CS230 : Computer Graphics

Animation

Tamar Shinar
Computer Science & Engineering
UC Riverside

Types of animation

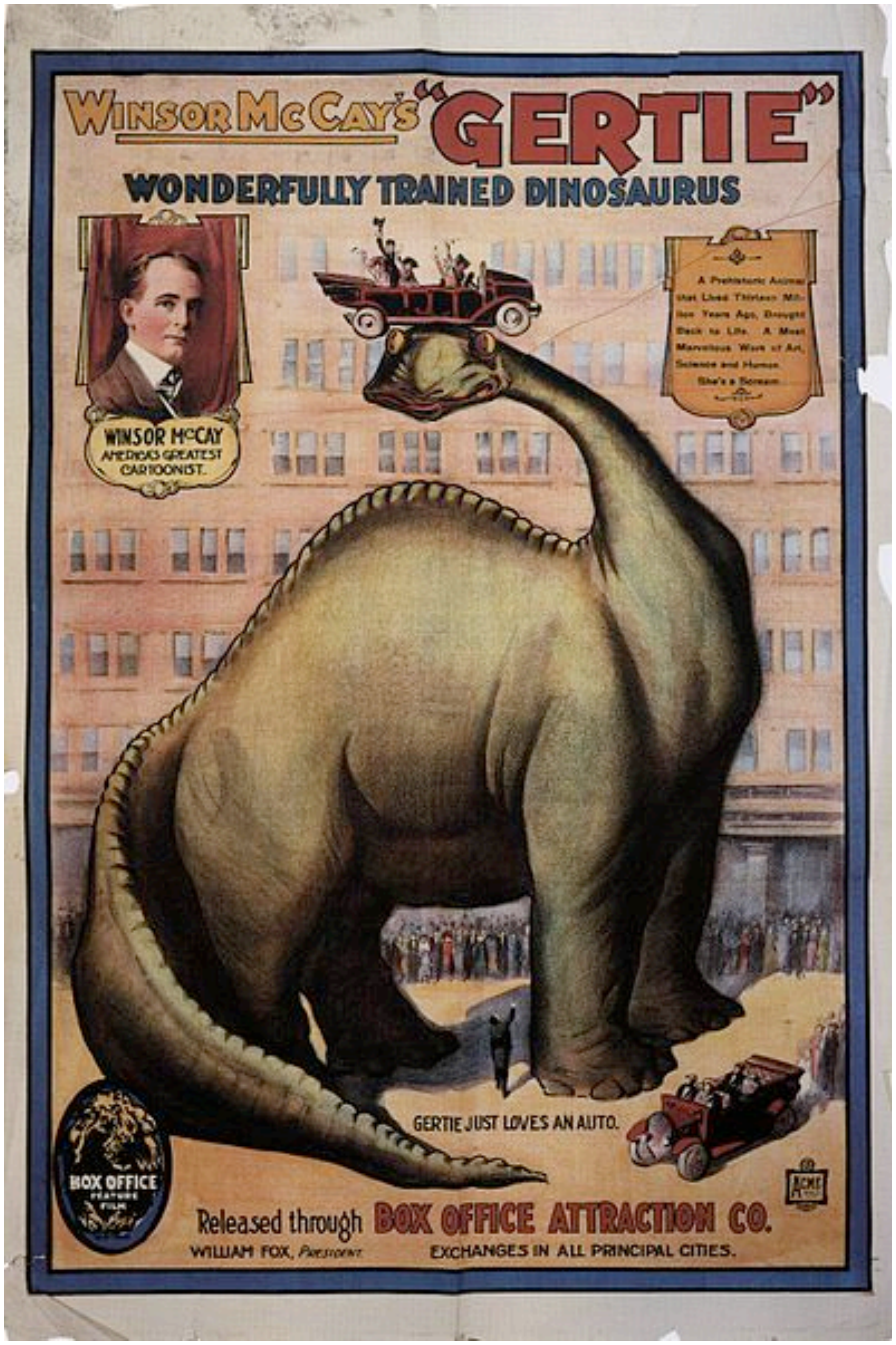
- keyframing
- procedural
- physics-based
- motion capture
- stop motion
- rotoscoping

history

Gertie the Dinosaur

1914
12 minutes
hand drawn
keyframe animation
registration
cycling

link

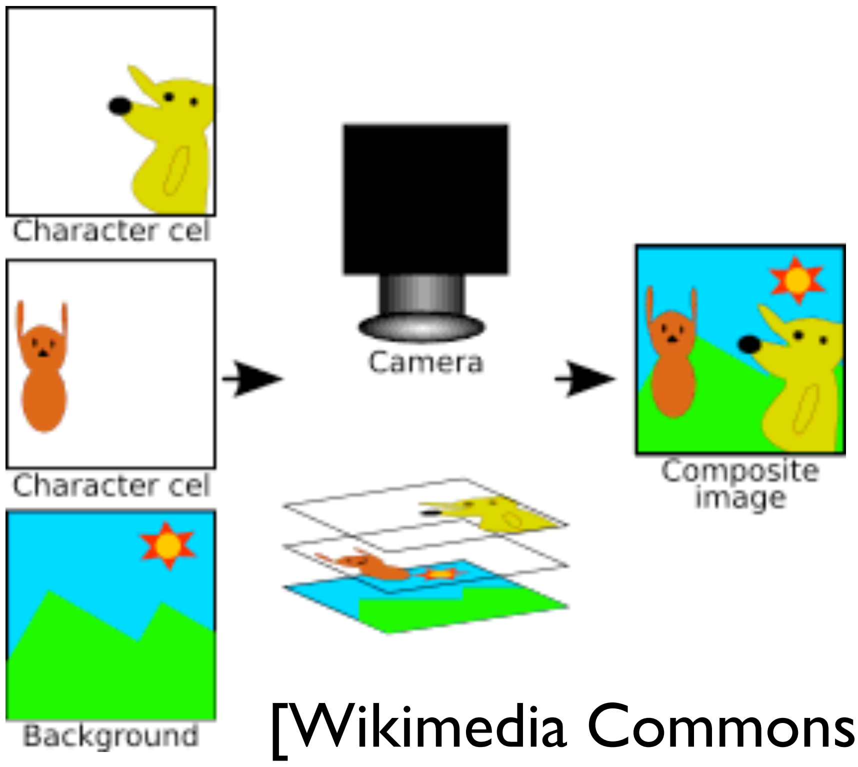

WINSOR McCAY'S "GERTIE"
WONDERFULLY TRAINED DINOSAURUS
WINSOR McCAY
AMERICA'S GREATEST CARTOONIST.
A Prehistoric Animal that Lived Thirteen Million Years Ago, Brought Back to Life. A Most Marvelous Work of Art, Science and Humor.
She's a Scream.
GERTIE JUST LOVES AN AUTO.
BOX OFFICE FEATURE FILM
ACME
Released through BOX OFFICE ATTRACTION CO.
WILLIAM FOX, PRESIDENT
EXCHANGES IN ALL PRINCIPAL CITIES.

Traditional animation

Cels

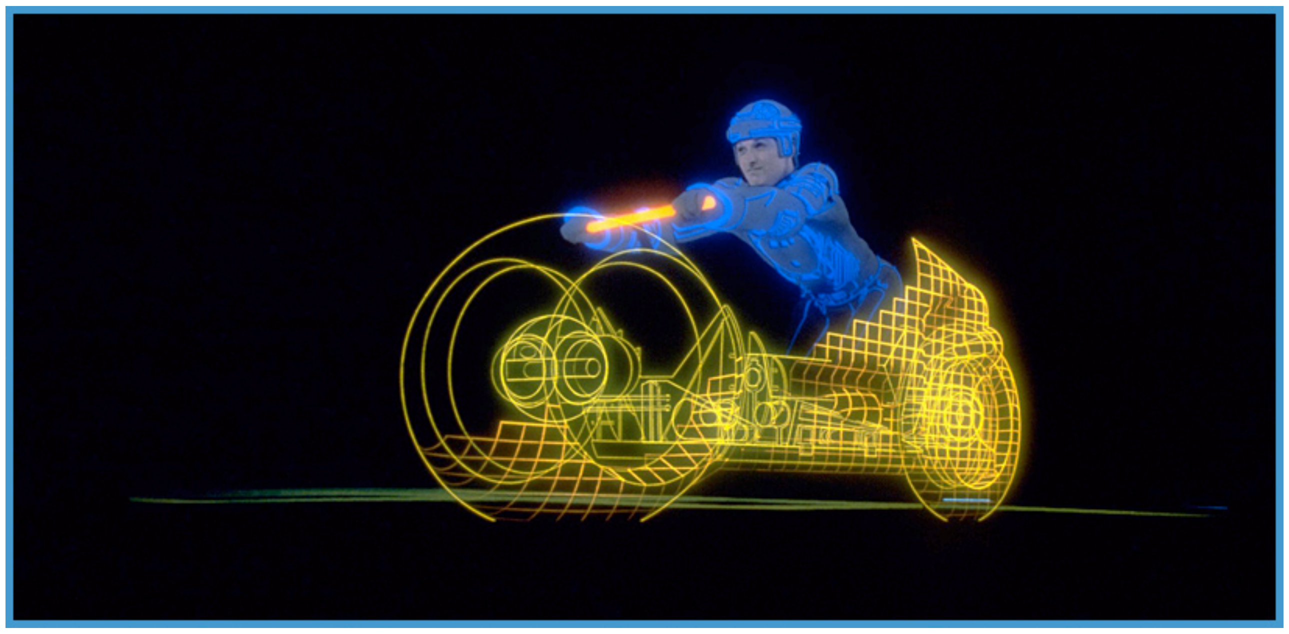
Multiplane camera

Sleeping Beauty, Disney, 1959

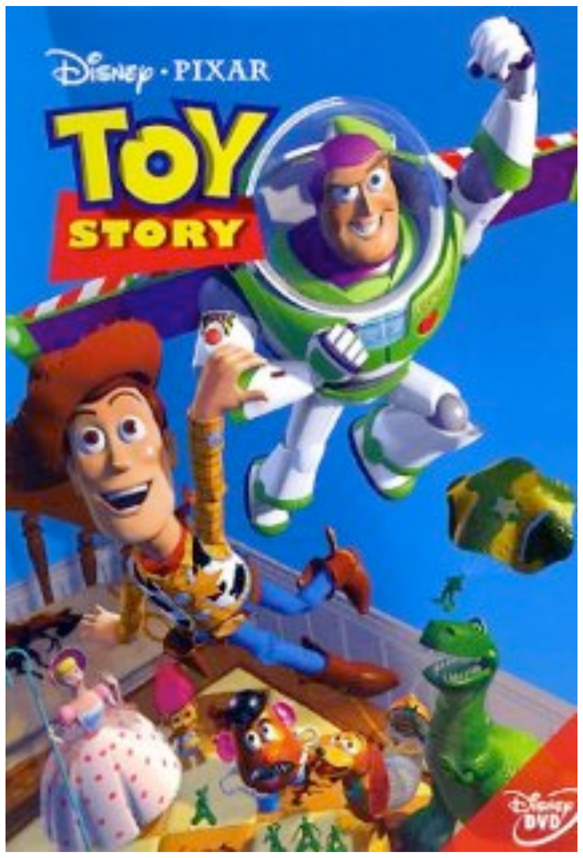


[Wikimedia Commons]

Realistic 3D animation



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



Performance capture

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,

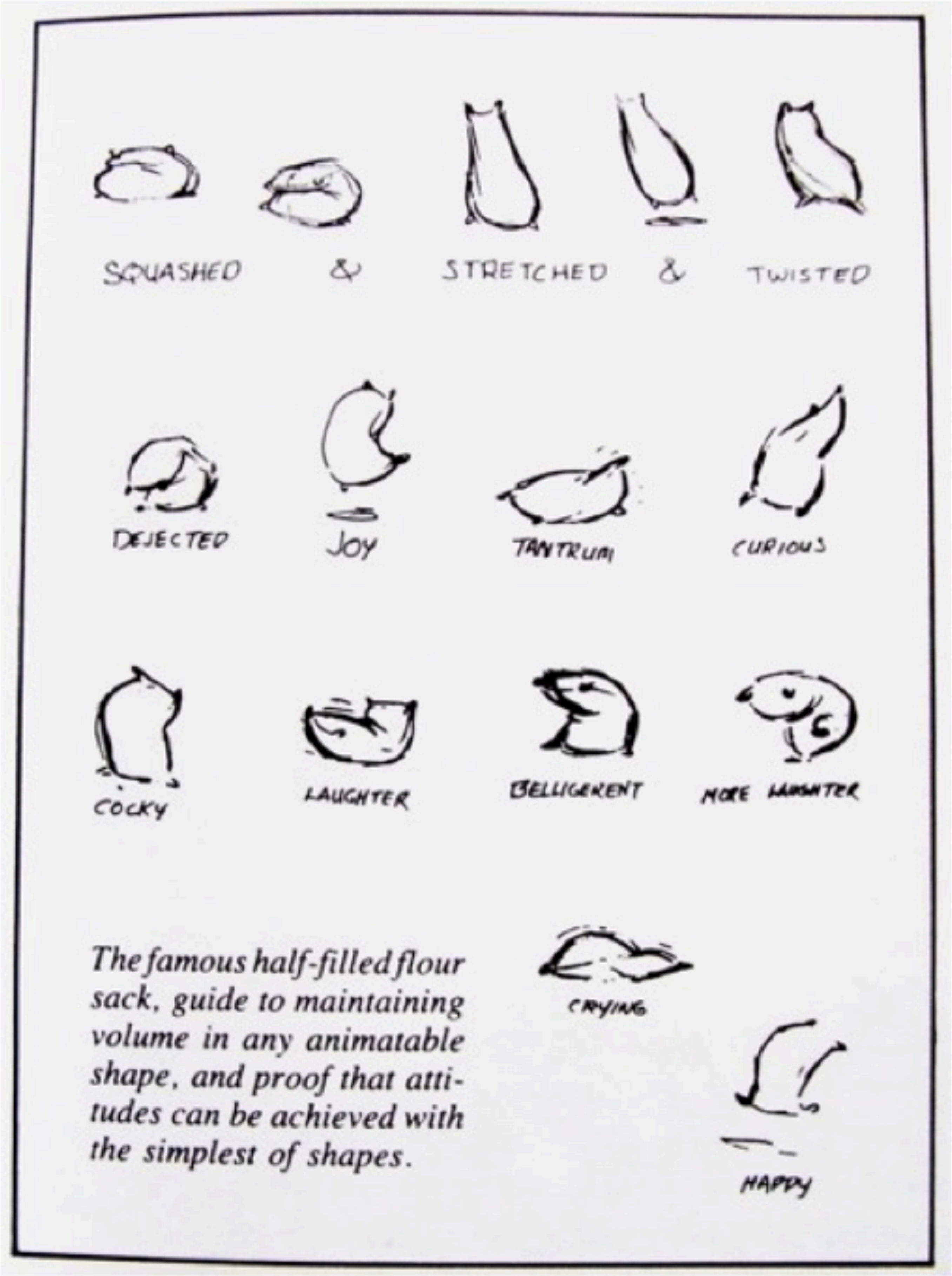
Disney's Paperman

Paperman and the Future of 2D Animation

trailer: http://www.youtube.com/watch?v=mM6cLnscmO8

making of: http://www.youtube.com/watch?v=TZJLtujW6FY&feature=youtube_gdata_player

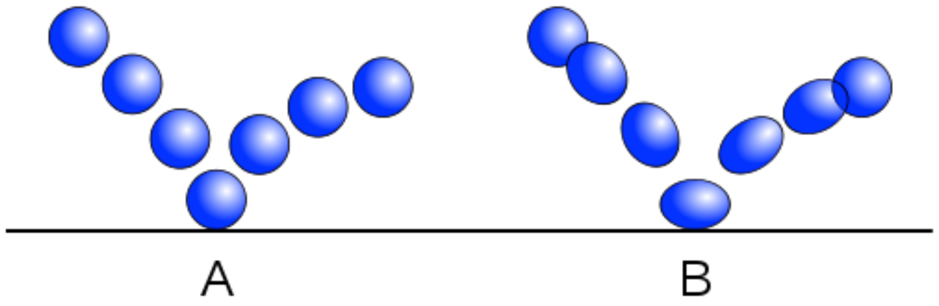
animation principles



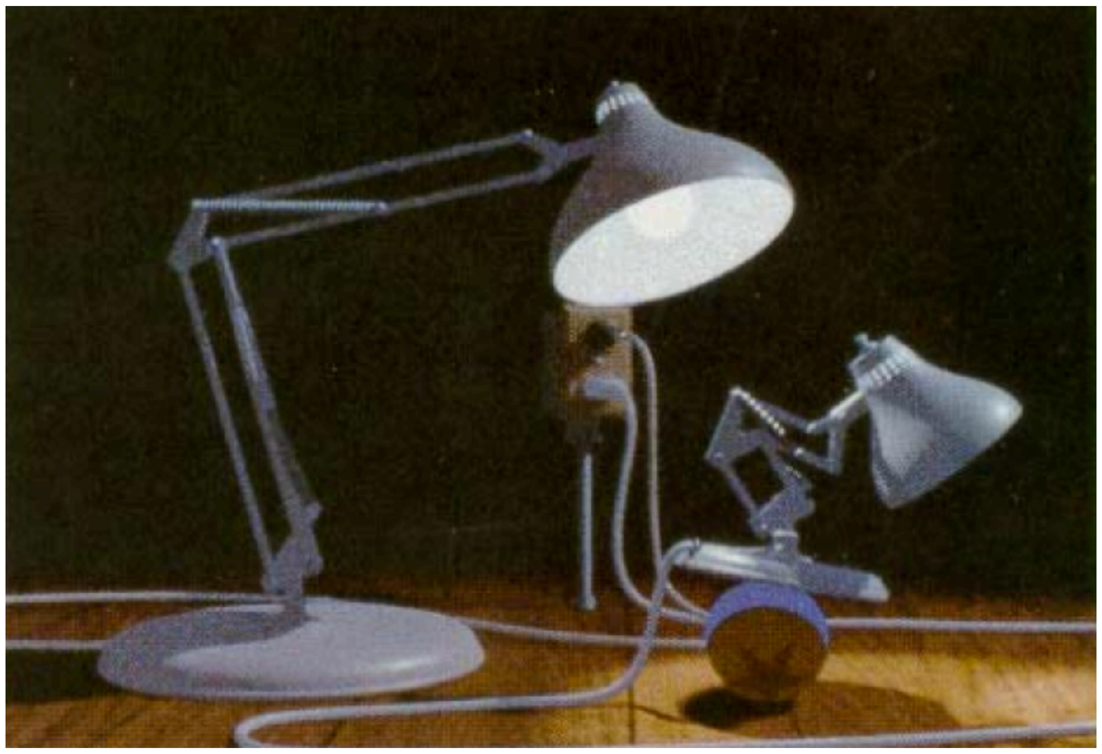
The famous half-filled flour sack, guide to maintaining volume in any animatable shape, and proof that attitudes can be achieved with the simplest of shapes.

- 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



[Lasseter 87]

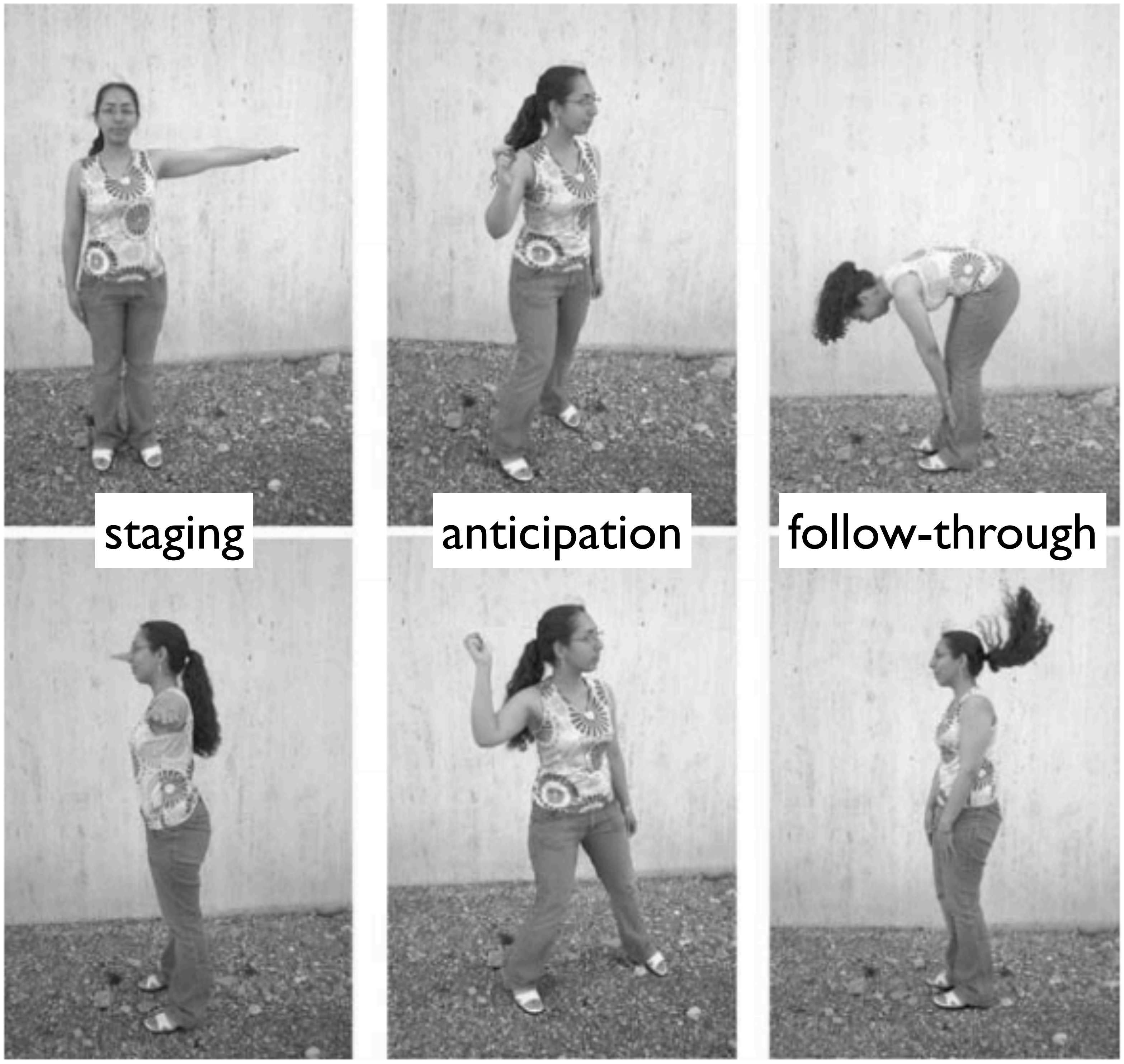
principles are related to the underlying physics of motion
timing: important information. ease in/ease out

Timing

- convey physical information - e.g., weight
- convey meaning of action
  - e.g., turning head to side
    - scanning vs. impact
- may be determined by pre-recorded audio

Staging, anticipation and follow-through

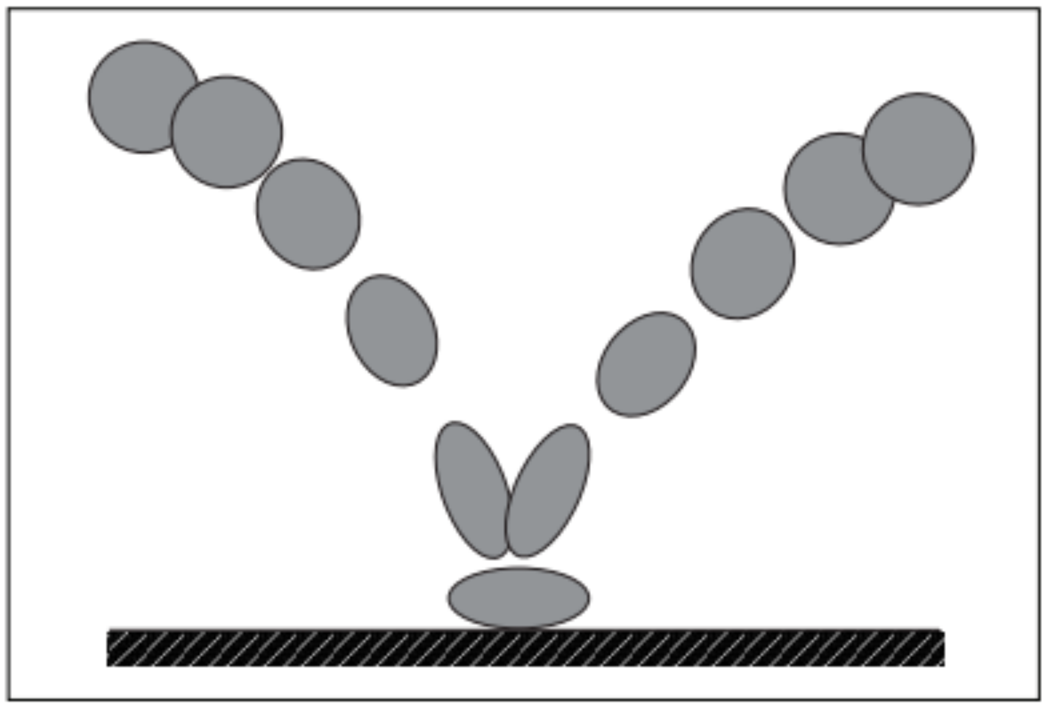
•**Staging**: deliberately highlight or obscure key features
•**Anticipation**: preceding a key action
•**Follow-through**: adds richness and realism



- make key actions more noticable

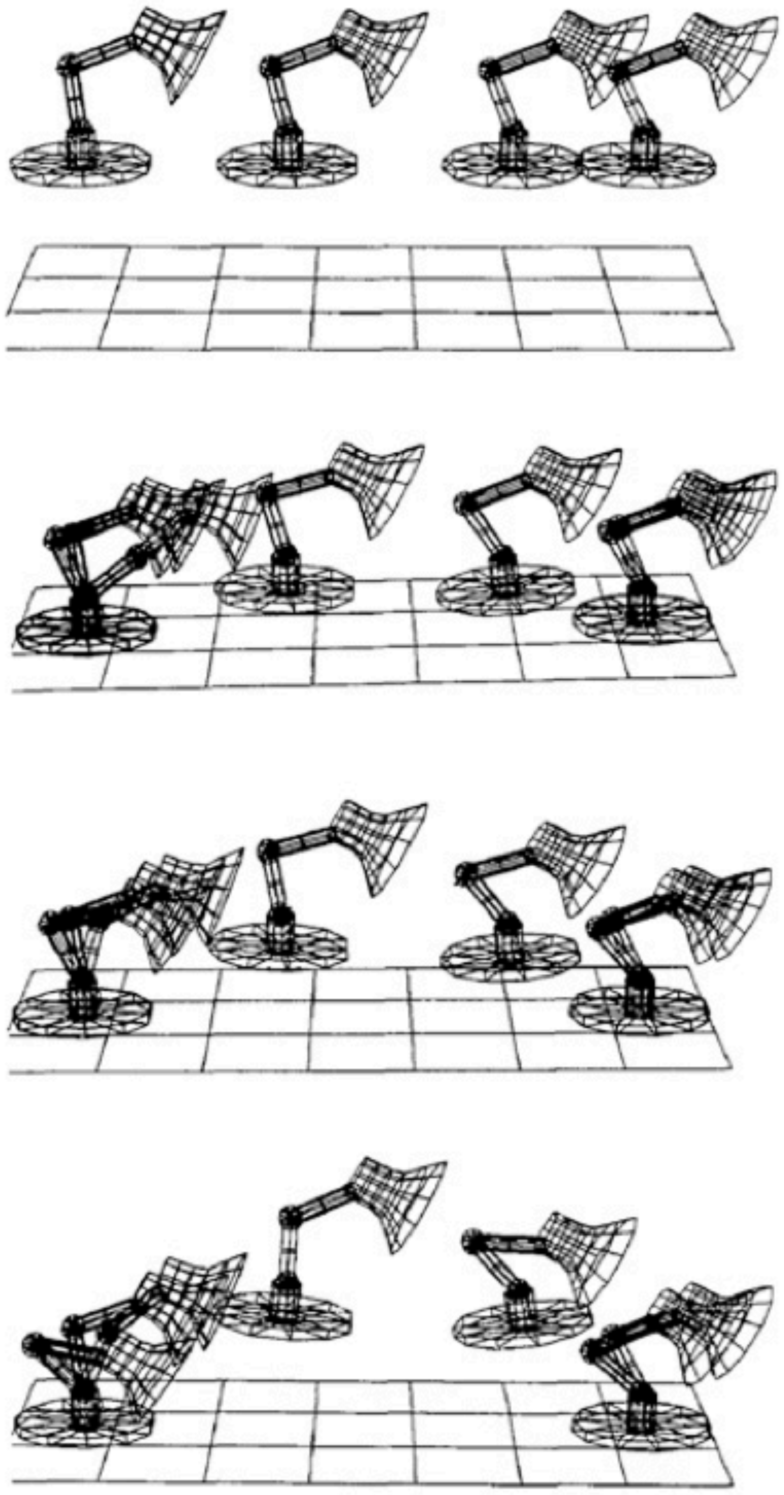
Squash and Strech

- natural motion
  - stretch in direction of motion0
  - squash when force applied
- preserve total volume



