Character Animation

COS 426, Fall 2022
Computer Animation

- Describing how 3D objects (& cameras) move over time

Pixar
Computer Animation

• Challenge is balancing between …
  • Animator control
  vs.
  • Physical realism
Computer Animation

• Manipulation
  • Posing
  • Effect of pose

• Interpolation
  • Keyframes
  • In-betweens

https://blenderartists.org/

focus.gscept.com
Character Animation Methods

• Modeling (manipulation)
  • Deformation
  • Blendshapes
  • Skeletons

• Interpolation
  • Key-framing
  • Kinematics
  • Motion Capture

https://blenderartists.org/
focus.gscept.com
Character Animation Methods

• Modeling (manipulation)
  • Deformation
  • Blendshapes
  • Skeletons

• Interpolation
  • Key-framing
  • Kinematics
  • Motion Capture

https://blenderartists.org/

focus.gscept.com
Deformation

• How to change a character’s pose?
  • Every vertex directly
  • Intuitive computation

https://www.youtube.com/watch?v=oxkf_N-QCNI
Deformation

• A HUGE variety of methods
  • Laplacian mesh editing
  • ARAP
  • CAGE Base
  • Barycentric coordinates
  • Heat diffusion
  • Variational
  • …
Deformation

• A HUGE variety of methods
  • Laplacian mesh editing
  • ARAP
  • CAGE Base
  • Barycentric coordinates
  • Heat diffusion
  • Variational
  • …
Overall framework

1. Compute differential representation

\[ \delta_i = L(v_i) = v_i - \frac{1}{d_i} \sum_{j \in N(i)} v_j \]

2. Pose modeling constraints

\[ v_i' = u_i, \quad i \in C \]
Overall framework

1. Compute differential representation

\[ \delta_i = L(v_i) = v_i - \frac{1}{d_i} \sum_{j \in N(i)} v_j \]

2. Pose modeling constraints

\[ v'_i = u_i, \quad i \in C \]

3. Reconstruct the surface – in least-squares sense

\[ \begin{pmatrix} L \\ L_c \end{pmatrix} \mathbf{v} = \begin{pmatrix} \delta \\ U \end{pmatrix} \]
Example

Laplacian Mesh Editing

A short editing session with the Octopus
Character Animation Methods

- Modeling (manipulation)
  - Deformation
  - Blendshapes
  - Skeletons

- Interpolation
  - Key-framing
  - Kinematics
  - Motion Capture

focus.gscept.com
https://blenderartists.org/
Blendshapes

- Blendshapes are an approximate semantic parameterization
- Linear blend of predefined poses
Blendshapes

https://www.youtube.com/watch?v=KPDfMpuK2fQ
Blendshapes

- Usually used for difficult to pose complex deformations
  - Such faces
- Given:
  - A mesh $M = (V, E)$ with $m$ vertices
  - $n$ configurations of the same mesh, $M_b = (V_b, E), b = 1 \ldots n$
Blendshapes

• Usually used for difficult to pose complex deformations
  • Such faces

• Given:
  • A mesh $M = (V, E)$ with $m$ vertices
  • $n$ configurations of the same mesh, $M_b = (V_b, E), b = 1 \ldots n$

• A new configuration is simply:
  • $M' = (\sum_{b=1}^{n} w_b V_b, E)$
Blendshapes

- Usually used for difficult to pose complex deformations
  - Such faces
- Given:
  - A mesh $M = (V, E)$ with $m$ vertices
  - $n$ configurations of the same mesh, $M_b = (V_b, E), b = 1 \ldots n$
- A new configuration is simply:
  - $M' = (\sum_{b=1}^n w_b V_b, E)$
- Delta formulation:
  - $M' = (\sum_{b=1}^n V_0 + w_b (V_b - V_0), E)$
    - A bit more convenient
- $M_0$ - the rest pose, $w_b$ blend weights
Character Animation Methods

