

# 三维场景交互接口场景物体匹配地形实验（python接口 sendUE4Pos）

## 1. 实验目的

了解如何通过Python在场景中布置障碍物，并使障碍物和地面贴合。

## 2. 实验要求

- 软件要求：Windows 10及以上版本；RflySim工具链<sup>[1]</sup>。
- 硬件要求：笔记本/台式电脑1台<sup>[2]</sup>。

## 3. 实验地址

例程目录：

[\[安装目录\]\RflySimAPIs\3.RflySim3DUE\0.ApiExps\e5\\_UEMapCtrl\2.TargetCreatePy](#)

- [XML](#)：标靶模型对应的xml文件
- [TargetCreateDemo.py](#)：此文件调用了"UE4CtrlAPI.py"中的接口
- [Python38Run.bat](#)：Python环境启动脚本
- [Ue4.bat](#)：打开RflySim3D
- [TargetCreateDemo2.py](#)：此文件调用了"UE4CtrlAPI.py"中的接口
- [UE4CtrlAPITest.py](#)：此文件调用了"UE4CtrlAPI.py"中的接口

## 4. 实验内容或步骤

### 4.1 步骤1 Python创建物体贴合地面实验（必做）

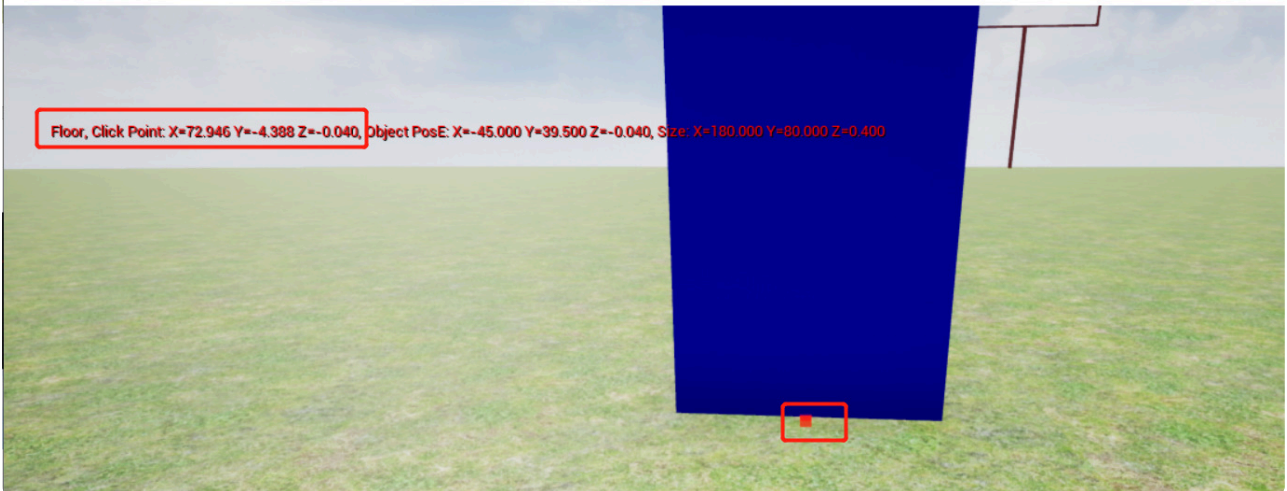
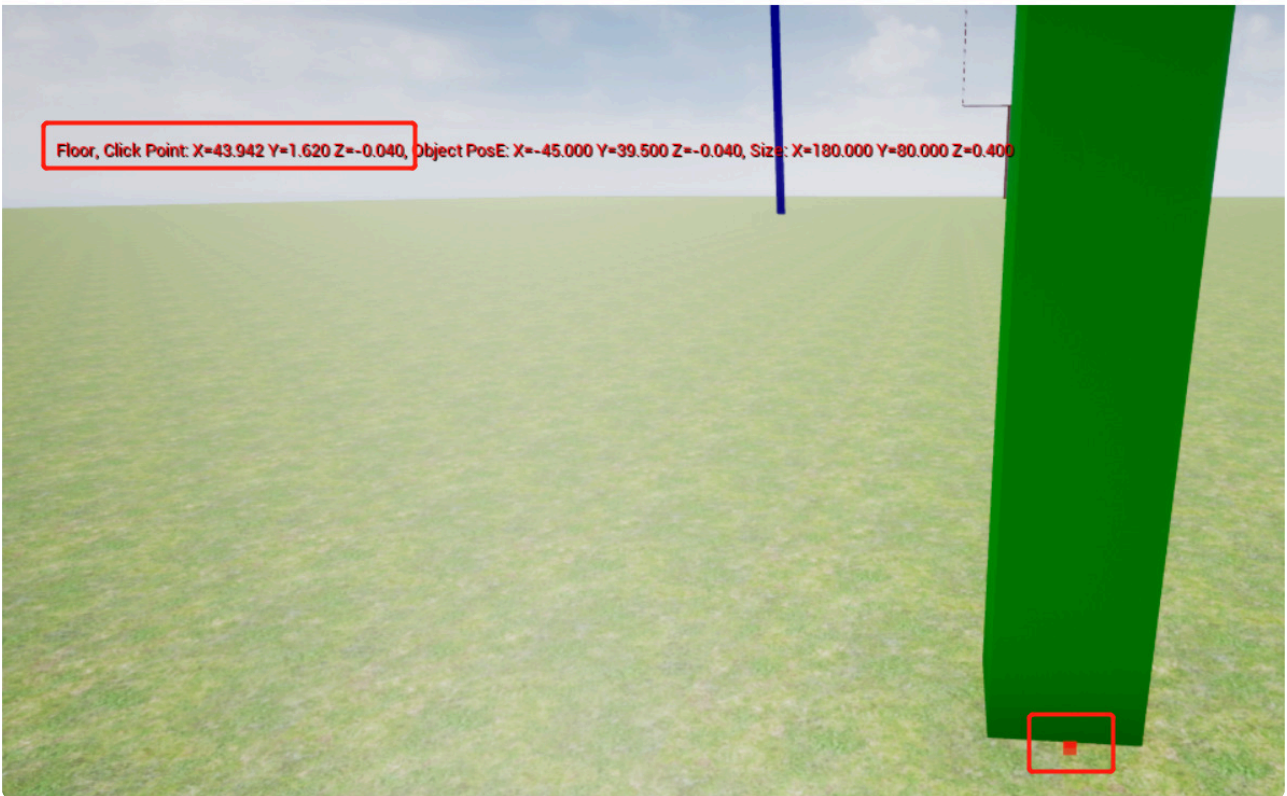
#### 步骤1: 启动RflySim3D切换到对应地图