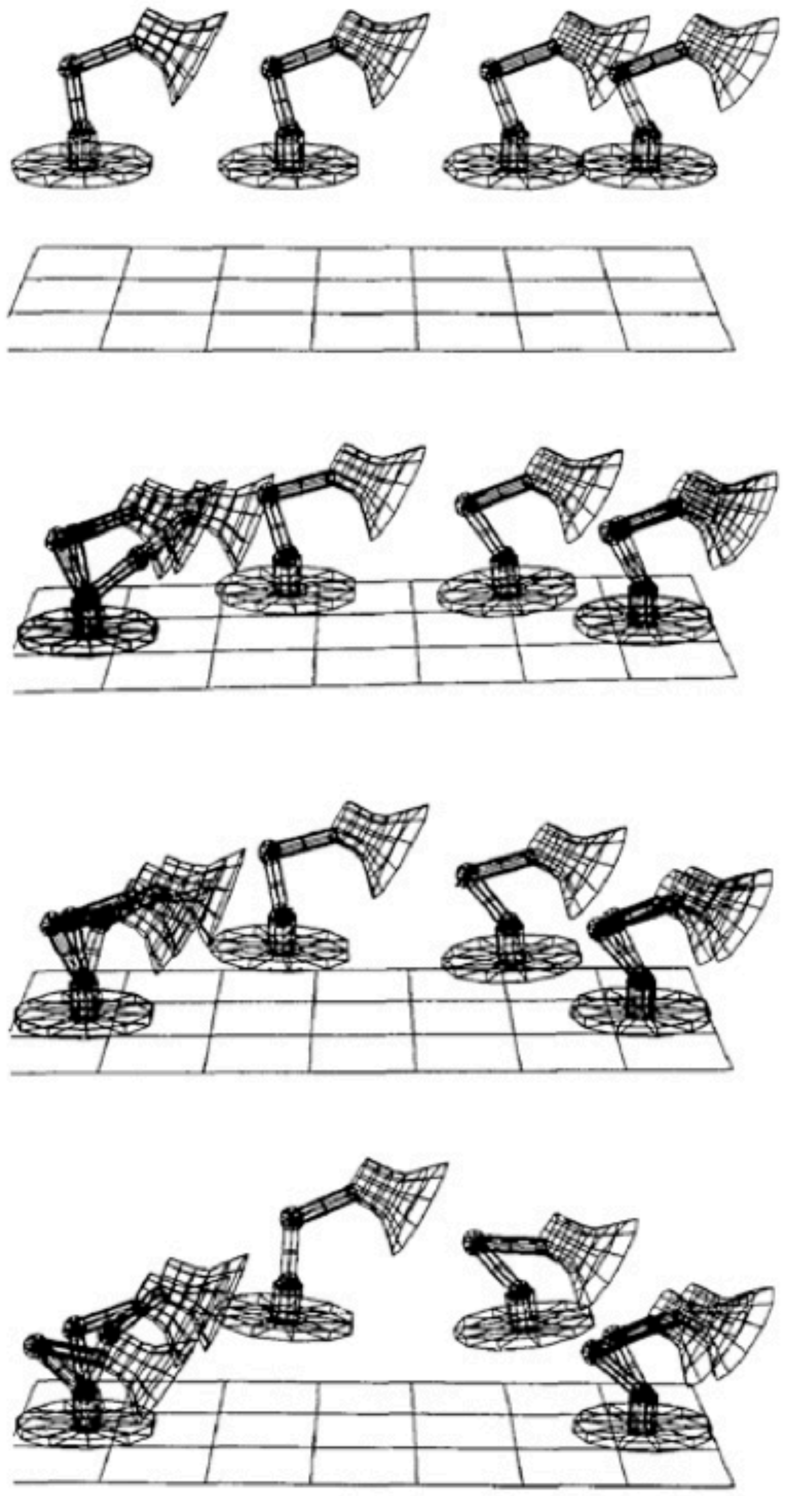
Physics-based animation

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



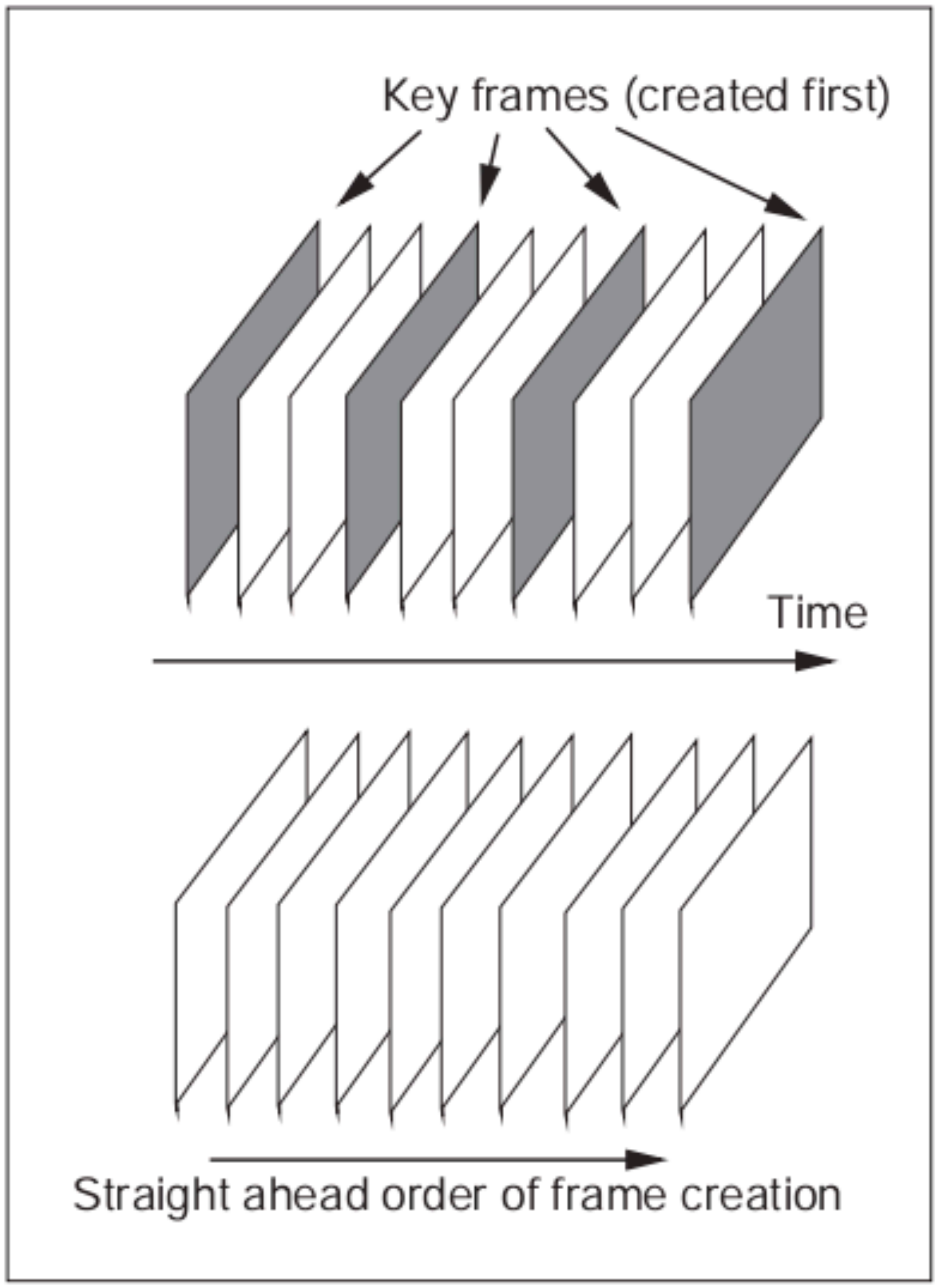
Physics-based animation

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



keyframe animation

Keyframe vs. straight-ahead

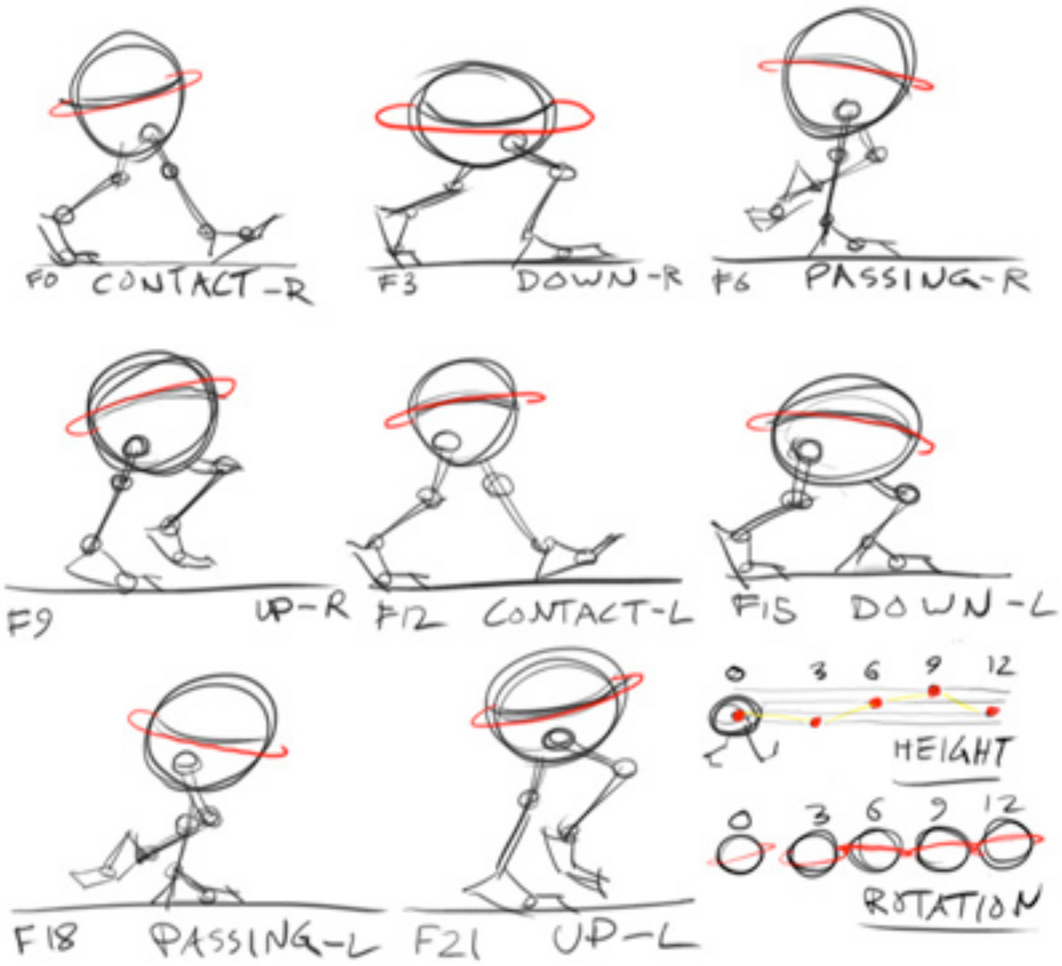

Key frames (created first)
Time
Straight ahead order of frame creation

- Keyframe: detailed planning of key poses and actions
  - greater precision
  - computer does "inbetweening"
- Straight-ahead
  - more spontaneous

key frame animation - detailed planning of key poses and actions
straight-ahead - more spontaneous result

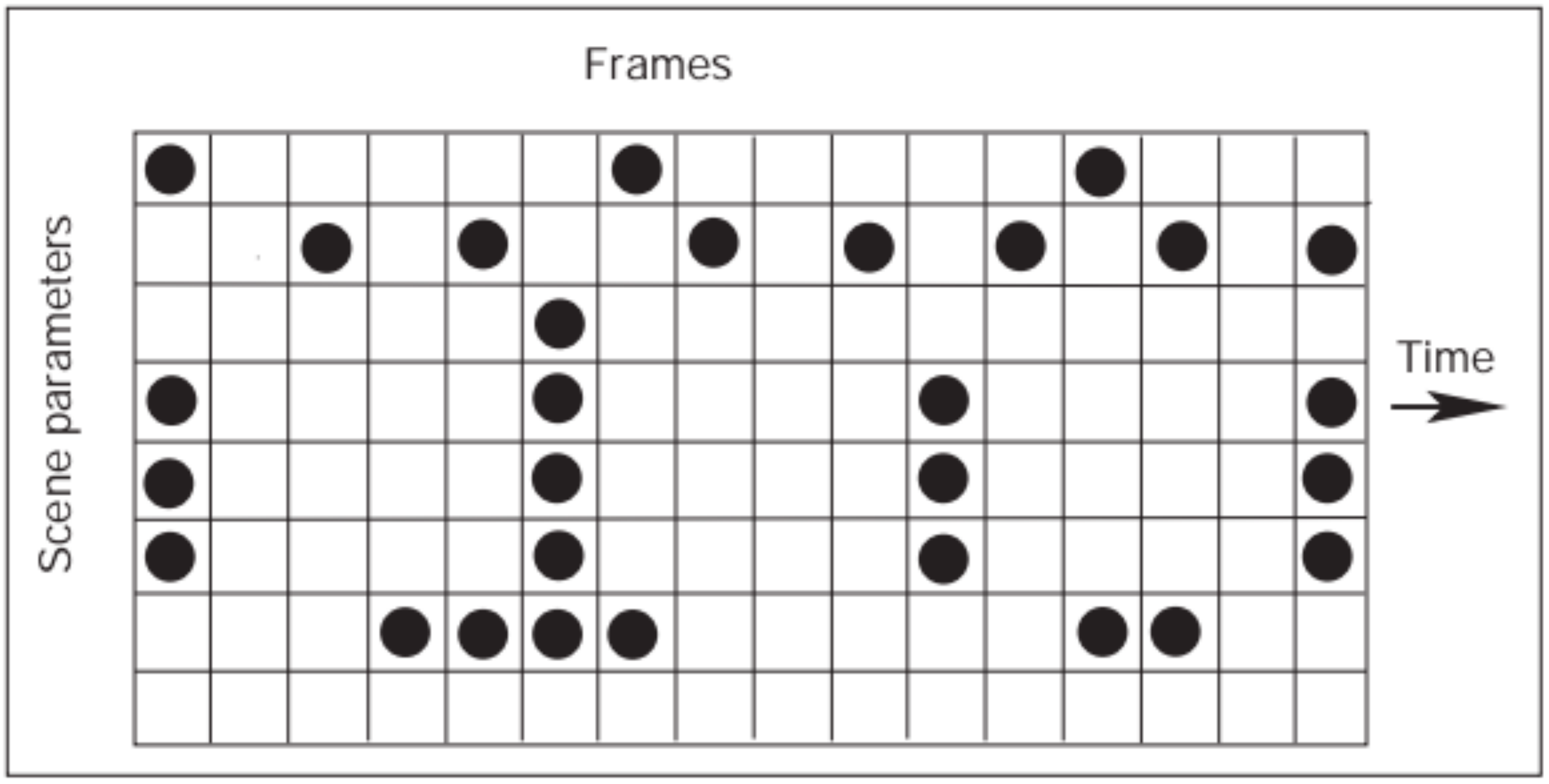
Keyframe animation

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

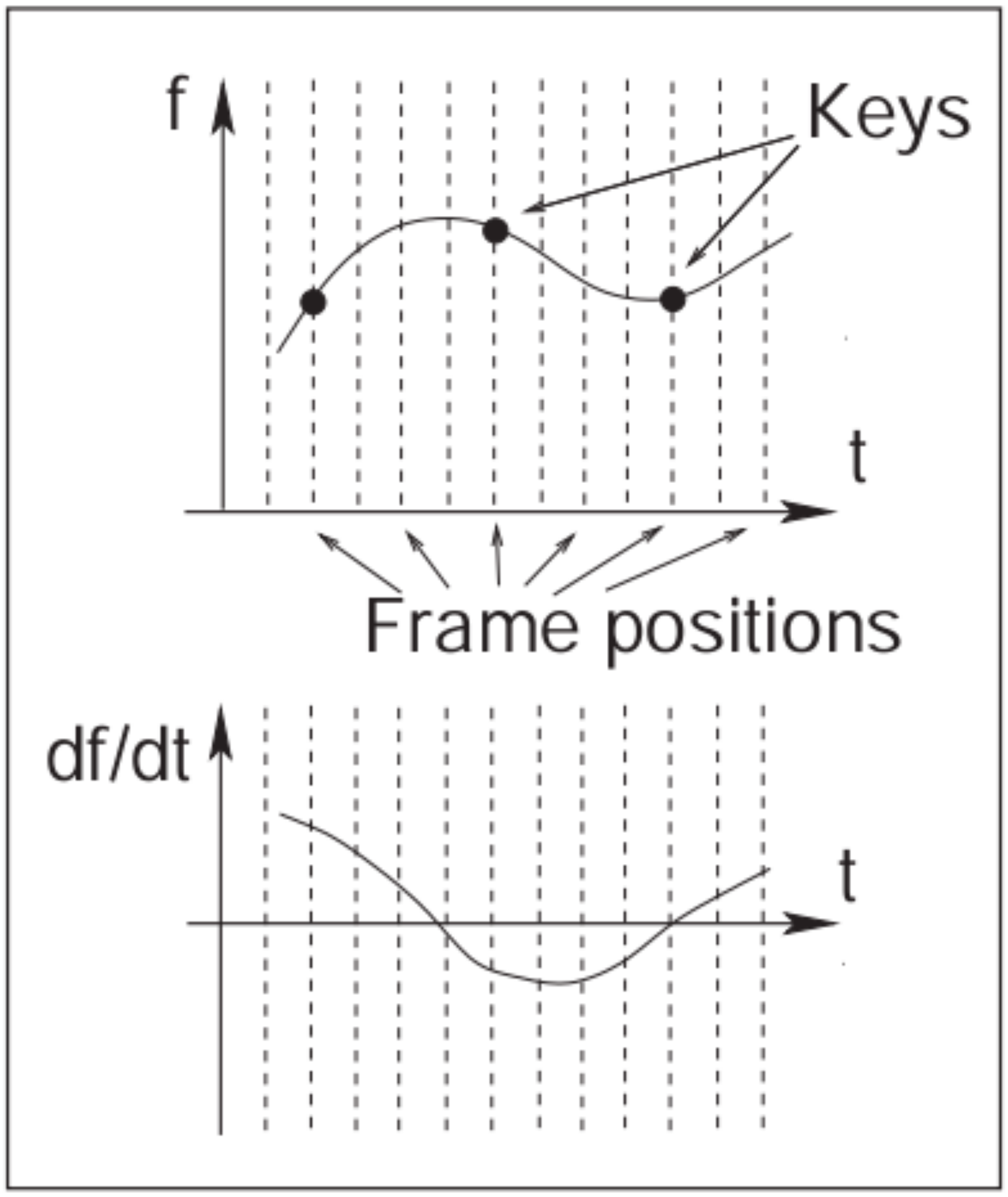


http://anim.tmog.net

Keyframing



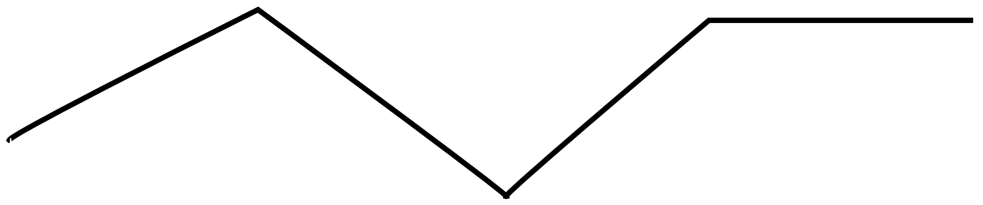
Different patterns of keys may be set for different aspects of the animation



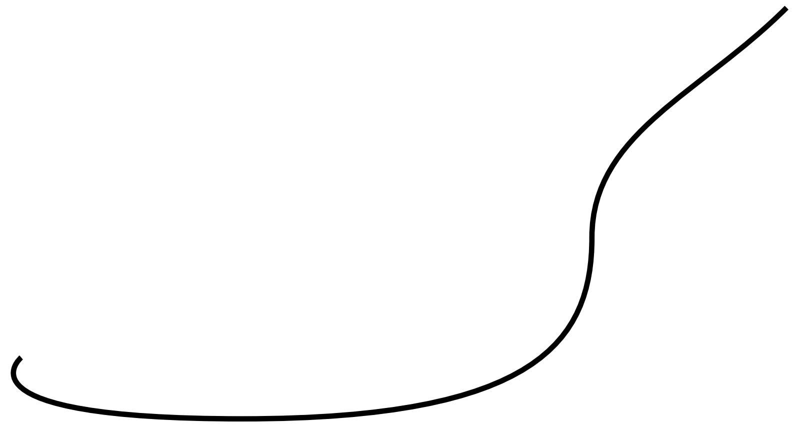
A continuous curve is fit through the key frames
Level of continuity affects motion

Splines may be used for the animation curve
can control curve – add key frames, modify tangents, etc.

Interpolation of keyframes

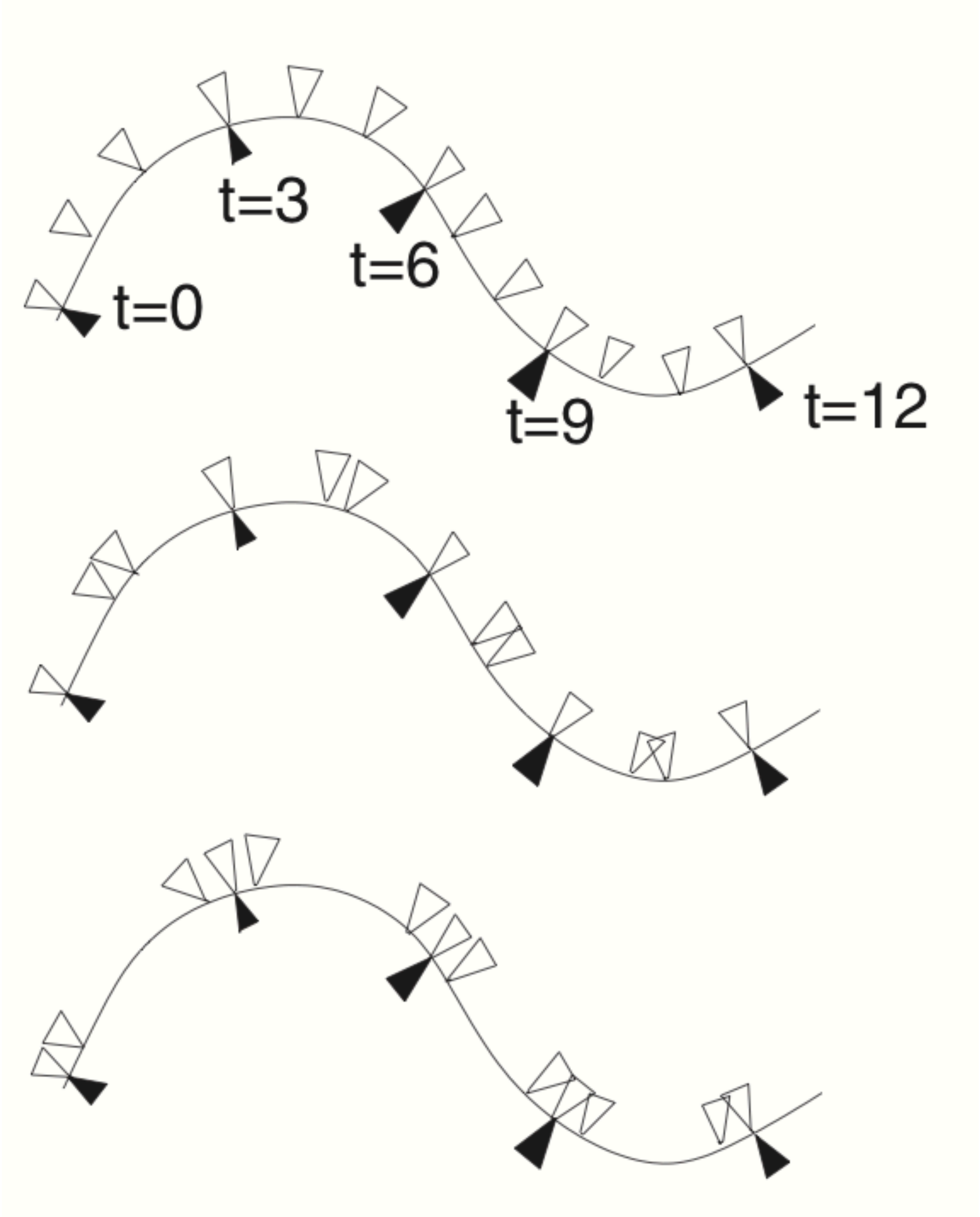


linear interpolation



spline interpolation

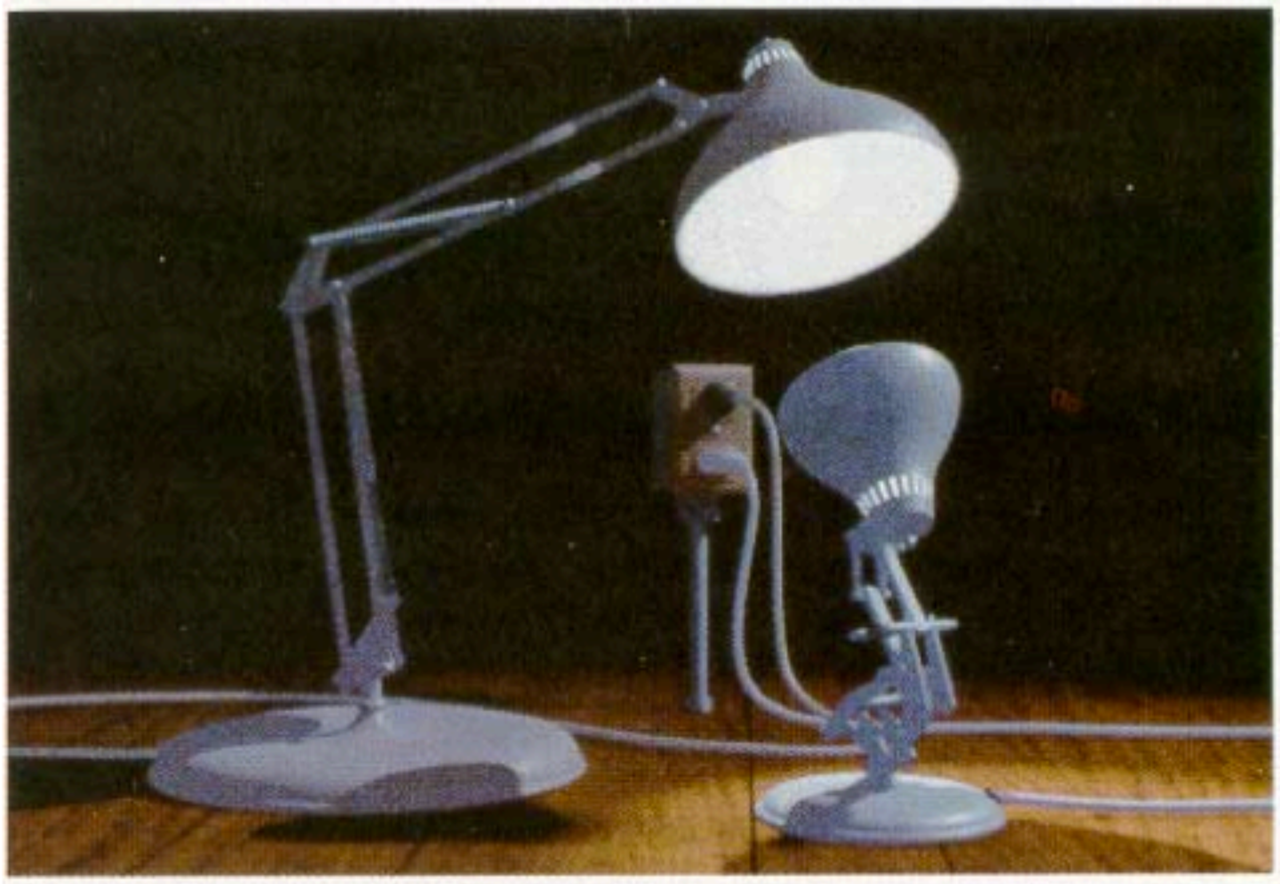
Straightforward to interpolate position but what about orientation?

