Skinning Mesh Animations

Doug L. James, Christopher D. Twigg
Carnegie Mellon University

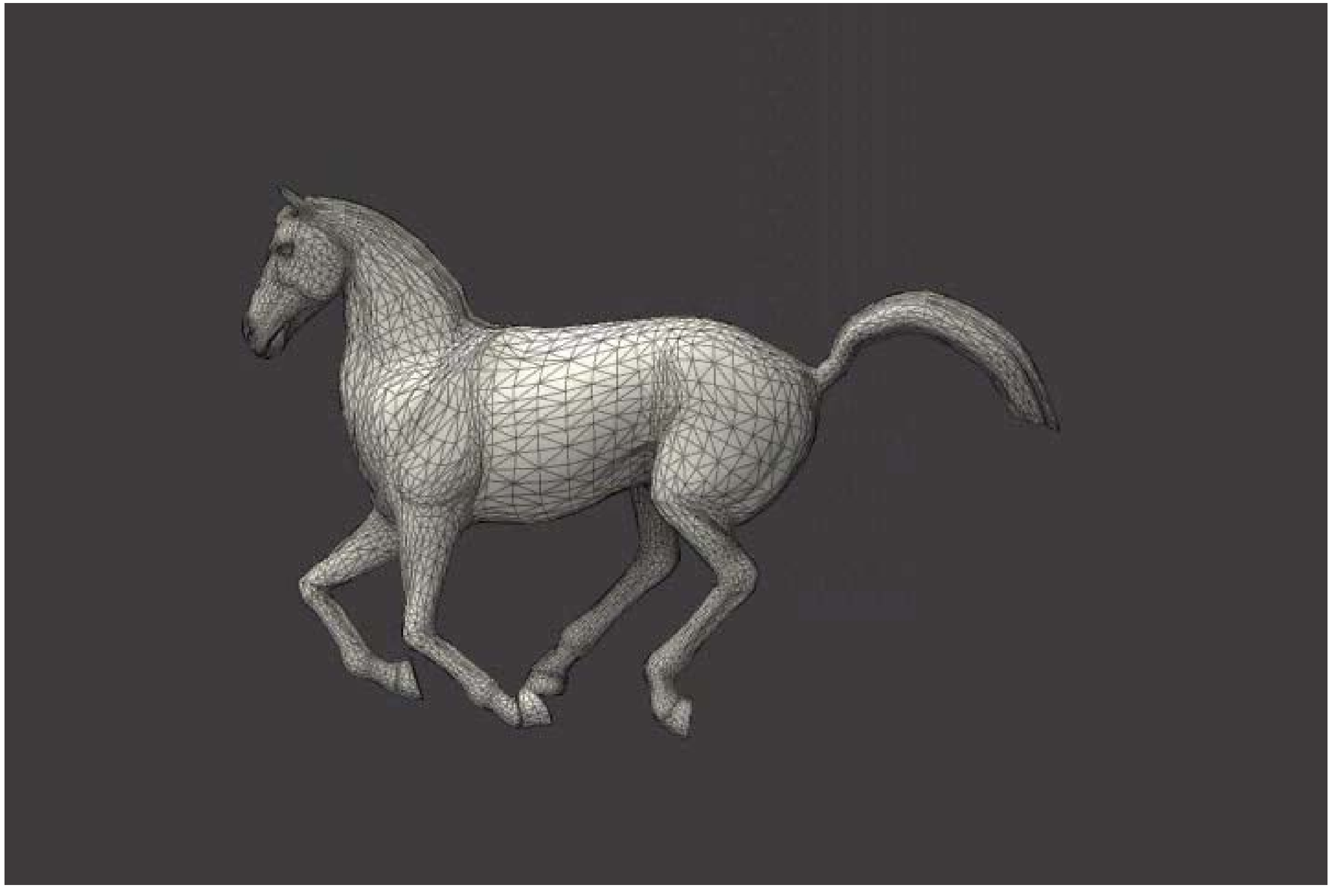
presented by Johannes Schmid

Outline

- Introduction & Motivation
- Overview & Details
- Results
- Discussion

Introduction

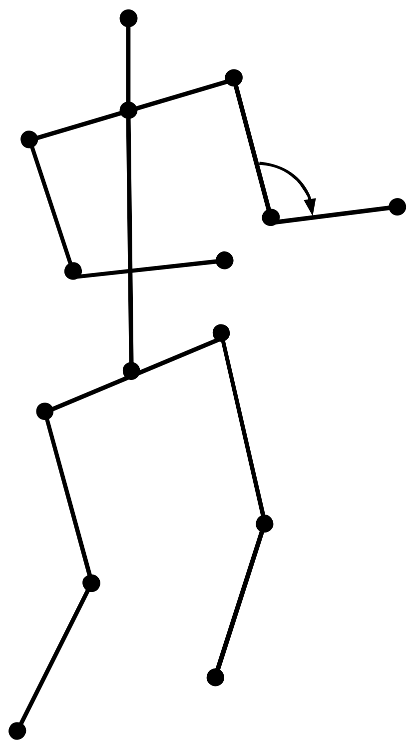
- Mesh sequence:

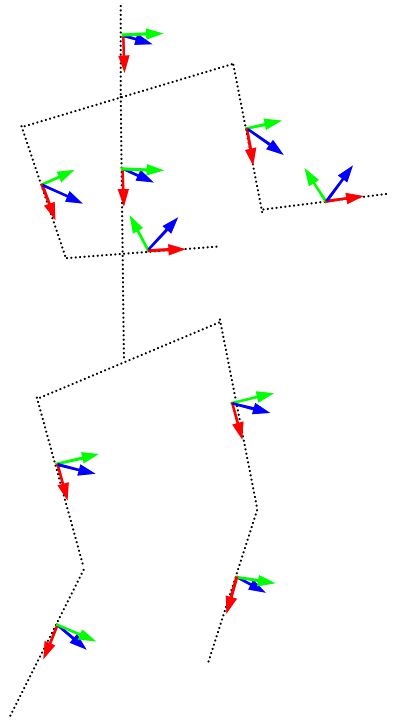


General setting

- Animation simplification
  - Direct data reduction (PCA, …)
  - Skeleton subspace deformation (SSD)

General setting





Traditional skeleton:
- Hierarchy
- Joints
- Angles

This paper‘s approach:
- No hierarchy
- Only transformations

Premises

