CS230 : Computer Graphics

Lecture 12: Introduction to Animation

Tamar Shinar
Computer Science & Engineering
UC Riverside

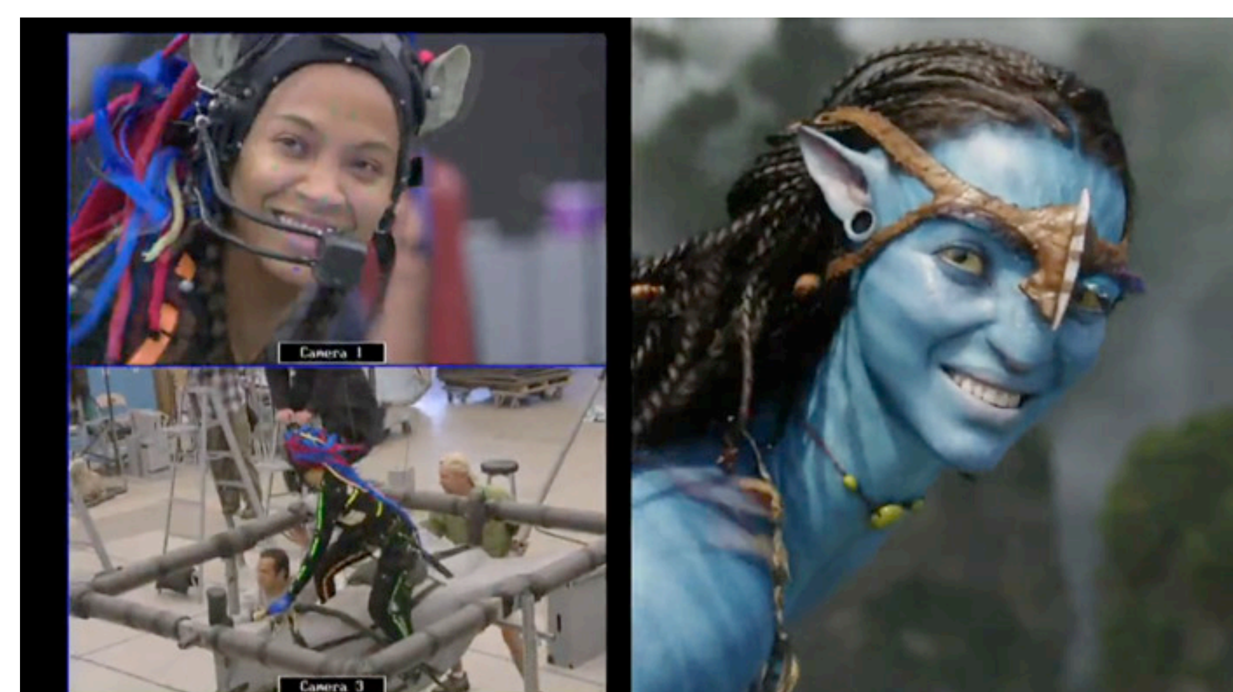
Types of animation

- keyframing
- rotoscoping
- stop motion
- procedural
- simulation
- motion capture

Performance capture

Rise of the Planet of the Apes, 2011

Lord of the Rings, 2001



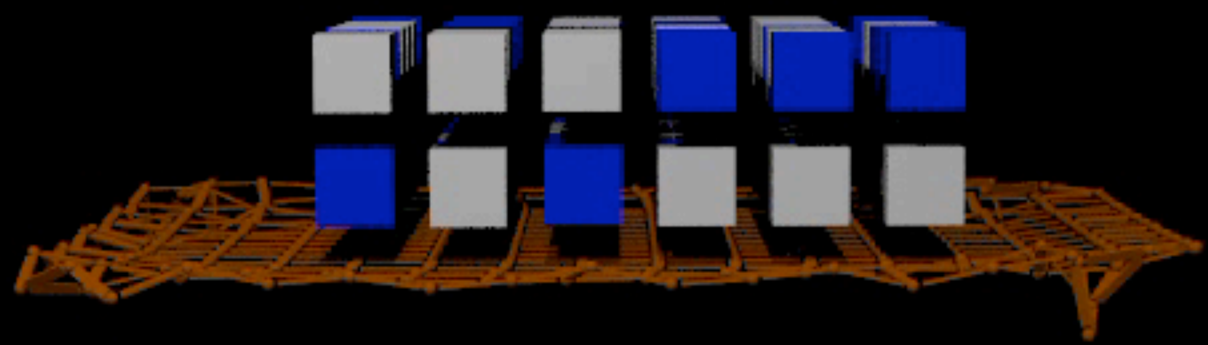
Avatar, 2009

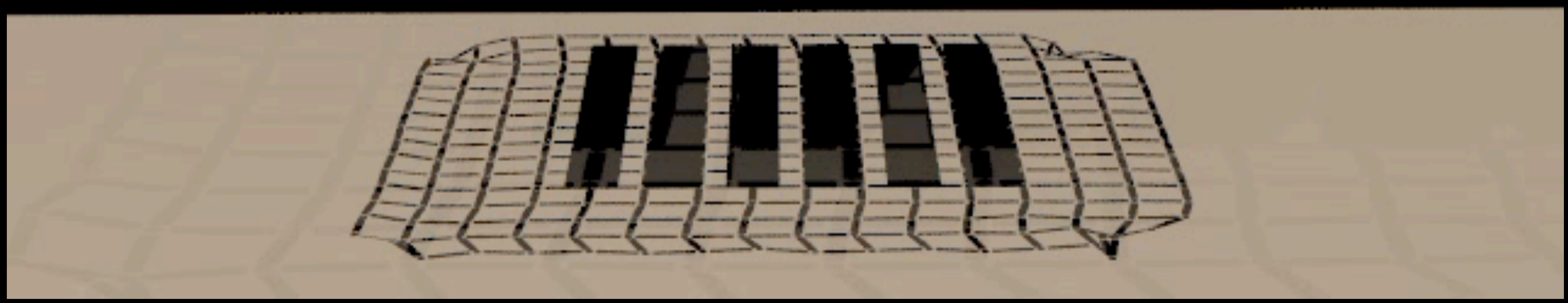
Andy Serkis – Gollum, Lord of the Rings
challenges – resolution, occlusion,

Rigid body simulation

Rachel Weinstein, Joey Teran and Ron Fedkiw

Rigid body simulation





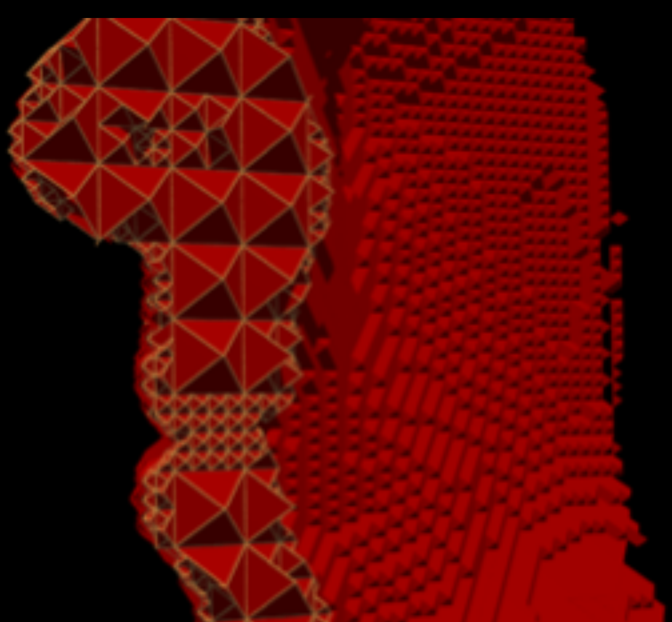
Rachel Weinstein, Joey Teran and Ron Fedkiw

