

## 第 13 章

# シェーディング

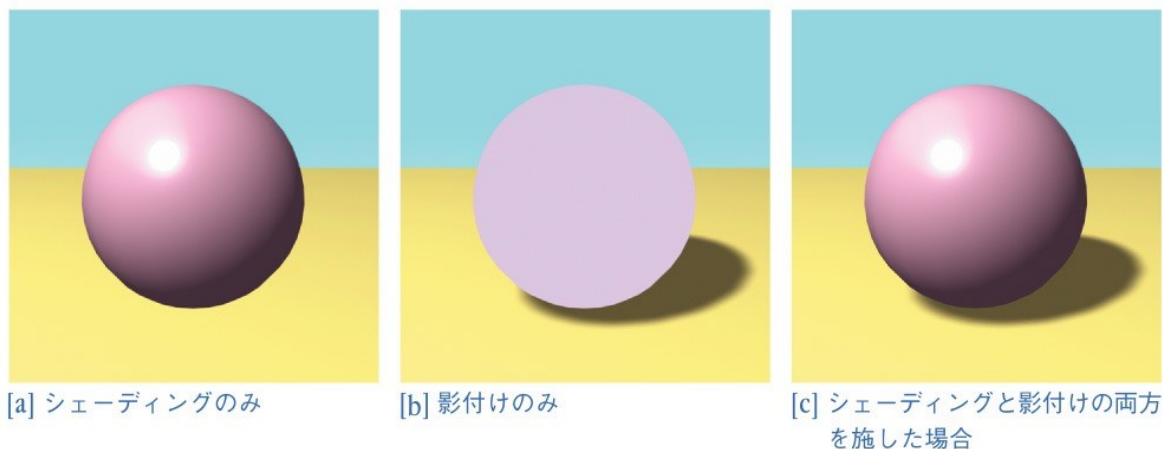
### 重要

- シェーディングモデル: 照明モデル, ライティングモデル
  - 局所照明モデル: 直接光
  - 大域照明モデル: 直接光 + 間接光 (反射, 透過, 屈折)
- 鏡面反射: Phong のモデル, Blinn のモデル, Cook-Torrance のモデル
- スムースシェーディング: グローシェーディング, フォンシェーディング

### 13.1 シェーディングモデル

光源によって照射された物体は図 13.1(a) に示すように, 物体表面の場所によって明るさがる. つまり, 光の照射方向と物体表面の向きによって明るさが変化している. このように光の当たり具合によって濃淡が変化する様子を計算して表示することを CG では陰付け (shading) とよぶ. また, 図 13.1(b) のように球によって光源からの光が遮られた床面の部分には影が生じる. この光が遮られて生じる影を計算して表示する処理を影付け (shadowing) とよぶ.

■図4.24——シェーディングと影付け



「コンピュータグラフィックス」2004年 / 財団法人画像情報教育振興協会 (CG-ARTS協会)

図 13.1 シェーディングとシャドウイング

どのような物理モデルを用いて光源によって照射された物体の色を決定するモデルをシェーディングモデル (Shading

Model) とよぶ。シェーディングモデルは、照明モデル (illumination model) やライティングモデル (lighting model) と呼ばれる。シェーディングモデルを大別すると直射光 (direct light) に関する物理的なモデルを用いて計算を行う局所照明モデル (local illumination model) と直射光に加え間接光 (indirect light) も考慮した物理的なモデルを用いて計算を行う大域照明モデル (global illumination model) がある。間接光も考慮した大域照明モデルの方が直接光だけを考慮した局所照明モデルよりも実写的な画像生成が可能である。

シェーディングモデルでは、光源 (light source) の種類や特性、光と対象物体との相互作用、反射、透過・屈折などの物質の光学的性質などを考慮する必要がある。光源の種類には、光線が同一方向に進む平行光線 (directional light)、一点の光源から光線が放射状に広がる点光源 (point light source)、光源がある長さを持つ線光源 (linear light source) やある大きさを持つ面光源 (area light source) がある。

