

< 作図例 >

本書で作成したプログラムの作図例をいくつか紹介します。

最初の例はカタツムリ曲面です。右の絵は半分に分断したカタツムリ曲面です。球面法の良いところは、通常では計算が難しいような図形も簡単な2次方程式に直し、交点や交点における法線ベクトルが求められることです。したがって、影もうまく描けています。

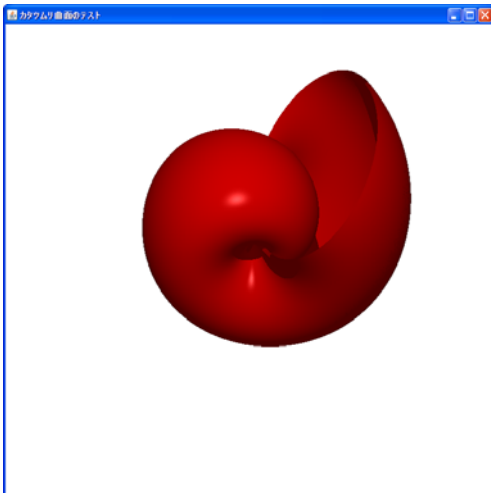


図1 カタツムリ曲面; 6.1 節(本文 105 ページ)参照

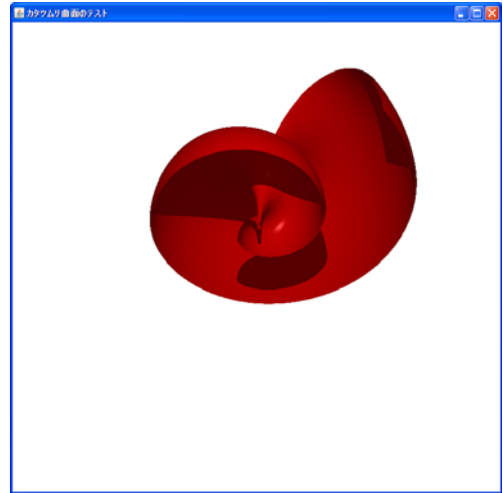


図2 切断カタツムリ曲面; 6.2 節(本文 105 ページ)参照

次の例はよく知られた球面をメッシュで描いたものです。このようによく知られた図形も表現を変えると面白いものになります。また、同じ手法で先に示したカタツムリ曲面を描いたものが右の絵です。

