



OLM Digital Inc.



人間やキャラクターの 表情アニメーション手法

安生健一

オー・エル・エム・デジタル

第10回「計算機を用いた数学研究セミナー」GCOE セミナー

コンピュータグラフィックスと数学の接点

京都大学理学部3号館 110講義室



Motivation

- Computer generated face/character
 - A key in visual storytelling
 - The most challenging object
- Our focus: facial animation
 - No perfect method to make it efficiently
 - Addresses many theoretical/practical problems

An overview

顔のモデリング、レンダリング、 アニメーション

Objects and issues

- 顔の幾何モデル、質感 Face geometry/ texture
 - Eyes, mouth, ears, nose, wrinkles...
 - Hair, teeth, tongue,...
- レンダリング
 - Photorealistic skin - subsurface scattering
 - Real-time rendering
- アニメーション
 - 効率の良い制作プロセスの構築のために
 - 表情, 唇の動きと音声の同期 (lip-synch)...



Face geometry

- “Sculpting” face (head): 彫刻を作るように

R. Parent: “Computer Animation (2nd ed.), Morgan Kaufman 2008.

- Subdivision surfaces: 再分割曲面
- Laplacian editing: ラプラシアンを用いた平滑化



Face geometry using digitizer

- Point-marking on the actual model (“mock”)
- Measuring the points by digitizer

“Tin Toy” by Pixar 1988

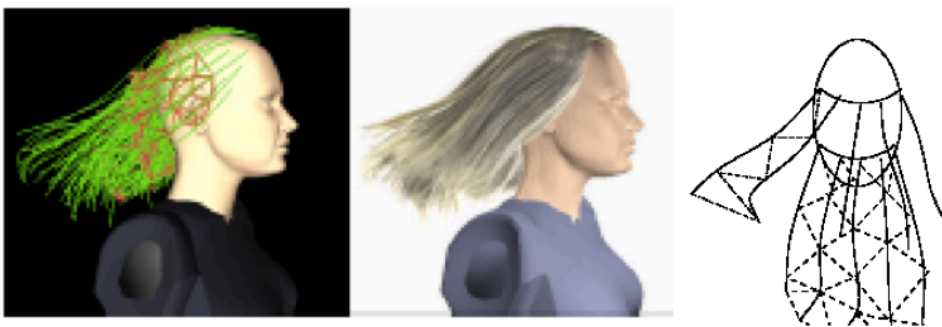
Modeling/ animating hair

- Typical case: Hair strand as a set of linear segments
- (Option) Partially modeled with texture



Yizhou Yu et al. In *Proc. Symposium on Computer Animation 2002*

Modeling/ animating hair



Yizhou Yu et al. In *Proc. Symposium on Computer Animation 2002*

- Placing “sparse” hair-guide models
- Calculating using the guide models
 - Dynamics
 - Hair-to-hair interaction
 - Interpolating them for finer hair animation and rendering



Rendering: BRDF vs. BSSRDF

- f_r : BSSRDF (Bidirectional Scattering Surface Reflectance Distribution Function)

$$L_o(x, \omega) = L_e(x, \omega) + \int_{\Omega_x} \underline{f_r(x, \omega', \omega)} L_i(x, \omega') \langle \omega', n \rangle d\omega'.$$



Rendering skin transparency

- BSSRDF
- Real-time rendering with GPU



J. Jimenez et al. IEEE CG&A 2010



Rendering hair

- Focusing on
 - Anisotropic reflection
 - Multiple scattering



A. Zinke et al: Dual Scattering Approximation for Fast Multiple Scattering in Hair (SIGGRAPH2008)

Facial expression

表情のモデル



FACS

● Facial Action Coding System

- Ekman and Friesen 1978 (心理学)
- 顔の表情をアクションユニット(Action Unit)に分類する
 - AUの例: brow lowerer, inner brow raiser, wink, cheek raiser, upper lip raiser, jaw drop,... (全部で46 AUs).
- あくまで「分類」が目的で時間変化(感情の変化)に伴う動きは考慮していない



FACS

AU Number	FACS Name	Muscular Basis
1	Inner Brow Raiser	Frontalis, Pars Medialis
2	Outer Brow Raiser	Frontalis, Pars Lateralis
4	Brow Lowerer	Depressor Glabellae; Depressor Supercilli; Corrugator
5	Upper Lid Raiser	Levator Palpebrae Superioris
6	Cheek Raiser	Orbicularis Oculi, Pars Orbitalis
7	Lid Tightener	Orbicularis Oculi, Pars Palpebralis
8	Lips Toward Each Other	Orbicularis Oris
9	Nose Wrinkler	Levator Labii Superioris, Alaeque Nasi
10	Upper Lip Raiser	Levator Labii Superioris, Caput Infraorbitalis
18	Lip Pucker	Incisivii Labii Superioris; Incisivii Labii Inferioris
20	Lip Stretcher	Risorius
22	Lip Funneler	Orbicularis Oris
23	Lip Tightener	Orbicularis Oris
24	Lip Pressor	Orbicularis Oris
25	Lips Part	Depressor Labii, or Relaxation of Mentalis or Orbicularis Oris
26	Jaw Drop	Masseter; Temporal and Internal Pterygoid Relaxed
27	Mouth Stretch	Pterygoids; Digastric
28	Lip Suck	Orbicularis Oris