Deformable object simulation

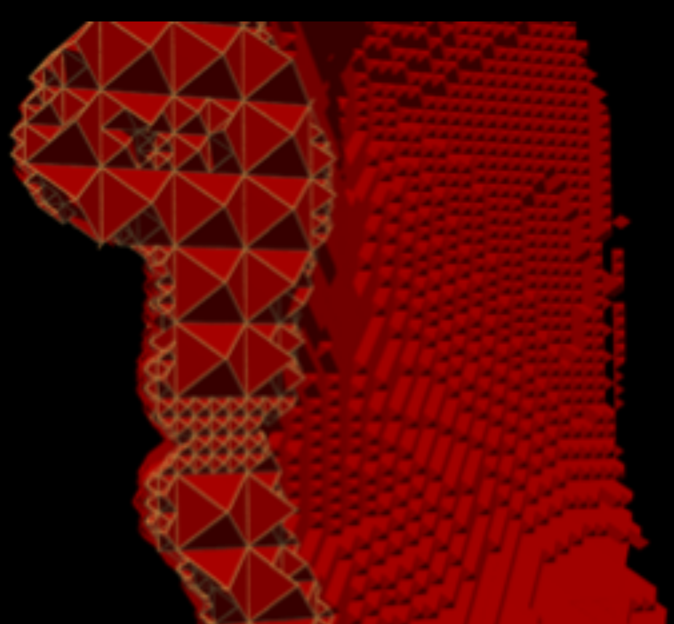
N. Molino, Z. Bao, R. Fedkiw

Selle et al., 2008

Facial animation

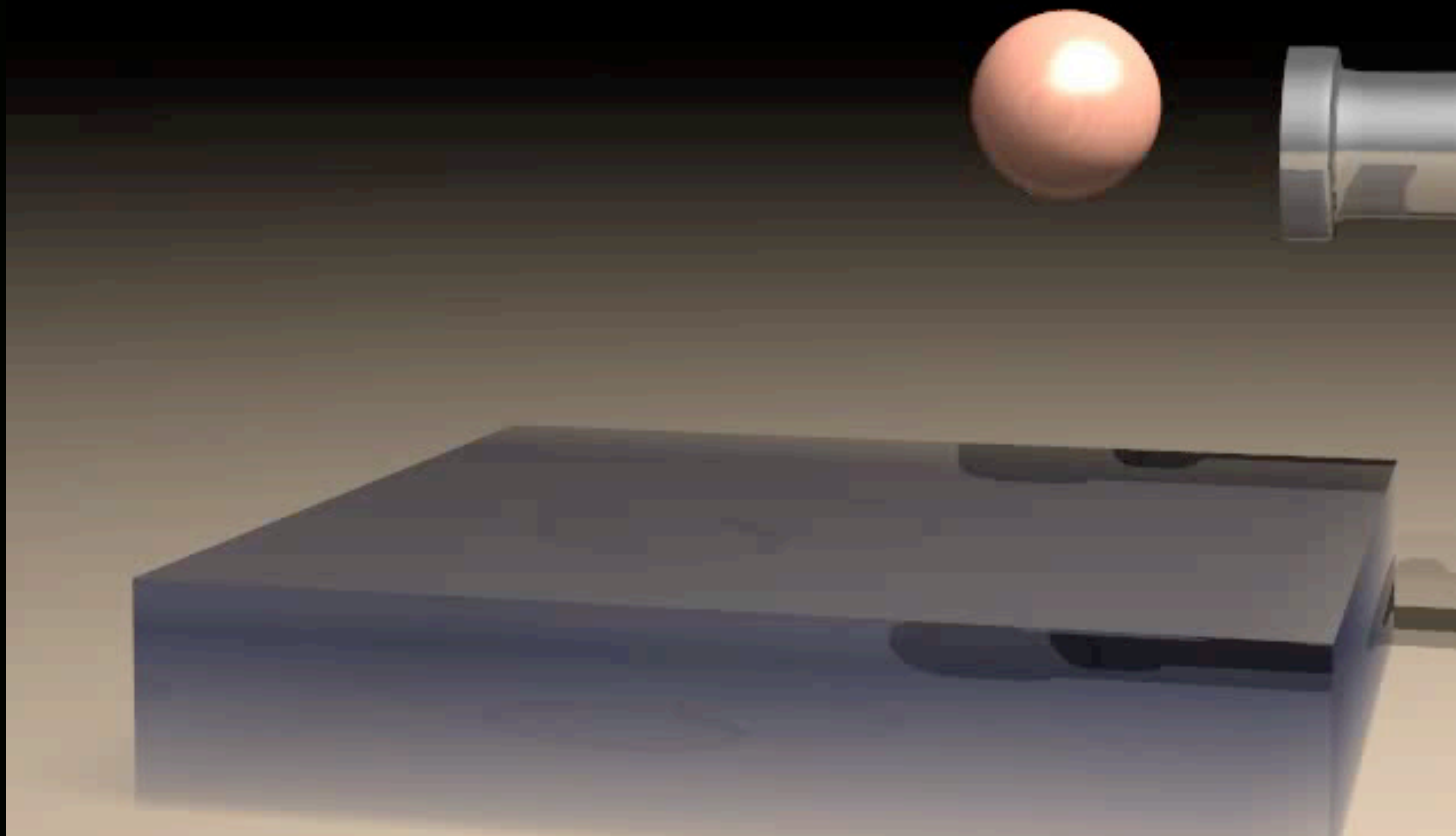


Facial animation



Fluid simulation

Fluid simulation

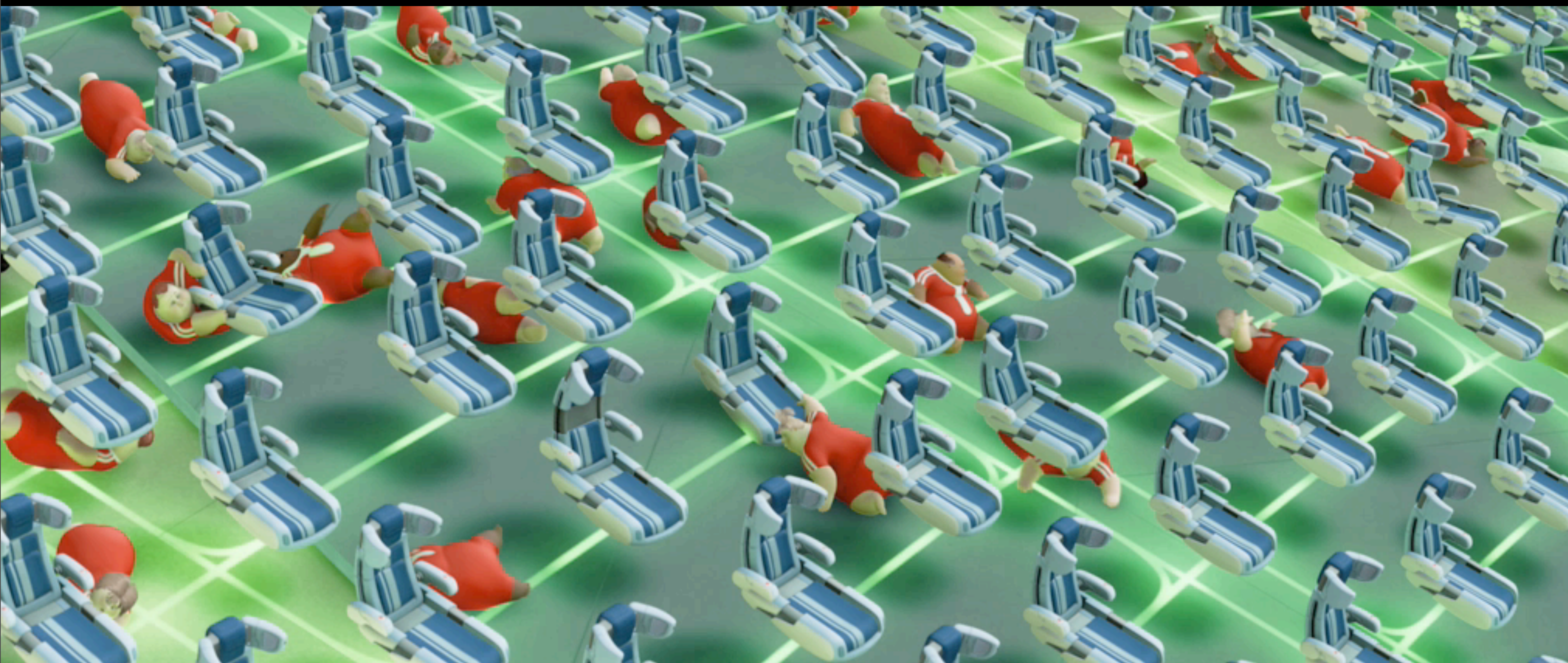


Control of virtual character

issues: control algorithms, interaction with environment

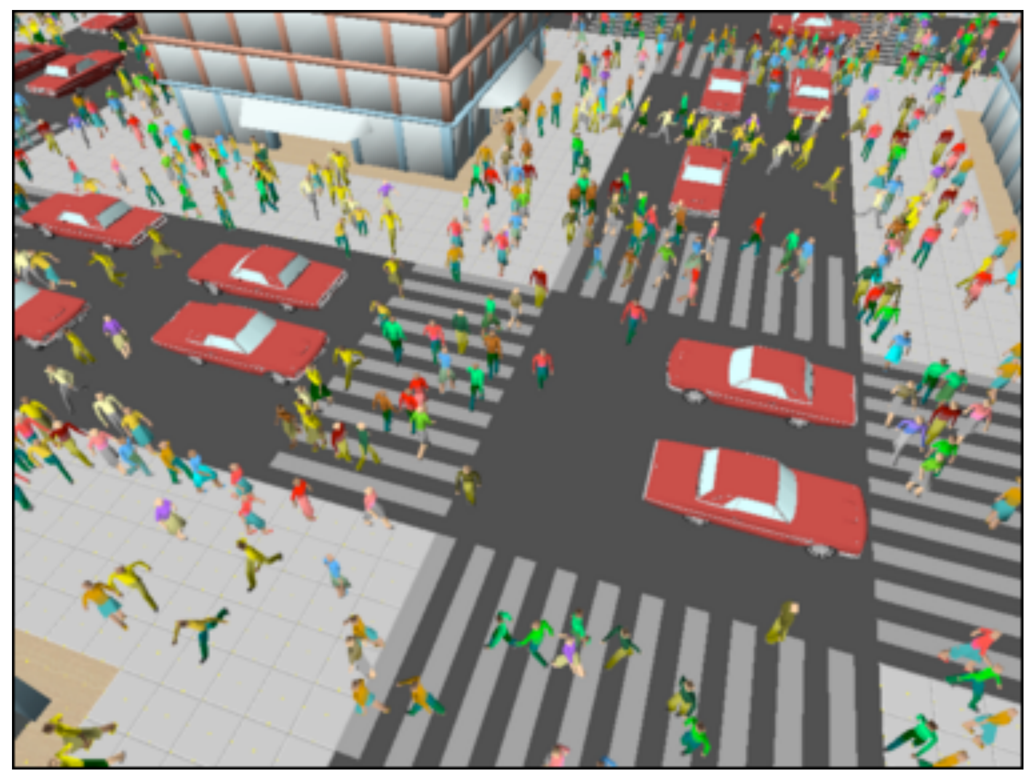
Control of virtual character

issues: control algorithms, interaction with environment rigid/deformable simulator in Pixar's *WALL-E*



rigid/deformable simulator in Pixar's *WALL-E*

Crowd simulation

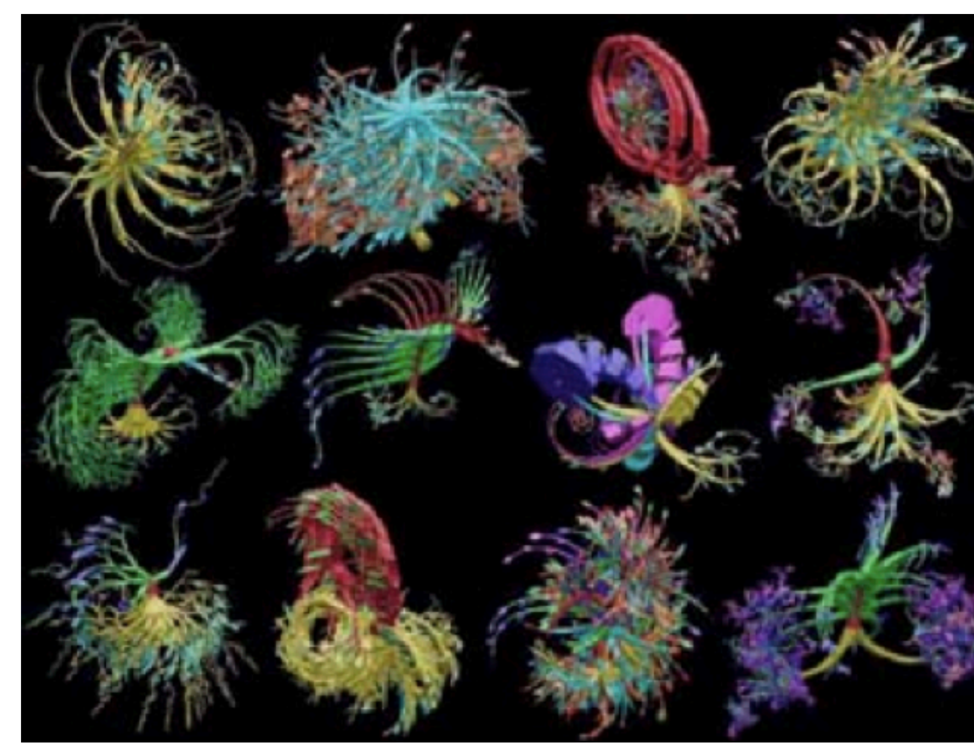


Treuille et al., 2006

- agent-based, model behavior
- also, “global effects” - e.g., incompressibility
- emergent phenomena

