Animations

Hakan Bilen
University of Edinburgh

Computer Graphics
Fall 2017

Some slides are courtesy of Steve Marschner and Kavita Bala
Artistic process

- What are animators trying to do?
  - What tools do they need?
- Basic principles are universal across media
  - 2D hand-drawn animation
  - 2D computer animation
  - 3D computer animation
- The 12 principles of animation laid out by Frank Thomas and Ollie Johnston in The Illusion of Life (1981)

https://www.youtube.com/watch?v=uDqjIdI4bF4
The 12 principles

1. Squash & Stretch
2. Anticipation
3. Staging
4. Straight ahead vs Pose-to-Pose
5. Follow through
6. Slow-in and slow-out
7. Arcs
8. Secondary action
9. Timing
10. Exaggeration
11. Solid drawing
12. Appeal
What is animation?

- Modeling = specifying shape
  - using all the tools we’ve seen: hierarchies, meshes, curved surfaces…

- Animation = specifying shape as a function of time
  - just modeling done once per frame?
  - yes, but need smooth, concerted movement

- “Straight ahead” is drawing frames in order
  - hard to get a character to land at a particular pose/time

- Keyframes
  - draw important poses and interpolate in-betweens
    - Key ideas
      - create high-level controls for adjusting geometry
      - interpolate these controls
Keyframe animation

Computer animation

• A 3D scene is animated by a set of parameters
  • Object and part locations, camera position, light source intensity

• Key frames to key values (interpolation)
  • https://www.youtube.com/watch?v=ZmNmQ3izNnI
Problems with interpolation

- Splines don’t always output the right thing

Problems
- 3D rotations
  - Euler angles don’t always interpolate in a natural way
- Invalid configurations
  - Go through walls, anatomically impossible poses

Solutions
- More keyframes
- Quaternions for rotation
- Incorporate kinematics and constraints
Gimbal lock

Occurs if during rotation one of the three rotation axes is by accident aligned with another
Quaternions

Given an unit length vector 
\[ \mathbf{u} = (u_x, u_y, u_z) \] and angle \( \theta \)

Quaternion can be written as

- \[ q = e^{\frac{\theta}{2}(u_x i + u_y j + u_z k)} \]
- \[ = \cos \frac{\theta}{2} + (u_x i + u_y j + u_z k) \sin \frac{\theta}{2} \]
- \[ = [s; \mathbf{v}] \]

Sum up and scale

- \[ q_1 + q_2 = [s_1 + s_2; \mathbf{v}_1 + \mathbf{v}_2] \]
- \[ \alpha q = [\alpha s; \alpha \mathbf{v}] \]

Spherical linear interpolation (slerp)
Controlling geometry

- Parameterize the motion by using smaller set of meaningful degrees of freedom (DOFs)

**Character with DOFs**

A visual description of the possible movements for the squirrel
Rigged character

- A digital skeleton
- Made up of joints and bones
- The controls are useful, intuitive DOFs for an animator to use

https://www.youtube.com/watch?v=cGvalWG8HBU
Articulated figures

Forward Kinematics

Inverse Kinematics
Forward and inverse kinematics

Constraints

- \( \theta_1 \)
- \( \theta_2 \)

Inverse kinematics
- Multiple solutions possible
- Consider constraints (bending knee in one direction)
Mesh skinning

A simple way to deform a surface to follow a skeleton

- Surface has control points $p_i$ (triangle vertices, spline control pts)
- Each bone has a transformation matrix $M_j$ (rigid motion)
- Every point-bone pair has weight $w_{ij}$
  - Only non-zero for nearby bones
  - Weights are input by the user

$$p'_i = \sum_j w_{ij} M_j p_i$$
Blend shapes

- Another very simple surface control scheme
- Based on interpolating among several key poses

http://videos.weebly.com/uploads/5/1/7/7/5177454/testblends_506.mp4
Motion capture

Movies

A method for creating complex motion quickly: measure it from the real world
Motion capture

Games
Motion capture

Magnetic MoCap
- Tethered, distortions from nearby metal, 60Hz

Mechanical MoCap
- Direct measures of joint angles, restrict motion

Optical markers
- Passive markers, observed by cameras, 8 or more cameras, occlusions are problematic
Motion capture

Data processing

• Which marker is which?
  • Start with a known pose and track

• Calibration: match skeleton, find offsets to markers
  • A nonlinear optimization

• Computing joint angles: explain data using skeleton DOFs
  • Inverse kinematics
Phase-Functioned Neural Networks for Character Control

Example

https://www.youtube.com/watch?v=Ul0Gilv5wvY&t=208s

Data Preprocessing

1. Motion Capture and Processing
2. Phase Extraction
3. Terrain Fitting

Training

4. PFNN Training by Backpropagation

Runtime

5. Realtime Character Control by User
Summary

• The 12 principles

• Keyframing
  - Gimbal lock
  - Quaternions

• Controlling geometry
  - Character rigging
  - Forward / inverse kinematics
  - (Deformations) Mesh skinning
  - Blending

• Motion capture

• Other techniques:
  - physics-based animation
  - procedural techniques

HAL 9000 Animations
https://www.youtube.com/watch?v=HurJ3b7n_8w
Reading

B1: Chapter 16