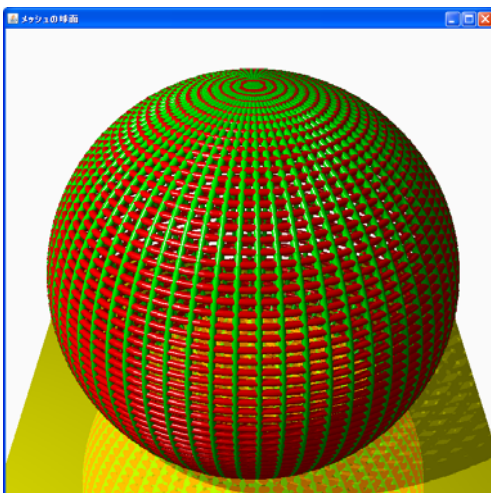


図3 メッシュの球面; 8.7 節(本文 172 ページ)参照

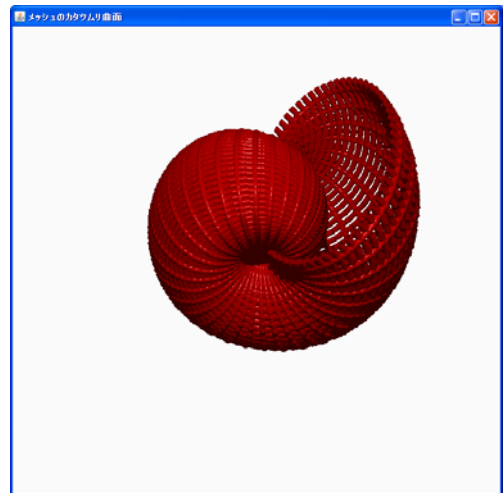


図4 メッシュのカタツムリ曲面; 8.10 節(本文 173 ページ)参照



図5 5.21節(本文94ページ)参照

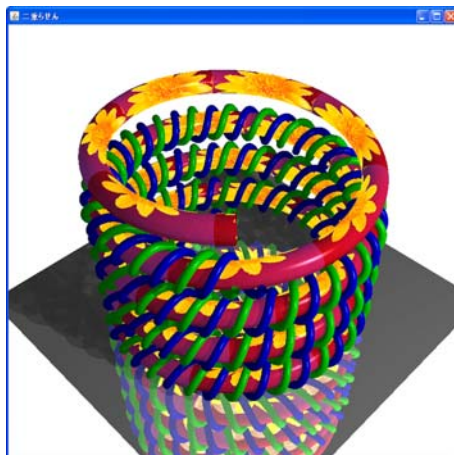


図6 5.23節(本文95ページ)参照



図7 7.26節(本文164ページ)参照

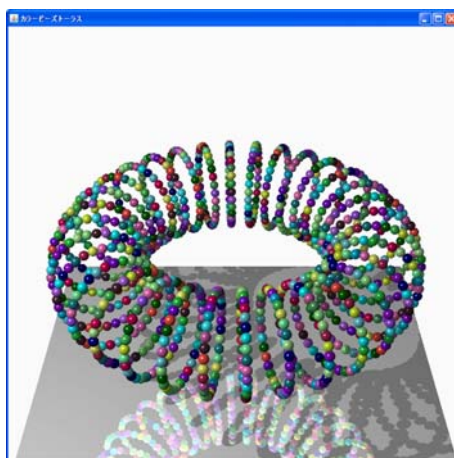


図8 8.13節(本文176ページ)参照

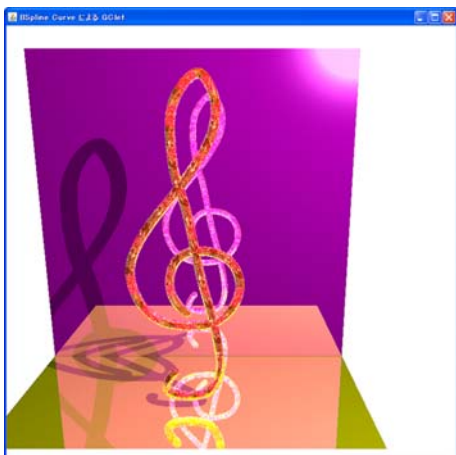


図9 9.4節(本文201ページ)参照

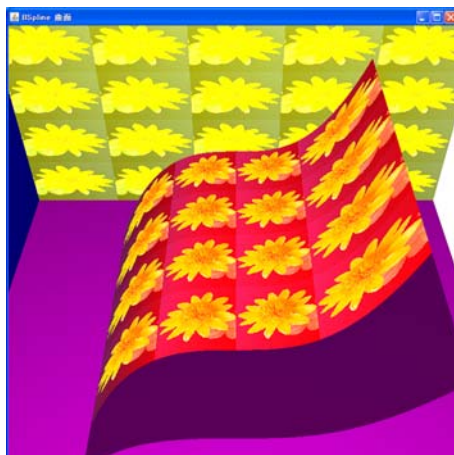


図10 10.2節(本文208ページ)参照

まえがき

本書の目的は表題の通り、「こんなに簡単におもしろい絵が描けるの。CG っておもしろい！」と感じてもらおうことです。ペイント系やドロー系のソフトウェアを使って絵を描くのも CG の 1 つの方法ですが、仕組みを理解するには自分でソフトウェアを作ってみるのが一番です。個人で大規模なソフトウェアを開発するのは無理ですが、仕組みを理解する程度でしたら、それほど難しくはありません。しかし、球面や四角形のような単純な図形しか描けないのでは興味が半減します。本書ではもっとさまざまな図形が描ける簡単な方法を提案します。

