

## コンピュータグラフィックスS

### 第4回 モデリング

システム創成情報工学科 尾下 真樹

2019年度 Q2

0

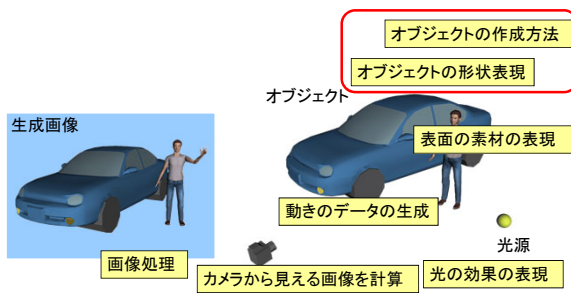
## 今日の内容

- 前回の復習
- モデリングの概要
- 3次元オブジェクトの表現方法
- 3次元オブジェクトの作成方法

1

## モデリング技術の位置づけ

- オブジェクトの形状表現・作成方法



2

## 教科書(参考書)

- 「コンピュータグラフィックス」  
CG-ARTS協会 編集・出版(3,200円)  
- 3章
- 「ビジュアル情報処理 -CG・画像処理入門-」  
CG-ARTS協会 編集・出版(2,500円)  
- 3章



3

## 参考書

- 「3DCGアニメーション」  
栗原恒弥 安生健一 著、技術評論社 出版  
- 18~51ページ
- 「3次元CGの基礎と応用」  
千葉則茂 土井章男 著、サイエンス社 出版  
- 1~12 ページ
- 「コンピュータグラフィックスの基礎知識」  
塩川厚 著、オーム社 出版  
- 60~67 ページ

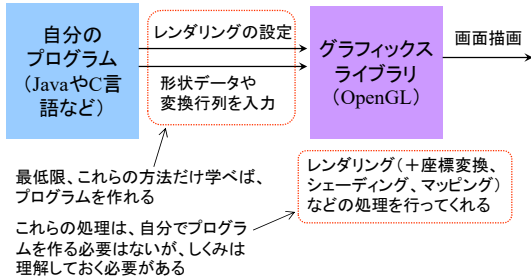
4

## 前回の復習

5

## グラフィックスライブラリの利用

### • 自分のプログラムとOpenGLの関係

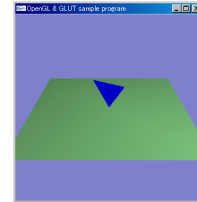


6

## サンプルプログラム

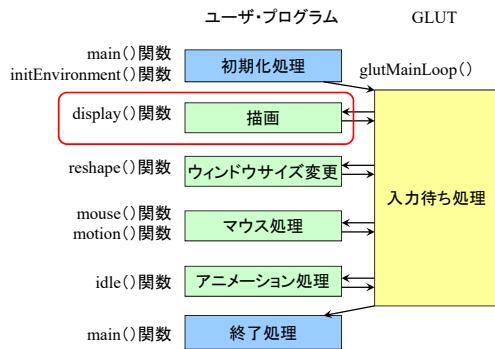
### • opengl\_sample.c

- 地面と1枚の青い三角形が表示される
- マウスの右ボタンドラッグで、視点を上下に回転



7

## サンプルプログラムの構成



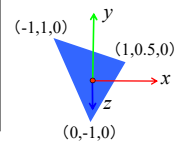
8

## ポリゴンの描画(今回の講義に関連)

### • 1枚の三角形を描画

- 各頂点の頂点座標、法線、色を指定して描画
- ポリゴンを基準とする座標系(モデル座標系)で頂点位置・法線を指定

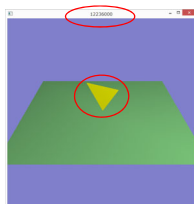
```
glBegin( GL_TRIANGLES );
glColor3f( 0.0, 0.0, 1.0 );
glNormal3f( 0.0, 0.0, 1.0 );
glVertex3f(-1.0, 1.0, 0.0 );
glVertex3f( 0.0,-1.0, 0.0 );
glVertex3f( 1.0, 0.5, 0.0 );
glEnd();
```



9

## 演習内容

1. コンパイル・実行できることを確認する
2. プログラムを修正して、以下の修正を行う
  - ウィンドウのタイトルに、自分の学生番号が表示されるようにする
  - 三角形の色を、青から黄に変更する
3. 修正が終わったら、Moodleからプログラムを提出



10

## 今回の講義と演習の内容

- 今回の講義で、さまざまなモデリングの手法を学習(プログラミング演習はなし)
- 次回以降の演習で、特に利用することが多いポリゴンモデルについて、C言語とOpenGLを使った、データ表現の方法や、描画の方法を学習する
  - 今回の講義のポリゴンモデルに関する内容は、特にきちんと理解しておく必要がある
  - 他の内容も、演習では扱わないが、基本知識としてきちんと押さえておく必要がある