FACS応用

- DreamWorks Animationでの例:

- 各AUをパラメータ制御
- 4層からなる制御システム:

Bone layer -

- 1) 頭蓋骨 --> 骨の上のflesh (顔表面)の動きの近似に用いる ;
- 2) muscles for the jaw.

Mask layer

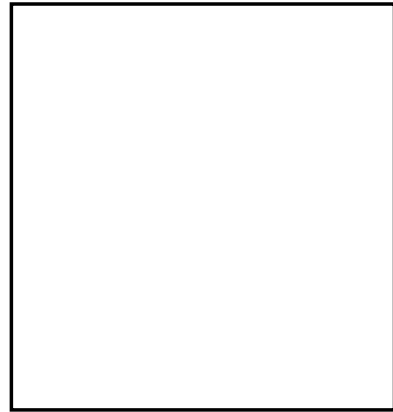
- 1) muscles for the flesh
- 2) low resolution surfaces
- edge = muscle

Flesh layer

- 1) fleshy fat gets pushed by around muscles; cheek, lips
- 2) moved by mask layer

Skin layer

- 1) fine detailed functionality e.g., 皺



Facial Animation

表情アニメーション

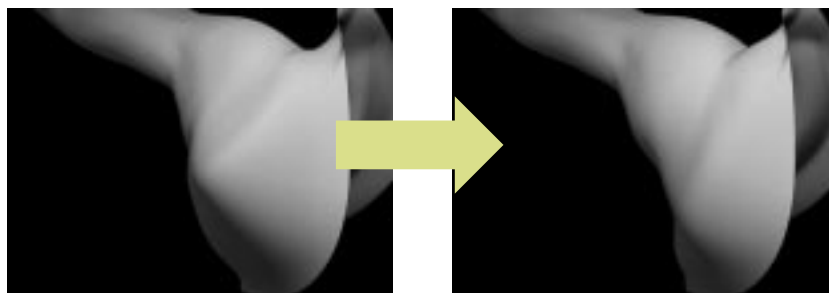


準備: CGモデル

- スケルトンモデル (skeleton model) は、階層的に定義されている.
- 各ポーズは頂点の位置ベクトル $v_i \in \mathbf{R}^3$ で定まるが、表面(skin)は、簡単には定まらない.
- **スキニング** (skinning): スケルトンモデルの位置、回転、スケール情報から表面の頂点位置を決定すること.



準備: さまざまな変形技術

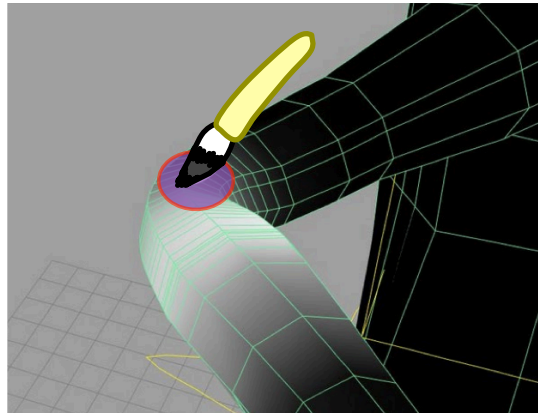


Skeleton subspace deformation
Pose space deformation
Cage deformers
Muscle systems



SSD: Skeleton Subspace Deformation

- 最も実地的な手法
- 各頂点の「変形」操作後の位置は、アフィン変換後の重み付け線形和で表される.
- 頂点のウェイト付けはペイント処理で行う



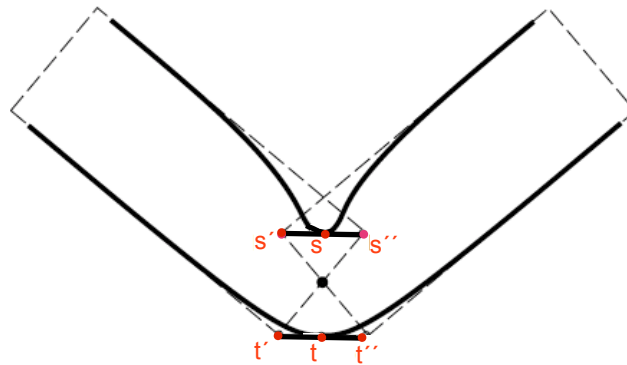
SSD: Skeleton Subspace Deformation

- 各ポーズは頂点の位置ベクトル $v_i \in \mathbf{R}^3$ で定まるが、表面(skin)の変形後の位置は、簡単には定まらない (それを定める標準的な手法がない).
- この問題に対する最初に出た方法、すなわち、スキニング方法が SSD である.

