Pro 3D Camera

Getting Started

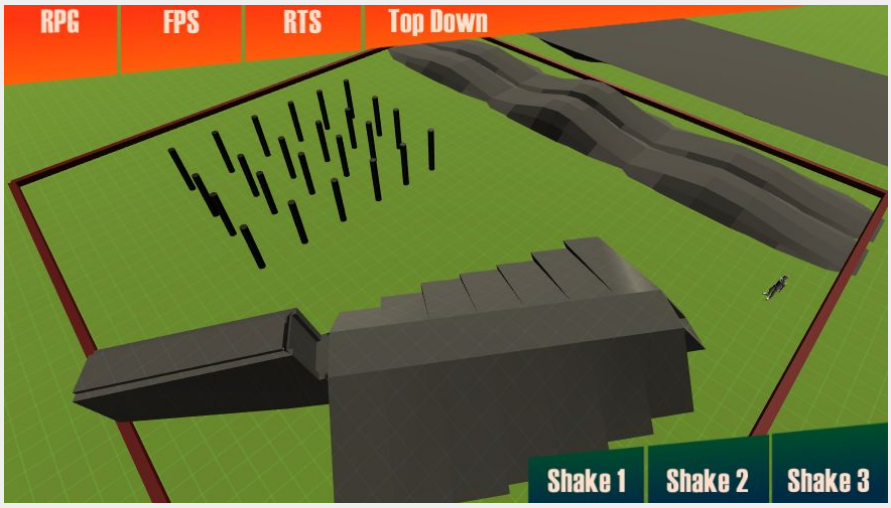
Thank you for your purchase.

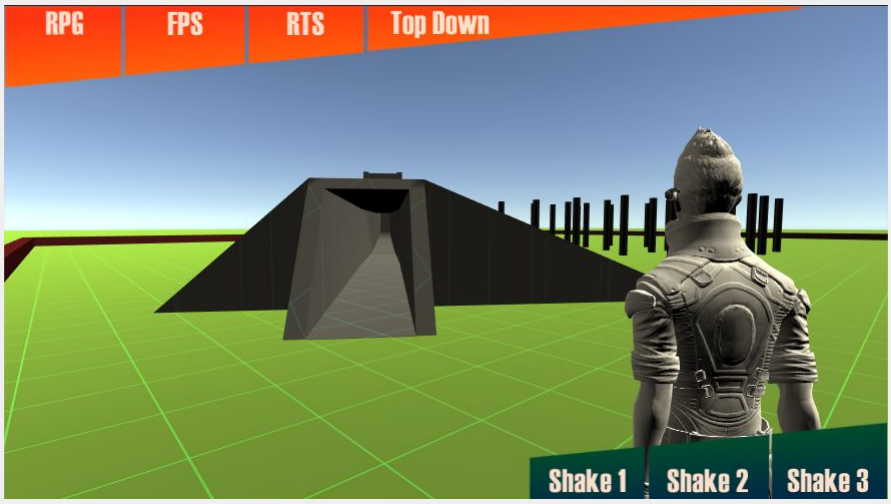
This small investment will make your life so much easier.

---

4 Cameras
1 Player Controller
Over 70 Settings
Easy Editor
Save Custom Settings Presets
API Documentation

Contents





Scene Components

The sample scene contains components for teaching you the basics of the Pro 3D Camera system. By studying the sample scene, you will learn the following:

- ➔ Preparing a custom target for the camera
- ➔ Tweaking the player controller
- ➔ How to use the API

The process to set the system up with custom variables (targets and environments) is quite simple, and we will cover the process in just a few simple steps.

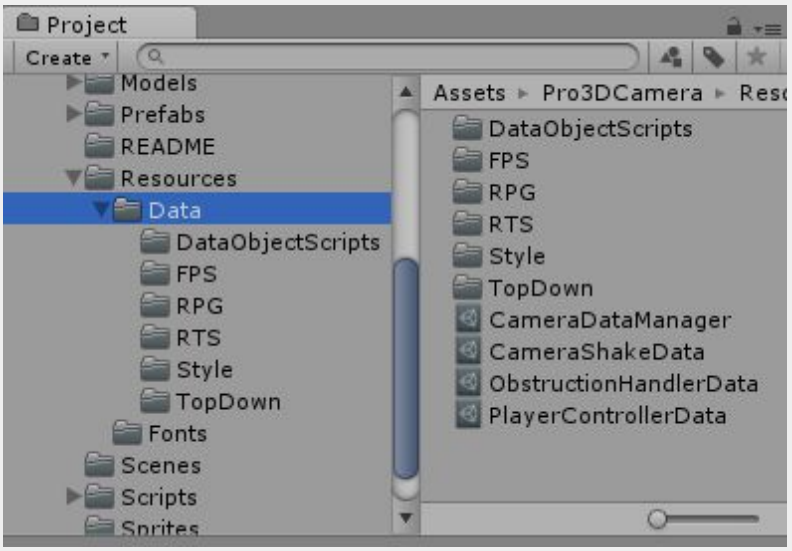
The test scene is built to give you a foundation for testing your camera settings. Within it, is wide open space, small nooks, and a mix between the two. In the rest of this document, you will receive tips about how to optimize the system for your projects.