11

## モデリングの概要

12

## モデリング

- モデリング (Modeling)
  - コンピュータ上で、物体の形のデータを扱うための技術
  - 形状の種類や用途によって、さまざまなデータ表現方法がある
    - ポリゴンモデルが代表的
    - 形状の種類や用途によっては、他の方法も必要になるため、きちんと理解しておく必要がある
  - 形状データの表現方法だけでなく、どのようにしてデータを作るかという、作成方法も重要になる

13

## モデリングの技術

- 3次元オブジェクトの表現方法
  - オブジェクトを計算機上でどのように表現するか
- 3次元オブジェクトの作成方法
  - 実際にオブジェクトのデータをどうやって作成するか
- 教科書によっては、両者が混在して説明されていることもあるが、きちんと区別して考える必要がある

14

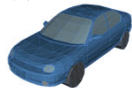
## 3次元モデルの表現方法

15

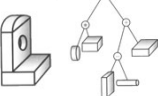
## 3次元モデルのデータ表現

- **サーフェスモデル**
  - ポリゴンモデル
  - 曲面パッチ
  - サブディビジョンサーフェス
- **ソリッドモデル**
  - 境界表現
  - CSGモデル
- **その他のモデル**
  - 特殊なデータに向けた表現

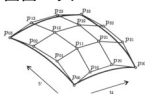
ポリゴンモデル



CSGモデル



曲面パッチ



16

## サーフェスモデル

- **オブジェクトの表面 (surface) 形状のモデル**
  - オブジェクトの「中身の情報」は持たない
  - コンピュータグラフィックスでは、表面形状のみが重要 (画像に影響しない内部情報は不要)
- **表面形状の表現方法**
  - ポリゴンモデル
  - 曲面パッチ
  - サブディビジョンサーフェス

17

## サーフェスモデルの表現方法

- ポリゴンモデル

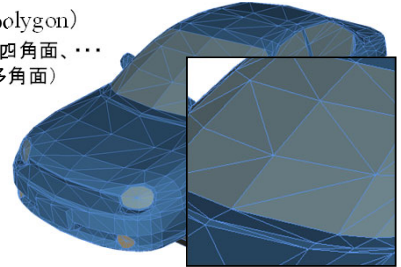
- 曲面パッチ
- サブディビジョンサーフェス



18

## ポリゴンモデル

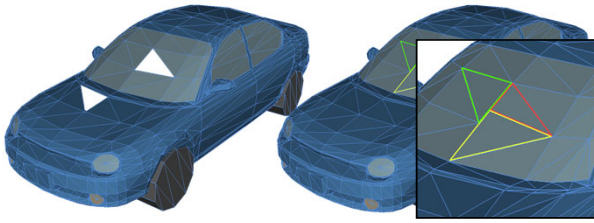
- ポリゴン(多角面)の集合によって表面形状(サーフェス)を表現
  - ポリゴン(polygon)
    - 三角面、四角面、...
    - n角面(多角面)



19

## サーフェスモデルの制約

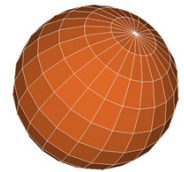
- 穴が空いてはいけいない
- 隣接するポリゴン同士は辺を共有している必要がある(必ずしも守られない場合もある)



20

## ポリゴンモデルの特徴

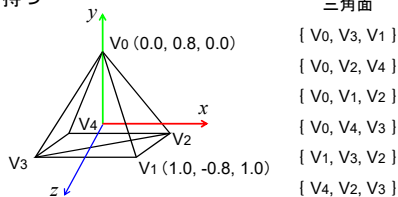
- メリット
  - 全てポリゴンだけなので処理が単純(高速)
  - 複雑な物体も表現できる
- デメリット
  - 曲面を表現することが難しい
  - データ量が大きくなる
- 用途
  - 現在主に使用されている技術
  - 特に、コンピュータゲームなど、高速に描画する必要のある用途に向いている



21

## ポリゴンモデルの表現例

- 四角すいの例
  - 5個の頂点と6枚の三角面(ポリゴン)によって表現できる
  - 各三角面は、どの頂点により構成されるかという情報を持つ



22

## ポリゴンモデルの表現例(続き)

- プログラムでの表現例(配列による表現)
  - 頂点座標の配列
  - ポリゴンを構成する頂点番号の配列

```
const int num_pyramid_vertices = 5; // 頂点数
const int num_pyramid_triangles = 6; // 三角面数

// 角すいの頂点座標の配列
float pyramid_vertices[ num_pyramid_vertices ][ 3 ] = {
    { 0.0, 1.0, 0.0 }, { 1.0, -0.8, 1.0 }, { 1.0, -0.8, -1.0 },
    { -1.0, -0.8, 1.0 }, { -1.0, -0.8, -1.0 }
};

// 三角面インデックス(各三角面を構成する頂点の頂点番号)の配列
int pyramid_tri_index[ num_pyramid_triangles ][ 3 ] = {
    { 0,3,1 }, { 0,2,4 }, { 0,1,2 }, { 0,4,3 }, { 1,3,2 }, { 4,2,3 }
};
```