光と物質との相互作用では、直接光と間接光を考慮する。直接光は光源から被照射面に直接到達する光線であり、間接光は光源から照射された光線が他の物体と相互作用を起こしたあとに別の物体の被照射面に到達する光線である。間接光では、他の物体との相互作用した後の光線のふるまいを計算するため、光線の反射 (reflection)、透過 (transmission)、屈折などの物質の光学的性質を考慮する必要がある。

反射光 (reflected light) は物体表面で反射、あるいは、物体内部で多重散乱した後に物体表面から出てきた光である。反射光は、拡散反射成分と鏡面反射成分に分けて取り扱うことができる。また、物質を透過した透過光 (transmitted light) や微粒子で散乱した散乱光 (scattered light) などもある。

## 13.2 鏡面反射 (specular reflection)

拡散反射は物体表面付近で光が多重散乱した後に物体表面から放射された光であるが、鏡面反射は物質表面で直接反射した光である。光沢のあるプラスチックや金属表面では、鏡面反射によってハイライト (highlight) が生じる。鏡面反射は図 13.2 に示すように、入射角と等しい反射角になる正反射方向に光が強く反射される。光源方向と物体面の向き、面の反射特性、視点方向に依存して反射光強度が変化する。ただし、実際の粗い面での鏡面反射は正反射方向と少しずれたところで反射強度が最大になる。

### 13.2.1 Phong のモデル

光沢のあるプラスチックなどの鏡面反射の計算には Phong のモデルが用いられる。Phong のモデルでは、次式に示すように視線方向と正反射方向のなす角  $\gamma$  の余弦の  $n$  乗で鏡面反射光の強さが減少する。

$$I = W(\alpha) I_i \cos^n \gamma = W(\alpha) I_i (\mathbf{R} \cdot \mathbf{V})^n \quad (13.1)$$

ただし、 $W(\alpha)$  は鏡面反射率で入射角  $\alpha$  の関数である。 $I_i$  は入射光の強さ、 $\mathbf{R}$  と  $\mathbf{V}$  はそれぞれ正反射方向ベクトルと視線方向ベクトルを表す。Phong のモデルでレンダリングした結果を図 13.3 に示す。

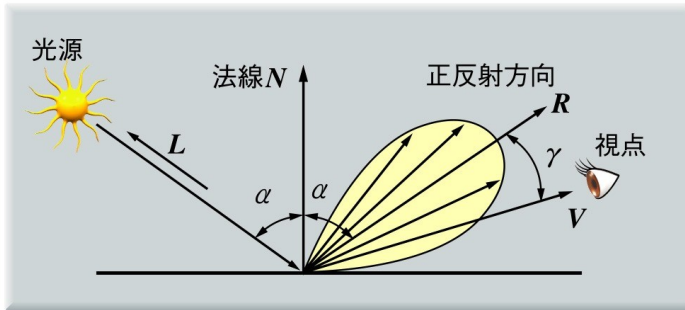
### 13.2.2 Blinn のモデル

Blinn のモデルでは、物体表面を微小な面の集合とみなし、微小面が面の法線方向を中心にガウシアン分布あるいは回転楕円体状分布していると仮定する。これらの分布関数は、図 13.4 に示す光源へのベクトル  $\mathbf{L}$  と視点へのベクトル  $\mathbf{V}$  の 2 等分ベクトル  $\mathbf{H}$ 、 $\mathbf{H}$  と面の法線ベクトル  $\mathbf{N}$  とのなす角  $\xi$  によって表される。

$$\mathbf{H} = \frac{\mathbf{L} + \mathbf{V}}{\|\mathbf{L} + \mathbf{V}\|} \quad (13.2)$$

微小面から完全鏡面反射して視点へ届く光の割合から反射面の強さを計算する。

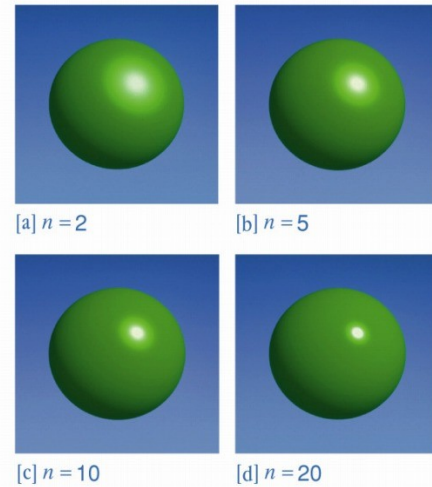
■図4.38——鏡面反射光



「コンピュータグラフィックス」2004年 / 財団法人画像情報教育振興協会 (CG-ARTS協会)

