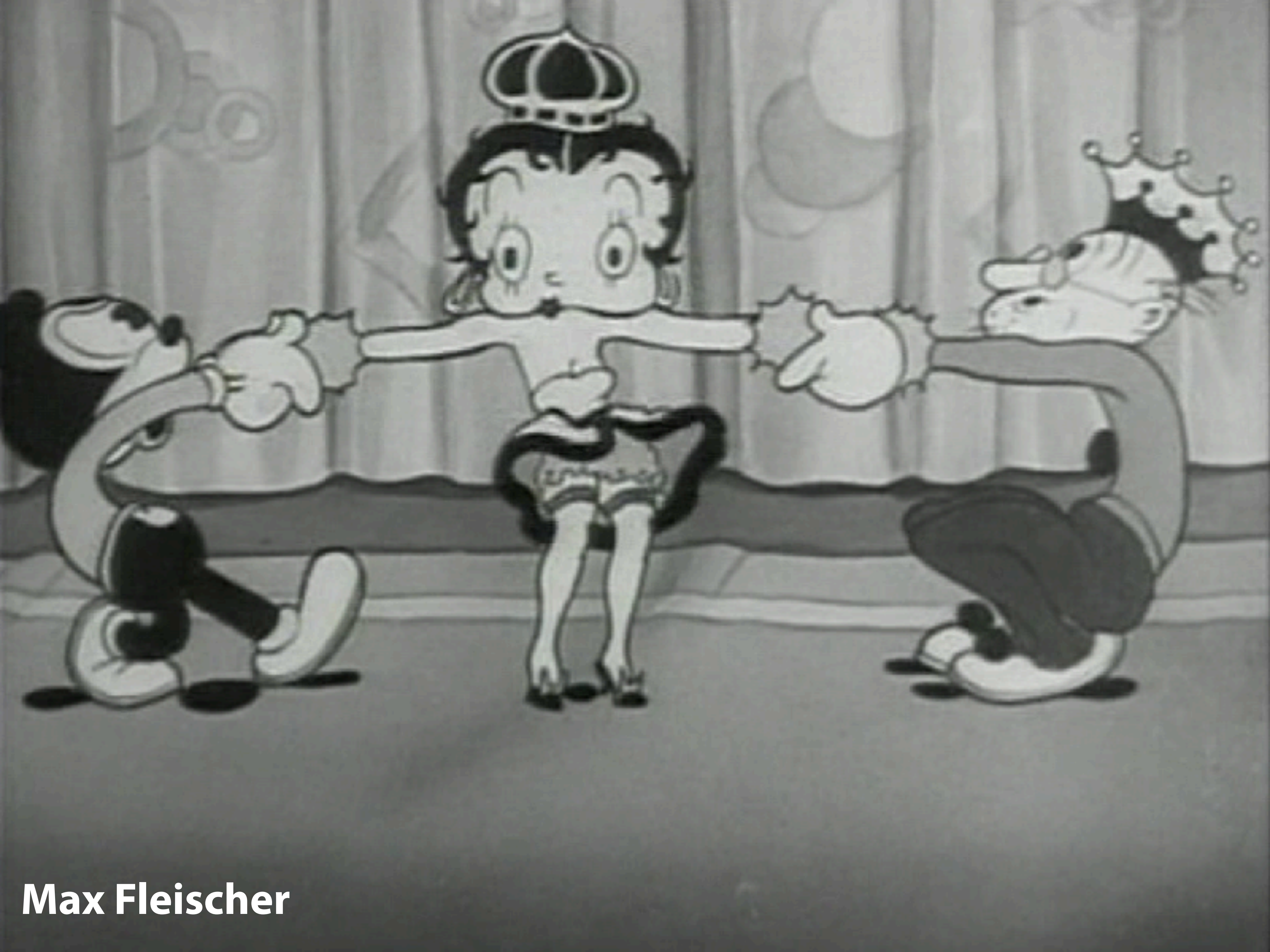
Random International, *Study for 15 Points*

Storyboard $\longrightarrow$ Keyframes $\longrightarrow$ In-betweens

Storyboard ⟶ Keyframes ⟶ In-betweens

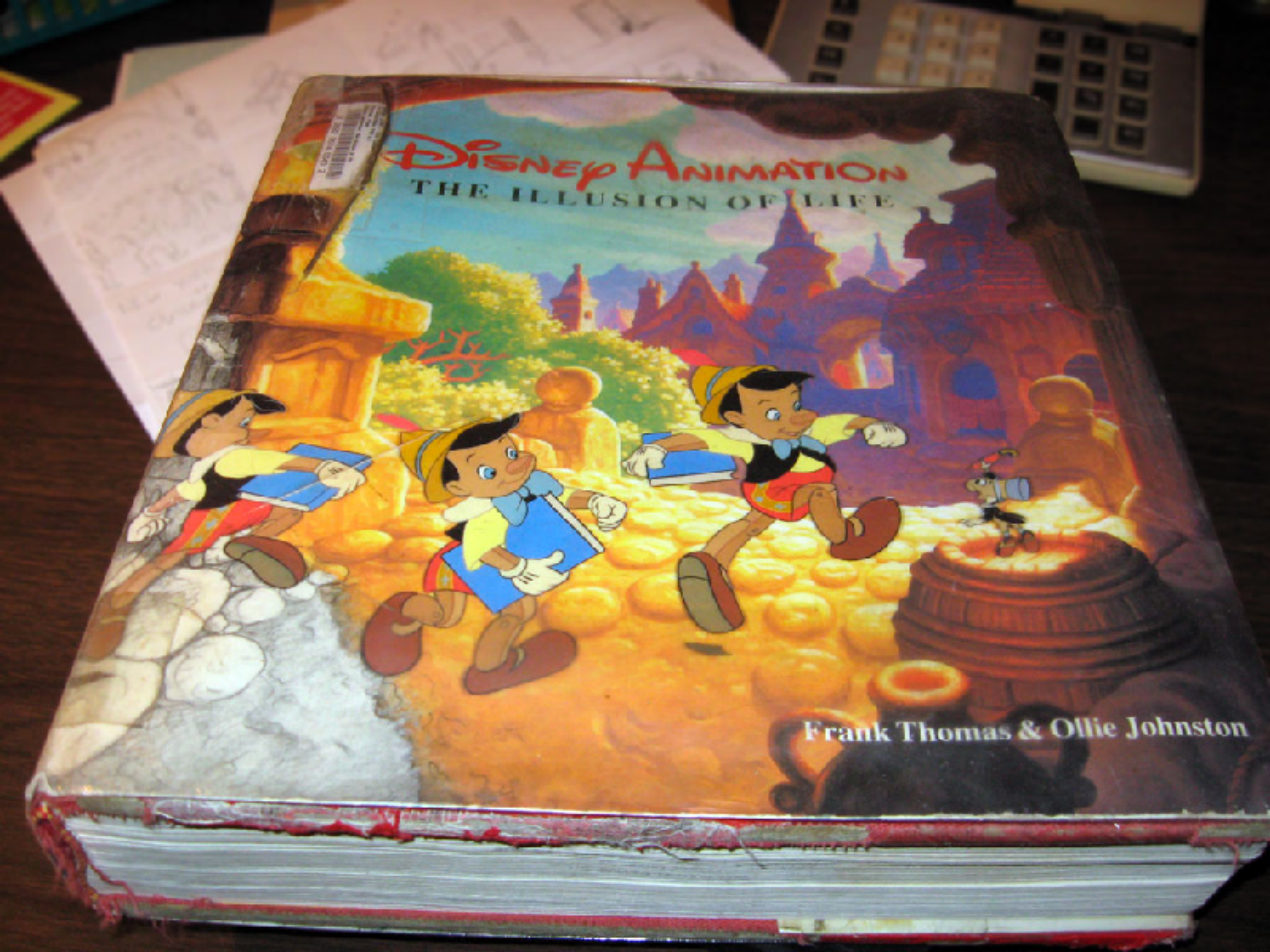
The animation historians at Disney say that “at most animation studios, the best animators only sketched a few animation drawings, leaving gaps in between. Later on, a person called an "inbetweener" would finish the scenes by drawing in between the areas that the animator had left.” Well, not much has changed about this key position.

