コンピュータグラフィクス論

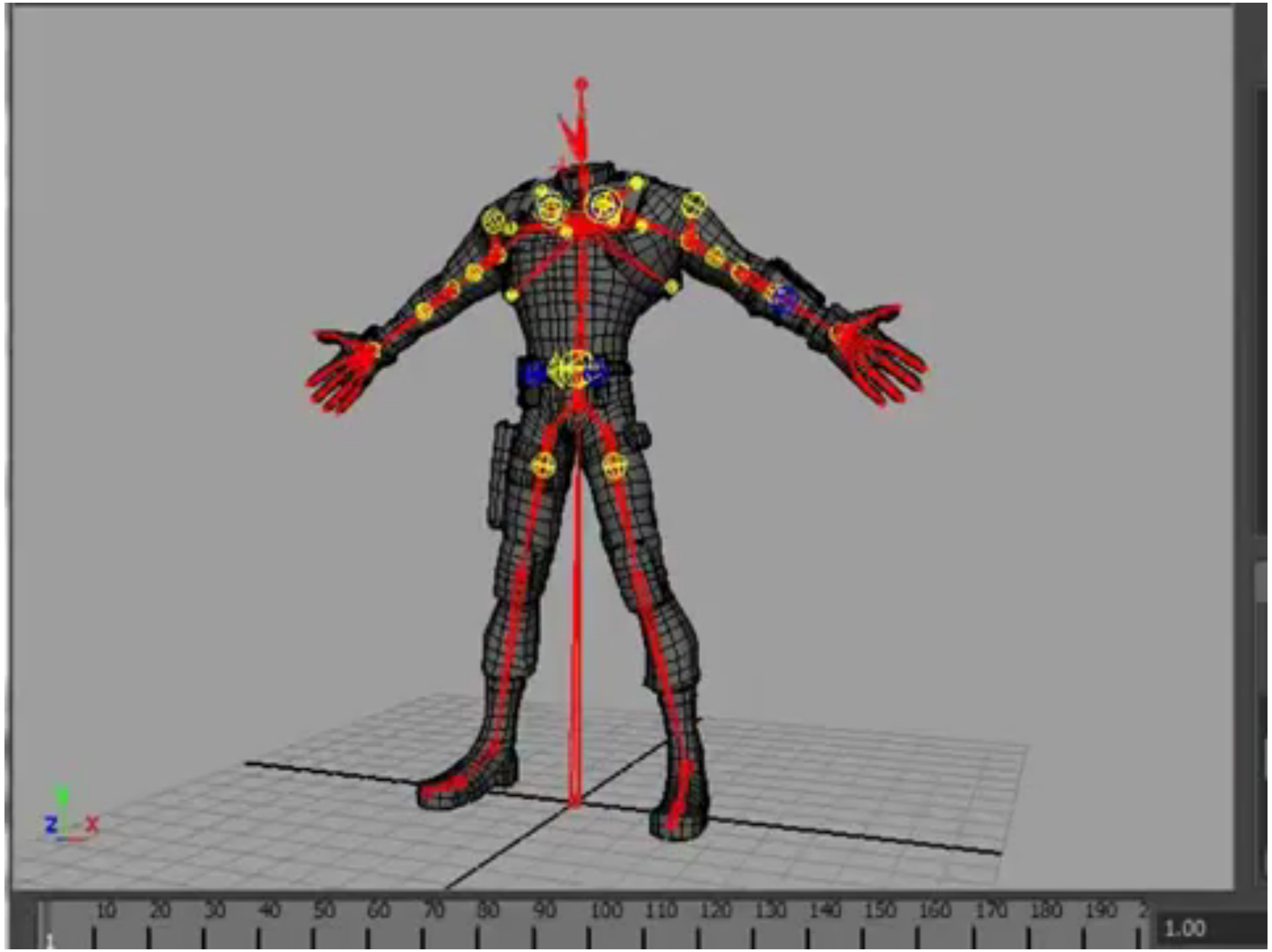
– アニメーション(1) –

2020年7月2日

高山 健志

スケルトンによるアニメーション

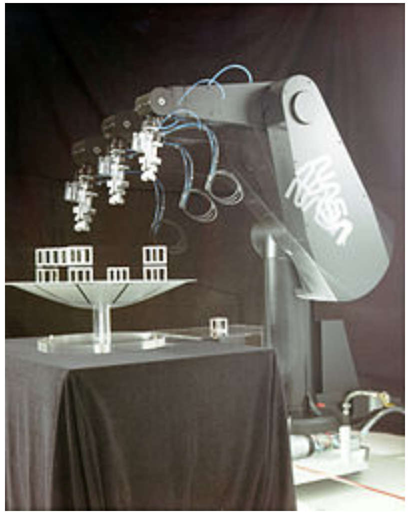
- 単純な仕組み
- 直感的な挙動
- 低い計算コスト

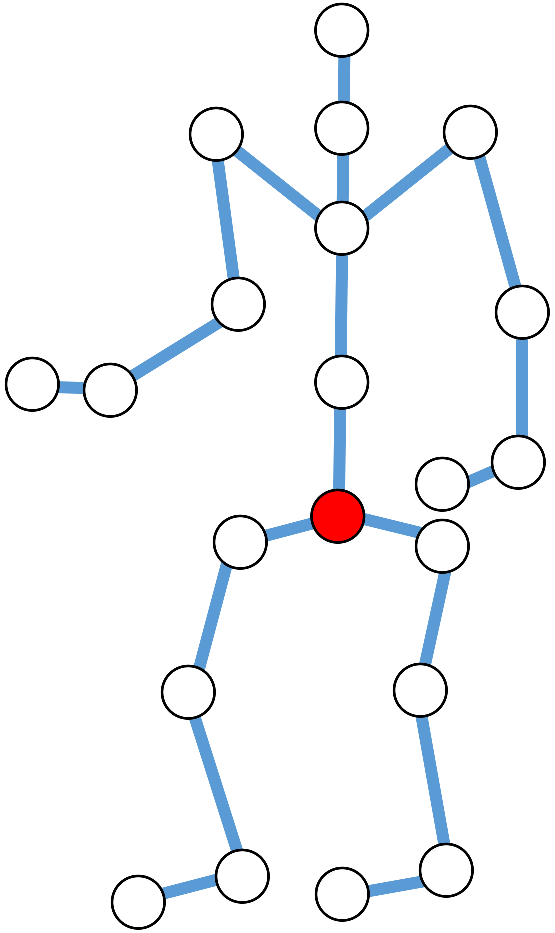


https://www.youtube.com/watch?v=DsoNab58QVA

スケルトンによる姿勢の表現

- ボーンと関節から成る木構造
- ボーンは親関節を基準とした相対的な回転角を保持
- 各関節の回転角によって全体の姿勢を決定
  (**F**orward **K**inematics)
- ロボティクス分野と深く関連

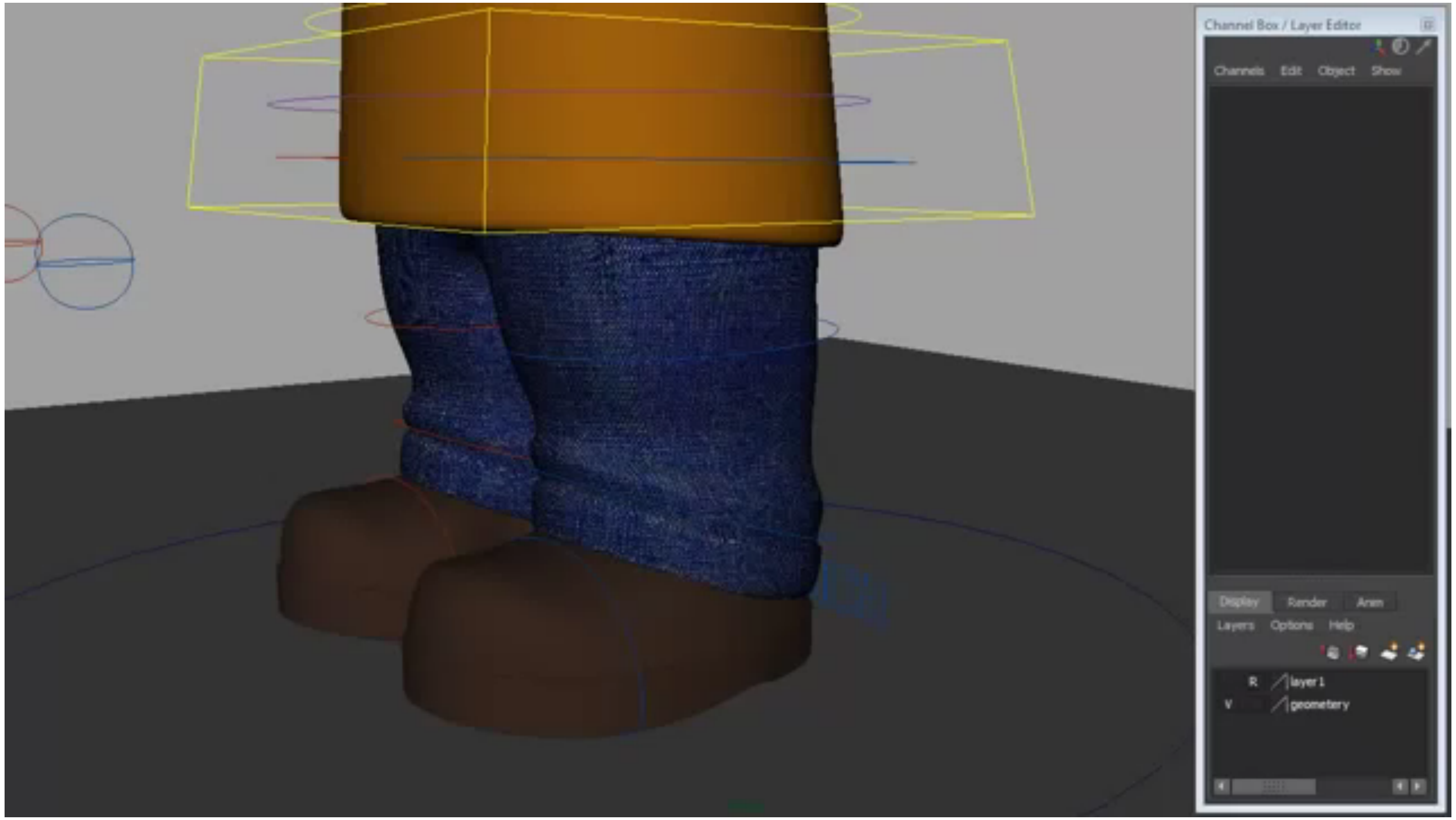




Inverse Kinematics

- 末端関節の位置を与えると、それを満たす関節角を逆算

- IK で手早く姿勢を作り、FK で微調整



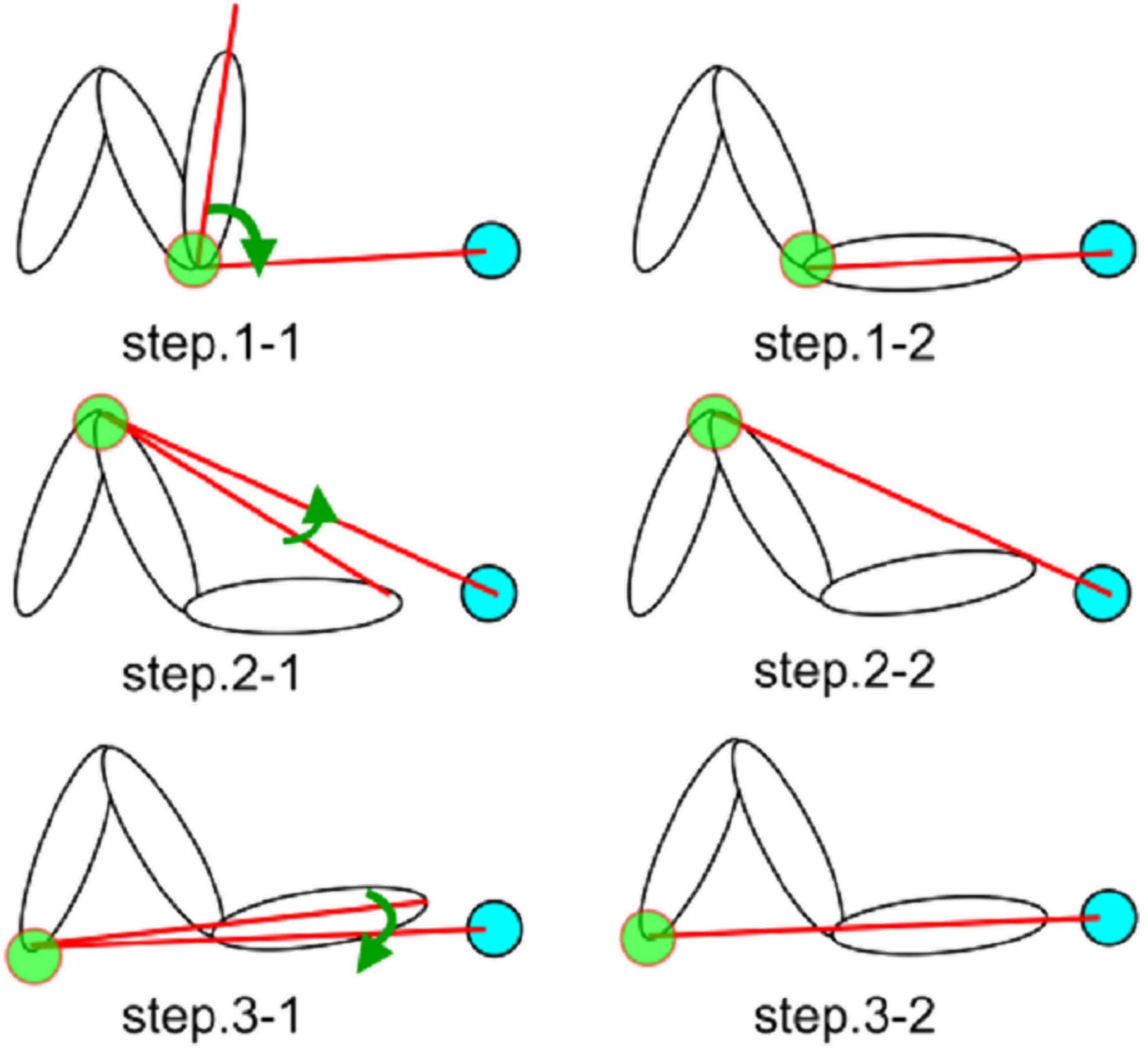
https://www.youtube.com/watch?v=e1qnZ9rV_kw

IK の一解法：Cyclic Coordinate Descent

- 関節角を一つずつ順番に変更
  - 末端関節を目標に近づける
  - 順番が重要！末端が最初

- 実装が簡単 ➔ 基本課題 (デモ)