23

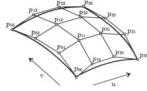
## サーフェスモデルの表現方法

- ポリゴンモデル
- **曲面パッチ**
- サブディビジョンサーフェス

ポリゴンモデル



曲面パッチ



サブディビジョンサーフェス



24

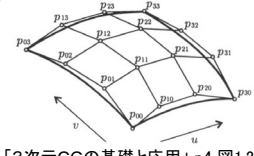
## 曲面パッチ

- **パラメトリック曲面の集合によりオブジェクトを表現**
  - 制御点と曲面式を用い、数式によって曲面を表現
  - 2つのパラメータ  $(u, v)$  で3次元空間の点  $(x, y, z)$  を定義
  - $(u, v)$  を  $(0, 0) \sim (1, 1)$  に変化させることで曲面になる
  - 曲面は  $n \times m$  個の格子状の制御点により定義される
  - 様々な種類の曲面式がある

$$S(u, v) = \sum_{i=0}^n \sum_{j=0}^m P_{ij} B_i^n(u) B_j^m(v)$$

$$B_k^n(t) = \frac{n!}{(n-k)!k!} t^k (1-t)^{n-k}$$

$P_{ij}$  は制御点、 $B(t)$  は重み関数



「3次元CGの基礎と応用」p4 図1.2

25

## いろいろな曲面式

- ベジェ曲面
- 有理ベジェ曲面
  - 頂点ごとに重み付けができる
  - 2次曲面(球など)も表現できる
- B-スプライン曲面
 

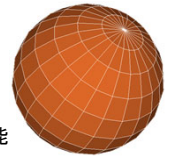
約40枚のベジェ曲面で構成  
「3次元CGの基礎と応用」p5 図1.4
- 非一様有理B-スプライン曲面(NURBS)
  - 1990年代前半くらいまでは、曲面を含むモデルを表現するためにNURBSがよく使われていた
  - 格子状の制御点でしか曲面を定義できないため、実際には複数の曲面パッチを組み合わせる必要がある



26

## 曲面パッチの利用

- **曲面パッチの描画方法**
  - 曲面パッチからポリゴンモデルに変換して描画することが多い
  - ポリゴンモデルの方が単純なので、データ量は増えても高速に描画可能
- **曲面パッチの利点**
  - 曲面を正確に表現できる
  - 特に曲面を持つ工業製品のモデリングなどでは重要
  - ポリゴンモデルと比較して、記録時のデータ量が少なくて済む



27

## 曲面パッチの特徴

- **メリット**
  - 少ないデータ量で曲面を持つオブジェクトを表現
- **デメリット**
  - 曲面パッチの制御点は格子状でなければならぬ
  - 曲面パッチ同士をうまくつなげることが難しい
- **特徴**
  - 工業製品・映画など精緻なモデルに使われている
  - 最近ではサブディビジョンサーフェスの方が使われつつある

28

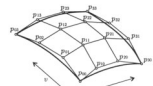
## サーフェスモデルの表現方法

- ポリゴンモデル
- 曲面パッチ
- **サブディビジョンサーフェス**

ポリゴンモデル



曲面パッチ



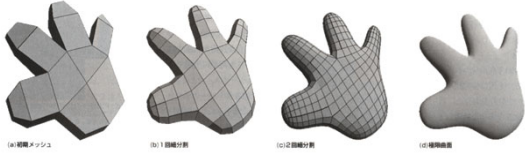
サブディビジョンサーフェス



29

## サブディビジョンサーフェス

- 細分割曲面 (sub-division surface)
  - ポリゴンモデルのポリゴンを繰り返し分割していくことで滑らかなポリゴンモデルを生成する技術



30

## 何のため?

- 曲面パッチは接続性に問題がある
  - パラメトリック曲面は格子状の制御点のみ
  - 複雑なモデルを作成するためには、複数の曲面パッチをつなぐ必要がある
  - 接続部分はなめらかでなくなる
  - 変形を加えると接続部分が途切れたりなめらかでなくなったりする
- サブディビジョンサーフェスではポリゴンモデル全体から一枚の曲面を生成するため、このような問題は生じない