- Mesh sequence:
  - $\mathbf{P} = (p^1, p^2, \ldots, p^S)$
    - $t = 1..S$: „time“
    - p^t: All vertices at step t
- Rest (or reference) pose: $\tilde{p}$

Goal

- Linear Blend Skinning:
  - $\mathsf{p}^t \approx \mathsf{T}^t \tilde{\mathsf{p}}$
  - $\mathsf{T}_i^t = \sum_{b \in \mathscr{B}_i} w_{ib} \bar{\mathsf{T}}_b^t$

vertex weights

bone transforms

vertex-bone influences

- We are looking for $\bar{\mathsf{T}}_b^t$, $\mathscr{B}_i$ and w_{ib}

Overview

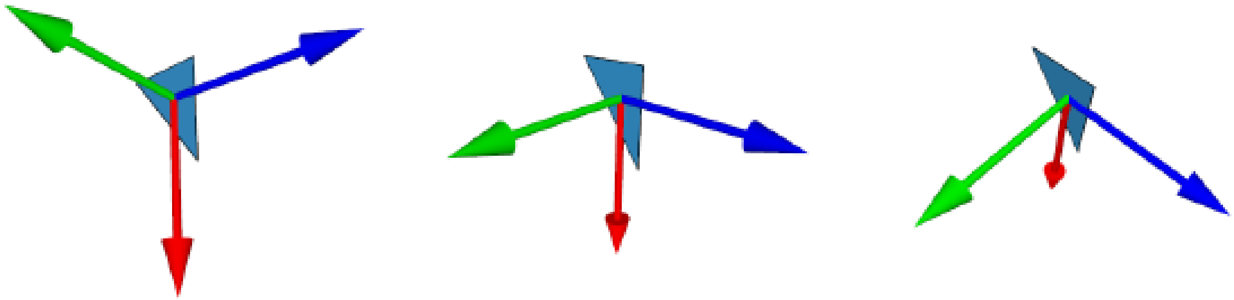
- Identify near-rigid structures
- Estimate bone transforms
- Estimate vertex weights
- Progressive skin corrections