- より高度な手法
  - ヤコビ法 (方向等の様々な制約)
  - 変形エネルギーの最小化 [Jacobson 12]



https://mukai-lab.org/content/CcdParticleInverseKinematics.pdf

変形エネルギーに基づく IK

https://www.youtube.com/watch?v=PRcXy2LjI9I

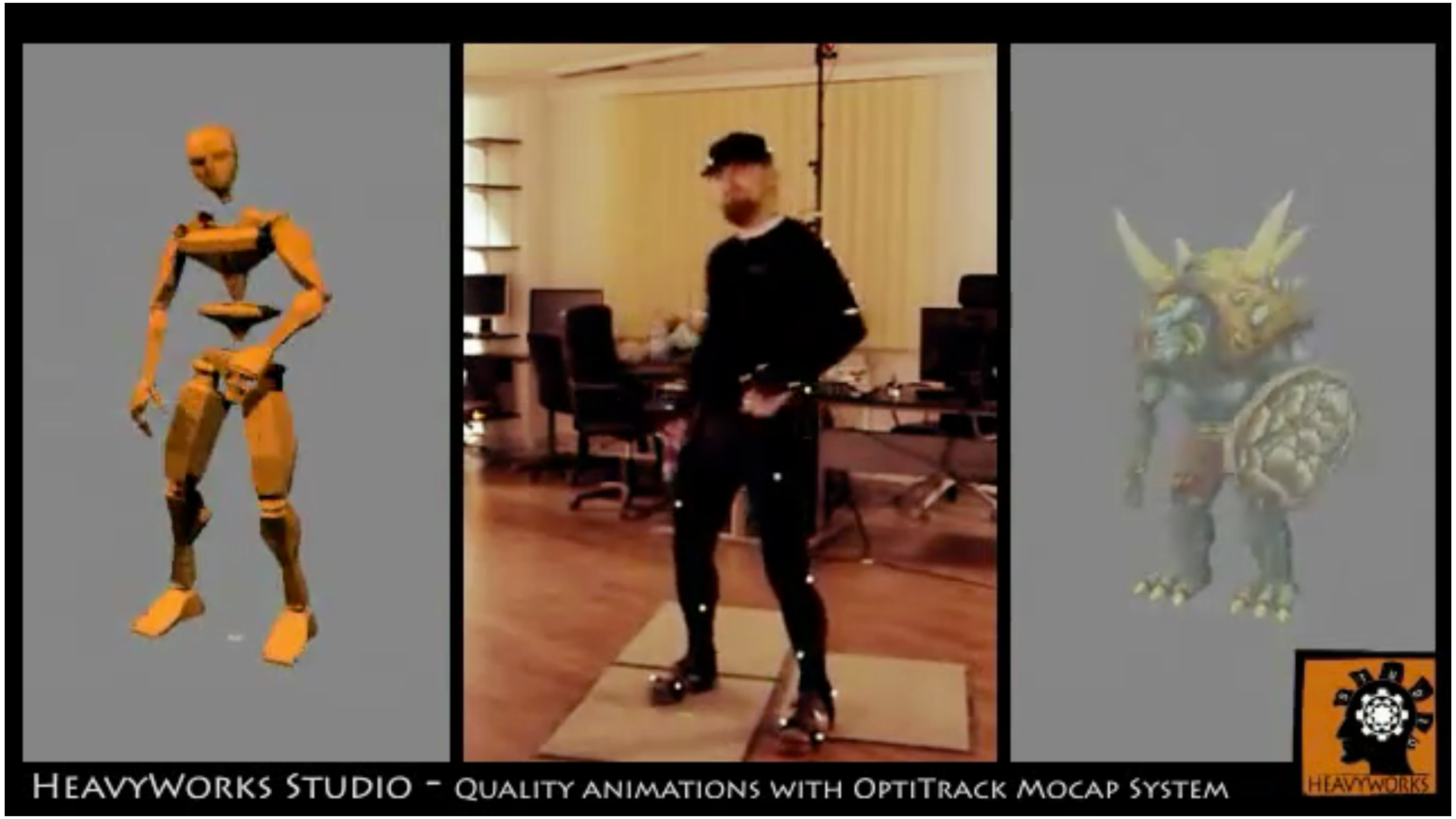
Fast Automatic Skinning Transformations [Jacobson SIGGRAPH12]

モーションデータの取得・生成方法

光学式モーションキャプチャ

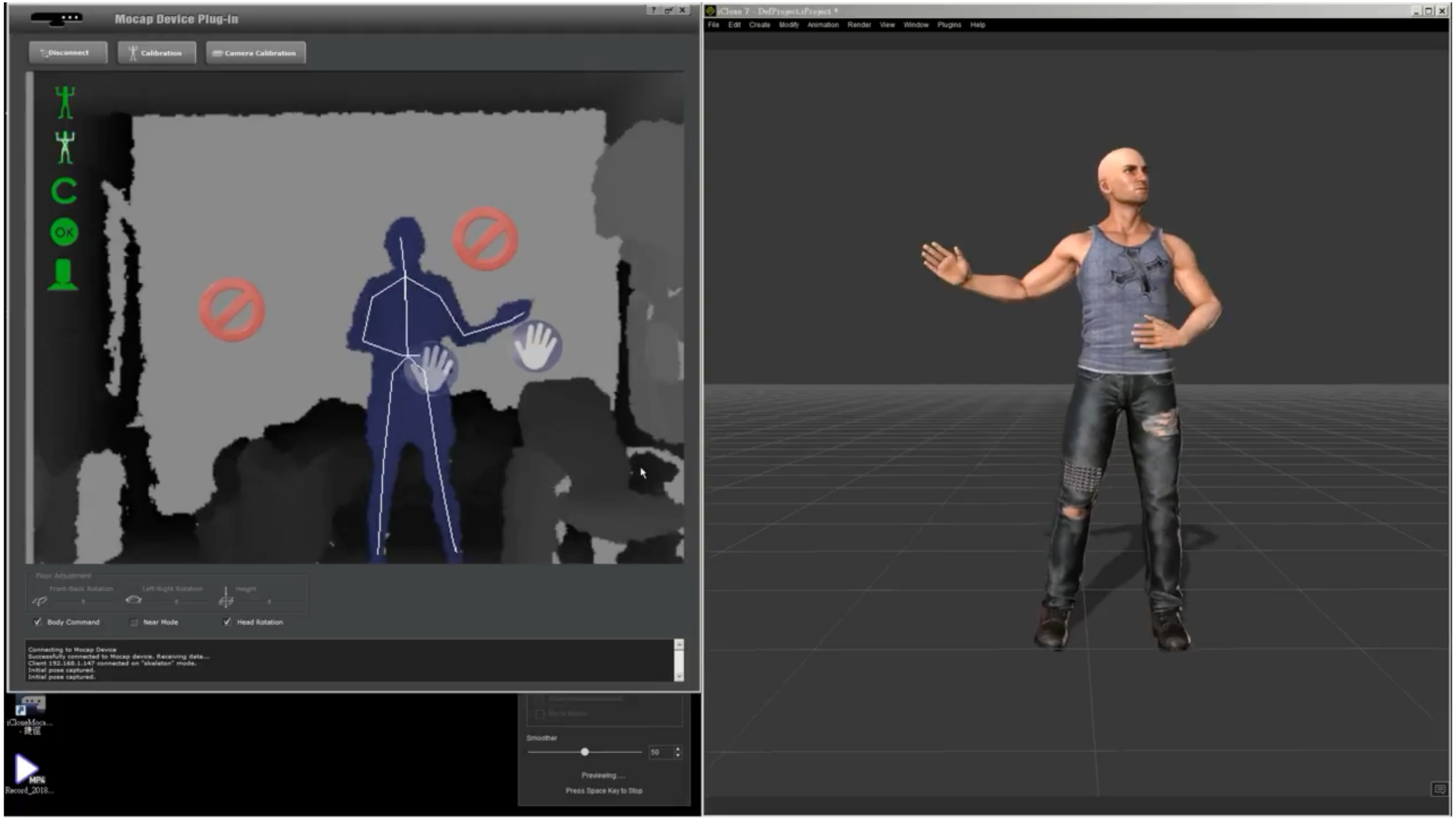
- 役者にマーカーを取り付け、多数 (~48) のカメラで撮影

from Wikipedia



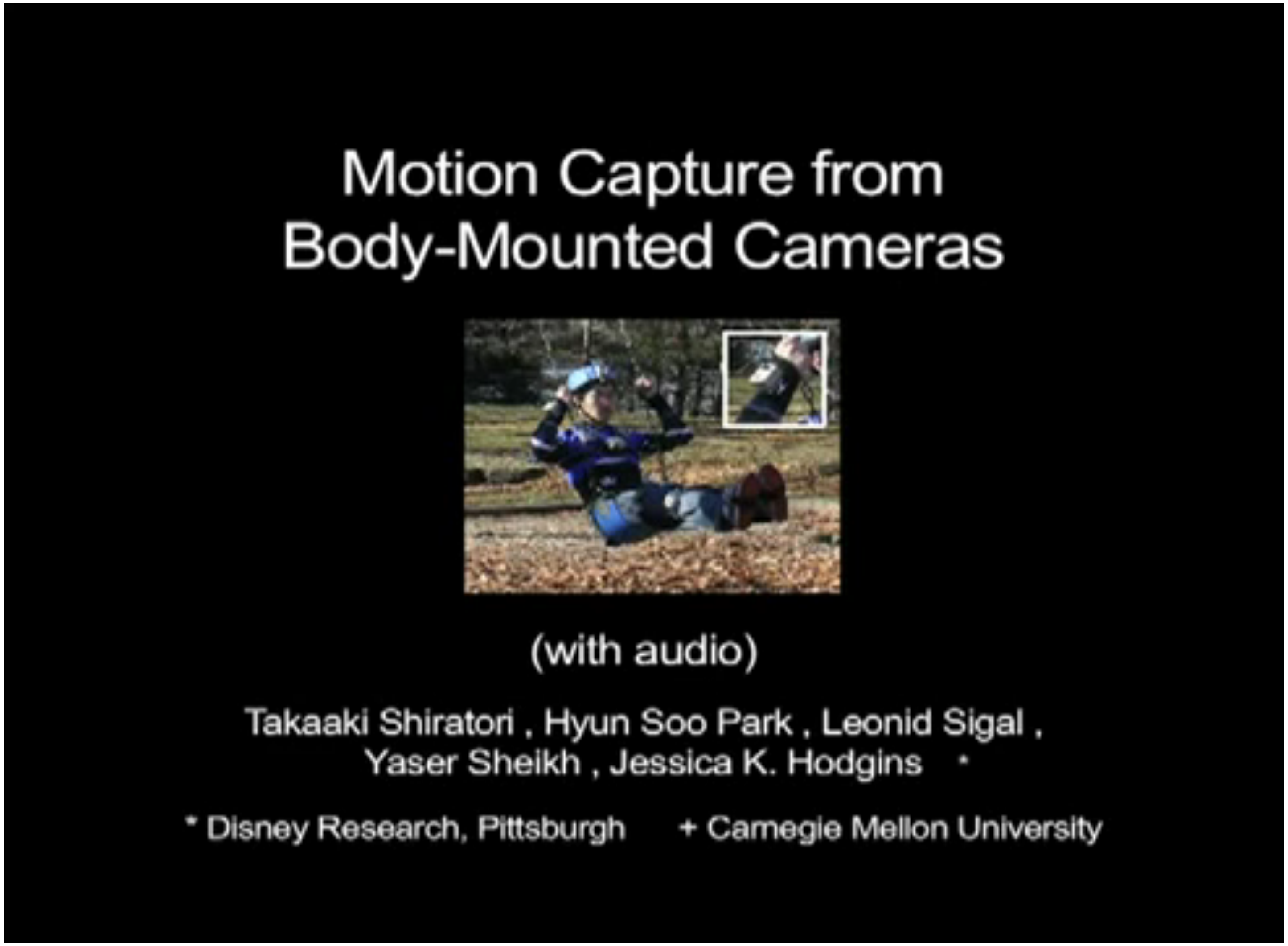
https://www.youtube.com/watch?v=c6X64LhcUyQ

安価なデプスカメラによるモーキャプ



https://www.youtube.com/watch?v=zXDuyMtzunA

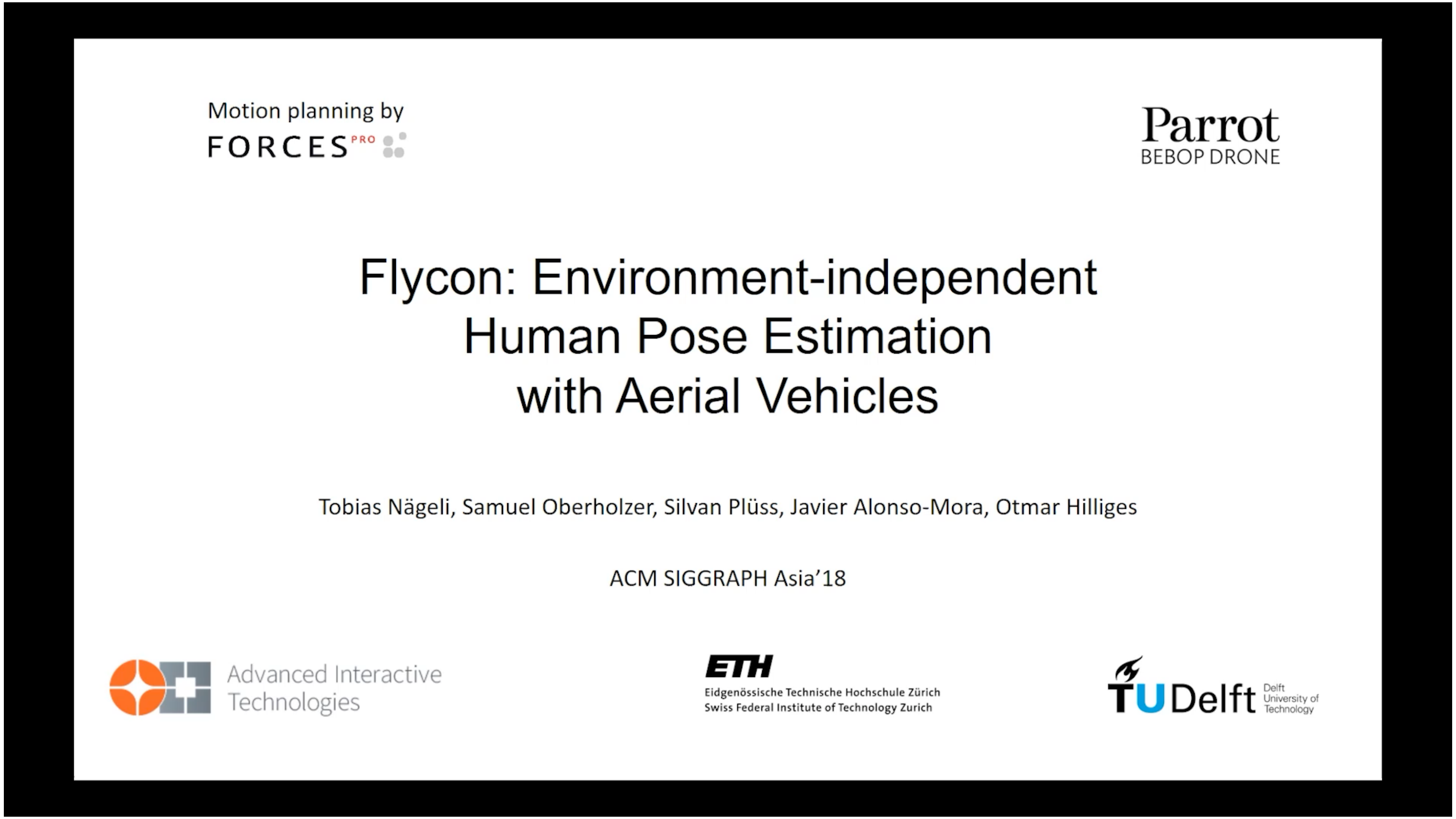
屋外で使えるモーキャプ



https://www.youtube.com/watch?v=xbI-NWMfGPs

Motion Capture from Body-Mounted Cameras [Shiratori SIGGRAPH11]