CG で図形を描く方法はさまざまありますが、本書ではレイトレーシングを使います。よく知られているように、レイトレーシングは簡単で、かつ優れたアルゴリズムです。たとえば、物体に陰影を付けたり、あるいは光の反射や透明な物体に関する透過・屈折の効果なども自然にとり込むことができます。

また、簡単に実装できるのですが、実はレイトレーシングには大きな弱点があります。レイトレーシングの原理を簡単に見てみると、目から出た視線（レイ）を考え、この視線と物体が交わったら、その交点を投影面（スクリーン）に投影し、絵を描きます。したがって、レイトレーシングの最も重要な部分はレイと物体の交差判定と、交わった場合その交点を具体的に計算する方法です。通常、図形は方程式で表されます。レイは直線ですからこれも方程式で表されます。したがって、交点は連立方程式の解として与えられます。図形が単純な場合、たとえば球面のような場合は連立方程式が 2 方程式になりますから、交差判定は 2 次方程式の判別式の正負で判定できます。判別式が正の場合、交点は 2 次方程式の解の公式を使うことで決定できます。しかし少し複雑な図形になると、連立方程式は一般に高次方程式（3 次や 4 次、あるいはもっと次数の高い方程式）で表され、2 次方程式のように簡単に解くことはできません。これはやっかいな問題です。この問題を解決するためにさまざまな手法が考案されてきました。本書では、球面法という新しい方法（正確には 1980 年に Whitted が考え出し、その後忘れられた方法）を使って、交差判定と交点計算を 2 次方程式の計算に直し、初心者でも複雑な図形を簡単に描けるようにしました。

レイトレーシングのもう 1 つの弱点はスクリーン上の各ピクセル 1 つ 1 つに対してレイを出し、交差判定と交点計算をする必要があるため、描画に時間がかかることです。この点についても昔からさまざまな研究がされてきました。しかしまだ満足のいく結果は得られていません。球面法においても同様ですが。最近のパソコンは高性能化しているため、本書で扱う図形くらいならば 20~30 秒くらいで描くことができます。これならば、それほどストレスを感じないでしょう。自分でたくさん絵を描いていると、レイトレーシングの改良につながるような何か新しいアイデアが得られるかもしれません。著者としては、ぜひそのようになることを願っています。

本書は大きく 4 つのパートからなっています。最初のパートは、球面と四角形の標準的なレイトレーシングの解説です。この 2 つの図形について、レイとの交差判定や交点計算、さらにシェーディング、テクスチャマッピング、反射および透過・屈折についての解説です。テクスチャマッピングの方法は図形ごとに異なりますが、それ以外のシェーディング、反射透過・屈折はどの図形でもほ

ぼ同じ（法線ベクトルのとり方は図形に依存します）です。第2のパートは球面法によるレイトレーシングの解説です。曲線や曲面を描きます。曲線そのものは光の作用に対して何の効果も与えませんから、CGでは太さを与え、チューブとして描かれます。第3のパートは球面法とFineStructure（微細構造）の解説です。球面法を使うとレイトレーシングで簡単に描ける図形が曲線や曲面まで広がりますが、さらにFineStructureを使うと、より一層面白い、楽しい図形を描くことができます。しかも、使うのは中学・高校で学ぶ初等幾何学や空間ベクトルの知識だけです。そして、最後のパートがB-Spline曲線とB-Spline曲面のレイトレーシングです。この2つの図形は球面法と相性が良くレイトレーシングの対象がさらに広がります。

もし、CGを最初から勉強するなら最初のパートから、CGについてある程度知っているので、球面法によるレイトレーシングを手取り早く知りたい人はパートIIから読まれるのがよいでしょう。パートIIIとパートIVはパートIIの知識を必要とするので、パートIIから読んでいただくことをお勧めします。

