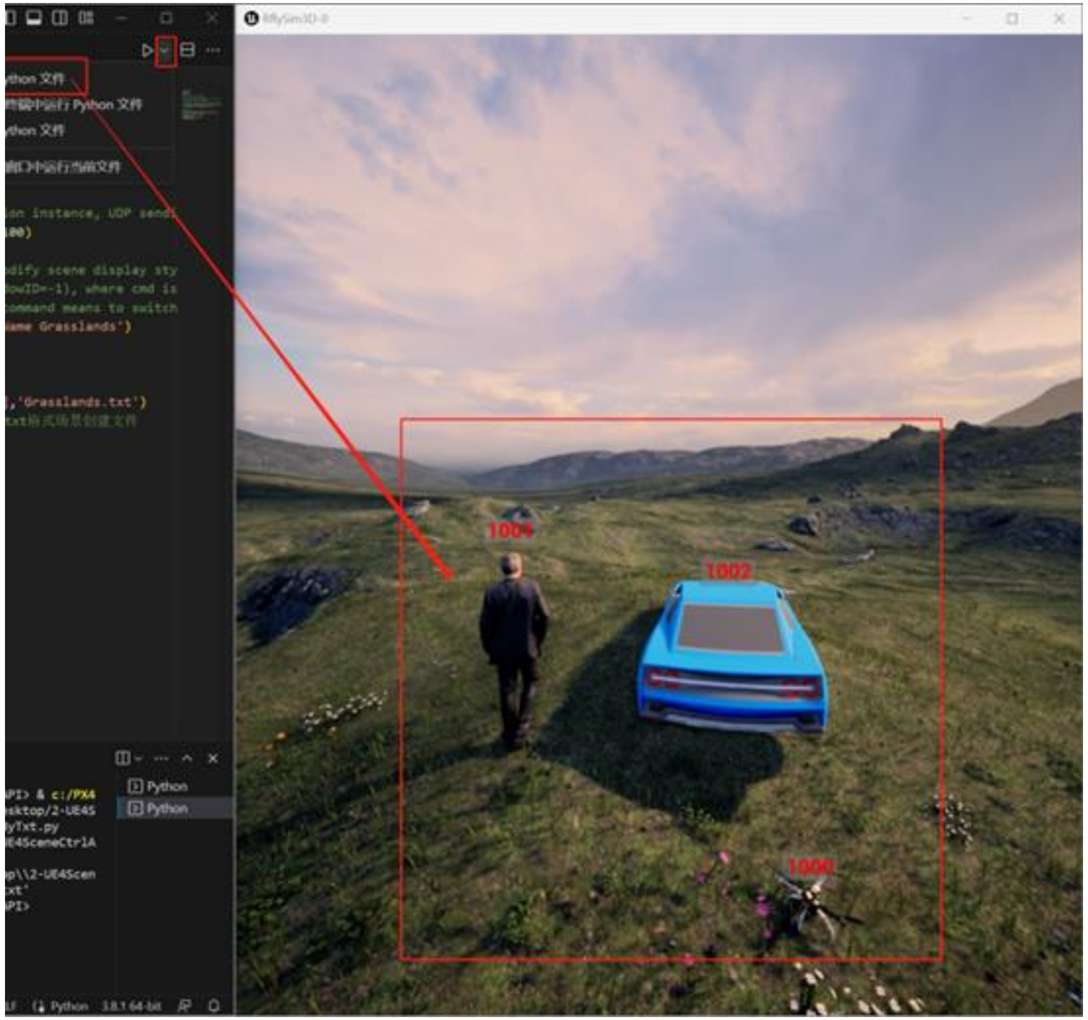
3.3 Simple scene control interface experiment (Bat script loading model)

This lab provides a quick way to lay out a RflySim3D scene using bat scripts and Python scripts. See the file PX4PSP \ RflySimAPIs \ 3.RflySim3DUE \0.ApiExps\e6_RflySim3DCtrlAPI\2.LoadModelsOnBat\Readme.pdf-for the specific experimental operation.



3.3 Simple scene control interface experiment (RflySim3D loads the txt file)

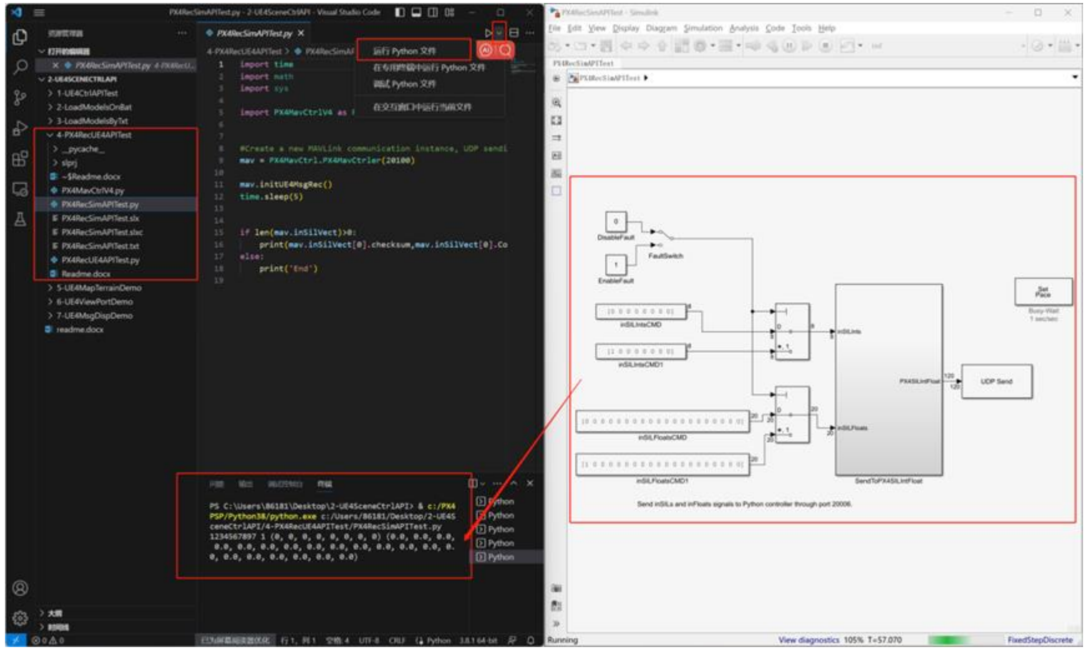
This lab provides a way to manipulate the RflySim3D scene by reading the txt configuration file. See the file PX4PSP \ RflySimAPIs \ 3.RflySim3DUE \0.ApiExps\e6_RflySim3DCtrlAPI\3.LoadModelsByTxt\Readme.pdf for the specific experimental operation.



3. Basic experimental case

3.3 Simple scene control interface experiment (external UDP communication verification)

This experiment uses Simulink to send data to python to verify the UDP communication interface of external program control RflySim3D. See the file PX4PSP \ RflySimAPIs \ 3.RflySim3DUE \0.ApiExps\e6_RflySim3DCtrlAPI\4.PX4RecUE4APITest\Readme.pdf-for the specific experimental operation.