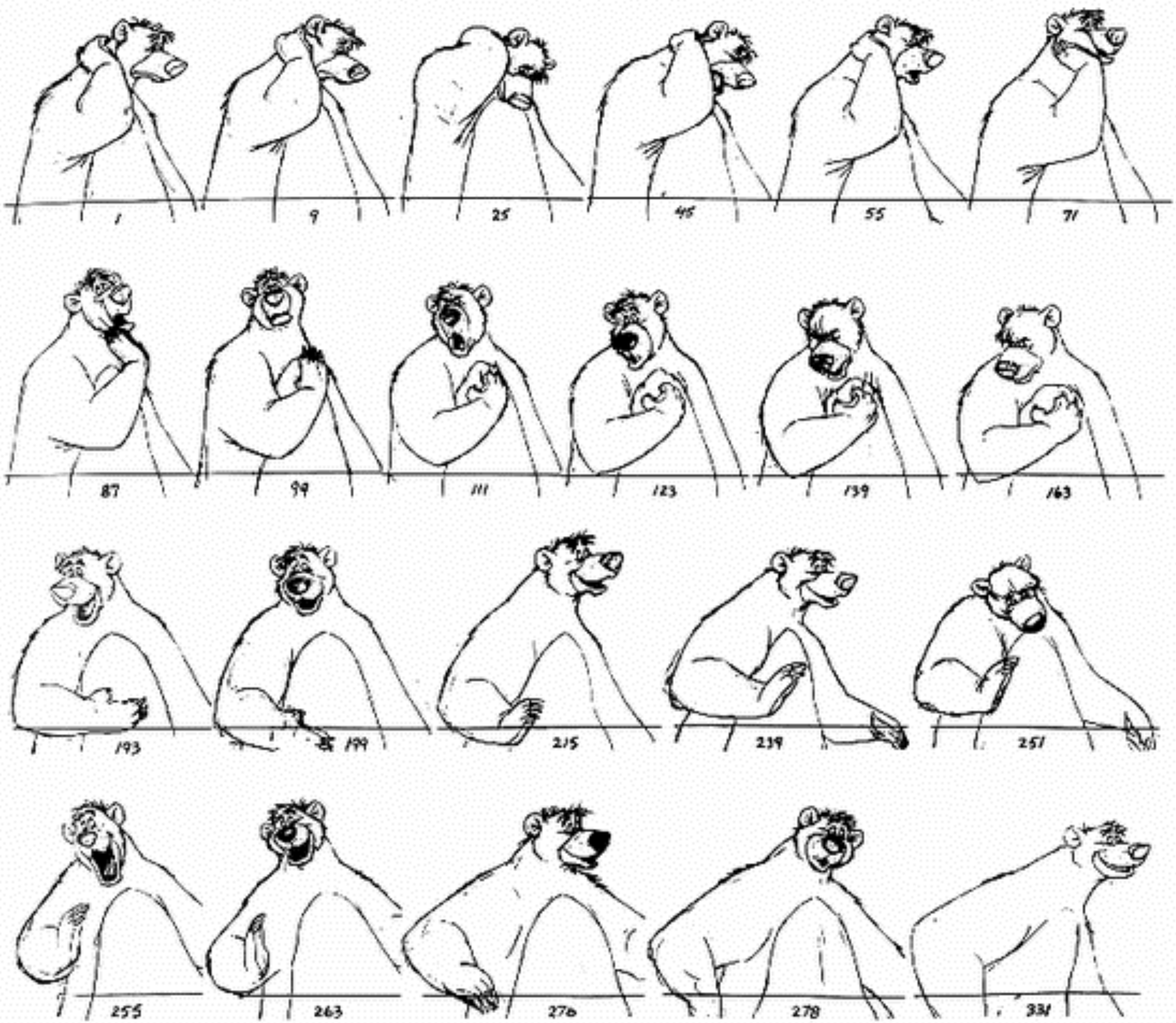
www.animationcareerreview.com



Max Fleischer

Max Fleischer


Disney Animation
THE ILLUSION OF LIFE
Frank Thomas & Ollie Johnston


1
9
25
45
55
71
87
99
111
123
139
163
193
199
215
239
251
255
263
270
278
331

PRINCIPLES OF TRADITIONAL ANIMATION APPLIED TO 3D COMPUTER ANIMATION

John Lasseter
Pixar
San Rafael
California

Pixar
San Rafael
California

ANIMATION
ANIMATION

ANIMATION
ANIMATION

Pixar

Animation principles

(stolen from the University of Washington)

Reading

Required:

- John Lasseter. Principles of traditional animation applied to 3D computer animation. Proceedings of SIGGRAPH (Computer Graphics) 21(4): 35-44, July 1987.

Recommended:

- Frank Thomas and Ollie Johnston, Disney animation: The Illusion of Life, Hyperion, 1981.
- Michael Comet tutorial (source for the ball and green bug examples in this lecture):

  http://www.comet-cartoons.com/3ddocs/charanim/index.html

Character animation

Goal: make characters that move in a convincing way to communicate personality and mood.

Walt Disney developed a number of principles.

Computer graphics animators have adapted them to 3D animation.

Animation Principles

The following are a set of principles to keep in mind:

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

Squash and stretch

Squash: flatten an object or character by pressure or by its own power.

Stretch: used to increase the sense of speed and emphasize the squash by contrast.

Note: keep volume constant!

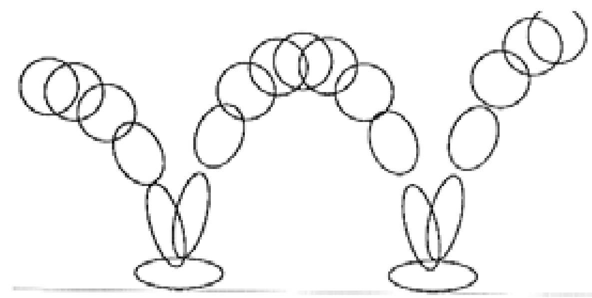


FIGURE 2. Squash & stretch in bouncing ball.

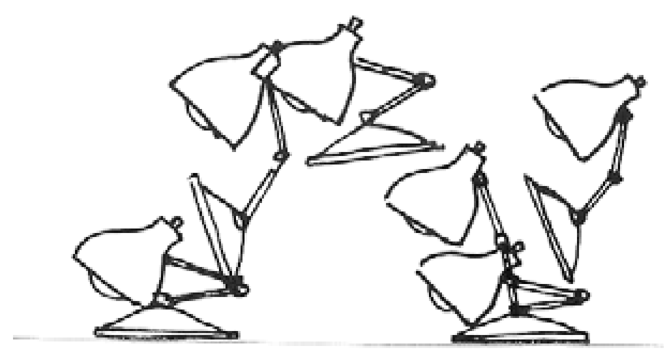


FIGURE 3. Squash & stretch in Luxo Jr.'s hop.

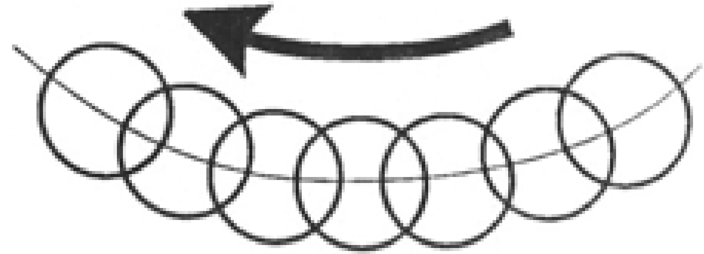


FIGURE 4a. In slow action, an object's position overlaps from frame to frame which gives the action a smooth appearance to the eye.

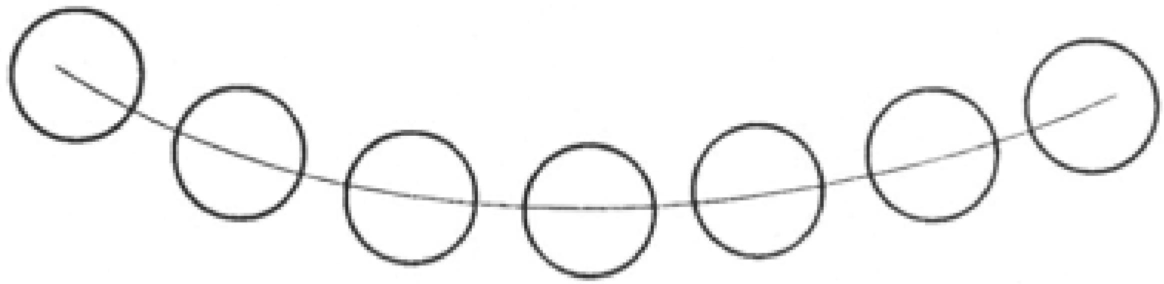


FIGURE 4b. Strobing occurs in a faster action when the object's positions do not overlap and the eye perceives seperate images.

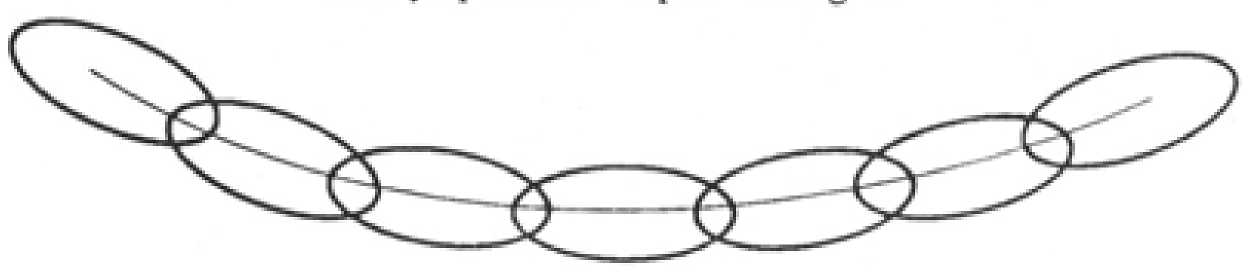
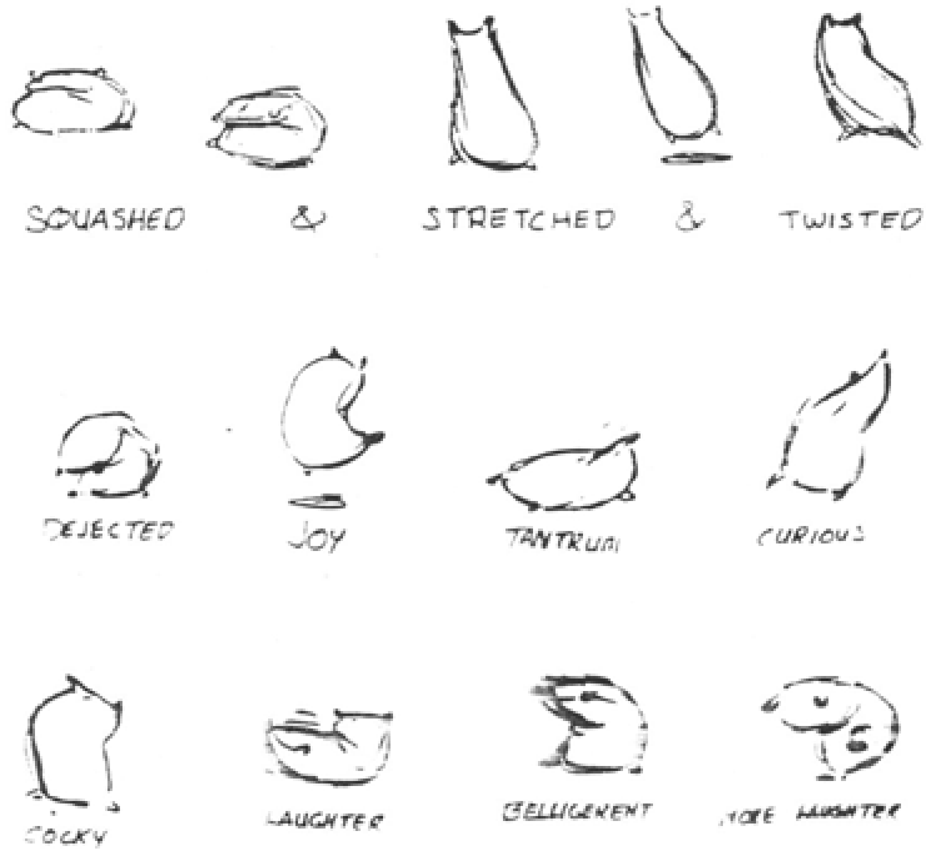