ドローンの自動追跡によるモーキャプ

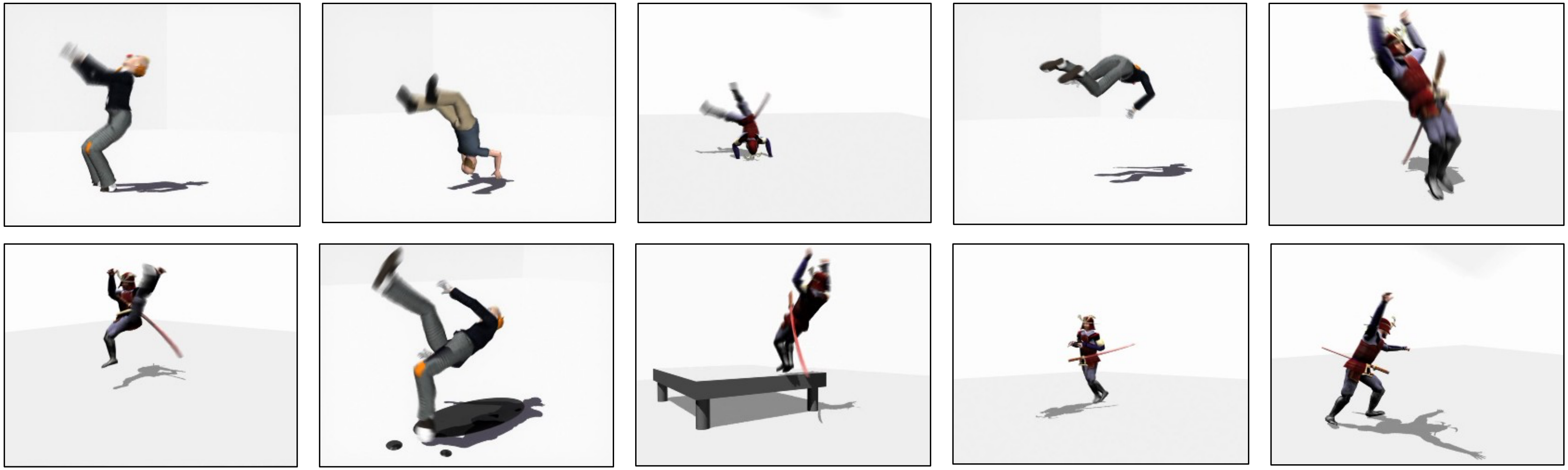


https://www.youtube.com/watch?v=iSJY-vHDmHQ

Flycon: real-time environment-independent multi-view human pose estimation with aerial vehicles [Nageli SIGGRAPHAsia18]

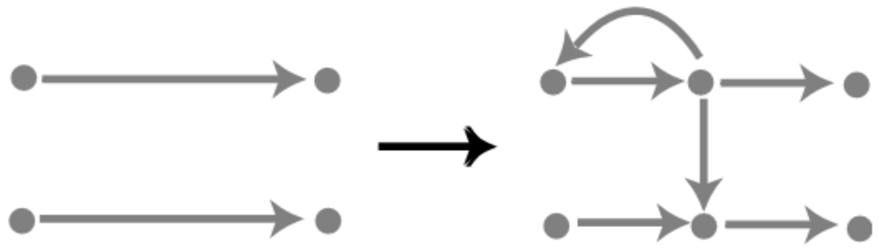
モーションデータベース

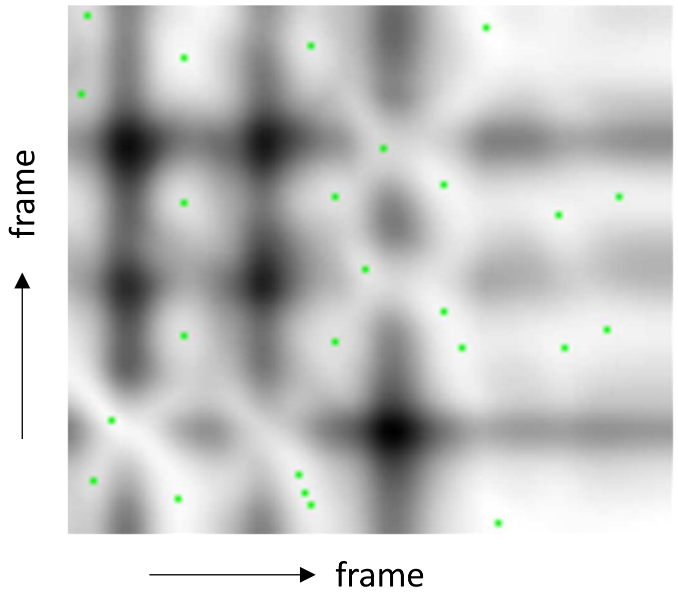
- http://mocap.cs.cmu.edu/
- 6 カテゴリ、合計 2605個
- 研究促進のために無償公開 (補間、連結、解析、検索、etc)



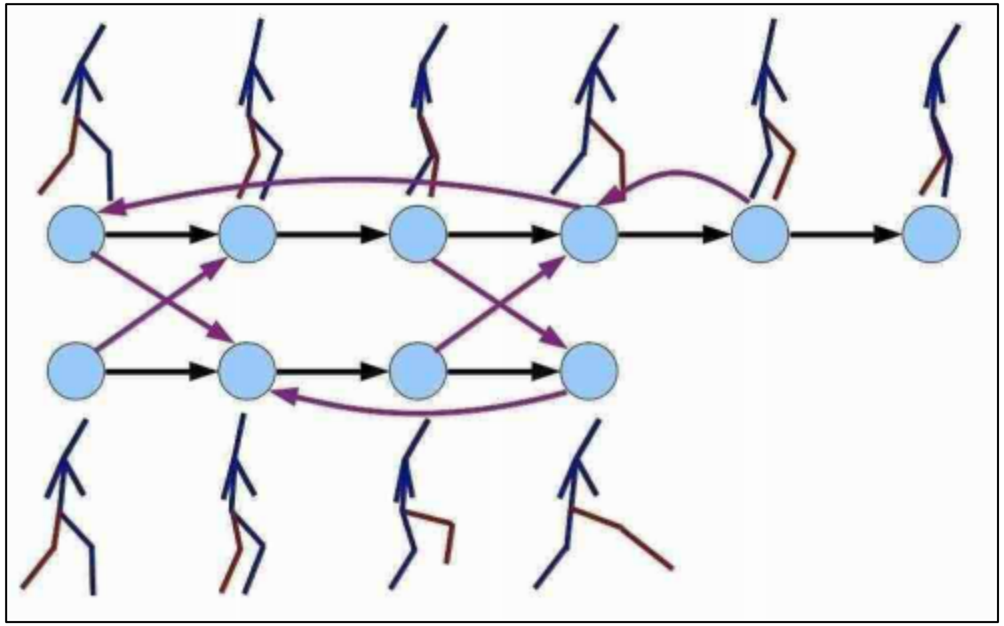
モーションの連結

- 二つのフレームで姿勢が似ていれば、遷移を許す





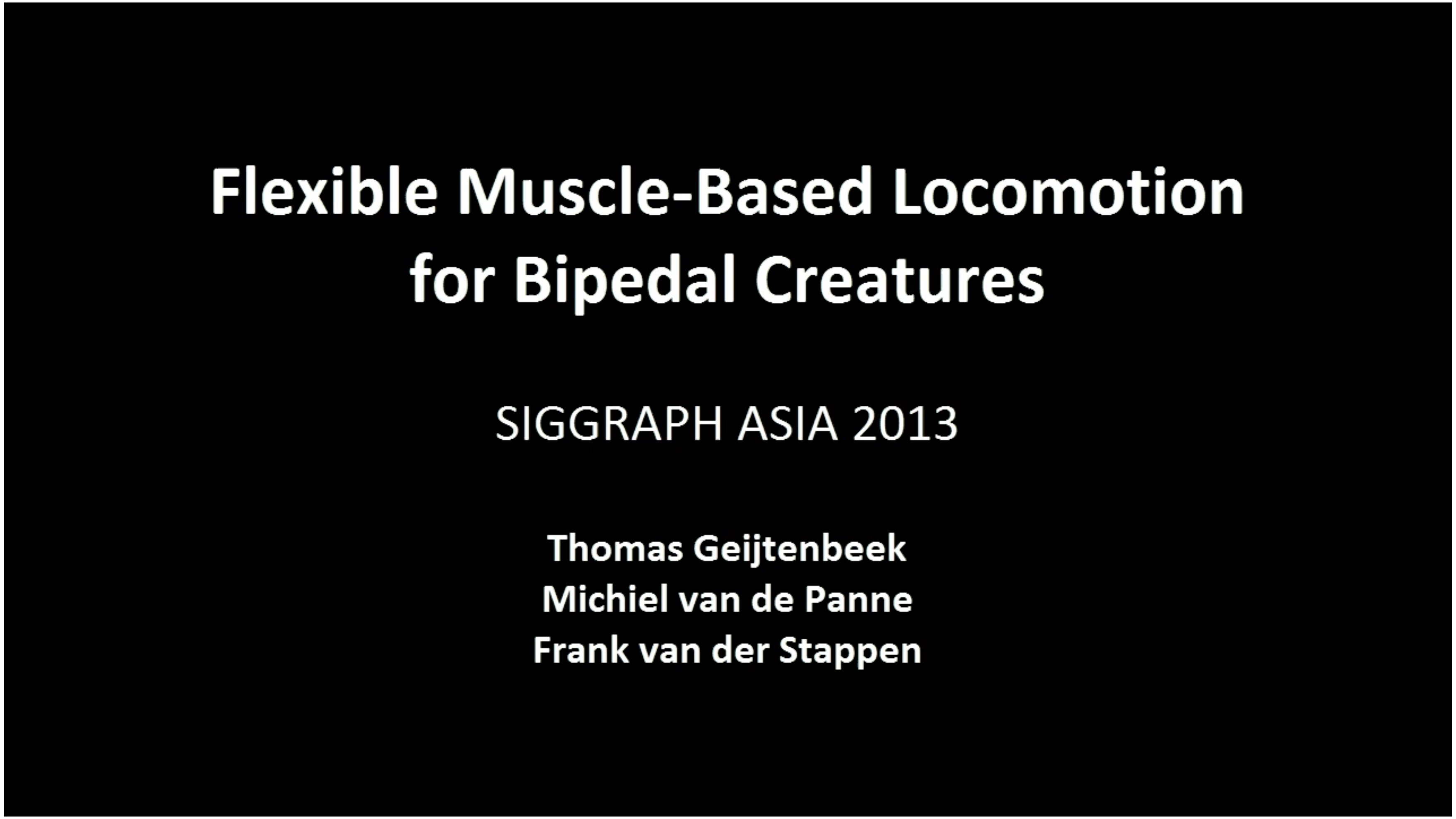
フレーム間の姿勢の類似度



Motion Graphs [Kovar SIGGRAPH02]
Motion Patches: Building Blocks for Virtual Environments Annotated with Motion Data [Lee SIGGRAPH06]
https://graphics.cs.wisc.edu/Papers/2002/KGP02/mograph.pdf

