**在游戏中改变地形高度**

Posted on 2013年04月17日 by U3d / [Unity3D 基础教程](http://www.unitymanual.com/category/manual/unity3d-%e5%9f%ba%e7%a1%80%e6%95%99%e7%a8%8b)/被围观 146 次

如果我们制作的游戏可以在游戏进行中，因为某种撞击或爆炸而使地面产生凹陷或突起等变形。这该怎做呢？使用很多的地板模型去动态拼接吗？还是利用程式去控制地板 Mesh 顶点的变化？这么做会不会有拼接不完整或是形状变得很奇怪的地方呢？如果对自己的电脑图学演算及本身的程式与建模技术相当有自信的话，可以尝试用自己的方式完成需求；但如果没有这些功力，可以考虑使用 Unity 内建的 TerrainData class。

使用 Unity 的 Terrain > Create Terrain 建立地形除了容易学且方便使用外，如果我们点选场景(Scene)视窗 的 tab 下面那个选单选择 Tex-Wire 来显示出场景中模型的网格，将会发现 terrain 上的网格是会动态变化的，完全没有高低起伏的平面，网格的分布是很平均的，但是当地形有起伏时，Unity 则会自动做些改变使地形更灵活；所以如果要动态的改变地形做些应用，使用 Unity 内建的 Terrain 也是个相当便利的。

完全平面的网格分布很平均，地形有起伏时会自动改变。

然而，我们现在看到的地形编辑是在 Unity Editor 中，那麽如何在游戏执行期控制它呢？我们可以从官网的 Scripting > Runtime Classes > Terrain 找到，这些地形资料其实是记录在 terrain.terrainData 中，其中包含了地形高度、贴图、树、植被等资料，详细内容可查看官网 Scripting > Runtime Classes > TerrainData，既然官方已经提供了这些东西，那麽在执行期要控制地形变化就不难了。

首先，我们来做些准备动作，因为地形是有高度的上下限，所以我们在设计地形时，要先利用 Terrain > Flatten Heightmap 把地面整平提高，这样才不会让地形到时候凹陷不下去。另外，就是利用 Terrain > Set Resolution 做些地形设定，例如 Terrain Width 及 Terrain Length 代表地形的长宽，它们的值储存在 terrain.terrainData.size；HeightMap Resolution 可说是等高线控制高度的控制密度，它同时代表 terrain.terrainData.heightmapWidth 及 terrain.terrainData.heightmapHeight 的值。

接下来，因为控制地形资料中高低位置的点是受到地形长宽及 HeightMap Resolution 值的影响，所以我们无法直接从 3D 空间座标去改变它，而必须经过换算去找到正确的控制点；通常建立地形时，预设地形的位置是会在 ( 0 , 0 , 0 )，但依实际需求，我们的地形可能也未必都在固定位置，所以这个相对位置也可能需要换算，经过这些换算，我们就能找出地形内的控制点，并有机会试图改变它们的高度，以下为换算方式：

|  |  |  |
| --- | --- | --- |
|  |  |  |

|  |  |
| --- | --- |
| 01 | *//地形资料* |
| 02 |  |
| 03 | TerrainData terrainData = terrain.terrainData; |
| 04 |  |
| 05 | *//物件在地形上的相对位置* |
| 06 |  |
| 07 | Vector3 terrainLocalPos = obj.transform.position - terrain.transform.position; |
| 08 |  |
| 09 | *//地形内的控制点* |
| 10 |  |
| 11 | Vector2 controlPos = new Vector2( terrainLocalPos.x / terrainData.size.x \* terrainData.heightmapWidth , terrainLocalPos.y / terrainData.size.z \* terrainData.heightmapHeight ); |

找到控制点之后，就可以开始改变高度，改变高度前必须要先取得原本的高度值，而 Unity 的地形高度资料数值是介於 0 ~ 1 之间，所以不管在 Set Resolution 里设定的 Terrain Height 值是多少，我们都必须要以此为比例去换算为正确的数值；我们可以使用 terrain.terrainData.GetHeight() 取得单一点的高度值或使用 terrain.terrainData.GetHeights() 取得一个范围的高度值，并利用 terrain.terrainData.SetHeights() 写入新的高度值。

其中，要特别注意的是 terrain.terrainData.GetHeight() 取得的数值是介於 0 到 Set Resolution 的 Terrain Height 值之间，terrain.terrainData.GetHeights() 取得的数值是介於 0 ~ 1 之间，而 terrain.terrainData.SetHeights() 写入的值必须介於 0 ~ 1 之间，如果写入的数值高於 1 或是低于 0，Unity 的地形系统将会自动调整高度为最高或最低 (0 或 Set Resolution 的 Terrain Height 值)，这样可能会使得地形变动太过突兀；以下分别列出以供参考：

第一种：

|  |  |  |
| --- | --- | --- |
|  |  |  |

|  |  |
| --- | --- |
| 01 | *//要增加的高度* |
| 02 |  |
| 03 | **float** addHeight = 1.0f; |
| 04 |  |
| 05 | *//取出原来的高度* |
| 06 |  |
| 07 | **float** oldHeight = terrainData.GetHeight( (**int**)controlPos.x , (**int**)controlPos.y ); |
| 08 |  |
| 09 | *//新的高度资料* |
| 10 |  |
| 11 | **float**[,] newHeightData = new **float**[1,1] { { ( oldHeight + addHeight ) / terrainData.heightmapScale.y } }; |
| 12 |  |
| 13 | *//设定新的高度资料* |
| 14 |  |
| 15 | terrainData.SetHeights( (**int**)controlPos.x , (**int**)controlPos.y , newHeightData ); |
| 16 |  |
| 17 |  |

第二种：

|  |  |  |
| --- | --- | --- |
|  |  |  |

|  |  |
| --- | --- |
| 01 | *//要增加的高度* |
| 02 |  |
| 03 | **float** addHeight = 1.0f; |
| 04 |  |
| 05 | *//取出原来的高度* |
| 06 |  |
| 07 | **float**[,] oldHeightData = terrainData.GetHeights( (**int**)controlPos.x , (**int**)controlPos.y , 1 , 1 ); |
| 08 |  |
| 09 | *//新的高度资料* |
| 10 |  |
| 11 | **float**[,] newHeightData = new **float**[1,1] { { addHeight / terrainData.heightmapScale.y + oldHeightData[0,0] } }; |
| 12 |  |
| 13 | *//设定新的高度资料* |
| 14 |  |
| 15 | terrainData.SetHeights( (**int**)controlPos.x , (**int**)controlPos.y , newHeightData ); |

以上，我们利用简短的程式码让游戏在执行期间改变地形高低，当然，每次只有一个点改变高度好像有点怪怪的，但依照以上的原理，我们可以轻松的去改变地形上任何位置的高度，所以，也可以利用找到的控制点延伸运算出一个范围的控制点，来做区域性的地形高低变动。