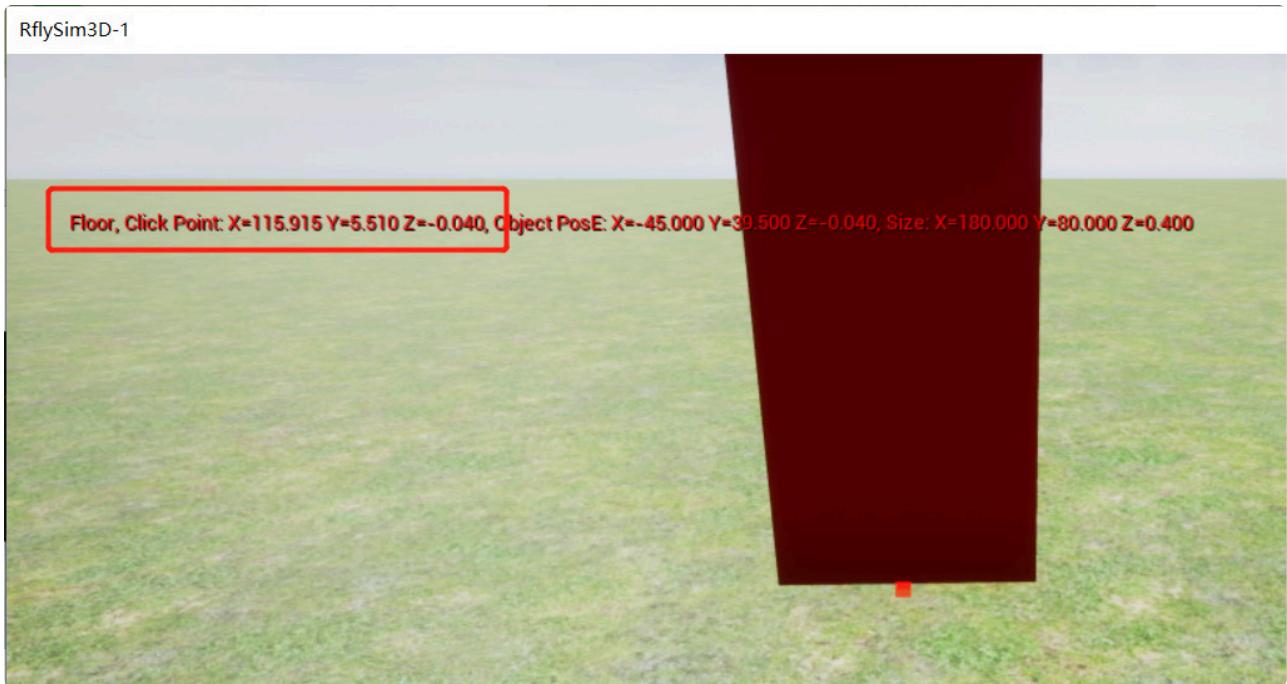
首先，打开RflySim3D程序，并反复按下"M"键直至切换到VisionRing地图。