図 13.2 鏡面反射

■図4.39——フォンの反射モデルによるハイライトの違い



「コンピュータグラフィックス」2004年 / 財団法人画像情報教育振興協会 (CG-ARTS協会)

図 13.3 フォンのモデル

### 13.2.3 Cook-Torrance のモデル

Cook-Torrance のモデルは Blinn のモデルを改良したモデルであり、金属表面での反射に適している。このモデルでは、反射率  $\rho_s$  の分布が光の波長や入射角に依存することを考慮している。

$$\rho_s = \frac{F_\lambda}{\pi} \frac{D_\xi G}{(N \cdot L)(N \cdot V)} \quad (13.3)$$

$G$  は微小面の幾何学的形状によって光が遮蔽されることによって反射光が減衰する割合を表す幾何学的減衰係数、 $N$  は法線ベクトルであり、 $L$  と  $V$  はそれぞれ光源へのベクトルと視点へのベクトルを表す。

物質によって定まる反射率  $F_\lambda$  には次式で示すフレネルの反射率が用いられる。

$$F_\lambda = \frac{1}{2} \left\{ \frac{\sin^2(\theta_1 - \theta_2)}{\sin^2(\theta_1 + \theta_2)} - \frac{\tan^2(\theta_1 - \theta_2)}{\tan^2(\theta_1 + \theta_2)} \right\} \quad (13.4)$$

ただし、 $k_r$  は異なる反射率を持つ媒質での反射率、 $\theta_1$  と  $\theta_2$  はそれぞれ入射角と屈折角を表す。

$D_\xi$  は微小面の分布関数であり次式に示すベックマン (Beckman) 分布関数が用いられる。

$$D_\xi = \frac{1}{4m^2 \cos^4 \xi} \exp - \left( \frac{\tan \xi}{m} \right)^2 \quad (13.5)$$

ただし、 $m$  は面の粗さを調整する係数で、値が小さいと滑らかで鏡面に近づく。

## 13.3 物体の反射特性

### 13.3.1 環境光反射

例題 13.1 環境光の反射係数を設定せよ。(ambient.pde)

```
1 boolean isAmbientLight = true, isAmbient = true;
2 void setup(){
```