Overview

- **Identify near-rigid structures**
- Estimate bone transforms
- Estimate vertex weights
- Progressive skin corrections

Identify structures

- Triangle rotation sequence:

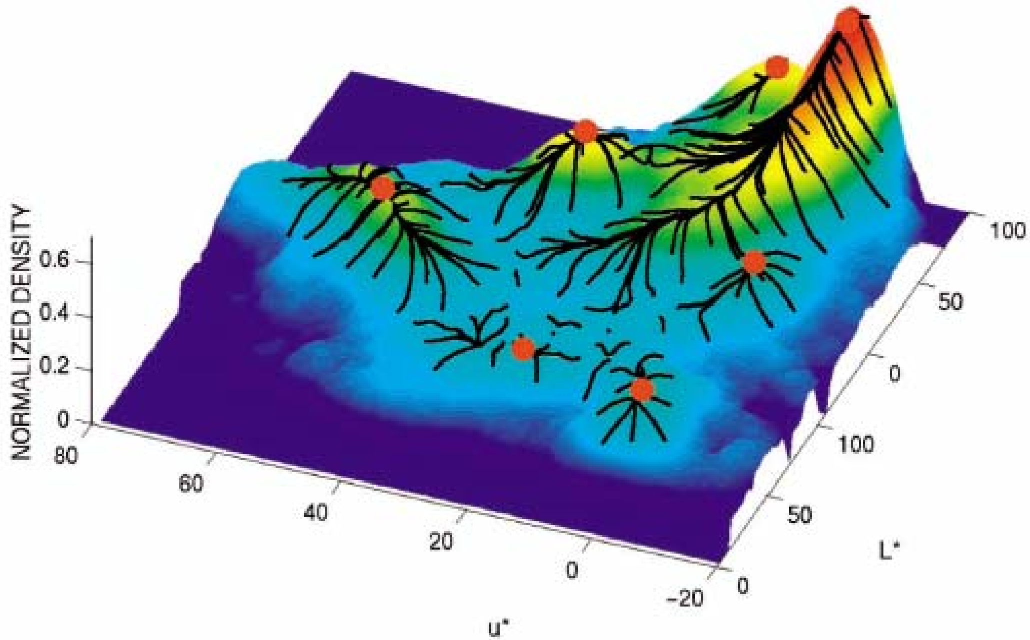


- Polar Decomposition: $\mathbf{F} = \mathbf{R}\mathbf{W}$

Rotation Stretch

Identify structures

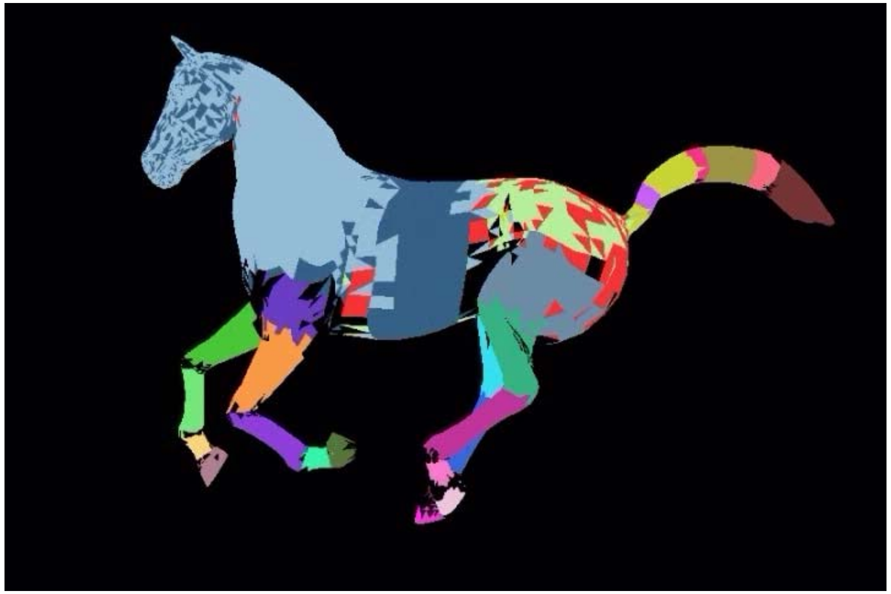
- $\mathbf{z}_j = \left(\mathrm{vec}(R_j^1),\ \ldots,\ \mathrm{vec}(R_j^S)\right)$
- Mean shift clustering over z_j