#### 步骤2: 获取放置标靶的坐标位置

依次将视角拖动到三个靶标的根部，双击根部的地面位置，得到期望放置靶标的坐标位置。如下图可知，绿、蓝、红三个靶标的地面位置分别为 $[43.943, 1.620, -0.04]$ ， $[72.946, -4.388, -0.04]$ ， $[115.915, 5.510, -0.04]$ 。





### 步骤3: 用python脚本创建场景

以UE4CtrlAPITest.py为模版，创建一个三维物体创建的程序。分别放置绿、蓝、红三个靶标到场景中。

将地图切换到VisionRing，切换地图的代码可表示为

```
ue.sendUE4Cmd(b'RflyChangeMapbyName VisionRing',0)
```

参考对应的xml文件，在ue.sendUE4Pos的命令中直接发送vehicleType来直达需要的样式，格式是：样式序号\*100000+classid（其中，样式序号=相同classid下模型的排列顺序-1，排列顺序由displayorder标签决定）。绿色圆形靶标的ID为200150（参考Ring\_Target\_Green.xml），同样的道理蓝色为100150（参考Ring\_Target\_Blue.xml），红色方框靶标的ID为151或0151（需要看XML文件Quad\_Target\_Red.xml）。因此，发送靶标的代码可以表示为

```
ue.sendUE4Pos(1,200150,0,[43.943,1.620,-0.04],[0,0,0])
```

