Lecture 28 – An hours worth of animation

Michael Gleicher
November, 2006
Used as notes, not projected as lecture

Computer Animation

- Worth its own course (at least)
  - We only get an hour (or less, since need to do evals)
- Go over some of the ideas / concepts
- See how some graphics concepts come into play
- A whole art form
- A wide range of technical challenges

What is Computer Animation?

- Making moving images with a computer
  - 3D movies and games
  - Web browser animations (annoying and useful)
  - Scientific Visualization and Design
  - Desktop animations – paper clip, windows moving
- Why is this hard/different
  - Need to consider how things move / what is their motion
  - Need things to be easier to control (so we can move them)
  - Lots of images, coherence issues

Principles of Animation

- Why animate the moving windows?
  - Looks cool
  - Easier to understand what’s going on
- Ideas from animation help everywhere
  - Animation was (historically) hard to produce
  - Need to be economical about drawing
  - Needed to learn how to communicate in moving images
- Early animators
  - Anything can happen in animation
    - A talking and dancing mouse?
  - How to make it understandable
  - Control surprise

Principles of Animation

- Developed in the late 20s early 30s
- Disney was a key player
- Exagerration
- Anticipation
- Follow Through and Overlap
- Secondary Action
- Squash and Stretch
- Staging
- Timing
- Slow in/slow out
- Straight ahead vs. pose to pose, Arcs, Appeal

How was this done historically

- Cel animation
  - Transparent sheets of celuloid
  - Allows characters drawn independent of background
  - Layers
- Character Animation
  - Needs a good artists
  - Lots of drawings to make
- Keyframing
  - Master artists draws a few poses
  - “Tweener” draws “in-betweens”


Keyframing by Computer

- Still how the best character animation is done
- Good artists can be extremely creative
- Set a small number of “key poses”
- Use interpolation to get in-betweens
- Big application of interpolating splines
  - Catmull-Rom Splines
  - TCB (tension continuity bias splines)
  - Cardinals with more control (still interpolate)
  - Change tension – per control point
  - Tension on each side of control point (bias)
  - Deviate from C(1) (continuity)

Parametric Models

- What do you interpolate?
  - Need to have a vector of numbers – point in pose space
- Controls or parameters
  - Need enough to be expressive
  - Few enough to be convenient
- Use position of every point on a mesh?
  - Lots of data to move around on every frame
- Use rigid transform?
  - Might not be enough to just move things around
  - OK for levels of detail
- Use deformations?

Articulated Figure Animation

- Humans and animals (vertebrate) modeled as rigid bones
  - Gets the main effect
- Bones are rigid (don’t really change size)
- Connected by joints (rotation)
- Configuration = position of “root” + orientations
- Can pick any point to be root
  - Center or pelvis
  - Foot for convenience

Why is hierarchical good?

- Fewer parameters
- Enforce essential constraints
  - Keep from stretching
  - Keep limbs from falling off
- Keep constraints when posing or interpolating
- Why is it bad?
  - Hard to position end points
  - Hard to enforce constraints on end points (footskate)
  - Parameters are coupled

Controlling Hierarchical Models

- Forward Kinematics
  - Specify angles, see what happens
- Inverse Kinematics
  - Specify end-effector positions, figure out where joints angles must be
- Doing IK
  - Might be no solutions
  - Might be lots of solutions
  - Non-linear equations
  - Easy to solve for special cases (2 link arms)

Drawing Hierarchical Models

- Simple: Draw rigid pieces
- Complex: (arbitrarily)
  - Compute some “skin” over the bones
  - Use simulation, or anything
- One tradeoff:
  - Use a simple “skinning” model
  - Have a single mesh for the object
  - Associate each vertex with multiple coordinate systems
  - Weights determine how much
Skinning

• Smooth skinning, linear blend skinning, …
• Blend positions (interpolate) or matrices
  – Note that interpolating matrices doesn’t preserve rigidity
• Good points
  – Easy
  – Efficient
  – Maps Nicely to hardware
• Bad points
  – Simple / hackish
  – Hard to find weights
  – Bad effects (collapsing, candy-wrapping)
• Improved methods keep coming…

Where does motion come from

• Keyframing
• Observation (motion capture)
• Procedural (compute it)
• Physics = a form of procedural
• Synthesis by example = combine procedure and observation

Animation by Observation

• Motion Capture
• Record the movements of a real performer
• Why?
  – Realistic motion
  – Get the actual person
  – Actors are directable
  – No need to have models of motion properties
    • Mathematical definition for “happy” or “skip”
• Why not?
  – Realistic motion
  – Get what actor does (at best)
  – Need special devices to record

How to do Motion Capture

• Video – not possible (yet)
  – Not enough information to make good measurements
  – Can’t get correspondences for triangulation
• Optical Motion Capture
  – Engineer away what’s hard about video
  – Dot markers – easy to find
    • Retro-reflective markers (lights on cameras)
    • LEDs (blinked to get correspondences)
  – Cameras only see bright spots
  – Many cameras (to do triangulation, disambiguation)
  – Some systems have dozens – if not hundreds – of cameras

Why not optical mocap

• Still get drop outs
• Real-time, online hard (drop outs, correspondence)

How to use motion capture

• Individual clips generally short
• String together into longer chains
• Transitions
  – Might be easy, might be hard
  – Look for easy cases
• Motion Graphs