シミュレーションによるモーション生成

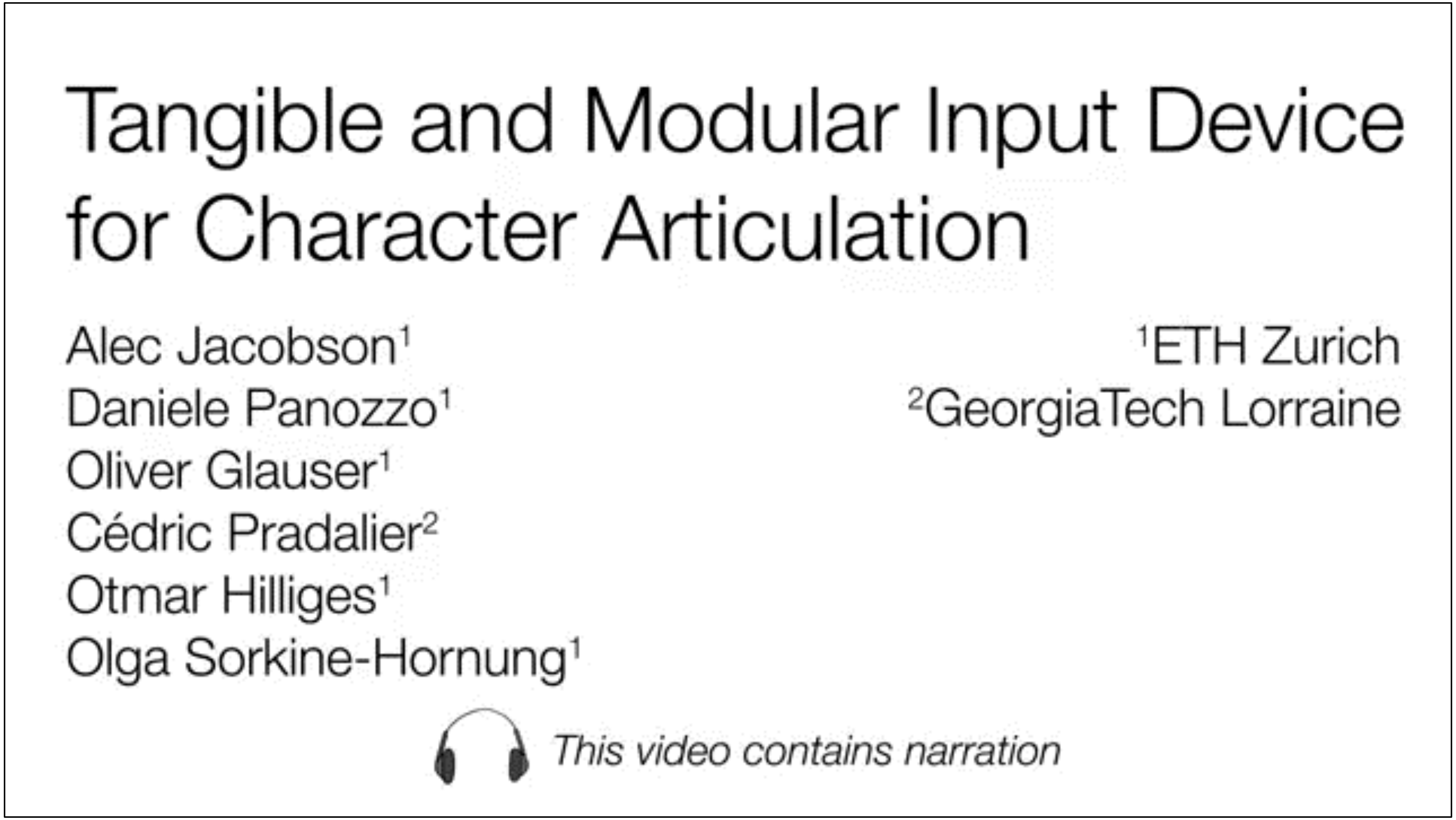
- モーキャプできない対象に使える
- 体型に合った自然な動作を生成できる
- 動的に変化する環境に適応できる



https://www.youtube.com/watch?v=pgaEE27nsQw

Flexible Muscle-Based Locomotion for Bipedal Creatures [Geijtenbeek et al. SIGGRAPH Asia 2013]

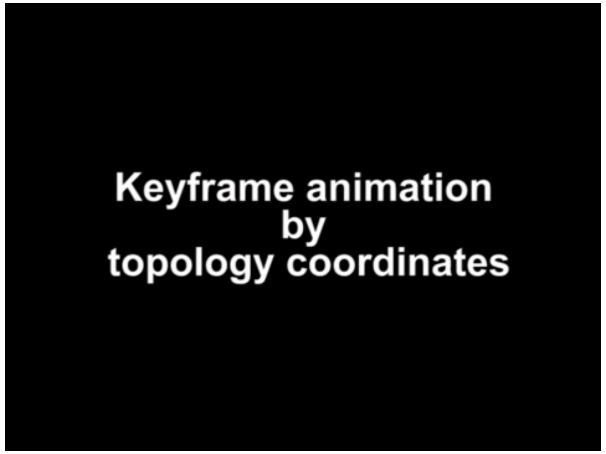
専用デバイスによるポーズ作成



Tangible and Modular Input Device for Character Articulation [Jacobson SIGGRAPH14]
Rig Animation with a Tangible and Modular Input Device [Glauser SIGGRAPH16]

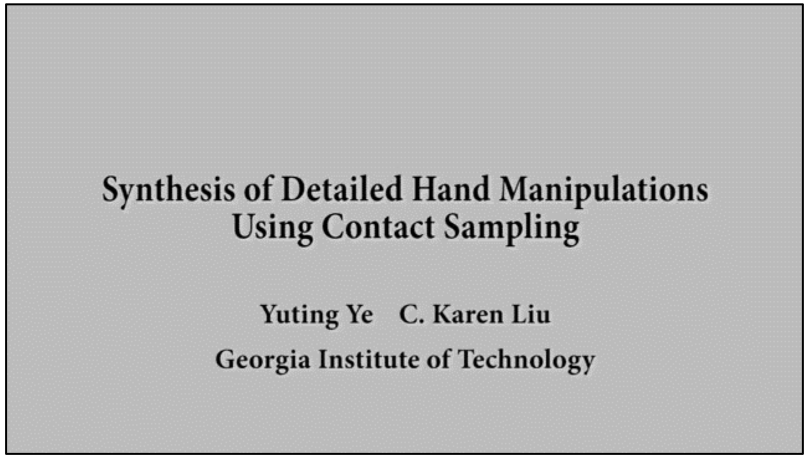
https://www.youtube.com/watch?v=vBX47JamMN0

キャラクタの動きに関する様々なトピック



複数キャラクタの
インタラクション

https://www.youtube.com/watch?v=1S_6wSKI_nU



物体をつかむ動作

https://www.youtube.com/watch?v=x8c27XYTLTo

群衆シミュレーション

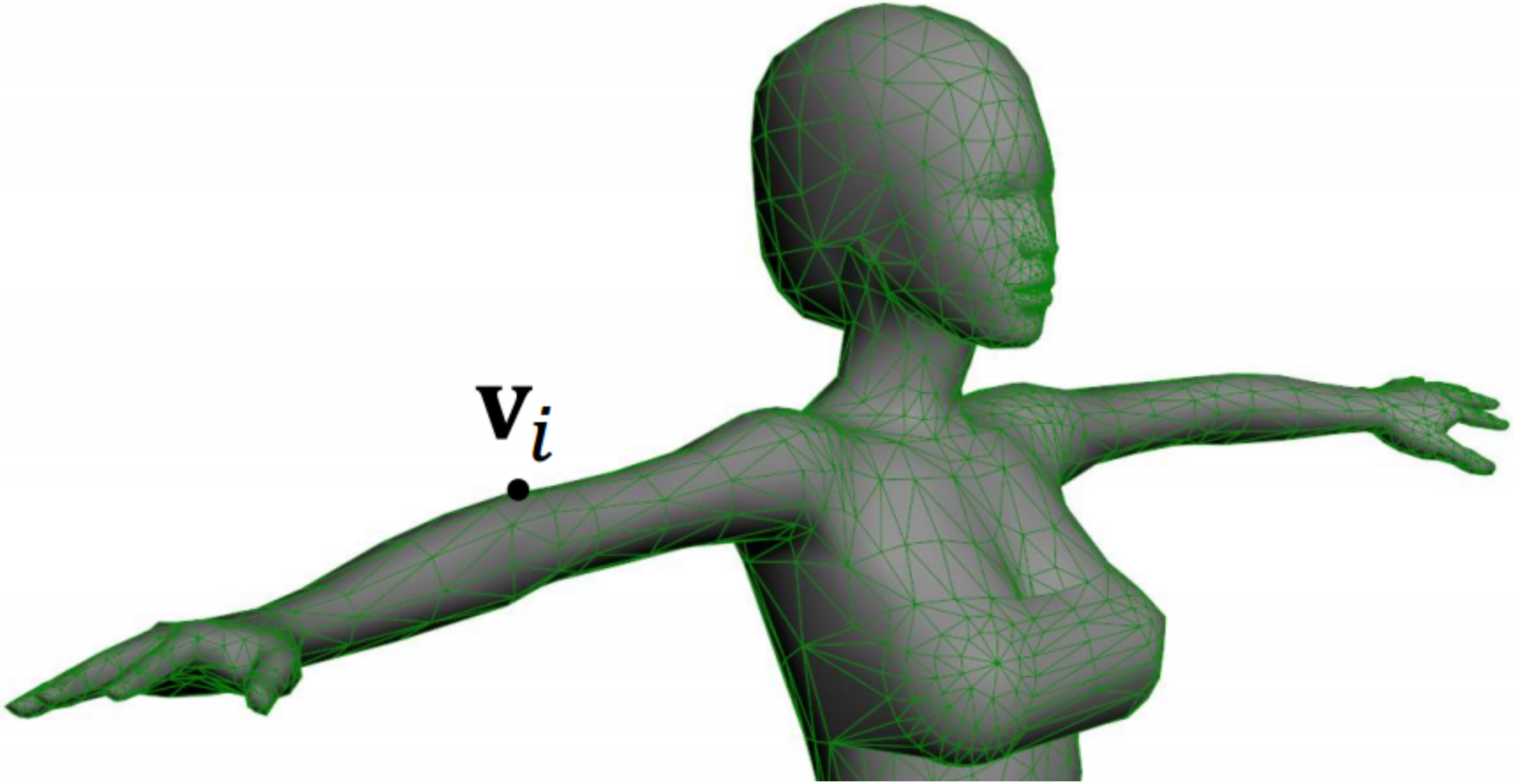
https://www.youtube.com/watch?v=pqBSNAOsMDc

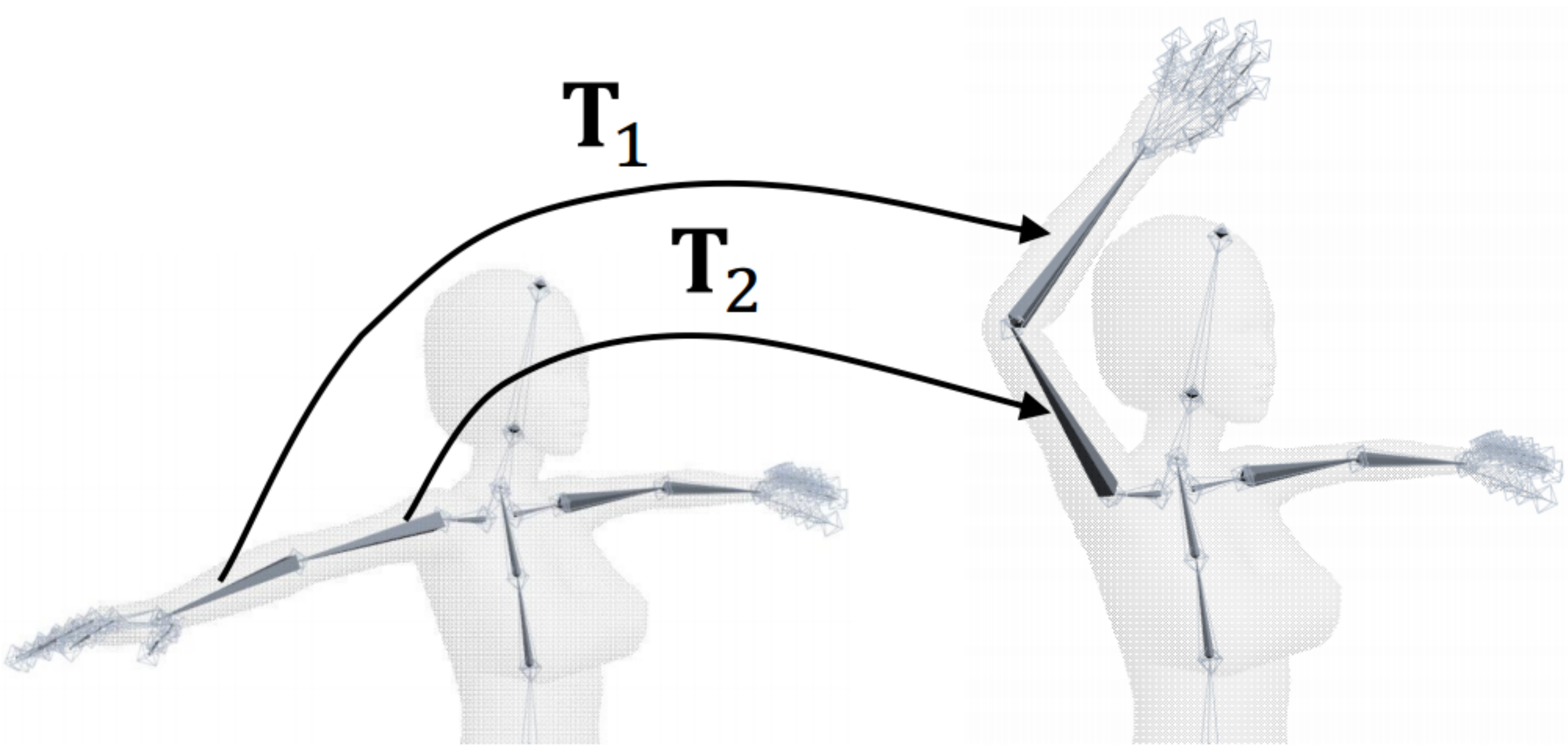
Path planning

https://vimeo.com/33409868

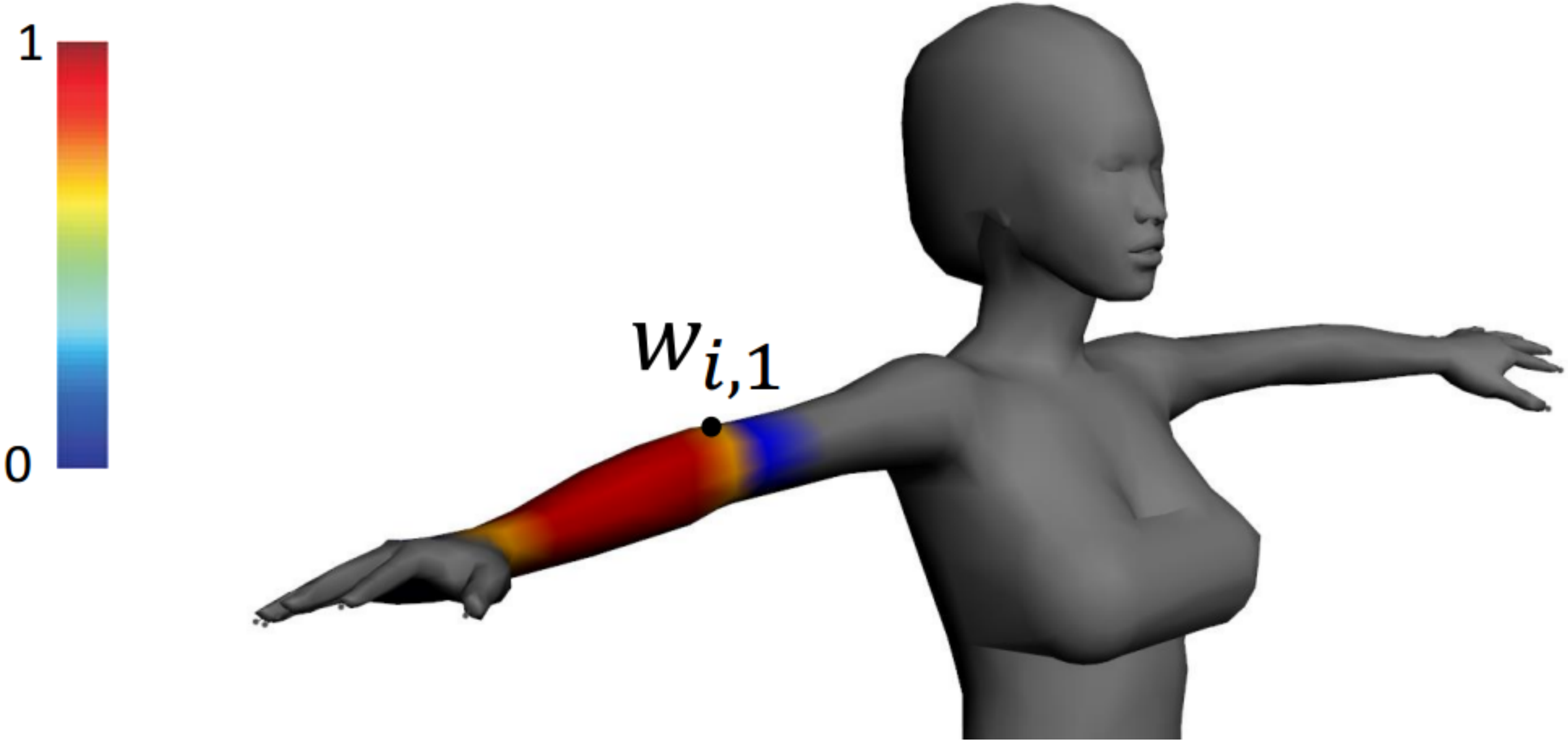
Character motion synthesis by topology coordinates [Ho EG09]
Aggregate Dynamics for Dense Crowd Simulation [Narain SIGGRAPHAsia09]
Synthesis of Detailed Hand Manipulations Using Contact Sampling [Ye SIGGRAPH12]
Space-Time Planning with Parameterized Locomotion Controllers.[Levine TOG11]