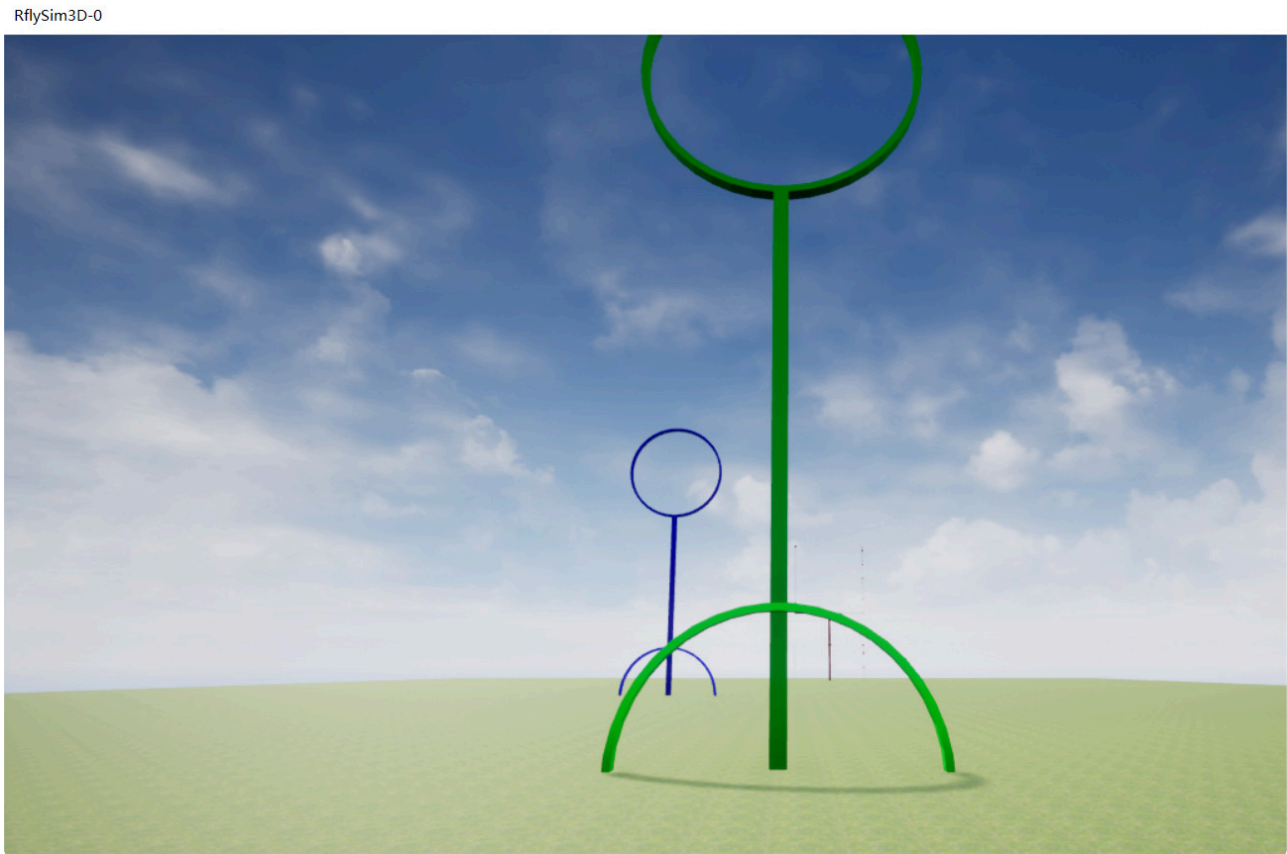
```
ue.sendUE4Pos(2,100150,0,[72.946,-4.388,-0.04],[0,0,0])
```

```
ue.sendUE4Pos(3,151,0,[115.915,5.510,-0.04],[0,0,0])
```

在文件夹下，双击Python38Run.bat，打开集成好的python环境，在该环境下运行UE4CtrlAPITest.py文件，输入  
python UE4CtrlAPITest.py

```
C:\Windows\system32\cmd.e x + v
Python3.8 environment has been set with openCV+pymavlink+numpy+pyulog etc.
You can use pip or pip3 command to install other libraries
Put Python38Run.bat into your code folder
Use the command: 'python XXX.py' to run the script with Python
C:\Users\uavcs\Desktop\demo\8.RflySimVision\0.ApiExps\1-UsageAPI\0.VisionSensorAPI\1.CameraImageGet>python VisionCapAPIDemo.py
```