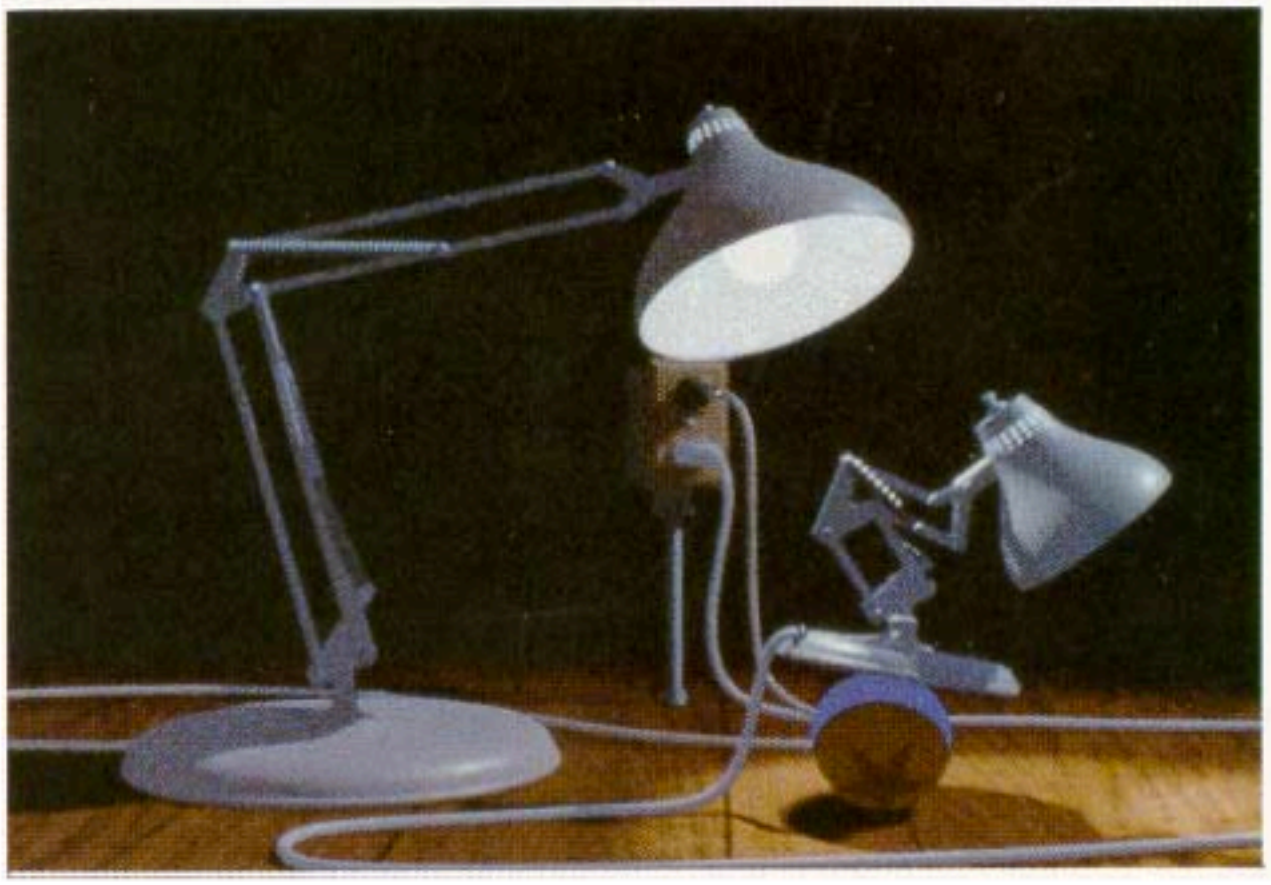


[Shirley and Marschner]

need to consider
both
shape of motion
and
speed of motion

Luxo Jr.

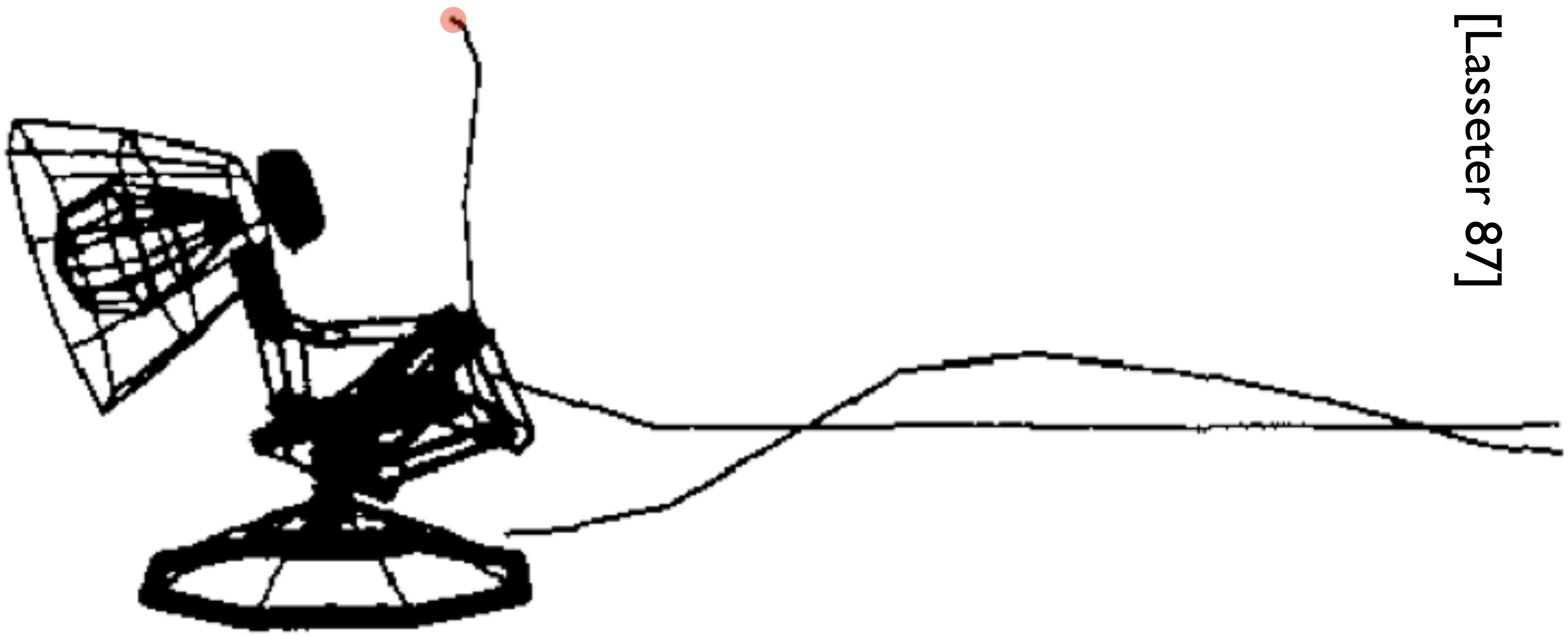




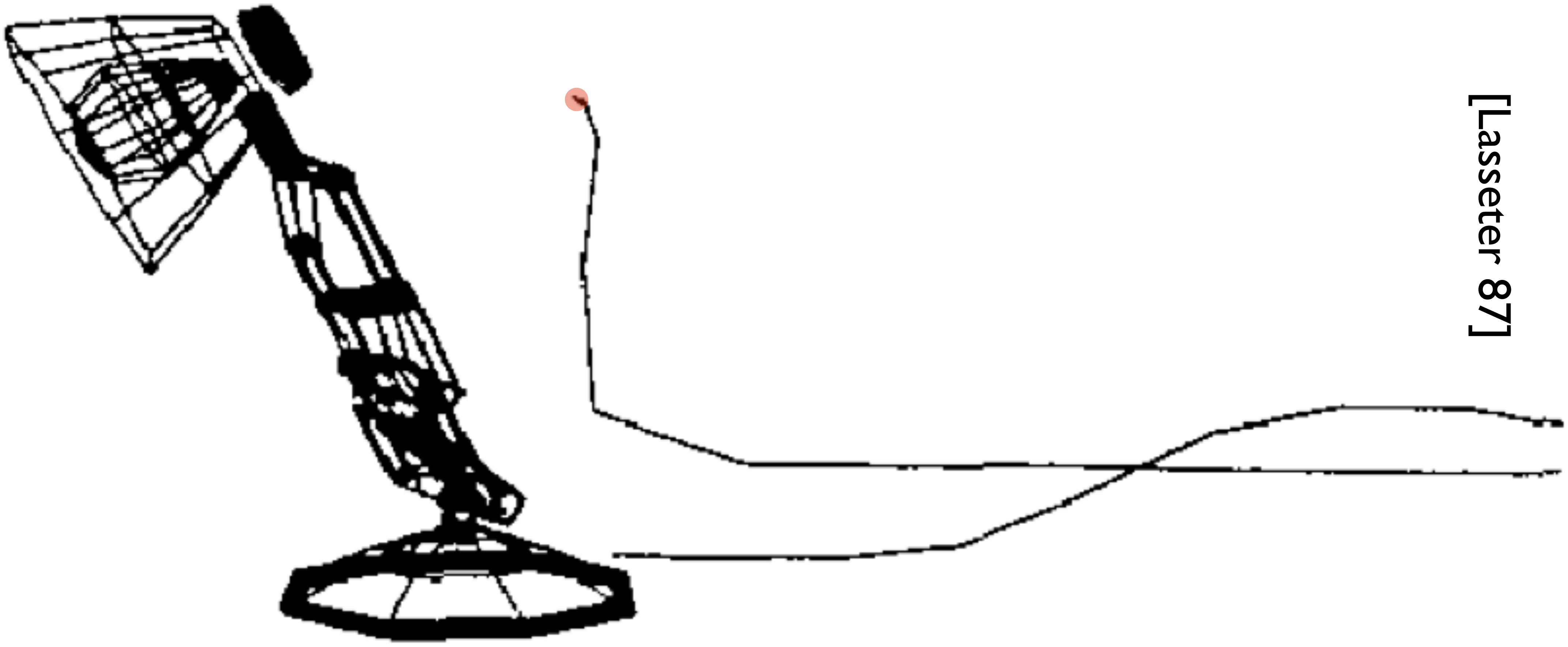
link: https://www.youtube.com/watch?v=wYfYtV_2ezs

- Overlapping action on cord
- squash and stretch in hop
- side-staging for clarity
- timing: ease in and ease out on the hop

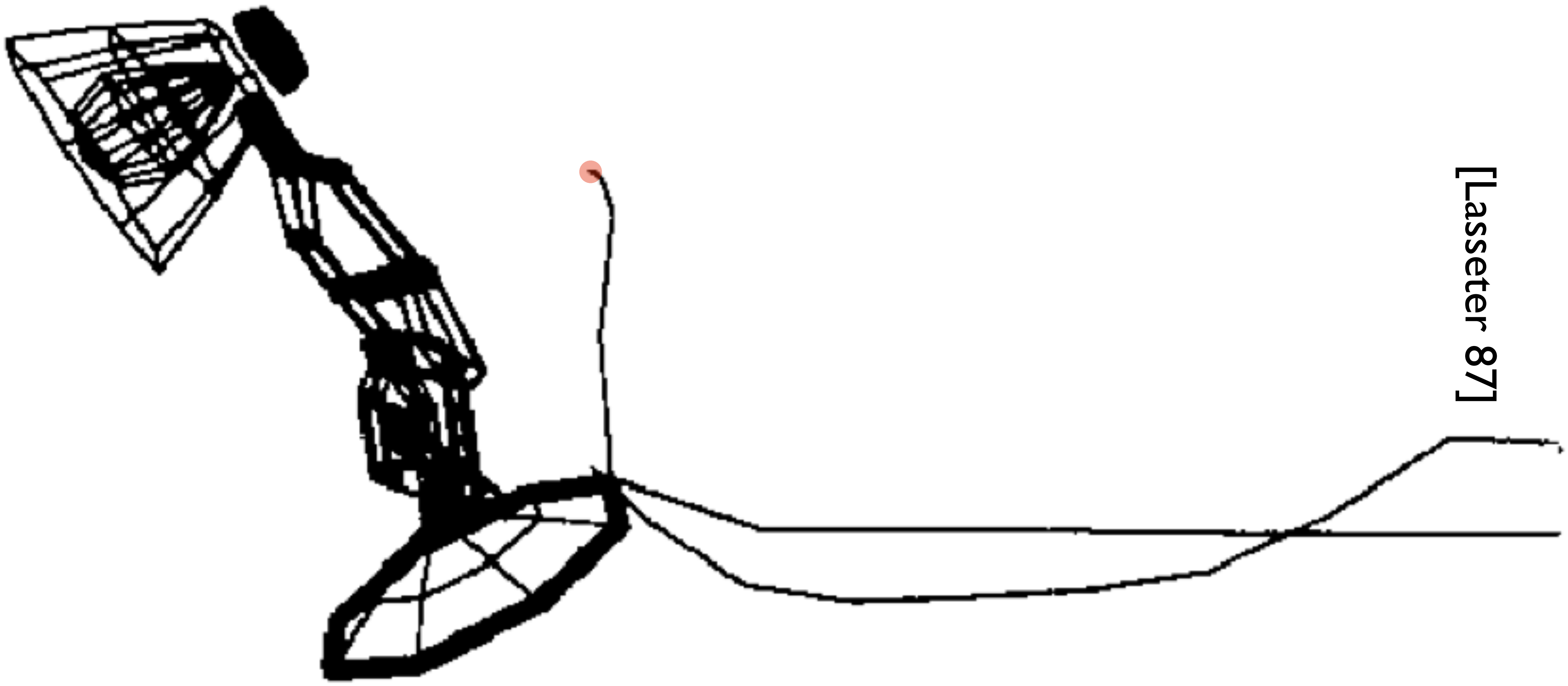
https://www.youtube.com/watch?v=wYfYtV_2ezs



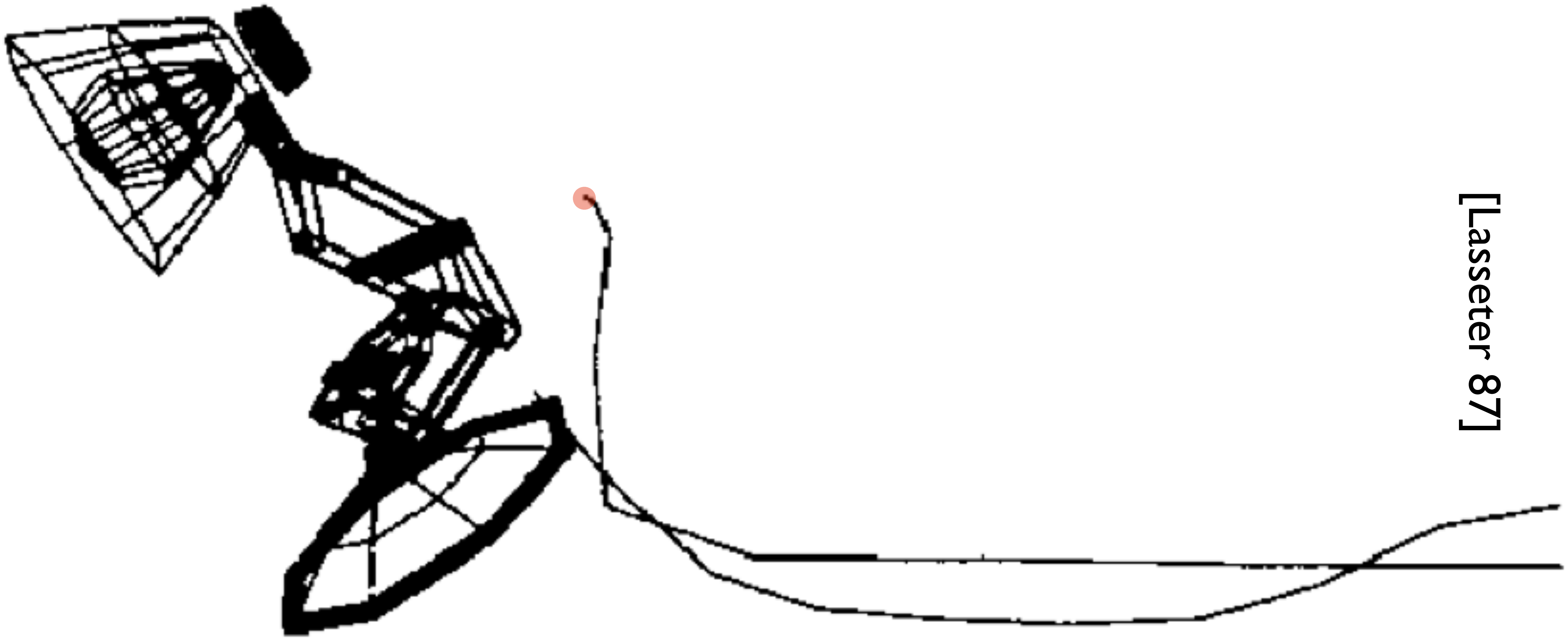
[Lasseter 87]



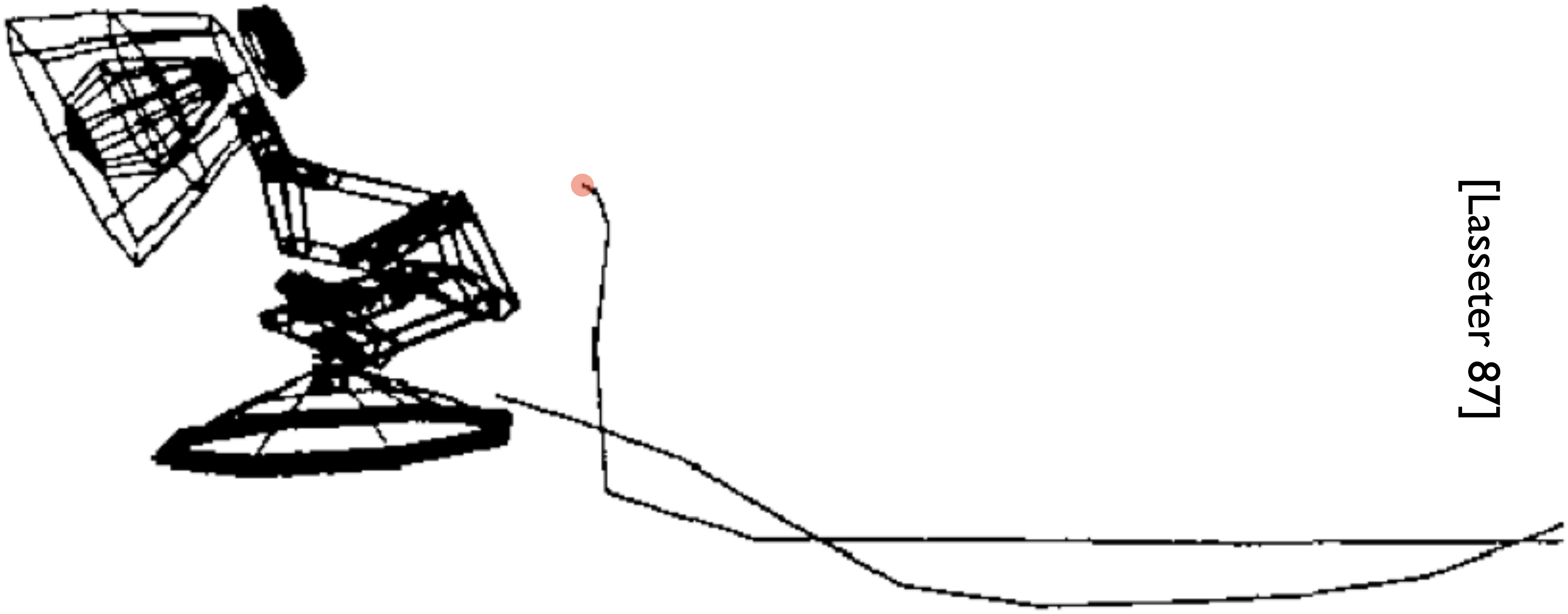
[Lasseter 87]



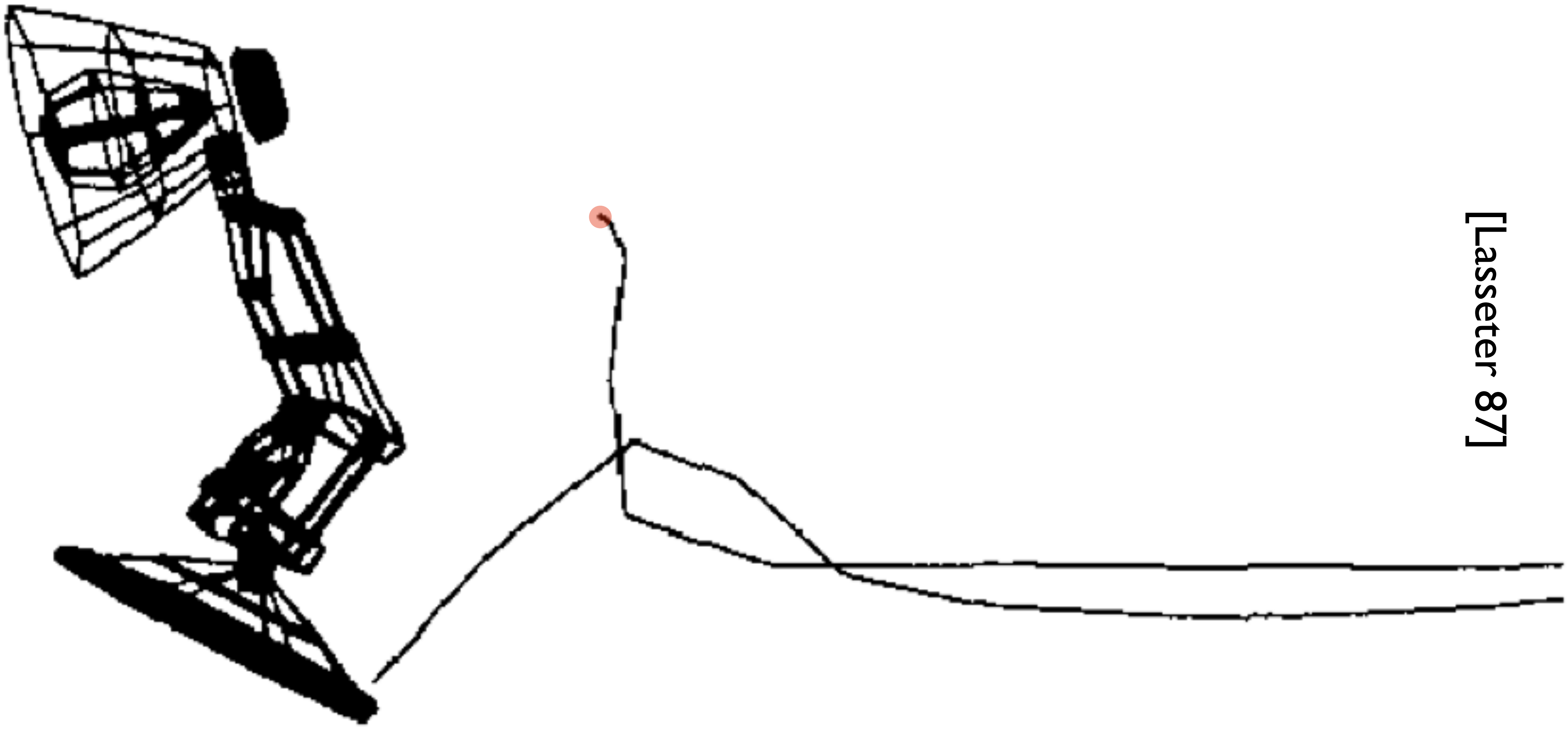
[Lasseter 87]



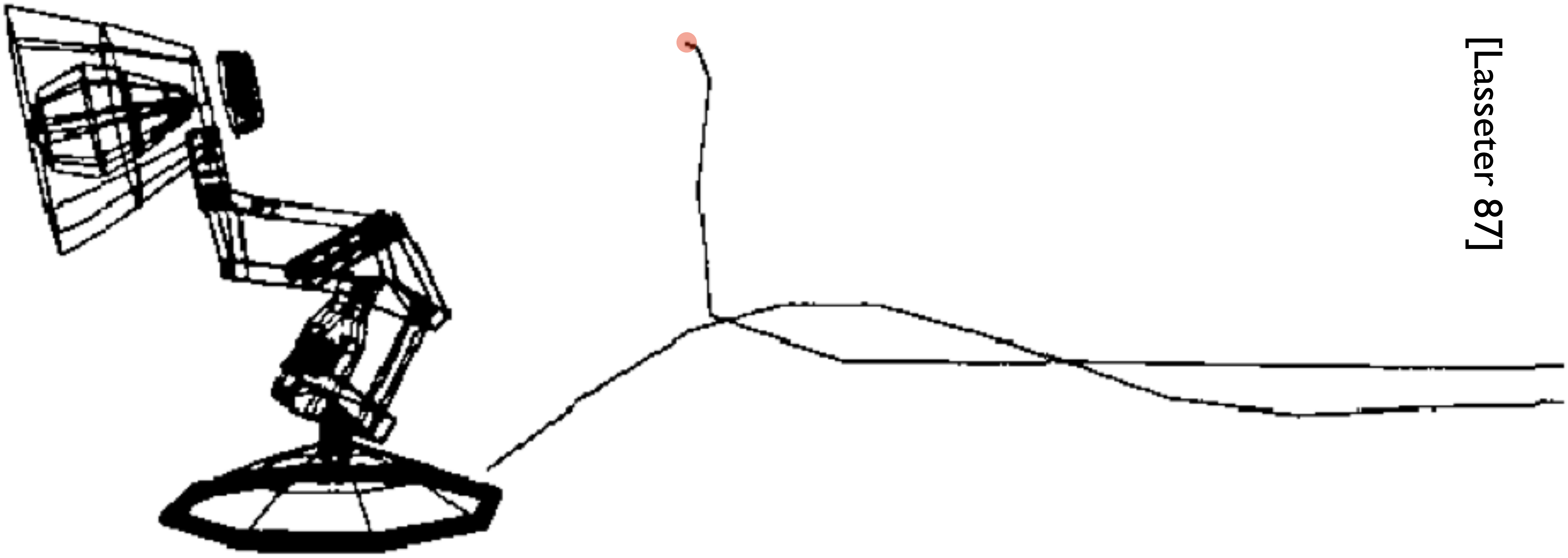
[Lasseter 87]



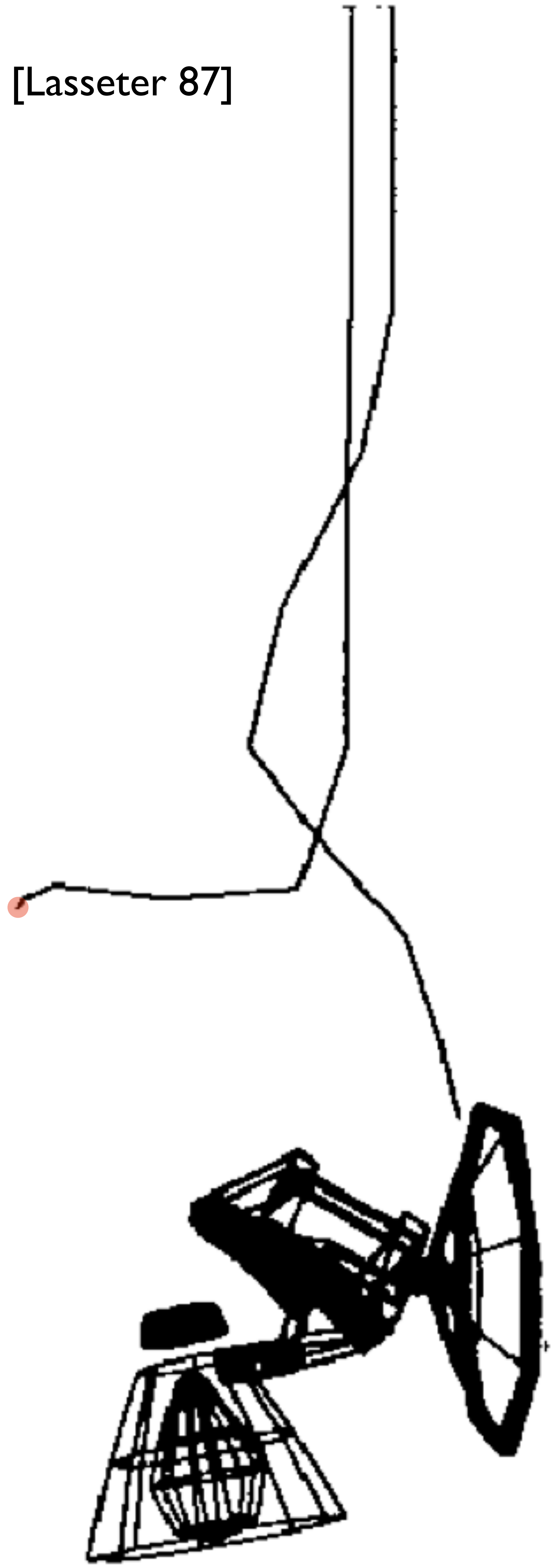
[Lasseter 87]