FIGURE 4c. Stretching the object so that it's positions overlap again will relieve the strobing effect.



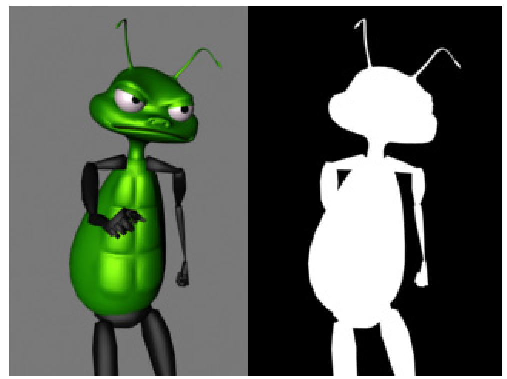
The famous half-filled flour sack, guide to maintaining volume in any animatable shape, and proof that attitudes can be achieved with the simplest of shapes.

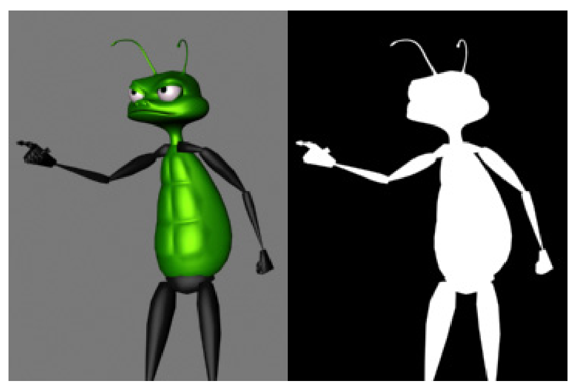
Staging

Present the idea so it is unmistakably clear.

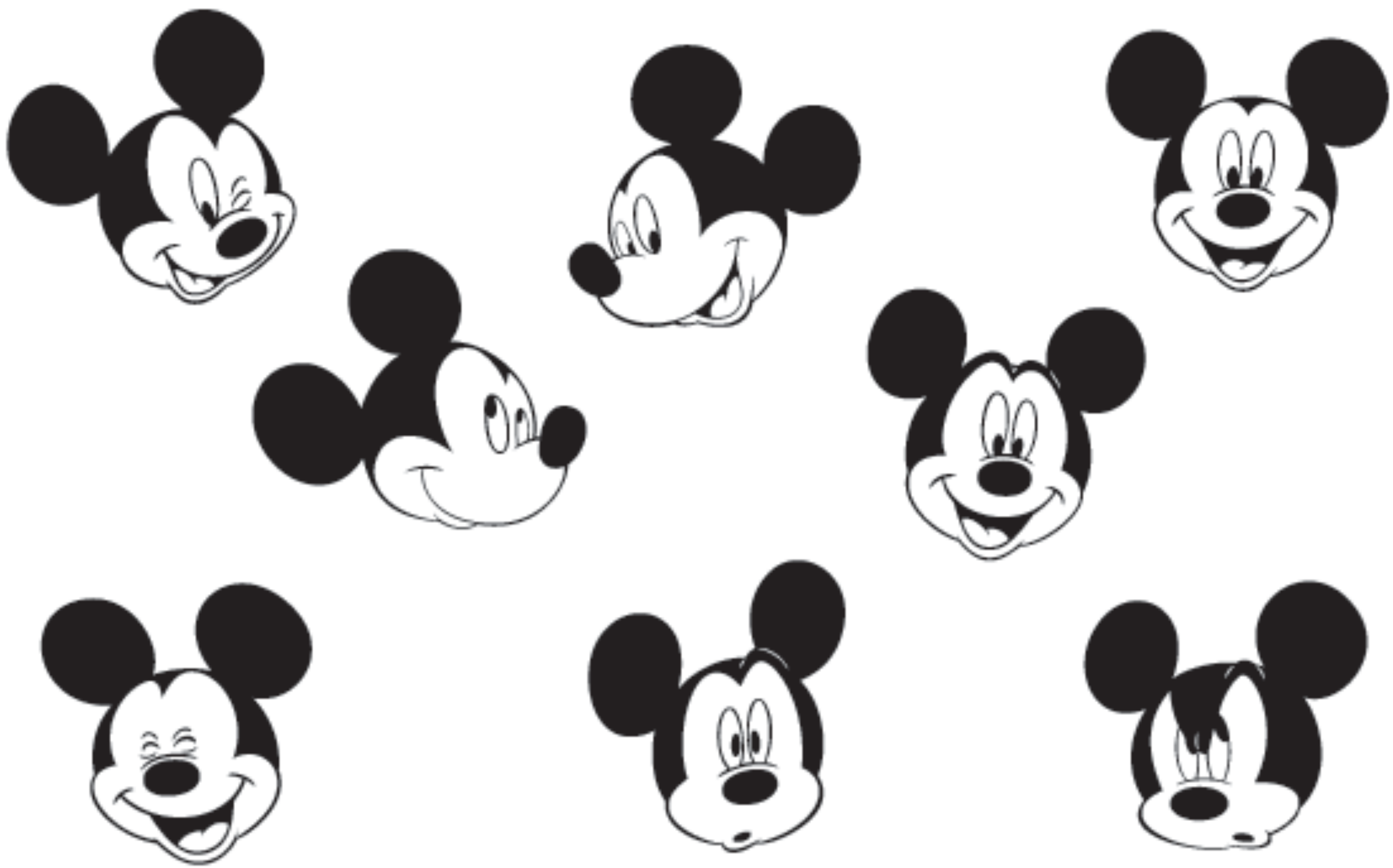
Audience can only see one thing at a time.

Useful guide: stage actions in silhouette.





In dialogue, characters face 3/4 towards the camera, not right at each other.



Timing

An action generally consists of anticipation, the action, and the reaction. Don't dwell too long on any of these.

Timing also reflects the weight of an object:

- light objects move quickly
- heavier objects move more slowly

Timing can completely change the meaning of an action.

Timing (cont'd)

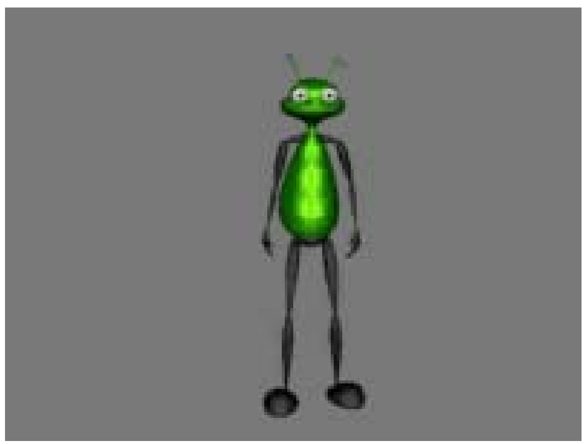
The many meanings of a simple head turn:

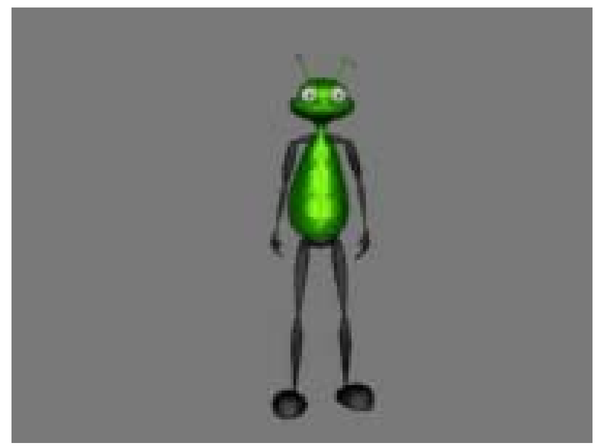
NO inbetweens	hit by a tremendous force.
ONE inbetween	hit by a brick, frying pan.
TWO inbetweens	nervous tic, muscle spasm.
THREE inbetweens	dodging a thrown brick.
FOUR inbetweens	giving a crisp order (move it!)
FIVE inbetweens	a more friendly order (c'mon!)
SIX inbetweens	sees a sportscar he always wanted
SEVEN inbetweens	trying to get a better look...
EIGHT inbetweens	searching for something on shelf
NINE inbetweens	considering thoughtfully
TEN inbetweens	stretching a sore muscle

Anticipation

An action has three parts: anticipation, action, reaction.