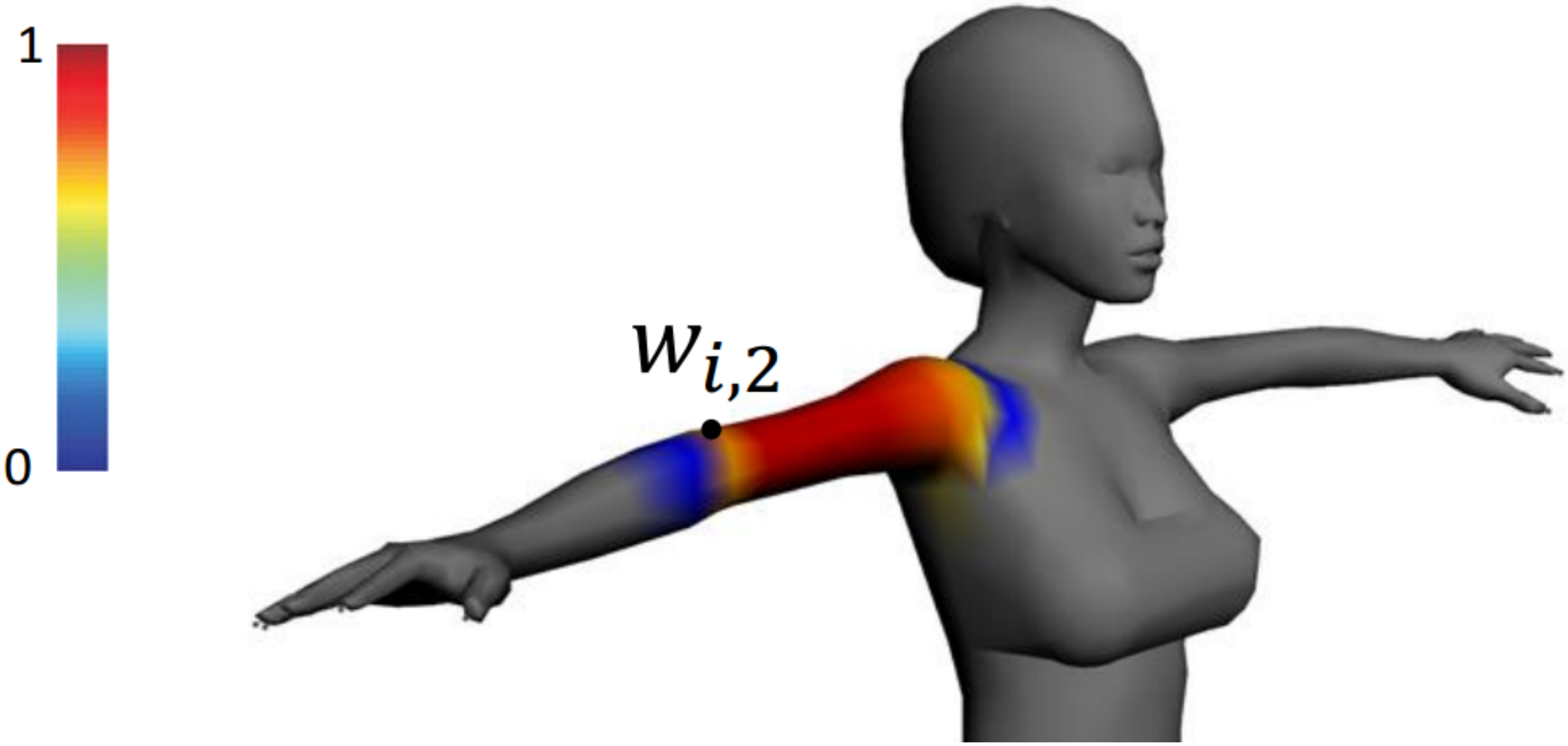
スキニング

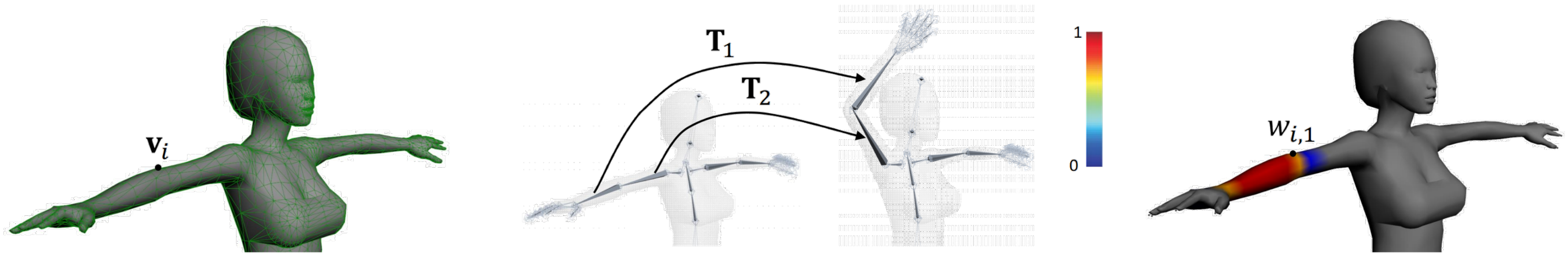

$\mathbf{v}_i$



$\mathbf{T}_1$
$\mathbf{T}_2$


1
0
$w_{i,1}$


1
0
$w_{i,2}$



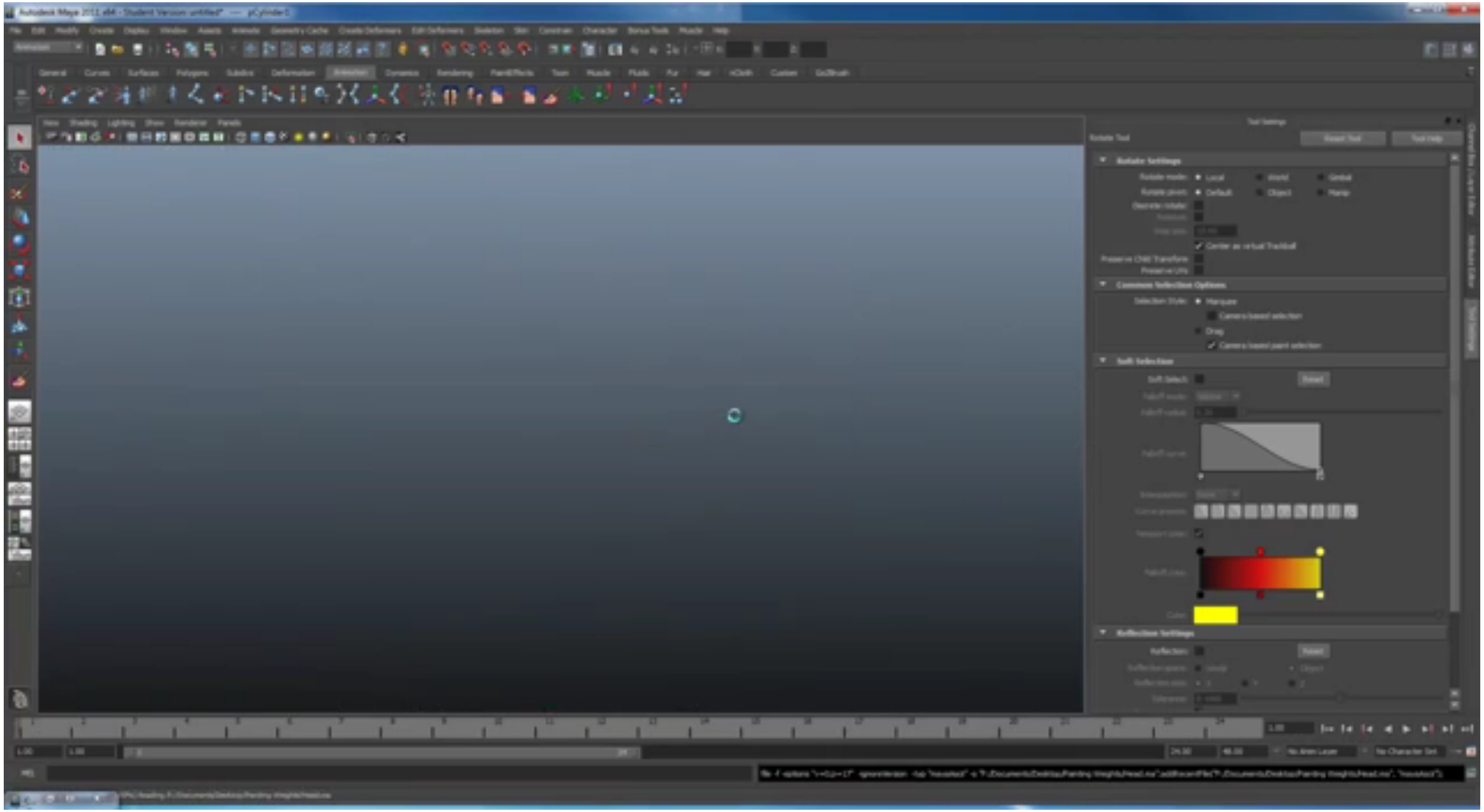
$$\mathbf{v}_i' = \mathrm{blend}\left(\left\langle w_{i,1}, \mathbf{T}_1 \right\rangle, \left\langle w_{i,2}, \mathbf{T}_2 \right\rangle, \dots \right)(\mathbf{v}_i)$$

- 入力
  - メッシュ頂点座標 $\{\mathbf{v}_i\}$ $i = 1, \dots, n$
  - ボーンの剛体変換 $\{\mathbf{T}_j\}$ $j = 1, \dots, m$
  - 各ボーンから各メッシュ頂点への重み $\{w_{i,j}\}$ $i = 1, \dots, n$ $j = 1, \dots, m$

- 出力
  - 変形後のメッシュ頂点座標 $\{\mathbf{v}_i'\}$ $i = 1, \dots, n$

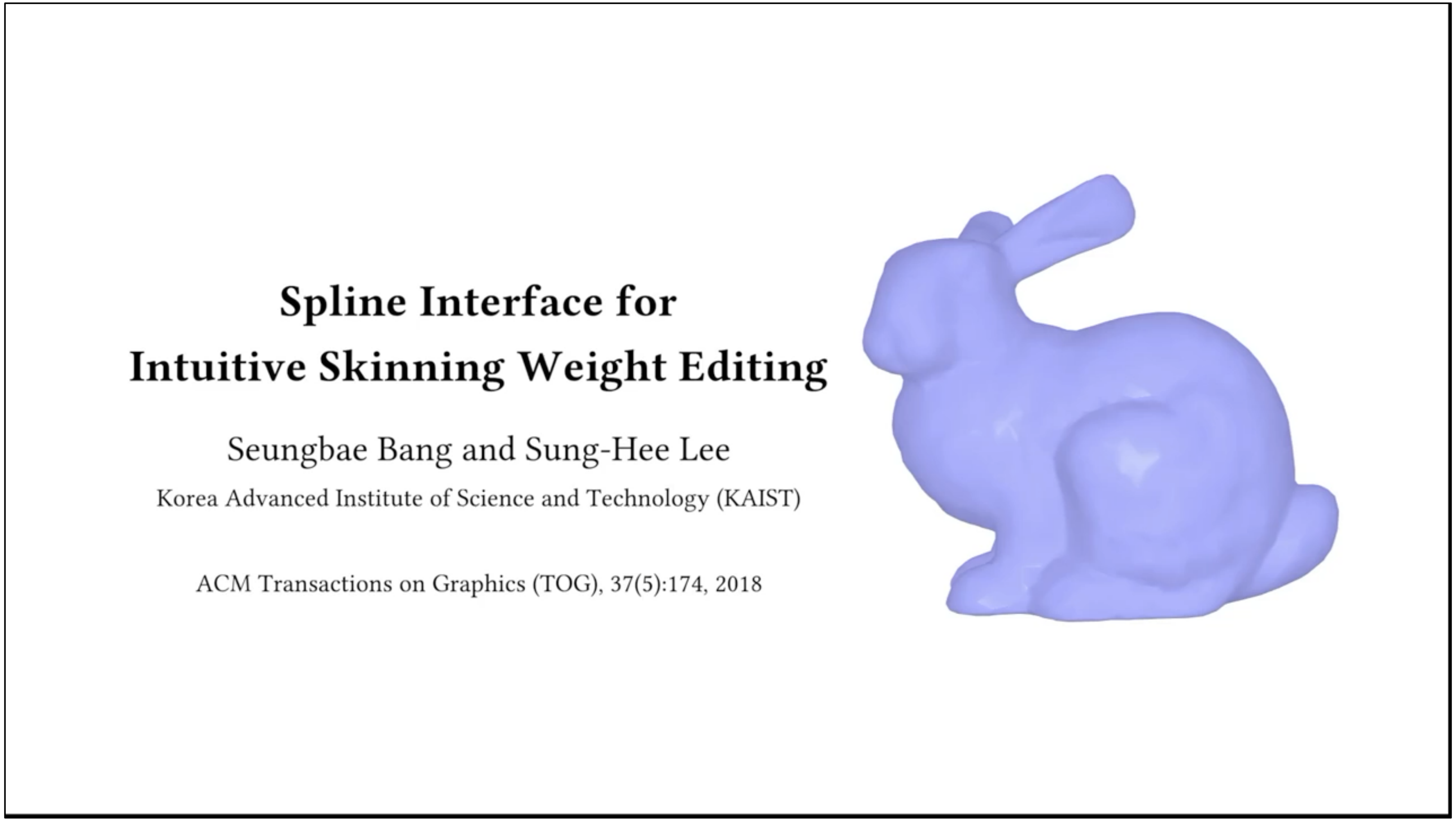
- 技術的なポイント
  - 重み $\{w_{i,j}\}$ をどう与えるか
  - 変換をどうブレンドするか

