

明暗 shading

主要任务

- 求出图像中各个像素的颜色亮度
- 用户指定颜色 + 插值
- 使用光照模型计算

OpenGL 中的明暗处理

简单光照模型 (局部光照模型)

- 物体的材质
 - 物体的几何属性 (法向)
 - 光源信息 (几何, 类型, 强度)
 - 观察者 / 相机的位置
-
- Gouraud 模型
 - OpenGL 利用简单光照模型计算出顶点的颜色
 - 插值出多边形内各点的颜色

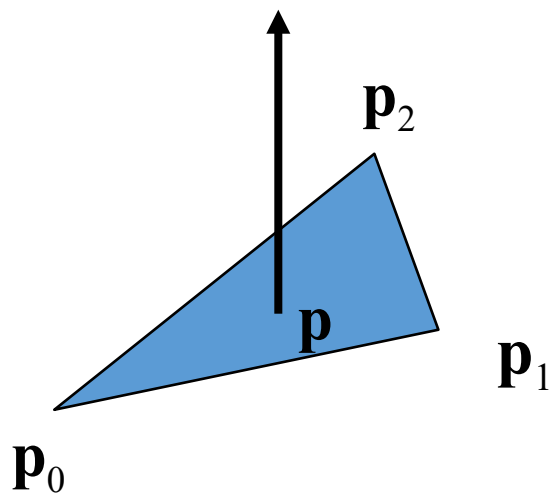
法向量

- 三角形

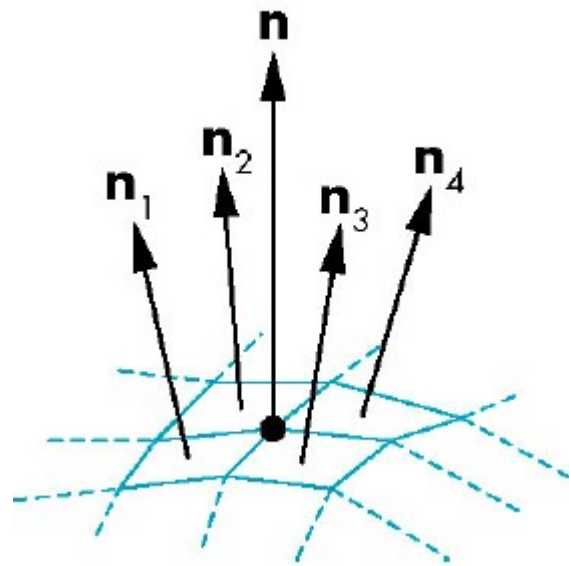
plane $\mathbf{n} \cdot (\mathbf{p} - \mathbf{p}_0) = 0$

$$\mathbf{n} = (\mathbf{p}_2 - \mathbf{p}_0) \times (\mathbf{p}_1 - \mathbf{p}_0)$$

normalize $\mathbf{n} \leftarrow \mathbf{n} / |\mathbf{n}|$



- 网格模型，有几个多边形交于某一该点
- 每个多边形有一个法向，
- 取这几个法向的平均得到该点的法向



$$n = \frac{n_1 + n_2 + n_3 + n_4}{|n_1 + n_2 + n_3 + n_4|}$$

有关函数

- 设置光源属性
- 设置物体的材质属性
- 设置顶点法向量！
- 启用光照

启用光照

- `glEnable(GL_LIGHTING)`
- 如果启用光照，`glColor()` 命令无效
- 可单独启用各光源：`glEnable(GL_LIGHTi)`, $i = 0, 1, \dots, 7$

法向量

- 是 OpenGL 状态机的一部分！
- `glNormal*()` ； `glNormal3d(x,y,z)`；； `glNormal3dv(p)`;
- 变换会影响其长度
- 通常需要法向量为单位向量
- `glEnable(GL_NORMALIZE)`

光照模型的参数

- `glLightModel{if}[v](参数 , 值)`
- `GL_LIGHT_MODEL_AMBIENT`, 整个场景中的环境光强
- `GL_LIGHT_MODEL_LOCAL_VIEWER` , 0.0 或 `GL_FALSE`, 不应用无穷远视点的简化计算
- `GL_LIGHT_MODEL_TWO_SIDED` , 0.0 或 `GL_FALSE`, 单独对多边形的两面进行明暗处理
- `GL_LIGHT_MODEL_COLOR_CONTROL, GL_SINGLE_COLOR`, 镜面光是否与漫反射和环境光分开计算