[Lasseter 87]

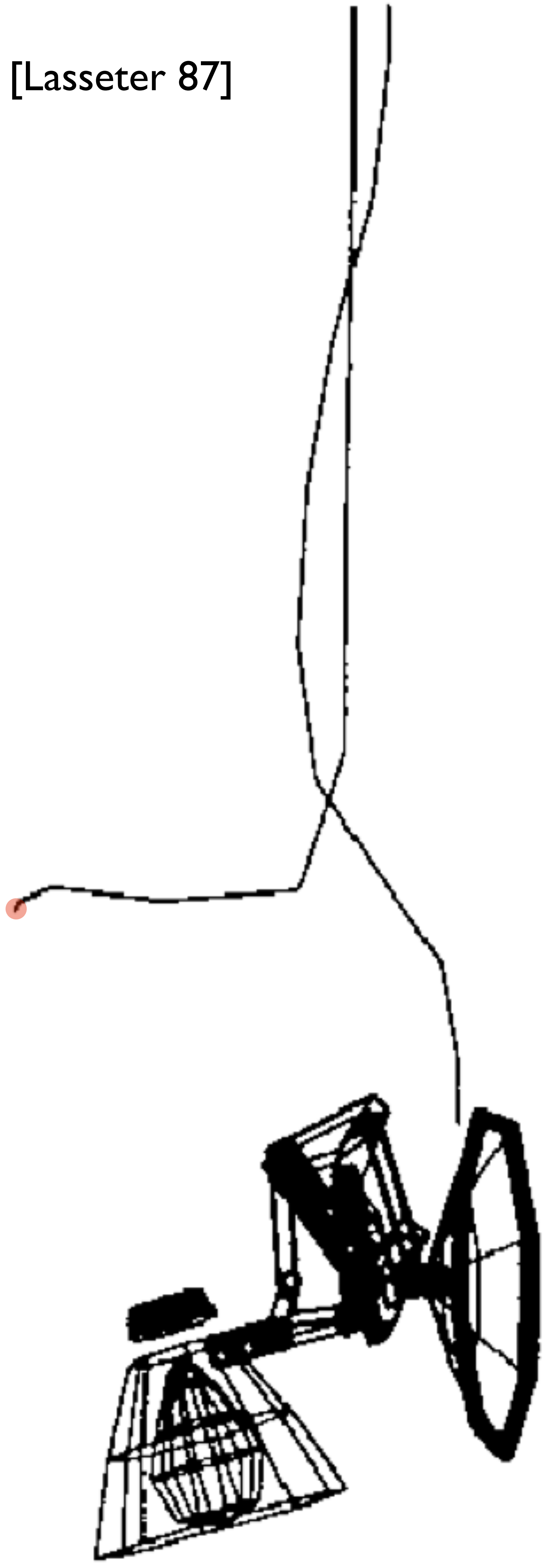


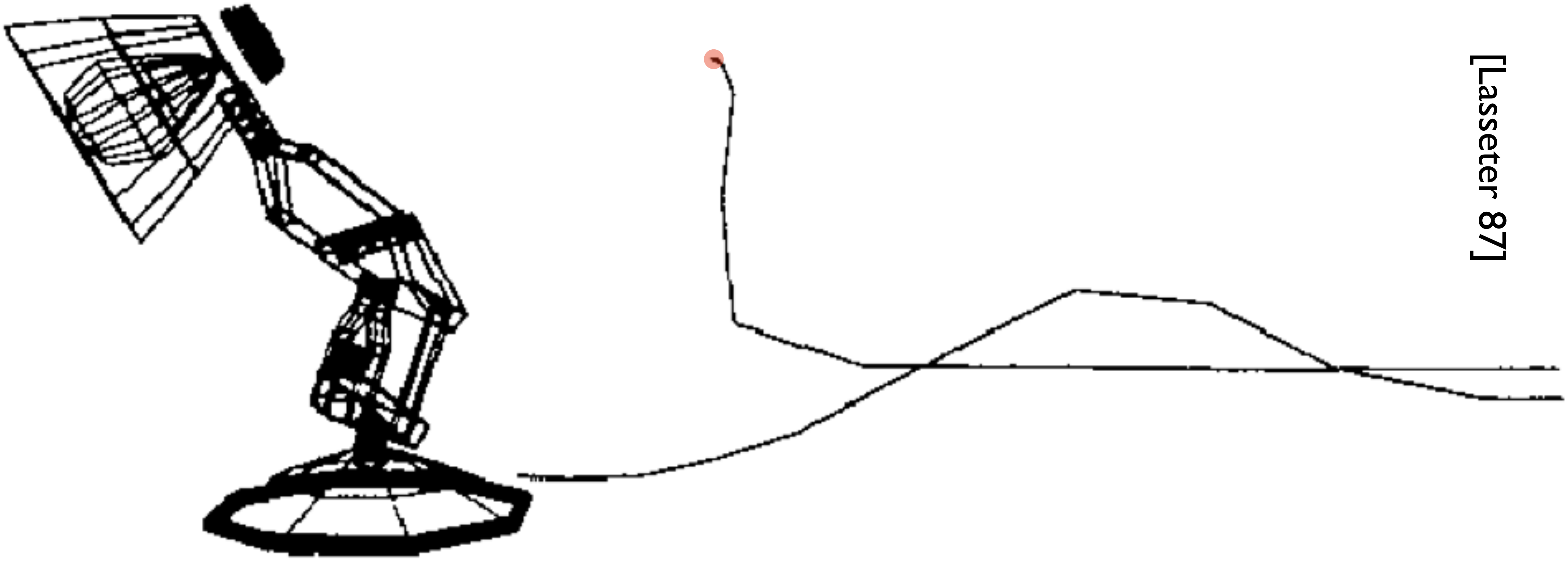
[Lasseter 87]



[Lasseter 87]

[Lasseter 87]

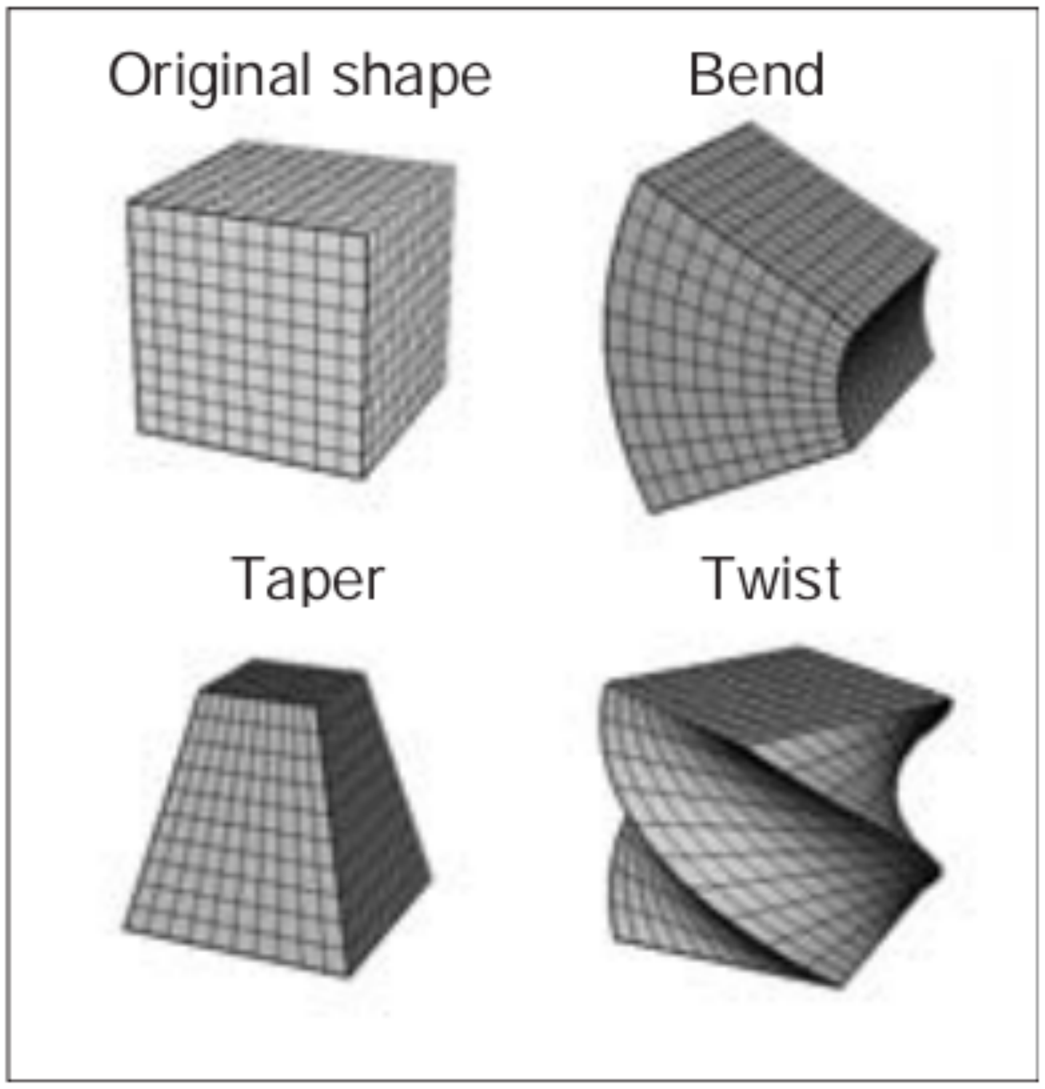




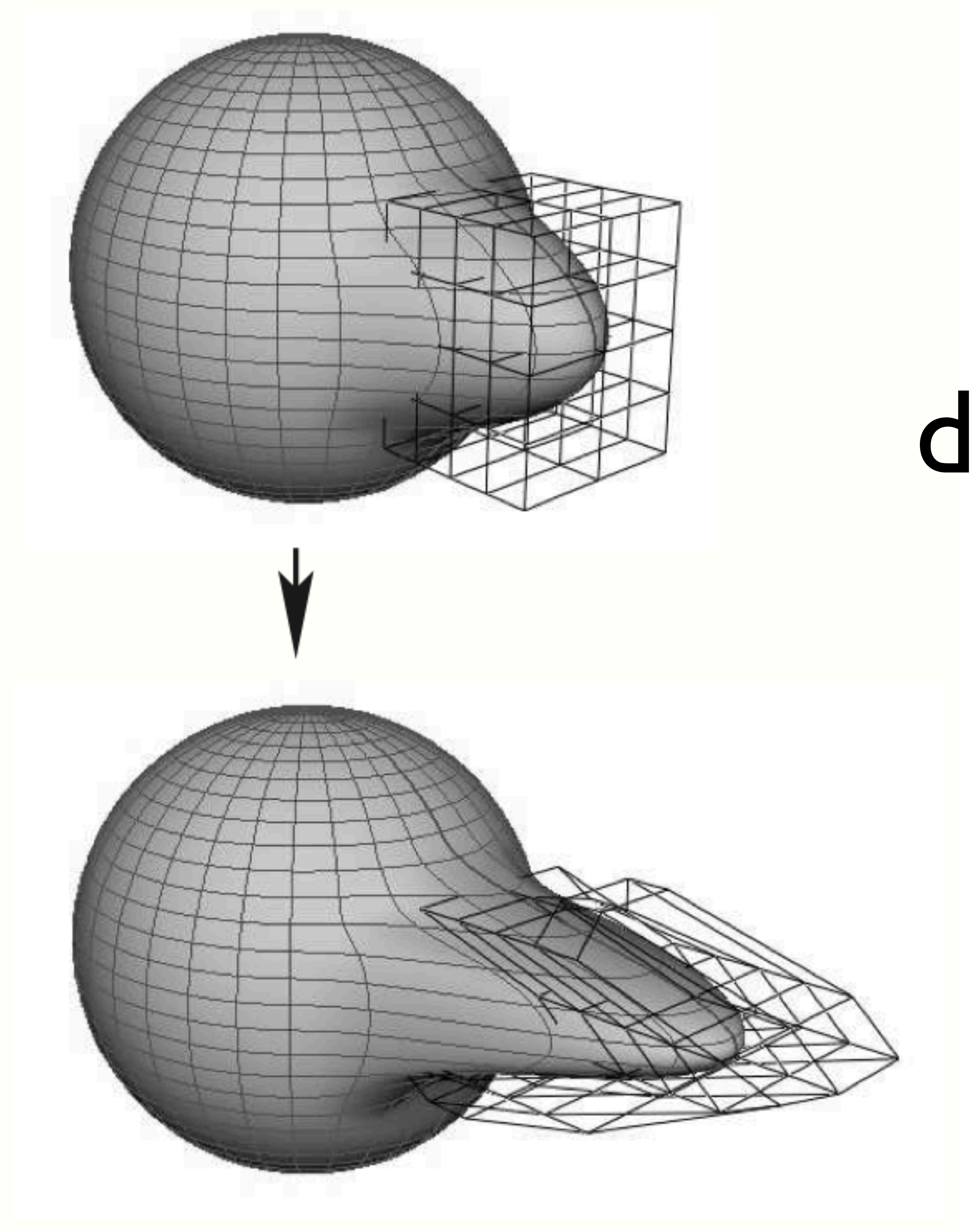
[Lasseter 87]

Deformations

Deformations



- •Can move individual vertices
- •Move a "seed vertex" and adjust a local area based on attenuation function
- •Use mathematical functions to transform original shape, e.g., figure



[Shirley and Marschner]

free form deformation

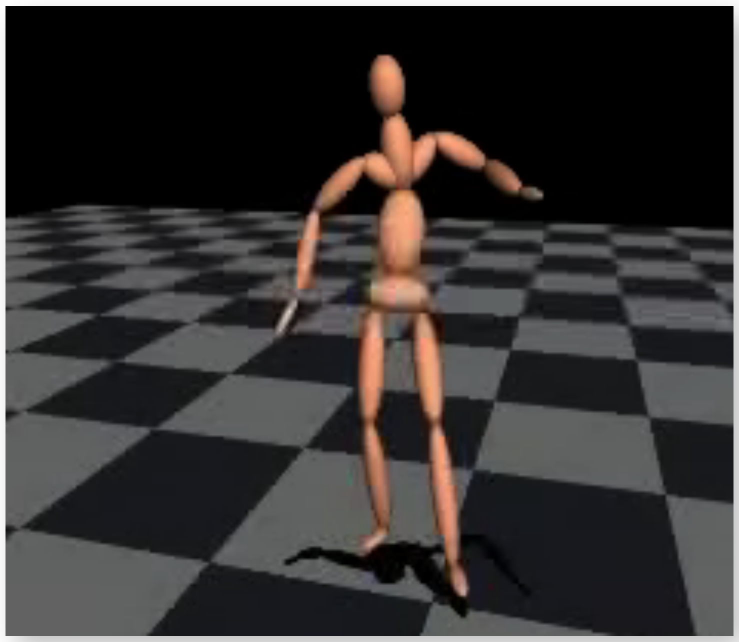
[Sederberg 1986]

Character Animation

Animation of articulated figures

- keyframing
- specialized deformation techniques
- physics-based simulation

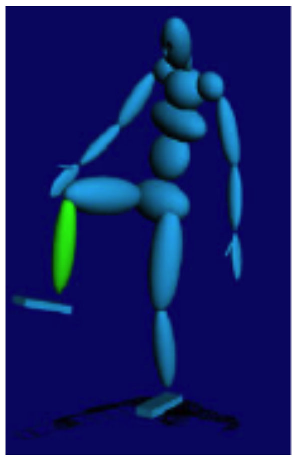
Keyframe character DOFs

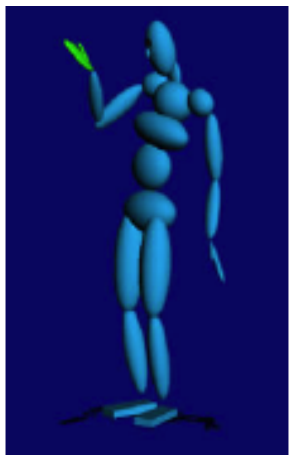


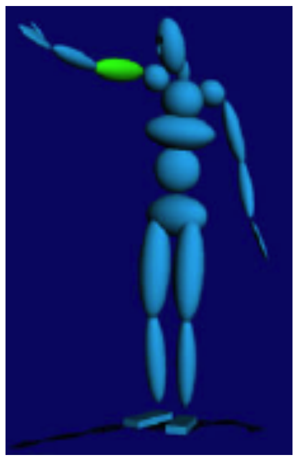
3 translational DOFs

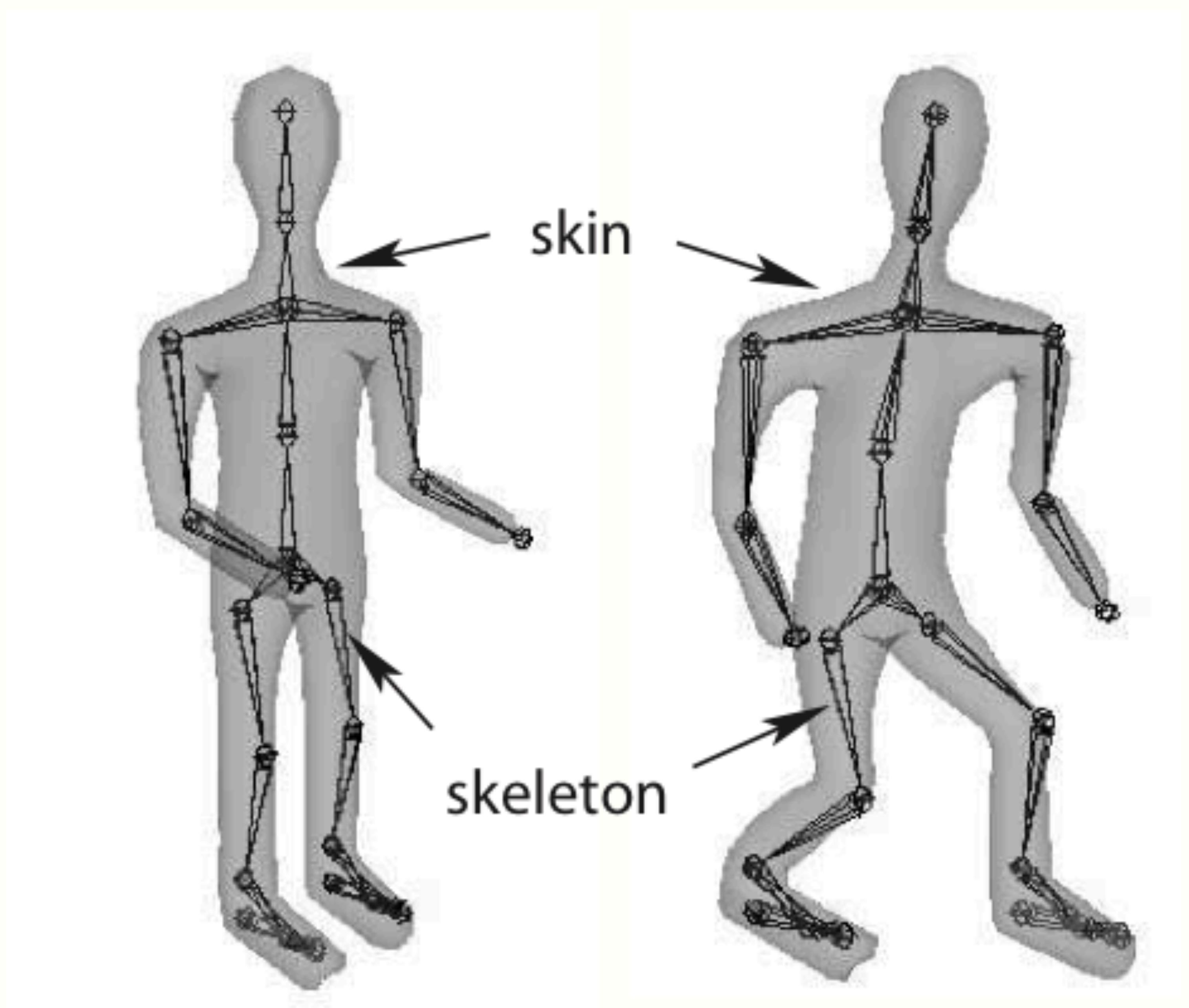
48 rotational DOFs

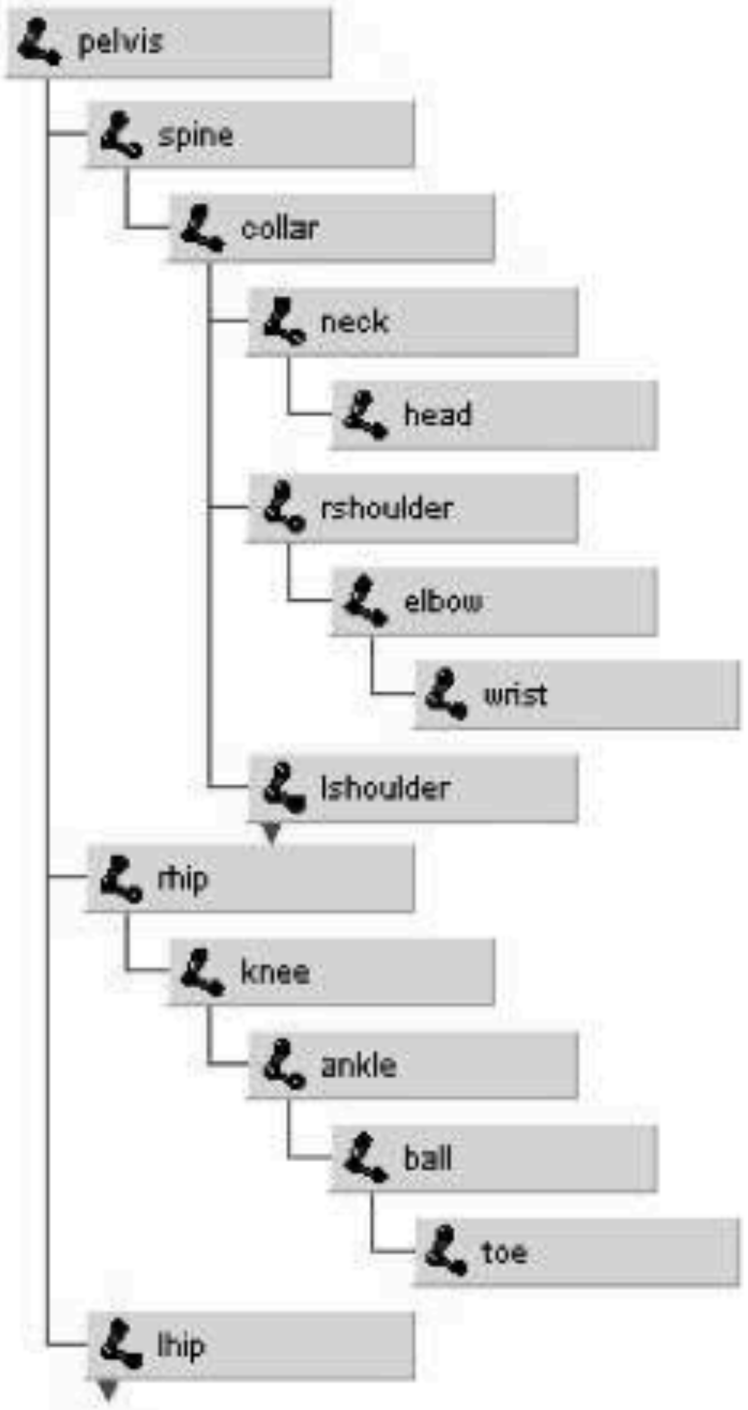
Each joint can have up to 3 DOFs











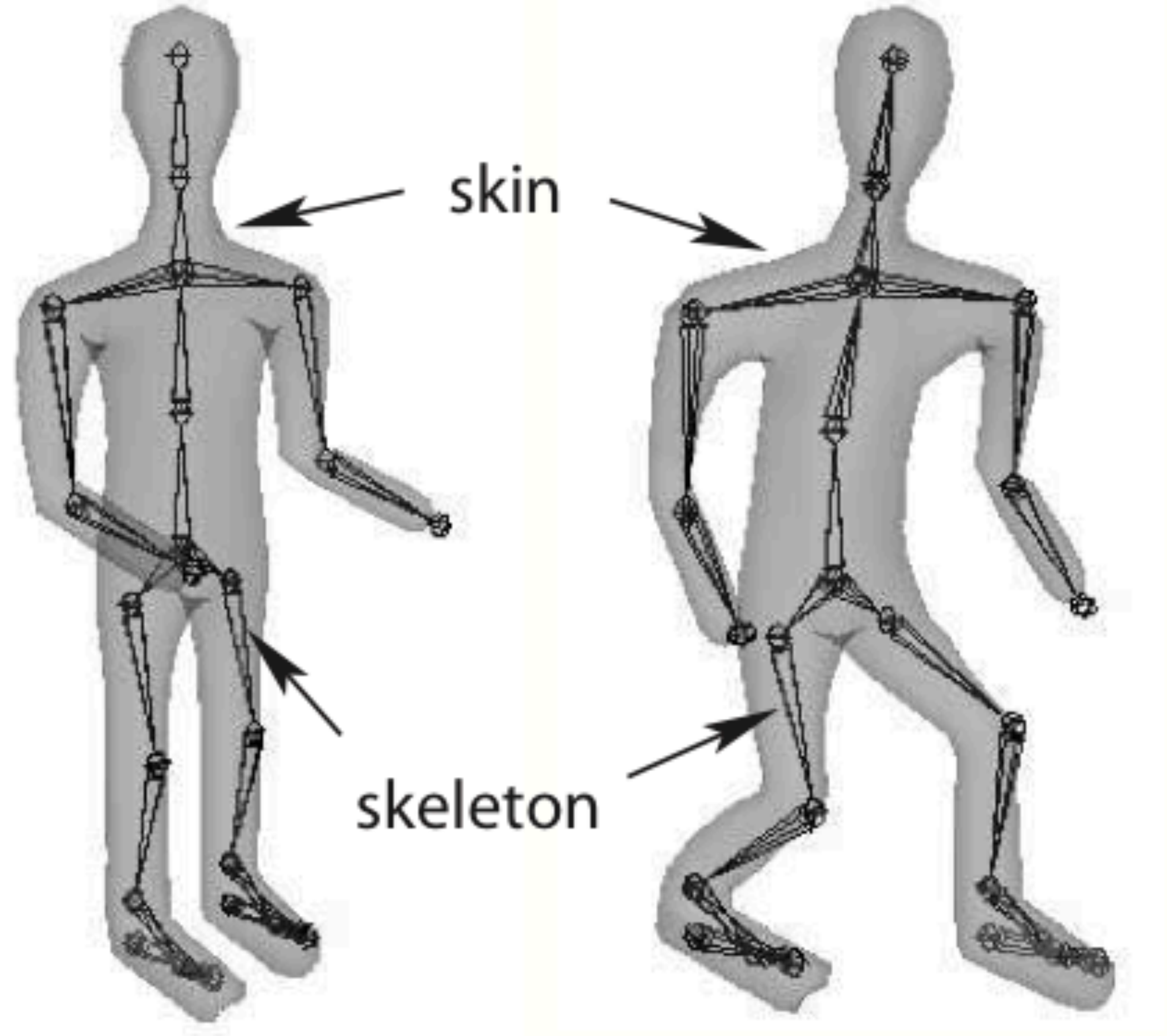
- pelvis
  - spine
    - collar
      - neck
        - head
      - rshoulder
        - elbow
          - wrist
      - lshoulder
  - rhip
    - knee
      - ankle
        - ball
          - toe
  - lhip