重みの与え方：手作業でペイント



https://www.youtube.com/watch?v=TACB6bX8SN0

手作業で重みを与えるUIに関する研究



https://www.youtube.com/watch?v=mfEP8BlXTgQ

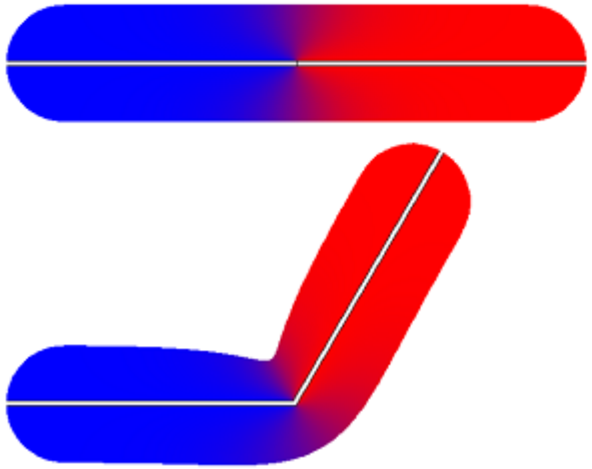
Spline Interface for Intuitive Skinning Weight Editing [Bang,Lee,TOG18]

重みの与え方：自動計算

- j 番目のボーンの重み w_j を、
  - j 番目のボーン上で 1 を取り、それ以外のボーン上で 0 を取り、
  - それ以外では滑らかなスカラー場

  として定式化

- 一階微分 $\int_\Omega \left\|\nabla w_j\right\|^2 dA$ を最小化 [Baran 07]
  - サーフェス上で近似的に解く ➔ 簡単、高速
- 二階微分 $\int_\Omega \left(\Delta w_j\right)^2 dA$ を最小化 [Jacobson 11]
  - 不等式制約 $0 \leq w_j \leq 1$ も導入
  - ボリューム上で二次計画問題を解く ➔ 高品質



Teddy/Pinocchio デモ

Automatic rigging and animation of 3d characters [Baran SIGGRAPH07]
Bounded Biharmonic Weights for Real-Time Deformation [Jacobson SIGGRAPH11]

変換の混合手法：**Linear Blend Skinning**

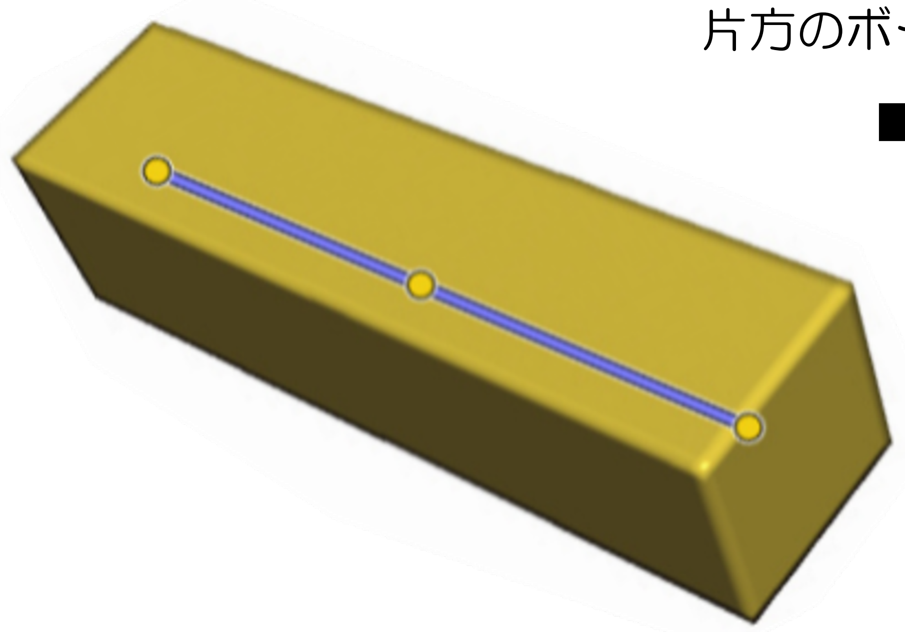
- 剛体変換 $\mathbf{T}_j$ は、回転行列 $\mathbf{R}_j \in \mathbb{R}^{3\times3}$ と移動ベクトル $\mathbf{t}_j \in \mathbb{R}^3$ を並べた 3×4行列として表される

$$\mathbf{v}_i' = \left(\sum_j w_{i,j} \begin{pmatrix} \mathbf{R}_j & \mathbf{t}_j \end{pmatrix}\right) \begin{pmatrix} \mathbf{v}_i \\ 1 \end{pmatrix}$$

- 単純で高速
  - 頂点シェーダで実装：フレーム毎に $\{\mathbf{v}_i'\}$ を GPU に送るのではなく、初期化時に $\{\mathbf{v}_i\}$ と $\{w_{i,j}\}$ を送り、フレーム毎に $\{\mathbf{T}_j\}$ を送る

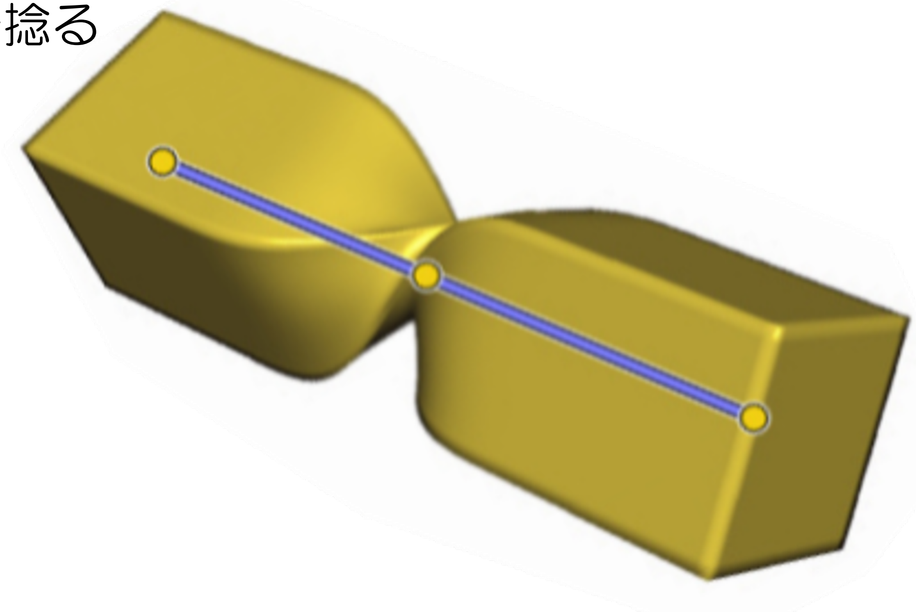
- 業界で最も一般的

LBS の欠陥：”candy wrapper” effect



初期形状と二つのボーン

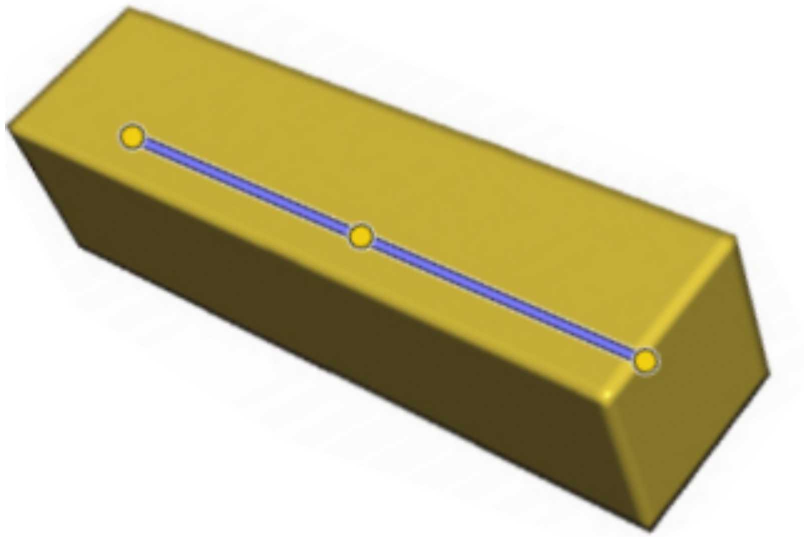
片方のボーンを捻る



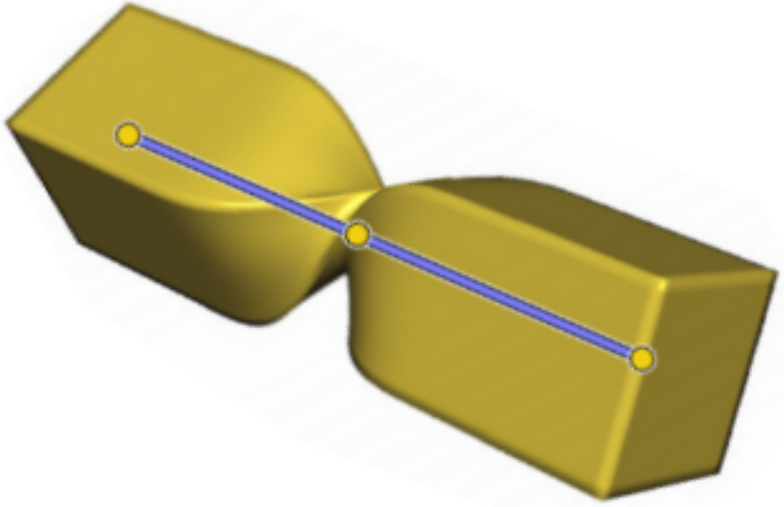
LBS による変形

- 剛体変換の線形和は剛体変換にならない！
  - 180度捻ると関節の周りが一点に凝縮

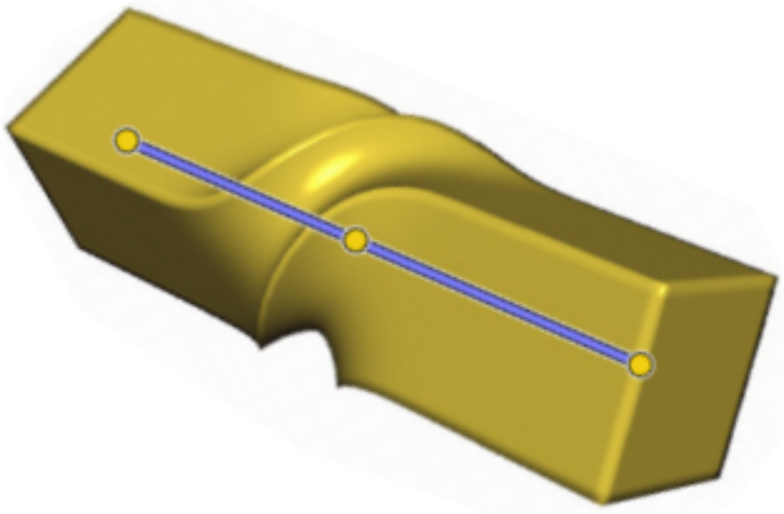
LBS に代わる手法：**Dual Quaternion Skinning**



初期形状



LBS による変形



DQS による変形

- アイディア
  - Quaternion (四つの実数) ➔ 3D 回転変換
  - Dual quaternion (二つの quaternion) ➔ 3D 剛体変換 (回転 + 移動)

Dual number と dual quaternion

- Dual number
  - $\varepsilon^2 = 0$ という演算規則を持つ dual 単位 ε を導入 (cf. 虚数単位 i)
  - Primal 成分と dual 成分 の和として dual number を定義： $\hat{a} \coloneqq a_0 + \varepsilon a_\varepsilon$

    $a_0, a_\varepsilon \in \mathbb{R}$
    - Dual 共役： $\bar{\hat{a}} = \overline{a_0 + \varepsilon a_\varepsilon} = a_0 - \varepsilon a_\varepsilon$
- Dual quaternion
  - Quaternionの 各成分が dual number であるようなもの
  - 二つの quaternion を使って書ける $\hat{\mathbf{q}} \coloneqq \mathbf{q}_0 + \varepsilon \mathbf{q}_\varepsilon$
    - Dual 共役： $\bar{\hat{\mathbf{q}}} = \overline{\mathbf{q}_0 + \varepsilon \mathbf{q}_\varepsilon} = \mathbf{q}_0 - \varepsilon \mathbf{q}_\varepsilon$
    - Quaternion 共役： $\hat{\mathbf{q}}^* = (\mathbf{q}_0 + \varepsilon \mathbf{q}_\varepsilon)^* = \mathbf{q}_0^* + \varepsilon \mathbf{q}_\varepsilon^*$

Geometric Skinning with Approximate Dual Quaternion Blending [Kavan TOG08]

Dual number / quaternion の演算規則

- Dual number $\hat{a} = a_0 + \varepsilon a_\varepsilon$ について：
  - 逆数　$\frac{1}{\hat{a}} = \frac{1}{a_0} - \varepsilon \frac{a_\varepsilon}{a_0^2}$
  - 平方根　$\sqrt{\hat{a}} = \sqrt{a_0} + \varepsilon \frac{a_\varepsilon}{2\sqrt{a_0}}$
  - 三角関数　$\sin \hat{a} = \sin a_0 + \varepsilon a_\varepsilon \cos a_0$
    $\cos \hat{a} = \cos a_0 - \varepsilon a_\varepsilon \sin a_0$

普通の四則演算と新しい規則 $\varepsilon^2 = 0$ を適用すれば、簡単に導出できる

テイラー展開より導出

- Dual quaternion $\hat{\mathbf{q}} = \mathbf{q}_0 + \varepsilon \mathbf{q}_\varepsilon$ について：
  - ノルム　$\|\hat{\mathbf{q}}\| = \sqrt{\hat{\mathbf{q}}^* \hat{\mathbf{q}}} = \|\mathbf{q}_0\| + \varepsilon \frac{\langle \mathbf{q}_0, \mathbf{q}_\varepsilon \rangle}{\|\mathbf{q}_0\|}$　4Dベクトルとしての内積
  - 逆元　$\hat{\mathbf{q}}^{-1} = \frac{\hat{\mathbf{q}}^*}{\|\hat{\mathbf{q}}\|^2}$
  - $\|\hat{\mathbf{q}}\| = 1$ となるものを単位 dual quaternion と呼ぶ
    - $\Leftrightarrow \|\mathbf{q}_0\| = 1$ かつ $\langle \mathbf{q}_0, \mathbf{q}_\varepsilon \rangle = 0$

