

# ESライブラリ&& ゲームプログラミング

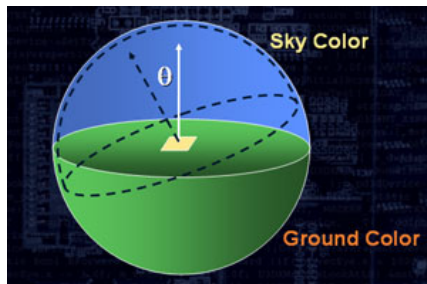
## バーテックスシェーダー編 - 第8回 半球ライティング

### 半球ライティング

- ・半球ライティングは、空と地面の2つの環境光を頂点の向きによって補間する照明処理
- ・シーンの各場所における局所的な間接光、相互反射を簡易的に表現できる
- ・半球の軸(地面の傾き)と法線ベクトル、空と地面の色から計算される

### 半球ライティング

半球ライティングは、環境光を表現する方法のひとつです。これまで単色だった環境光ですが、半球ライティングでは、空から降り注ぐ光の色と地面から照り返される光の色の2つ用意し、頂点の向きによって2つを補間することにより、環境光の表現力をより豊かにするものです。モデルの位置によって色を変えることにより、シーン各場所における局所的な間接光を表現できます。



半球ライティングでは、次の3つの要素と頂点の法線をもとに、以下の計算式で求めます。

1. 上半球の色(空、光源の色など)
2. 下半球の色(地面など照り返しの色)
3. 半球の軸(地形の回転軸、地面の傾きなど)

アンビエント = 上半球の色 \*  $\cos$  + 下半球の色 \*  $-\cos$

頂点の向きと軸の角度によって、上と下の色の混合率を変えるというものです。頂点と軸が同じ向きであればあるほど、上向きと見なし、上半球の色の割合が大きくなり、逆は下向きなので下半球の色の影響が大きくなるようになっています。

ここで、上記の計算式は、 $-1.0 \sim +1.0$ の範囲になります。色の範囲は $0.0 \sim 1.0$ なので、式を変形し、色の有効範囲内に収まるようにします。 $\cos$ に1を足して( $0.0 \sim 2.0$ の範囲になる)、色との乗算結果を $1/2$ ( $0.5$ 倍)すれば、 $0.0 \sim 1.0$ の範囲になります。

よって、

$$\begin{aligned} & \text{上半球の色} * \cos + \text{下半球の色} * -\cos \\ & \text{上半球の色} * (\cos + 1) * 0.5 + \text{下半球の色} * (-\cos + 1) * 0.5 \\ = & \text{上半球の色} * (1 + \cos) * 0.5 + \text{下半球の色} * (1 - \cos) * 0.5 \\ = & (\text{上半球の色} * (1 + \cos) + \text{下半球の色} * (1 - \cos)) * 0.5 \end{aligned}$$

ここで、 $\theta$ は「頂点の向きと軸のなす角」です。 $\cos$ は「なす角の余弦」ですが、これは内積におきかえることができます。よって $\cos$ は「頂点の法線ベクトル 内積 軸のベクトル」となります。

## 課題

半球ライティングの計算式をもとに、半球ライティングでアンビエントを計算するバーテックスシェーダーを作成しましょう。