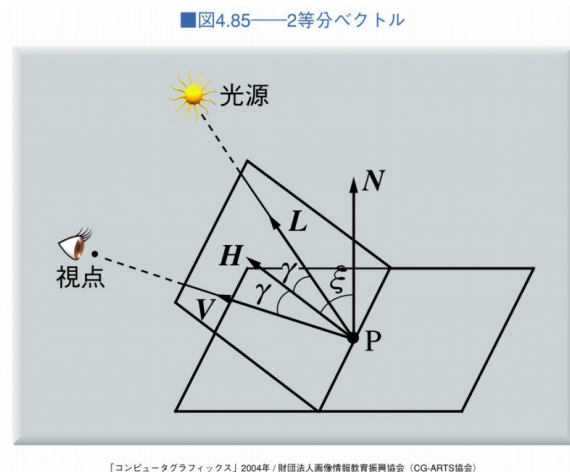


図 13.4 ブリンのモデル

■図4.86——クック・トランスのモデルを用いて金属の反射を表示した例

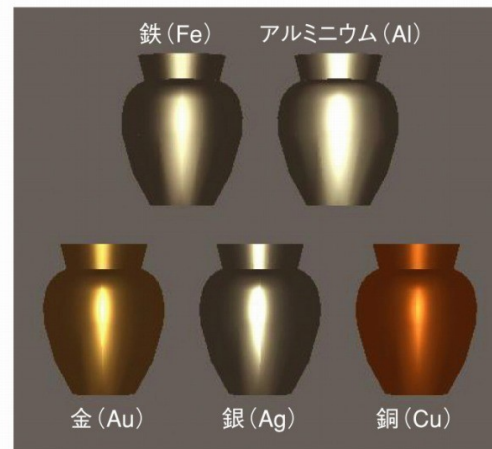


図 13.5 Cook-Torrance のモデル

```

3   size(400, 300, P3D);
4   noStroke();
5 }
6
7 void keyPressed(){
8     switch(key){
9         case 'l':
10            isAmbientLight = !isAmbientLight;
11            break;
12        case 'a':
13            isAmbient = !isAmbient;
14            break;
15    }
16 }
17
18 void draw(){
19     background(32);
20     directionalLight(255, 255, 255, map(mouseX, 0, width, 1, -1), map(mouseY, 0, height,
21         1, -1), -1);
22     if(isAmbientLight) ambientLight(128, 128, 0);
23     if(isAmbient) ambient(0, 128, 0);
24     else fill(128);
25     translate(width / 2, height / 2, 0);
26     sphere(100);
27 }

```

#### 環境光反射係数の設定

```

ambient(v1, v2, v3); // ambientLight で設定した環境光成分 (r, g, b) に対する
// 反射係数を設定 (255 で完全に反射, 128 で半分)

```

例題 13.2 鏡面反射係数を設定せよ. (specular.pde)

```
1 void setup(){
2     size(400, 300, P3D);
3     noStroke();
4     fill(192);
5 }
6
7 void draw(){
8     sphereDetail(int(map(mouseX, 0, width, 3, 200)));
9     background(32);
10    lightSpecular(255, 255, 255);
11    directionalLight(255, 255, 255, -.5, .5, -1);
12    translate(100, 150, 0);
13    specular(255, 255, 255);
14    sphere(60);
15    translate(200, 0, 0);
16    specular(255, 0, 0);
17    sphere(60);
18 }
```

## 鏡面反射係数の設定

```
specular(v1, v2, v3); // 鏡面反射の色を設定
```

## 球の詳細度の設定

```
sphereDetail(res); // セグメントの数を設定
```

## 13.3.3 光沢

例題 13.3 光沢を表現するプログラムを作成せよ. (shininess.pde)

```
1 void setup(){
2     size(400, 300, P3D);
3     noStroke();
4     fill(0, 192, 192);
5 }
6
7 void draw(){
8     sphereDetail(128);
9     background(32);
10    lightSpecular(255, 255, 255);
11    directionalLight(255, 255, 255, -.5, .5, -1);
12    specular(255, 255, 255);
13    shininess(map(mouseX, 0, width, 1, 500));
14    translate(width / 2, height / 2, 0);
15    sphere(100);
16 }
```

## 光沢の設定

```
shininess(shine); // 輝きの度合
```

### 13.3.4 発光

例題 13.4 発光する物体を表現するプログラムを作成せよ. (emissive.pde)

```

1  boolean isEmissive = true;
2  void setup(){
3      size(400, 300, P3D);
4      noStroke();
5      fill(255);
6  }
7
8  void keyPressed(){
9      switch(key){
10         case 'e':
11             isEmissive = !isEmissive;
12             break;
13         }
14     }
15
16     void draw(){
17         background(32);
18         directionalLight(255, 255, 255, -.5, .5, -1);
19         if(isEmissive) emissive(0, map(mouseX, 0, width, 0, 255), 0);
20         else emissive(0, 0, 0);
21         translate(width / 2, height / 2, 0);
22         sphere(100);
23     }

```

#### 発光色の設定

```
emissive(v1, v2, v3); // 発光色の設定
```

## 13.4 スムーズシェーディング (Smooth Shading)

曲面で表現された物体をレンダリングする場合、各画素に対応する面上点の法線を計算しシェーディングを行うことは膨大な計算量を必要とする。そのためポリゴンによる曲面近似を行いレンダリングを行うことが多い。