31

## Catmull-Clark 細分割曲面

- Catmull-Clark 細分割曲面 [1978]
  - 任意の多角形で構成されるポリゴンモデルに適用可能
  - n角形をn個の四角形に分割 → 全体がなめらかになるように各頂点を移動

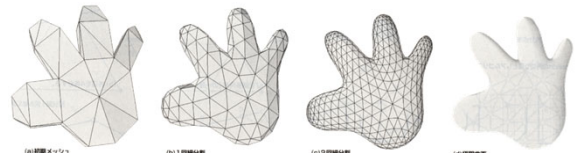


日経CG 2000年10月号「分かるサブディビジョン・サーフェス入門」p160

32

## Loop 細分割曲面

- Loop 細分割曲面 [1987]
  - 三角形のみで構成されるポリゴンモデルにのみ適用可能
  - 1つの三角形を4つに分割 → 全体がなめらかになるように各頂点を移動



日経CG 2000年10月号「分かるサブディビジョン・サーフェス入門」p161

33

## サブディビジョンサーフェスの特徴

- 曲面パッチの接続性の問題を解決
- モデリングが容易
  - おおまかなポリゴンモデルから自動的になめらかな曲面が生成される
  - 頂点や辺に重みを設定することで、各部のなめらかさも制御可能
- 少ないデータ量でオブジェクトを表現できる
  - Web3D(WWWを通じて3次元オブジェクトを表示・操作するような応用)などにも適している



Geri's Game (Pixar)  
[DeRose SIGGRAPH '98]

34

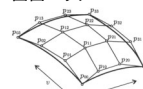
## サーフェスモデルのまとめ

- ポリゴンモデル
- 曲面パッチ
- サブディビジョンサーフェス

ポリゴンモデル



曲面パッチ



サブディビジョンサーフェス



35

### 3次元モデルのデータ表現

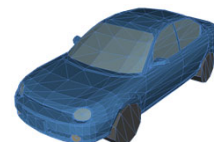
- サーフェスモデル
  - ポリゴンモデル
  - 曲面パッチ
  - サブディビジョンサーフェス
- **ソリッドモデル**
  - 境界表現
  - CSGモデル
- その他のモデル
  - 特殊なデータに向けた表現



36

### ソリッドモデル

- オブジェクトの固体 (solid) 形状のモデル
  - 「中身の情報」を持つモデル
  - オブジェクト同士の集合演算、正確な交差判定、物理計算などが可能になる
- ソリッドモデルの表現方法にいくつかある
  - 境界表現
  - CSGモデル



37

### ソリッドモデルの表現方法

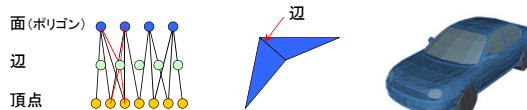
- **境界表現**
- CSG (Constructive Solid Geometry)



38

### 境界表現

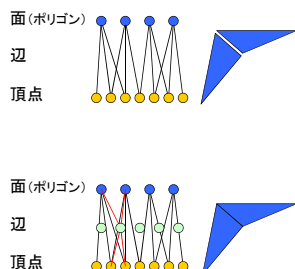
- 境界表現 (boundary representation)
- ポリゴンモデルに、隣接する面同士のつながりの情報を追加したもの
  - ポリゴンの集合+線分・頂点の情報(それぞれの線分・頂点にどのポリゴンが接続しているか)
  - 単にばらばらのポリゴンではなく、つながった表面とすることで、「中身」を表すことができる



39

### ポリゴンモデルと境界表現

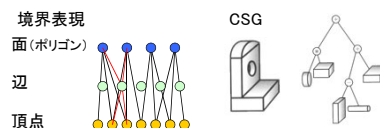
- ポリゴンモデル
  - 各ポリゴンが独立
- 境界表現
  - 隣接するポリゴンの情報を格納



40

### ソリッドモデルの表現方法

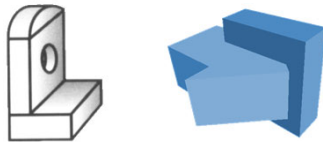
- 境界表現
- **CSG (Constructive Solid Geometry)**



41

## CSG

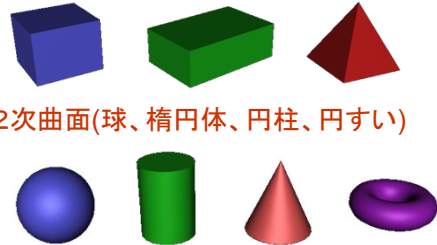
- **CSG (Constructive Solid Geometry)**  
 - プリミティブ(基本形状)の組み合わせにより複雑なオブジェクトを表現する方法



42

## プリミティブ

- 立方体、直方体、角すい
- 2次曲面(球、楕円体、円柱、円すい)