本書はJavaの解説書ではありません。したがって、Javaの文法に関する説明はほとんどありません。しかし、心配はいりません。本書ではJavaのAPI関数はほとんど使っていません。むしろ自分でAPI関数に相当するものを作るのが本書のもう1つの目的です。したがって、少し極端な言い方をすれば、for文とif文が書けて、かつ、CGに興味がある方なら簡単に読むことができます。ぜひ、チャレンジしてみてください。

目次

まえがき iii

第I部 標準的な図形のレイトレーシング

第1章 レイトレーシングの原理	3
1.1 作成するファイルの関係	3
1.2 レイトレーシングの原理	5
1.3 投影面の座標系	6
1.4.1 投影面に座標系を定める	6
1.4.2 座標軸ベクトルの取り方	8
1.4.3 投影面上に座標系を定めるプログラムコード例	9
1.4 基礎共通ファイル(レイトレーシングの枠組みと支援をするクラス)	9
1.4.1 枠組みとレイの走査を実行するクラス: MyJPanel	9
1.4.2 レイトレーシングのパラメータをセットするクラス: RayData	11
1.4.3 図形を格納するクラス: DisplayList	12
1.4.4 交点データを格納するクラス: KotenData	12
1.4.5 図形のひな形を与える抽象クラス: Shape	13
1.4.6 シェーディングを実行するクラス: Shade	14
1.4.7 光源の位置や強度をセットするクラス: Light	16
1.4.8 ベクトル計算を支援するクラス: Vec	18
第2章 球面と四角形のレイトレーシング	19
2.1 球面	19
2.1.1 球面とレイとの交差判定・交点計算	19
2.2 シェーディング(陰影付け)とシャドウイング	21
2.2.1 物体と光の反射	21
2.2.2 シェーディングモデル	22
2.2.3 シャドウイング	25
2.2.4 テストプログラムと実行結果	25
2.3 反射, 透過, 屈折	28
2.3.1 反射と透過・屈折のパラメータ	29
2.3.2 ピクセルの色	30
2.3.3 再帰的計算のプログラムコード	30
2.3.4 反射, 透過・屈折のテストプログラムと実行結果	32
2.4 球面へのテクスチャマッピング	33
2.4.1 基本のテクスチャマッピング	34

2.4.2	球面のテクスチャマッピング・テストプログラムと実行結果	36
2.4.3	テクスチャーの繰り返しマッピング	37
2.4.4	球面への繰り返しマッピング・テストプログラムと実行結果	38
2.5	テクスチャマッピング, 反射, および透過・屈折を組み入れたテストプログラムと実行例	39
2.6	長方形	41
2.6.1	長方形の定義	41
2.6.2	平面とレイとの交差判定・交点計算	43
2.6.3	角領域とレイとの交差判定	44
2.6.4	長方形とレイとの交差判定・交点計算	44
2.6.4	長方形へのマッピング	45
2.6.5	テクスチャマッピングのテストプログラム	47
第3章	その他の標準図形	49
3.1	空間内の円板	39
3.1.1	円柱のテストプログラムと実行結果	50
3.2	円柱	54
3.2.1	円柱とレイの交差判定・交点計算	54
3.2.2	円柱のテクスチャマッピング	56
3.3	円錐	57
3.3.1	円錐とレイの交差判定・交点計算	57
3.3.2	交点における法線ベクトル	59
3.3.3	円錐の定義クラス	60
3.3.4	円柱, 円錐のテストプログラムと実行例	62
3.4	複合図形	65
3.4.1	立方体	65
3.4.2	穴の開いた球面	67
3.4.3	穴あき球面のテストプログラムと実行結果	68
第II部	球面法による曲線と曲面のレイトレーシング	
第4章	曲線, 曲面用拡張共通ファイル	71
4.1	球面法とは?	73
4.2	拡張共通ファイル(球面法の基本データを作成するクラス)	73
4.2.1	曲線のひな形を与えるクラス: Curve.java	73
4.2.2	曲面のひな形を与えるクラス: Surface.java	74
4.2.3	チューブ用の球面データを作成するクラス: SphericalCurveData.java	74
4.2.4	曲面用の球面データを作成するクラス: SphericalSurfaceData.java	75
4.2.5	交差判定を統括するクラス: KyotsuFSHantei.java	75
第5章	曲線のレイトレーシング	71
5.1	曲線用球面データ	77
5.1.1	曲線用球面データの作り方	77

