Geometry Processing

6 Deformation

Ludwig-Maximilians-Universität München

Announcements

- Guest Talk: Industrial Modeling (scheduled at 22.02.2021, will send email for further information)
    - From WAY Engineering Group
    - Good opportunities to learn about real-world industrial level meshes (large and messy geometric data)
    - Connection and maybe intern
- No more coding projects, just focus on polish your proposed individual project :)
    - But still open for submission, but no template provided
- Register to the "exam" (project presentation, just for reporting the grades to examination office)
    - Bachelor: https://uni2work.ifi.lmu.de/course/W20/IfI/GP/exams/GP-Exam/show
    - Master: https://uni2work.ifi.lmu.de/course/W20/IfI/PGP/exams/PGP-Exam/show

Session 6: Deformation

- Motivation
- Surface Deformation
- Space Deformation
- Skinning
- Summary

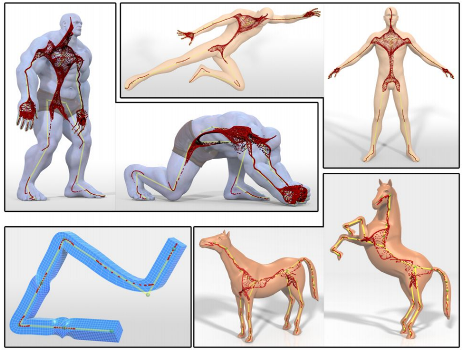
Large Existing Applications

Character animations

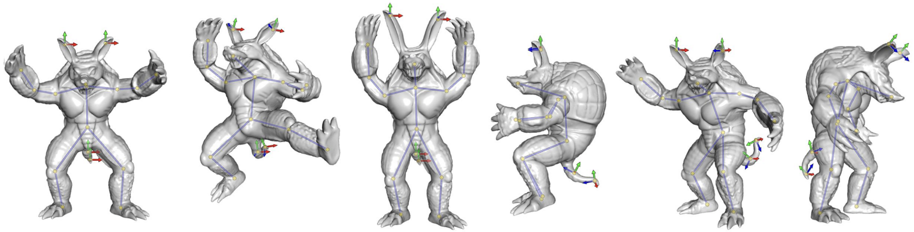
Image editing

"Memoji"

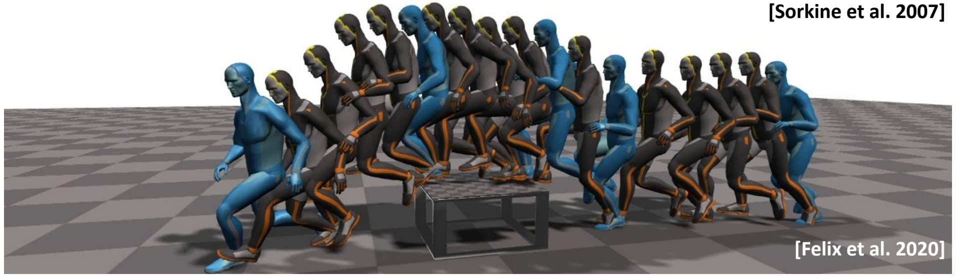
...



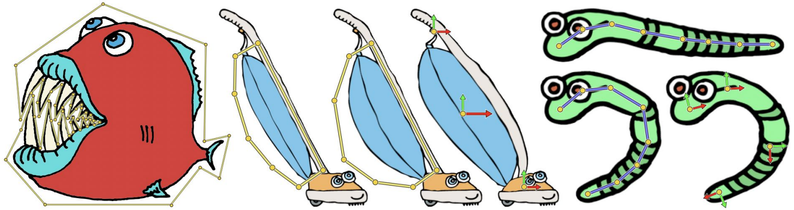
[Le et al.2016]



[Sorkine et al. 2007]



[Felix et al. 2020]



[Jacobson et al. 2011]

Image from Apple Inc.