这里在visionring地图原标靶对应位置生成了新的标靶，但是由于没有考虑圆心到地面的高度，我们生成的圆靶标都不是底部贴在地面上。



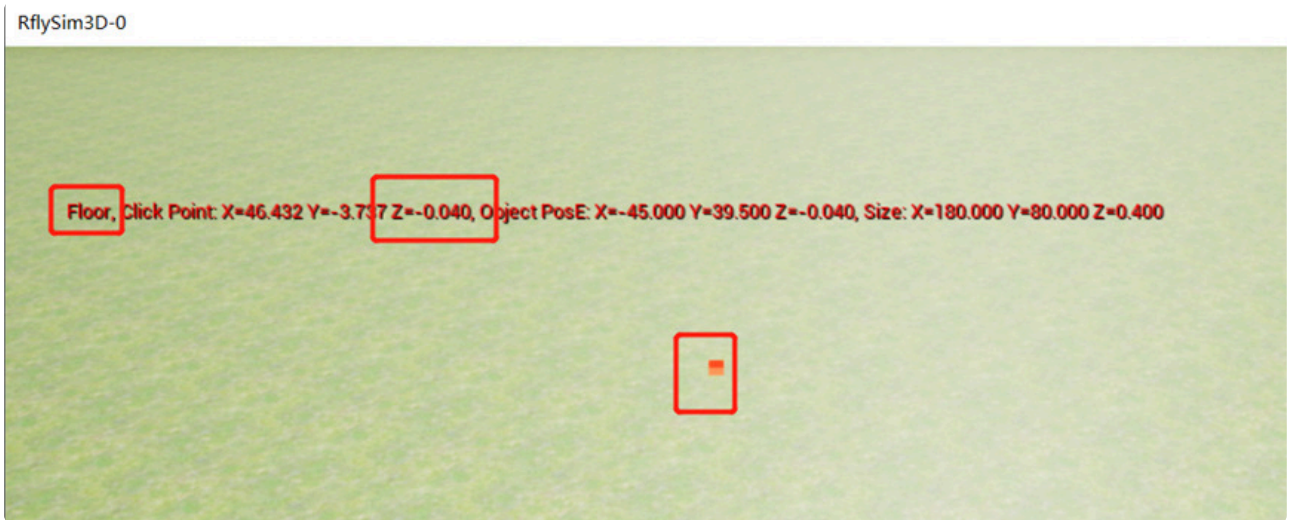
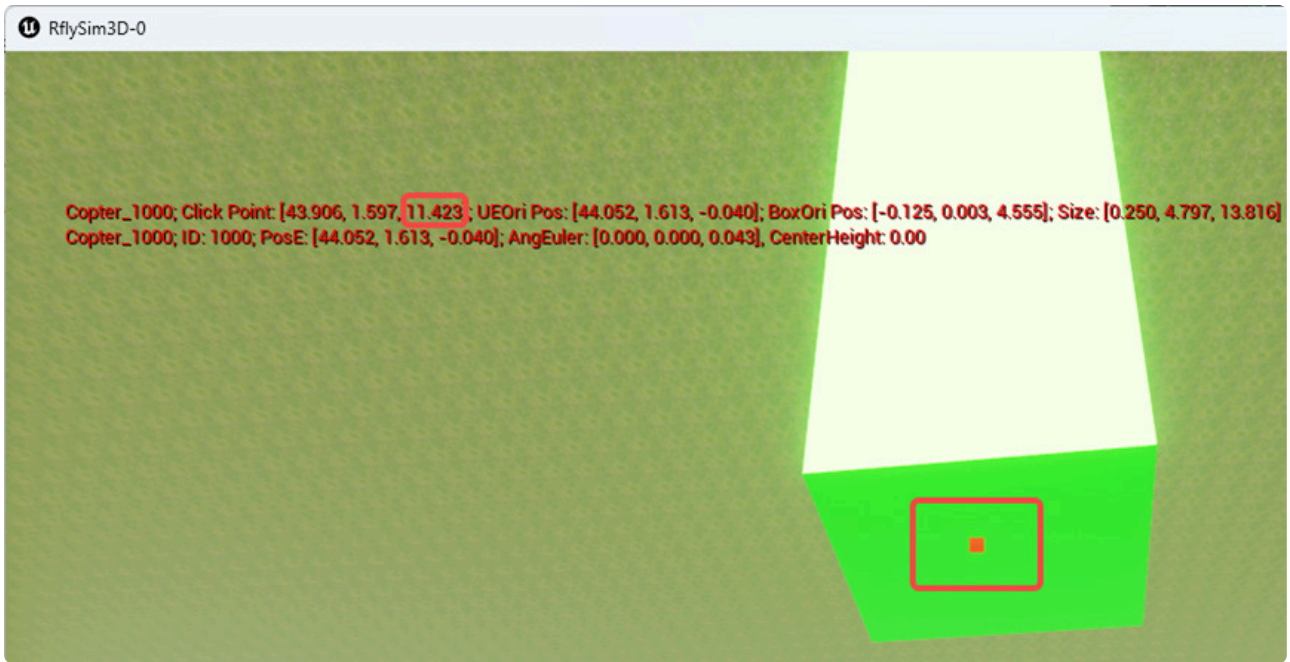
## 步骤4: 修改python脚本

鼠标双击地面和每个标靶的底部，得到每个物体中心到底部的高度差，可以得到绿色和蓝色圈需要上移11.463m（这里不需要考虑XML里面的缩放），红色方框靶标需要10.448m。因此，重新调整后的物体创建语句为