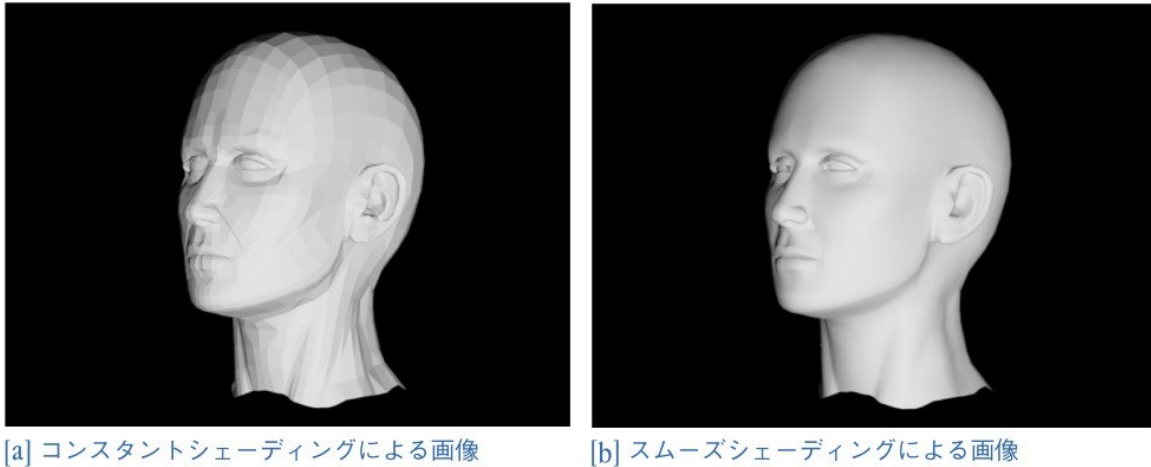
コンスタントシェーディング (constant shading) は、ポリゴンの代表点の明るさを計算してそのポリゴン全体を一定の明るさで表示する方法である。図 13.6(a) にコンスタントシェーディングを行った結果を示す。図 13.6(a) では、人間の知覚特性の一つである輝度変化に敏感で、実際の輝度差よりも大きく感じるマッハバンド効果 (Mach band effect) の影響で、ポリゴン間の境界が目立って知覚される。

一方、図 13.6(b) は、ポリゴン内の輝度を内挿したスムーズシェーディング (smooth shading) の結果を示している。この方法には、ポリゴン内の輝度を補間するグローシェーディング (Gouraud shading) と法線ベクトルを補間するフォンシェーディング (Phong shading) がある。

### 13.4.1 グローシェーディング

グローシェーディングでは、ポリゴン頂点の輝度を求め、頂点の輝度のバイリニア補間 (bilinear interpolation) でポリゴン内の輝度を求める。図 13.7 に示すようにポリゴンの各頂点を  $A, B, C, D$  とし、各頂点の輝度をそれぞれ

■図4.46——コンスタントシェーディングとスムーズシェーディングによる表示の違い



「コンピュータグラフィックス」2004年 / 財団法人画像情報教育振興協会（CG-ARTS協会）

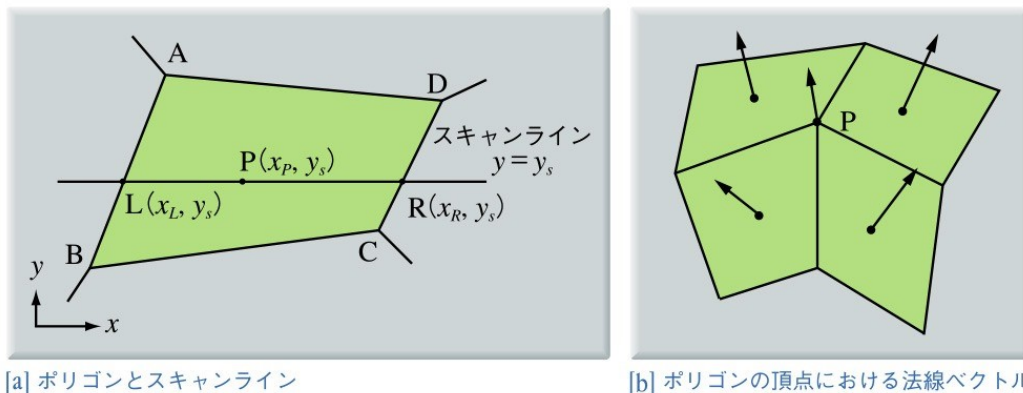
図 13.6 コンスタントシェーディングとスムーズシェーディング

$I_A, I_B, I_C, I_D$  とする。オブジェクトを近似する前の曲面から各頂点の法線を求め、その法線から輝度を算出する。曲面の式が利用できない場合は、隣接ポリゴンの法線を平均して頂点の法線を計算する。ポリゴン上の点  $P(x_p, y_p)$  の輝度  $I_P$  は次式で求められる。