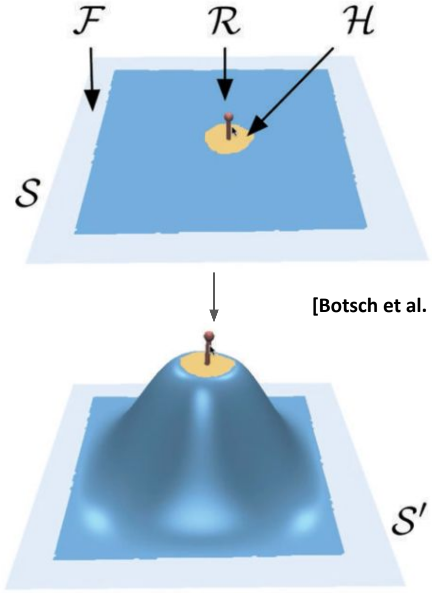
Problem Settings

The deformation of a given surface $\mathcal{S}$ into the desired surface $\mathcal{S}'$

- A displacement function $\mathbf{d}(\mathbf{p})$ on each vertex $\mathbf{p} \in \mathcal{S}$
- Desired surface is determined by the displacement $\mathcal{S}' = \{\mathbf{p} + \mathbf{d}(\mathbf{p}) | \mathbf{p} \in \mathcal{S}\}$

User inputs (constraints):

- Handle region $\mathcal{H}$ such that $\mathbf{d}(\mathbf{p}_i) = \bar{\mathbf{d}}_i, \forall \mathbf{p}_i \in \mathcal{H}$
- Fixed region $\mathcal{F}$ such that $\mathbf{d}(\mathbf{p}_i) = 0, \forall \mathbf{p}_i \in \mathcal{F}$



[Botsch et al. 2008]

Optimization goal: **Determine the displacement for remaining region**

$$\mathbf{d}(\mathbf{p}_i), \forall \mathbf{p}_i \in \mathcal{R} = \mathcal{S} \setminus (\mathcal{H} \cup \mathcal{F})$$

Approaches

Surface deformation

- Shape is an empty shell: Curve for 2D and surface for 3D deformation
- Deformation only defined on the shape
- Deformation coupled with shape representation

Space deformation

- Shape is volumetric: Planar domain in 2D and Polyhedral domain in 3D
- Deformation defined in neighborhood of shape
- Can be applied to any shape representation

Session 6: Deformation

- Motivation
- **Surface Deformation**
- Space Deformation
- Skinning
- Summary

Transformation Propagation

Basic Idea: Smooth blending between the transformed handle and the fixed region

Controlled by a scalar field:

- Smoothly blends between 0 (fixed region) and 1 (the handle)

Distance function:

$d_{\mathcal{F}}(\mathbf{p})$ *distance* from vertex $\mathbf{p}$ to the *fixed region*

$d_{\mathcal{H}}(\mathbf{p})$ *distance* from vertex $\mathbf{p}$ to the *handle region*

Scalar field:

$$s(\mathbf{p}) = \frac{d_{\mathcal{F}}(\mathbf{p})}{d_{\mathcal{F}}(\mathbf{p}) + d_{\mathcal{H}}(\mathbf{p})}$$

[Botsch et al. 2008]

Transformation Propagation (2) Smooth Blending

Recall Laplacian smooth, consider the scalar field as the harmonic field on the surface, we have:

$$\Delta s(\mathbf{p}_i) = 0, \mathbf{p}_i \in \mathcal{R}$$
$$s(\mathbf{p}_i) = 1, \mathbf{p}_i \in \mathcal{H}$$
$$s(\mathbf{p}_i) = 0, \mathbf{p}_i \in \mathcal{F}$$

Replace the Laplacian via Laplace-Beltrami operator, that turns into a linear system.

The resulting scalar field is used to damp the transformation of the handle for each vertex in the remaining region:

$$\mathbf{p}_i' = s(\mathbf{p}_i)\mathbf{T}_{\mathcal{H}}(\mathbf{p}_i) + (1 - s(\mathbf{p}_i))\mathbf{p}_i$$

Handle Transformation

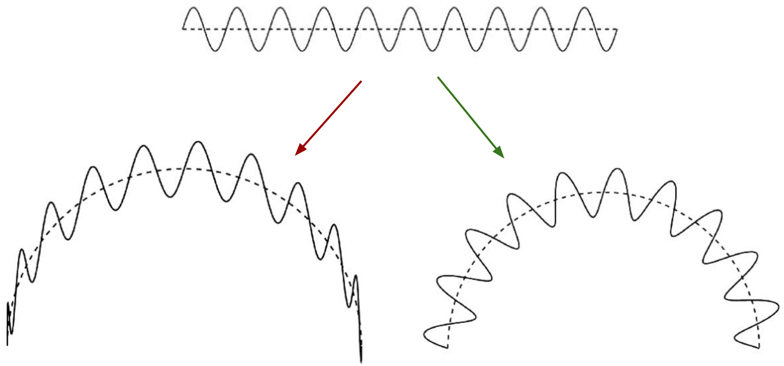
Easy to implement but typically not result in geometrically intuitive solution.