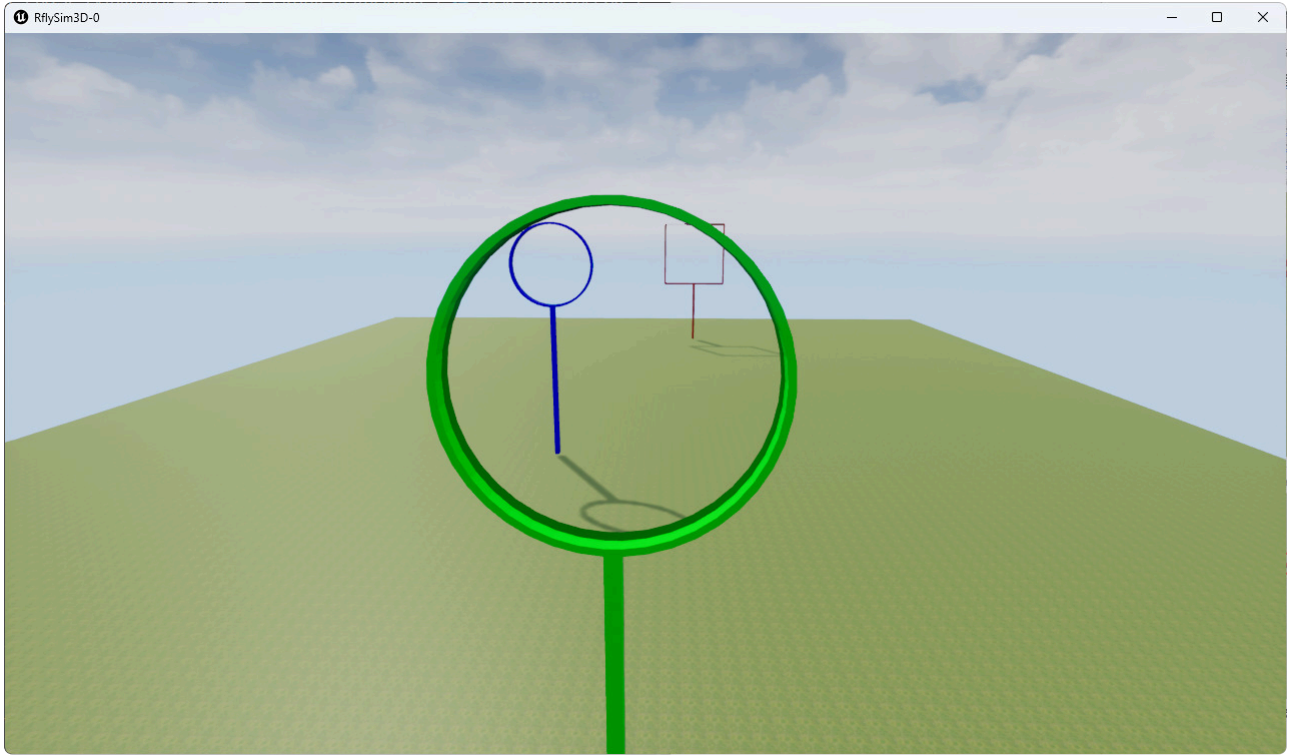
```
ue.sendUE4Pos(1,200150,0,[43.943,1.620, -11.465],[0,0,0])
```

```
ue.sendUE4Pos(2,100150,0,[72.946,-4.388,-11.465],[0,0,0])
```

```
ue.sendUE4Pos(3,151,0,[115.915,5.510,- 10.448],[0,0,0])
```