Artificial life

- plants - movement and growth
- evolving artificial life



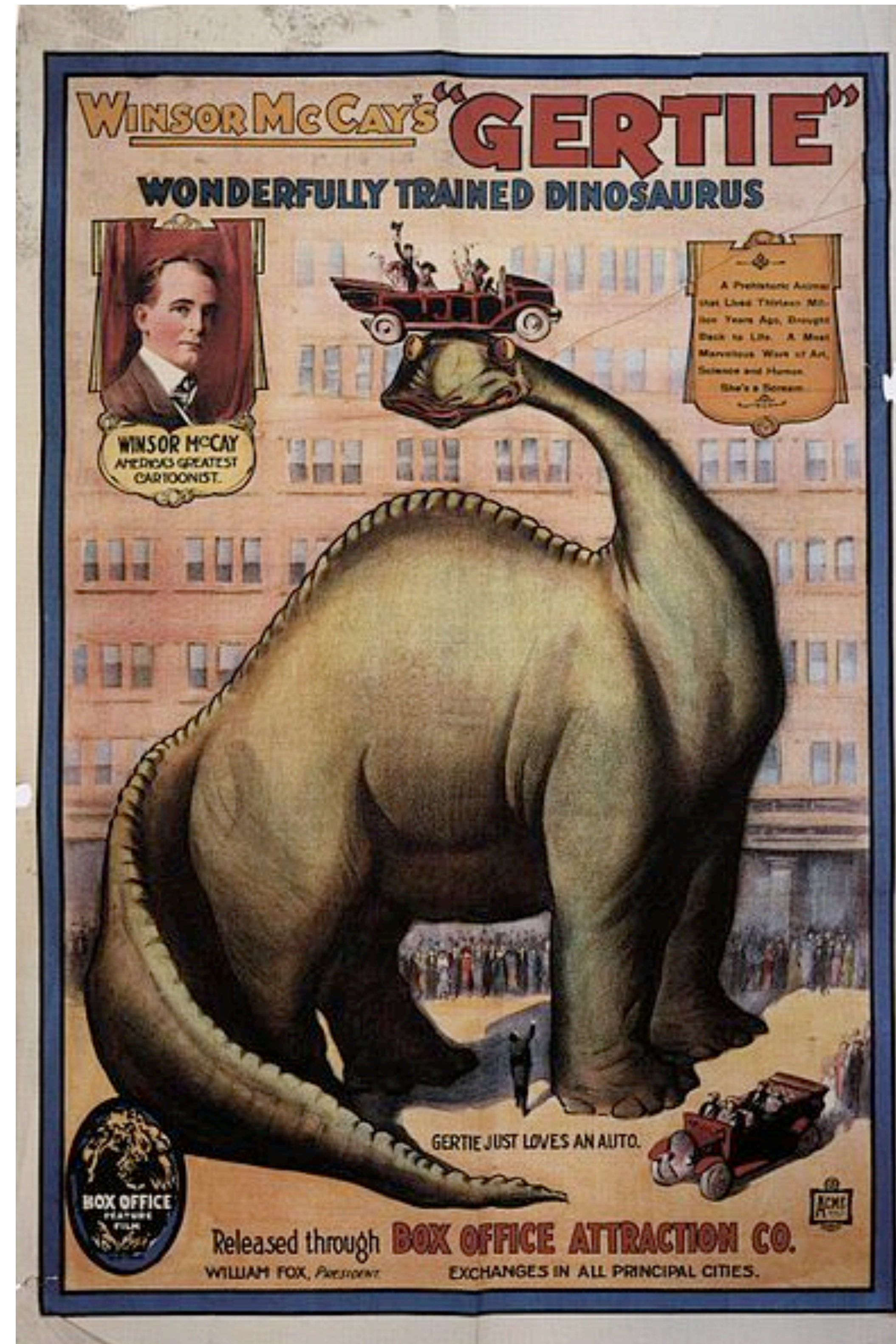
virtual worlds, special effects, games

history

Gertie the Dinosaur

1914
12 minutes
hand drawn
keyframe animation
registration
cycling

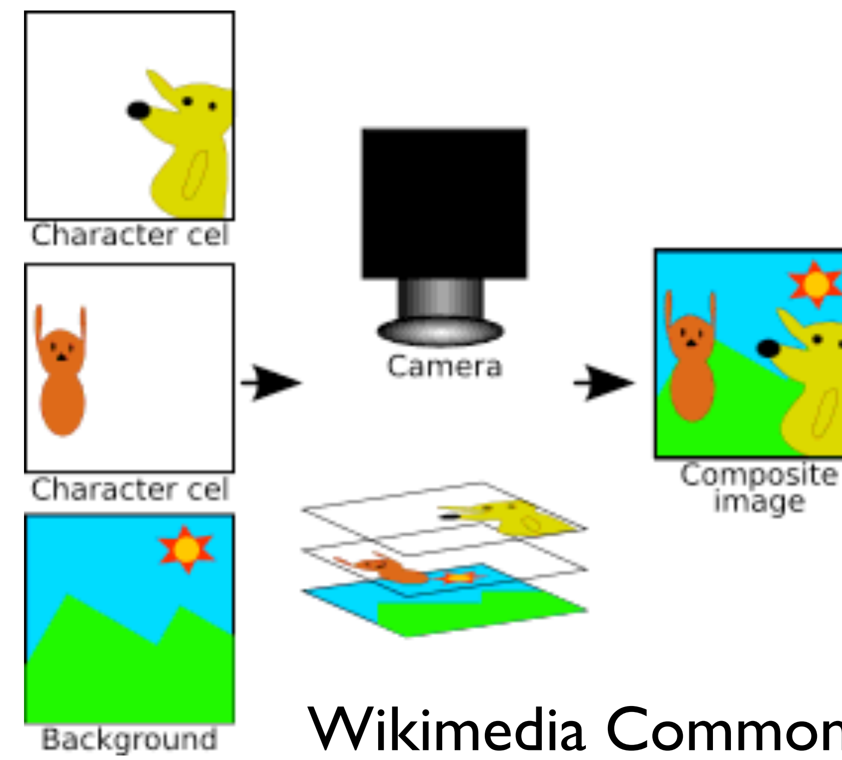
link



Traditional animation

- Cels
- Multiplane camera

Sleeping Beauty, Disney, 1959



Wikimedia Commons

Realistic 3D animation

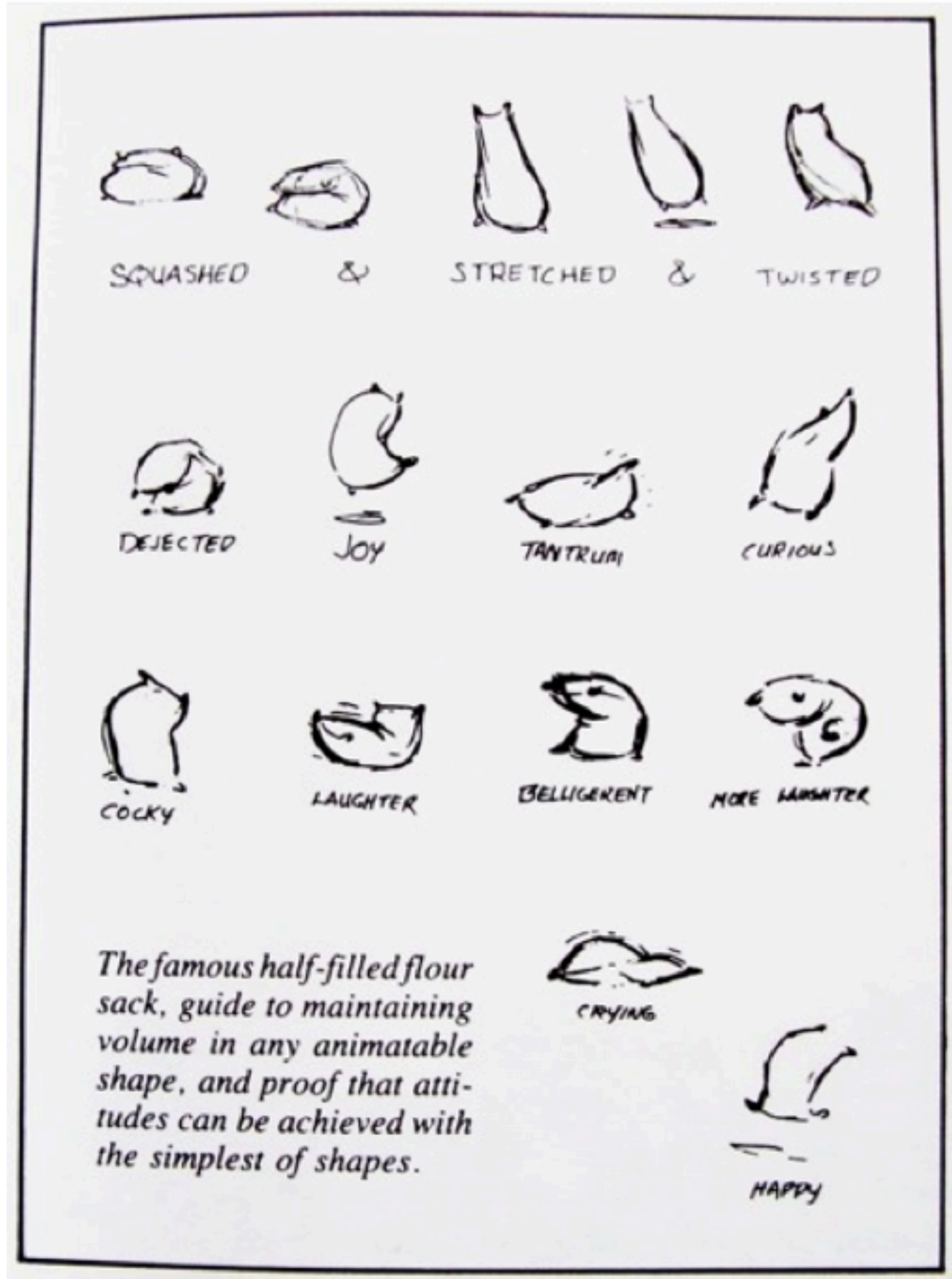
- Disney's Tron, 1981
- Pixar's Toy Story, 1995, first 3D feature