Multi-Scale Deformation

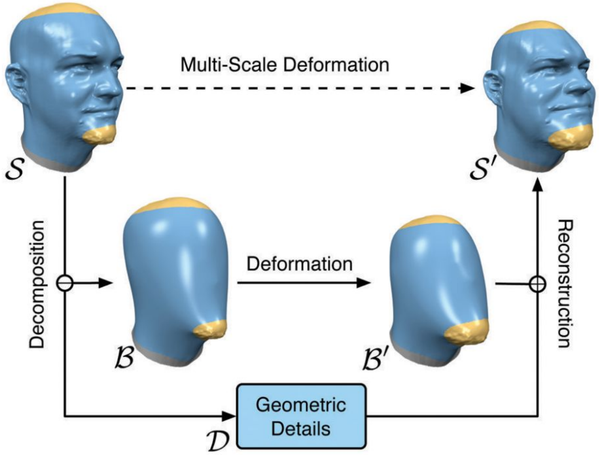
Basic concept: decompose the object into multiple frequency and deform low frequencies (global shape) while *preserving the high frequency details*

- Low frequencies correspond to the smooth global shape
- High frequencies correspond to the fine-scale details

Not a specific approach but a general framework of doing deformation tasks



displacement with regard to global coordinate system and local tangent plane



[Botsch et al. 2008]

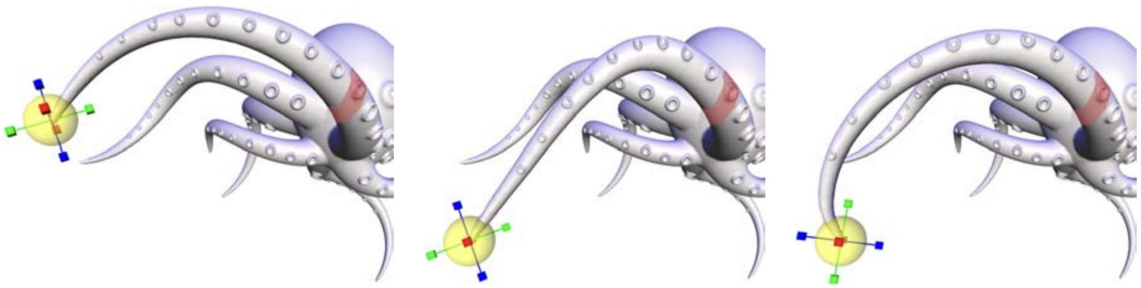
Laplacian Editing **[Sorkine et al. 2004]**

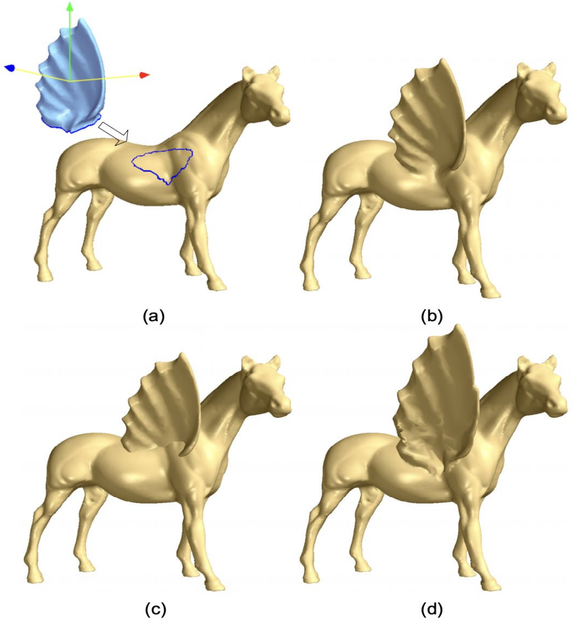
Manipulate per-vertex Laplacian

1. Compute initial Laplacian (scalar)

2. Manipulate Laplacian coordinates (local transformation)

3. Find new coordinates that match the target Laplacian coordinates





Similar approach uses gradient in Poisson gradient mesh editing **[Yu et al. 2004]**

