# 凹凸映射(bump mapping)

**作用：**

使用一张纹理来修改模型表面的法线，以便为模型提供更多的细节。这种方法不会真的改变模型的顶点位置，只是让模型看起来像是凹凸不平。

**实现方法：**

1：使用一张高度纹理(height map)来模拟表面位移(displacement),然后得到一个修改后的法线值。这种方法叫——**高度映射(height mapping)。**

2：使用一张法线纹理(normal map)来直接存储表面法线，这种方法叫——法线映射(normal mapping)。(常用选择)。

**实现思路：**

第二种方法：

a.在顶点函数中实现：

*a1.将主纹理和凹凸纹理的坐标信息，分别保存在uv.xyzw分量中(通宵使用同一组纹理坐标即可，即两纹理共用\_Mathf.\_ST)。*

*a2.将模型空间下，切线方向、副切线方向、法线方向按行排列来得到从模型空间到切线空间的变换矩阵。(内置宏 TANGENT\_SPACE\_ROTATION)。*

*a3.使用内置函数UnityObjSpaceLightDir()和UnityObjSpaceViewDir()得到模型空间下的光照和视角方向。*

*a4.再利用变换矩阵rotation把它们从模型空间变换到切线空间中。*

b.在片元函数中实现：

b1.归一化光照向量和视角向量。

b2.对bumpmap进行采样，得到切线空间下的法线向量。如果该法线纹理的类型没有设置成Normal map，就需要把法线纹理中，经过映射后的像素值，反映射回来。

b3.

详见7.2.1章节。代码2/BumpMapping