Performance Capture

- Final Fantasy 2001
- Lord of the Rings 2001
- Beowulf 2007
- Avatar, 2009
- Adventures of Tintin, 2011

Lord of the Rings, 2001

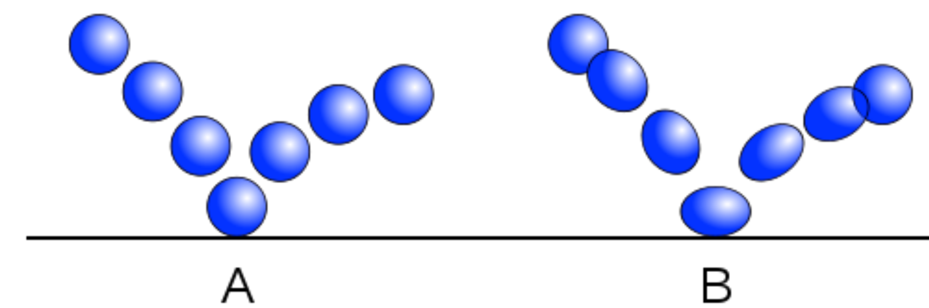
animation principles



- animation can bring even a flour sack to life
- animations principles common to any type of animation

12 principles of animation

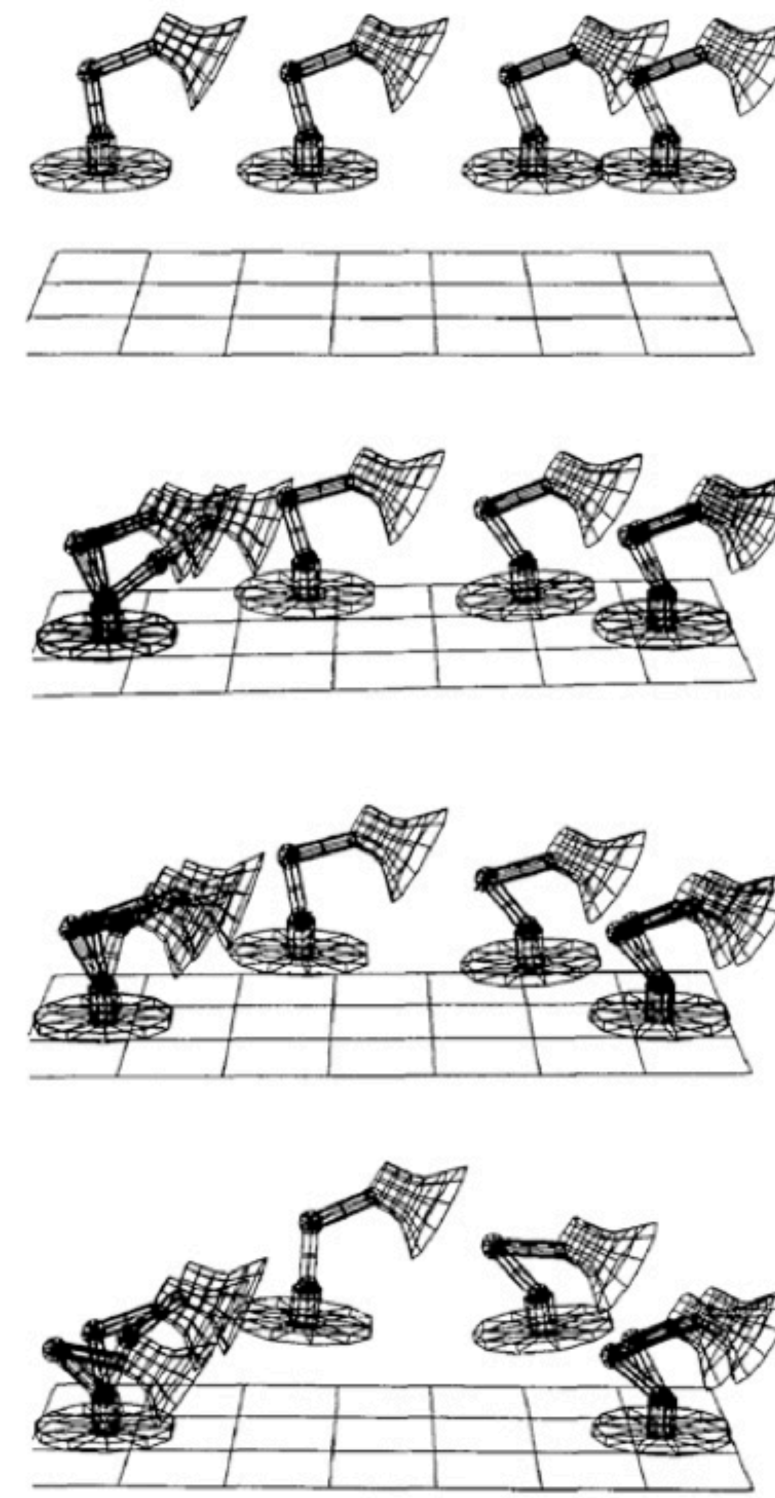
1. Squash and stretch
2. Anticipation
3. Staging
4. Straight ahead action and pose to pose
5. Follow through and overlapping action
6. Slow in and slow out
7. Arcs
8. Secondary action
9. Timing
10. Exaggeration
11. Solid drawing
12. Appeal



principles are related to the underlying physics of motion
timing: important information. ease in/ease out

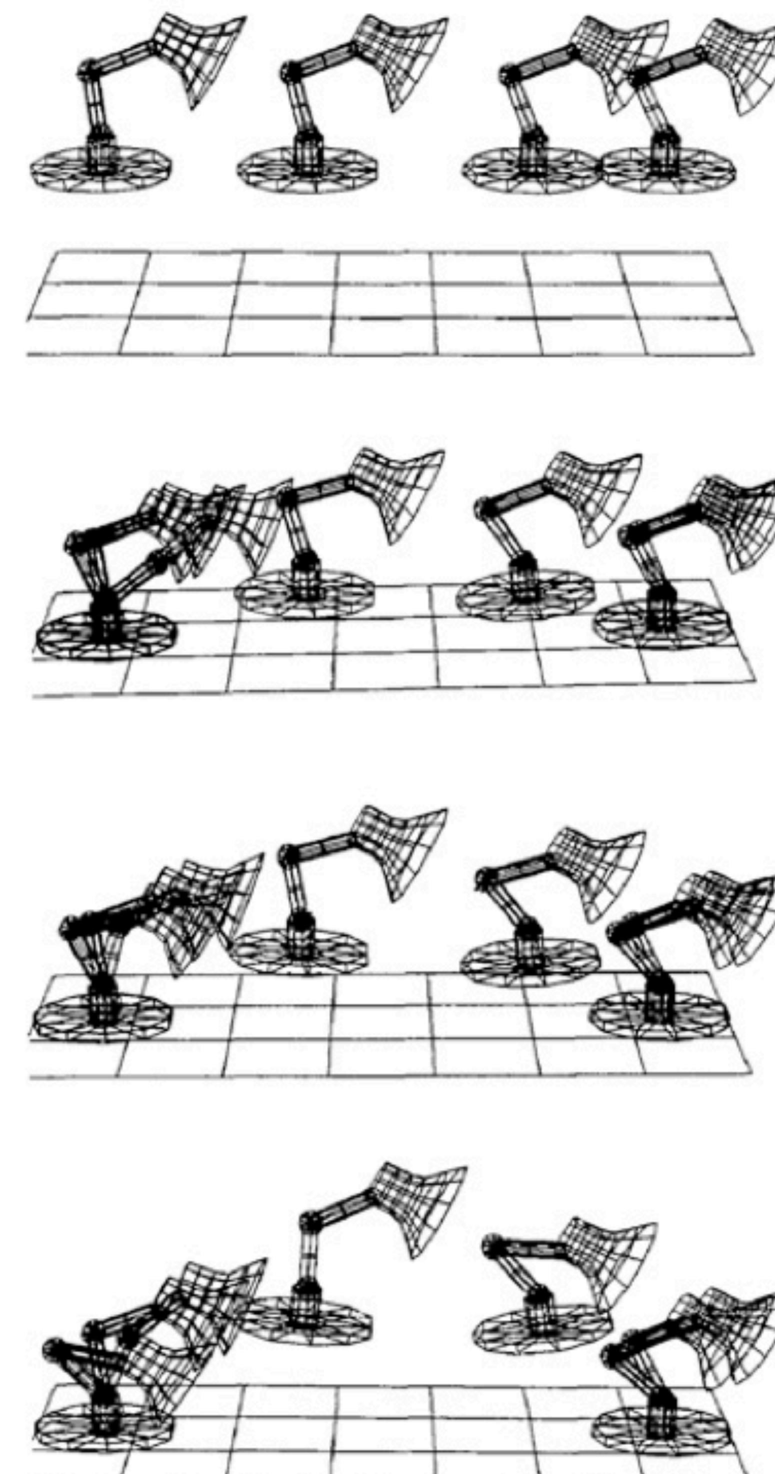
Physics-based animation

- Many animation principles follow from underlying physics
  - anticipation, follow through, secondary action, squash and stretch, ...
- *Spacetime Constraints*, Witkin and Kass 1988