As-Rigid-As-Possible Deformation (ARAP) [Sorkine et al. 2007]

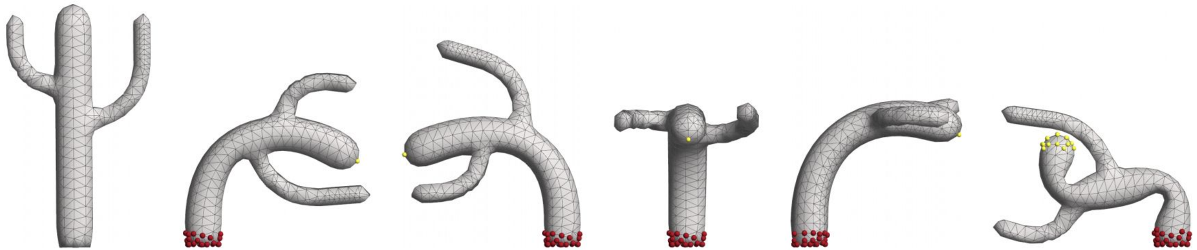
Basic idea: deformed object should only apply rotation and translation (rigid), no scaling and shearing.

Define and minimize energy function:

$$\mathcal{E} = \sum_{i=1}^{N} A_i \sum_{j} w_{ij} ||(\mathbf{p}_i' - \mathbf{p}_j') - \mathbf{R}_i(\mathbf{p}_i - \mathbf{p}_j)||$$

all verts — one ring — rotation matrix

Use alternating minimization technique (EM algorithm in machine learning) to minimize the energy.



Handle and Fixed Region

Interactive Deformation

Deformation often has higher demand on interactivity, so that either designer can iterating their idea quickly or renderer can manipulate geometry in real-time.

Implementation thinking:

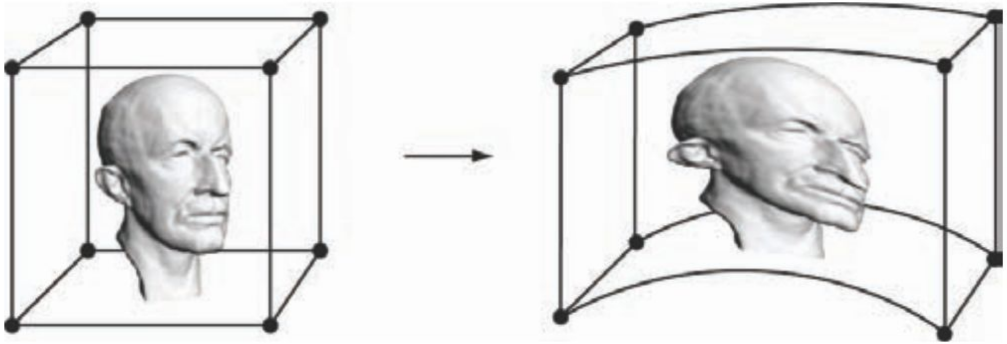
- "Interaction cost": Designing minimum user inputs (select handle and fixed region, drag handle)
- "Intuitive deformation": Transformation works as "expected", global deformation that preserves local details

Session 6: Deformation

- Motivation
- Surface Deformation
- Space Deformation
- Skinning
- Summary

Space-based Deformation

Space-based deformations (meshless mapping) deforms the ambient space and thus implicitly deform the embedded objects.



Two classic approaches

- Lattice-based freeform deformation
- Cage-based freeform deformation

Lattice-based Freeform Deformation

Space deformation represented by a trivariate tensor-product spline function

$$\mathbf{d}(\mathbf{u}) = \sum_{l=1}^{n} \delta \mathbf{c}_l N_l(\mathbf{u})$$

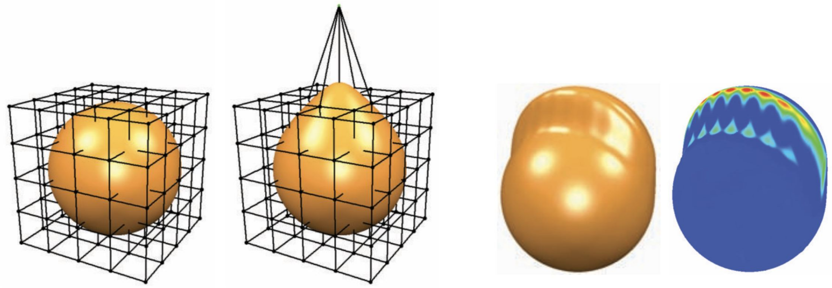
Control points　　B-spline basis

Each original vertex $\mathbf{p}_i \in \mathcal{S}$ has a corresponding parameter value $\mathbf{u}_i = (u_i, v_i, w_i)$