• Modeling (manipulation)
  • Deformation
  • Blendshapes
  • Skeletons

• Interpolation
  • Key-framing
  • Kinematics
  • Motion Capture
Articulated Figures

- Character poses described by set of rigid bodies connected by “joints”
Articulated Figures

• Well-suited for humanoid characters

- Animation focuses on joint angles, or general transformations

Rose et al. ’96
Character Animation Methods

- Modeling (manipulation)
  - Deformation
  - Blendshapes
  - Skeletons

- Interpolation
  - Key-framing
  - Kinematics
  - Motion Capture

https://blenderartists.org/

focus.gscept.com
Forward Kinematics

- Describe motion of articulated character

\[ \begin{align*}
X &= (x, y) \\
\Theta_1 & \quad \Theta_2 \\
(0,0) & \quad \text{"End-Effector"}
\end{align*} \]
Forward Kinematics

• Animator specifies joint angles: \( \Theta_1 \) and \( \Theta_2 \)
• Computer finds positions of end-effector: \( X \)
Forward Kinematics

• Animator specifies joint angles: $\Theta_1$ and $\Theta_2$
• Computer finds positions of end-effector: $X$

\[
X = (l_1 \cos \Theta_1 + l_2 \cos(\Theta_1 + \Theta_2), l_1 \sin \Theta_1 + l_2 \sin(\Theta_1 + \Theta_2))
\]
Forward Kinematics Parameterization

- Joint motions specified e.g. by spline curves

\[ X = (x, y) \]

\[ Q_1, Q_2 \]

\[ \Theta_1, \Theta_2 \]

\[ (0,0) \]
Example: Walk Cycle

• Articulated figure:
Example: Walk Cycle

• Hip joint orientation:
Example: Walk Cycle

• Knee joint orientation:
Example: Walk Cycle

- Ankle joint orientation:
Example: walk cycle

https://www.youtube.com/watch?v=DuUWxUitJos
Inverse Kinematics

- What if animator knows position of “end-effector”?

\[
X = (x, y)
\]

(0,0)
Inverse Kinematics

- Animator specifies end-effector positions: \( X \)
- Computer finds joint angles: \( \Theta_1 \) and \( \Theta_2 \):
Inverse Kinematics

• Animator specifies end-effector positions: \( X \)
• Computer finds joint angles: \( \Theta_1 \) and \( \Theta_2 \):

\[
\Theta_2 = \cos^{-1}\left(\frac{x^2 + y^2 - l_1^2 - l_2^2}{2l_1l_2}\right)
\]

\[
\Theta_1 = \frac{- (l_2 \sin(\Theta_2)) x + (l_1 + l_2 \cos(\Theta_2)) y}{(l_2 \sin(\Theta_2)) y + (l_1 + l_2 \cos(\Theta_2)) x}
\]
Inverse Kinematics

- End-effector positions can be specified by spline curves

\[ X = (x, y) \]

\[ \Theta_1 \]

\[ \Theta_2 \]

\[ (0,0) \]
Inverse Kinematics

- Problem for more complex structures
  - System of equations is usually *under-constrained*
  - Multiple solutions

\[
X = (x, y)
\]

Three unknowns: \( \Theta_1, \Theta_2, \Theta_3 \)

Two equations: \( x, y \)
Inverse Kinematics

- Solution for more complex structures:
  - Find best solution (e.g., minimize energy in motion)
  - Non-linear optimization

\[ X = (x, y) \]
Kinematics

- **Advantages**
  - Simple to implement
  - Complete animator control

- **Disadvantages**
  - Motions *may not follow physical laws*
  - Tedious for animator
Beyond Skeletons...

