

# ここにも数学！

## コンピュータグラフィックス

第16回 JST数学キャラバン 「拡がりゆく数学 in 水戸 2016」  
廣瀬三平 (芝浦工業大学)

# コンピュータグラフィックスとは

- ・ コンピュータを用いて作成される画像 (Wikipedia)
  - ・ 2次元：図や絵を書く
  - ・ 3次元：3次元空間内の仮想的な立体物を作り出す
- ・ 中国語で書くと“计算机图形学”

# 応用：エンターテインメント

- ・ 映画、アニメーション

- ・ ジュラシックワールド © 2015 ILM/Universal Pictures and Amblin Entertainment

- ・ ポケットモンスター © Nintendo ・ Creatures ・ GAME FREAK ・ TV Tokyo ・ ShoPro ・ JR Kikaku © Pokemon © 2016 ピカチュウプロジェクト

- ・ シドニアの騎士 © TSUTOMU NIHEI ・ KODANSHA / KOS Production Committee

# 応用：リアルタイムCG

- ・ ゲーム

- ・ Final Fantasy Real-time Tech Demo © 2013 SQUARE ENIX CO., LTD.

- ・ Web

- ・ WebGL Skin Rendering Demo © AlteredQualia

# 応用：工業製品のデザイン

- ・ Computer Aided Design

- ・ デザイン、応力解析 © 2015 Dassault Systèmes SolidWorks Corp.

# 応用：シミュレーション

- ・ 雪のシミュレーション

Stomakhin, Schroeder, Chai, Teran, Selle / ACM SIGGRAPH 2013

- ・ 流体のシミュレーション

# 関連分野：画像処理

Kwatra, Schodl, Essa, Turk, Bobick / ACM SIGGRAPH 2003

Perez, Gangnet, Blake / ACM TOG 2003

# 関連分野：その他いろいろ

- ・ 3Dプリンタ

- ・ 3次元の物体の印刷技術

- ・ 建築・医療・教育・航空宇宙などで使われ始めている

- ・ コンピュータビジョン

Berthouzoz, Garg, Kaufman, Grinspun, Agrawala /  
ACM SIGGRAPH 2013

- ・ コンピュータに”目”を持たせる技術

- ・ 画像からの情報を元に、物体を認識

- ・ 顔の認識、物体の認識、写真の物体の3次元化

- ・ 製品の製作

- ・ 服、紙飛行機、凧、ぬいぐるみ

Umetani, Koyama, Schdmit, Igarashi /  
ACM SIGGRAPH 2014



# CGの国際会議 SIGGRAPH

- ・ 毎年夏に開催されるCGの国際的な会議、展示会
- ・ Disney、Pixar、DreamWorksなどが参加
- ・ 様々なイベントが開催
  - ・ テクニカルペーパー
  - ・ アニメーションフェスティバル

# CGと数学

- ・ CGは数学により出来ているといっても過言でない
- ・ CGを動機とした数学もある
- ・ 以下ではこれを“鑑賞”していく

# 3次元の動きの記述と四元数 (1/3)

- ・ 複素数  $a + bi$   $i^2 = -1$ 
  - ・ 2つの実数を並べたもの
  - ・ 普通の数と同様に足し算や掛け算が出来る
  - ・ 2次元 (平面) の動きを記述
    - ・ 複素数  $a + ib$  を足すことは平行移動
    - ・ 実数  $r$  をかけることは拡大、縮小
    - ・ 複素数  $\cos \theta + i \sin \theta$  をかけることは回転

# 3次元の動きの記述と四元数 (2/3)

- ・ 四元数  $a + bi + cj + dk$   $i^2 = j^2 = k^2 = ijk = -1$ 
  - ・ 4つの実数を並べたものであり、複素数と同様に足し算や掛け算などが出来る
  - ・ イギリスの数学者ハミルトンにより複素数の拡張として考案 (1843)
  - ・ 四元数  $bi + cj + dk$  を3次元ベクトル  $(b, c, d)$  と考えると、四元数の足し算などの演算がベクトルの動きに対応
    - ・ 四元数を足すことは平行移動
    - ・ 実数をかけることは拡大、縮小
    - ・ 四元数をかけることは (ある軸に沿った) 回転

# 3次元の動きの記述と四元数 (3/3)

- ・ 3次元の動き
  - ・ 簡明に記述するのは大変 (行列を用いる方法もある)
  - ・ 四元数を用いて記述することにより、3次元の物体の動きが非常に簡単になる
    - ・ ベクトルを用いた回転  $\cos \theta p + (1 - \cos \theta) \langle n, p \rangle n + \sin \theta n \times p$
    - ・ 四元数を用いた回転  $\bar{\lambda} p \lambda$
  - ・ さらに物体の変形、例えば骨とその周りの皮膚の動きに利用された (2007)