Anatomical motivation: a muscle must extend before it can contract.



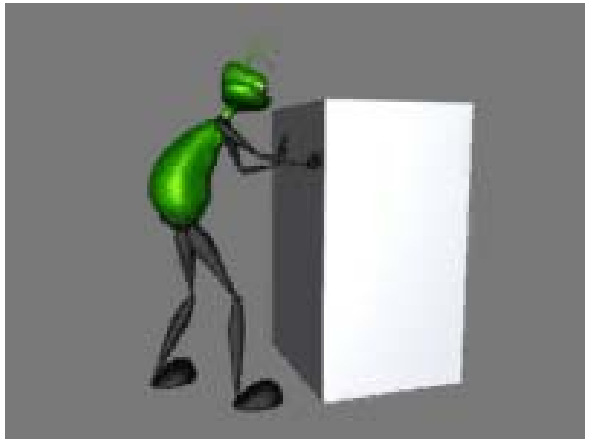


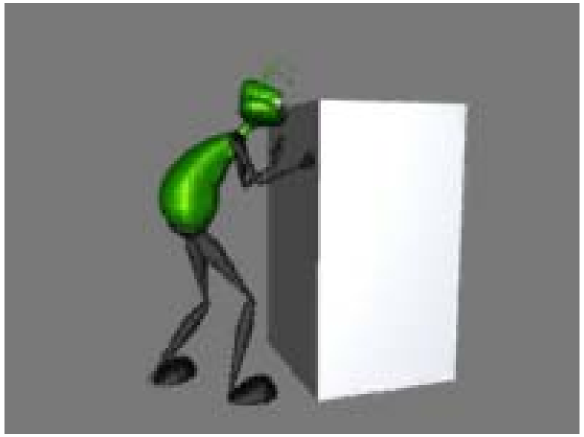
Prepares audience for action so they know what to expect.

Directs audience's attention.

Anticipation (cont'd)

Amount of anticipation (combined with timing) can affect perception of speed or weight.

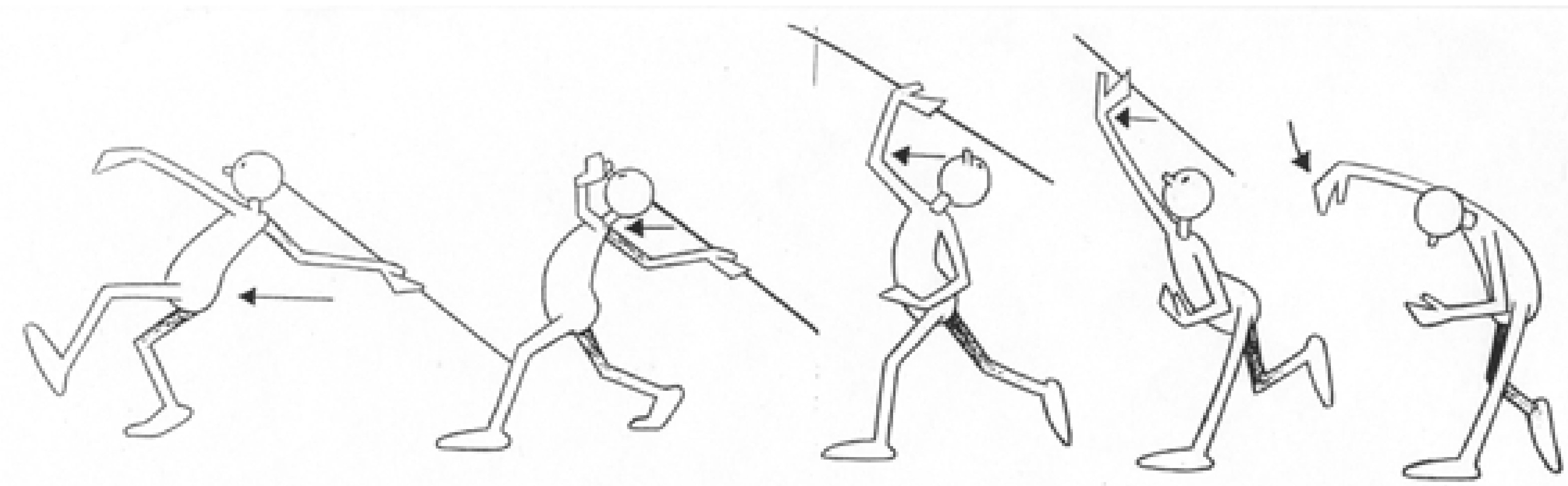




Follow through

Actions seldom come to an abrupt stop.

Physical motivation: inertia



Overlapping action

One part intiates ("leads") the move. Others follow in turn.

Hip leads legs, but eyes often lead the head.

Loose parts move slower and drag behind (sometimes called "secondary motion").

Overlaps can apply to intentions. Example: settling into the house at night.

- Close the door
- Lock the door
- Take off the coat
- etc...

Each action doesn't come to a complete finish before the next starts.

Secondary action

An action that emphasizes the main point but is secondary to it.

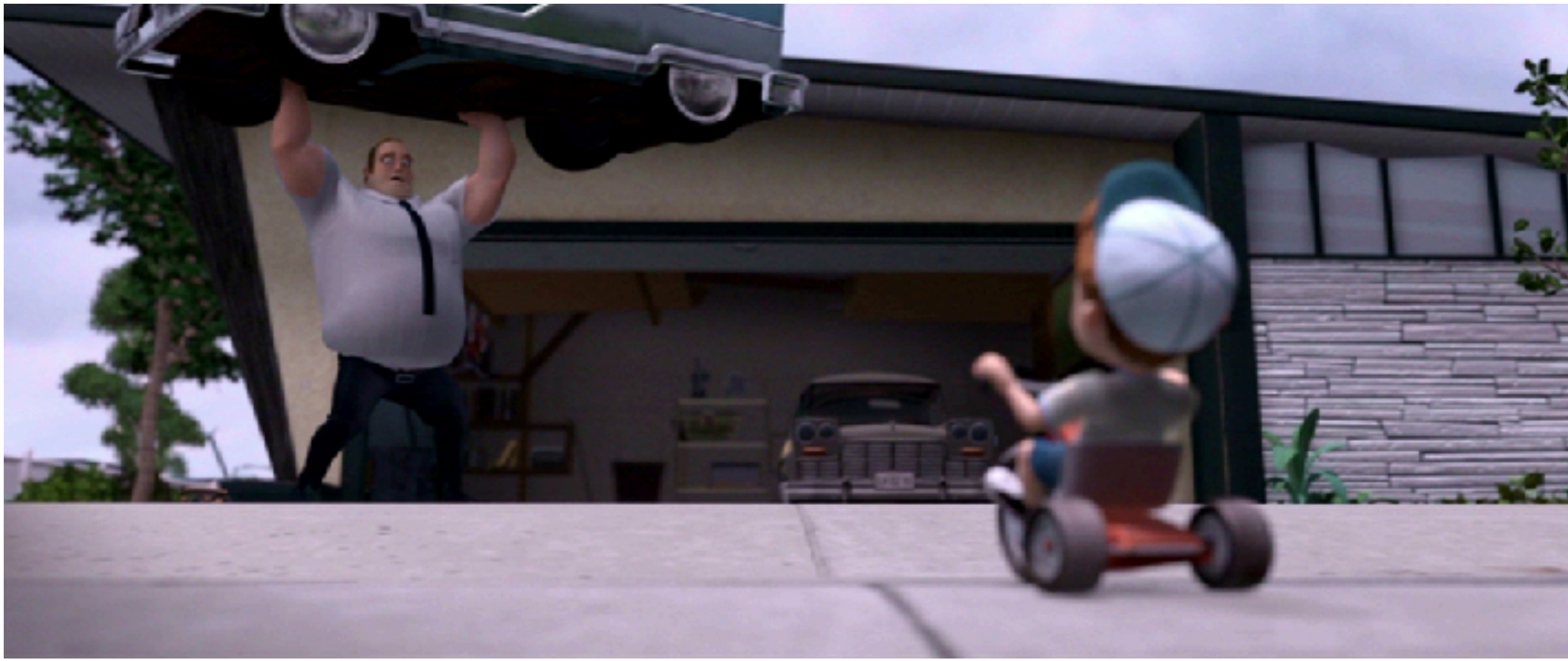
Slow in and slow out

An extreme pose can be emphasized by slowing down as you get to it (and as you leave it).

In practice, many things do not move abruptly but start and stop gradually.

Exaggeration

Get to the heart of the idea and emphasize it so the audience can see it.



THIS IS WHAT'S CALLED A "WOODEN" CHARACTER.