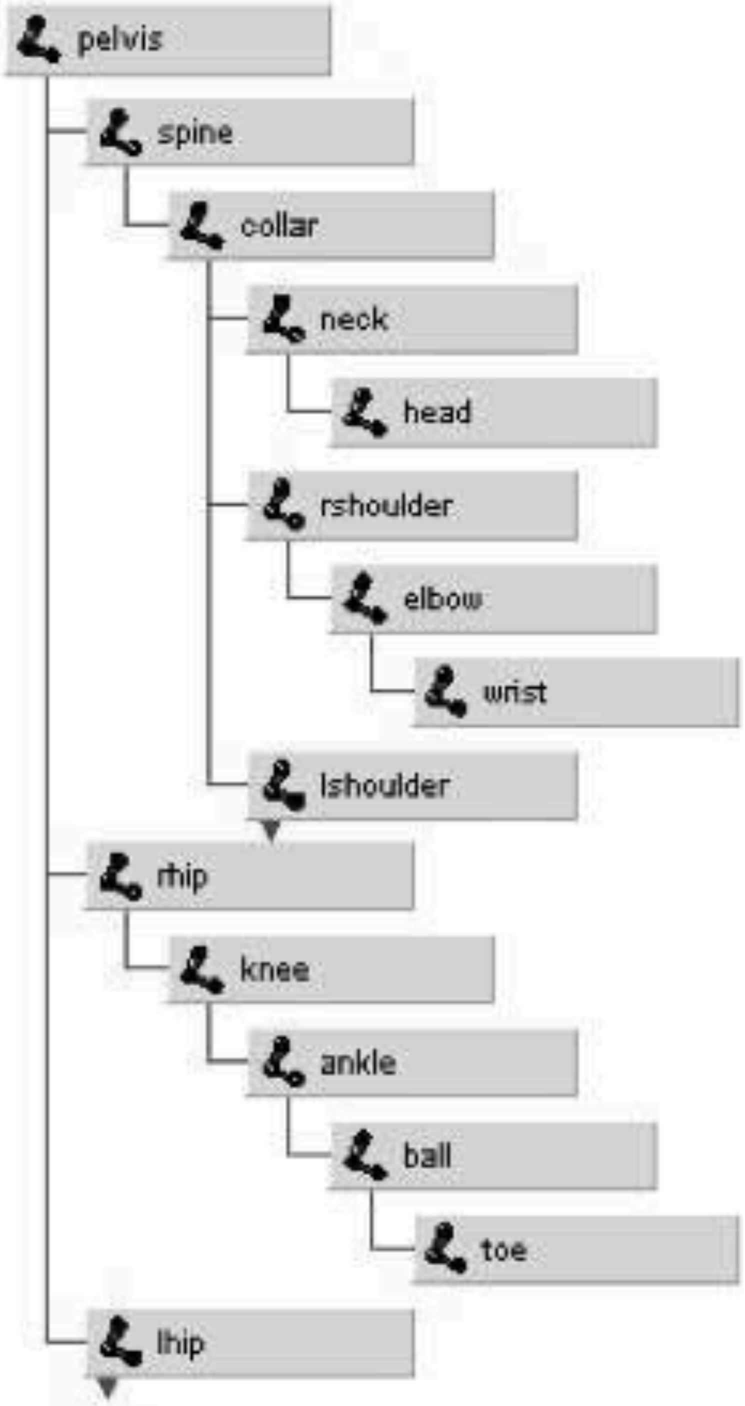
[Shirley and Marschner]

Joint hierarchy serves as kinematic abstraction of character

- forward kinematics - set joint angles starting at a root and working down the tree
- inverse kinematics - set end effector (e.g., hand) and solve for state of dofs up to root

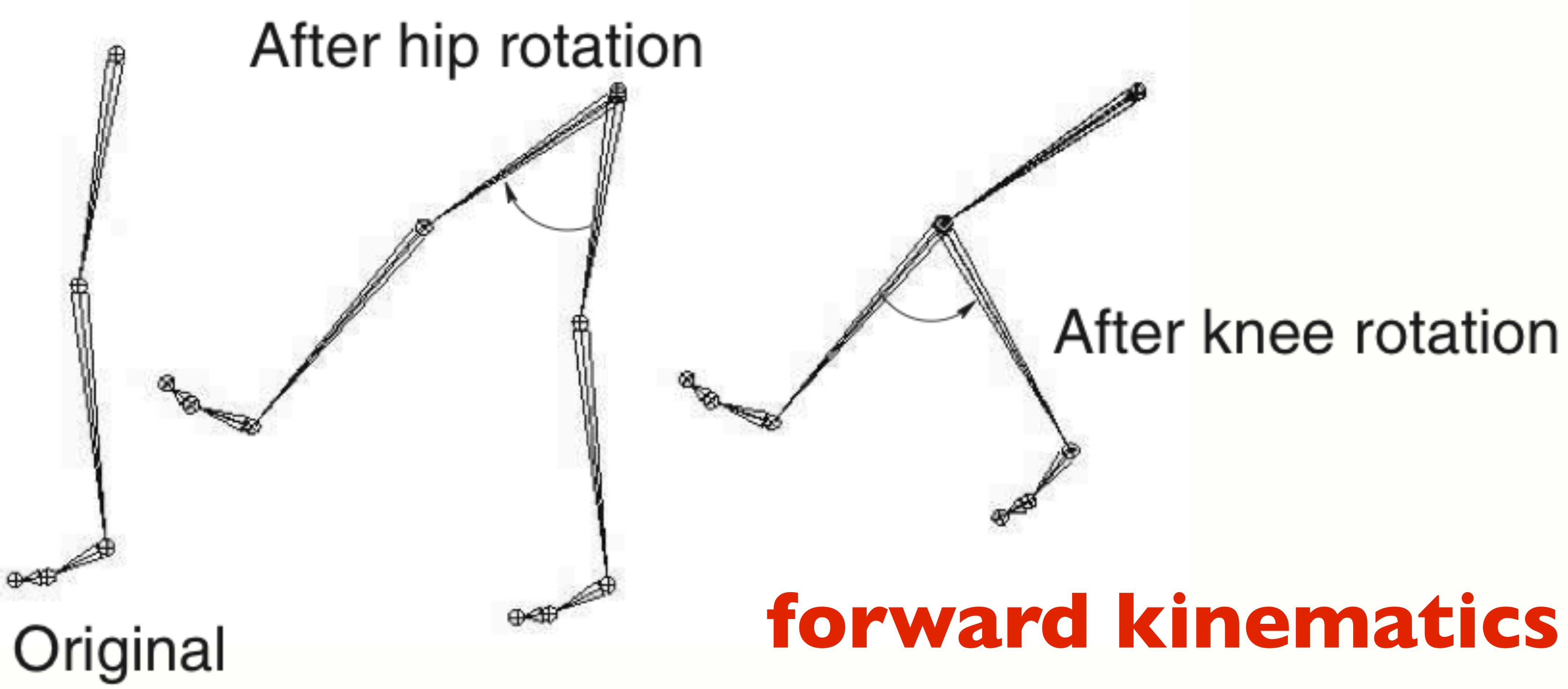
repositioning skeleton deforms attached skin



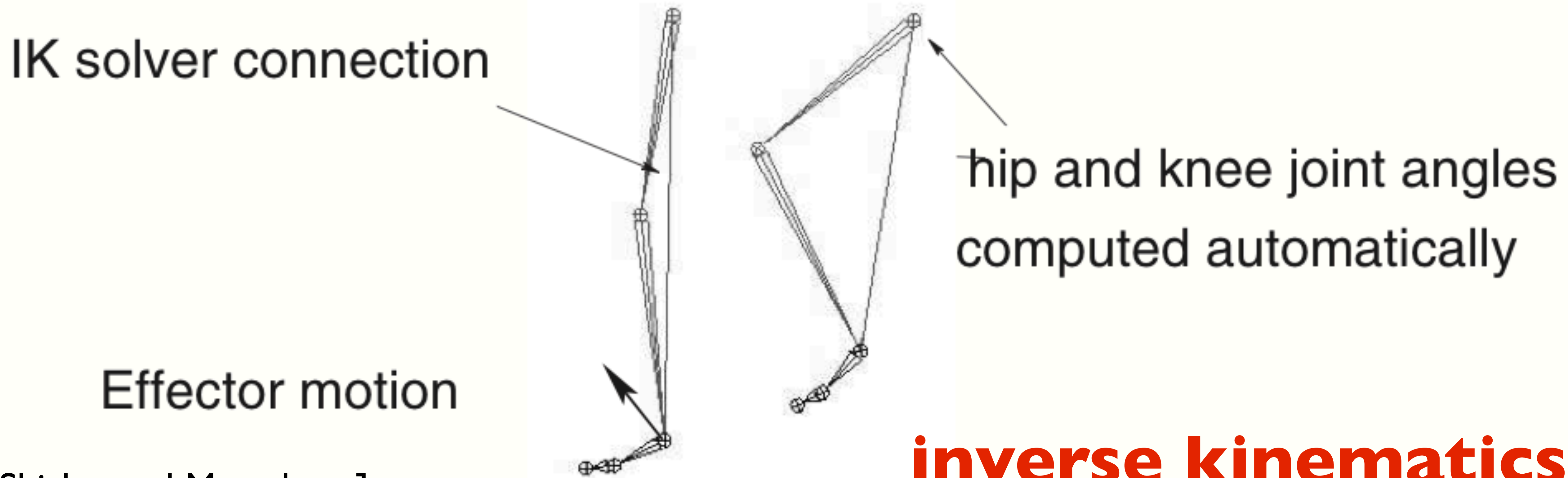


[Shirley and Marschner]

- forward kinematics - set joint angles starting at a root and working down the tree
  - stack is a natural data structure to use when processing the transformations in a depth-first approach
- inverse kinematics - set end effector (e.g., hand) and solve for state of dofs up to root



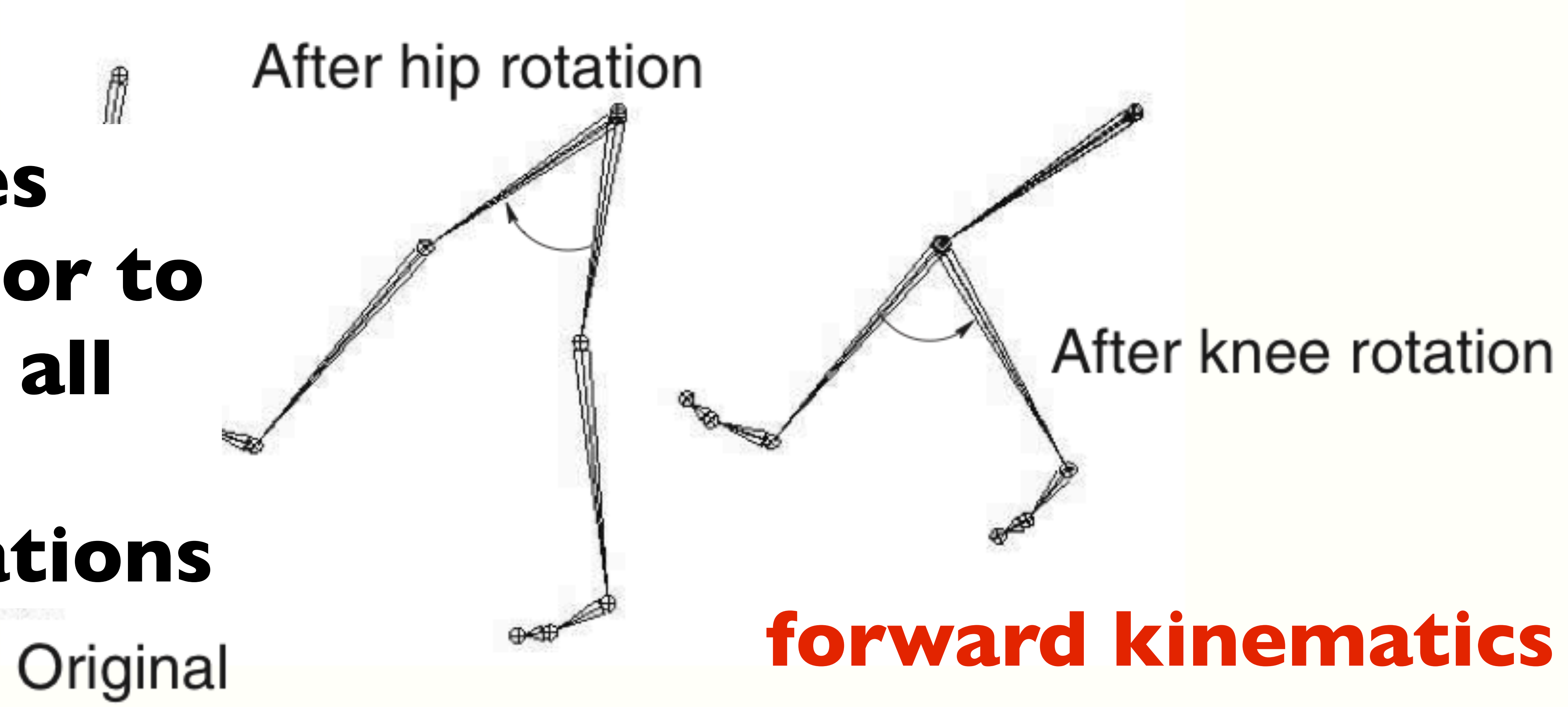
forward kinematics



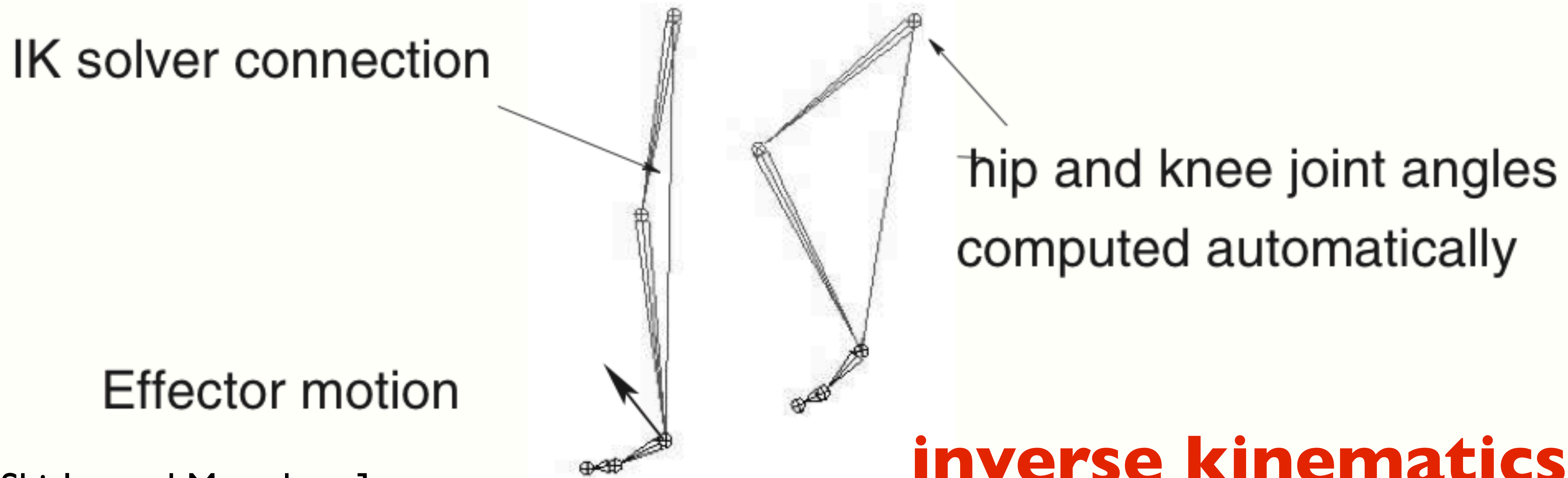
inverse kinematics

[Shirley and Marschner]

requires animator to specify all joint orientations



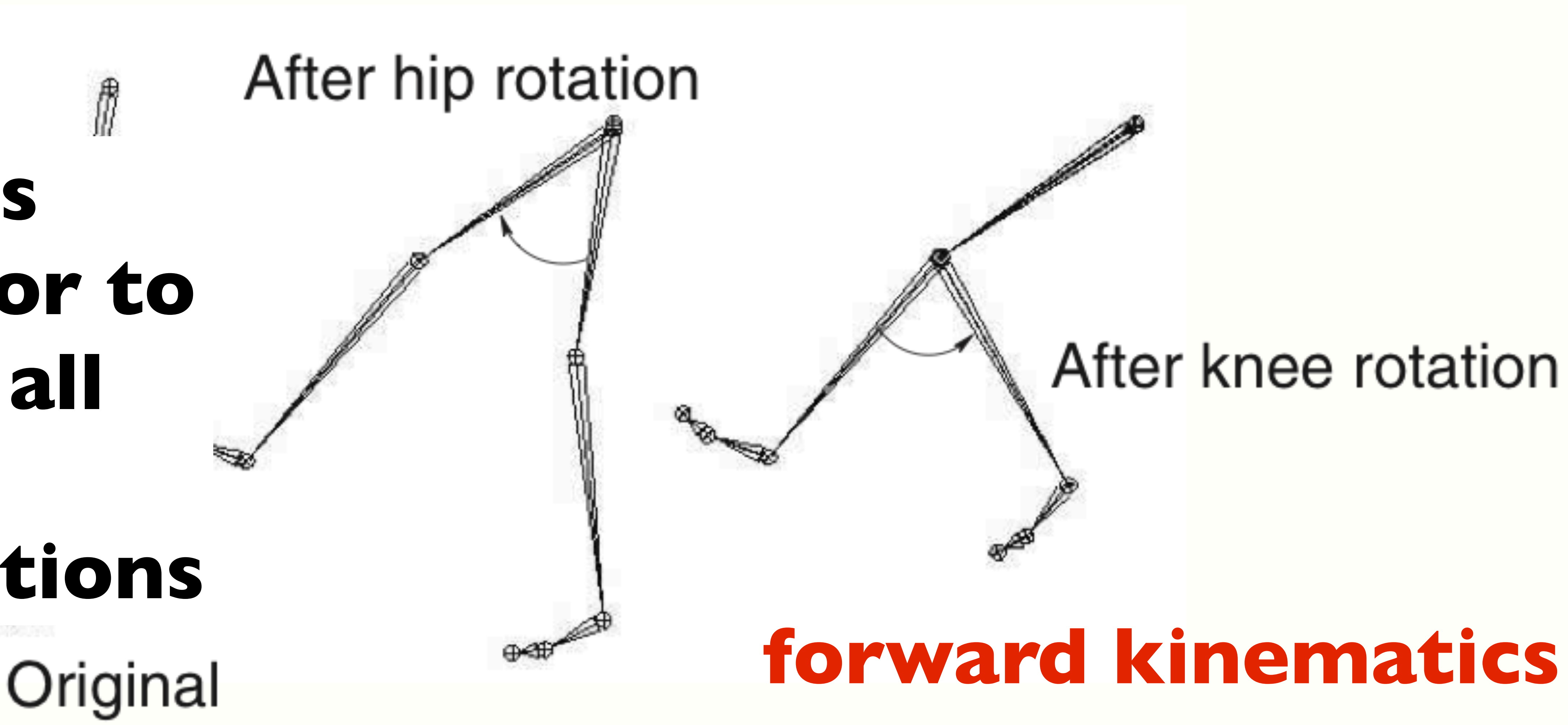
forward kinematics



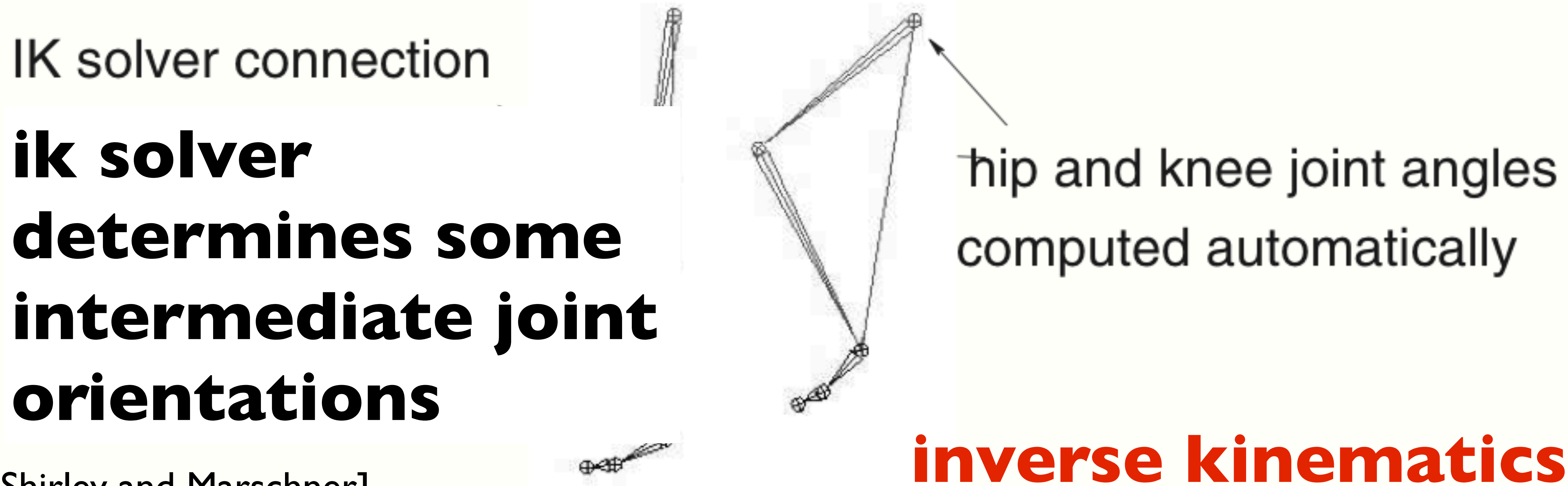
inverse kinematics

[Shirley and Marschner]

requires animator to specify all joint orientations



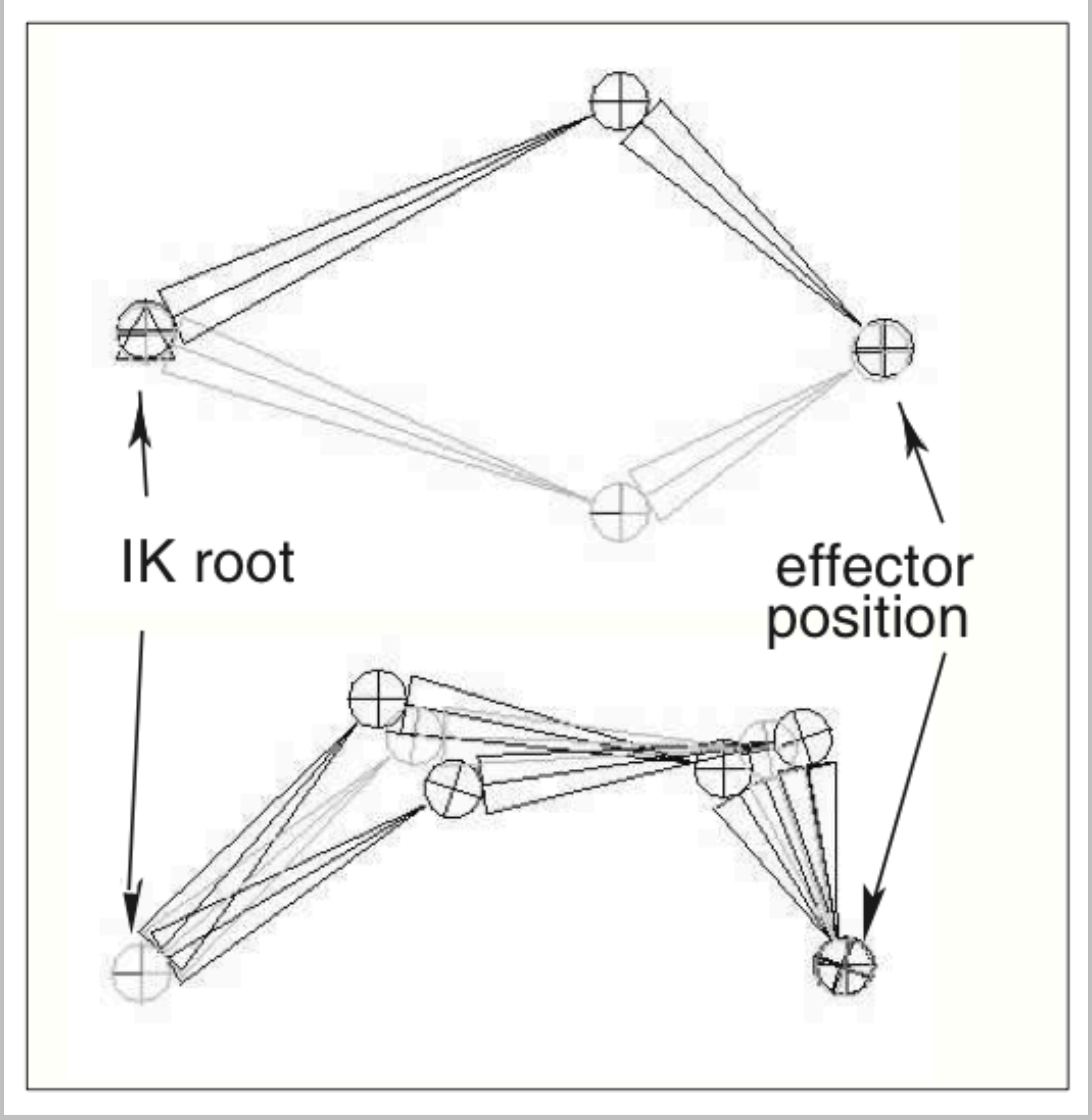
forward kinematics



ik solver determines some intermediate joint orientations

[Shirley and Marschner]

inverse kinematics



multiple possible states of joints

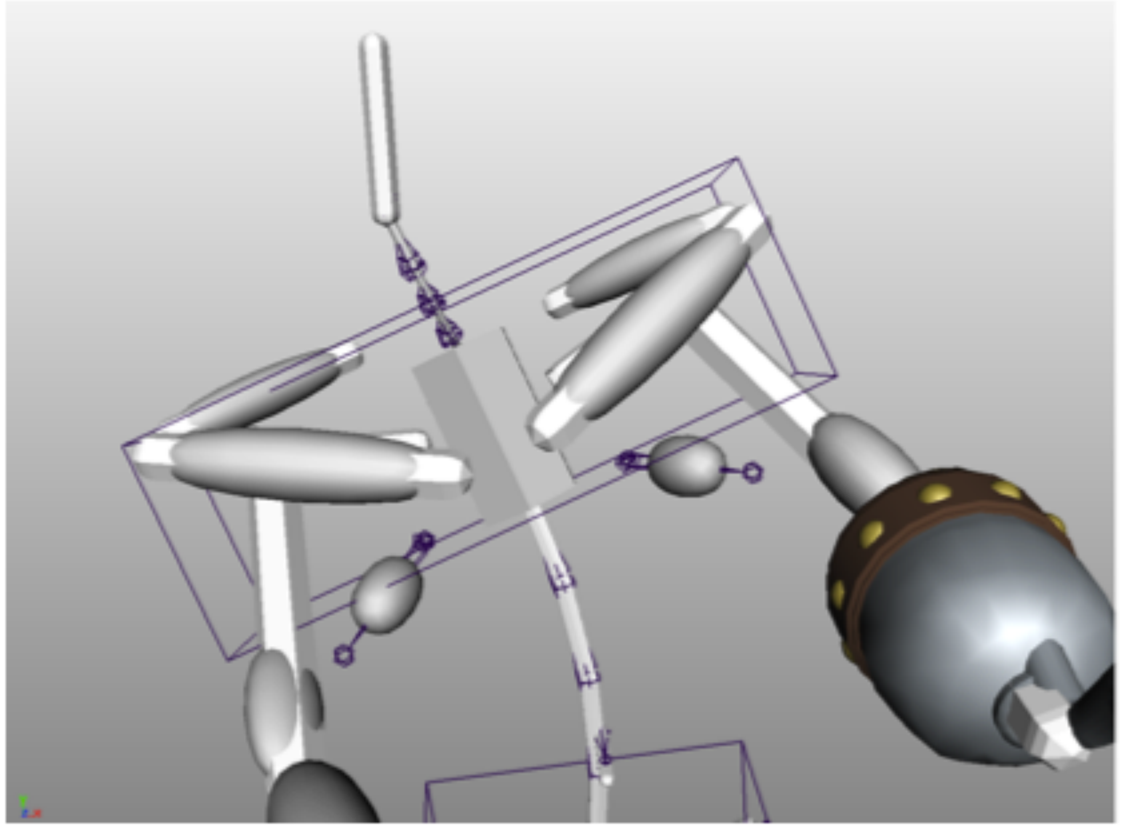
[Shirley and Marschner]