43

## 2次曲面

- 2次元の方程式で表現できるさまざまな形状



$$\sqrt{(x-x_0)^2 + (y-y_0)^2 + (z-z_0)^2} = r$$

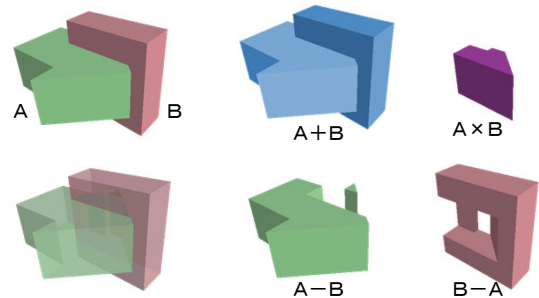


$$\sqrt{(x-x_0)^2 + (y-y_0)^2} = r, z_0 < z < z_1$$

44

## CSGの集合演算の例

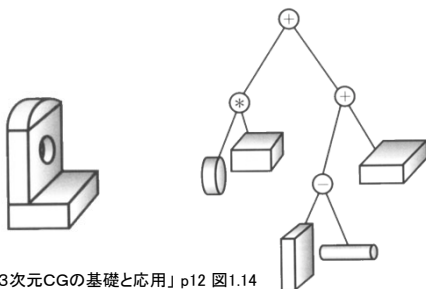
- 集合演算(ブーリアン演算)



45

## CSG表現の例

- 集合演算のツリー構造によって物体を表現



「3次元CGの基礎と応用」p12 図1.14

46

## CSGの特徴

- **メリット**  
 - 簡単なオブジェクトであればデータ量が少なく済む(計算量も少なくなる)
- **デメリット**  
 - 複雑なオブジェクトを表現することは難しい
- **用途**  
 - 比較的単純な工業製品のデザインなどには適している  
 - 現在は、実用レベルではあまり使われていない

47



### 3次元モデルのデータ表現

- サーフェスモデル
  - ポリゴンモデル
  - 曲面パッチ
  - サブディビジョンサーフェス
- ソリッドモデル
  - 境界表現
  - CSGモデル
- その他のモデル
  - 特殊なデータに向けた表現

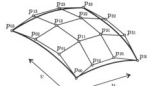
ポリゴンモデル



CSGモデル



曲面パッチ



48

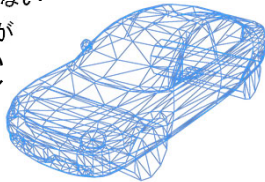
### その他の表現

- ワイヤースケルトン
- 高さによる表現
- ボクセル(ヴォリュームデータ)
- パーティクル
- メタボール

49

### ワイヤースケルトンモデル

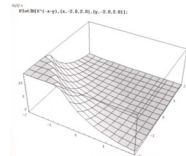
- 稜線だけのモデル
  - 頂点とそのつながり(辺)の情報を持つ
- 図形を線だけで描画すれば良い場合に有効
  - 現在はほとんど使われない
  - 昔はディスプレイの能力が低く、線画しか描けない状況もあったので、ワイヤースケルトンモデルもそれなりに有効だった



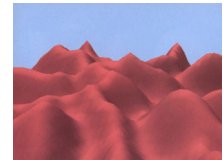
50

### 高さによる表現

- グリッド上の各座標の高さのデータを持つ
  - 地図などのデータに適している
    - 高さに限らず、平均雨量や関数値など2次元平面上に分布している値であれば何でも良い
    - 立体交差などはそのままでは表せない



「3次元CGの基礎と応用」p7 図1.6

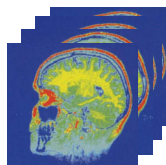
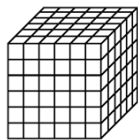


p6 図1.5

51

### ボクセル(ヴォリュームデータ)

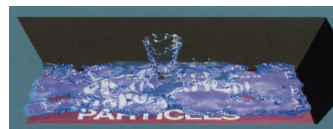
- ボクセルの色・濃度などによって立体を表す
  - 2次元格子データ(画像)の各要素はピクセル
  - 3次元格子データ(立体)の各要素はボクセル
- 立体の中身の情報が重要とされるような用途で利用
  - CTスキャン、気体・流体の解析データなど
  - ボクセルデータに特化した描画方法がある(詳しくは後日の講義で紹介)



52

### パーティクル

- 細かい粒子の集まりによって物体を表す
  - 炎や気体や流体などの不定形な物体
  - 衣服や髪の毛などの柔らかい物体
    - 対象の種類に応じて粒子の色・形、動きのモデルを設計