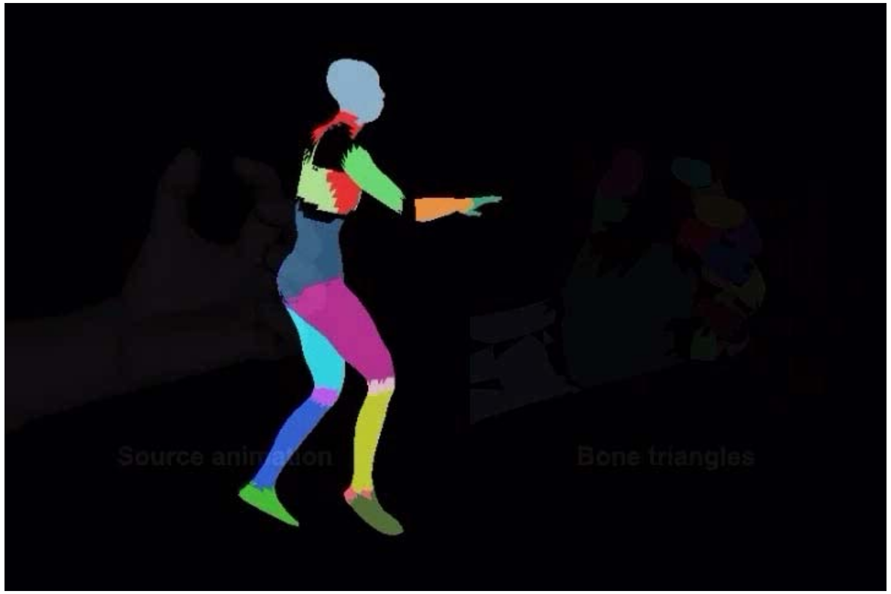


Identify structures

- Triangles within ε of found modes/bones are core triangles
  - Strongly associated with bone
  - Make up near-rigid structure
  - Total fraction of core triangles determines quality

Identify structures





Overview

- Identify near-rigid structures
- **Estimate bone transforms**
- Estimate vertex weights
- Progressive skin corrections

Estimate bone transforms

- Rigid bones
  - Rotation and translation
  - Take average rotation of core triangles
    ⇨ Arithmetic mean of triangle rotation mat.
  - Find translational part by fitting to the core triangle centroids

Estimate bone transforms

- Flexing bones
  - Bones can stretch and shear (not bend)
  - Rigid bones: rotation/translation pair $(\mathbf{R},\mathbf{v})$
  - Flexing bones: $(\mathbf{F},\mathbf{v})$ with $\mathbf{F} = \mathbf{RW}$
  - LS fit to match motion of core triangle

Overview

- Identify near-rigid structures
- Estimate bone transforms
- **Estimate vertex weights**
- Progressive skin corrections

Estimate mesh skin

- Remember: $\mathsf{T}_i^t = \sum_{b \in \mathcal{B}_i} w_{ib} \bar{\mathsf{T}}_b^t$
- Find $\mathcal{B}_i$:
  - Find β bone transforms ($\bar{\mathsf{T}}_b^t$) which individually lead to the best possible result

Estimate mesh skin

- Find vertex weights w_{ib}:
  - Match transformed vertices to mesh sequence

$$\sum_{b \in \mathscr{B}_i} (\bar{\mathsf{T}}_b^t \tilde{\mathsf{p}}_i) w_{ib} = \mathsf{p}_i^t, \qquad t = 1 \ldots S$$