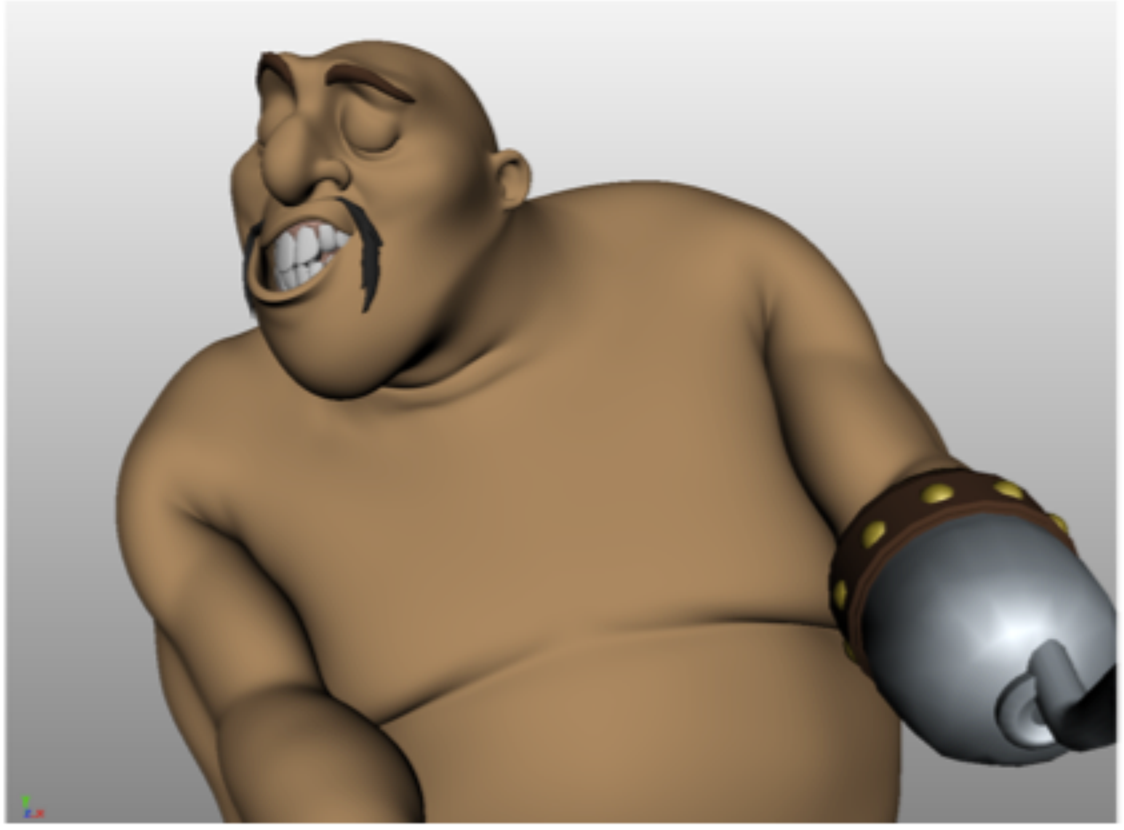
inverse kinematics

Physics-based deformations

Physics-based deformations

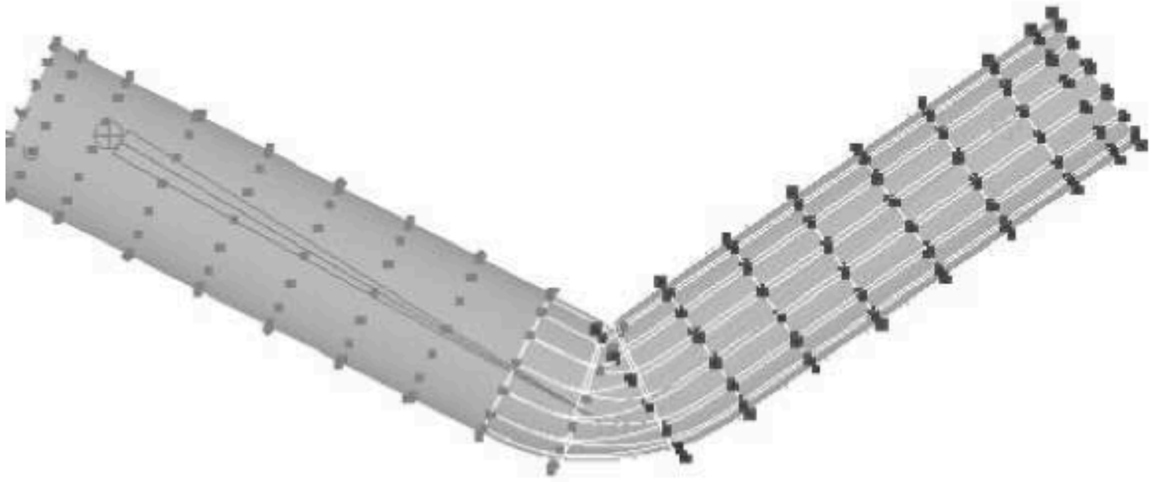
Character Skinning

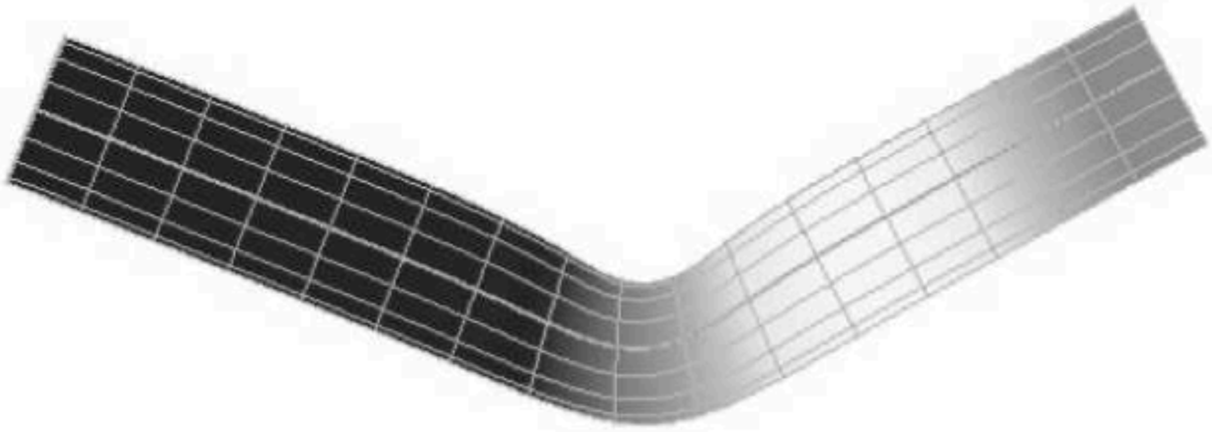


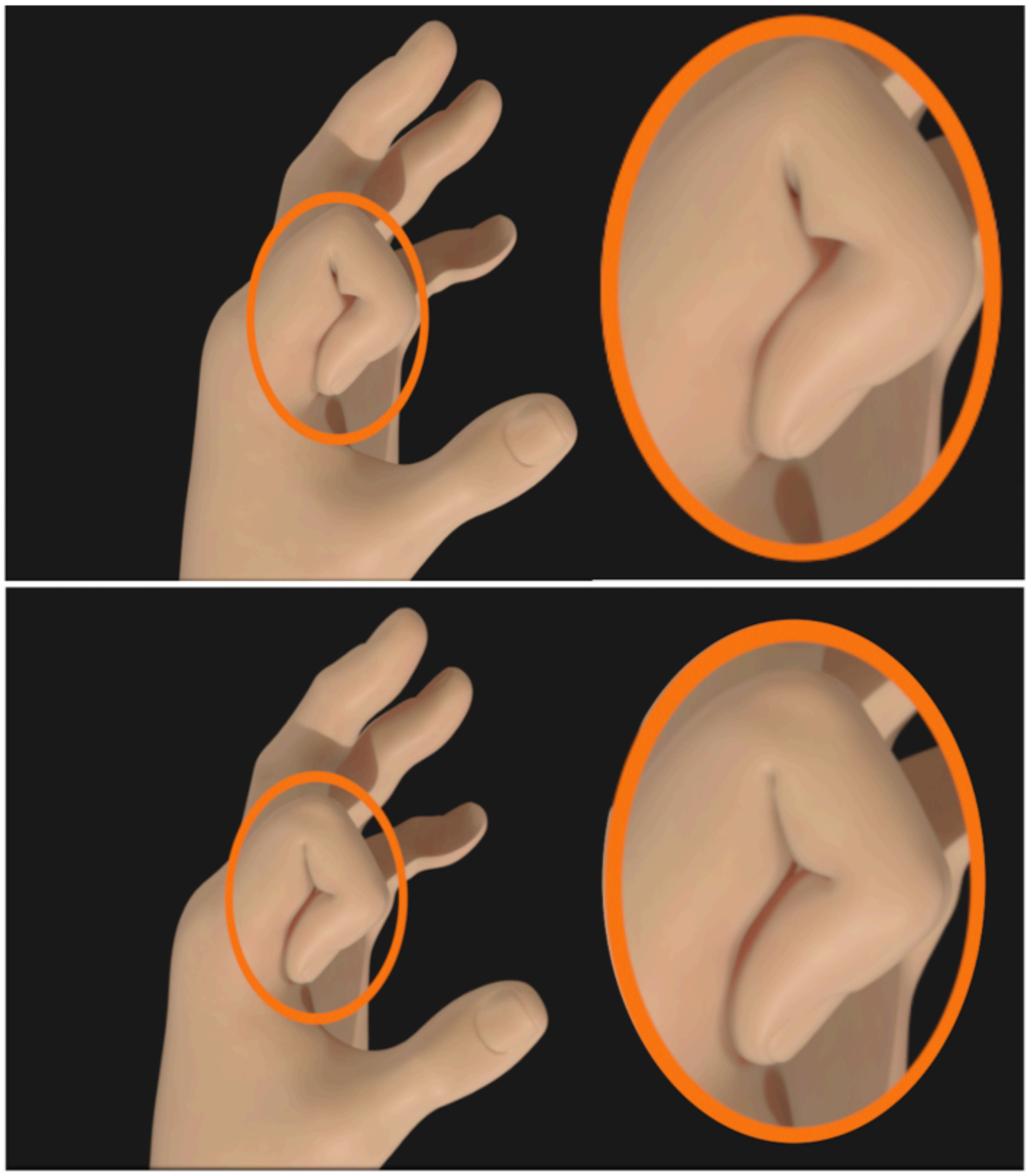


[McAdams et al. 2011]

Character Skinning





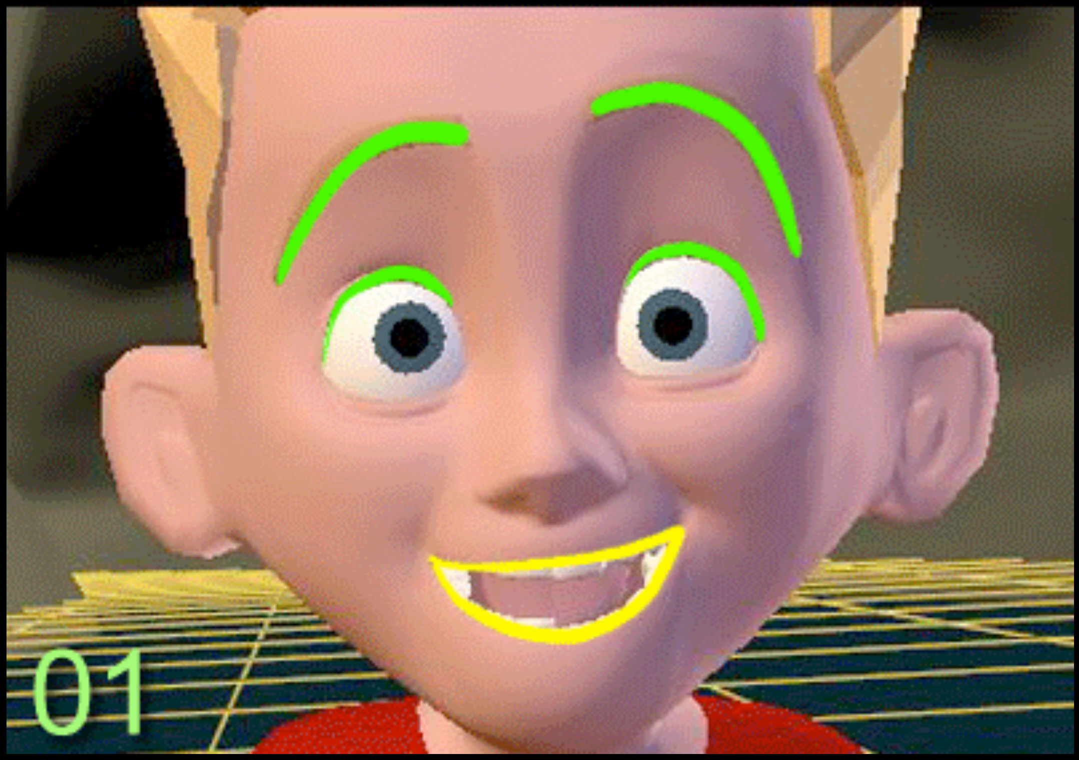


[McAdams et al. 2011]

facial animation

- Skeletons not appropriate
- Use a set of expressive parameters, e.g.,
    - how wide the eyes are
    - eyebrows height and shape
    - mouth shape