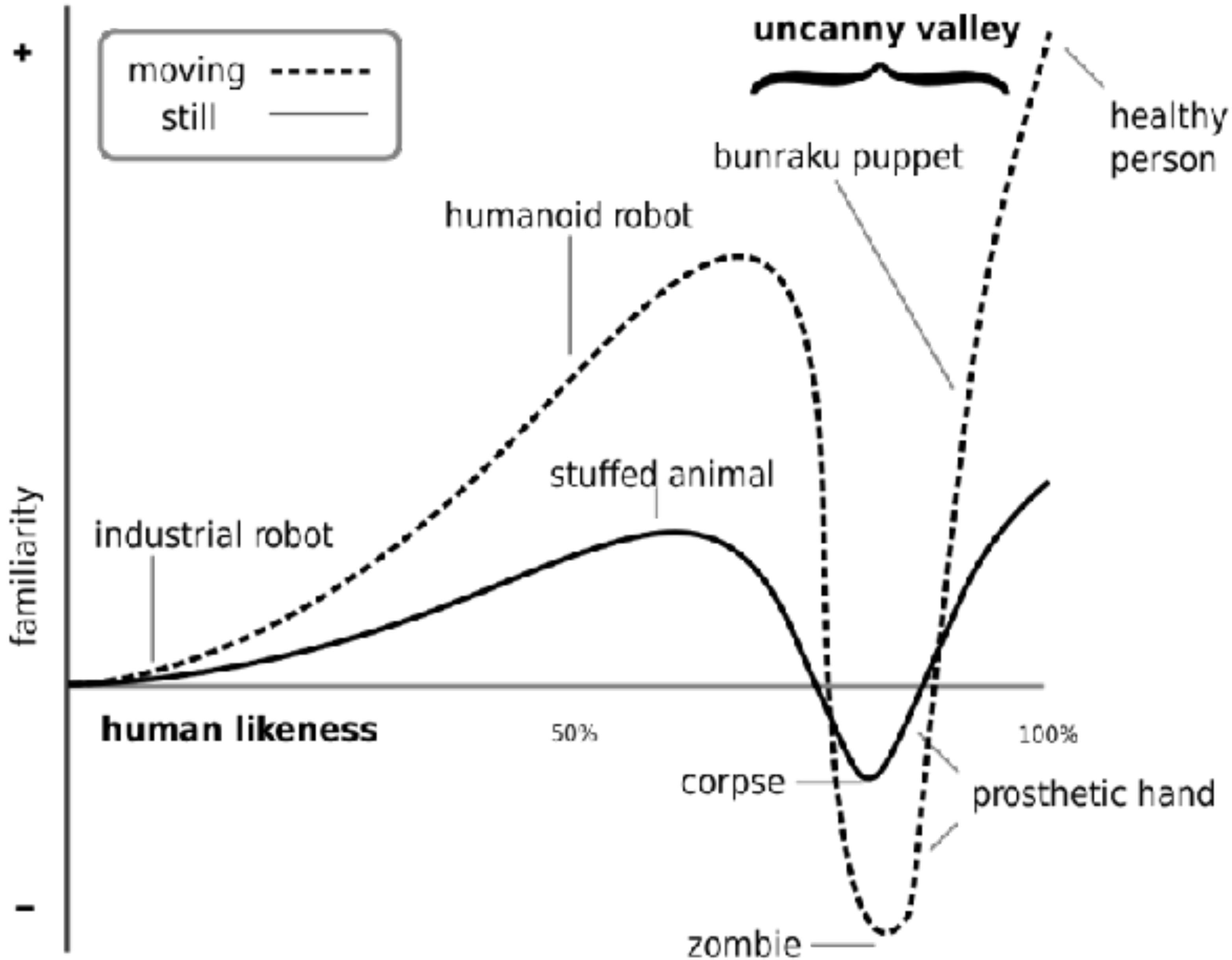
EACH EYE, EAR, ARM, HAND, FINGER, LEG, COLLAR, SHOE, ETC. LOOKS THE SAME AS ITS COUNTERPART. THE RESULT IS A VERY STIFF LOOKING POSE.

...THIS CHARACTER LOOKS MORE NATURAL SIMPLY BECAUSE EACH PART OF THE BODY VARIES IN SOME WAY FROM THE CORRESPONDING OPPOSITE PART.

EYES IN PERSPECTIVE

FINGERS THAT VARY GIVE THE HANDS A MORE DYNAMIC LOOK.

The Uncanny Valley



Three main computer animation techniques

- Keyframing (by hand)
- Physical simulation
- Motion capture

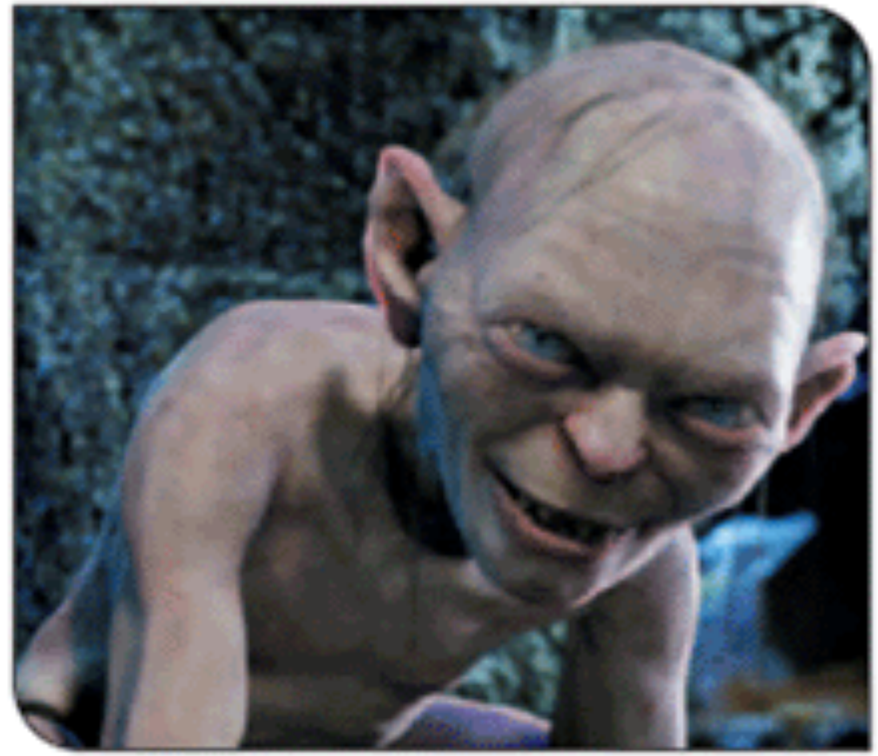
Three main trade-offs

- Physical realism
- Editability
- Directorial control

[redacted]

[redacted]

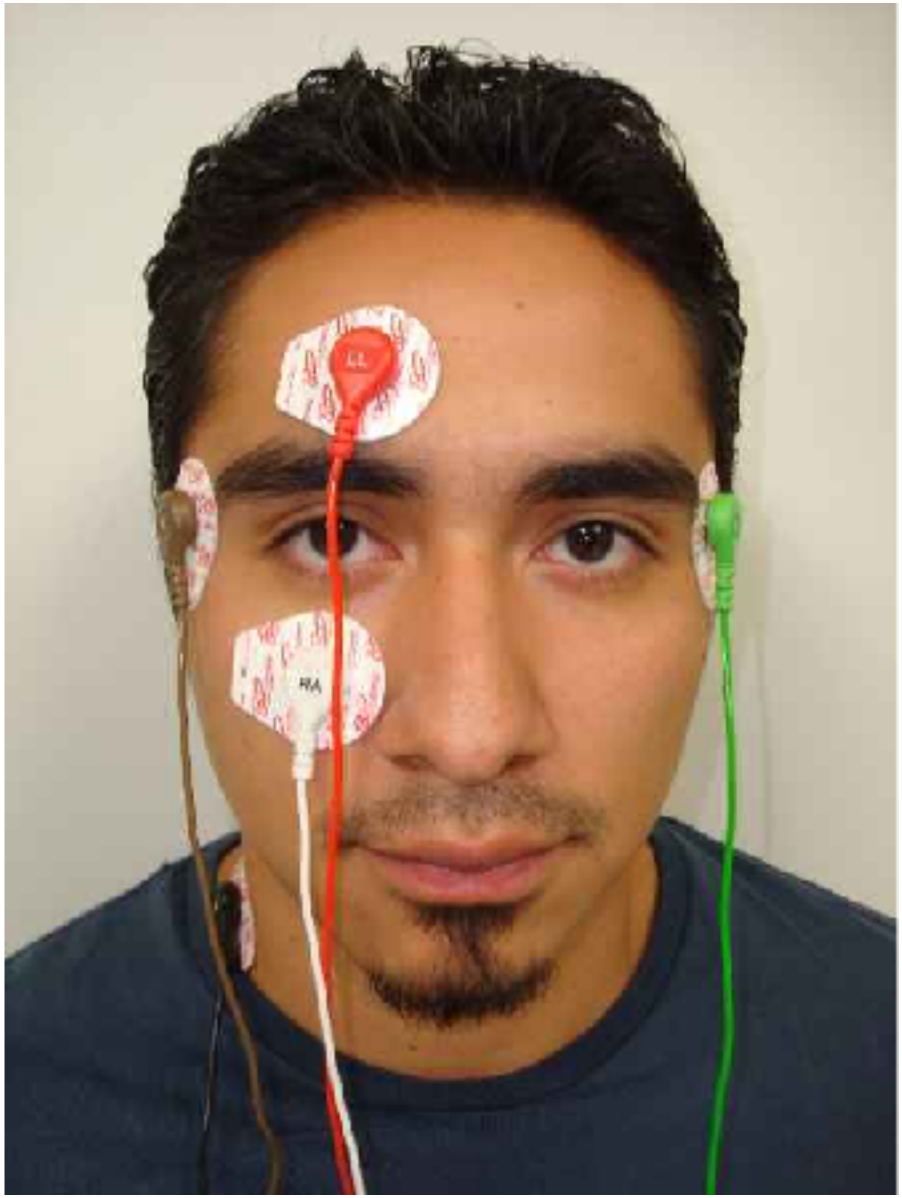
Motion capture



Motion capture of body and face

Eye tracking (with vision-based capture)

Electrooculography



Arc Length Parameterization

1. Given spline path $P(u) = [x(u), y(u), z(u)]$, compute arclength of spline as a function of u: $s = A(u)$.
2. Find the inverse of $A(u)$: $u = A^{-1}(s)$.
3. Substitute $u = A^{-1}(s)$ into $P(u)$ to find motion path parameterized by arclength, s: $P(s) = P(A^{-1}(s))$.

u (and thus s) should be global parameters, extending across all segments of the original spline.