Physics-based animation

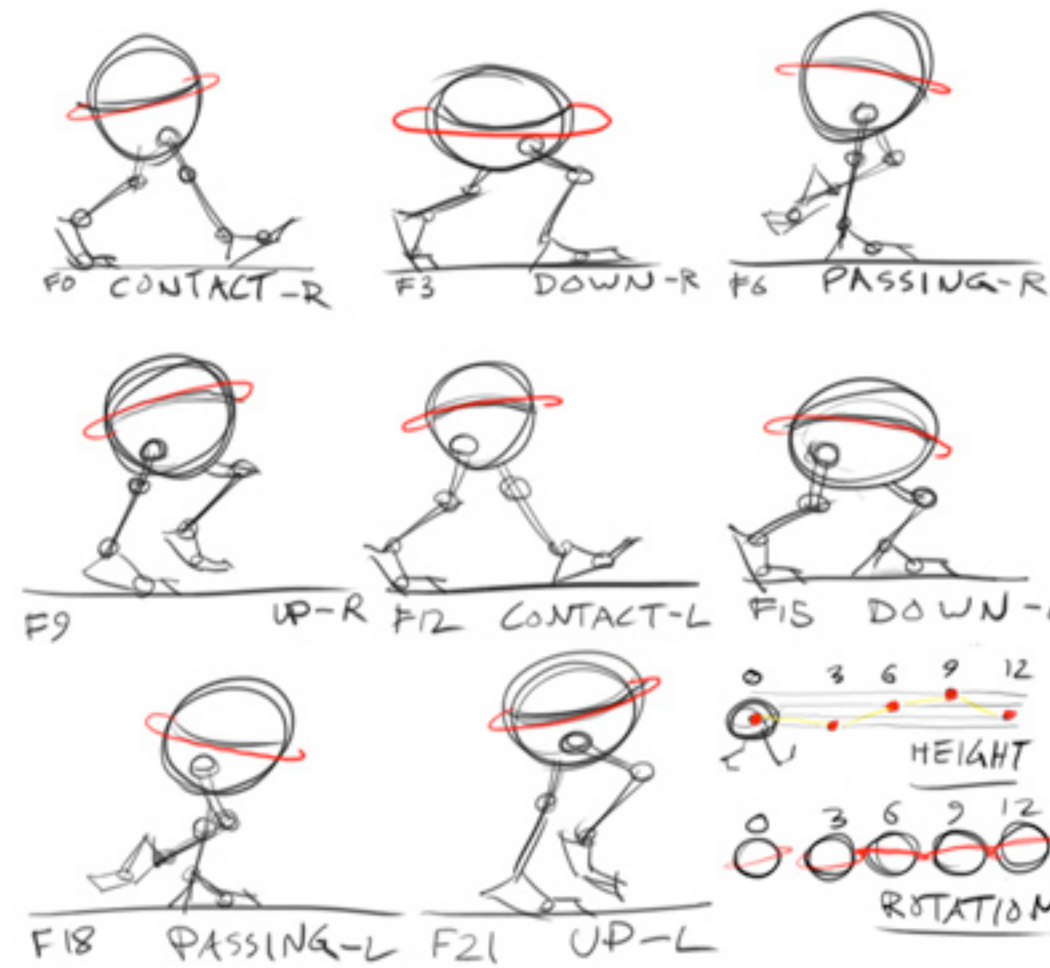
- Many animation principles follow from underlying physics
    - anticipation, follow through, secondary action, squash and stretch, ...
- *Spacetime Constraints*, Witkin and Kass 1988



keyframe animation

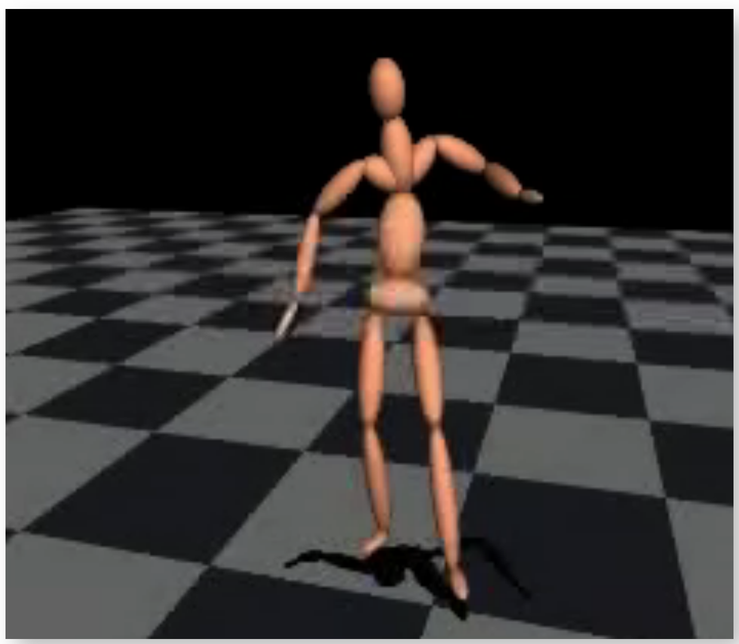
Keyframe animation

- draw a series of poses
- fill in the frames in between ("inbetweening")
  - computer animation uses interpolation



http://anim.tmog.net

Keyframe character DOFs

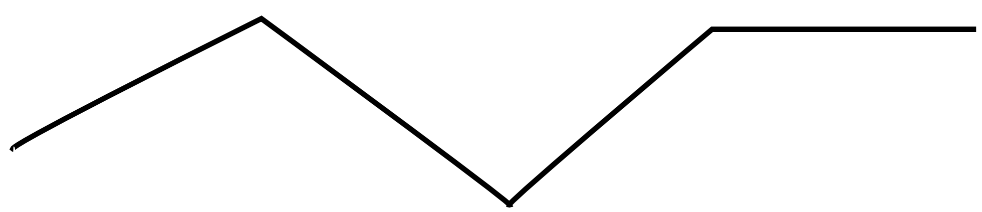


3 translational DOFs

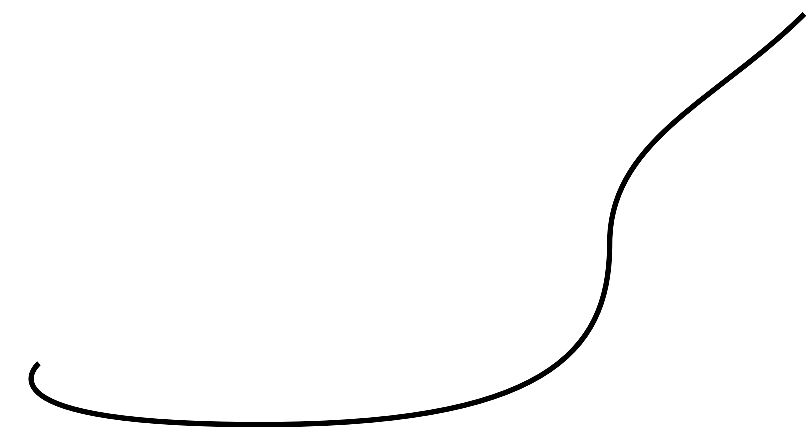
48 rotational DOFs

Each joint can have up to 3 DOFs

Interpolation of keyframes



linear interpolation



spline interpolation

Straightforward to interpolate position but what about orientation?

general rotations

Rotation

$$\begin{bmatrix} x' \\ y' \\ z' \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\theta & -\sin\theta & 0 \\ 0 & \sin\theta & \cos\theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

X axis

$$\begin{bmatrix} x' \\ y' \\ z' \\ 1 \end{bmatrix} = \begin{bmatrix} \cos\theta & 0 & \sin\theta & 0 \\ 0 & 1 & 0 & 0 \\ -\sin\theta & 0 & \cos\theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

Y axis

$$\begin{bmatrix} x' \\ y' \\ z' \\ 1 \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta & 0 & 0 \\ \sin\theta & \cos\theta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

Z axis

The rows and columns are orthonormal

Rotation about an arbitrary axis

Rotating about an axis by theta degrees

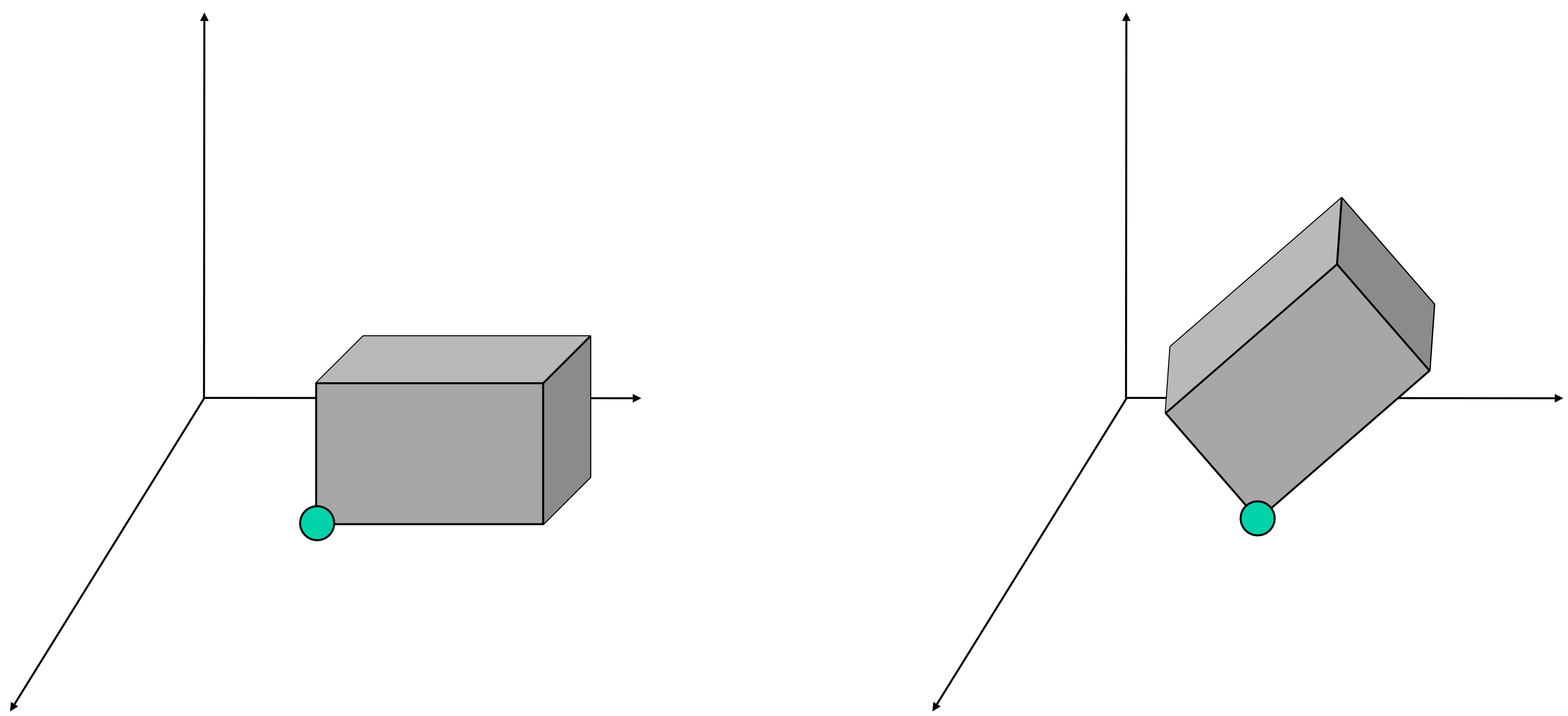
- Rotate about x to bring axis to xz plane
- Rotate about y to align axis with z -axis
- Rotate theta degrees about z
- Unrotate about y, unrotate about x

$$M = Rx^{-1}\ Ry^{-1}\ Rz(\theta)\ Ry\ Rx$$

- Can you determine the values of Rx and Ry?