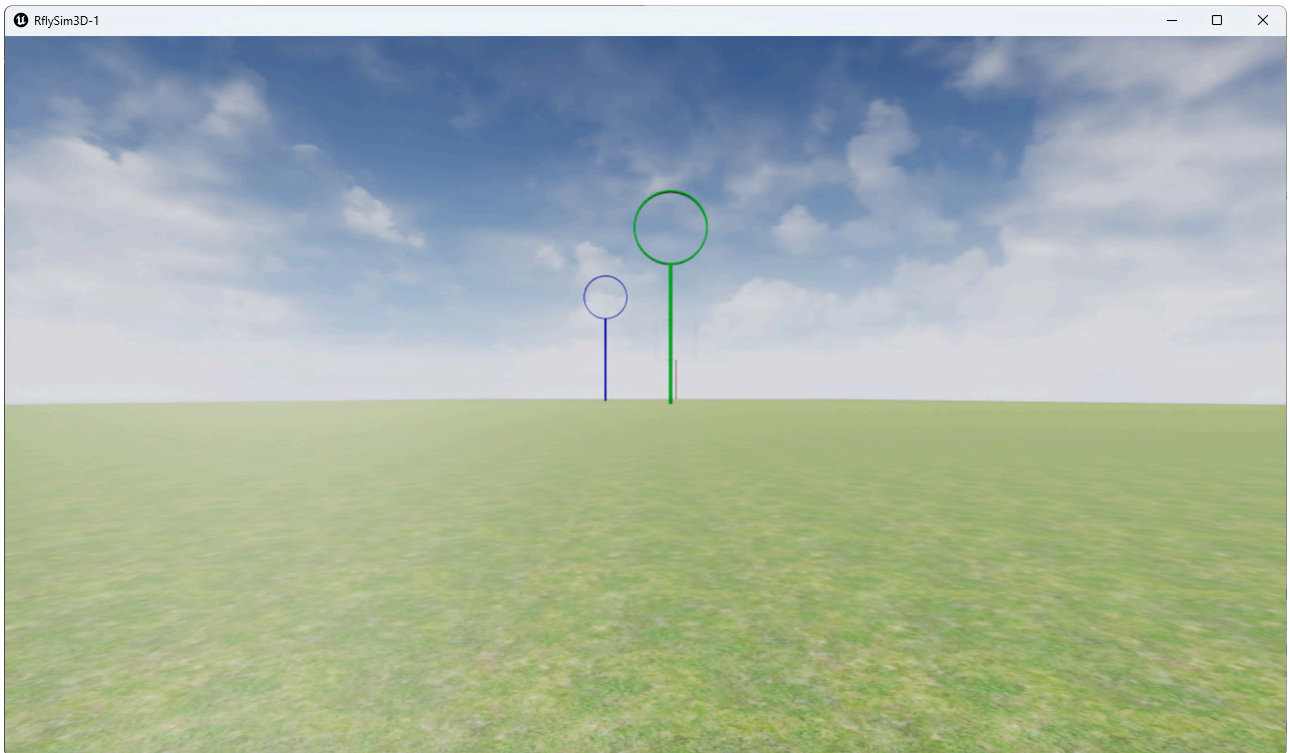
按下ESC清除创建的物体，并运行修改后的python脚本。可见场景原有标靶旁生成了相同的新标靶。