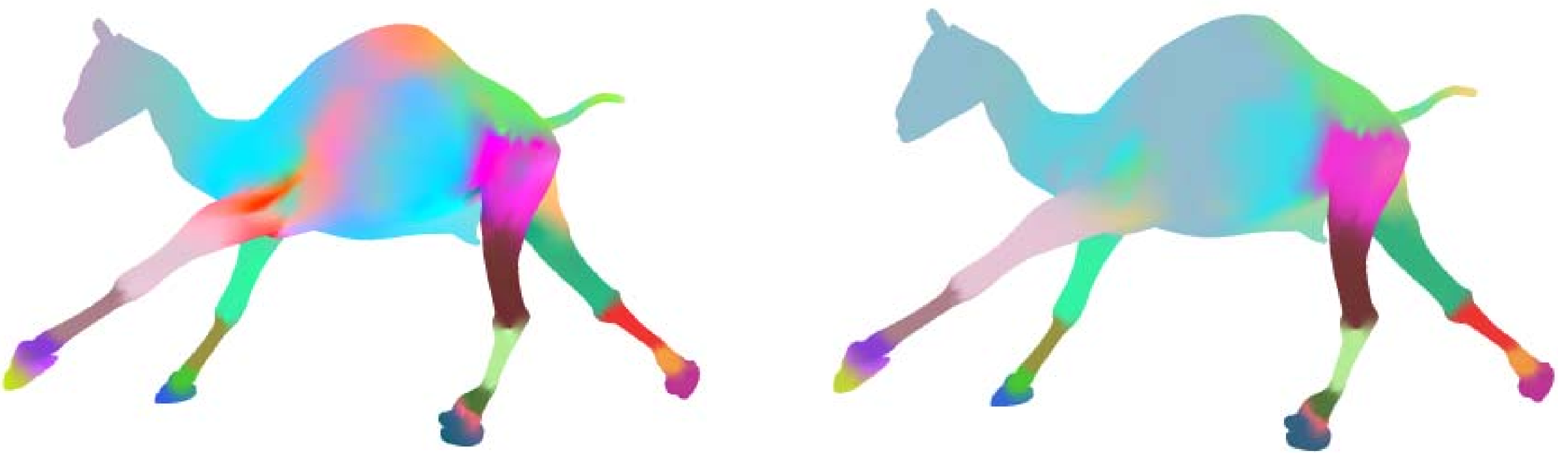
  - Weights must sum up to 1

$$\sum_b w_{ib} = 1$$

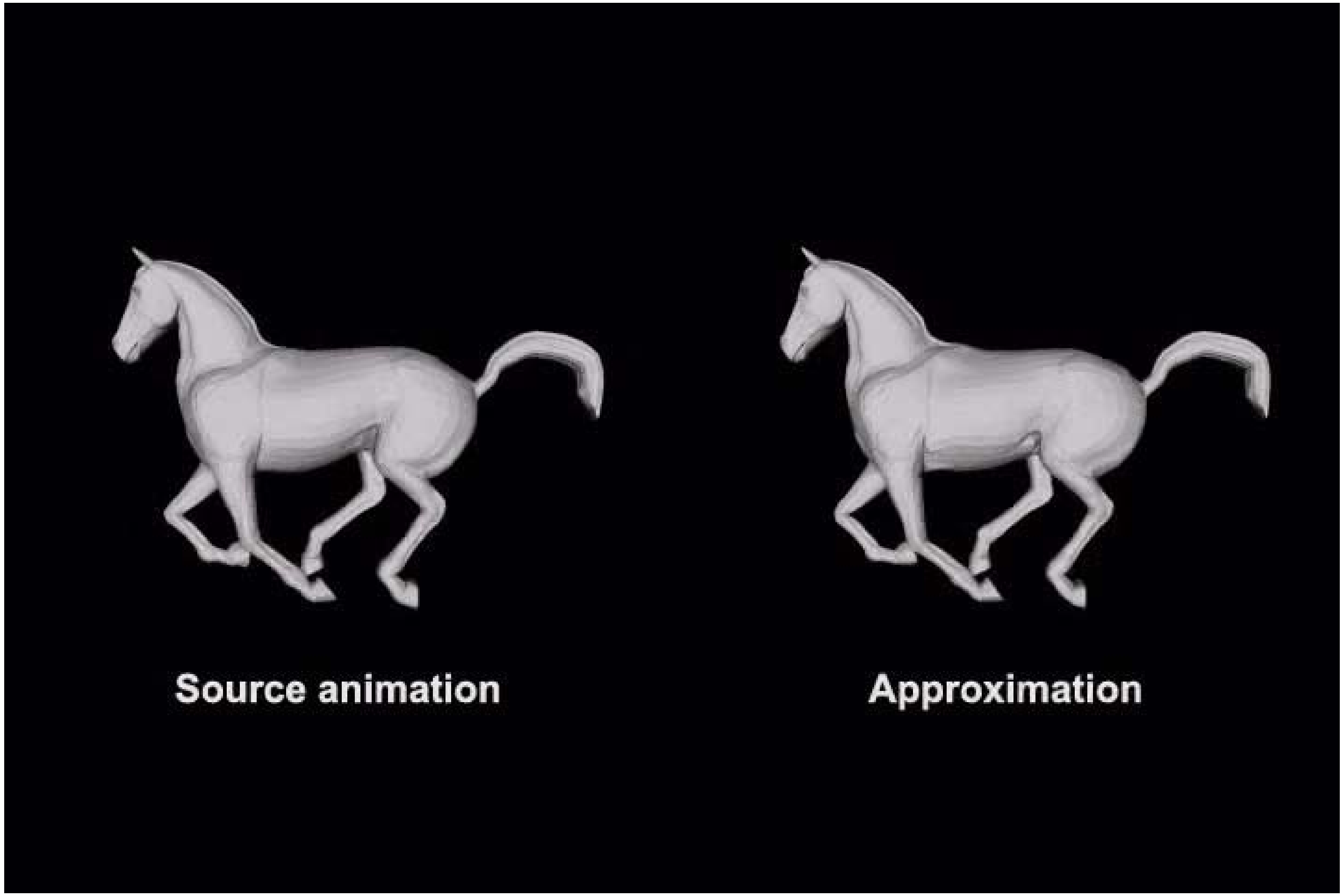
Estimate mesh skin

Estimate mesh skin

- Weight over-fitting
  - Negative weights lead to unstable skins