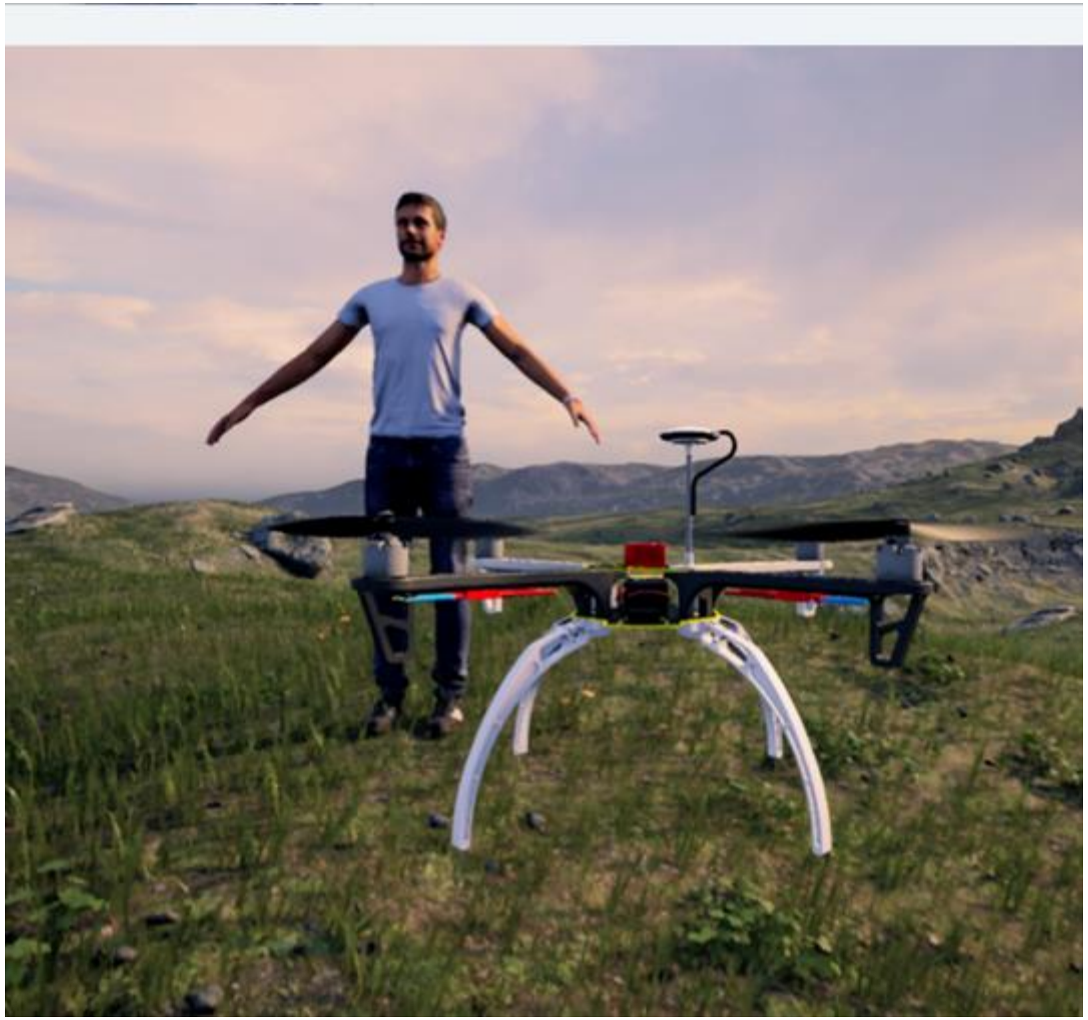


飞思实验室

3. Basic experimental case

3.3 Simple scene control interface experiment (creation of moving objects in the scene)

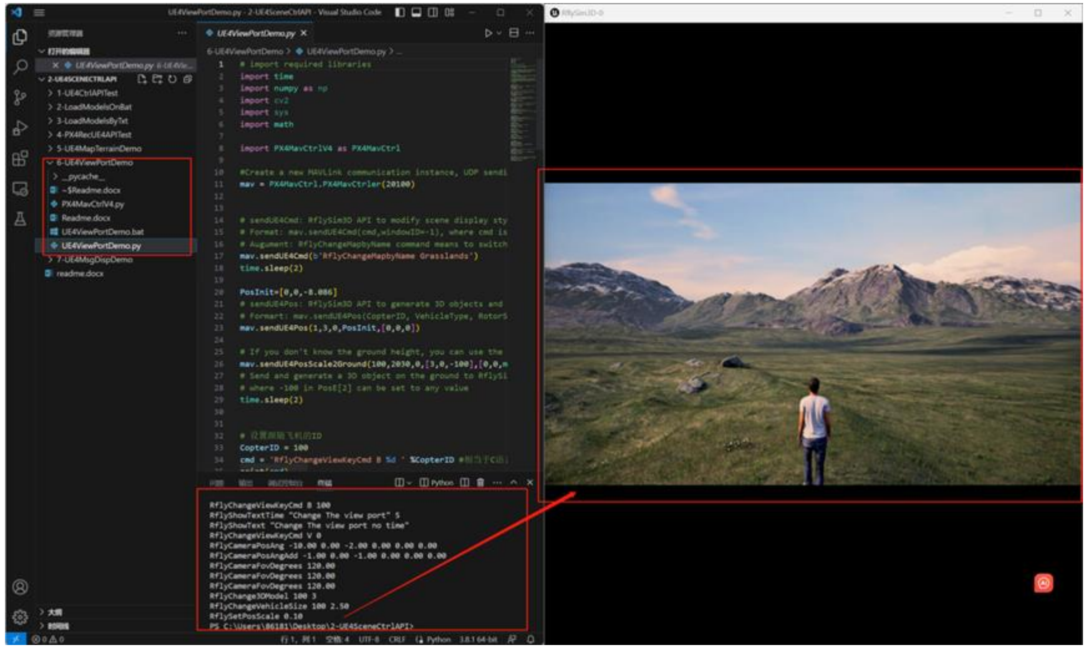
This experiment provides an example of creating a person and a quadrotor with an initial position close to the ground through the python interface, and continuously adjusting the position of the quadrotor by sending UDP in a loop. See the file PX4PSP \ RflySimAPIs \ 3.RflySim3DUE \0.ApiExps\e6_RflySim3DCtrlAPI\5.RflySim3DMapTerrainDemo\Readme.pdf-for the specific experimental operation.



飞思实验室

3.3 Simple scene control interface experiment (view angle adjustment in the scene)

This experiment provides an example of creating a person and a quadrotor with an initial position close to the ground through the python interface, and continuously adjusting the perspective in the scene by changing the UDP commands sent. See the file PX4PSP \ RflySimAPIs \ 3.RflySim3DUE \0.ApiExps\e6_RflySim3DCtrlAPI\6.RflySim3DViewPortDemo\Readme.pdf-for the specific experimental operation.



Outline

1. Experimental platform configuration
2. Introduction to key interfaces
3. Basic Experimental Case (Free Version)
4. Advanced Interface Lab (Personal Edition)
5. Advanced Case Lab (Collection)
6. Extended Case (Full Version)
7. Brief summary

4. Advanced Interface Lab

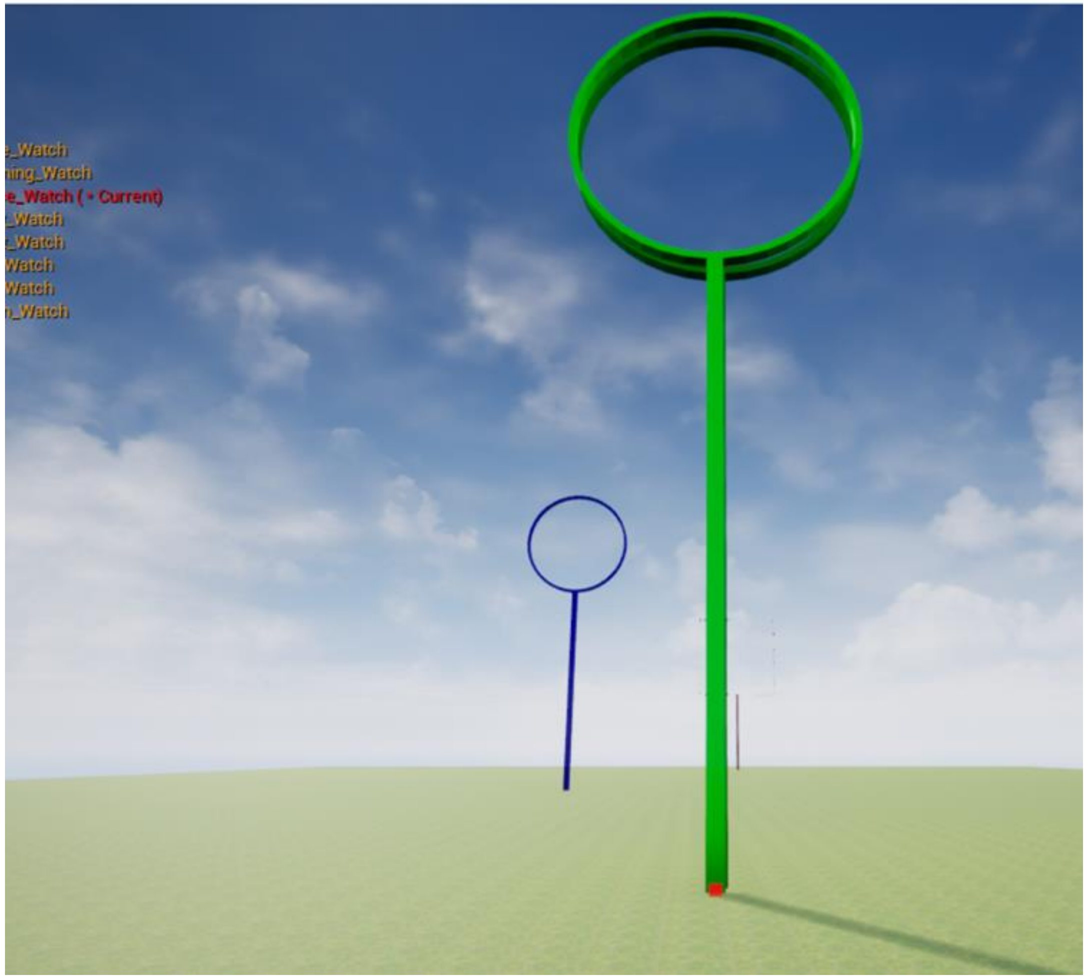
- **4.1 Overview of Advanced Scene Control Interface (Quick Scene Placement)**
- **This series of experiments provides a set of methods to quickly arrange and adjust the scene in RflySim3D (that is, without opening the UE engine). You can directly use the shortcut key or call the python interface, which is mainly used to create some obstacles and targets.**

Lay out target scenes with shortcut key	0.ApiExps\e7_UEMapCtrl\1.TargetCreateKey\Readme.pdf -
Set up target scenes through python scripts	0.ApiExps\e7_UEMapCtrl\2.TargetCreatePy\Readme.pdf -
Placement and Terrain Matching Target	0.ApiExps\e7_UEMapCtrl\3.TargetPlace\Readme.pdf -

