Guest Lecture
CS 445

David Brogan
dbrogan@cs.virginia.edu

Introduction

Animators study how things move and create computer graphics representations that look “good enough”

Rendering is: mapping light sources and surfaces to a vector of pixel colors

Animation is: mapping objects, intentions, and external forces to a vector of new object positions / orientations

Study how things move

Who else does this?

- Biomechanists
- Physics and sensors
- Artists
  - Intuition and mind’s eye

Generate graphics that is “good enough”

Who else studies this?

- Perceptual psychologists
- Artists

Monet
La Cathédrale de Rouen (1894)  The Bull (1946)

University of Utah
Completing the mapping

Bridge gap between knowledge of how things move to how they need to be rendered

- Artists use their minds and hands
- Computer scientists use math and programs

Animation timeline

Persistence of vision

- Thaumotrope (1800s)
- Flipbook
- Zoetrope (1834)
- Shadow puppets

Photography

- Muybridge (1885)
- Film projector (Edison, 1891)

First Animation

- 1896, Georges Melies, moving tables
- 1900, J. Stuart Blackton, added smoke

First celebrated cartoonist

- Winsor McCay
- Little Nemo (1911)
- Gertie the Dinosaur (1914)
Old-school animation technology

*Patents issued in early 1900s*
- Translucent cels used in layers for compositing
- Gray-scale drawings (cool!)
- Pegs for registration (alignment) of overlays
- Large background drawings and panning camera
- Rotoscoping: drawing images on cells by tracing over previously recorded live action (MoCap)

Perception

*Computer graphics rendering can rely on four-hundred years of perception research by artists*
- The best animators have is eighty years of Disney

In 1550, after 100 years of refining the art of perspective drawing, artists were shocked to think that the geometric purity of their modeled world didn’t map to recent discoveries of the human eye. They couldn’t even imagine how cognition affected what one “saw.” 200 more years would pass.

Disney

*Advanced animation more than anyone else*
- First to have sound in 1928, Steamboat Willie
- First to use storyboards
- First to attempt realism
- Invented multiplane camera

Modern animation technology

*Keyframing*
- Digital inbetweens

*Motion Capture*
- What you record is what you get

*Simulation*
- Animate what you can model (with equations)
Keyframing

Traditional animation technique
Dependent on artist to generate ‘key’ frames
Additional, ‘inbetween’ frames are drawn automatically by computer

Keyframing

How are we going to interpolate?

Figure 1.1 - Three keyframes. Three keyframes representing a ball on the ground, at its highest point, and back on the ground.

From "The computer in the visual arts", Spalter, 1999.

Linear Interpolation

Figure 10.5 - Inbetweening with linear interpolation. Linear interpolation creates inbetween frames at equal intervals along straight lines. The ball moves at a constant speed. Ticks indicate the locations of inbetween frames at regular time intervals (determined by the number of frames per second chosen by the user).

Simple, but discontinuous velocity
Nonlinear Interpolation

Smooth ball trajectory and continuous velocity, but loss of timing

Easing

Adjust the timing of the inbetween frames. Can be automated by adjusting the stepsize of parameter, t.

Style or accuracy?

Interpolating time captures accuracy of velocity
Squash and stretch replaces motion blur stimuli and adds life-like intent

Traditional Motivation

Ease-in and ease-out is like squash and stretch
Can we automate the inbetweens for these?

"The Illusion of Life, Disney Animation"
Thomas and Johnson
More squash and stretch

Anticipation and staging

Don’t surprise the audience
Direct their attention to what’s important

Follow through

Audience likes to see resolution of action
Discontinuities are unsettling

Interpolation

- Many parameters can be interpolated to generate animation
- Simple interpolation techniques can only generate simple inbetweens
- More complicated inbetweening will require a more complicated model of animated object and simulation
Kinematics
The study of object movements irrespective of their speed or style of movement

Degrees of Freedom
How many variables define configuration?

- Six
  - 2-base, 1-shoulder, 1-elbow, 2-wrist

Count the DOFs
Some joints have lots of DOFs

3 DOF joints
- Gimbal
- Spherical (doesn’t possess singularity)

2 DOF joints
- Universal
Forward vs. Inverse Kinematics

**Forward Kinematics**
- Compute configuration (pose) given individual DOF values
  - Good for simulation

**Inverse Kinematics**
- Compute individual DOF values that result in specified end effector position
  - Good for control

---

**What is Inverse Kinematics?**

Forward Kinematics

Traverse kinematic tree and propagate transformations downward
- Use stack
- Compose parent transformation with child’s
- Pop stack when leaf is reached

---

**Inverse Kinematics (IK)**

Given end effector position, compute required joint angles

*In simple case, analytic solution exists*
- Use trig, geometry, and algebra to solve

---

**Given end effector position, compute required joint angles**

In simple case, analytic solution exists
- Use trig, geometry, and algebra to solve

---

What is Inverse Kinematics?

Forward Kinematics

Traverse kinematic tree and propagate transformations downward
- Use stack
- Compose parent transformation with child’s
- Pop stack when leaf is reached

---

What is Inverse Kinematics?

Forward Kinematics

Traverse kinematic tree and propagate transformations downward
- Use stack
- Compose parent transformation with child’s
- Pop stack when leaf is reached

---

What is Inverse Kinematics?

Forward Kinematics

Traverse kinematic tree and propagate transformations downward
- Use stack
- Compose parent transformation with child’s
- Pop stack when leaf is reached

---

What is Inverse Kinematics?