Composite Transformations

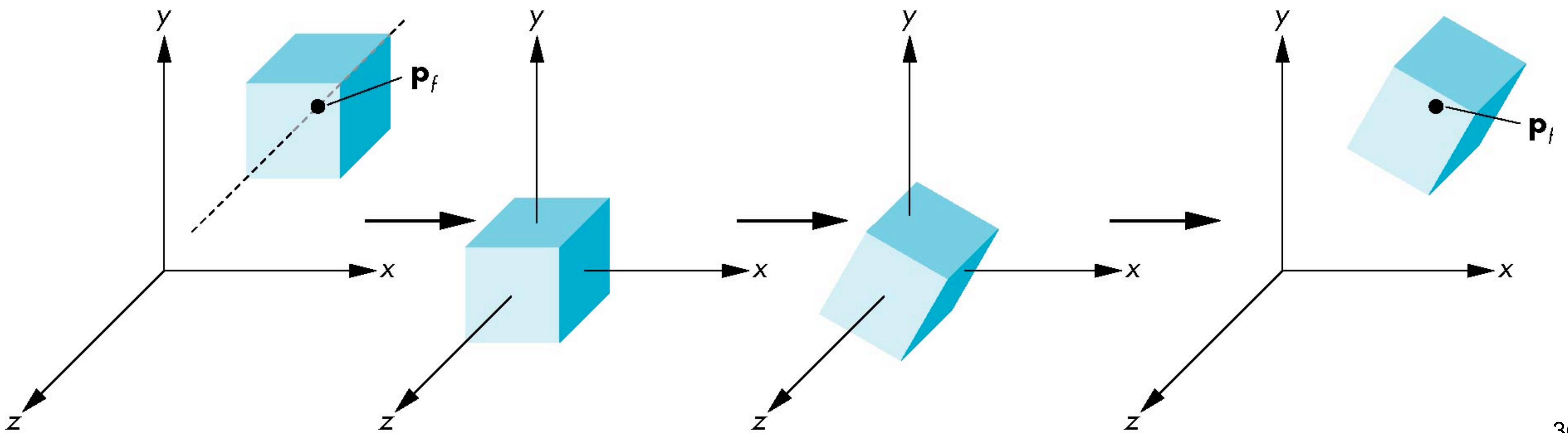
- Rotating about a fixed point
  - **basic** rotation alone will rotate about origin but we want:



Composite Transformations

- Rotating about a fixed point
  - Move fixed point (px,py,pz) to origin
  - Rotate by desired amount
  - Move fixed point back to original position

$$\mathbf{M} = \mathbf{T}(px, py, pz)\, \mathbf{R_z}(\theta)\, \mathbf{T}(-px, -py, -pz)$$



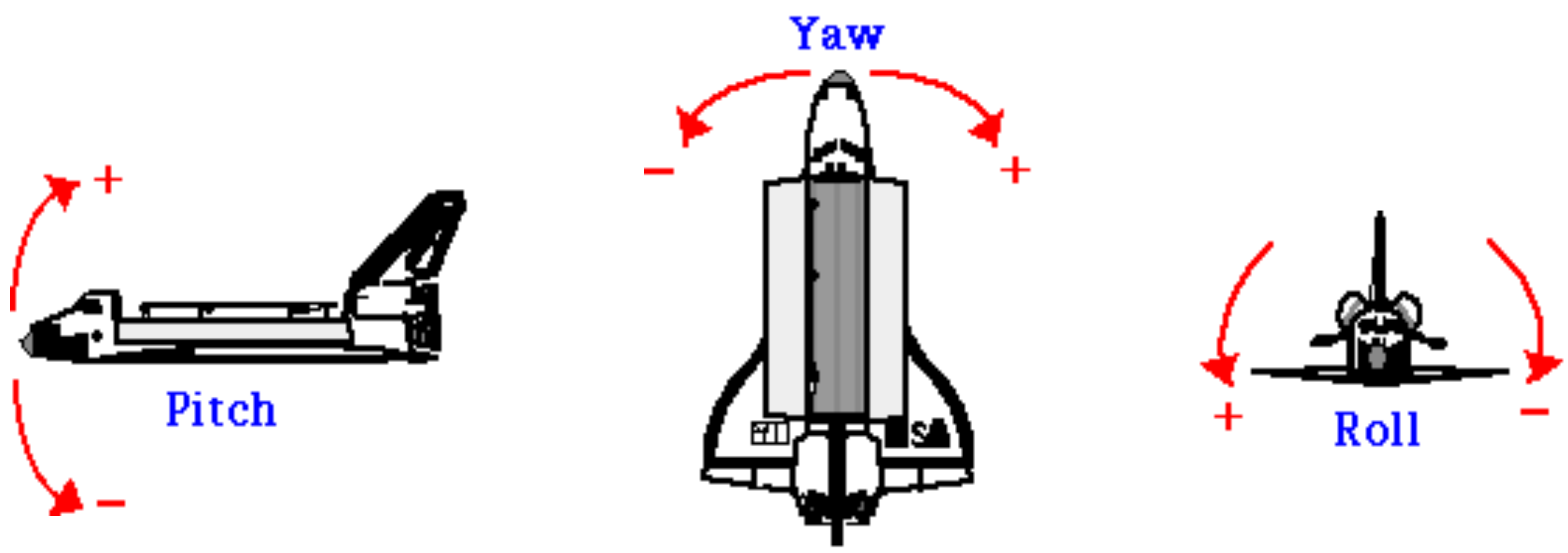
Euler’s Rotation Theorem

Any displacement of a rigid body such that a point on the rigid body remains fixed, can be described as a rotation by some angle about some axis

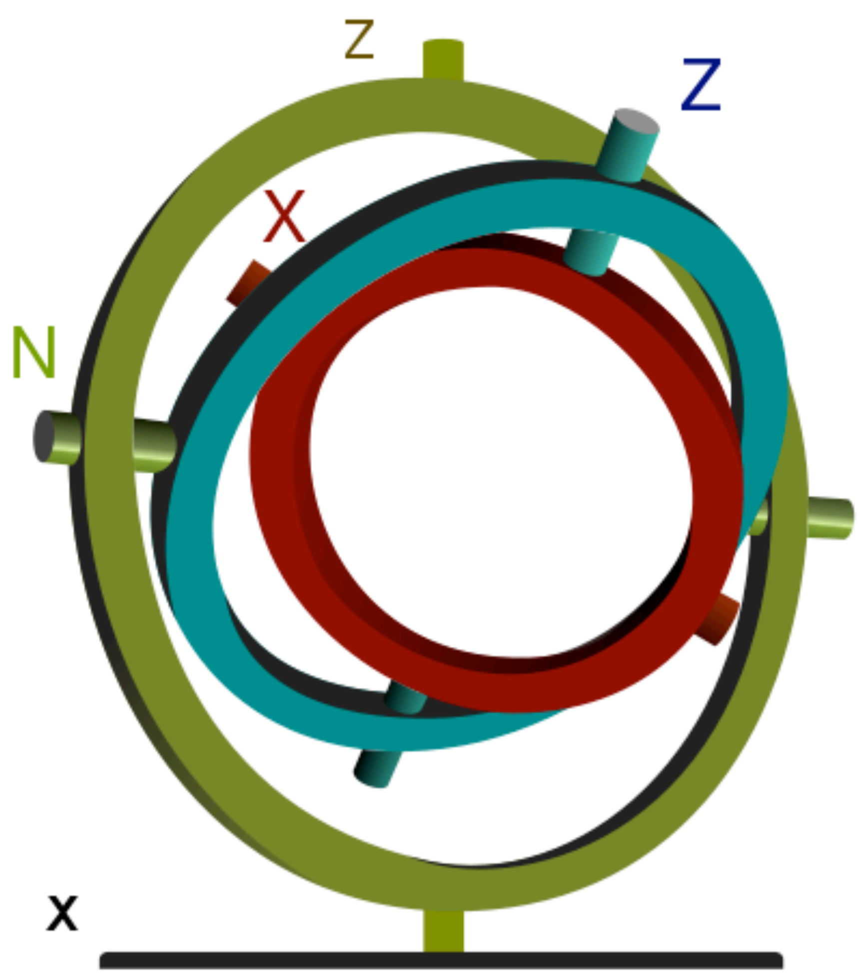
euler angles

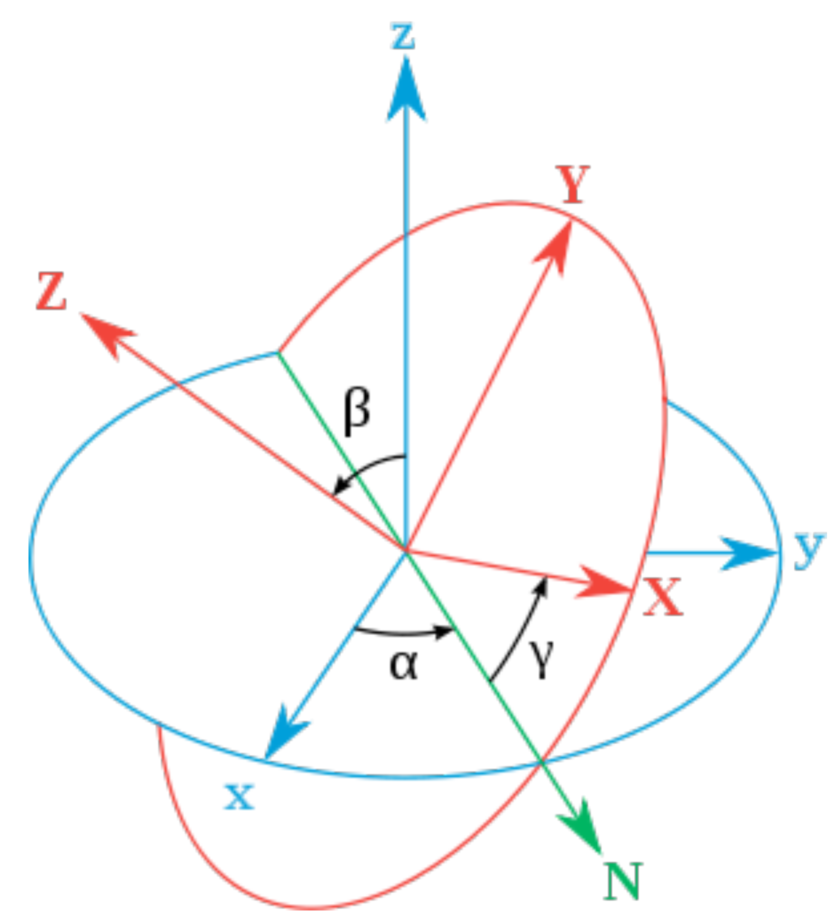
Euler Angles

- A general rotation is a combination of three elementary rotations: around the x-axis (x-roll) , around the y-axis (y-pitch) and around the z-axis (z-yaw).