4. Advanced Interface Lab

4.1 Advanced Scene Control Interface Experiment (Shortcut Key Layout Target)

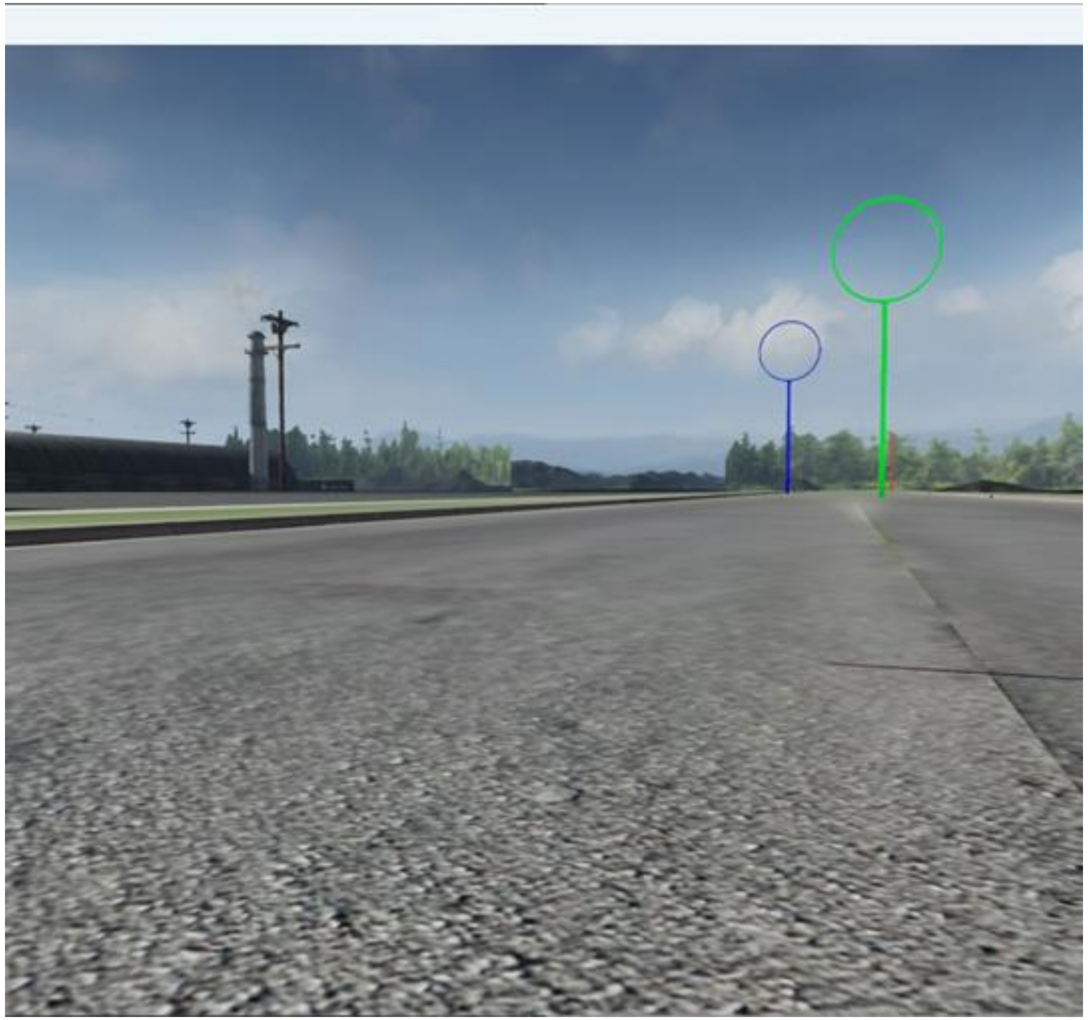
This lab provides the process of creating a target with a short cut key and modifying the XML file to match the terrain. See PX4PSP \ RflySimAPIs \ 3. RflySim3DUE \0.ApiExps\e7_UEMapCtrl\1.TargetCreateKey\Readme.pdf-for specific experimental operations.



4. Advanced Interface Lab

4.1 Advanced scene control interface experiment (python script layout target)

This lab provides the process of creating targets in batches through python scripts and matching them with the terrain (without modifying the XML file, it can quickly adapt to different scenes). See PX4PSP \ RflySimAPIs \ 3. RflySim3DUE \0. ApiExps\e7_UEMapCtrl\2.Target CreatePy\Readme.pdf-for specific experimental operations.

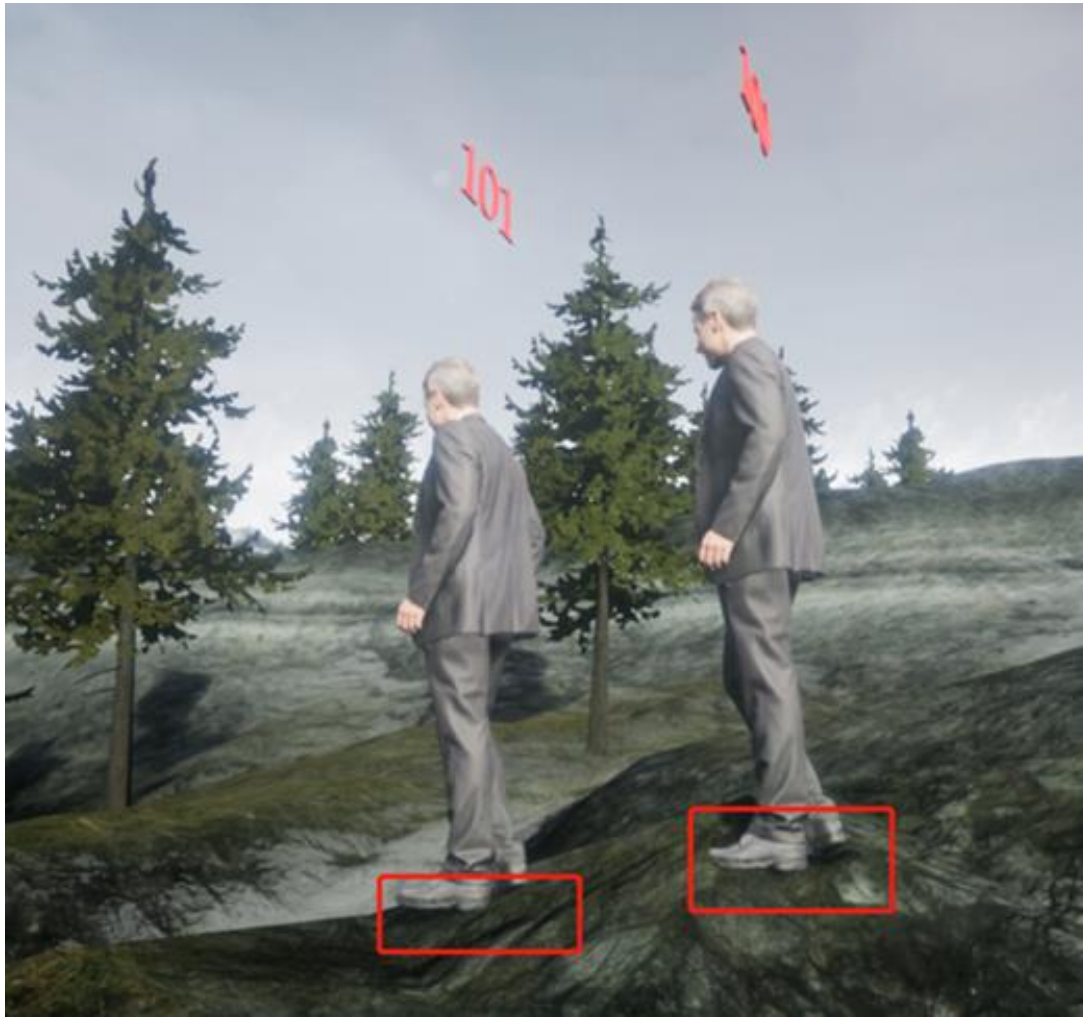


飞思实验室

4. Advanced Interface Lab

4.1 Advanced scene control interface experiment (python script to arrange pedestrians)

This experiment provides the process of creating a character through a python script and matching it with the terrain (depending on the calling interface, you can manually modify the initial position and height of the model, or automatically match the terrain). Refer to PX4PSP \ RflySimAPIs \ 3.RflySim3DUE \0.ApiExps\e7_UEMapCtrl\3.TargetPlace\Readme.pdf-for detailed experimental operation.



4. Advanced Interface Lab

- **4.1 Advanced Scene Control Interface Overview (Fast Track Generation)**
- **This series of experiments provides a set of UDP messages sent by simulink module to create objects and generate trajectories, which is mainly used to create moving obstacles in batches.**

Simulink acquires the terrain and generates the ground motion trajectory	0.ApiExps\e7_UEMapCtrl\4.TrajGen\Readme.pdf -
Experiments of generating homogeneous multi-object motion trajectories by Simulink	0.ApiExps\e7_UEMapCtrl\5.TrajGenMulti\Readme.pdf -
Experiments on Generating Motion Trajectories of Heterogeneous Multi-objects by Simulink	0.ApiExps\e7_UEMapCtrl\6.HeterTrajGenMulti\Readme.pdf -
Simulink generate circular track of motorcade	0.ApiExps\e7_UEMapCtrl\7.TenCarCircleCtrl\Readme.pdf -

4. Advanced Interface Lab

4.1 Advanced Scene Control Interface Experiment (Simulink Module Generates Pedestrian Trajectories)

In this experiment, the terrain height map matrix is obtained by MATLAB, and the pedestrian trajectory fitting the ground movement is generated by running the Simulink module. See PX4PSP \ RflySimAPIs \ 3. RflySim3DUE \0.ApiExps\e7_UEMapCtrl\4.TrajGen\Readme.pdf-for specific experimental operations.