  - Use nonnegative least squares (NNLS) to obtain strictly positive weights



Almost there...

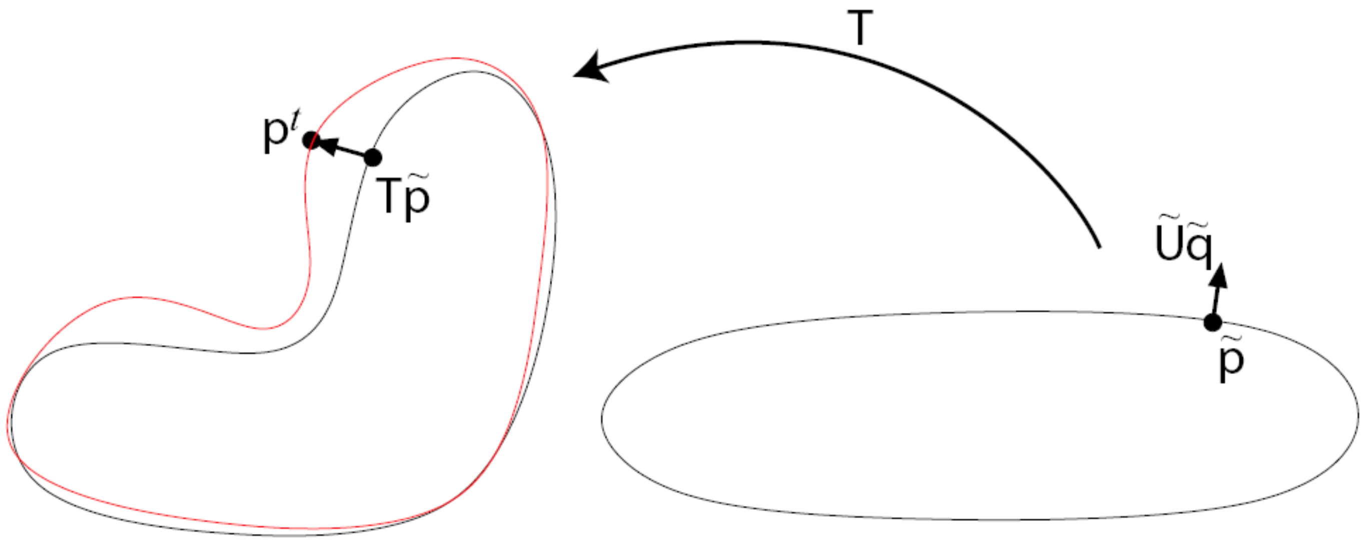


Overview

- Identify near-rigid structures
- Estimate bone transforms
- Estimate vertex weights
- **Progressive skin corrections**