- Skinning
Kinematic Skeletons

- Hierarchy of transformations ("bones")
  - Changes to parent affect all descendental bones

- So far: bones affect objects in scene or parts of a mesh
  - Equivalently, each point on a mesh acted upon by one bone
  - Leads to discontinuities when parts of mesh animated
Kinematic Skeletons

• Hierarchy of transformations ("bones")
  • Changes to parent affect all descendental bones

• So far: bones affect objects in scene or parts of a mesh
  • Equivalently, each point on a mesh acted upon by one bone
  • Leads to discontinuities when parts of mesh animated

• Extension: each point on a mesh acted upon by more than one bone
Linear Blend Skinning

- Each vertex of skin potentially influenced by all bones
  - Normalized weight vector $w^{(v)}$ gives influence of each bone transform
  - When bones move, influenced vertices also move
Linear Blend Skinning

- Each vertex of skin potentially influenced by all bones
  - Normalized weight vector $w^{(v)}$ gives influence of each bone transform
  - When bones move, influenced vertices also move

- Computing a transformation $T_v$ for a skinned vertex
  - For each bone
    - Compute global bone transformation $T_b$ from transformation hierarchy
  - For each vertex
    - Take a linear combination of bone transforms $T_v$ with $T_v = \sum_{b \in B} w^{(v)}_b T_b$
    - Apply transformation $T_v$ to vertex in original pose
Linear Blend Skinning

• Each vertex of skin potentially influenced by all bones
  • Normalized weight vector $w(v)$ gives influence of each bone transform
  • When bones move, influenced vertices also move

• Computing a transformation $T_v$ for a skinned vertex
  • For each bone
    • Compute global bone transformation $T_b$ from transformation hierarchy
  • For each vertex
    • Take a linear combination of bone transforms $T_v$ with
    • Apply transformation $T_v$ to vertex in original pose

$$T_v = \sum_{b \in B} w_b^{(v)} T_b$$

• Equivalently, transformed vertex position is weighted combination of positions transformed by bones

$$v_{transformed} = \sum_{b \in B} w_b^{(v)} (T_b v)$$
Assigning *Weights*: “Rigging”

- Painted by hand
- Automatic: function of relative distances to nearest bones
  - Smoothness of skinned surface depends on smoothness of weights!
Assigning Weights: “Rigging”

- Painted by hand
- Automatic: function of relative distances to nearest bones
  - Smoothness of skinned surface depends on smoothness of weights!
  - Other problems with extreme deformations
    - Many solutions
Character Animation Methods

- Modeling (manipulation)
  - Deformation
  - Blendshapes
  - Skeletons

- Interpolation
  - Key-framing
  - Kinematics
  - Motion Capture

https://blenderartists.org/
focus.gscept.com
Keyframe Animation

- Define character poses at specific time steps called “keyframes”
Keyframe Animation

- Interpolate variables describing keyframes to determine poses for character in between
Keyframe Animation

• Inbetweening:
  • Linear interpolation - usually not enough continuity

H&B Figure 16.16
Keyframe Animation

• Inbetweening:
  • Spline interpolation - maybe good enough
Character Animation Methods

- Modeling (manipulation)
  - Deformation
  - Blendshapes
  - Skeletons

- Interpolation
  - Key-framing
  - Kinematics
  - Motion Capture

https://blenderartists.org/

focus.gscept.com
Motion Capture

- Measure motion of real characters and then simply “play it back” with kinematics.
Motion Capture

- Measure human motion
- Play back with kinematics

https://www.youtube.com/watch?v=MVvDw15-3e8
Motion Capture for Faces

- Could be applied on different parameters
  - Skeleton Transformations
  - Direct mesh deformation

- Advantage:
  - Physical realism

- Challenge:
  - Animator control
Summary

- **Kinematics**
  - Animator specifies poses (joint angles or positions) at keyframes and computer determines motion by kinematics and interpolation

- **Dynamics**
  - Animator specifies physical attributes, constraints, and starting conditions and computer determines motion by physical simulation

- **Motion Capture**
  - Compute captures motion of real character and provides tools for animator to edit it