Geometric Skinning with Approximate Dual Quaternion Blending [Kavan TOG08]

Dual quaternion による剛体変換

- 平行移動成分が $\vec{\mathbf{t}} = (t_x, t_y, t_z)$ で、回転成分が $\mathbf{q}_0$ (単位quaternion)であるような剛体変換を表す単位 dual quaternion：

$$\hat{\mathbf{q}} = \mathbf{q}_0 + \frac{\varepsilon}{2}\vec{\mathbf{t}}\mathbf{q}_0$$

注意：3Dベクトルは、実数成分を持たないquaternionと見なす

- 単位 dual quaternion $\hat{\mathbf{q}}$ による、3D座標 $\vec{\mathbf{v}} = (v_x, v_y, v_z)$ の剛体変換：

$$\hat{\mathbf{q}}(1 + \varepsilon\vec{\mathbf{v}})\overline{\hat{\mathbf{q}}^*} = 1 + \varepsilon\overrightarrow{\mathbf{v}'}$$

  - $\overrightarrow{\mathbf{v}'}$ が変換後の3D座標

Geometric Skinning with Approximate Dual Quaternion Blending [Kavan TOG08]

Dual quaternion による剛体変換

- $\hat{\mathbf{q}} = \mathbf{q}_0 + \frac{\varepsilon}{2}\vec{\mathbf{t}}\mathbf{q}_0$
- $\hat{\mathbf{q}}(1+\varepsilon\vec{\mathbf{v}})\overline{\hat{\mathbf{q}}^*} = \left(\mathbf{q}_0 + \frac{\varepsilon}{2}\vec{\mathbf{t}}\mathbf{q}_0\right)(1+\varepsilon\vec{\mathbf{v}})\overline{\left(\mathbf{q}_0+\frac{\varepsilon}{2}\vec{\mathbf{t}}\mathbf{q}_0\right)^*}$

$$= \left(\mathbf{q}_0 + \frac{\varepsilon}{2}\vec{\mathbf{t}}\mathbf{q}_0\right)(1+\varepsilon\vec{\mathbf{v}})\overline{\left(\mathbf{q}_0^* + \frac{\varepsilon}{2}\left(\vec{\mathbf{t}}\mathbf{q}_0\right)^*\right)}$$

$$\boxed{\begin{aligned}\left(\vec{\mathbf{t}}\mathbf{q}_0\right)^* &= \mathbf{q}_0^*\left(\vec{\mathbf{t}}\right)^* \\ &= -\mathbf{q}_0^*\vec{\mathbf{t}}\end{aligned}}$$

$$= \left(\mathbf{q}_0 + \frac{\varepsilon}{2}\vec{\mathbf{t}}\mathbf{q}_0\right)(1+\varepsilon\vec{\mathbf{v}})\overline{\left(\mathbf{q}_0^* - \frac{\varepsilon}{2}\mathbf{q}_0^*\vec{\mathbf{t}}\right)}$$

$$= \left(\mathbf{q}_0 + \frac{\varepsilon}{2}\vec{\mathbf{t}}\mathbf{q}_0\right)(1+\varepsilon\vec{\mathbf{v}})\left(\mathbf{q}_0^* + \frac{\varepsilon}{2}\mathbf{q}_0^*\vec{\mathbf{t}}\right)$$

$$= \left(\mathbf{q}_0 + \frac{\varepsilon}{2}\vec{\mathbf{t}}\mathbf{q}_0\right)\left(\mathbf{q}_0^* + \varepsilon\vec{\mathbf{v}}\mathbf{q}_0^* + \frac{\varepsilon}{2}\mathbf{q}_0^*\vec{\mathbf{t}} + \cancelto{0}{\frac{\varepsilon^2}{2}\vec{\mathbf{v}}\mathbf{q}_0^*\vec{\mathbf{t}}}\right)$$

$$= \mathbf{q}_0\mathbf{q}_0^* + \frac{\varepsilon}{2}\vec{\mathbf{t}}\mathbf{q}_0\mathbf{q}_0^* + \varepsilon\mathbf{q}_0\vec{\mathbf{v}}\mathbf{q}_0^* + \cancelto{0}{\frac{\varepsilon^2}{2}\vec{\mathbf{t}}\mathbf{q}_0\vec{\mathbf{v}}\mathbf{q}_0^*} + \frac{\varepsilon}{2}\mathbf{q}_0\mathbf{q}_0^*\vec{\mathbf{t}} + \cancelto{0}{\frac{\varepsilon^2}{4}\vec{\mathbf{t}}\mathbf{q}_0\mathbf{q}_0^*\vec{\mathbf{t}}}$$

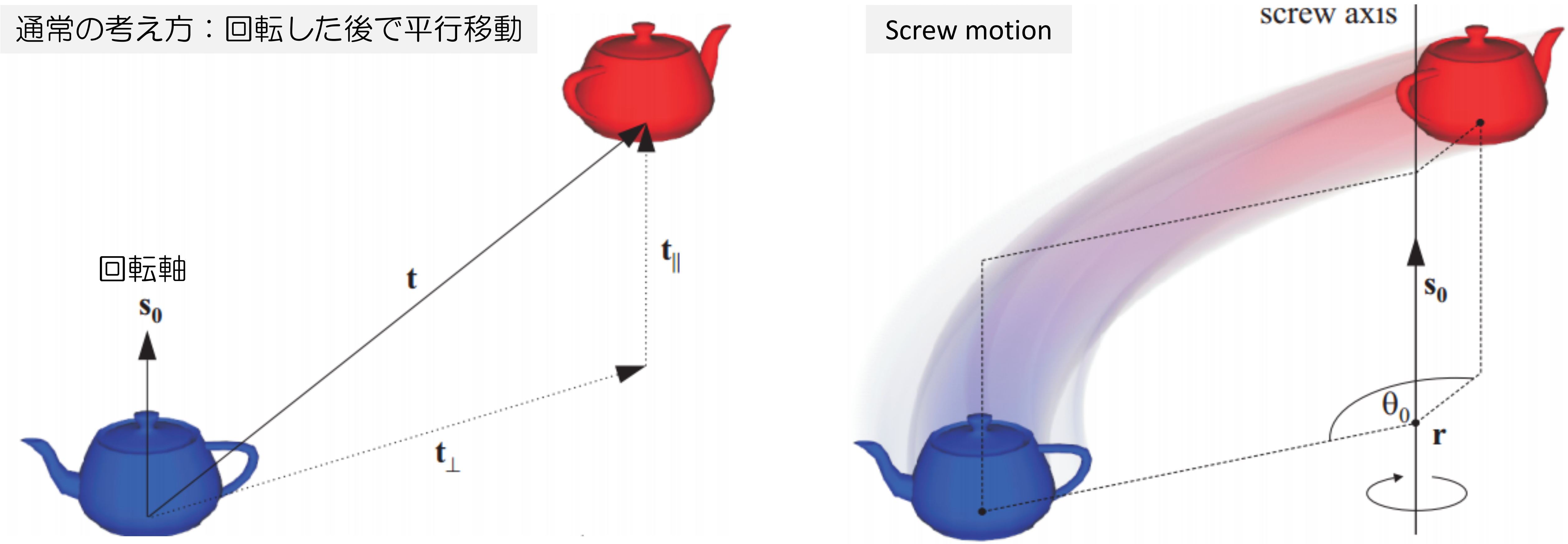
$$\boxed{\|\mathbf{q}_0\|^2 = 1}$$

$$= 1 + \varepsilon\left(\vec{\mathbf{t}} + \mathbf{q}_0\vec{\mathbf{v}}\mathbf{q}_0^*\right)$$

3D座標 $\vec{\mathbf{v}}$ をquaternion $\mathbf{q}_0$ で回転した結果

Geometric Skinning with Approximate Dual Quaternion Blending [Kavan TOG08]

"Screw motion" としての剛体運動



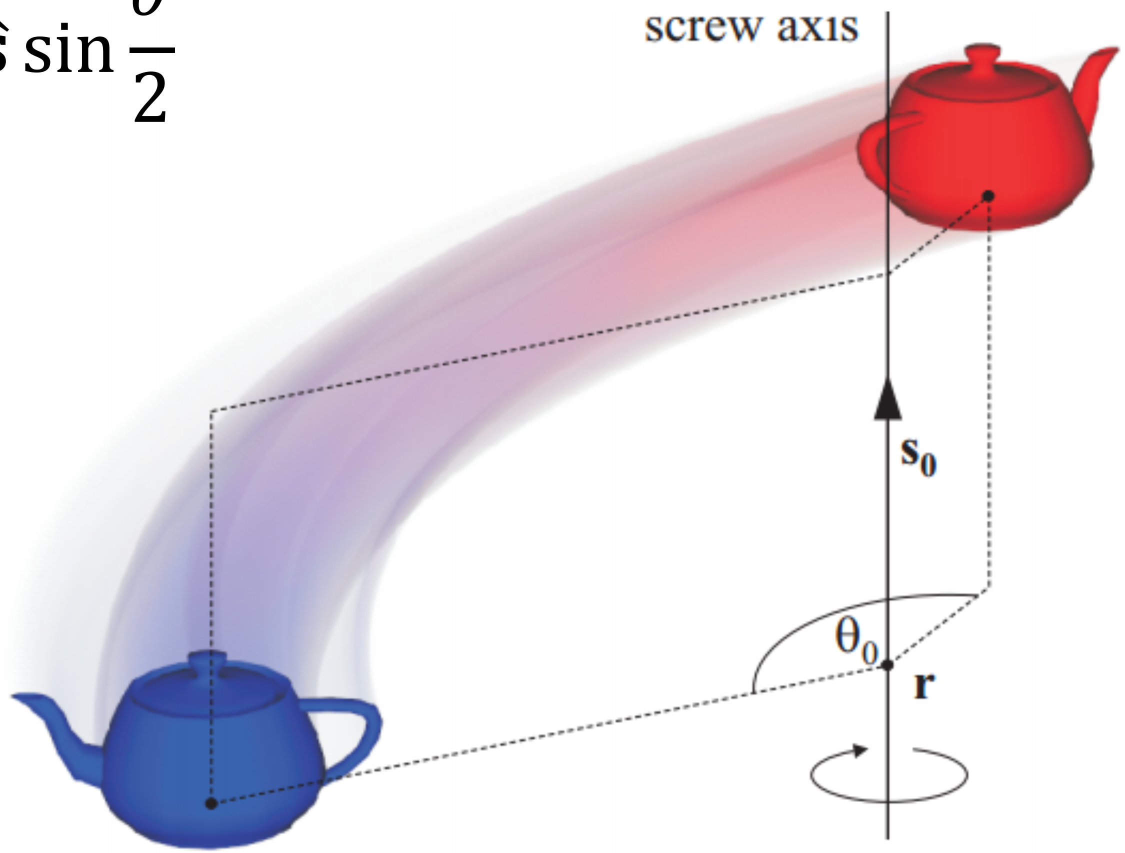
- 任意の剛体運動は、screw motion として一意に記述できる

Geometric Skinning with Approximate Dual Quaternion Blending [Kavan TOG08]

Screw motion と dual quaternion

- 単位 dual quaternion $\hat{\mathbf{q}}$ は、以下の形で表せる：
$$\hat{\mathbf{q}} = \cos\frac{\hat{\theta}}{2} + \hat{\mathbf{s}}\sin\frac{\hat{\theta}}{2}$$
  - $\hat{\theta} = \theta_0 + \varepsilon\theta_\varepsilon$　　$\theta_0, \theta_\varepsilon$：実数
  - $\hat{\mathbf{s}} = \overrightarrow{\mathbf{s}_0} + \varepsilon\overrightarrow{\mathbf{s}_\varepsilon}$　　$\overrightarrow{\mathbf{s}_0}, \overrightarrow{\mathbf{s}_\varepsilon}$：単位3Dベクトル

- 幾何的な意味
  - $\overrightarrow{\mathbf{s}_0}$：回転軸方向
  - θ_0：回転量
  - θ_ε：回転軸方向の平行移動量
  - $\overrightarrow{\mathbf{s}_\varepsilon}$：回転軸が$\vec{\mathbf{r}}$を通るとき、
    $\overrightarrow{\mathbf{s}_\varepsilon} = \vec{\mathbf{r}} \times \overrightarrow{\mathbf{s}_0}$を満たす



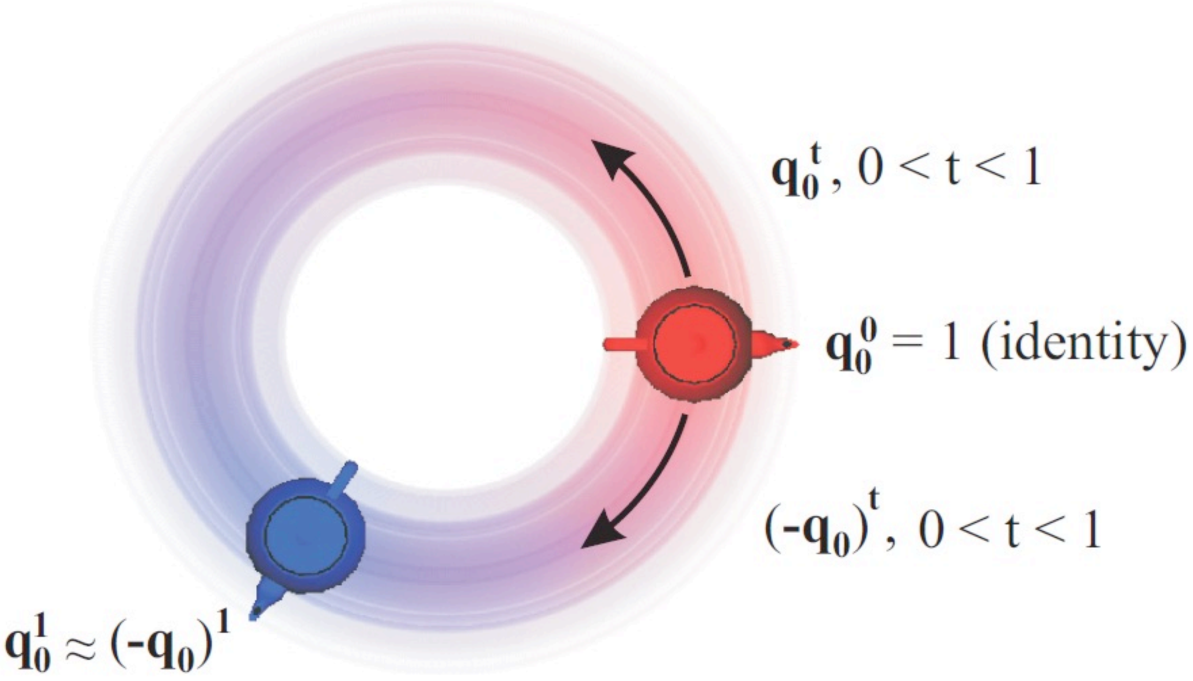
Geometric Skinning with Approximate Dual Quaternion Blending [Kavan TOG08]