such that $\mathbf{p}_i = \sum_l \mathbf{c}_l N_l(\mathbf{u}_i)$

The deformation can be consider as the transformation of vertices: $\mathbf{p}_i' = \mathbf{p}_i + \mathbf{d}(\mathbf{u}_i)$

The remaining problem is to solve the displacement function.



*artifacts

Cage-based Freeform Deformation

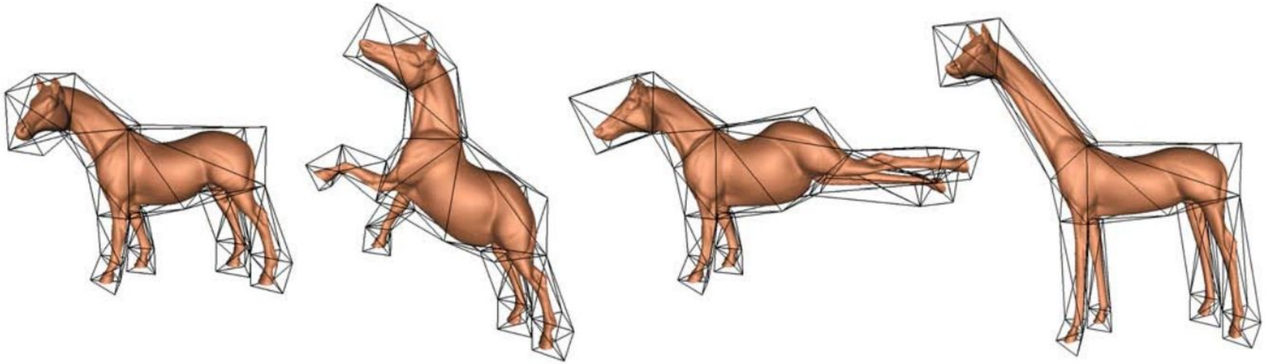
A generalization of the lattice-based freeform deformation

Control cages are typically coarse, arbitrary triangle mesh enclosing the object to be modified

The vertices $\mathbf{p}_i$ of original mesh $\mathcal{S}$ can be represented as linear combinations of the cage's control vertices $\mathbf{c}_l$ by

$$\mathbf{p}_i = \sum_{l=1}^{n} \mathbf{c}_l \varphi_l(\mathbf{p}_i)$$

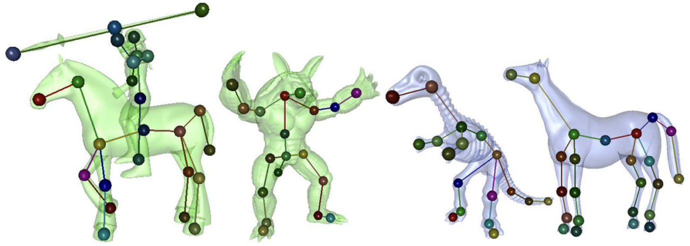
where $\varphi_l(\mathbf{p}_i)$ are *generalized barycentric coordinates*, such as mean value coordinates.



Session 6: Deformation

- Motivation
- Surface Deformation
- Space Deformation
- Skinning
- Summary

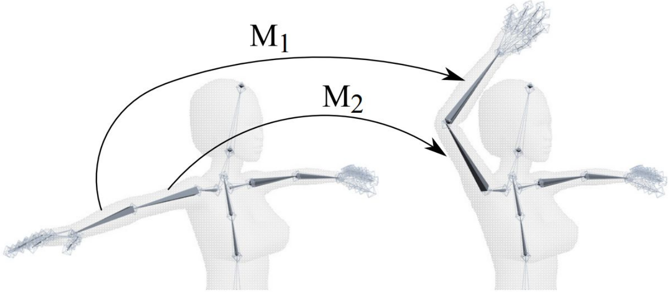
Linear Blend Skinning



Direct skeletal shape deformation

Input data assumption

- **Rest pose shape**: original undeformed polygon mesh
- **Bone transformation**: a list of transformation matrices
- **Skinning weights**: amount of influence of a bone on a vertex



$$\mathbf{v}_i' = \left(\sum_{j=1}^{m} w_{ij} \mathbf{T}_j \right) \mathbf{v}_i$$