「3次元CGの基礎と応用」p117 図12.10

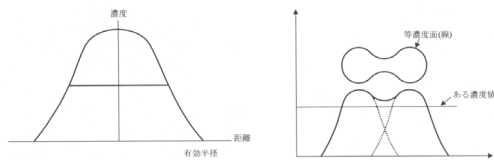


p120 図12.14

53

### メタボール

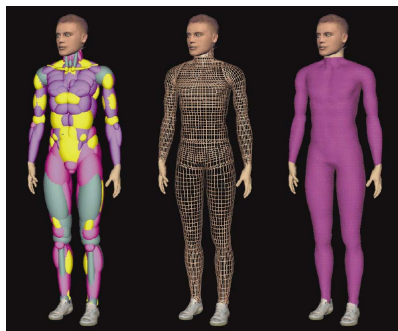
- 濃度を持った粒子の集まりによって表現
- 全ての粒子の濃度を合計
- ある濃度値の面をサーフェスとして描画



「3次元CGの基礎と応用」p11 図1.11, 図1.12

54

### メタボールの応用例



P.Kalra et al., IEEE Computer Graphics and Applications, Sept 1998

55

### 3次元モデルのデータ表現のまとめ

- **サーフェスモデル**
  - ポリゴンモデル
  - 曲面パッチ
  - サブディビジョンサーフェス
- **ソリッドモデル**
  - 境界表現
  - CSGモデル
- **その他のモデル**
  - 特殊なデータに向けた表現

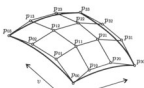
ポリゴンモデル



CSGモデル



曲面パッチ



56

### 各用途ごとの主なモデリング方法

- 映画・工業製品設計などに使われる精緻なモデル
  - サブディビジョンサーフェス、ポリゴンモデル、曲面パッチ (NURBUS)
- コンピュータゲーム向けなどの比較的粗いモデル
  - ポリゴンモデル、サブディビジョンサーフェス
- 特殊なオブジェクトの表現
  - パーティクルやボクセル(炎・煙・水)
  - 高さによる表現(地面)
  - ボクセル(物体のスキャン結果or解析データ)
- 現在はあまり使われていないモデル
  - ワイヤーフレーム、CSG、メタボール、曲面パッチ

57

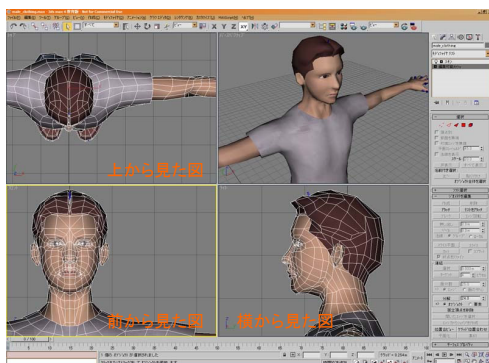
### 3次元モデルの作成方法

- 市販のソフトウェアを利用する方法が一般的
  - 基本的にはポリゴンを使ったモデリング
    - 頂点や面のデータを直接修正
    - おおまかな形状を簡単に作るための機能もある
    - おおまかなポリゴンモデルから、自動的になめらかな曲面パッチや再分割曲面を生成する機能もある
  - プロの制作者でも、多くの時間や手間がかかる
- 実物の形状をセンサーで取得する方法もある(詳しくは後述)

58

59

### 直接モデリング



60

### 直接モデリング

- 3次元空間に点や面を直接入力していく
  - 2次元の画面を使って3次元空間の点を入力するために、3面図が主に使われる
  - 頂点を追加していき、それらをつなぐことで面を作成していく
- モデルの修正のための操作
  - 頂点・面の移動
  - 頂点・面の削除
  - 辺・面の分割

61

### 3次元マウス

- 3次元座標を直接入力できるデバイス
  - 特殊・高価なのであまり使われていない
    - 一部のデザイン専門分野などで利用



3次元マウス Magellan, Logicad3D 3次元触覚デバイス PHANTOM, SensAble

62

### 一般的なモデリングの手順

1. おおまかな形状を作成
  - 作成するオブジェクトの種類に応じて、なるべく簡単に全体的な形状を作れる方法を使用
2. 細部の形状を調整
  - おおまかな形状を少しずつ分割・変形していく
  - 最終的には頂点単位で細かく直接修正する(直接モデリング)
  - 最初から頂点単位で直接モデリングしていくこともある