SSD: Skeleton Subspace Deformation

- メッシュ頂点毎に重み付きの同次変換:

$$\mathbf{v}'_i(\mathbf{p}) = \sum_{j=1}^J w_{i,j} \mathbf{M}_j(\mathbf{p}) \mathbf{v}_i = \left(\sum_{j=1}^J w_{i,j} \mathbf{M}_j(\mathbf{p}) \right) \mathbf{v}_i,$$



図の例では、上記の変換を施すと線分 $s's''$ 上のどこかに移る.

→ これは 'collapsing elbow' problem を引き起こしてしまう(後述).

SSD: 問題点

- Joint collapsing artifact
- Candy-wrapper artifact



SSD: Skeleton Subspace Deformation

- メッシュ頂点毎に重み付きの同次変換:

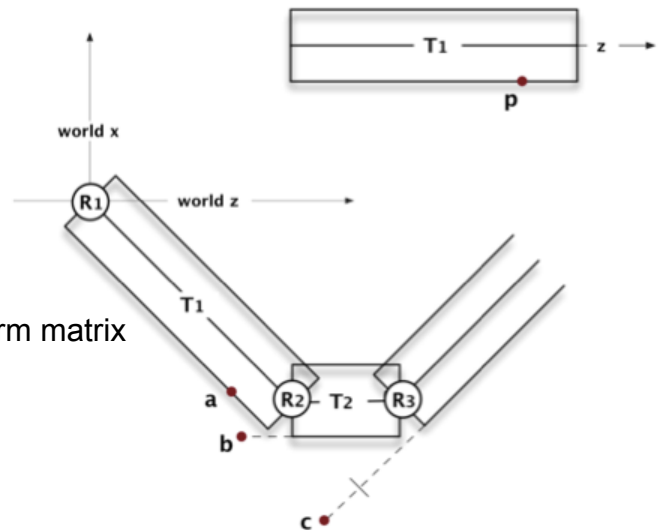
$$S(v) = \sum_{j=1}^{n_{joint}} w_j(v) T_j v$$

$S(v)$: Skin deformation

$v = (x, y, z, 1)$: vertex

$w_j(v)$: weight at vertex v

$T_j \in R^{4 \times 4}$: homogeneous transform matrix



PSD: Pose Space Deformation

- アーティストが頂点を編集

$$S(v) \Rightarrow v_k$$

- Skin deformation変換前の移動に置き換え

$$v_k = S(v_0 + d_k)$$

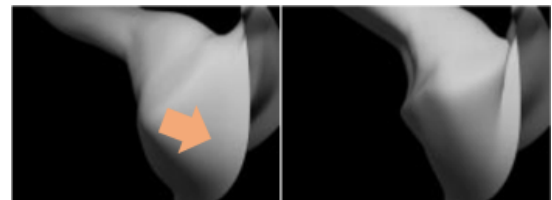
$$d_k = \left(\sum_{j=1}^{n_{joint}} w_j(v) T_j \right)^{-1} v_k - v_0$$

$v_0 \in R^4$: Skin deformation前の頂点座標

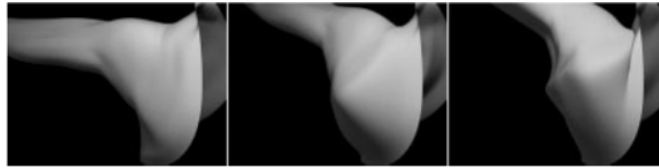
$v_k \in R^4$: 編集後の頂点座標

$d_k \in R^4$: 頂点移動

$T_j \in R^{4 \times 4}$: 同次変換行列



- Results:



- Applicable for facial animation

ブレンドシェープ

Original formulation
Direct manipulation
Learned direct manipulation



Blendshapes ブレンドシェープ

- Semantic Approach
- Facial Animation のスタンダード
 - 色々な顔の表情の「決め」ポーズをまず沢山作る
 - それらの線形和で顔を作り、その重みを時間変化(補間)してアニメーションにする
 - 直感的だが手間がかかる