4. Advanced Interface Lab

4.1 Advanced Scene Control Interface Experiment (Simulink Module Generates Homogeneous Multi-Model Trajectories)

In this experiment, the terrain height map matrix is obtained by MATLAB, and the trajectories of multiple quadrotors of the same style moving at a certain height from the ground are generated by running the Simulink module. See PX4PSP \ RflySimAPIs \ 3.RflySim3DUE \0.ApiExps\e7_UEMapCtrl\5.TrajGenMulti\Readme.pdf-for specific experimental operations.

4. Advanced Interface Lab

4.1 Advanced scene control interface experiment (Simulink module generates heterogeneous multi-model trajectories)

In this experiment, the terrain height map matrix is obtained by MATLAB, and the trajectories of several different models fitting the ground motion are generated by running the Simulink module. See PX4PSP \ RflySimAPIs \ 3.RflySim3DUE \0.ApiExps\e7_UEMapCtrl\6.HeterTrajGenMulti\Readme.pdf-for specific experimental operations.

4.1 Advanced scene control interface experiment (Simulink generates circular trajectory of motorcade)

In this experiment, the terrain height map matrix is obtained by MATLAB, and the circular formation of vehicles moving above the ice surface is generated by running the Simulink module. See PX4PSP \ RflySimAPIs \ 3.RflySim3DUE \0.ApiExps\e7_UEMapCtrl\7.TenCarCircleCtrl\Readme.pdf for specific experimental operations.

4. Advanced Interface Lab

TXT LOG RE COVERY RFLY SIM3D SCENE EXPERIMENT	0.APIEXPS\E7_UEMAPCTRL\8.TXTMAPCRTLSCRIPT\README.PDF -
Acquire position and collision data of all dynamically created object in RflySim3D	0.ApiExps\e9_RflySim3DPosGet\Readme.pdf -
Application method of special effect of platform	0.ApiExps\e7_UEMapCtrl\10.EffectPlugins\Readme.pdf -
Xplosive special effect trigger experiment	0.ApiExps\e6_RflySim3DCtrlAPI\12.DamageModel\Readme.pdf -

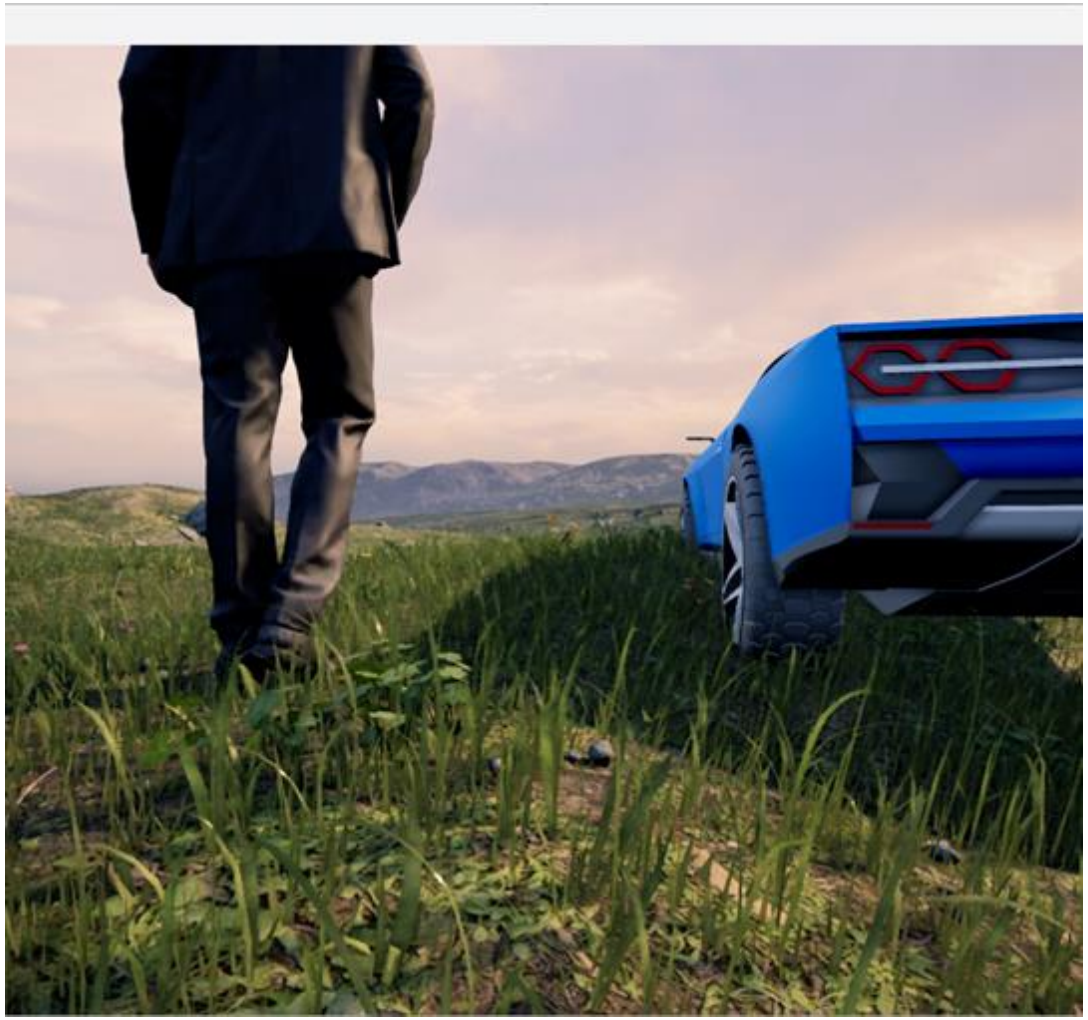
- **4.1 Overview of advanced scene control interface (calling of special effects in other scenes)**
- **This series of experiments provides some methods to obtain the data in the RflySim3D scene, as well as some triggering methods for common scene effects.**

飞思实验室

4. Advanced Interface Lab

4.1 Advanced scene control interface experiment (RflySim3D switching map control script experiment)

This lab provides a way to preserve some of the arrangements made in a particular map scene by clicking and creating an object log with RflySim3D. For example, when RflySim3D enters the 3Ddisplay map, the scene of the last layout is automatically loaded. Refer to PX4PSP \ RflySimAPIs \ 3.RflySim3DUE \0.ApiExps\e7_UEMapCtrl\8.TXTMapCrtlScript\Readme.pdff or specific experimental operations.

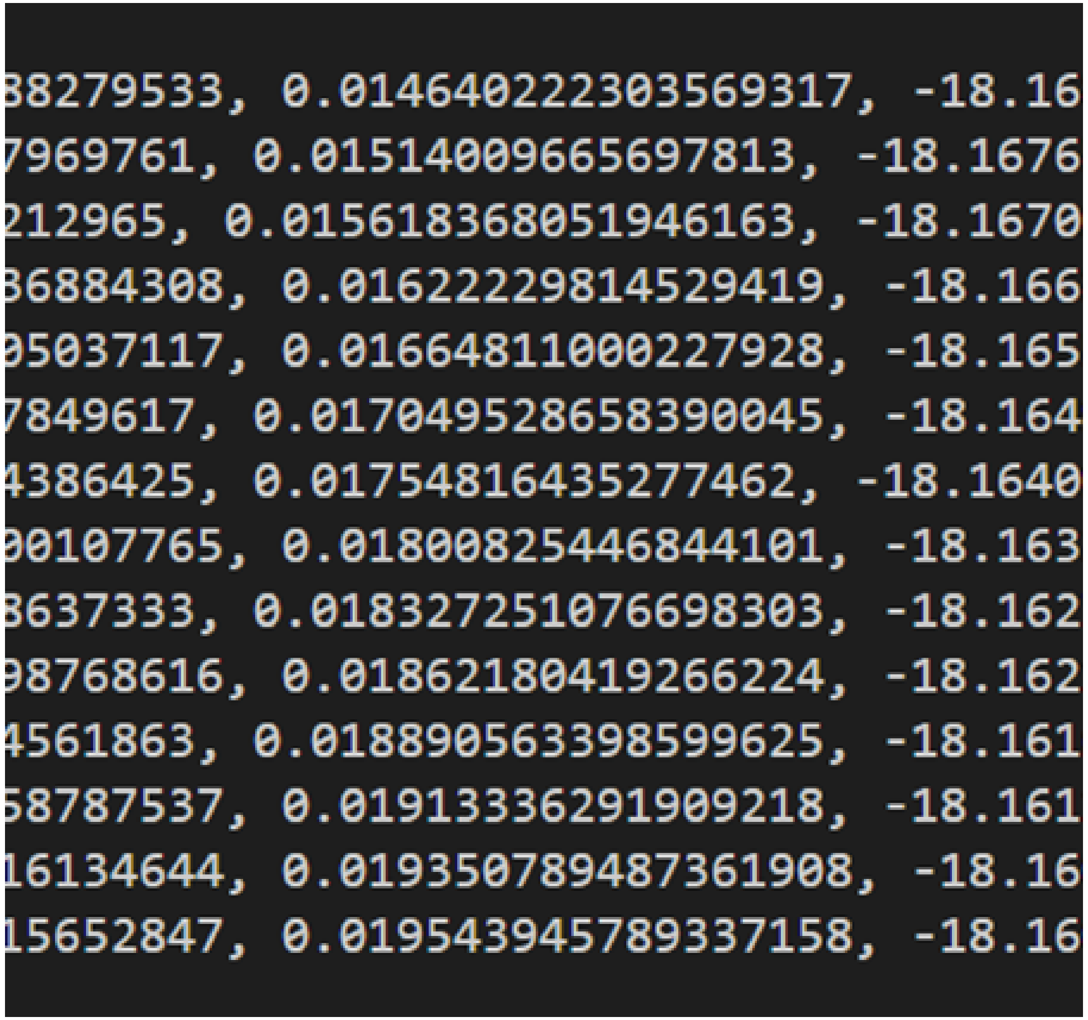


飞思实验室

4. Advanced Interface Lab

4.1 Advanced scene control interface experiment (experiment of obtaining all dynamically created object positions and collision data in RflySim3D)