# 表情の編集とベクトル (1/2)

- ・ ベクトル

- ・ 幾つかの数字が集まったもの
- ・ 2次元ベクトルは2つの数、3次元ベクトルは3つの数
- ・ 物体の位置を表している
- ・ 足し算や内積、長さを求めることなどが行うことができた

- ・ 足し算  $(x_1, x_2) + (y_1, y_2) = (x_1 + y_1, x_2 + y_2)$

- ・ 内積  $(x_1, x_2) \cdot (y_1, y_2) = x_1y_1 + x_2y_2$

- ・ 長さ  $|(x_1, x_2)| = \sqrt{x_1^2 + x_2^2}$

# 表情の編集とベクトル (2/2)

- ・ 表情の編集
  - ・ 表情は人間が特に注目する部分であり、映画の製作において重要
  - ・ 感情が混ざった微妙な表情は鼻や目、口の位置を調整することによって作成だが非常に大変
  - ・ そこで顔の表情を大きなベクトルで表す
    - ・ 表情を表すベクトル (nose, left eye, right eye, ..., mouth)
  - ・ 笑顔、怒り、悲しみなど代表的な顔のベクトルを用意し、足すことにより微妙な表情を出す

# 流体と微分方程式 (1/2)

- ・ 微分方程式
  - ・ 微分は微小量の変化を記述するもの
  - ・ この微分を含むような方程式を微分方程式と呼ぶ
  - ・ 自然現象を含む様々な現象を記述

- ・ 波 
$$\frac{\partial^2 u}{\partial t^2} = \frac{\partial^2 u}{\partial x^2}$$

- ・ 熱 
$$\frac{\partial u}{\partial t} = \frac{\partial^2 u}{\partial x^2}$$

- ・ ソリトン (非線形な波) 
$$\frac{\partial u}{\partial t} + 6u \frac{\partial u}{\partial x} + \frac{\partial^3 u}{\partial x^3} = 0$$

- ・ 量子力学 
$$\sqrt{-1}\hbar \frac{\partial u}{\partial t} = -\frac{\hbar^2}{2m} \frac{\partial^2 u}{\partial x^2} + V(x)u$$

- ・ 渦糸 
$$\frac{\partial \gamma}{\partial t} = \frac{\partial \gamma}{\partial s} \times \frac{\partial^2 \gamma}{\partial s^2}$$



# 流体と微分方程式 (2/2)

- ・ 流体の方程式 
$$\frac{\partial \mathbf{v}}{\partial t} + (\mathbf{v} \cdot \nabla) \mathbf{v} = -\nabla p + \nu \Delta \mathbf{v} + \mathbf{f}(\mathbf{x}, t)$$
- ・ Navier-Stokes方程式と呼ばれる流体の動きを表す方程式
- ・ CGでは1990年代半ばから考察
- ・ 応用すると水の流れや雲、火、煙の動きを表すことが可能

# リアルな画像の作成と積分方程式 (1/2)

- ・ 積分方程式

- ・ 積分は（条件を満たすものの）総和を表すもの

- ・ この積分を含む方程式を積分方程式と呼ぶ

- ・ 信号処理  $g(x) = \int_a^b K(x, t) f(t) dt$

- ・ 粘弾性物質  $g(x) = \int_a^x K(x, t) f(t) dt$

- ・ レンダリング方程式

$$L_o(\mathbf{x}, \omega_o, \lambda, t) = L_e(\mathbf{x}, \omega_o, \lambda, t) + \int_{\Omega} f_r(\mathbf{x}, \omega_i, \omega_o, \lambda, t) L_i(\mathbf{x}, \omega_i, \lambda, t) (\omega_i \cdot \mathbf{n}) d\omega_i$$

# リアルな画像の作成と積分方程式 (2/2)

- ・ リアルな画像の作成
  - ・ 目に見える景色は入る光の総和（積分）によって求まる
  - ・ これを式で表したものが先のレンダリング方程式

$$L_o(\mathbf{x}, \omega_o, \lambda, t) = L_e(\mathbf{x}, \omega_o, \lambda, t) + \int_{\Omega} f_r(\mathbf{x}, \omega_i, \omega_o, \lambda, t) L_i(\mathbf{x}, \omega_i, \lambda, t) (\omega_i \cdot \mathbf{n}) d\omega_i$$

- ・ これを用いると画像のリアルさが飛躍的に向上 (1986)

# まとめとコメント

- ・ CGはエンターテインメントだけでなく、シミュレーションや可視化と関わっており、重要性はますます増大していくと考えられる
- ・ CGでは古典的から最新まで様々な数学が用いられている
- ・ 素朴なアイデアでも非常に強力な手法になることもある
  - ・ 今回の説明した数学も基本的な部分は100年以上前のものばかり

# 参考

- ・ 藤堂英樹、明治大学工学部におけるコンピュータグラフィックスの授業資料