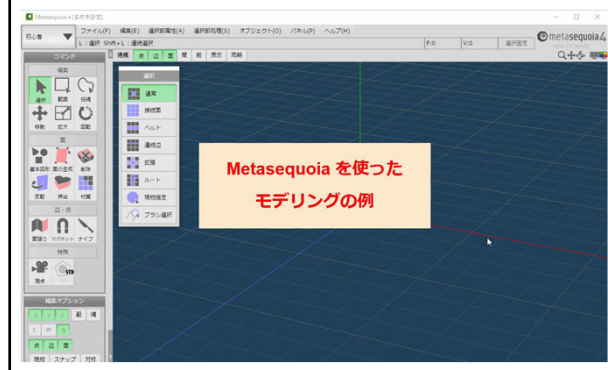
63

### モデリングのためのソフトウェア

- 統合システム(モデリング+アニメーション)
  - Softimage
  - Maya
  - 3ds Max
  - Lightwave 3D
  - Shade
- モデリング専用のソフトウェア
  - Metasequoia
- CAD(主に工業製品の設計用のソフトウェア)
  - AutoCAD等
- 体験版や解説書もあるので、興味がある人は、自分で使ってみると良い

64

### モデリングの例



65

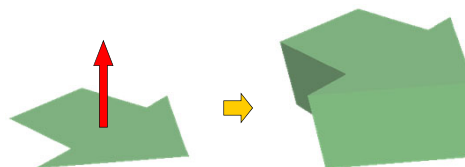
### おおまかな形状を作るための手法

- 平行移動体・回転体
- 集合演算
- 粘土のような変形操作
- センサーによるモデルの取得
- 手続き計算によるモデリング
- 2次元の絵から立体を計算

66

### 平行移動体

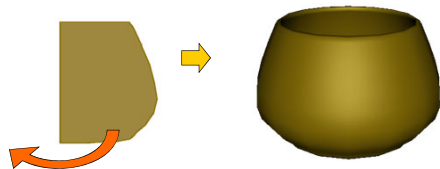
- 平面のオブジェクトを平行移動させてオブジェクトを作る



67

### 回転体

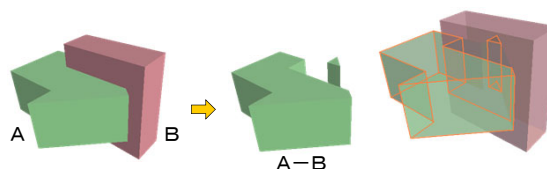
- 稜線のデータを作成し、それを回転させることでオブジェクトを作る
  - 線対称のオブジェクト(柱、壺、カップなど)を作成するのに適している



68

### 集合演算

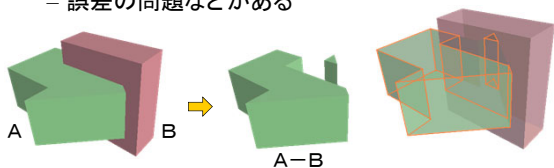
- 形状同士の積・和・差によって、新しい形状を得る処理
  - 形状を、3次元空間中の領域とみなして、領域同士の集合演算を行う



69

### 集合演算の利用

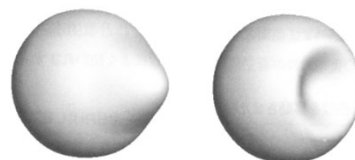
- CSGモデルで主に使われる
  - CSGモデルの場合は、特別な処理はいらない
- サーフェスモデルにも適用可能
  - 境界表現に変換し、数値的に計算する必要がある
  - 誤差の問題などがある



70

### 粘土のような操作

- 粘土のように変形することで形を修正



「コンピュータグラフィックスの基礎知識」p65 図2-14

71

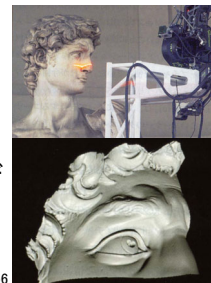
### おおまかな形状を作るための手法

- 平行移動体・回転体
- 集合演算
- 粘土のような変形操作
- **センサーによるモデルの取得**
- 手続き計算によるモデリング
- 2次元の絵から立体を計算

72

### センサーによるモデルの取得

- センサーを使ってカメラから物体までの距離を計測
  - 奥行き情報が撮れる  
赤外線センサを使用
  - 細かい点の集まりがデータとして得られる
  - 点群データをポリゴンモデルに変換

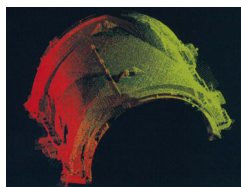
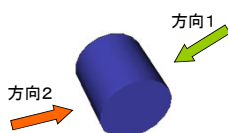


日経CG 2000年 9月号 p156

73

### 点群データの問題点

- カメラから見える範囲しかスキャンできない
  - 複数の方向からスキャンしたデータを合成する必要がある
    - 物体が小さい場合は回転台に載せて周囲から撮影