Progressive skin corrections

- As described in *Kry, Paul G. et al. 2002: EigenSkin*



- Transform errors back to rest pose
- Perform data reduction (PCA, SVD)
- Add result to rest pose

Results

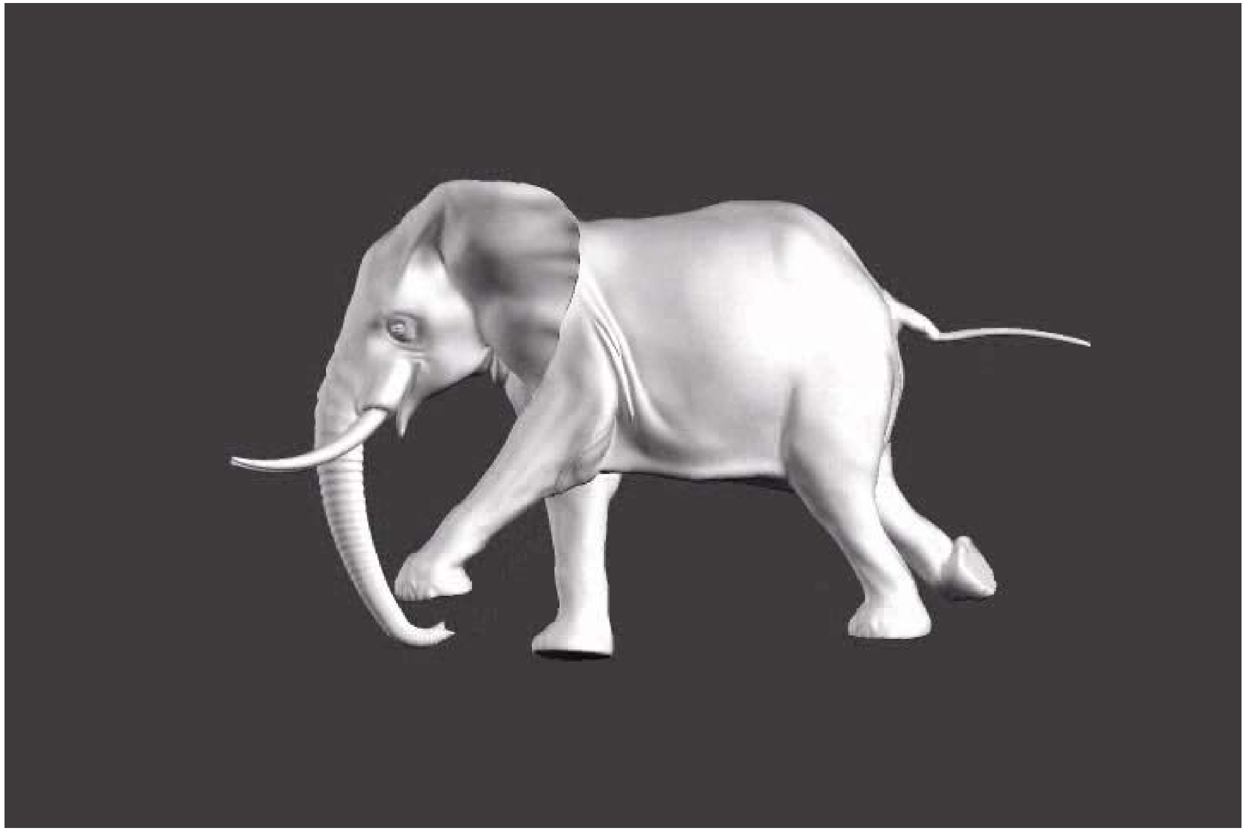
- Benefits of skinned meshes:
  - Animation compression