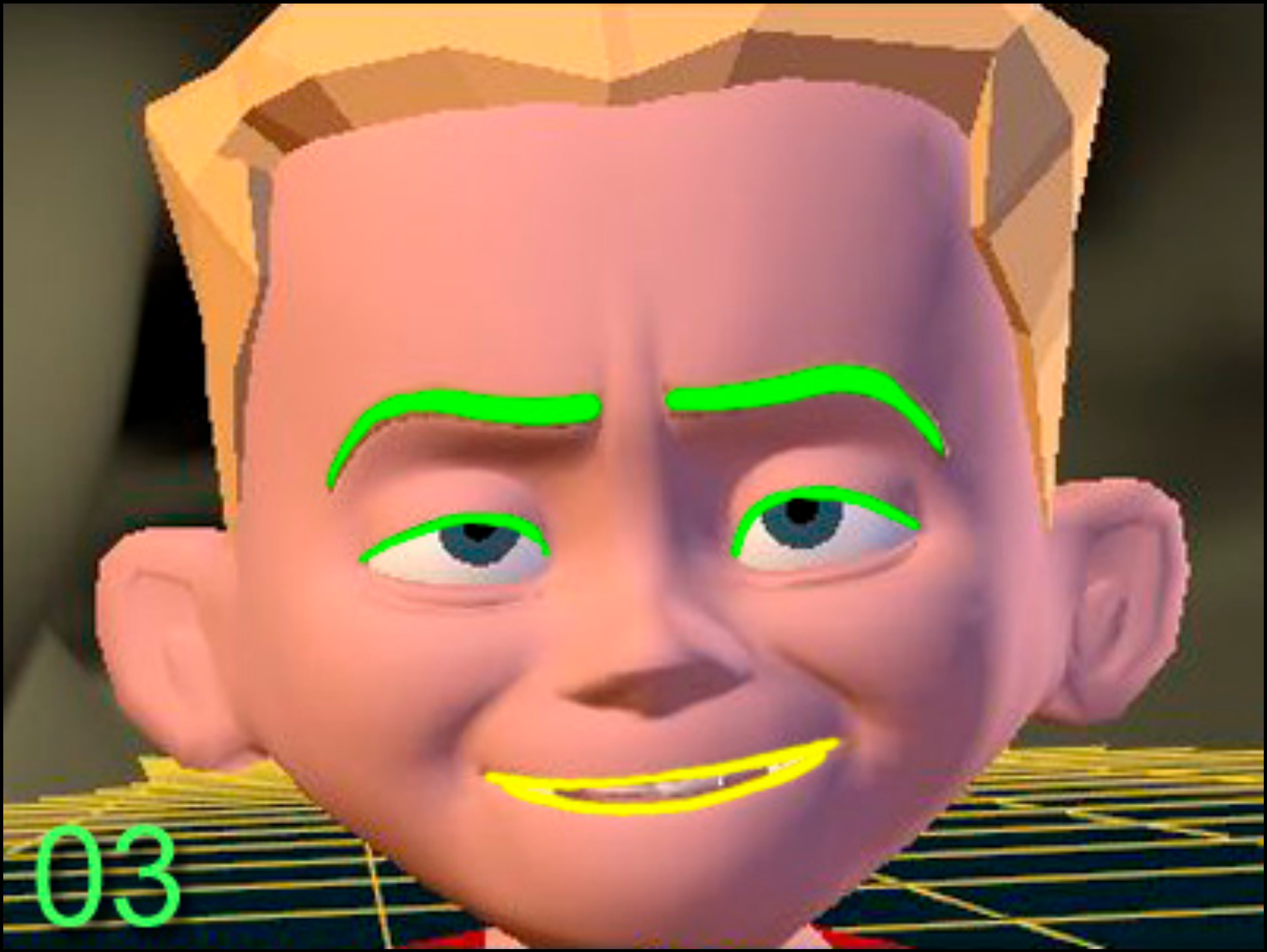
©2004 Disney/Pixar



http://www.navone.org/HTML/Tutorial_DashTake.htm

01

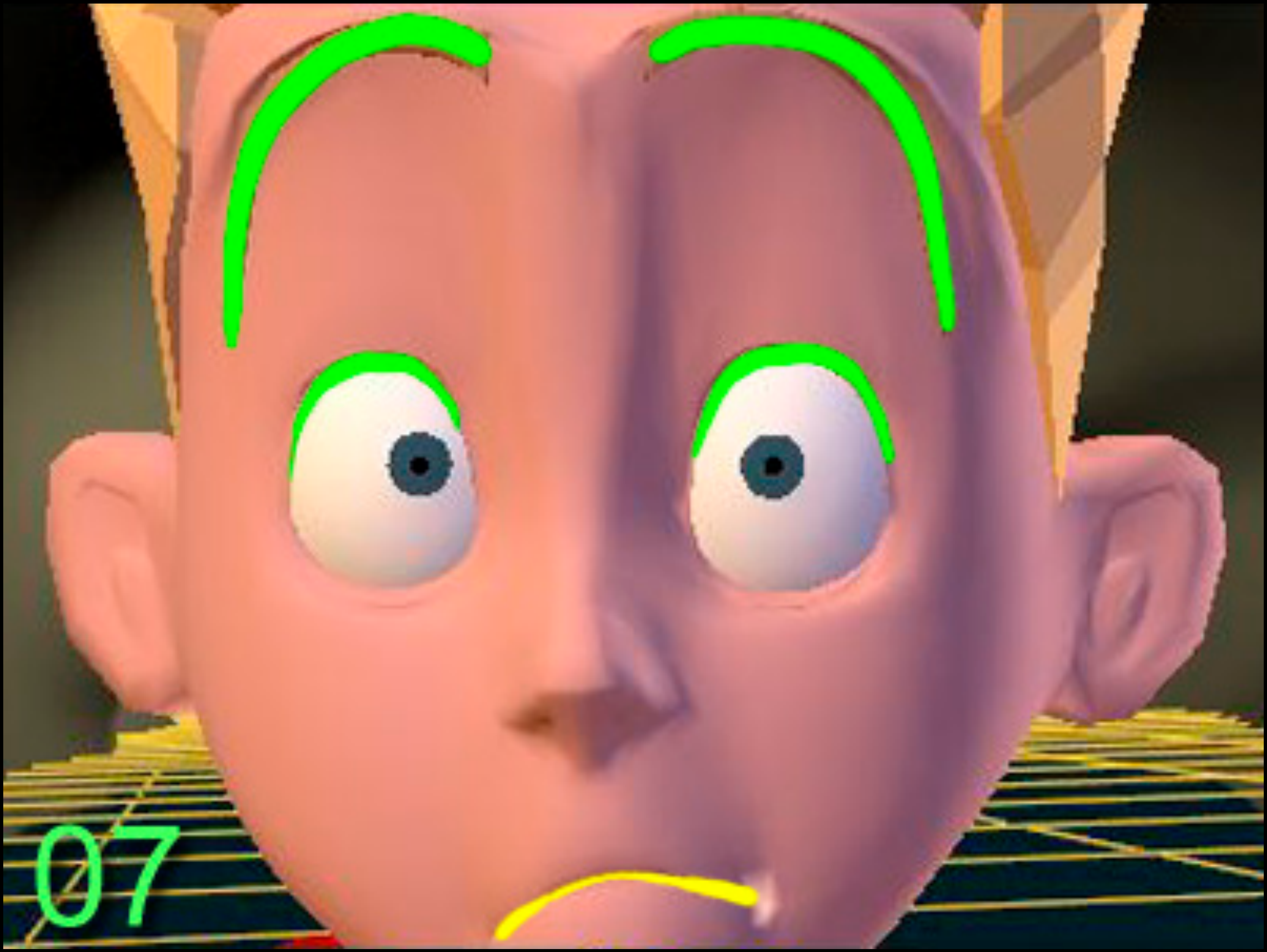
02

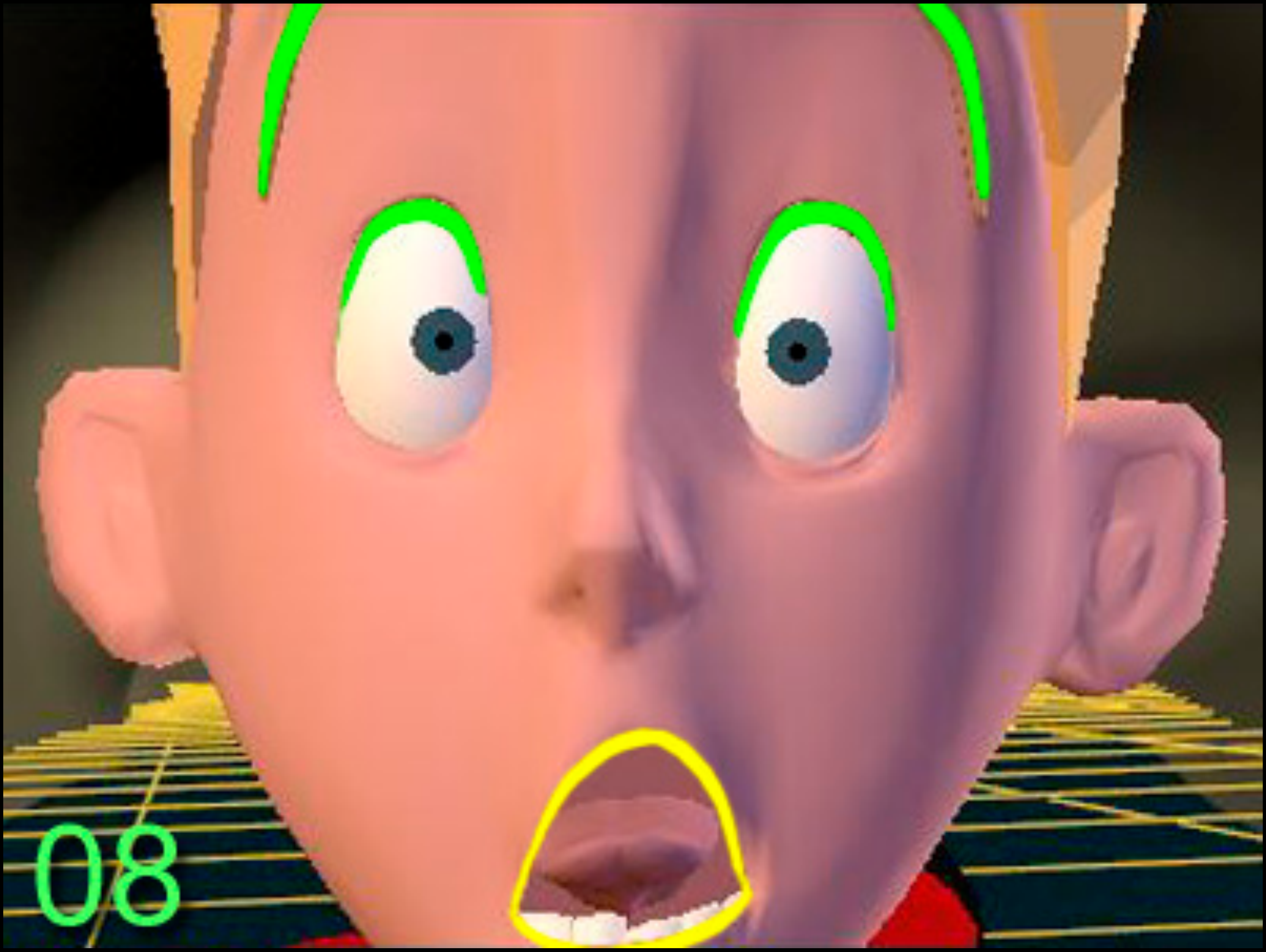

03

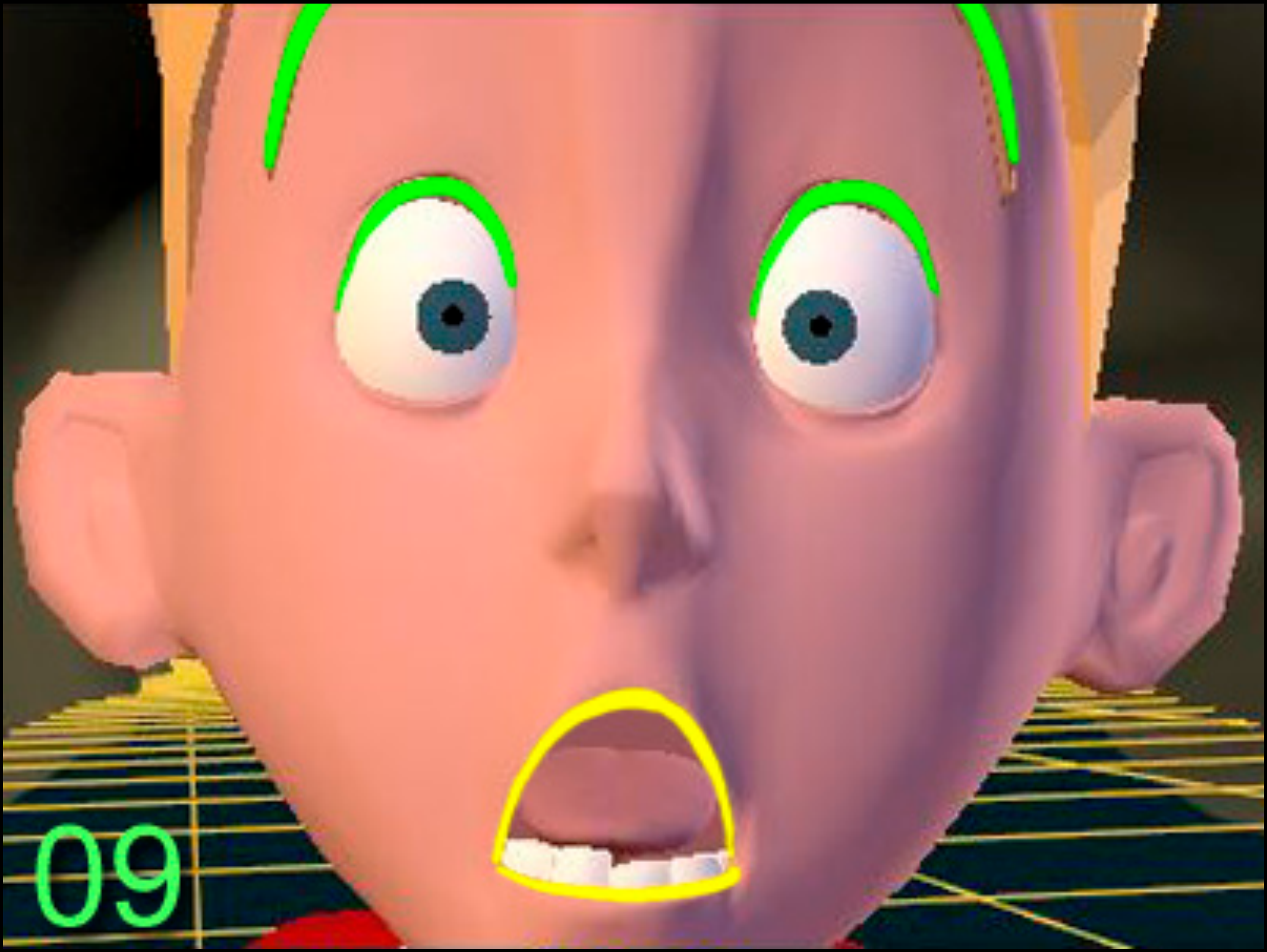
04

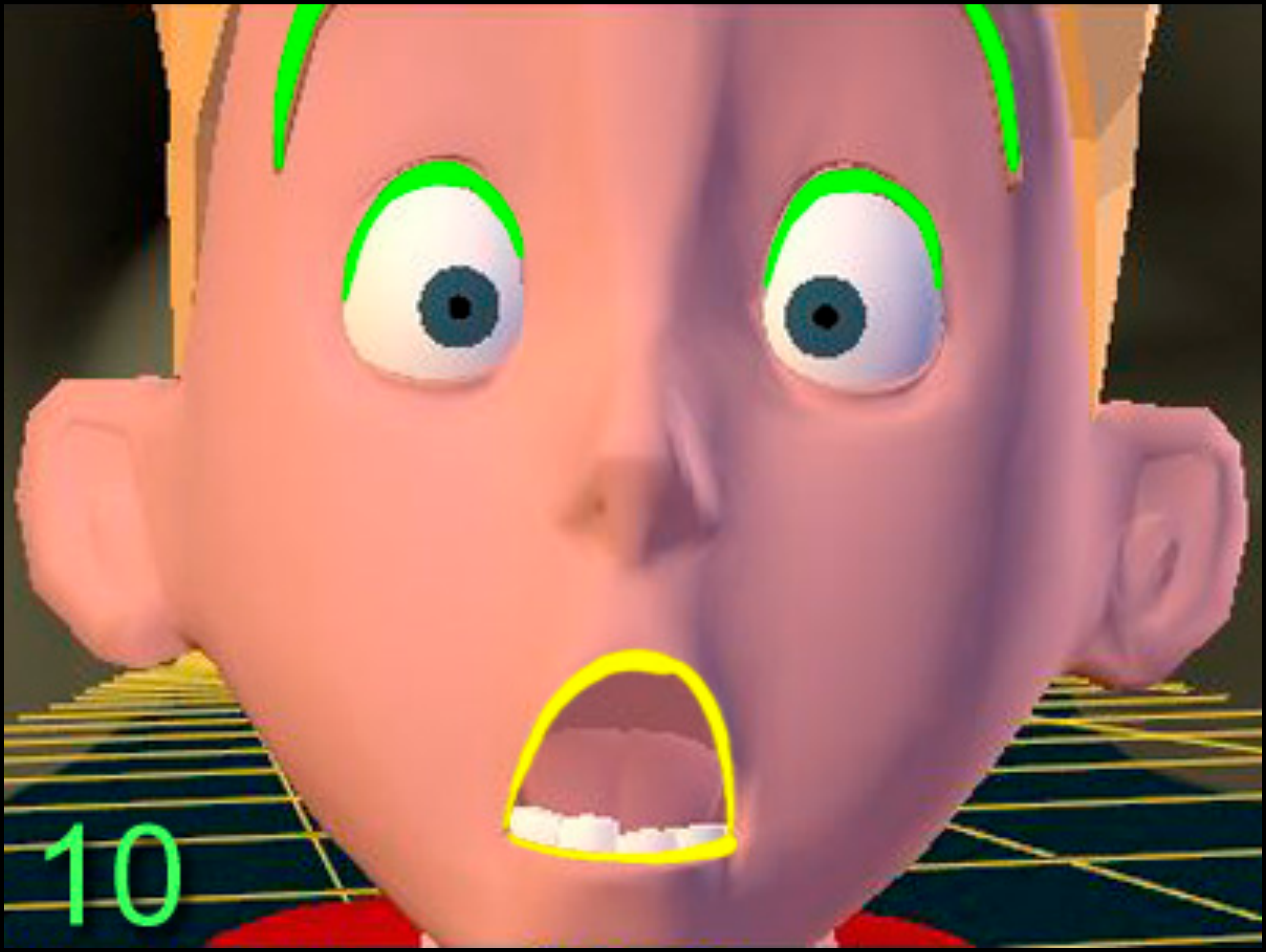
05

06

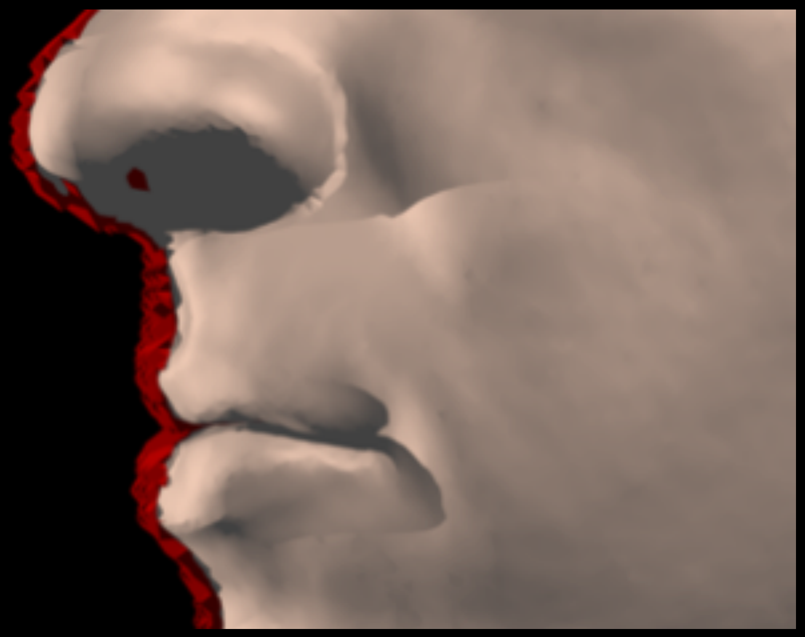

07

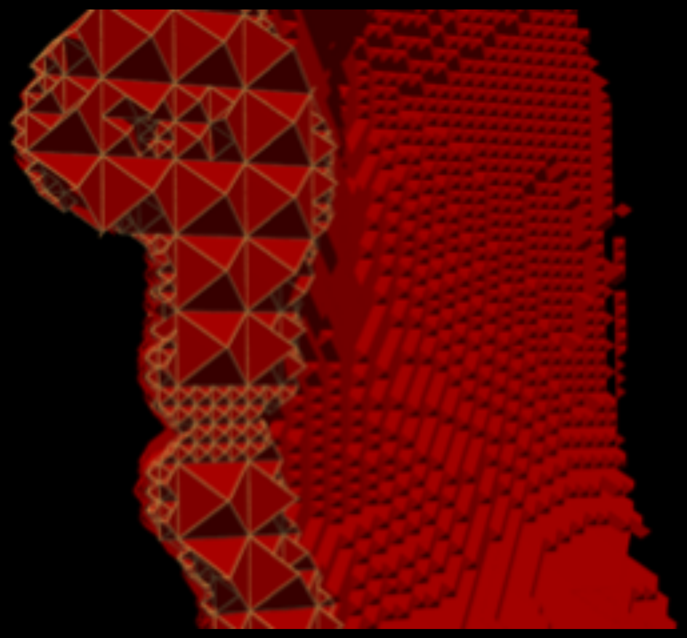

08


09

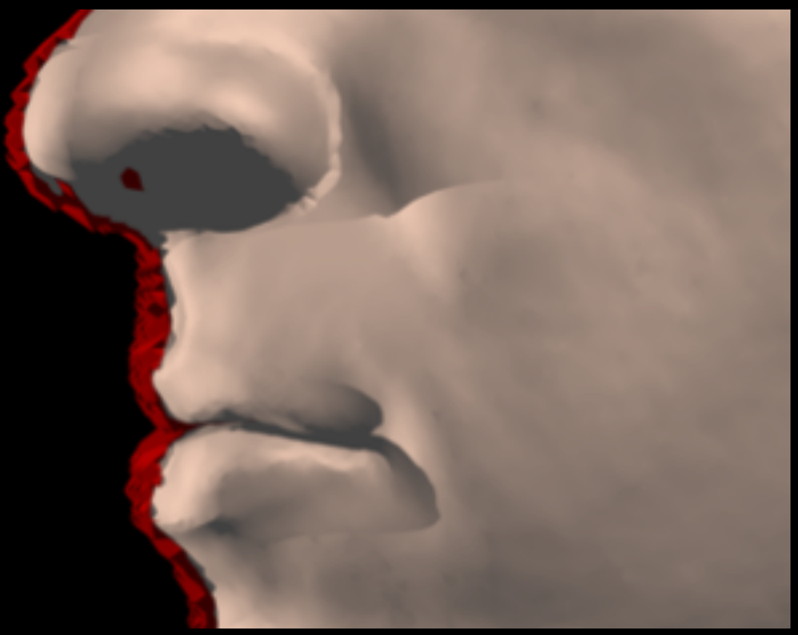

10

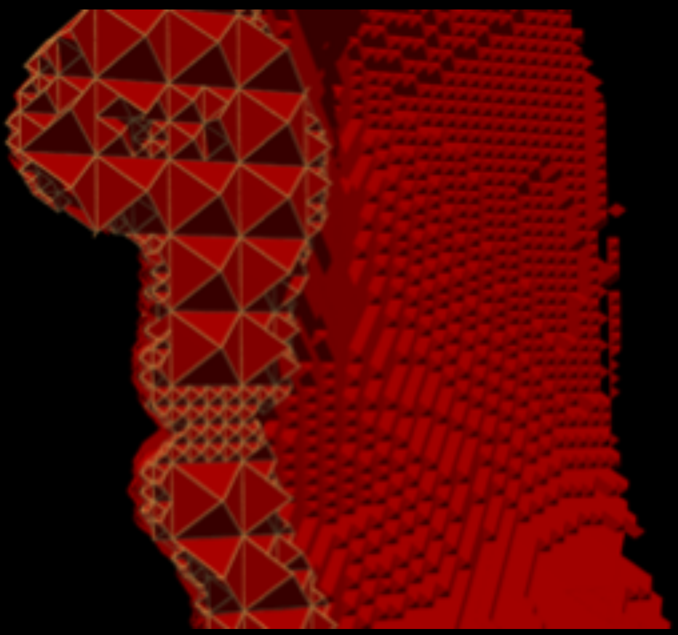
Facial animation





Facial animation

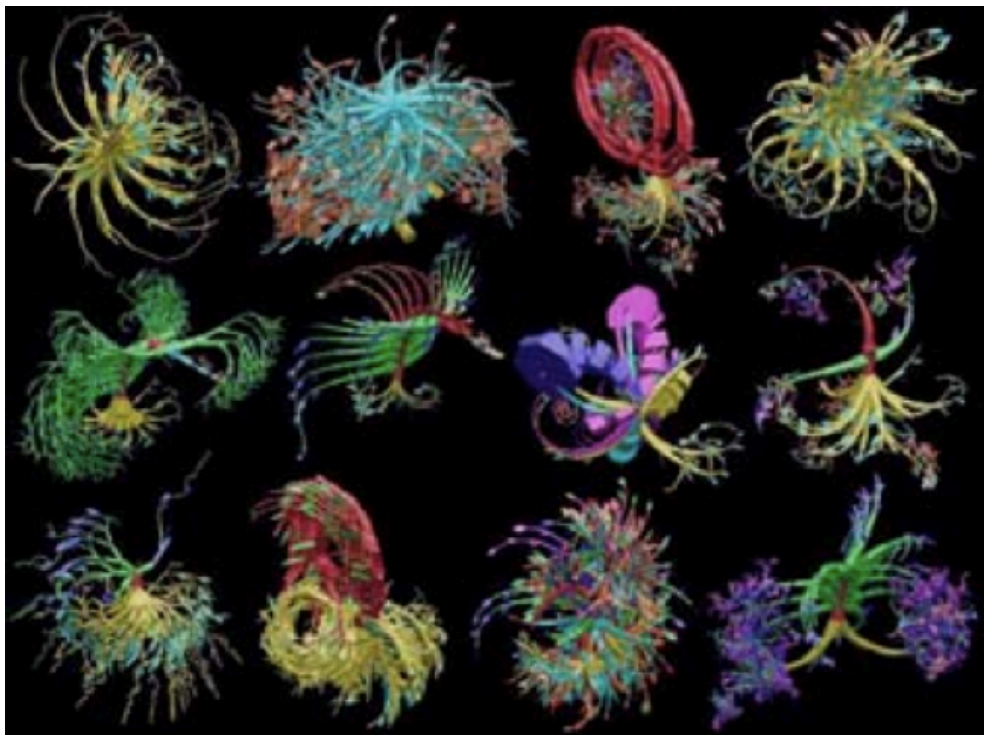




procedural animation

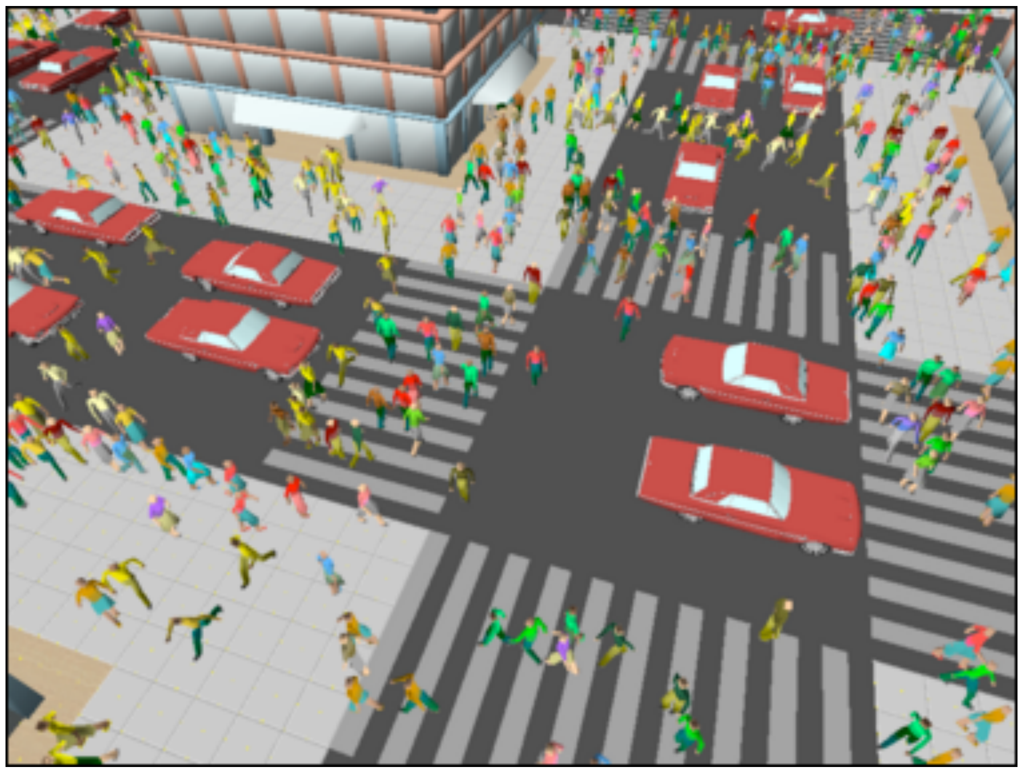
Artificial life

- plants - movement and growth
- evolving artificial life



virtual worlds, special effects, games

Crowd simulation



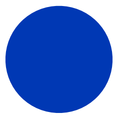
[Treuille et al. 2006]

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

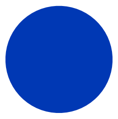
physics-based animation

Particles

Particle: basic dynamic object

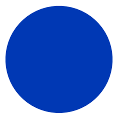


Particle: basic dynamic object



mass m

Particle: basic dynamic object

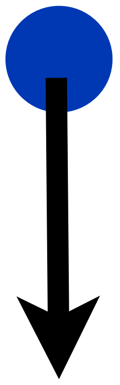


mass m

3 dof

$$\vec{X} = (x, y, z)$$

Particle: basic dynamic object



mass m

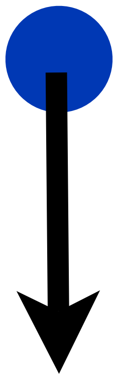
3 dof

$$\vec{X} = (x, y, z)$$

forces: e.g., gravity

$$\vec{F} = -m\vec{g}$$

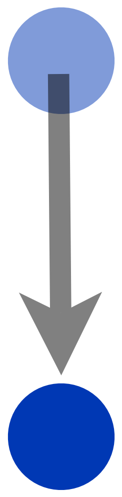
Particle: basic dynamic object



Equations of motion:
Newton's 2nd Law

$$\vec{F} = m\vec{a}$$

Particle: basic dynamic object



Equations of motion:
Newton's 2nd Law

$$\vec{F} = m\vec{a}$$

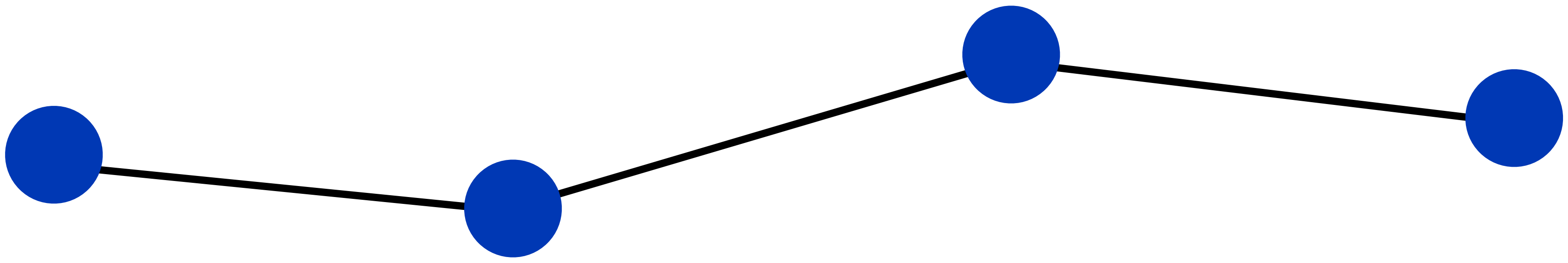
$$\frac{d\vec{x}}{dt} = \vec{v}$$

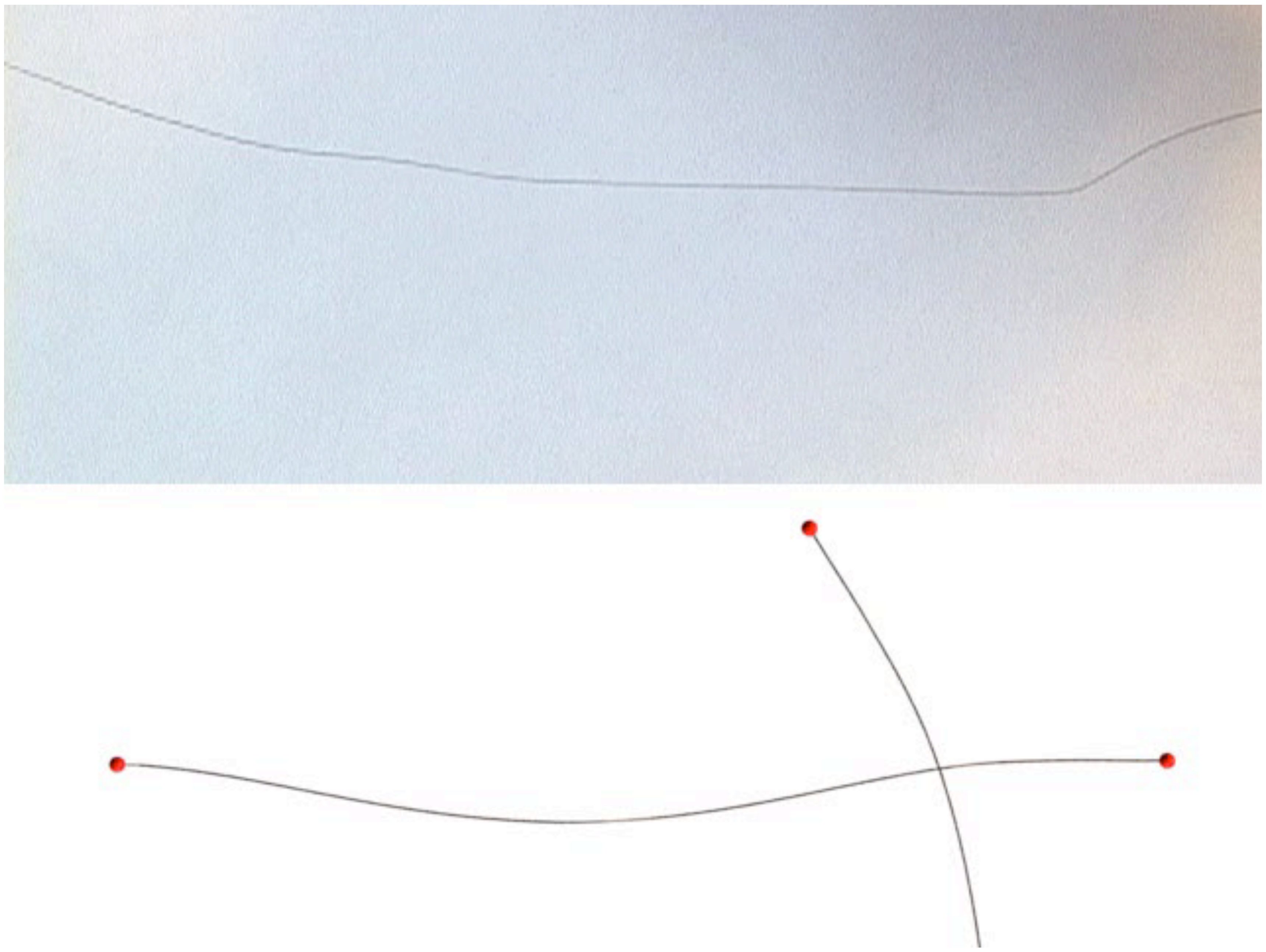
$$m\frac{d\vec{v}}{dt} = \vec{F}$$

System of ODEs

Deformable bodies

Connect a bunch of particles into a 1D line segment with springs

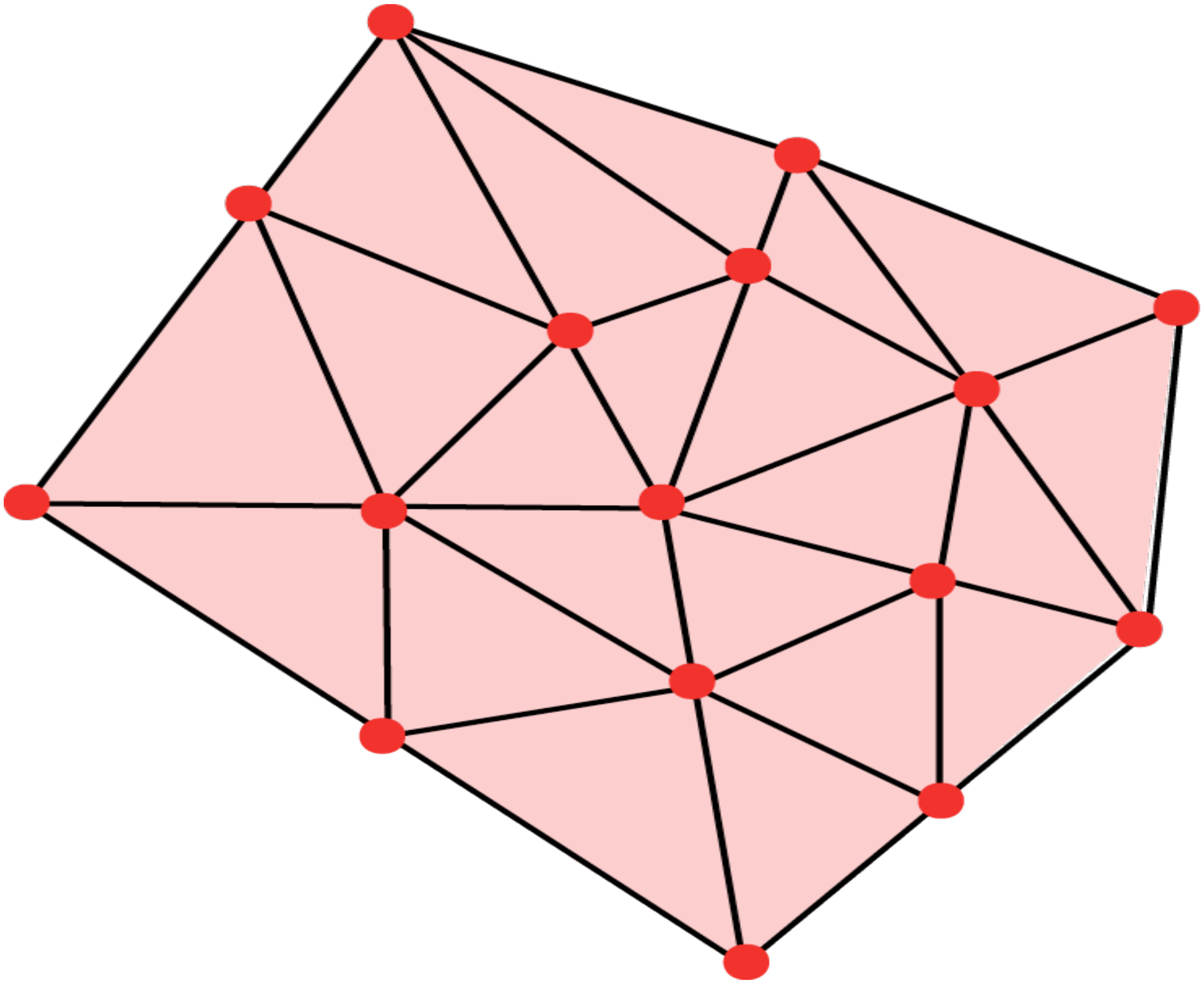




A Mass Spring Model for Hair Simulation

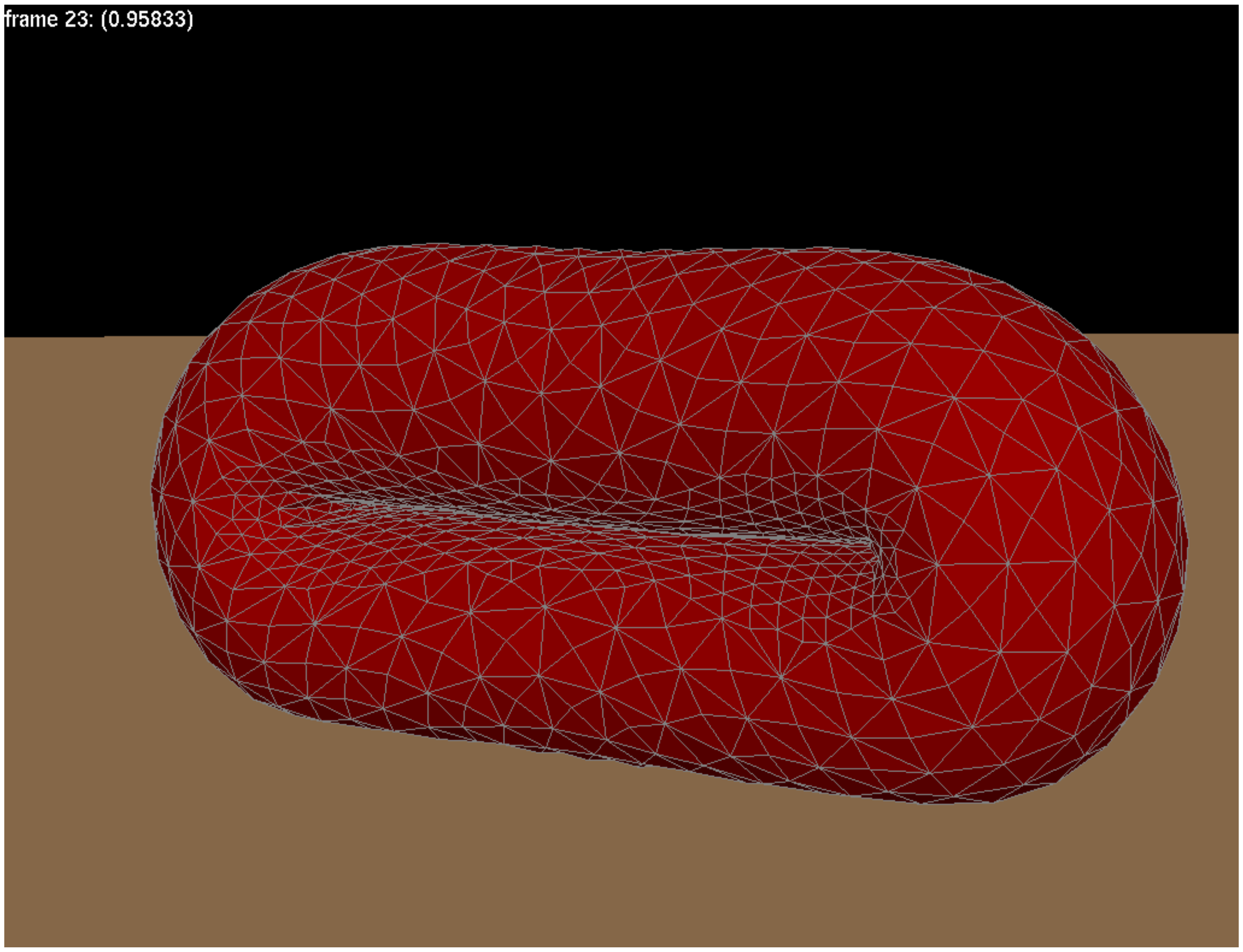
Selle, A., Lentine, M., G., and Fedkiw, R. ACM Transactions on Graphics SIGGRAPH 2008, ACM TOG 27, 64.1-64.11 (2008)