Velocity Control

To control velocity along a spline motion path,

- Let $s = f(t)$ specify distance along the spline as a function of time t.
- The function $f(t)$, being just a scalar value, can be supported with a functional animation technique (i.e., another spline).
- The function $f(t)$ may be specifed as the integral of yet another function, $v(t) = df(t)/dt$, the *velocity.*

  Integral of a spline function can be found analytically, through a computation of the control points.
- The motion path as a function of time t is thus given by $P(A^{-1}(f(t)))$.

Problems with Arc-Length Parameterization

1. The arc-length $s = A(u)$ is given by the integral

$$s = A(u) = \int_0^u \sqrt{\left(\frac{dx(v)}{dv}\right)^2 + \left(\frac{dy(v)}{dv}\right)^2 + \left(\frac{dz(v)}{dv}\right)^2}\, dv$$

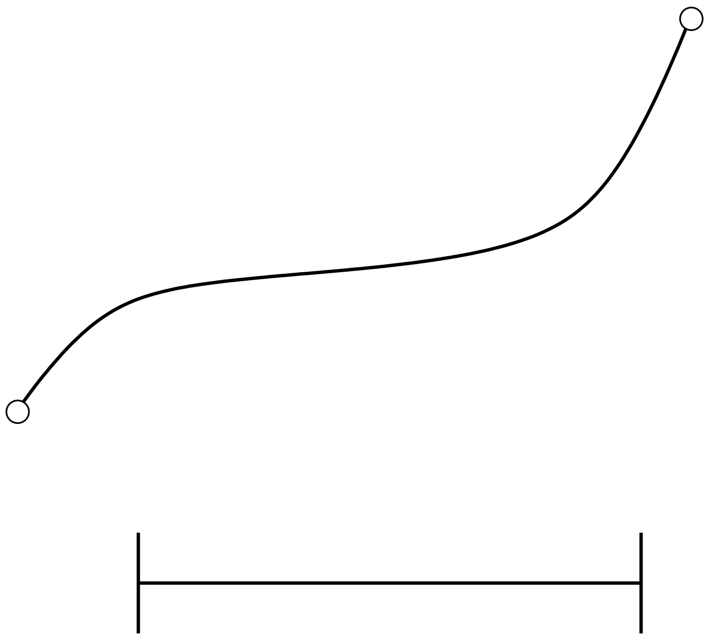
   No analytic solution if the motion path is a cubic spline.

2. Since $A(u)$ has no analytic form, $A^{-1}(s)$ has no analytic form.

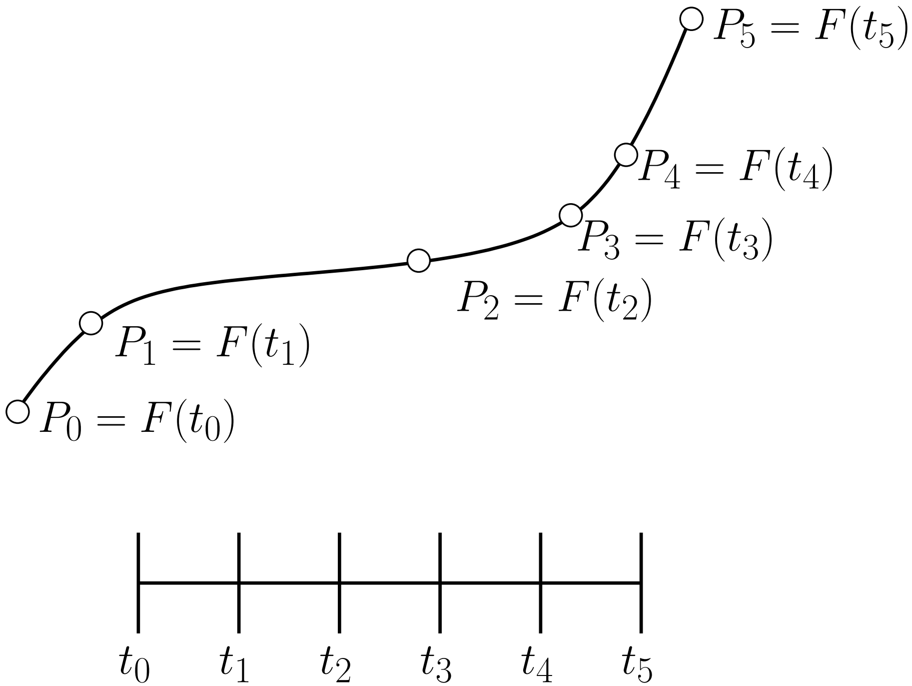
Issues in Real-Time Animation

- Exact arc-length parameterization not feasible.
- Alternative: compute points on the spline at equally-spaced parametric values,
  use linear interpolation along these chords.
- The linear interpolation should consider the distance between samples to maintain constant velocity.