Forward Kinematics

Traverse kinematic tree and propagate transformations downward
- Use stack
- Compose parent transformation with child’s
- Pop stack when leaf is reached
What is Inverse Kinematics?

**Inverse Kinematics**

\[ \theta = f^{-1}(\mathbf{x}) \]

---

**Failures of simple IK**

*Multiple Solutions*

---

**Failures of simple IK**

*Infinite solutions*

---

Analytic solution of 2-link inverse kinematics:

\[
\begin{align*}
x_1^2 + y_1^2 &= a_1^2 + \left(a_2 - x_2 \right)^2 - 2a_2 (x_1 \cos(\theta_1) - y_1 \sin(\theta_1)) \\
\cos(\theta_1) &= \frac{x_1^2 + y_1^2 - a_1^2 - a_2^2}{2a_2 x_2} \\
\end{align*}
\]

For greater accuracy:

\[
\begin{align*}
\tan(\theta_2) &= \frac{1 - \cos(\theta_1)}{\sin(\theta_1)} \\
\cos(\theta_2) &= \frac{2x_0 y_0 + 2y_1 - 2x_1 + a_2^2 - a_1^2}{a_2^2 + y_0^2 - a_1^2} \\
\tan(\theta_2 + \epsilon) &= \frac{a_1^2 + a_2^2 - (x_1^2 + y_1^2)}{(x_1^2 + y_1^2) - (a_1^2 - a_2^2)} \\
\theta_2 &= -\epsilon \tan^{-1}
\end{align*}
\]
Failures of simple IK

Solutions may not exist

Iterative IK Solutions

Frequently analytic solution is infeasible

Use Jacobian

- Derivative of function output relative to each of its inputs

If $y$ is function of three inputs and one output

$$\delta y = \frac{\partial y}{\partial x_1} \delta x_1 + \frac{\partial y}{\partial x_2} \delta x_2 + \frac{\partial y}{\partial x_3} \delta x_3$$

- Represent Jacobian, $J(X)$, as a 1x3 matrix of partial derivatives

Jacobian

In another situation, end effector has 6 DOFs and robotic arm has 6 DOFs

$f(x_1, ..., x_6) = (x, y, z, r, p, y)$

Therefore $J(X) = 6 \times 6$ matrix

Jacobian

Relates velocities in parameter space to velocities of outputs

$$\dot{Y} = J(X) \cdot \dot{X}$$

If we know $Y_{\text{current}}$ and $Y_{\text{desired}}$, then we subtract to compute $Y_{\text{dot}}$

Invert Jacobian and solve for $X_{\text{dot}}$
Keyframing

Strengths
- Animator has exacting control (Woody’s face)

Weaknesses
- Interpolation hooks must be simple and direct
  - Remember the problems with Euler angle interp?
- Time consuming and skill intensive
- Difficult to reuse and adjust

Motion capture

Examples
- Sports video games
  - Madden Football
- Many movie characters
  - Phantom Menace
- Cartoons
Motion capture strengths

*Exactly captures the motions of the actor*
- Michael Jordan’s video game character will capture his style

*Easy to capture data*

Motion capture weaknesses

*Noise, noise, noise!*
- Magnetic system interference
- Visual system occlusions
- Mechanical system mass
- Tethered (wireless is available now)

Motion capture weaknesses

*Aligning motion data with CG character*
- Limb lengths
- Idealized perfect joints

*Reusing motion data*
- Difficult to scale in size (must also scale in time)
- Changing one part of motion

Motion capture weaknesses

*Blending segments*
- Motion clips are short (due to range and tethers)
- Dynamic motion generation requires blending at run time
- Difficult to manage smooth transition
**Examples**

*Inanimate video game objects*
- GT Racer cars
- Soapbox about why this is so cool

*Special effects*
- Explosions, water, secondary motion
- Phantom Menace CG droids after they were cut in half

---

**Procedural animation**

- Very general term for a technique that puts more complex algorithms behind the scenes
- Technique attempts to consolidate artistic efforts in algorithms and heuristics
- Allows for optimization and physical simulation

---

http://taat.fi/taat/porrasturvat/

www.sodaplay.com
Sampling plausible solutions to multi-body constraint problems

2D Ball

Probability function

\[ p_{\text{ball}}(A) \propto e^{-\frac{1}{2}(\frac{x}{\sigma_x})^2} \prod_i e^{-\frac{1}{2}(\frac{\theta_i}{\sigma_{\theta_i}})^2} \]

Take the log

\[ \log(p(A)) = -\frac{1}{2} \left( \frac{x - D}{\sigma_d} \right)^2 - \frac{1}{2} \sum_{1 \leq i \leq 5} \left( \frac{\theta_i}{10.0} \right)^2 + C \]

Procedural animation strengths

- Animation can be generated ‘on the fly’
- Dynamic response to user
- Write-once, use-often
- Algorithms provide accuracy and exhaustive search that animators cannot

Procedural animation weaknesses

- We’re not great at boiling human skill down to algorithms
  - How do we move when juggling?
- Difficult to generate
- Expensive to compute
- Difficult to force system to generate a particular solution
  - Bicycles will fall down
Animation

It all boils down to

• How to most cheaply generate animation that looks
good enough subject to:
  – Reusable
  – Scalable
  – Timely
• A function of what resources are available:
  – Artists, mocap data, scientific formula, …