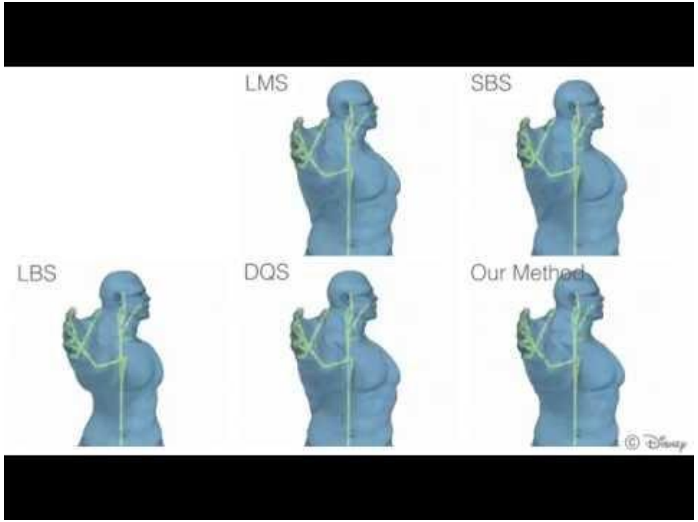
Similar terminologies: skeleton-*subspace* deformation, (single-weight-)enveloping, matrix-palette skinning

LBS and More

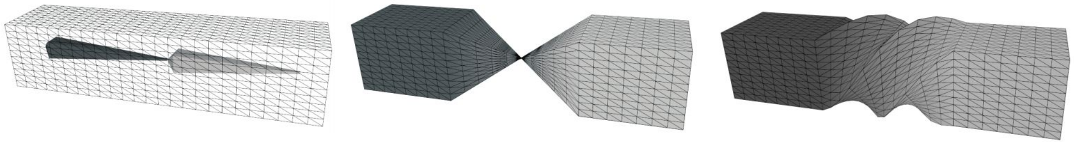
Multilinear skinning

Nonlinear skinning

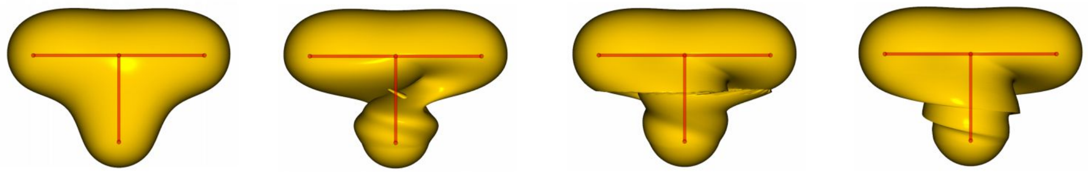
...



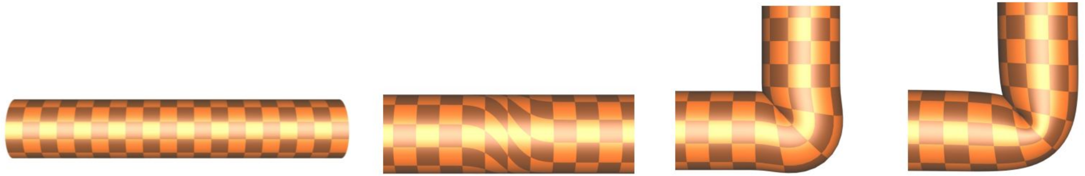
CoRs [Le et al. 2016]



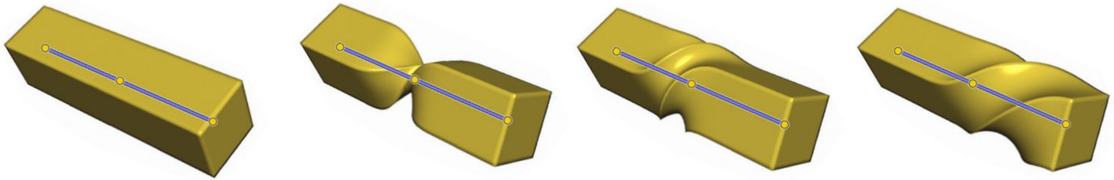
Rest Pose | LBS | Dual quaternion skinning