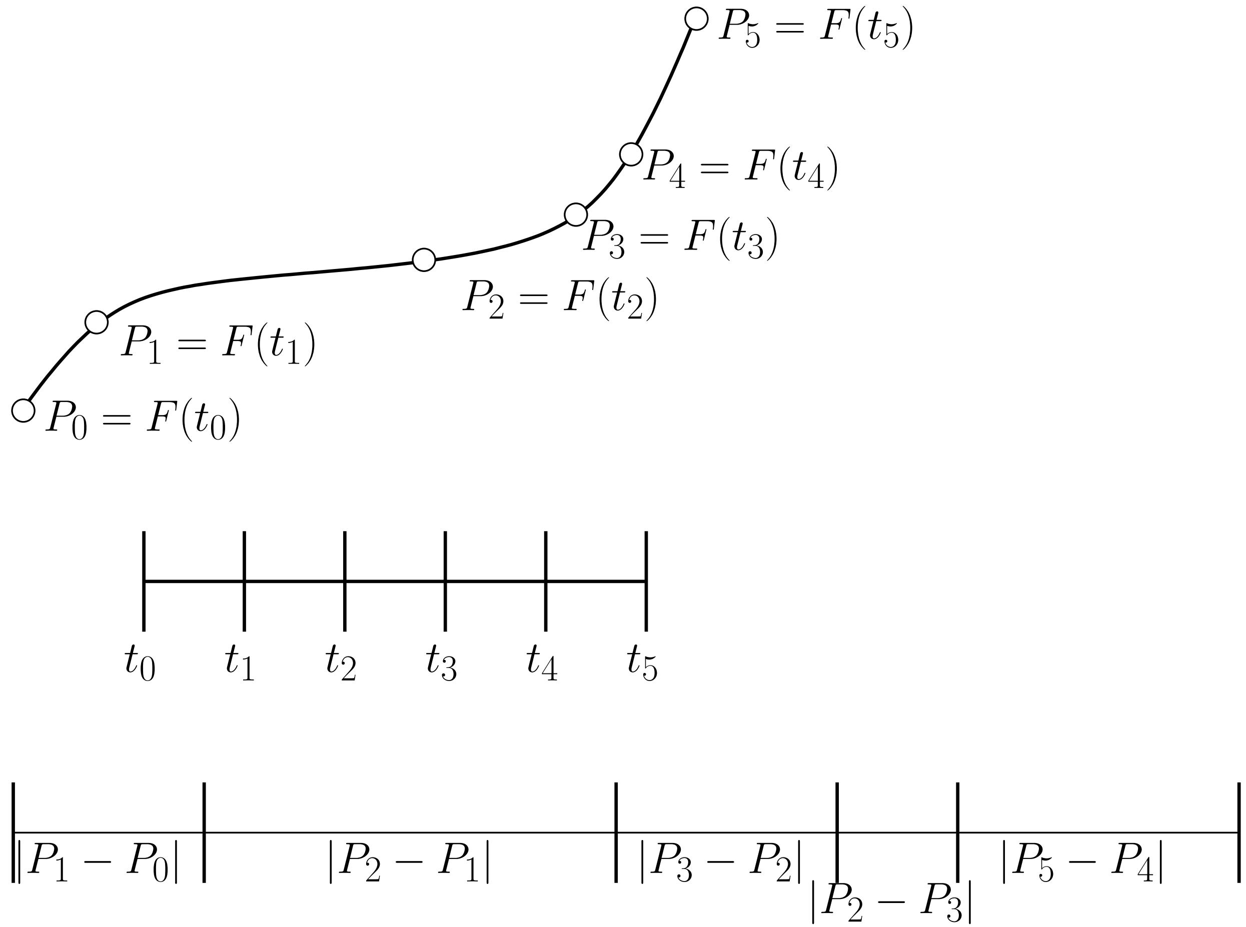
Approximate Arc Length Parameterization



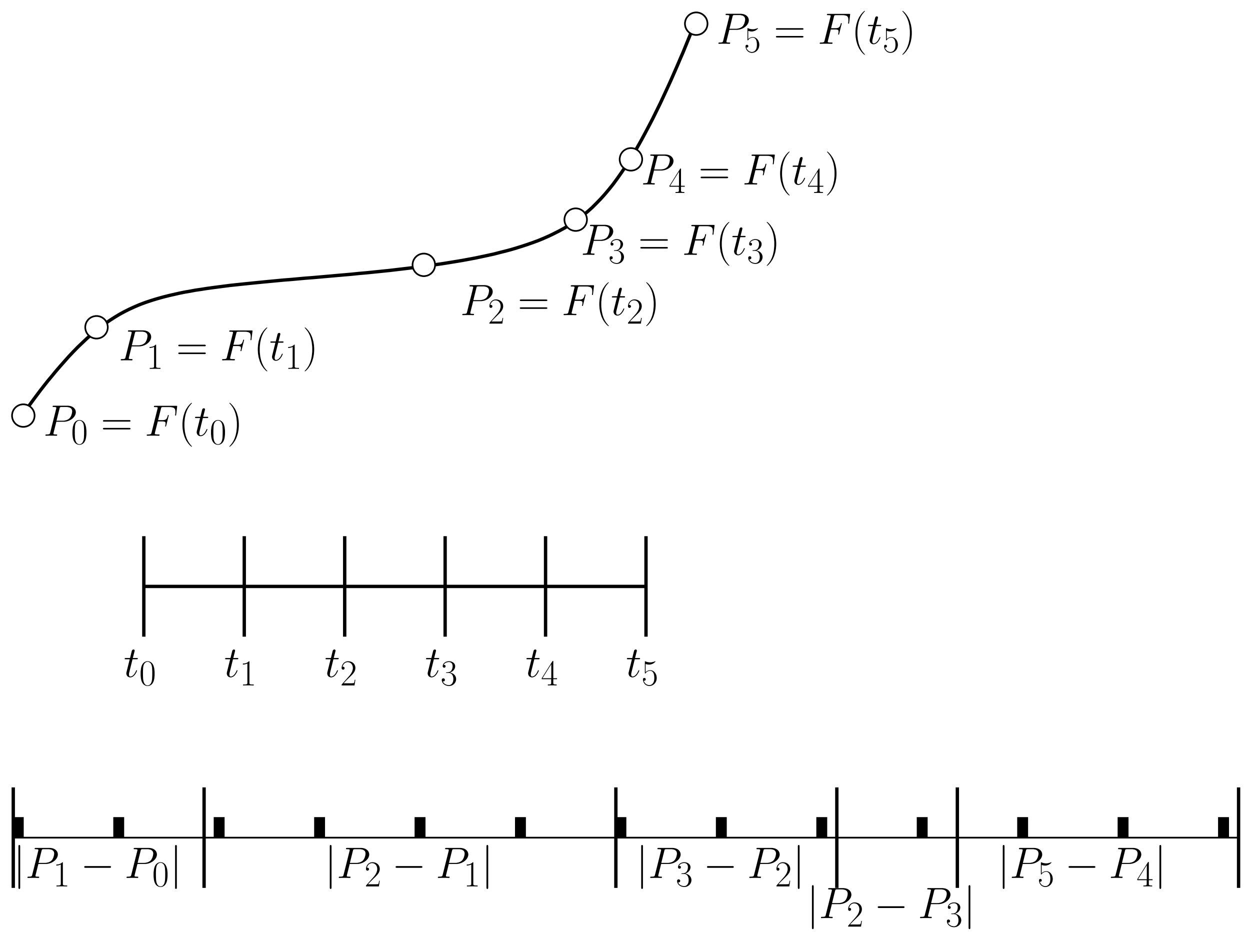
Approximate Arc Length Parameterization



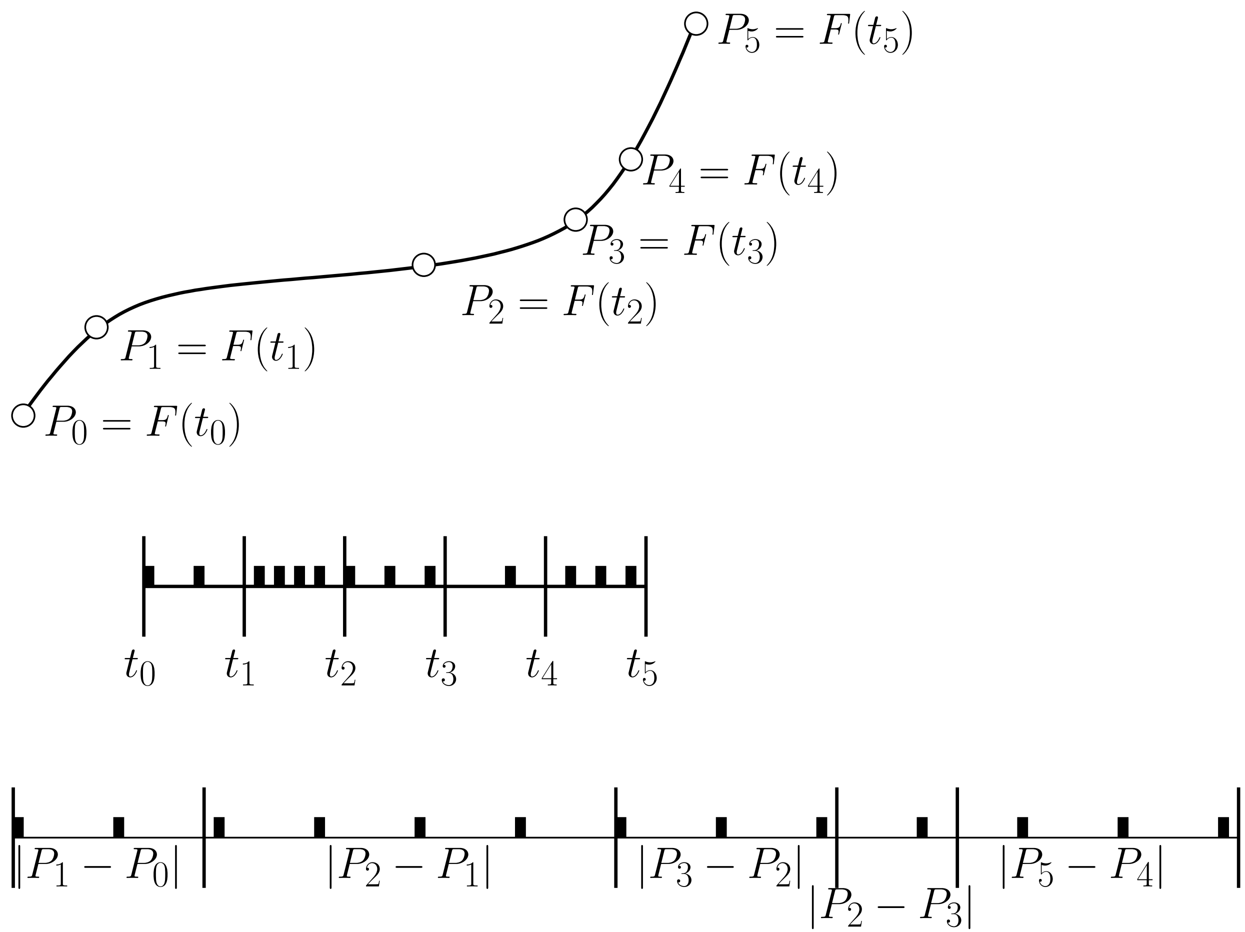
Approximate Arc Length Parameterization



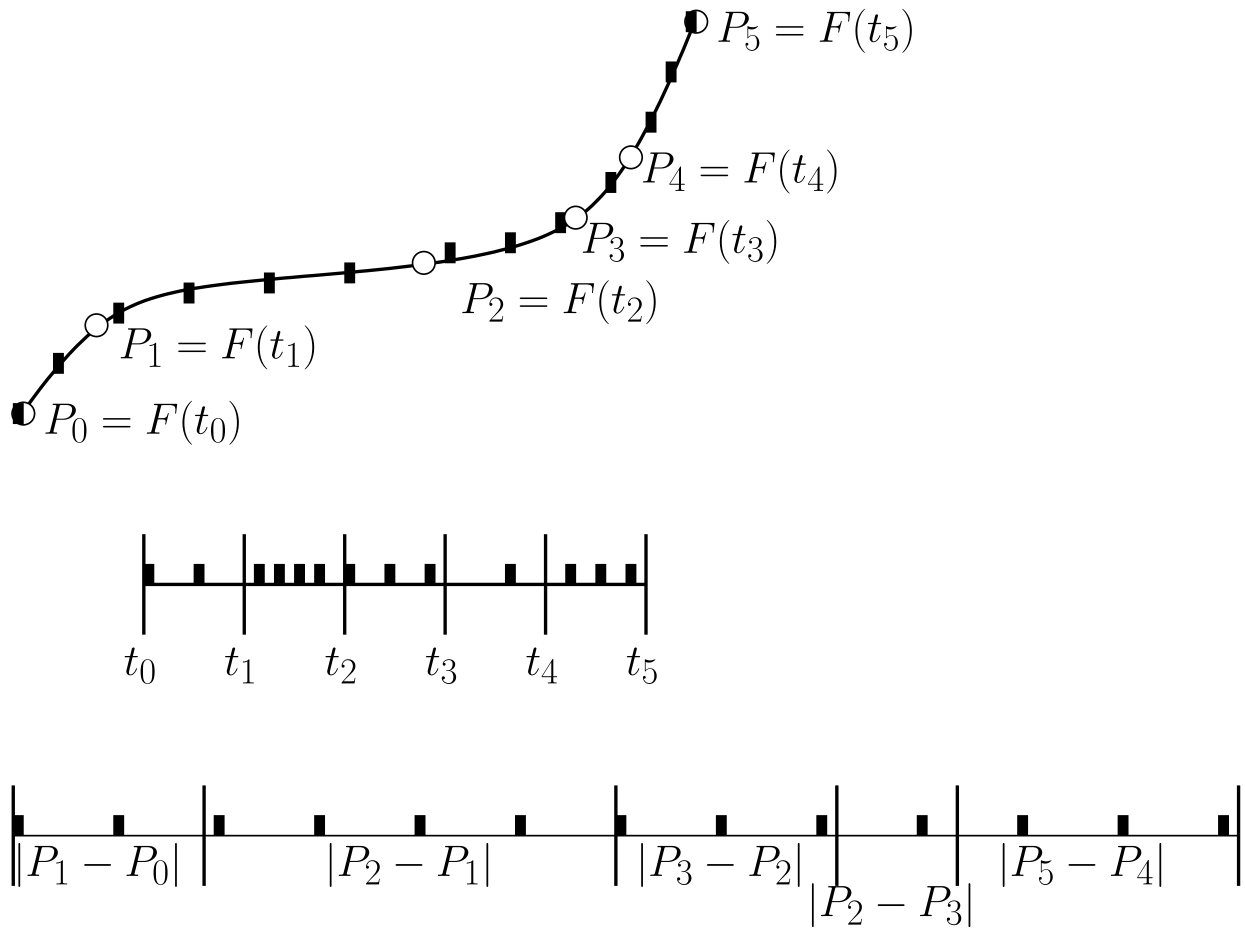
Approximate Arc Length Parameterization



Approximate Arc Length Parameterization



Approximate Arc Length Parameterization



Orientation Representation and Interpolation

Interpolate angles rather than transformation matrices.