二つの剛体変換の補間

- 線形補間＋正規化 (nlerp)

$$\mathrm{nlerp}(\hat{\mathbf{q}}_1, \hat{\mathbf{q}}_2, t) \coloneqq \frac{(1-t)\hat{\mathbf{q}}_1 + t\hat{\mathbf{q}}_2}{\|(1-t)\hat{\mathbf{q}}_1 + t\hat{\mathbf{q}}_2\|}$$

- 注意：$\hat{\mathbf{q}}$と$-\hat{\mathbf{q}}$は同じ剛体変換を表すが、過程が正反対
- $\hat{\mathbf{q}}_1$ と $\hat{\mathbf{q}}_2$ それぞれの non-dual な quaternion の 4D 内積が負であれば、$\hat{\mathbf{q}}_1$ の補間相手を $-\hat{\mathbf{q}}_2$ とする



Dual quaternion による剛体変換のブレンド

$$\mathrm{blend}(\langle w_1, \hat{\mathbf{q}}_1\rangle, \langle w_2, \hat{\mathbf{q}}_2\rangle, \dots) := \frac{w_1\hat{\mathbf{q}}_1 + w_2\hat{\mathbf{q}}_2 + \cdots}{\|w_1\hat{\mathbf{q}}_1 + w_2\hat{\mathbf{q}}_2 + \cdots\|}$$

- Quaternion による回転と同様
- 入力データ形式が LBS と同一、計算コスト低い
- 市販CGソフトの多くに標準装備

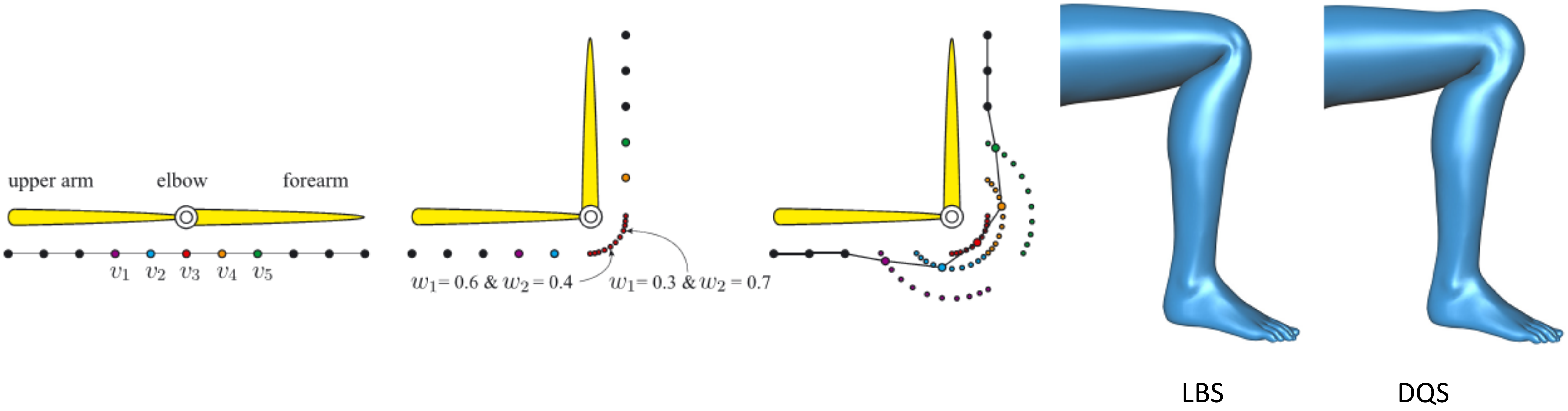
197.4 FPS

122 FPS

Geometric Skinning with Approximate Dual Quaternion Blending [Kavan TOG08]

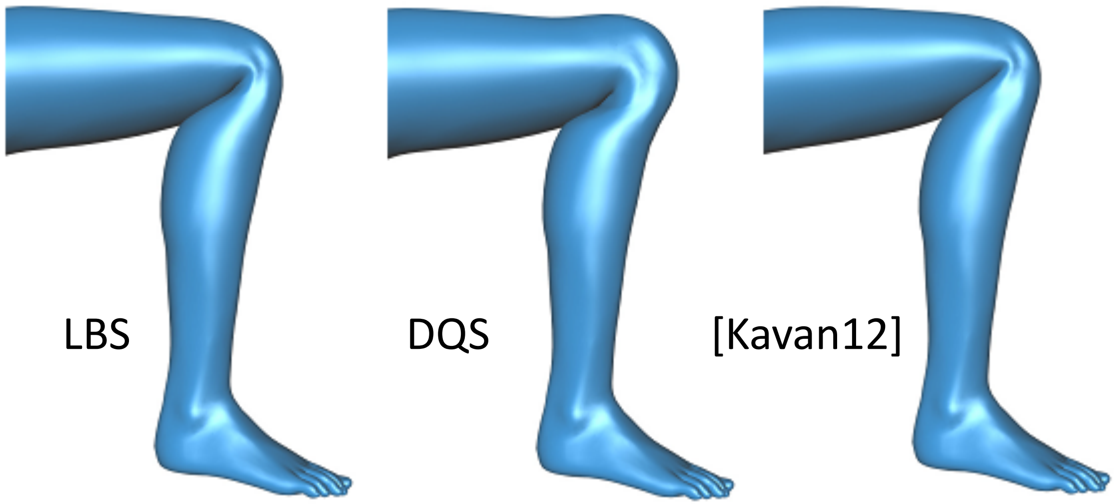
DQS の欠点："bulging" effect

- 曲げの際に、関節を中心とした球面上に沿ったような軌跡を描く

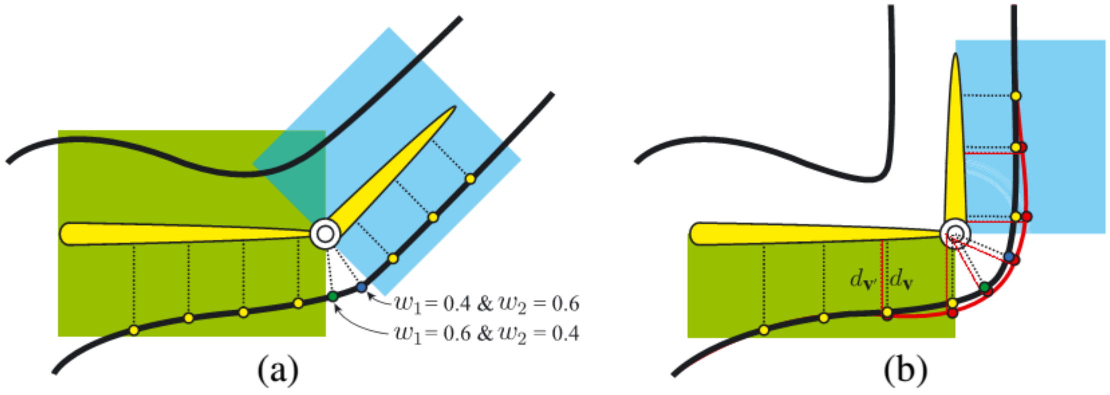


Elasticity-Inspired Deformers for Character Articulation [Kavan SIGGRAPHAsia12]
Bulging-free dual quaternion skinning [Kim CASA14]

DQS の欠点の克服



変換を bend と twist に分解し、別々に補間 [Kavan12]



DQS で動かした後、法線方向にオフセット [Kim14]

Elasticity-Inspired Deformers for Character Articulation [Kavan SIGGRAPHAsia12]
Bulging-free dual quaternion skinning [Kim CASA14]

自己交差を回避するスキニング

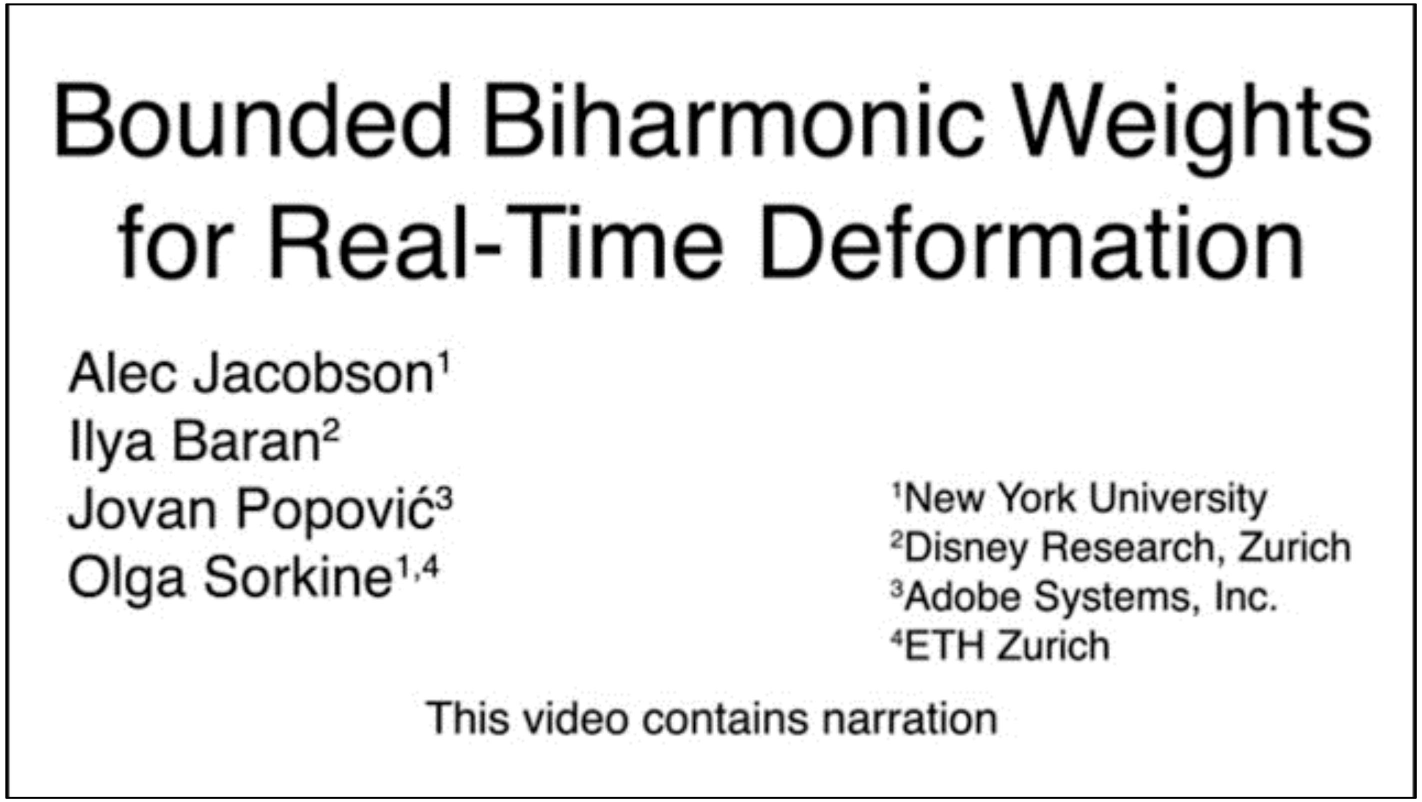
- 陰関数の性質を活用

https://www.youtube.com/watch?v=RHySGIqEgyk

Implicit Skinning; Real-Time Skin Deformation with Contact Modeling [Vaillant SIGGRAPH13]

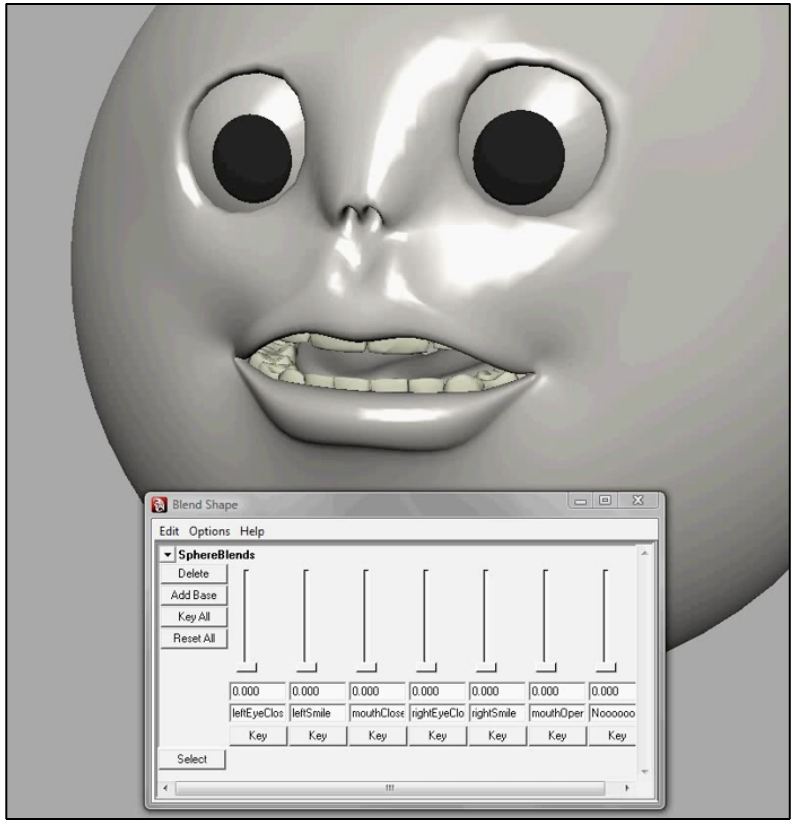
スケルトン以外の変形インタフェース

点、ケージ、スケルトンの統合 [Jacobson 11]



https://www.youtube.com/watch?v=P9fqm8vgdB8

BlendShape



https://www.youtube.com/watch?v=hb5zH4V37r8

Bounded Biharmonic Weights for Real-Time Deformation [Jacobson SIGGRAPH11]

参考情報

- http://en.wikipedia.org/wiki/Motion_capture
- http://skinning.org/
- http://mukai-lab.org/category/library/legacy
- CG Gems JP 2012 Chapter 8 インバースキネマティクス