日経CG 2000年 9月号 p158

74

### センサーによるモデリングの特徴

- **メリット**
  - 実物に則したモデルが得られる
  - うまいければ自動化できる(時間がかからない)
- **デメリット**
  - 効率的なポリゴンモデルに変換するのが難しい
  - 複数の視点のデータを統合するのが難しい
  - 作成されたモデルを修正するのが難しい
- **主な用途**
  - 歴史的遺物などのなるべく正確な記録

75

### おおまかな形状を作るための手法

- 平行移動体・回転体
- 集合演算
- 粘土のような変形操作
- センサーによるモデルの取得
- **手続き計算によるモデリング**
- 2次元の絵から立体を計算

76

### 手続き計算によるモデリング

- **木の生長シミュレーション**
  - 枝の生え方、日光や風量などのパラメタを与えると自動的に樹木を生成
    - 例えば、枝が?cm進むごとに、?股に枝分かれする、などの成長モデルを設定し、モデルに従ってランダムに木を生成
- **地形・都市の自動生成**



教科書 p109 図12.1

77

### おおまかな形状を作るための手法

- 平行移動体・回転体
- 集合演算
- 粘土のような変形操作
- センサーによるモデルの取得
- 手続き計算によるモデリング
- 2次元の絵から立体を計算

78

### 2次元の絵から立体を計算

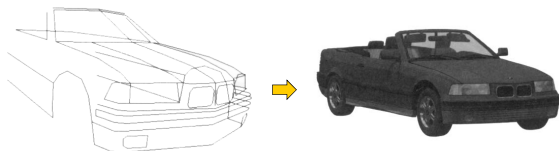
- 手軽に大まかな形を入力するためのインターフェース
  - 六角大王
  - Teddy

79

### 六角大王

- 株式会社 終作 (古島終作 氏)
  - オブジェクトが左右対称であることを利用して、斜めから見た図を描くと自動的に立体を生成

<http://www.shusaku.co.jp/www/>



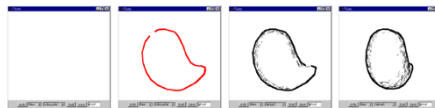
六角大王 Super オフィシャルトレーニングブック

80

### Teddy

- 東京大学 五十嵐先生
  - ペンで2次元の図を描くとオブジェクトを生成
  - 2次元の図に適当に厚みをつけることで立体化
  - 切断、枝の追加・削除などのインターフェースを提供
    - マジカルスケッチ (モデリングソフト)、らくがき王国 (ゲーム) などで商品化

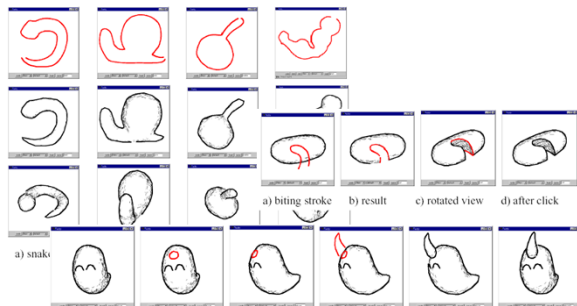
<http://www-ui.is.s.u-tokyo.ac.jp/~takeo/research/teddy/teddy-j.htm>



[Igarashi, Teddy, SIGGRAPH 98]

81

### Teddy



[Igarashi, Teddy, SIGGRAPH 98]

82

### 今日のまとめ

- オブジェクトの表現方法
  - ソリッドモデル (プリミティブと組み合わせ)
  - サーフェスモデル (ポリゴン、曲面パッチ)
  - その他 (高さ、ボクセル、パーティクル)
- オブジェクトの作成方法
  - 直接モデリング
  - 平行移動体、回転体、集合演算
  - スキャナーによる点群データの取得
  - 直感的なモデリングするための方法など

83

## データ表現方法のまとめ

- **サーフェスモデル**
  - ポリゴンモデル
  - 曲面パッチ
  - サブディビジョンサーフェス
- **ソリッドモデル**
  - 境界表現
  - CSGモデル
- **その他のモデル**
  - 特殊なデータに向けた表現

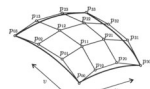
ポリゴンモデル



CSGモデル



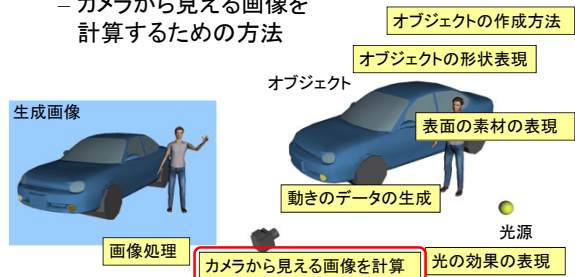
曲面パッチ



84

## 次回予告

- **レンダリング**
  - カメラから見える画像を計算するための方法



85