Rest Pose | LBS | Dual quaternion skinning (DQS) | Differential blending



Rest Pose | DQS: twist | DQS: bend | Elasticity-inspired deformers

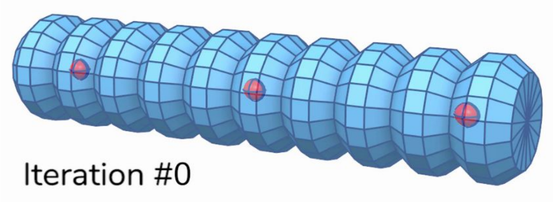


Rest Pose | LBS | DQS | Stretchable-twistable bones

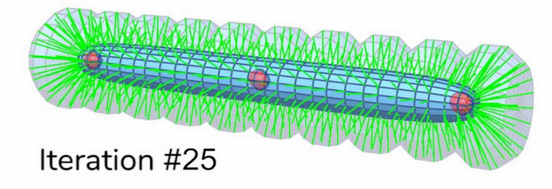
[Kavan 2014]

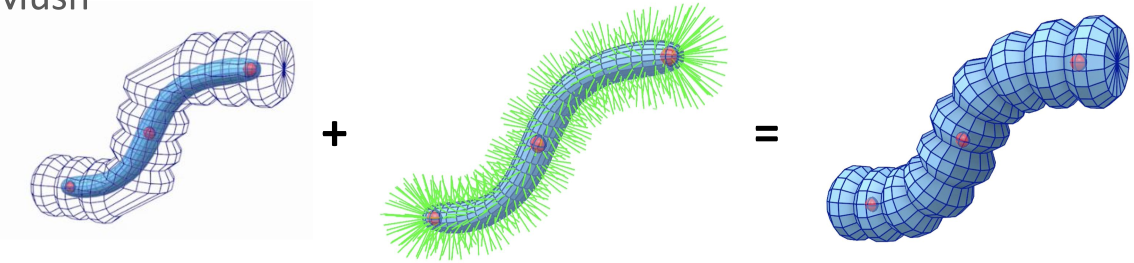
Delta Mush (DM) [Mancewicz et al. 2014]

- Rigid binding using global large-scale solver
- Mush = Smoothing
  - Laplacian smooth
  - Shrink geometry
  - Lose surface detail
- Delta = Rest Pose - Rest Pose Mush
  - Encode surface's details as displacements
  - Stored at local frame defined by the mush
  - Pre-computed once
- Delta Mush = Delta + Deformed Mush



