18 立方体贴图

18.1 什么是立方体贴图

立方体贴图的主要思路是：存储6个纹理，将它们分别看作立方体的6个面。在Direct3D中，立方体图被表示为一个由6个元素所构成的纹理数组。

寻找立方体图中纹素的方法与普通的2D纹理并不相同，此时不再用2D纹理坐标来指定纹素，而是使用3D纹理坐标：它定义了一个起点位于原点的查找向量v。向量v与立方体图相交处的纹素即为v的3D坐标所对应的纹素。

在HLSL中，立方体纹理用TextureCube类型表示。(对立方体图进行采样代码)

18.2 环境贴图

立方体图的主要应用是环境贴图。其思路是使视场角为90度的摄像机位于场景中某物体的中心点O处。这样一来，此摄像机就能沿x,y,z三轴的正负共6个方向观察，并以这6个视角来截取场景中的图像(此物体O除外)。由于视场角为90度，所以这6张以物体O视角截取的图像，涵盖了包围着它的整个环境。接着，我们把这6张周围环境图像存于一个立方体图中，这也正是环境图这个名字的由来。

在为立方体图纹理资源创建SRV时，其维度指定为D3D12\_SRV\_DIMENSION\_TEXTURECUBE且使用TextureCube属性。

18.3 绘制天空纹理

我们能够利用环境图绘制天空纹理。首先，要围绕整个场景来创建一个巨大的球体。这种方法的思路就是将纹理图投影到球面之上。

假设天空球距摄像机是无限远的，这样一来，无论摄像机移到场景中的哪个角落，我们都无法更加接近或远离天空球面。要实现这种无限远的天穹，可以在世界空间里简单地将天空球的中心置于摄像机上，天空球会随着摄像机而移动。

(实现天穹效果的着色器文件)

18.4 模拟反射

本节我们将展示如何运用环境图去模拟来自周围环境的镜面反射。当我们为构建环境图而关于点O渲染场景时，实则是在点O处记录来自四面八方的光照数据。换句话说，环境图存储的是从各个方向照射到点O处的光照值，因此我们可以把环境图上的每个纹素看作一个光源。通过这些数据便可以近似地计算出来自周围环境光的镜面反射情况。

由于讨论的是与反射相关的内容，所以自然少不了提及菲涅尔效应。它基于表面的材质属性，光向量(反射向量)与法线之间的夹角，从而确定从环境中反射到观察者眼中的光量。除此之外，我们还要根据材质的光泽度增减反射值，即粗糙材质反射的光量较低，即便如此，这些较小的反射值仍不可忽略。

(立方体图查找向量的计算方法)

18.5 动态立方体

若采用实现生成的立方体图，我们就不能用它来捕捉哪些动态物体，这也就意味着不能绘制出动态物体的反射镜像。为了克服这种限制，就应在运行时动态地构建立方体图。即我们在每一帧都要将摄像机置于场景之内，以它作为立方体图的原点，沿着坐标轴共六个方向将场景分六次逐个渲染到立方体图的对应面上。由于在每一帧都会重建立方体图，因此能捕捉到场景中的动态物体及其动态的反射镜像。

18.5.1 动态立方体图辅助类

18.5.2 构建立方体图资源

18.5.3 分配额外的描述符堆空间

18.5.4 构建描述符

18.5.5 构建深度缓冲区

18.5.6 立方体图的视口与裁剪矩形

18.5.7 设置立方体图摄像机

18.5.8 对立方体图进行绘制

18.6 用几何着色器绘制动态立方体图

在上一节中，我们将场景反复绘制6次，并依次渲染到每个立方体图的面上，以此来生成立方体图。绘制调用是有开销的，所以应尽量减少调用次数。Direct3D 10有一名为”CubeMapGS”的示例，它通过几何着色器仅需绘制一遍场景即可渲染好一幅立方体图。

(例子见书本)