This experiment provides a method to obtain the position and collision data of all dynamically created objects in RflySim3D through the python interface provided by the platform. See PX4PSP \ RflySimAPIs \ 3.RflySim3DUE \0.ApiExps\e9_RflySim3DPosGet\Readme.pdf-for specific experimental operations.



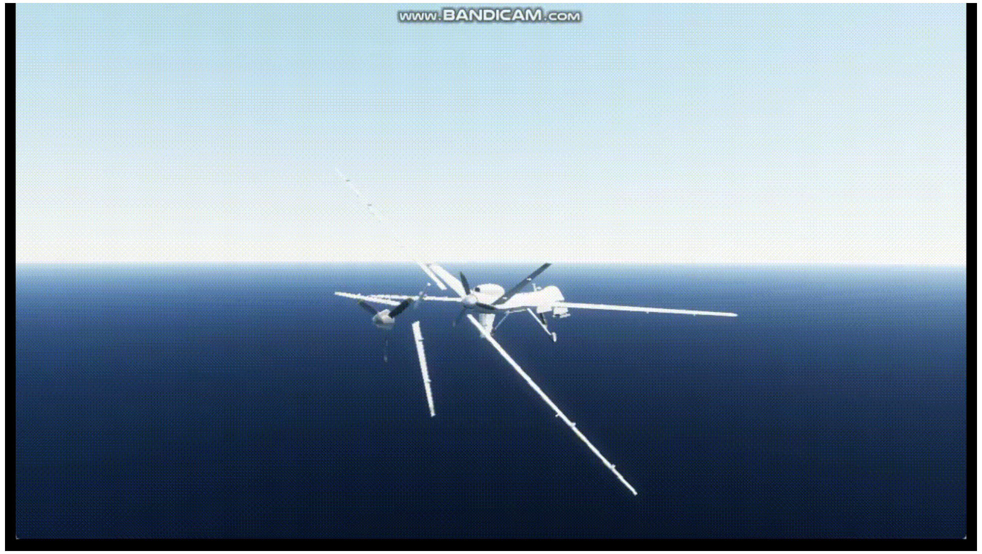
- **4.2 Overview of vehicle model adjustment interface**
- **This series of experiments provides a set of methods to adjust the three-dimensional model by modifying the XML file, mainly including the relative relationship between the actuators. As well as methods to adjust the relative relationship between models through simulink modules or python scripts.**

Actuator binding experiment	0.ApiExps\e8_UAVCtrl\1.ActuatorBinding\Readme.pdf -
Actuator control experiment	0.ApiExps\e8_UAVCtrl\2.ActuatorCtrl\Readme.pdf -
Simulink Vehicle Model Binding	0.ApiExps\e8_UAVCtrl\3.VehicleAttachSim\Readme.pdf -
Python vehicle model bindings	0.ApiExps\e8_UAVCtrl\4.VehicleAttachPy\Readme.pdf -

4. Advanced Interface La

4.2 Test of vehicle model adjustment interface (actuator binding and control interface)

These two experiments respectively provide executor components that are associated with each other by modifying the XML script binding (see PX4PSP \ RflySimAPIs \ 3. RflySim3DUE \0.ApiExps\e8_UAVCtrl\1.ActuatorBinding\Readme.pdf-) and how to modify the XML file to verify the super-8-dimensional actuator control (see PX4PSP \ RflySimAPIs \ 3.RflySim3DUE \0.ApiExps\e8_UAVCtrl\2.ActuatorCtrl\Readme.pdf for details).



4. Advanced Interface Lab

4.2 Test of vehicle model adjustment interface (model binding interface)

These two experiments are provided by the Simulink module (see PX4PSP \ RflySimAPIs \ 3.RflySim3DUE \0.ApiExps\e8_UAVCtrl\3.VehicleAttachSim\Readme.pdf-) and Python scripts (see PX4PSP \ RflySimAPIs \ 3.RflySim3DUE \0.ApiExps\e8_UAVCtrl\4.VehicleAttachPy\Readme.pdf-) send to RflySim3D that method of the structure that define the attachment relationship between the carriers.

Outline

1. Experimental platform configuration
2. Introduction to key interfaces
3. Basic Experimental Case (Free Version)
4. Advanced Interface Lab (Personal Edition)
5. Advanced Case Lab (Collection)
6. Extended Case (Full Version)
7. Brief summary

5. Advanced Case Lab

- **5.1 Getting Started with Advanced Scenario Development Tools**
- **This series of experiments provides a set of dependent software for developing 3D simulation scenes based on UE engine, including commonly used 3D processing software (processing models) 3dsMax and SketchUp, and real-time visualization preview software Twinmotion based on UE. And a global large scene plug-in Cesium for Unreal supporting the UE engine. These experiments are only a brief introduction to the basic functions of the above tools, and the specific use methods can refer to the learning path in the routine.**

UE Blueprint Simple Getting Started	0.ApiExps\e0_DevToolsUsage\1.UEBlueprintUsage\Readme.pdf
Getting Started with 3ds Max Simplicity	0.ApiExps\e0_DevToolsUsage\2.3dsMaxUsage\Readme.pdf
SketchUp installation and ease of up	0.ApiExps\e0_DevToolsUsage\5.SketchUpUsage\Readme.pdf -
Twinmotion installation and use	0.ApiExps\e0_DevToolsUsage\6.TwinmotionUsage\Readme.pdf -
Installation and use of Cesium for Unreal.	0.ApiExps\e0_DevToolsUsage\3.CesiumForUnrealUsage\Readme.pdf -

飞思实验室

5.2 UE5 scene import (built-in scene)

This lab provides a process for baking a scene from UE5 and importing RflySimUE5 \ RflySim3D and CopterSim. Here, the acquisition of the terrain configuration file is the same as that of the UE4 scene, and the step of scene baking is different from that of the UE4. See the file PX4PSP \ RflySimAPIs \ 3.RflySim3DUE \0.ApiExps\e0_DevToolsUsage\4.UE5StarterContent\Readme.pdf-for the specific experimental operation. The import effect is as shown in the figure:

5. Advanced Case Lab

5.3 Twinmotion based near-ground scene construction (demo scene)

This experiment provides a process of importing scenes from Twinmotion to UE4/UE5 through Datasmith plug-in, processing and baking scenes in UE4/UE5, and importing RflySim3D and CopterSim. The Datasmith plug-in is used here to make the import effect more accurate. For the specific version configuration, refer to the PX4PSP \ RflySimAPIs \ 3.RflySim3DUE \ API. PDF. See the file PX4PSP \ RflySimAPIs \ 3.RflySim3DUE \0.ApiExps\e0_DevToolsUsage\6.TwinmotionUsage\Readme.pdffor the specific operation of this experiment. The import effect is as shown in the figure:

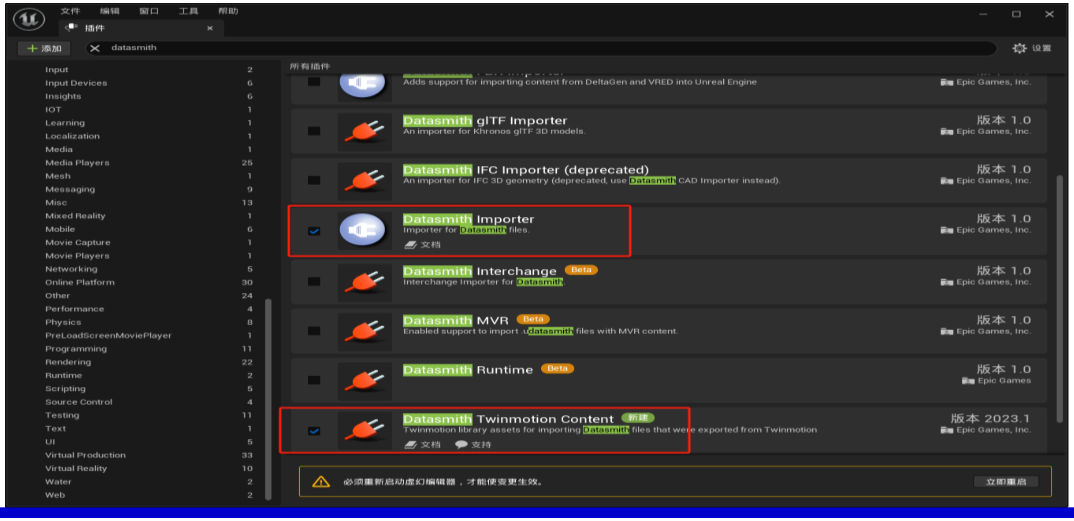
5. Advanced Case Lab

Note: The material calculation methods of the two import methods are different, and the methods of importing the baked scene into RflySim3D are also different. It is recommended to import UE5 for baking

5.3 Twinmotion based near-ground scene construction (demo scene)

If the Twinmotion scene is imported into UE4 as a.tm file, the Twinmotion importer and Twinmotion content plug-in as shown in the upper right figure need to be enabled in UE4.

