

## 学习笔记\_法线贴图

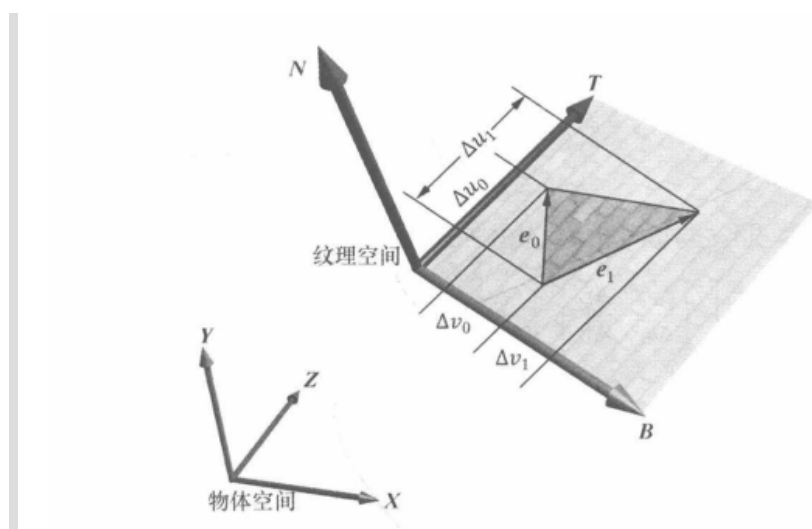
笔记本: DirectX 12

创建时间: 2022/9/20 11:23

更新时间: 2022/9/21 8:52

作者: handsome小赞

- 假设以下示例所用的是 24位图像格式，即将每个颜色分量都存于 1个字节之中。因此，每个颜色分量的取值范围都为  $[0,255]$ 。至于 32位的图像格式，其中的alpha分量可以保留不用，也可以用于存储其他的标量数据，如地形的高度图，高光图（也称为镜面反射等）。浮点格式不必进行压缩，但是这将耗费更多的内存。
- 纹理空间 /切线空间



- 顶点切线空间

逐顶点，通过共享顶点v的每个三角形的 切/副切向量平均值，求得 切/副切向量 T/B

只需存储 T、N，即可通过  $N \times T$  求得 B

- 在切线空间与物体空间之间进行转换

- 由切线空间变换至物体空间

$$M_{object} = \begin{bmatrix} T_x & T_y & T_z \\ B_x & B_y & B_z \\ N_x & N_y & N_z \end{bmatrix}$$

- 若由物体空间变换至切线空间，因为上述矩阵为正交矩阵，其逆矩阵就是转置矩阵

- 法线贴图的着色器代码

法线贴图的流程大致如下。

1. 通过艺术加工工具或图像处理程序来创造预定的法线图，并将它存于图像文件之中。在应用程序初始化期间以这些图像文件来创建 2D 纹理。
2. 针对每一个三角形，计算其切向量  $T$ 。通过对网格中共享顶点  $v$  的所有三角形的切向量求取平均值，就可以获取此网格中每个顶点  $v$  处的切向量（在演示程序中，由于使用的是简单的几何图形，因此可以直接指定出相应的切向量。但是，如果处理的是由 3D 建模程序创建的不规则形状的三角形网格，那么就需要按上述方法来计算切向量的平均值）。
3. 在顶点着色器中，将顶点法线与切向量变换到世界空间，并将结果输出到像素着色器。
4. 通过插值切向量与插值法向量来构建三角形表面每个像素点处的 TBN 基，再以此 TBN 基将从法线图中采集的法向量由切线空间变换到世界空间。这样一来，我们就拥有了取自法线图的世界空间法向量，并可将其用于往常的光照计算。

下列代码通过使  $T$  减去其  $N$  方向上的分量（投影），再对结果进行规范化处理，从而使  $T$  成为规范化向量且正交于  $N$ 。这里假设 `unitNormalW` 为规范化向量。

```
// 构建正交规范基
float3 N = unitNormalW;
float3 T = normalize(tangentW - dot(tangentW, N)*N);
float3 B = cross(N, T);
```