$$I_P = \frac{x_R - x_P}{x_R - x_L} I_L + \frac{x_P - x_L}{x_R - x_L} I_R \quad (13.6)$$

$$I_L = \frac{y_s - y_B}{y_A - y_B} I_A + \frac{y_A - y_s}{y_A - y_B} I_B, \quad I_R = \frac{y_s - y_C}{y_D - y_C} I_D + \frac{y_D - y_s}{y_D - y_C} I_C \quad (13.7)$$

■図4.48——グローのスムーズシェーディングの計算方法



「コンピュータグラフィックス」2004年 / 財団法人画像情報教育振興協会（CG-ARTS協会）

図 13.7 グローシェーディング

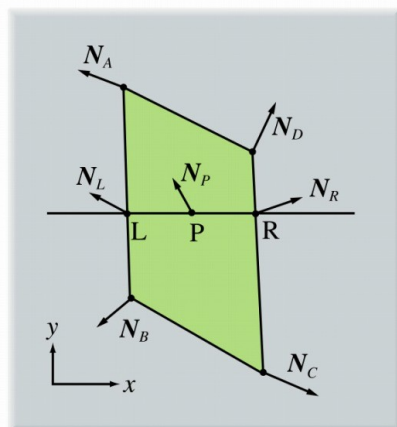
グローシェーディングは計算量が少ないが、輝度の補間を用いているのでハイライトの表現が困難であるとともに、ポリゴン境界での輝度値は連続であるが輝度値変化が不連続になるため視覚的に必ずしも滑らかに見えるとは限らない。



### 13.4.2 フォンシェーディング

フォンシェーディングは法線ベクトルの補間によってスムーズシェーディングを行う方法である。ベクトルの正規化による計算量が増大するが、ポリゴン内の輝度は求めた法線ベクトルを用いて計算するためハイライトの欠落がないシェーディングを行うことができる。図 13.9 にグローシェーディングとフォンシェーディングによる結果を示す。

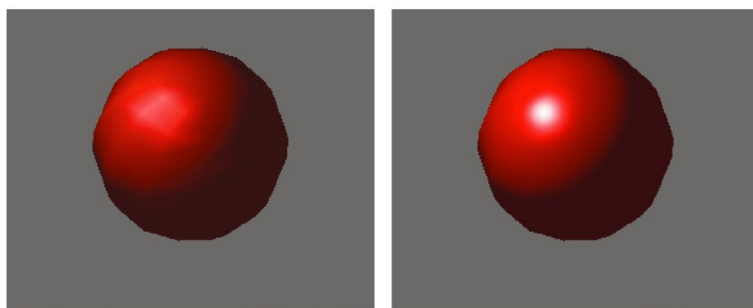
■図4.49——フォンのスムーズシェーディングの計算方法



「コンピュータグラフィックス」2004年 / 財団法人画像情報教育振興協会 (CG-ARTS協会)

図 13.8 フォンのスムーズシェーディング

■図4.47——グローとフォンのスムーズシェーディングの比較



[a] グローシェーディング (輝度を内挿する) [b] フォンシェーディング (法線ベクトルを内挿する)

「コンピュータグラフィックス」2004年 / 財団法人画像情報教育振興協会 (CG-ARTS協会)

図 13.9 グローとフォンのスムーズシェーディング

## 13.5 練習問題

問題 13.1 ambient, emissive, shininess, specular のパラメータ設定を調節して、金属、プラスチック、電球 (発光体) のような物体を作成せよ。(web にあるサンプルプログラム materialProperties.pde を参考にすると良い。)