Gimbal and Euler Angles

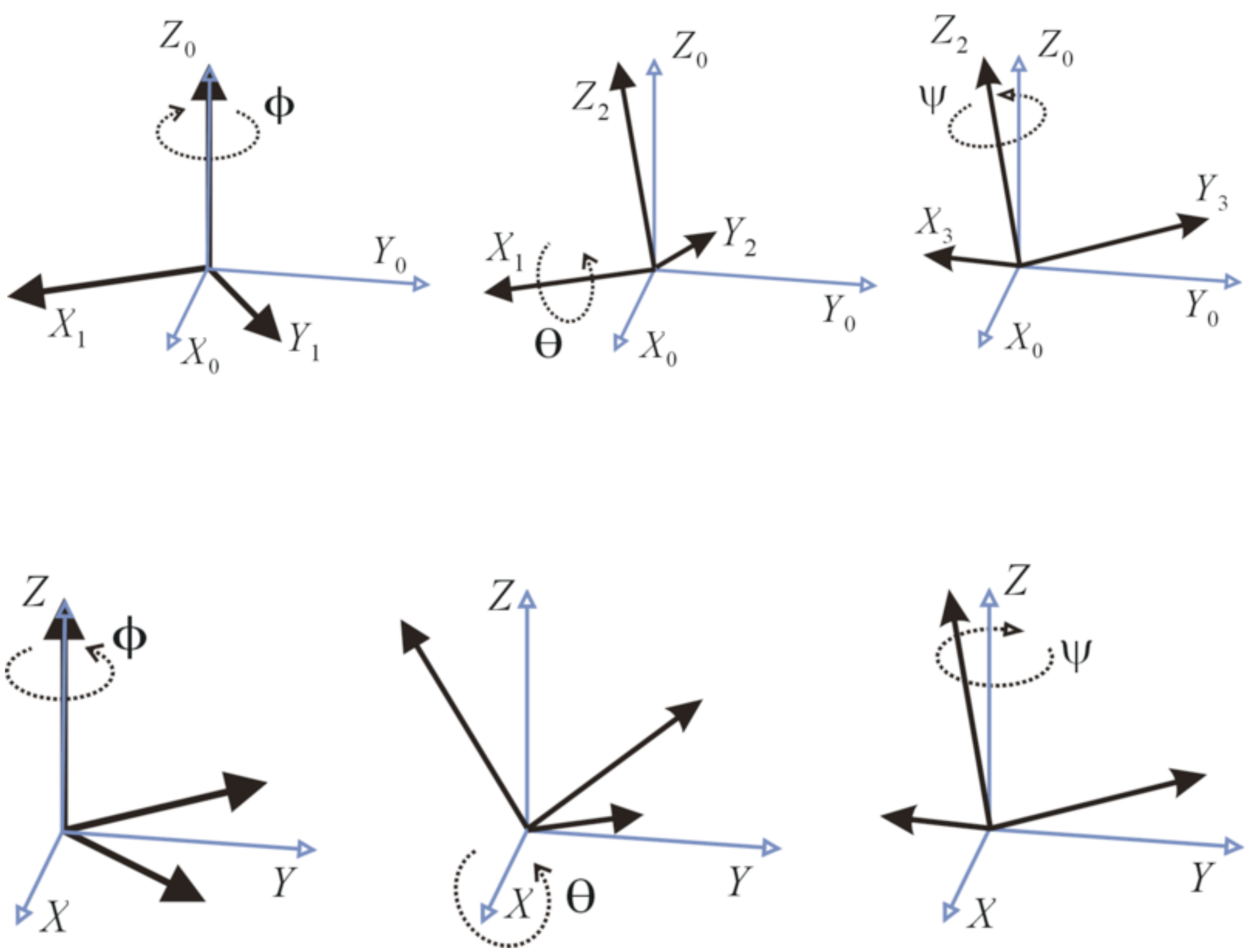


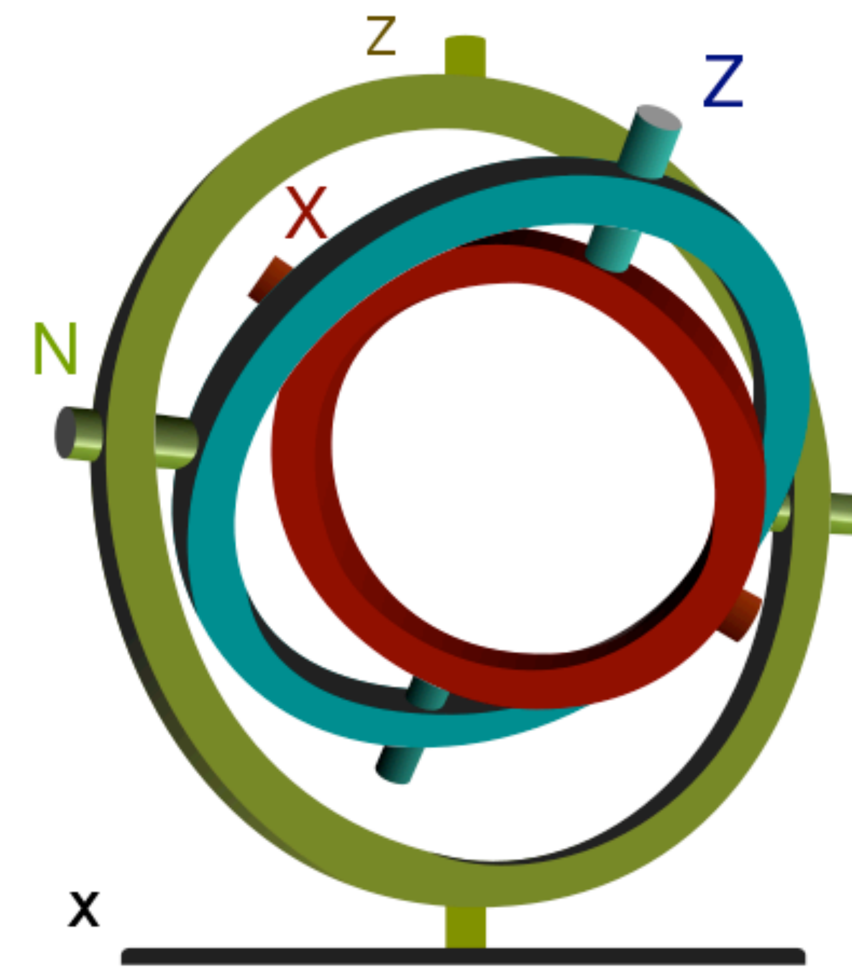


Z-X'-Z"

Wikimedia Commons

Extrinsic vs. Intrinsic rotations





Wikimedia Commons

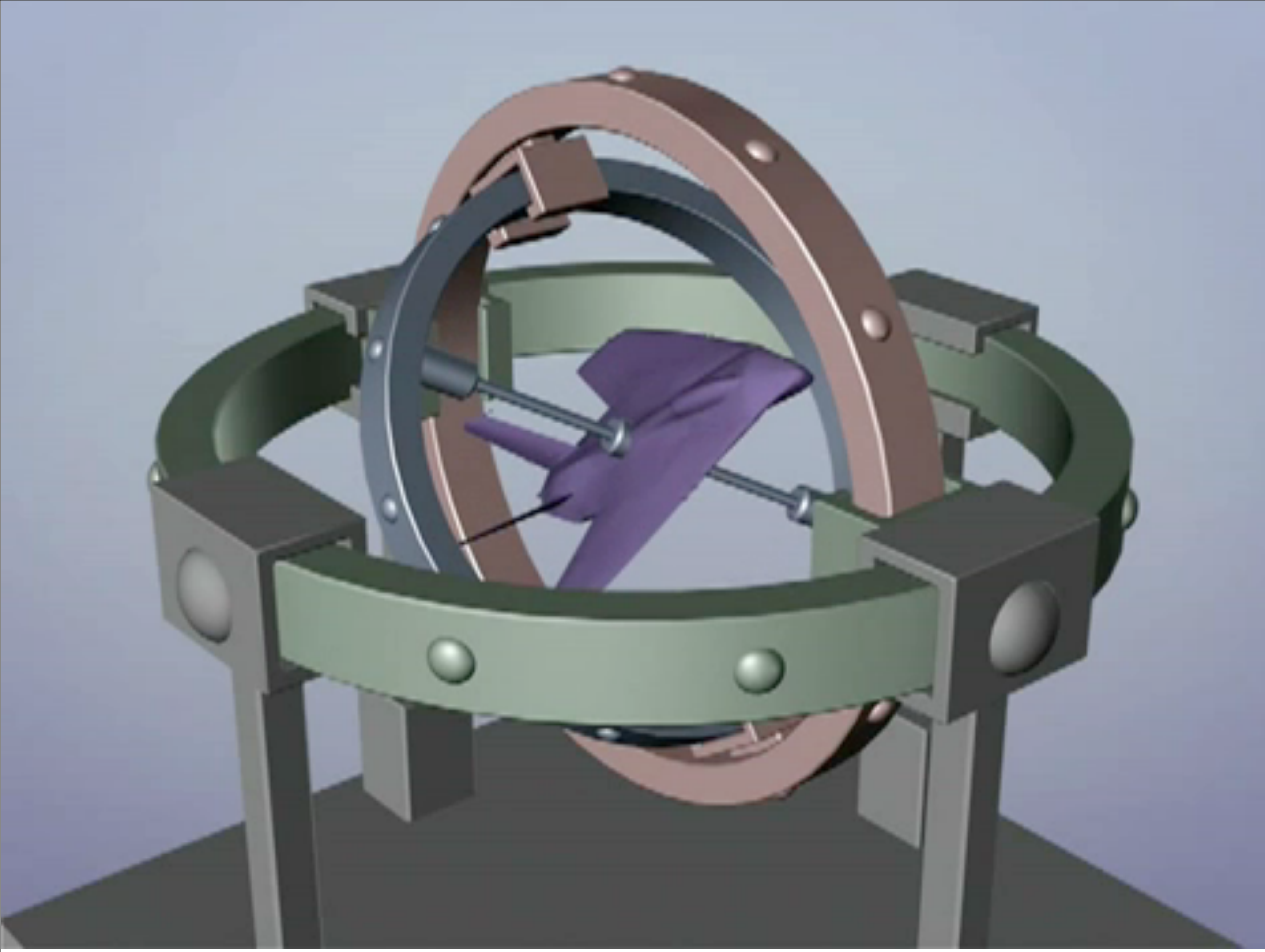
Euler Angles and Rotation Matrices

$$x-roll(\theta_1) = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\theta_1 & \sin\theta_1 & 0 \\ 0 & -\sin\theta_1 & \cos\theta_1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