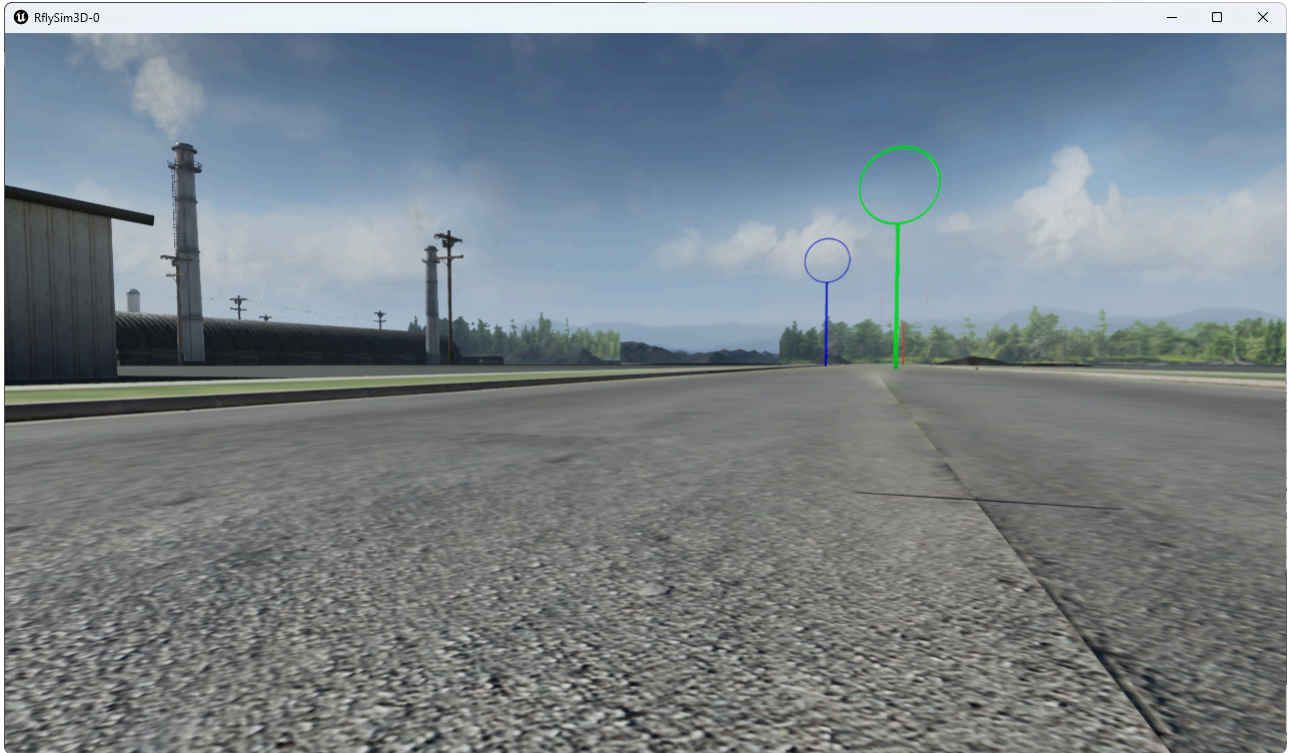
## 步骤5: 在不同场景生成标靶

按前述方法修改python脚本，可在不同场景中生成这三个靶标。

运行TargetCreateDemo.py可在空白场景中复现VisionRing中的三个标靶。



运行TargetCreateDemo2.py可在OldFactory中创建三个标靶。



## 4.2 步骤2 Vscode调试运行实验（选做）

### 准备工作

- 先确保已经按 [RflySimAPIs\1.RflySimIntro\2.AdvExps\e3.PythonConfig\Readme.pdf](#) 步骤，正确配置VS Code环境。或者配置了自己的Pycharm等自定义Python环境。
- 其他步骤与上文相同，在运行python文件时，可使用VS Code（或Pycharm等工具）来打开python文件，并阅读代码，修改代码，调试执行等。

### 扩展实验

- 请自行使用VS Code阅读[TargetCreateDemo.py](#)，[TargetCreateDemo2.py](#)，[UE4CtrlAPITest.py](#)源码，通过程序跳转，了解每条代码的执行原理；再通过调试工具，验证每条指令的执行效果。

## 5. 关键知识点

### 关键知识点1

使用sendUE4Pos接口创建物体，需要详细地设定物体的xyz坐标，因此如果想要物体完全贴在（站在）地面上，就需要去测量地表的高度（这里通过在RflySim3D中双击地面想要放置靶标的位置，就能在Click Point中读出z的坐标）。

根据在地图场景中测量得到的地形高度，创建人物的初始位置

```
UE4CtrlAPI.sendUE4Pos(copterId, vehicleType, MotorRPMMean, PosE, AngEuler, windowID)
```

示例：`ue.sendUE4Pos(100,30,0,[ -16.479,1.45,-177.181],[0,0,math.pi])`

## 6. 参考资料

1. XML文件规则 [../../API.pdf](#)
2. RflySim3D快捷键接口总览 [../../API.pdf](#)
3. RflySim3D控制台命令接口总览 [../../API.pdf](#)

## 7. 常见问题

### Q1：在场景中创建的物体无法与地面贴合怎么办？

A1：这是因为在设置物体的Z轴坐标时，没有考虑到物体中心点到物体底部的距离。需要先在RflySim3D中双击地面位置获取准确的Z坐标，然后根据物体模型的尺寸调整Z值，使物体底部与地面贴合。

### Q2：如何确定不同颜色靶标的ID？

A2：可以通过参考XML文件夹下的对应模型文件来确定ID。例如，绿色圆形靶标参考Ring\_Target\_Green.xml，蓝色靶标参考Ring\_Target\_Blue.xml，红色方框靶标参考

## Q3: 如何在不同地图场景中使用相同的靶标创建脚本?

A3: 只需在脚本开头添加地图切换命令: `ue.sendUE4Cmd(b'RflyChangeMapbyName [地图名称];0)`, 然后使用对应地图中测量得到的坐标即可。

---

1. <https://rflysim.com/> ↩
2. 推荐配置请见: <https://rflysim.com/doc/zh/HowToInstall.pdf> ↩