If the Twinmotion scene is imported into UE5 as a.udata Smith file, the Datasmith importer and Twinmotion content plug-in as shown in the lower right figure need to be enabled in UE5.



5.4 Cesium-based global large scene construction (oblique photography conversion)

This experiment provides a set of processes for converting the OSGB model converted from aerial photography data in some areas of Hong Kong into 3D Tiles format recognized by RflySim3D in CesiumLab and importing it into RflySim3D. See the file PX4PSP \ RflySimAPIs \ 3.RflySim3DUE \2.AdvExps\e2_CesiumScene\1.ObliModelMap\Readme.pdf-for the specific experimental operation. The import effect is as shown in the figure:

5. Advanced Case Lab

5.4 Cesium-based global large scene construction (oblique photography conversion)

Oblique photography converted OSGB tiles can be downloaded directly from the Urban Planning Maps website.

The conversion tool of CesiumLab is used here to automatically identify the zero point coordinates and the number of tiles of the OSGB tilt model.

RflySim3D recognizes tiles in the 3D Tiles format because the Cesium for Unreal plug-in is already built into it.

5.5 Blueprint Model Import Process Overview

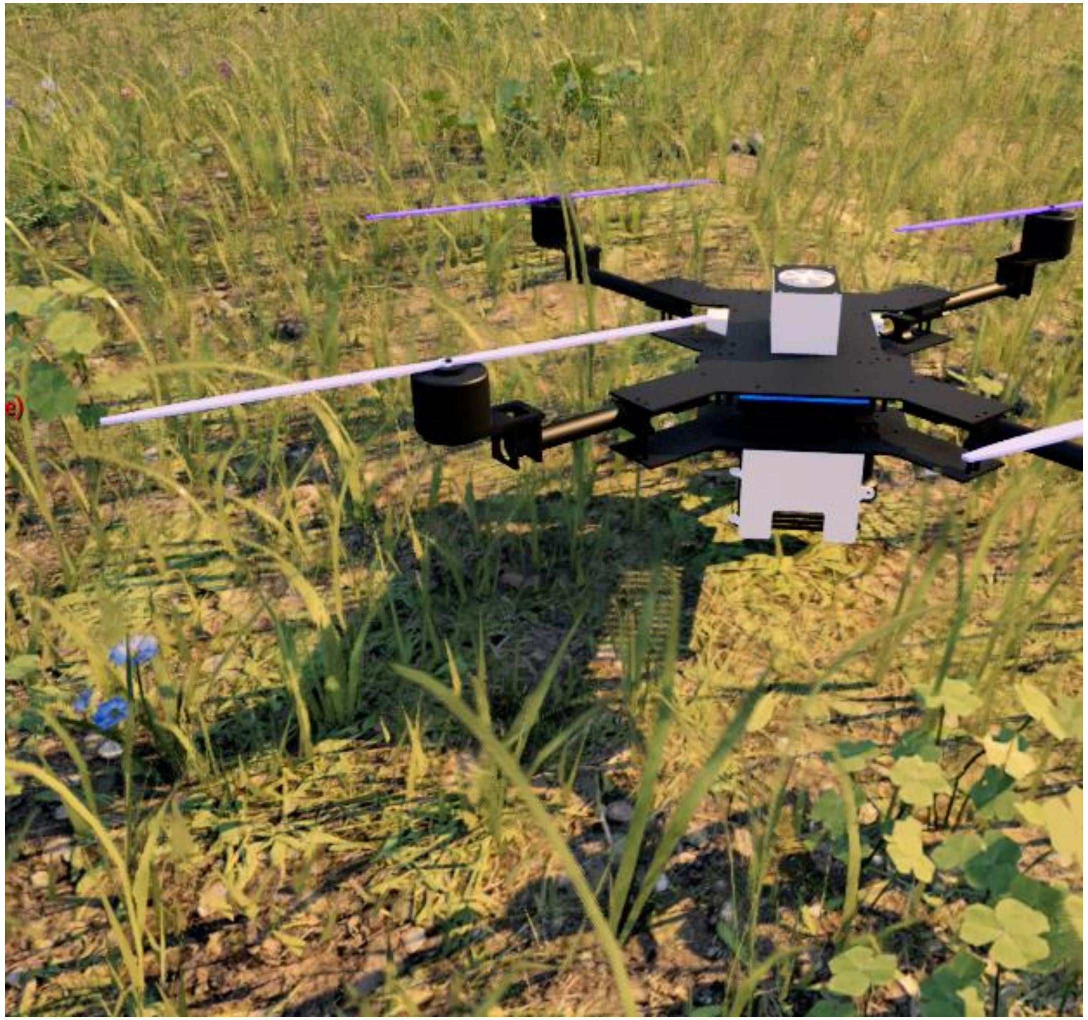
This series of experiments provides a set of processes to import the grid body and animation sequence of the model into the UE, add blueprint logic in the UE, bake and import Rfly Sim3D, and finally trigger the blueprint through the python interface to verify its animation control effect. Here, a quadrotor model with static mesh and a fixed-wing model with skeletal mesh with animation sequences and blueprint animation are taken as examples.

RflySim3D Blueprint Interface Lab	2.AdvExps\e1_BlueprintModel\1.BlueprintBuild\Readme.pdf
Unreal Mall Fixed-wing Blueprint Model Import	2.AdvExps\e1_BlueprintModel\2.BPModelLoad\Readme.pdf

5. Advanced Case Lab

5.5 Blueprint model import process experiment (quadrotor model)

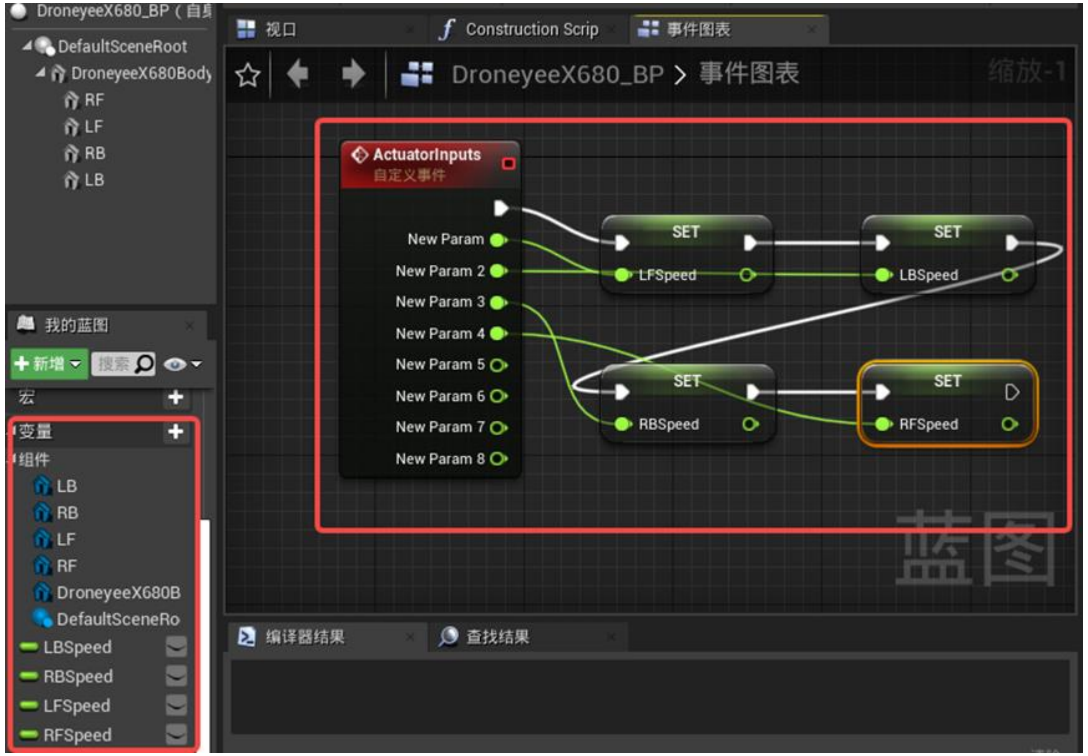
In this experiment, the fuselage and propeller of the Droneyee X680 quadrotor are imported into the UE respectively, and a blueprint actor is used to splice them into a complete quadrotor, and then a blueprint event is added to the actor. Finally, the baked model is imported into RflySim3D together with the matching XML configuration file. See PX4PSP \ RflySimAPIs \ 3. RflySim3DUE \2.AdvExps\e1_BlueprintModel\1.BlueprintBuild\Readme.pdf-



5.5 Blueprint model import process experiment (quadrotor model)

Blueprint events for Blueprint Actor here include the Blueprint interfaces provided by RflySim3D: ActuatorInputsExt, ActuatorInputs, and Event Tick, which controls the propeller rotation animation sequence.