- In two dimensions, only one angle is involved and animation is straightforward.
- In three dimensions, orientation requires three degrees of freedom.

  Interpolation is much nastier and harder to visualize.

Approaches

Two standard approaches to the orientation interpolation. Both related to specifying orientation in the first place:

Euler angles. Three angles: x-roll followed by y-roll followed by z-roll.

– Has defects: parameterization singularities, anisotropy, "gimbal lock", unpredictable interpolation.

– Hard to solve inverse problem: given orientation, what are the angles?

+ Widely used in practice.

+ Easy to implement.

+ Inexpensive computationally.

Quaternions. Four-dimensional analogs of complex numbers.

+ Isotropic: avoid the problems of Euler angles.
+ Inverse problem easy to solve.
+ The user interface tends to be simpler.
– More involved mathmatically.
– Interpolation can be expensive in practice.

Euler Angles

- With $c_a = \cos(\theta_a)$ and $s_a = \sin(\theta_a)$,

$$\begin{aligned} R(\theta_x, \theta_y, \theta_z) &= \begin{bmatrix} c_y c_z & c_y s_z & -s_y & 0 \\ s_x s_y c_z - c_x s_z & s_x s_y s_z + c_x c_z & s_x c_y & 0 \\ c_x s_y c_z + s_x s_z & c_x s_y s_z - s_x c_z & c_x c_y & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \\ &= R_x(\theta_x) R_y(\theta_y) R_z(\theta_z), \end{aligned}$$

  where $R_x(\theta_x)$, $R_y(\theta_y)$ and $R_z(\theta_z)$ are the standard rotation matrices.

- Given a point P represented as a homogeneous row vector, the rotation of P is given by $P' = PR(\theta_x, \theta_y, \theta_z)$.
- Animation between two rotations involves simply interpolating independently the three angles θ_x, θ_y, and θ_z.

Problems with the Euler angle approach include:

Parametric Singularity: A degree of freedom can suddenly vanish.

Anisotropy: The order of the axes is important.

Nonobviousness: The parameters lack useful geometric significance.

Inversion: Finding the Euler angles for a given orientation is difficult (not unique).

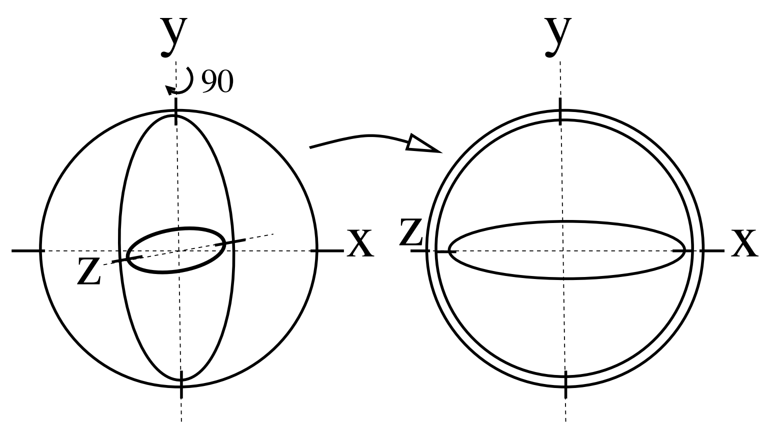
Coordinate system dependence: The orientation of the coordinate axes is important.

Gimbal Lock: An Example

- *Gimbal lock* is an example of a **parametric singularity**.
- Gimbal lock is a mechanical problem that arises in gyroscopes as well as Euler angles.
- Set $\theta_y = \pi/2 = 90^\circ$, and set θ_x and θ_z arbitrarily. Then $c_y = 0$, $s_y = 1$ and the matrix $R(\theta_x, \pi/2, \theta_z)$ can be reduced to

$$
\begin{aligned}
R(\theta_x, \theta_y, \theta_z) &= \begin{bmatrix} 0 & 0 & -1 & 0 \\ s_x c_z - c_x s_z & s_x s_z + c_x c_z & 0 & 0 \\ c_x c_z + s_x s_z & c_x s_z - s_x c_z & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \\
&= \begin{bmatrix} 0 & 0 & -1 & 0 \\ \sin(\theta_x - \theta_z) & \cos(\theta_x - \theta_z) & 0 & 0 \\ \cos(\theta_x - \theta_z) & \sin(\theta_x - \theta_z) & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}.
\end{aligned}
$$

- A y-roll by $\pi/2$ rotates the x-axis onto the negative z axis, and so a x-roll by θ has the same effect as a z-roll by $-\theta$.



- Gimbal lock can be very frustrating in practice:
  - During interactive manipulation the object will seem to “stick”;
  - Certain orientations can be hard to obtain if approached from the wrong direction;
  - Interpolation through these parametric singularities will behave strangely.

In 2D, orientation can be represented by a single number: an angle.

In 2D, orientation can be represented by a single number: an angle.

…well, not quite—a real number doesn't capture the observation that the space of angles is closed and bounded.

Represent the orientation θ via the complex number

$$e^{i\theta} = \cos\theta + i\sin\theta$$

In general, every unit complex number can be interpreted as an orientation in this way!

Nice properties:

- Multiplying complex numbers composes orientations
- Multiply any complex number by an orientation to rotate it

William Rowan Hamilton (1805–1865)

Quaternions

$$\mathbb{H} = \{a + bi + cj + dk | a, b, c, d \in \mathbb{R}\}$$

Quaternions

$$\mathbb{H} = \{a + bi + cj + dk | a, b, c, d \in \mathbb{R}\}$$

$$i^2 = j^2 = k^2 = ijk = -1$$


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

Norm of a quaternion

$$||q|| = \sqrt{a^2 + b^2 + c^2 + d^2}$$

q is a unit quaternion if $||q|| = 1$.

Unit complex numbers are isomorphic to the circle S^1.

Unit complex numbers are isomorphic to the circle S^1.

Unit quaternions are isomorphic to the unit 4-dimensional sphere S^3!

There is a natural mapping from S^3 to the space $\mathrm{SO}(3)$ of all 3D orientations.

There is a natural mapping from S^3 to the space $\mathrm{SO}(3)$ of all 3D orientations.

$$K : \mathbf{S^3} \to \mathbf{SO}(3) : \begin{bmatrix} a \\ b \\ c \\ d \end{bmatrix} \mapsto \begin{bmatrix} a^2 + b^2 - c^2 - d^2 & 2(bc - ad) & 2(ac + bd) \\ 2(ad + bc) & a^2 - b^2 + c^2 - d^2 & 2(cd - ab) \\ 2(bd - ac) & 2(ab + cd) & a^2 - b^2 - c^2 + d^2 \end{bmatrix}$$

There is a natural mapping from S^3 to the space $\mathrm{SO}(3)$ of all 3D orientations.

$$K : \mathbf{S^3} \rightarrow \mathbf{SO}(3) : \begin{bmatrix} a \\ b \\ c \\ d \end{bmatrix} \mapsto \begin{bmatrix} a^2+b^2-c^2-d^2 & 2(bc-ad) & 2(ac+bd) \\ 2(ad+bc) & a^2-b^2+c^2-d^2 & 2(cd-ab) \\ 2(bd-ac) & 2(ab+cd) & a^2-b^2-c^2+d^2 \end{bmatrix}$$

…so quaternions "are" orientations.

This mapping plays nicely with the algebra of quaternions:

This mapping plays nicely with the algebra of quaternions:

$$K(q_1 q_2) = K(q_1)K(q_2)$$

Quaternions are naturally connected to “axis-angle” rotation.

Geodesic paths in S^3 yield optimal orientation animations!

Given quaternions q_1 and q_2 :

Given quaternions q_1 and q_2 :

Define $\omega = \cos^{-1}(q_1 \cdot q_2)$, the angle between the quaternions in 4D space.

Given quaternions q_1 and q_2 :

Define $\omega = \cos^{-1}(q_1 \cdot q_2)$, the angle between the quaternions in 4D space.

$$Q(t) = \frac{\sin(1-t)\omega}{\sin\omega} q_1 + \frac{\sin t\omega}{\sin\omega} q_2$$

Given quaternions q_1 and q_2 :

Define $\omega = \cos^{-1}(q_1 \cdot q_2)$, the angle between the quaternions in 4D space.

$$Q(t) = \frac{\sin(1-t)\omega}{\sin\omega} q_1 + \frac{\sin t\omega}{\sin\omega} q_2$$

SLERP: spherical linear interpolation.