ブレンドシェープ : Formulation

- 顔ひとつひとつは、ポリゴンモデルとする
- 顔のポリゴンを形成する頂点列 $(x_1, y_1, z_1), (x_2, y_2, z_2), \dots$ を一列に並べた(縦)ベクトルを考える.
- 一般的な顔 $\mathbf{f}$ は、neutral face を $\mathbf{f}_0$ と記し、行列 $\mathbf{B}$ は target face $\mathbf{f}_i$ と $\mathbf{f}_0$ の差 $(\mathbf{f}_i - \mathbf{f}_0)$ を並べて作る:
$$\mathbf{f} = \mathbf{B}\mathbf{w} + \mathbf{f}_0$$
 - これにより、 $0 \leq \mathbf{w} \leq 1$ の範囲で顔のバリエーションが出来る
 - $\mathbf{w}$ は各キーフレーム(keyframe)での設定をアニメータが行い、後はコンピュータで補間される



ブレンドシェープ：利点と欠点

- 単純な仕組み、直感的
 - スライダーバーを動かして w を調整する
 - 微調整には向いている
- [問題] ポーズが増えると操作が膨大な手間となる
 - 映画「アバター」では主人公に2000個以上のtarget shapeを制作: 一人を作るのに 2.5年



Direct Manipulation Blendshapes

- [問題] の解決策
- 顔モデルへの直接(編集)操作から w を割り出す
 - 直接操作とスライダー調整という2つの編集モードで表情を編集できる
 - See more in [Lewis and Anjyo 2010: IEEE CG&A]



Direct Manipulation Blendshapes

- [解法] 次の最適化問題を解く

$$\min_{\Delta w} \left\{ \left\| \overline{B} \Delta w - \overline{\Delta m} \right\|^2 + \alpha \left\| \Delta w \right\|^2 \right\}$$

$\overline{\Delta m}$: 制約(ピン止め)または、移動(ドラッグ)を施した頂点についての変化量.

$\overline{B}$: B から上記の操作に対応した頂点列を抜き出した行列

Δw : 上記操作にともなう重み係数の変位

Direct Manipulation Blendshapes: 評価

- 操作性の向上: 当たりをつけるラフデザイン作業がスピードアップ

タスク項目	デザイン時間
DMB によるおおまかなデザイン	10 分
スライダー(グラフエディター)に詳細なデザイン	30 分



Learned Direct Manipulation

- We have lots of motion capture data or animation archives.
- We want to accelerate the edit process by “learning”.



DMB応用編 (学習版)

- **Step 1:** アニメーション(時系列データ) の学習: 次の共分散行列 A の (正固有値の) 固有ベクトルを用いて顔ベクトルを表現する

$$A = E[(x - e_0)(x - e_0)^T]$$

- ここで x は時系列データ(縦ベクトル)を表す
- e_0 は時系列データからできる平均ベクトル
- A は顔の頂点ベクトル総数×3 のサイズの正方行列

- **Step 2:** 前記の固有ベクトルを並べた行列を U とし、顔ベクトルをこれら固有ベクトルの線形和で表すときの係数を c とし、次の最適化問題を解く:

$$\min_{\Delta c} \left\{ \left\| \overline{U} \Delta c - \overline{\Delta m} \right\|^2 + \beta \left\| \Delta c \right\|_A^2 \right\}$$

- ここで $U = (u_1, u_2, \dots, u_p)$; $A u_i = \lambda_i u_i$ で $\lambda_i > 0$ ($1 \leq i \leq p$).

$$\left\| d \right\|_A^2 := \sum_{i=1}^p \frac{d_i^2}{\lambda_i}$$

- **考察:** DMBとLDMのそれぞれで、似たような最適化問題を解いた
 - どちらも正値対称行列 A から作られる像空間 $A(\mathbf{R}^n)$ における極値問題を解いている.
 - DMB では $A = E$ (単位行列)
 - LDM では $A =$ 共分散行列
 - 像空間 $A(\mathbf{R}^n)$ を、有限次元の再生核ヒルベルト空間 (RKHS, *Reproducing Kernel Hilbert Space*) と見なすと、ノルム $\|\cdot\|_A$ は RKHSに対するノルムと一致している.



LDM: w と c の関係

- 学習効果を w の値として反映させたい
- もともと次の関係がある:

$$Uc + e_0 = Bw + f_0$$

- 従って c と w との関係は次のようになる:

$$w = (B^T B)^{-1} B^T (Uc + e_0 - f_0)$$

$$c = U^T (Bw + f_0 - e_0)$$



Lipsync + DMB

DMBの応用



リップシンク

- 音声に合わせて口の動きを自動的に作成
 - : 無表情のリップシンク
 - ×: 表情を後付けするのは大変



PSDで表情を作る

- “表情付き”- PSD
 - 表情毎に基本となる口の形を準備
 - 表情+口の形の補間が可能に

●●● Demo