Setting Up Your Target

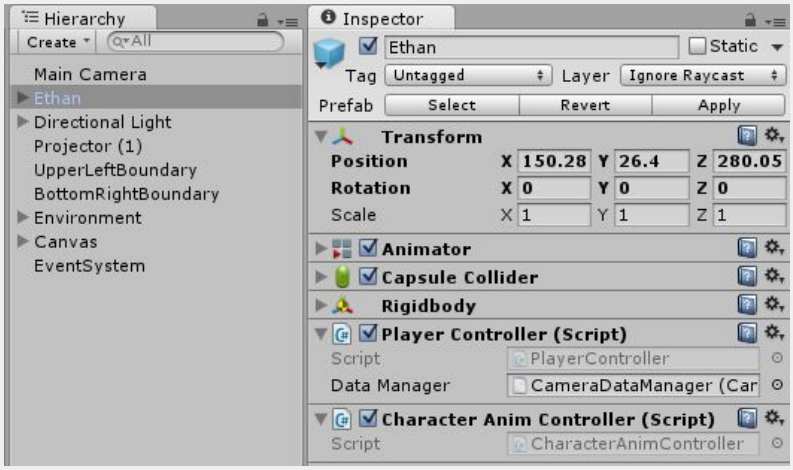
There are three things you should consider when setting up your own target and camera:

- ➔ Player Controller
- ➔ Obstruction Handler
- ➔ Camera Control

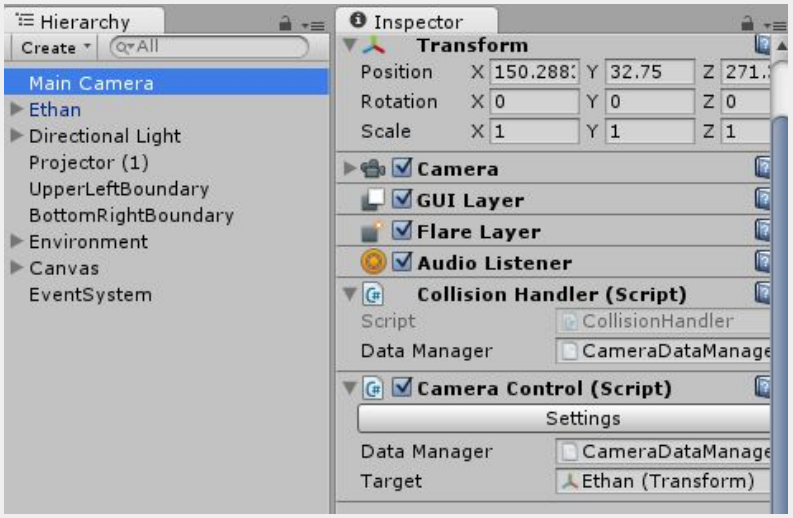
For each of these sub-systems, you will need to attach the overseeing data manager, which can be found at Pro3DCamera -> Resources -> Data. See the figure to the right.



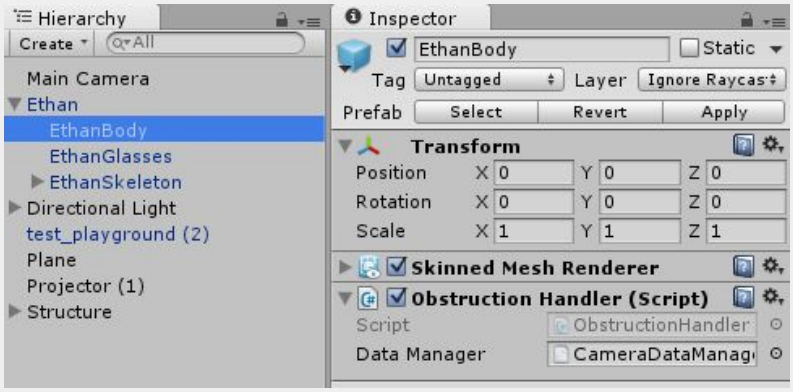
Drag this data manager object into the necessary component field for your player controller, obstruction handler, and camera control. Attempt to match the images below.



The CameraDataManager object placed on the PlayerController of your target.



The CameraDataManager placed on the CameraControl and CollisionHandler components on your main camera object. Note the target field of the CameraControl component.



The CameraDataManager object placed with any SkinnedMeshRenderer - on the Obstruction Handler component of your target.

Successfully placing the CameraDataManager object on the necessary components will fully prepare the Camera Control settings for use. Be sure to drag your custom target on the target field of the CameraControl component.

As a quick note - the rigidbody of your target should adhere to the one on the Ethan prefab in the example scene. Refer to his rigidbody to ensure yours matches up.

For more details on player controller settings, see the Reference guide which accompanied this in the Pro3DCamera -> README folder.

Using the API

The scripting API provides you with some handy functions for getting data or manipulating camera controls at runtime. You can read the complete reference guide in the other PDF that came with this package in the Pro3DCamera -> README folder.

With the full reference list, you will find how easy it is to accomplish necessary tasks - such as switching the camera type, changing targets, or offsetting the camera at runtime. In short, there are only three steps when accessing the API functions and properties. Those steps are outlined on the left.

Step 1:

Access the namespace

```
using Pro3DCamera; //use this to access the Camera Control API
```

Step 2:

Access the current CameraControl object

```
CameraControl _camControl;

void Start()
{
    _camControl = Camera.main.GetComponent<CameraControl>();
}
```

Step 3:

Call the function or property you need to use!

```
//Transition to a new camera type
_camControl.SetCameraType(CameraControl.CameraType.TOP_DOWN);
```