5.2 パラメトリック曲線のチューブ化と交点計算	80
5.2.1 球面とレイの交点	81
5.2.2 蓋をしたチューブの生成	81
5.2.3 チューブの生成	83
5.2.4 内部チェックのアルゴリズム	84
5.2.5 テストプログラムと実行結果	88
5.3 チューブのテクスチャマッピング	90
5.3.1 チューブとテクスチャマッピング	90
第6章 曲面のレイトレーシング	77
6.1 曲面用球面データ	101
6.1.1 定義域の分割	101
6.1.2 レベル n の小球面の作成	102
6.1.3 レベル $(n - 1)$ の小球面の作成	102
6.1.4 再帰的交差判定のメソッド	105
6.2 曲面とレイとの交差判定・交点計算	105
6.3 曲面の書式と実行例	106
6.3.1 曲面の書式	106
6.3.2 テストプログラムと実行方法	107
6.4 曲面のテクスチャマッピング	109
第III部 球面法と Fine Structure	
第7章 球面法と Fine Structure	119
7.1 Fine Structure	19
7.2 FineStructure としての立方体	121
7.2.1 小立方体を作成するクラス	121
7.2.2 moving frame、立方体の基準ベクトルと8個の頂点	121
7.2.3 立方体との交差判定・交点計算	123
7.2.4 立方体の面の塗り分け	126
7.2.5 立方体のバリエーション	127
7.2.6 テストプログラムと実行方法	131
7.3 その他の曲線用 FineStructure	132
7.3.1 円柱	132
7.3.2 円錐	134
7.3.3 球面	137
7.3.4 テストプログラムと実行結果	138
7.3.5 三角柱	139
7.3.6 テストプログラムと実行結果	145
7.3.7 円板	147
7.3.8 テストプログラムと実行結果	150
7.3.9 四角形	151

7.3.10 テストプログラムと実行結果	158
7.4 複合図形	159
7.4.1 クロス円板の三葉結び目	160
7.4.2 鉛筆の三葉結び目	161
第 8 章 曲面と FineStructure	173
8.1 曲面用 FineStructure	173
8.1.1 小立方体	173
8.1.2 円柱	178
8.1.3 円錐	181
8.1.4 球面	183
8.1.5 三角柱	185
8.1.6 円板および四角形	187
8.1.7 複合図形	192
第 IV 部 B-Spline 曲線 , B-Spline 曲面	
第 9 章 B-Spline 曲線	199
9.1 球面法と自由曲線、自由曲面	199
9.2 B-Spline 曲線	199
9.2.1 B-Spline 曲線とは	199
9.2.2 簡単な例	200
9.2.3 球面法への応用	202
9.3 球面データ作成	203
9.3.1 新しい曲線クラスとそれを扱うクラス	203
9.3.2 Knot ベクトルの作成	204
9.3.3 基底関数の作成	205
9.3.4 曲線上の座標を計算するメソッド	205
9.3.5 球面データの作成メソッド	206
9.4 ユーザープログラムの書き方	208
9.4.1 プログラム例と実行結果	209
第 10 章 B-Spline 曲面	213
10.1 B-Spline 曲面	213
10.2 球面データの作成	213
10.2.1 球面データの作成メソッド	214
10.3 プログラム例と実行結果	217
定型文索引	220
クラス索引	221
事項索引	224