# 17.1 立方体贴图映射

在本章中，我们学习立方体贴图（cube map），它本质上是一种由6幅纹理组成的、按特殊方式解释的纹理数组。通过使用立方体贴图映射，我们可以很容易地对一个天空进行纹理映射或模拟反射。

**学习目标**

1．了解什么是立方体贴图以及如何在HLSL代码中对它们进行采样。

2．学习如何使用DirectX纹理工具创建立方体贴图。

3．学习如何使用立方体贴图来模拟反射。

4．学习如何使用立方体贴图来对一个天空穹顶进行纹理映射。

立方体贴图映射的实现思路是：将6幅纹理想像为关于某个坐标系原点和轴对齐的立方体的6个平面（“立方体贴图”这个名字正是由此而来）。由于立方体纹理是轴对齐的，它的每个平面都沿着3个主轴的方向放置；所以，我们可以根据与平面相交的主轴方向（±X，±Y，±Z）标识立方体贴图的每个平面。Direct3D提供了D3D11\_TEXTURECUBE\_FACE枚举类型来完成这一工作：

typedef enum D3D11\_TEXTURECUBE\_FACE

{

D3D11\_TEXTURECUBE\_FACE\_POSITIVE\_X = 0,

D3D11\_TEXTURECUBE\_FACE\_NEGATIVE\_X = 1,

D3D11\_TEXTURECUBE\_FACE\_POSITIVE\_Y = 2,

D3D11\_TEXTURECUBE\_FACE\_NEGATIVE\_Y = 3,

D3D11\_TEXTURECUBE\_FACE\_POSITIVE\_Z = 4,

D3D11\_TEXTURECUBE\_FACE\_NEGATIVE\_Z = 5,

} D3D11\_TEXTURECUBE\_FACE;

立方体贴图存储在一个纹理数组中，它包含6幅纹理：

**1．**索引0表示+X平面。

**2．**索引1表示–X平面。

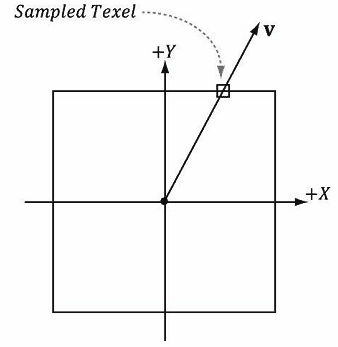
**3．**索引2表示+Y平面。

**4．**索引3表示–Y平面。

**5．**索引4表示+Z表面。

**6．**索引5表示–Z表面。

与2D纹理映射不同，我们在这里不能使用2D纹理坐标指定纹理元素。要指定一个立方体贴图中的纹理元素，我们必须定义一个从原点引出的查找向量**v**，通过3D纹理坐标指定纹理元素。查找向量v与立方体贴图相交的地方（参见图 11.1）就是3D纹理坐标**v**对应的纹理元素。我们在第8章讨论的纹理过滤概念同样适用于立方体贴图采样。

****

**图17.1 为简单起见，我们在2D空间中描述一概念；读者可以把个正方形想像为一个3D空间中的立方体。该正方形表示一个与坐标系原点和轴对齐的立方体贴图。我们从原点引出一个向量v。与v相交的纹理元素就是所要采样的纹理元素。在本图中，v与立方体贴图的+Y平面相交。**

**注意**：查找向量的大小并不重要；重要的只是它的方向。使用方向相同、大小不同的两个向量对立方体贴图进行采样得到的结果完全相同。

在HLSL中，立方体纹理由**TextureCube**类型表示。下面的代码片段说明了应该如何对立方体贴图进行采样：

TextureCube gCubeMap;

SamplerState gTriLinearSam

{

Filter = MIN\_MAG\_MIP\_LINEAR;

AddressU = Wrap;

AddressV = Wrap;

};

...

// 在像素着色器中

float3 v = float3(x,y,z); // 用于查找纹理坐标的向量

float4 color = gCubeMap.Sample(gTriLinearSam,v);

**注意**：查找向量和立方体贴图应该使用相同的坐标系。例如，当立方体贴图使用世界坐标系时（即，立方体平面与世界空间的主轴对齐），查找向量也应该使用世界坐标系。