Results

- Benefits of skinned meshes:
  - Hardware acceleration

Results

- Benefits of skinned meshes:
  - Rest pose editing

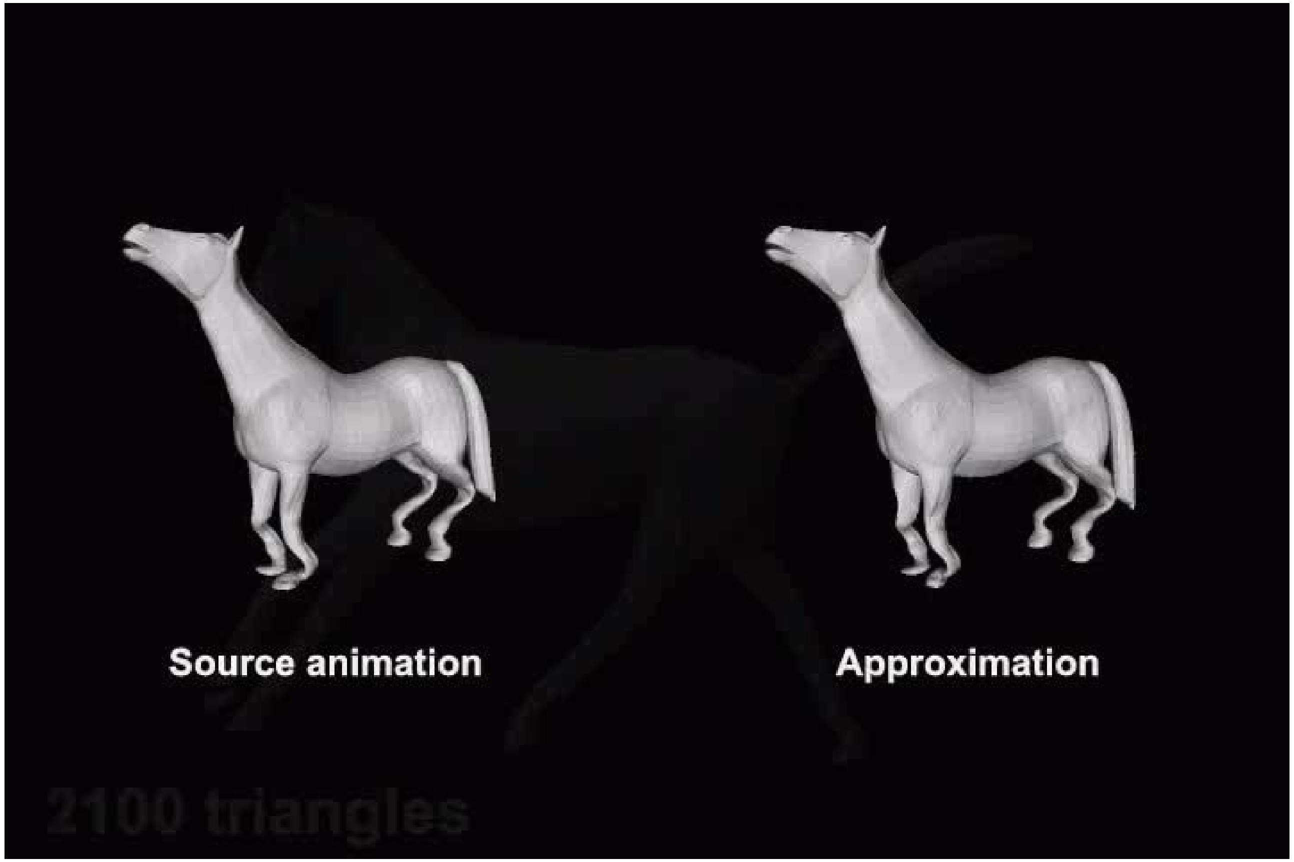


Results

- Benefits of skinned meshes:
  - Fast collision detection

Results

- Problems:
  - Highly deformable models



Results

- Benefits of skinned meshes:
  - Animation compression
  - Hardware acceleration
  - Rest pose editing
  - Collision detection
- Problems:
  - Highly deformable models
- Computation of skin: Order of Minutes

Opinions & Discussion