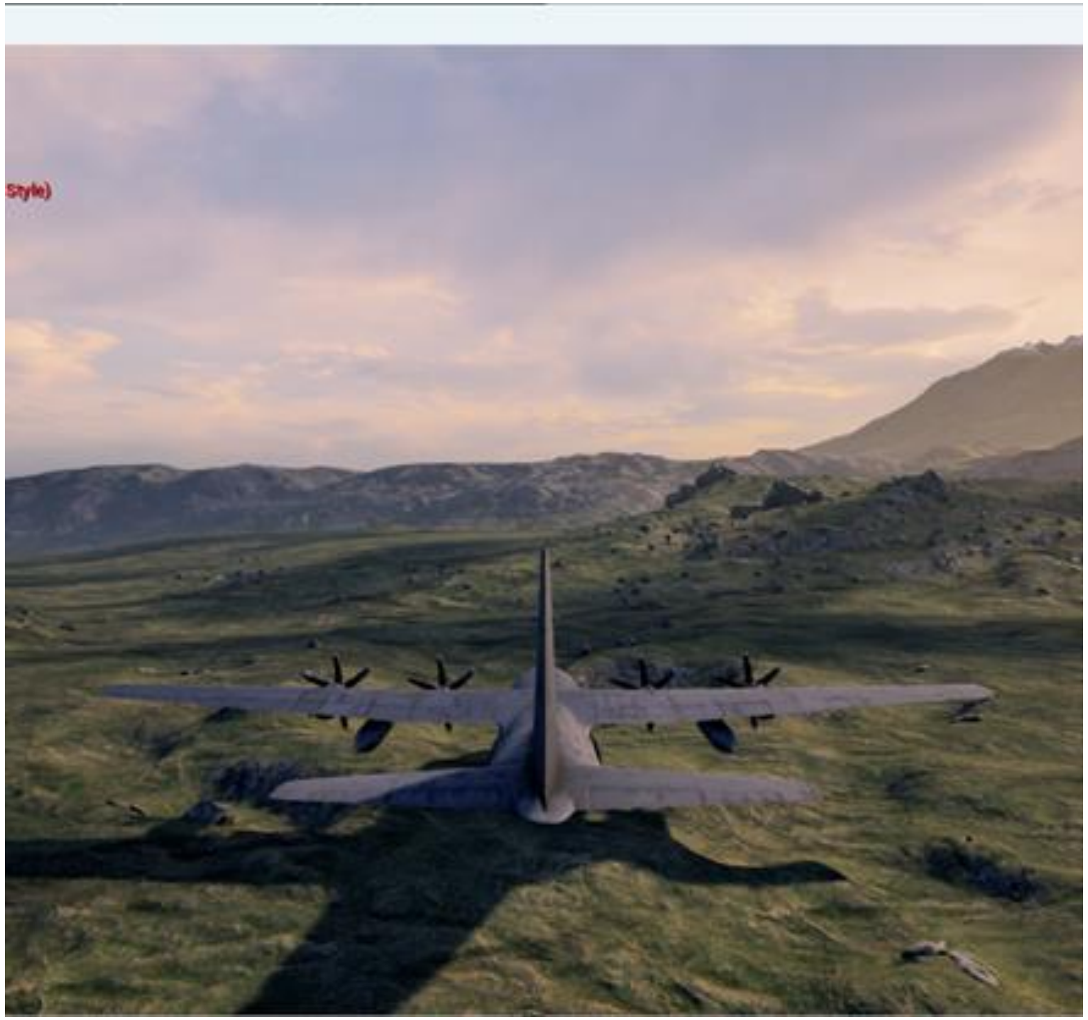
In the supporting XML file, the meshpath tag needs to point to the blueprint Actor, and the corresponding model type needs to be changed to blueprint.



5. Advanced Case Lab

5.5 Blueprint model import process experiment (fixed wing model)

In this experiment, the existing W est _ Transport _ C130J fixed-wing m odel in Unreal Mall is imported into a UE project, the blueprint interface of RflySim3D platform is added to the e vent of the existing blueprint actor, an d finally baked and imported into Rfl ySim3D together with the supporting XML configuration file. See PX4PSP \ RflySimAPIs \ 3. RflySim3DUE \2.Ad vExps\e1_BlueprintModel\2.BPModel Load\Readme.pdf-



5. Advanced Case Lab

5.5 Blueprint model import process experiment (fixed wing model)

The blueprint interface provided by RflySim3D is added to the event of the blueprint actor here: ActuatorInputs is used to pass in the 8 control surfaces of the fixed wing, and Actuator InputsExt is used to trigger special effects such as damage effects. In this experiment, the animation effect of each model component is controlled by a separate animation blueprint.

5.5 Blueprint Model Import Process Experiment (Blueprint Interface Call Method)

This lab provides a set of python interfaces to verify the import of the blueprint model. Comprises the steps of sending a console command of RflySim3D through a python script to trigger the blueprint interface and directly triggering the blueprint interface through a python command. Refer to PX4PSP \ RflySimAPIs \ 3.RflySim3DUE\0.ApiExps\e4_UE4CtrlPy\Readme.pdf for specific experimental operations.

Outline

1. Experimental platform configuration
2. Introduction to key interfaces
3. Basic Experimental Case (Free Version)
4. Advanced Interface Lab (Personal Edition)
5. Advanced Case Lab (Collection)
6. Extended Case (Full Version)
7. Brief summary

6. Expand the case

6.1 SketchUp + Twinmotion custom scene creation process

This experiment uses Sketchup to draw the villa scene model, imports Twinmotion to replace finer materials through Datasmith plug-in, renders it into a more realistic scene, adds dynamic effects and lighting in UE, and imports RflySim3D after baking. See PX4PSP \ RflySim APIs \ 3.RflySim3DUE \3.CustExps\e1_CusContentSU\Readme.pdf for detailed experiment process.

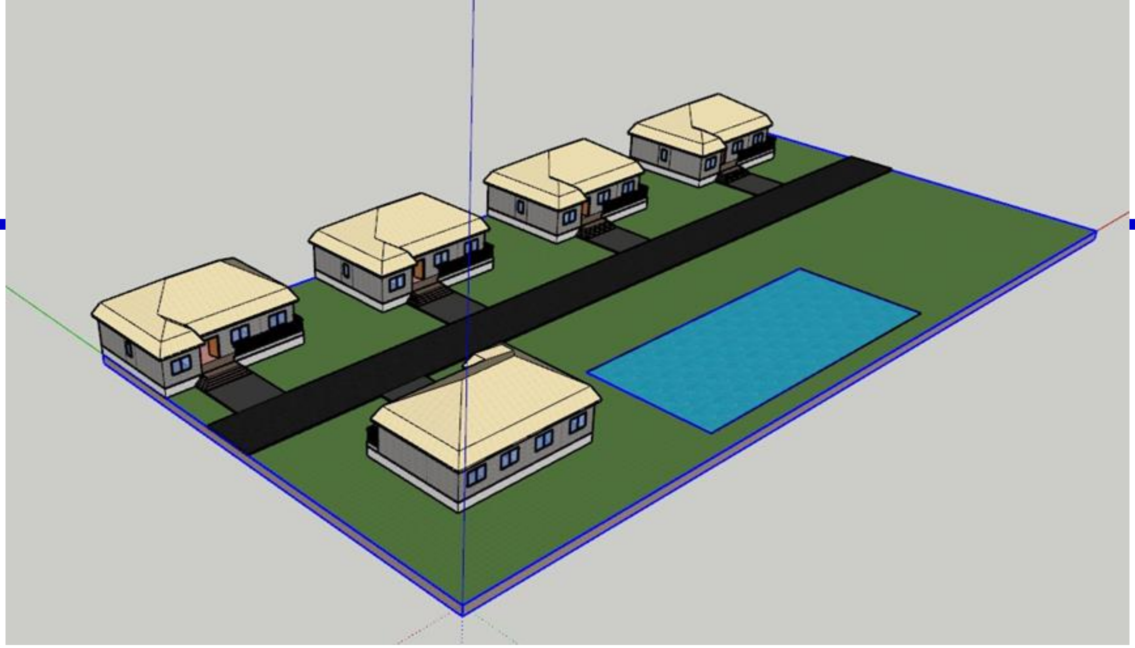
飞思实验室

6. Expand the case

6.1 SketchUp + Twinmotion custom scene creation process

Sketchup draws the villa scene model as shown in the upper right figure.

Twinmotion replaces finer mate rials and renders a more realistic s cene, as shown in the lower right i mage. Materials and lighting effect s need to be added again after the s cene is imported into the UE, beca use the material mapping relations hip has changed.