定义点光源

- 位置
- 光强：漫反射光、镜面光和环境光的 RGBA 值

```
GLfloat diffuse0[]={1.0,0.0,0.0,1.0};  
GLfloat ambient0[]={1.0,0.0,0.0,1.0};  
GLfloat specular0[]={1.0,0.0,0.0,1.0};  
GLfloat light0_pos[]={1.0,2.0,3.0,1.0};
```

```
glEnable(GL_LIGHTING);  
glEnable(GL_LIGHT0);  
glLightfv(GL_LIGHT0, GL_POSITION, light0_pos);  
glLightfv(GL_LIGHT0, GL_AMBIENT, ambient0);  
glLightfv(GL_LIGHT0, GL_DIFFUSE, diffuse0);  
glLightfv(GL_LIGHT0, GL_SPECULAR, specular0);
```

- 位置
 - 齐次坐标形式
 - 如果 $w = 1.0$, 指定的是一个空间位置
 - 如果 $w = 0.0$, 指定的是一个平行光源，所给定的是入射光方向
- 光源是几何对象，它的位置或方向受模型视图矩阵的影响

- 距离项，衰减

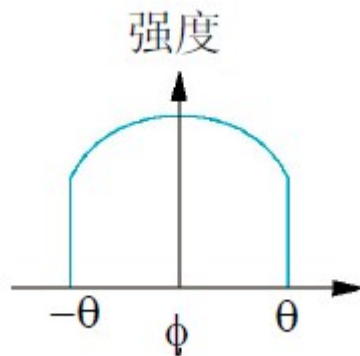
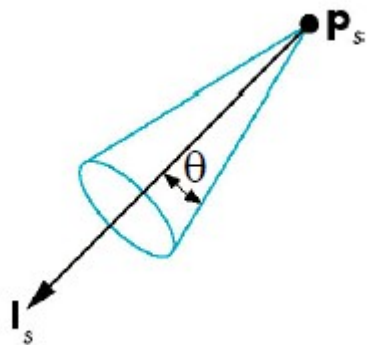
- 光强反比于距离的因子 $a + bd + cd^2$
- 默认值： $a = 1.0, b = c = 0.0$

- 改变方法

- `glLightf(GL_LIGHT0, GL_CONSTANT_ATTENUATION, 2.0);`
- `glLightf(GL_LIGHT0, GL_LINEAR_ATTENUATION, 1.0);`
- `glLightf(GL_LIGHT0, GL_QUADRATIC_ATTENUATION, 0.0);`

聚光灯

- `glLightfv ()`
- 方向 : `GL_SPOT_DIRECTION`
- 角度范围 : `GL_SPOT_CUTOFF`
- 衰减指数 : `GL_SPOT_EXPONENT`



- 全局环境光
- `GLfloat global_ambient[]={0.2,0,0,1};`
- `glLightModelfv(GL_LIGHT_MODEL_AMBIENT,global_ambient);`

材质属性

- 是 OpenGL 状态机的一部分
- 与简单光照模型中的各项匹配
- `glMaterial{if}[v]()`

```
GLfloat ambient[]={0.2,0.2,0.2,1.0};
```

```
GLfloat diffuse[]={1.0,0.8,0.0,1.0};
```

```
GLfloat specular[]={1.0,1.0,1.0,1.0};
```

```
GLint shine = 100;
```

```
glMaterialfv(GL_FRONT, GL_AMBIENT, ambient);
```

```
glMaterialfv(GL_FRONT, GL_DIFFUSE, diffuse);
```

```
glMaterialfv(GL_FRONT, GL_SPECULAR, specular);
```

```
glMateriali(GL_FRONT, GL_SHININESS, shine);
```

前面与后面

- 默认状态下只是对物体的前面进行明暗处理
- 如果设置了两面光照，每一面都可以具有自己的属性：
- `glMaterialf()`
 - `GL_FRONT`, `GL_BACK`, 或者 `GL_FRONT_AND_BACK` 指定

发射项

- 模拟光源
- 该项的颜色不受任何其它光源或者变换的影响
- `GLfloat emission[]={0.0,0.3,0.3,1.0};`
- `glMaterialfv(GL_FRONT, GL_EMISSION, emission);`

多边形的明暗处理

- 对每个顶点进行明暗处理的计算
- 默认状态下，多边形内部的颜色是顶点颜色的线性插值
- `glShadeModel(GL_SMOOTH);`
- 设置 `glShadeModel(GL_FLAT);` 第一个顶点的颜色确定整个多边形的颜色

总结

- Enable shading and select model
- Specify normals
- Specify material properties
- Specify lights

实验内容

- 场景中有三个光源，物体为茶壶或其它。
- 自行设定光源和物体材质的基本属性。
- 用可通过键盘或鼠标改变光源的位置和类型。