Laplacian smooth



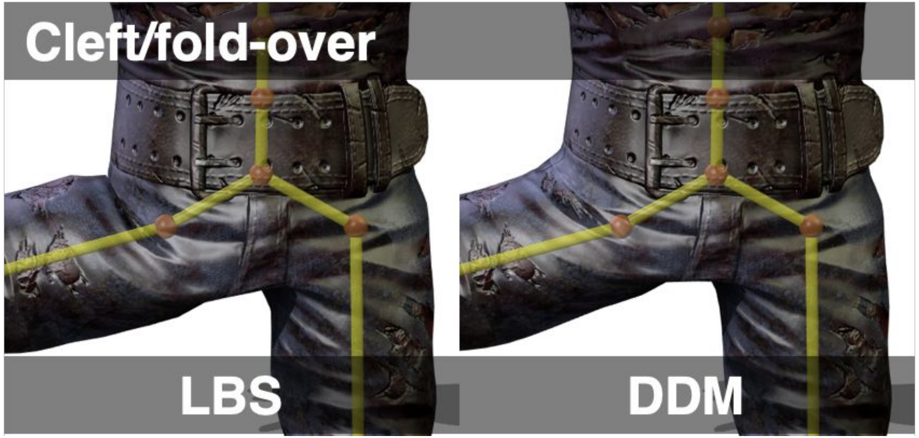


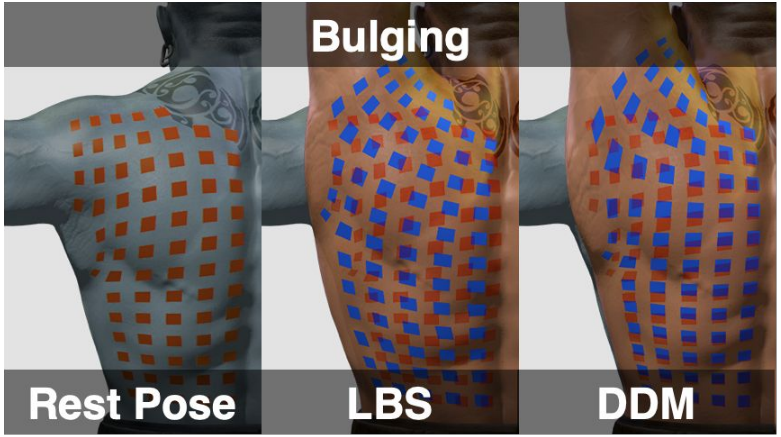
Direct Delta Mush [Le et al. 2019]

Real-time acceleration via

- Direct computation (runtime)
- Pre-computation (once)

Delta Mush like quality for easy authoring without cleft and bulging





Session 6: Deformation

- Motivation
- Surface Deformation
- Space Deformation
- Skinning
- Summary

Summary

- Deformation is a surface to surface mapping process
- Surface-based approach applies directly on surface and offer precise control on surface
- Space-based approach interpolates space and mostly meshless
- Skinning combines surface-based and space-based approaches that enable more flexibility of authoring in character modeling process

Homework 6: Delta Mush (optional)

Implement the Delta Mush in **halfedge representation**.

1. Compute Laplacian smooth on rest pose
2. Compute delta of vertices
3. Apply bone transformation
4. Apply Laplacian smooth on transformed object
5. Recover vertex positions

Implementation details on handling user inputs:

1. Create skeleton on rest pose
2. Interactive bone transformation controller

Further Readings: Deformation (1)

[Pauly et al. 2003] Pauly, Mark, et al. "Shape modeling with point-sampled geometry." ACM Transactions on Graphics (TOG) 22.3 (2003): 641-650.

[Sorkine et al. 2004] Sorkine, Olga, et al. "Laplacian surface editing." Proceedings of the 2004 Eurographics/ACM SIGGRAPH symposium on Geometry processing. 2004.

[Yu et al. 2004] Yu, Yizhou, et al. "Mesh editing with poisson-based gradient field manipulation." ACM SIGGRAPH 2004 Papers. 2004. 644-651.

[Sorkine et al. 2007] Sorkine, Olga, and Marc Alexa. "As-rigid-as-possible surface modeling." Symposium on Geometry processing. Vol. 4. 2007.

[Botsch et al. 2008] Botsch, Mario, and Olga Sorkine. "On linear variational surface deformation methods." IEEE transactions on visualization and computer graphics 14.1 (2008): 213-230.

[Mancewicz et al. 2014] Mancewicz, Joe, et al. "Delta Mush: smoothing deformations while preserving detail." Proceedings of the Fourth Symposium on Digital Production. 2014.

[Le et al. 2019] Le BH, Lewis JP. "Direct delta mush skinning and variants." ACM Trans. Graph.. 2019 Jul 12.

Further Readings: Deformation (2)

[Ju et al. 2005] Ju, Tao, Scott Schaefer, and Joe Warren. "Mean value coordinates for closed triangular meshes." ACM SIGGRAPH 2005 Papers. 2005. 561-566.

[Jacobson et al. 2011] Jacobson, Alec, et al. "Bounded biharmonic weights for real-time deformation." ACM Trans. Graph. (TOG) 30.4 (2011): 78.

[Zhang et al. 2014] Zhang, Juyong, et al. "Local barycentric coordinates." ACM Trans. Graph. (TOG) 33.6 (2014): 1-12.

[Kavan 2014] Kavan, Ladislav. "Direct Skinning Methods and Deformation Primitives." ACM SIGGRAPH 2014. 2014

[Wang et al. 2015] Wang, Yu, et al. "Linear subspace design for real-time shape deformation." ACM Trans. Graph. (TOG) 34.4 (2015): 1-11.

[Le et al. 2016] Le, Binh Huy, and Jessica K. Hodgins. "Real-time skeletal skinning with optimized centers of rotation." ACM Trans. Graph. (TOG) 35.4 (2016): 1-10.

[Felix et al. 2020] Harvey, Félix G., et al. "Robust motion in-betweening." ACM Trans. Graph. (TOG) 39.4 (2020): 60-1.

Thanks! What are your questions?

Next session: Data-driven Approach