飞思实验室

6. Expand the case

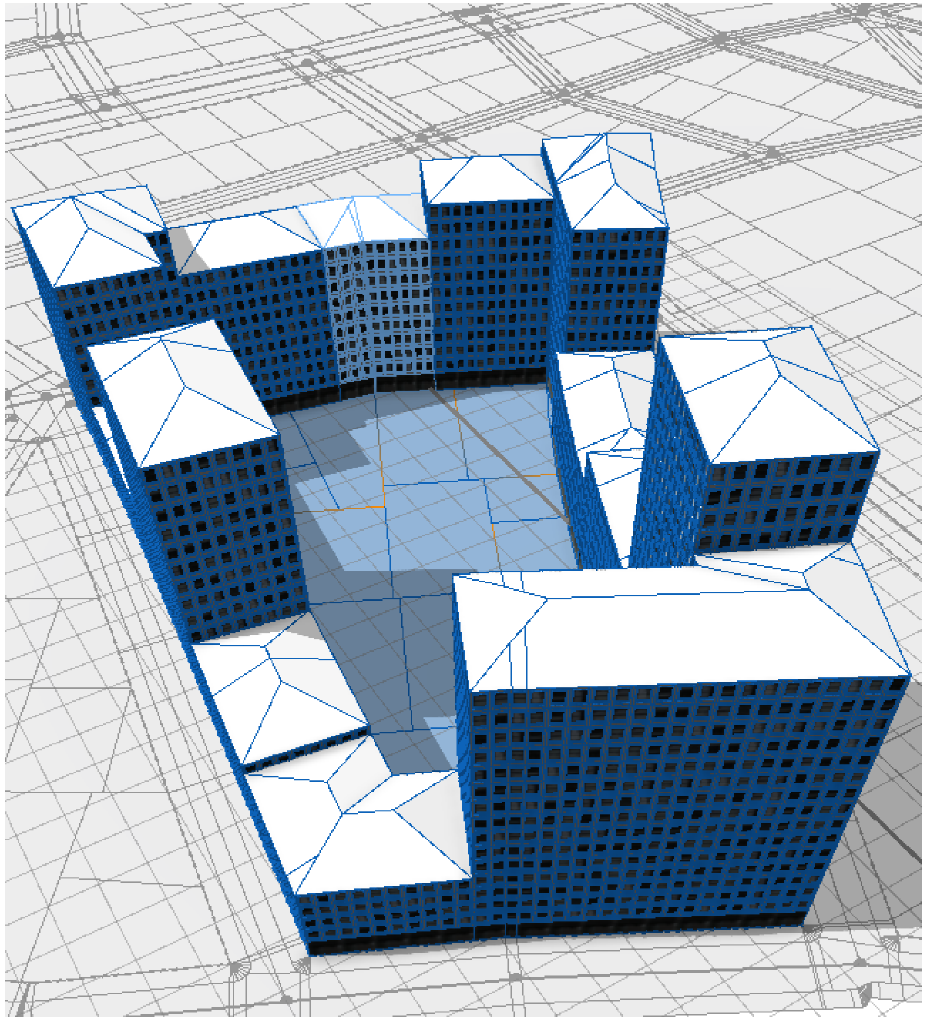
6.2 RflySim3D custom fixed-wing model loading process

In this experiment, the customized fixed-wing model is adjusted in 3ds Max, and the eight control surfaces and the fuselage are imported into UE as static grids to add materials and bake, and finally imported into RflySim3D together with the supporting model configuration file XML to show the effect. The MQ-9 Reaper aircraft was used as an example for this experiment. See the file PX4PSP \ RflySimAPIs \ 3.RflySim3DUE \3.CustExps\e2_CusLoadFixWing\Readme.pdf-for the specific experimental operation.

6. Expand the case

6.3 GIS Services (Getting Started with City Engine)

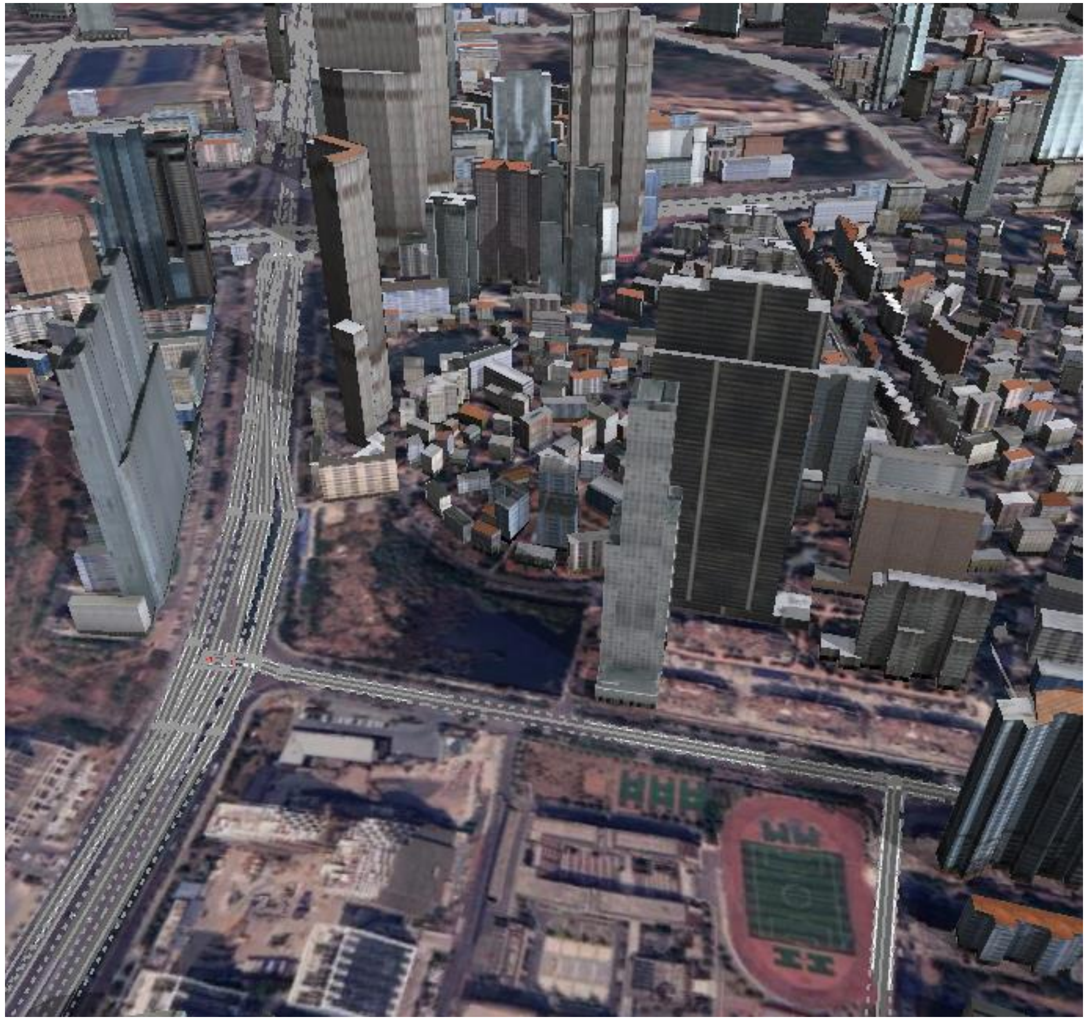
In this experiment, CityEngine was used to build a 3D building model in a specified block using cga rules. See the file PX4PSP \ RflySimAPIs \ 3.RflySim3DUE \3.CustExps\e3_CityEngineExp\1.CityEngineUsage\Readme.pdf for the specific experimental operation.



6. Expand the case

6.3 GIS service (city scene creation experiment based on CityEngine)

This experiment provides a set of methods to construct the corresponding 3D model according to the geographic information (image and elevation data) and road building information (vector data) using cga rules. See the file PX4PSP \ RflySimAPIs \ 3.RflySim3DUE \3.CustExps\e3_CityEngineExp\2.CitySceneBuild\Readme.pdf-for the specific experimental operation.



飞思实验室

6. Expand the case

6.3 GIS service (multi-level image overlapping white edge processing flow)

This experiment provides a set of processes to process multi-level image white edges by using these two commonly used geographic information system (GIS) software, Global Mapper and ArcGIS, respectively. See the file PX4PSP \ RflySimAPIs \ 3.RflySim3DUE \3.CustExps\e4_MultOverlapareasProcess\Readme.pdf-for the specific experimental operation.

Outline

1. Experimental platform configuration
2. Introduction to key interfaces
3. Basic Experimental Case (Free Version)
4. Advanced Interface Lab (Personal Edition)
5. Advanced Case Lab (Collection)
6. Extended Case (Full Version)
7. Brief summary

7. Brief summary

- This lecture mainly explains the development course of unmanned system 3D simulation scene, which is divided into three parts: basic experiment, advanced experiment and extended case, so that students can be familiar with the production and import process of various scene models and the scene control interface provided by RflySim3D as soon as possible.
- The basic experiment is based on the learning of simple scene and model import process and basic scene control interface, and the advanced experiment is based on the learning route of near-ground scene and global scene construction → blueprint model → complete scene control interface.

If you have any questions, please go to the https://doc.rflysim.com/ for more information.

RflySim More Tutorials

Scanning code consultation and communication

Freescale RflySim Technology Exchange Group

Thank you!