$$\mathrm{y\text{-}pitch}(\theta_2) = \begin{pmatrix} \cos\theta_2 & 0 & -\sin\theta_2 & 0 \\ 0 & 1 & 0 & 0 \\ \sin\theta_2 & 0 & \cos\theta_2 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

$$z-yaw(\theta_3) = \begin{pmatrix} \cos\theta_3 & \sin\theta_3 & 0 & 0 \\ -\sin\theta_3 & \cos\theta_3 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

$$R(\theta_1,\theta_2,\theta_3) = \begin{pmatrix} c_2c_3 & c_2s_3 & -s_2 & 0 \\ s_1s_2c_3 - c_1s_3 & s_1s_2s_3 + c_1c_3 & s_1c_2 & 0 \\ c_1s_2c_3 + s_1s_3 & c_1s_2s_3 - s_1c_3 & c_1c_2 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$



quaternions

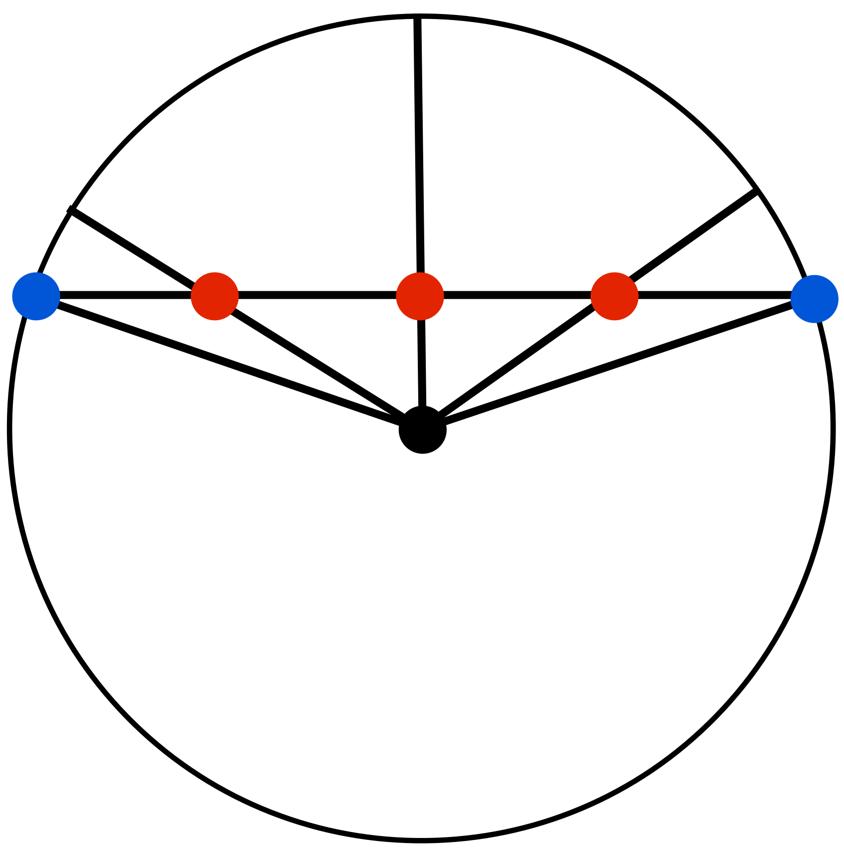

Here as he walked by
on the 16th of October 1843
Sir William Rowan
in a flash of genius discovered
the fundamental formula for
quaternion multiplication
$i^2 = j^2 = k^2 = ijk = -1$
on a stone

Quaternions

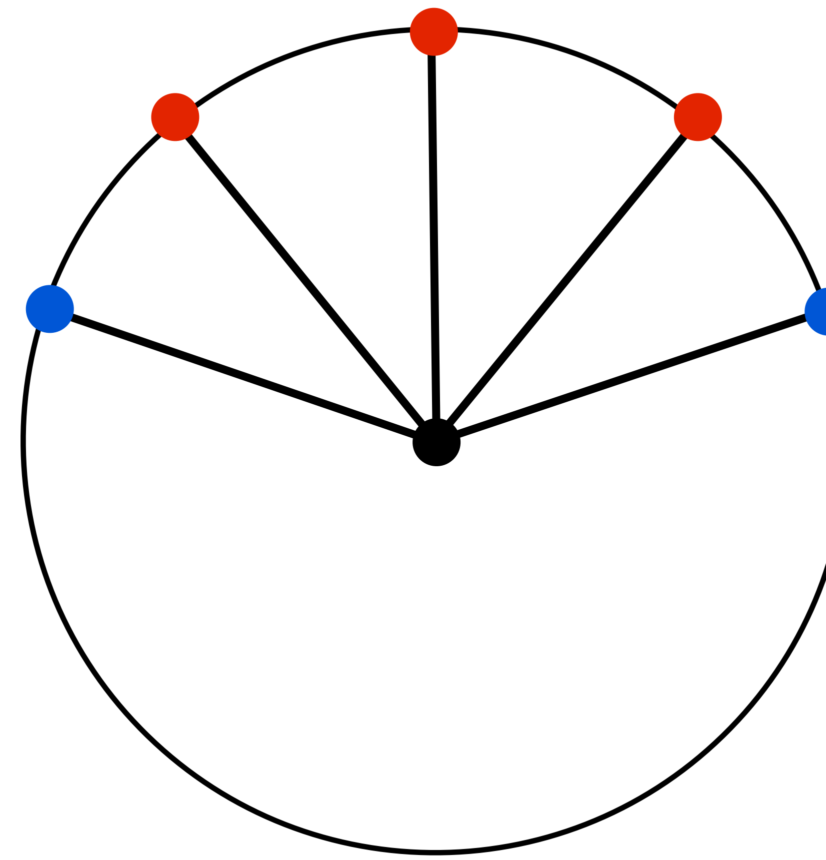
- axis/angle representation
- interpolates smoothly
- easy to compose

<whiteboard>

Quaternion Interpolation



linear

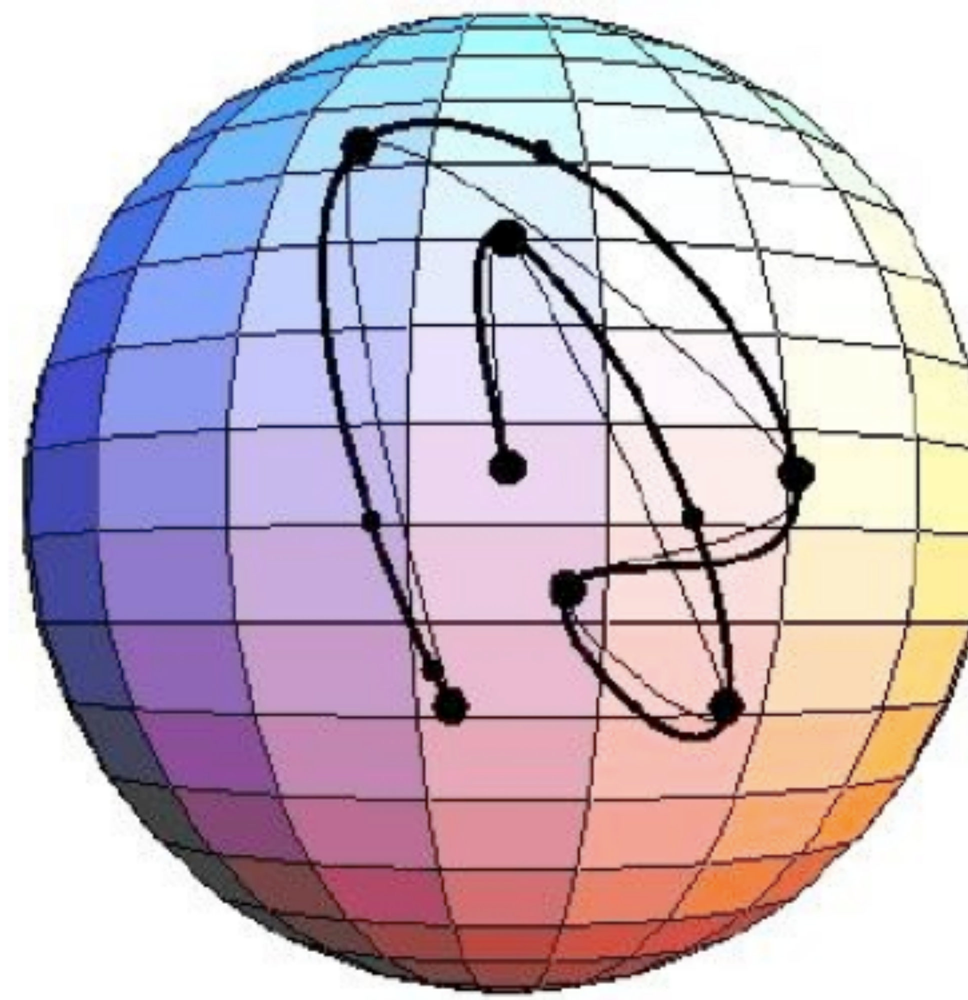


spherical linear
"slerp"

linear: treat quaternions as 4-vectors, note non-uniform speed
spherical linear: constant speed

Higher order interpolation

- Bezier curve
- Shoemake, *Animating rotation with quaternion curves*, 1985



Matrix form

$$\mathbf{q} = \begin{bmatrix} w \\ x \\ y \\ z \end{bmatrix}$$

$$\mathbf{R}(\mathbf{q}) = \begin{bmatrix} 1-2y^2-2z^2 & 2xy+2wz & 2xz-2wy & 0 \\ 2xy-2wz & 1-2x^2-2z^2 & 2yz+2wx & 0 \\ 2xz+2wy & 2yz-2wx & 1-2x^2-2y^2 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Rotations in Reality

- It's easiest to express rotations in Euler angles or Axis/angle

- We can convert to/from any of these representations

- Choose the best representation for the task
  - input:Euler angles
  - interpolation: quaternions
  - composing rotations: quaternions, orientation matrix