Connect a bunch of particles into a 2D mesh

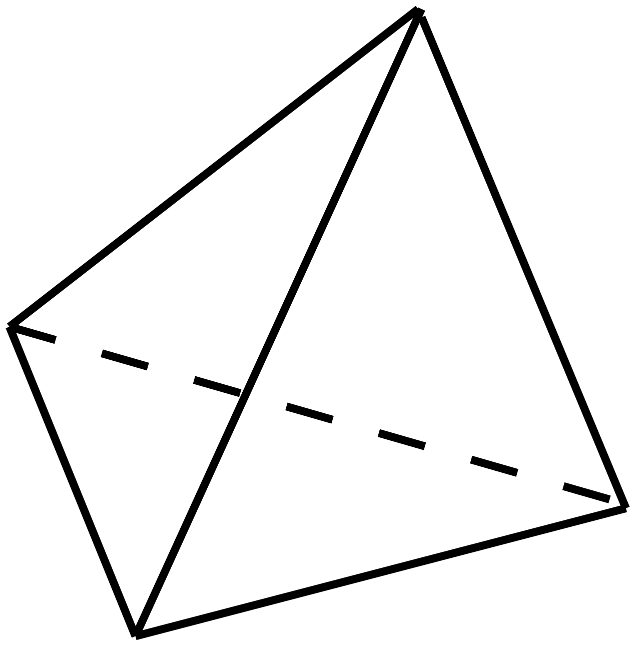


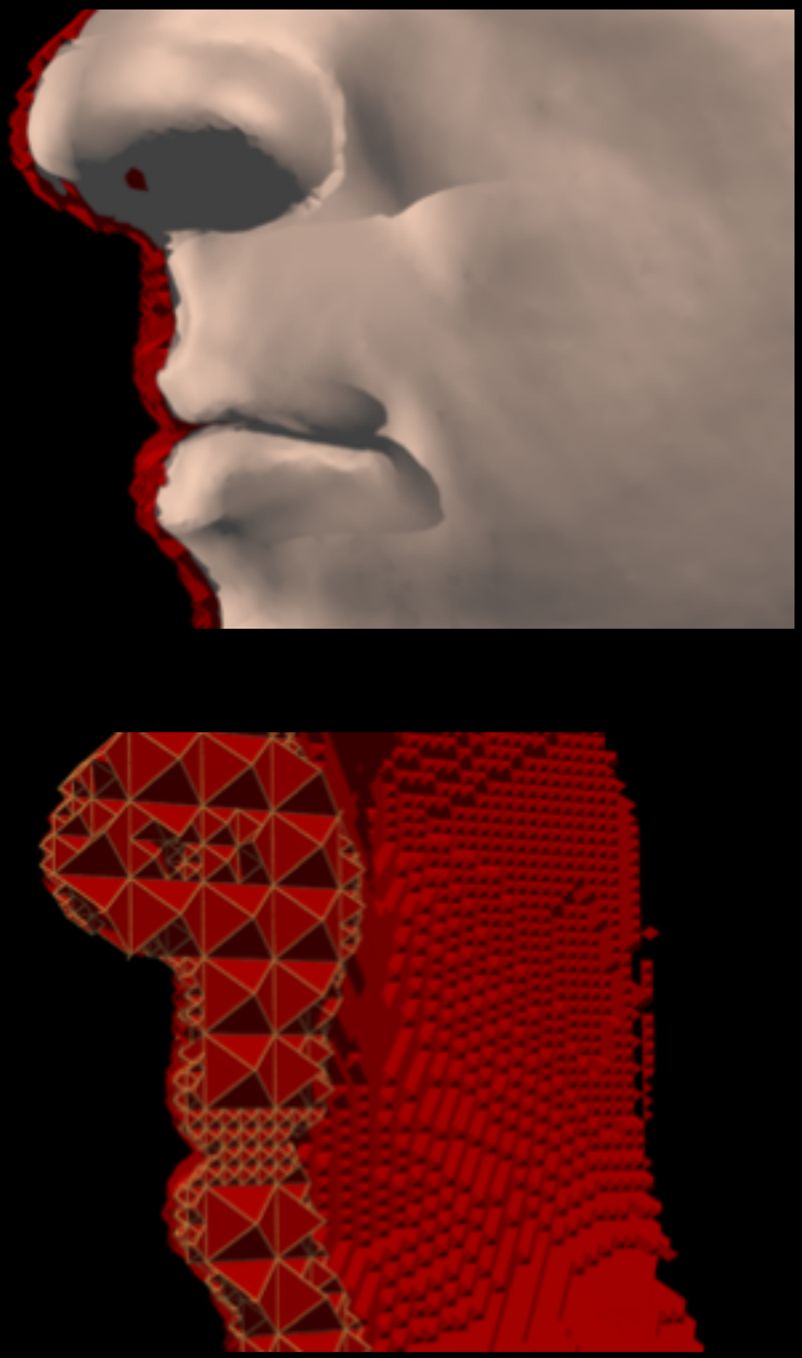
Selle, A., Su, J., Irving, G., and Fedkiw, R. IEEE Transactions on Visualization and Graphics (TVCG) 15(2) 339-350

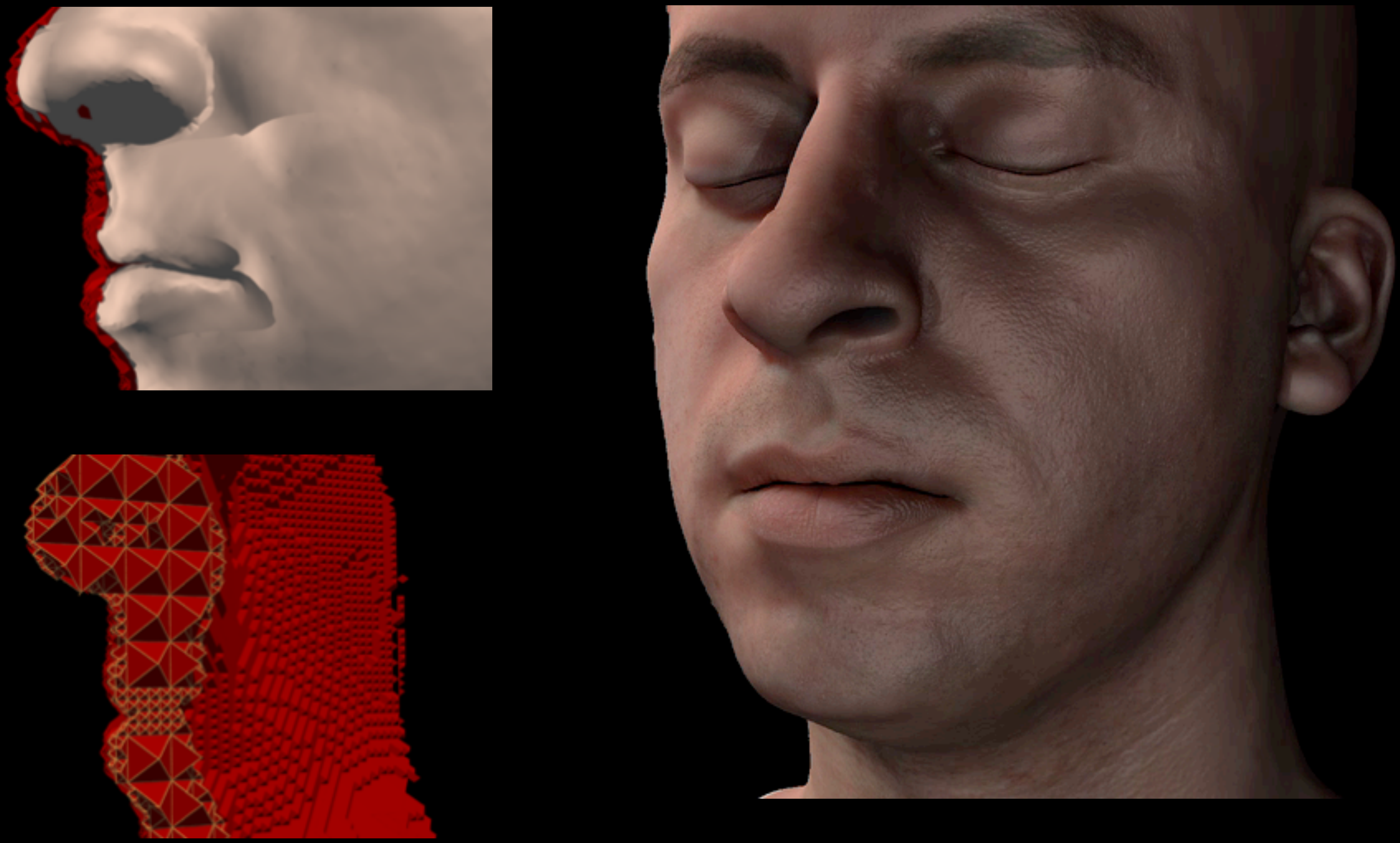
Connect a bunch of particles into a 3D mesh



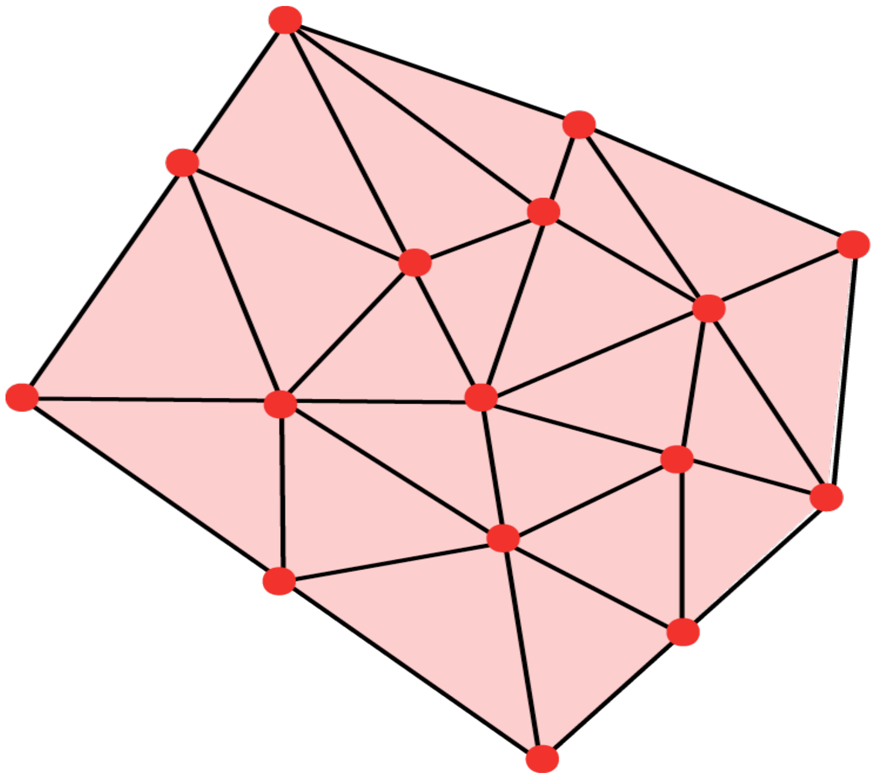
tetrahedron







Deformable bodies: equations of motion



Equations of motion:
Newton’s 2nd Law

$$\vec{F} = m\vec{a}$$

$$\frac{d\vec{x}}{dt} = \vec{v}$$

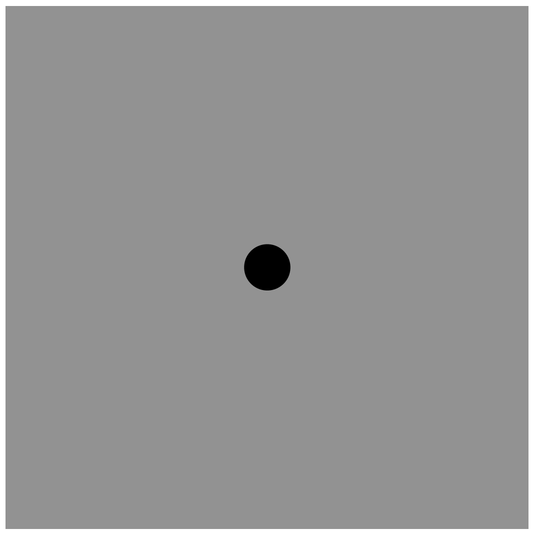
$$m\frac{d\vec{v}}{dt} = \underset{\uparrow}{\vec{F}}$$

contains spatial derivatives

System of PDEs

Rigid bodies

Rigid bodies



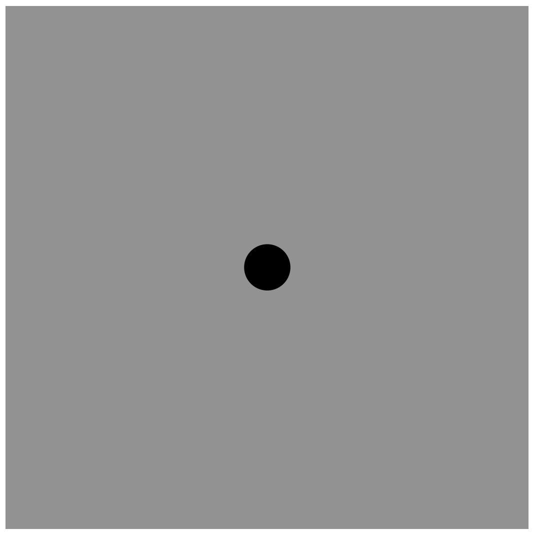
$$(\vec{X}, \vec{\Omega})$$

$$(\vec{F}, \vec{\tau})$$

- •Limit of infinite stiffness – rigid approximations encapsulates the constitutive model
  - •Body now has 6 degrees of freedom – 3 for position and 3 for orientation
- •Forces acting on the body result both in a net force and net torque on the body
- •Must handle elastic collisions – by defining a coefficient of restitution
  - •Deformable objects undergo inelastic collisions and store energy at the collisions which then causes the bounce
- •Resulting evolution equations are a system of ODEs
- •Typically integrated with explicit methods
  - •No time step restriction for stability since the rigid approximation means information through the body is propagated instantaneously
  - •i.e., we don’t need to account for finite propagation of information through a mesh in computing the time step

Rigid bodies

6 dofs



$$(\vec{X}, \vec{\Omega})$$

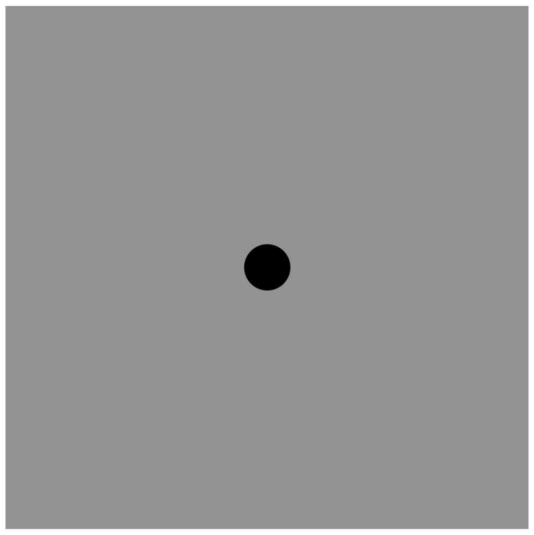
$$(\vec{F}, \vec{\tau})$$

- Limit of infinite stiffness – rigid approximations encapsulates the constitutive model
    - Body now has 6 degrees of freedom – 3 for position and 3 for orientation
- Forces acting on the body result both in a net force and net torque on the body
- Must handle elastic collisions – by defining a coefficient of restitution
    - Deformable objects undergo inelastic collisions and store energy at the collisions which then causes the bounce
- Resulting evolution equations are a system of ODEs
- Typically integrated with explicit methods
    - No time step restriction for stability since the rigid approximation means information through the body is propagated instantaneously
    - i.e., we don't need to account for finite propagation of information through a mesh in computing the time step

Rigid bodies

6 dofs

forces and torques



$$(\vec{X}, \vec{\Omega})$$

$$(\vec{F}, \vec{\tau})$$

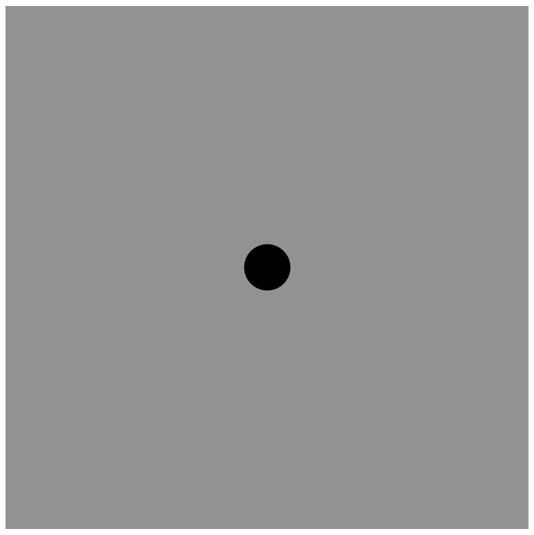
- •Limit of infinite stiffness – rigid approximations encapsulates the constitutive model
  - •Body now has 6 degrees of freedom – 3 for position and 3 for orientation
- •Forces acting on the body result both in a net force and net torque on the body
- •Must handle elastic collisions – by defining a coefficient of restitution
  - •Deformable objects undergo inelastic collisions and store energy at the collisions which then causes the bounce
- •Resulting evolution equations are a system of ODEs
- •Typically integrated with explicit methods
  - •No time step restriction for stability since the rigid approximation means information through the body is propagated instantaneously
  - •i.e., we don’t need to account for finite propagation of information through a mesh in computing the time step

Rigid bodies

6 dofs

forces and torques

elastic collisions



$(\vec{X}, \vec{\Omega})$

$(\vec{F}, \vec{\tau})$

•Limit of infinite stiffness – rigid approximations encapsulates the constitutive model

 •Body now has 6 degrees of freedom – 3 for position and 3 for orientation

•Forces acting on the body result both in a net force and net torque on the body

•Must handle elastic collisions – by defining a coefficient of restitution

 •Deformable objects undergo inelastic collisions and store energy at the collisions which then causes the bounce

•Resulting evolution equations are a system of ODEs

•Typically integrated with explicit methods

 •No time step restriction for stability since the rigid approximation means information through the body is propagated instantaneously

 •i.e., we don’t need to account for finite propagation of information through a mesh in computing the time step

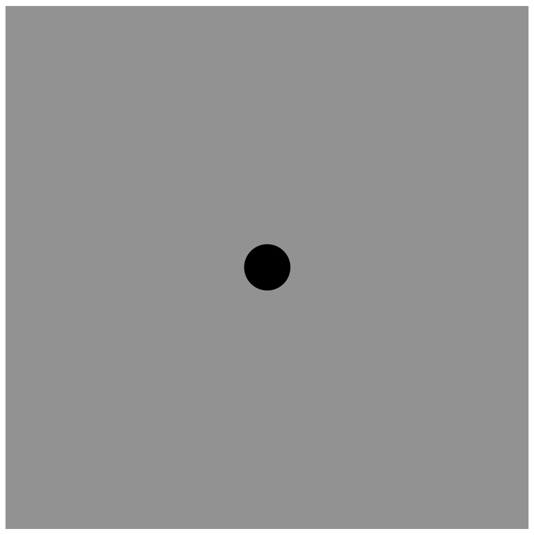
Rigid bodies

6 dofs

forces and torques

elastic collisions

ODEs



$$(\vec{X}, \vec{\Omega})$$

$$(\vec{F}, \vec{\tau})$$

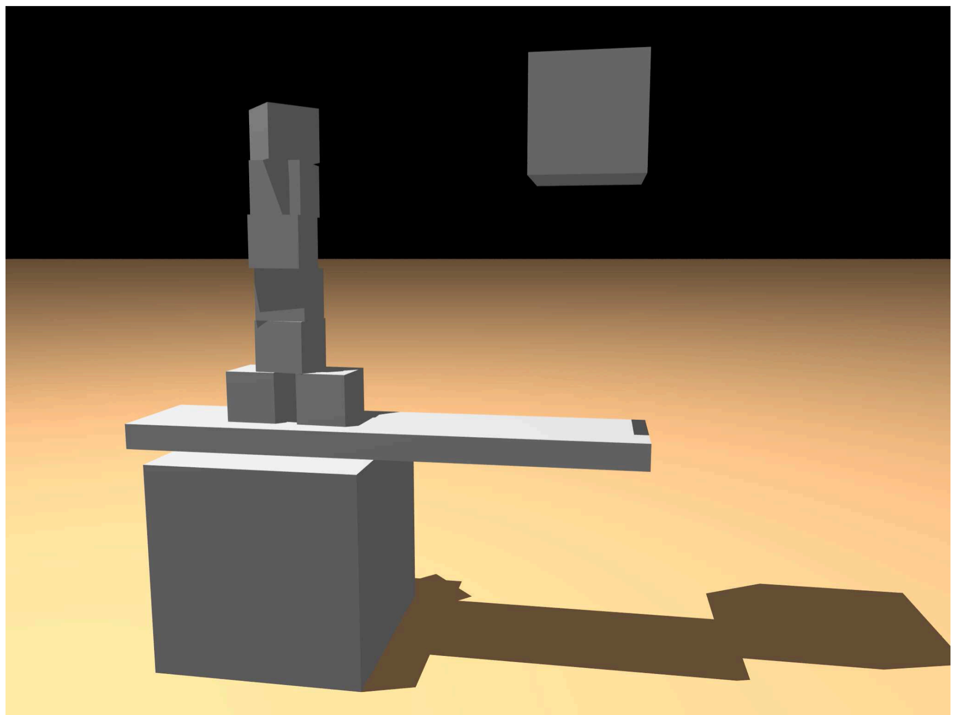
- Limit of infinite stiffness – rigid approximations encapsulates the constitutive model
    - Body now has 6 degrees of freedom – 3 for position and 3 for orientation
- Forces acting on the body result both in a net force and net torque on the body
- Must handle elastic collisions – by defining a coefficient of restitution
    - Deformable objects undergo inelastic collisions and store energy at the collisions which then causes the bounce
- Resulting evolution equations are a system of ODEs
- Typically integrated with explicit methods
    - No time step restriction for stability since the rigid approximation means information through the body is propagated instantaneously
    - i.e., we don’t need to account for finite propagation of information through a mesh in computing the time step

Rigid body phenomena

- One of the main challenges in rigid body simulation is collisions vs. contact
  - Special care to make sure that things in sliding contact don’t bounce
  - But that things get a chance to bounce before sticking
- Getting a stable stack is also hard
  - Shock propagation
- Friction between rigid bodies is harder than deformable
  - Actually takes place over some finite time step dt and along nonlinear path, i.e., the body will rotate due to friction
  - Should actually be an integral – first order approximation can cause weirdness
- Articulation – very useful for creatures!
  - velocity constraints plus drift correction
  - We do pd control similarly
- Also fracture

Rigid body phenomena

stacking

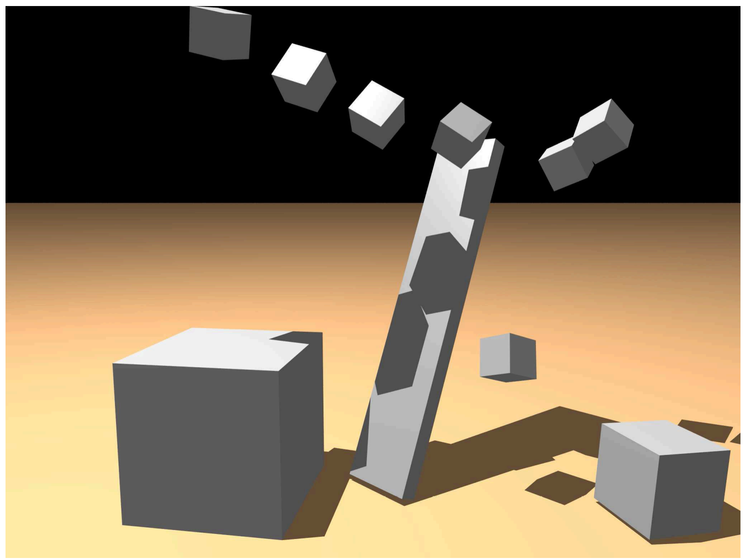


- •One of the main challenges in rigid body simulation is collisions vs. contact
    - •Special care to make sure that things in sliding contact don’t bounce
    - •But that things get a chance to bounce before sticking
- •Getting a stable stack is also hard
    - •Shock propagation
- •Friction between rigid bodies is harder than deformable
    - •Actually takes place over some finite time step dt and along nonlinear path, i.e., the body will rotate due to friction
    - •Should actually be an integral – first order approximation can cause weirdness
- •Articulation – very useful for creatures!
    - •velocity constraints plus drift correction
    - •We do pd control similarly
- •Also fracture

Rigid body phenomena

stacking

collisions, contact



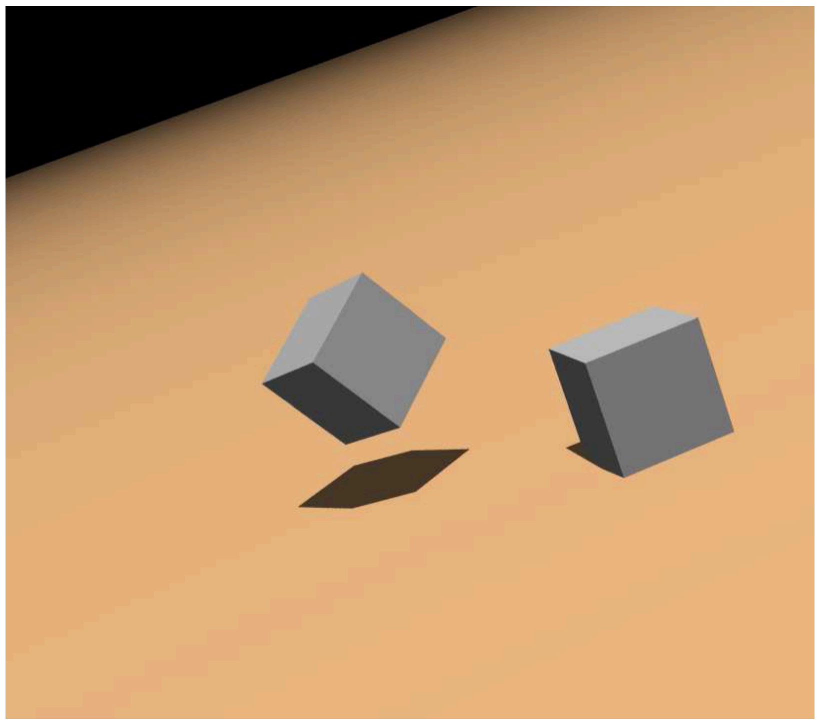
- One of the main challenges in rigid body simulation is collisions vs. contact
    - Special care to make sure that things in sliding contact don’t bounce
    - But that things get a chance to bounce before sticking
- Getting a stable stack is also hard
    - Shock propagation
- Friction between rigid bodies is harder than deformable
    - Actually takes place over some finite time step dt and along nonlinear path, i.e., the body will rotate due to friction
    - Should actually be an integral – first order approximation can cause weirdness
- Articulation – very useful for creatures!
    - velocity constraints plus drift correction
    - We do pd control similarly
- Also fracture

Rigid body phenomena

stacking

collisions, contact

friction



- One of the main challenges in rigid body simulation is collisions vs. contact
  - Special care to make sure that things in sliding contact don't bounce
  - But that things get a chance to bounce before sticking
- Getting a stable stack is also hard
  - Shock propagation
- Friction between rigid bodies is harder than deformable
  - Actually takes place over some finite time step dt and along nonlinear path, i.e., the body will rotate due to friction
  - Should actually be an integral – first order approximation can cause weirdness
- Articulation – very useful for creatures!
  - velocity constraints plus drift correction
  - We do pd control similarly
- Also fracture

Rigid body phenomena

stacking

collisions, contact

friction

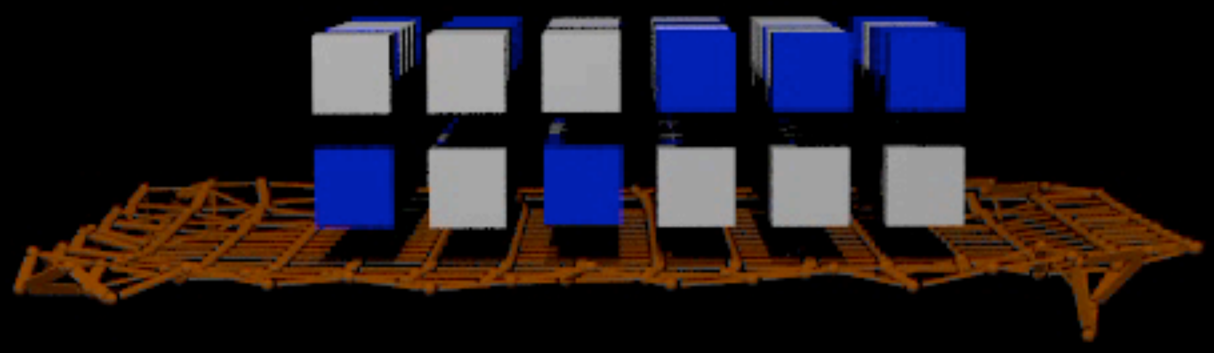
articulation, control

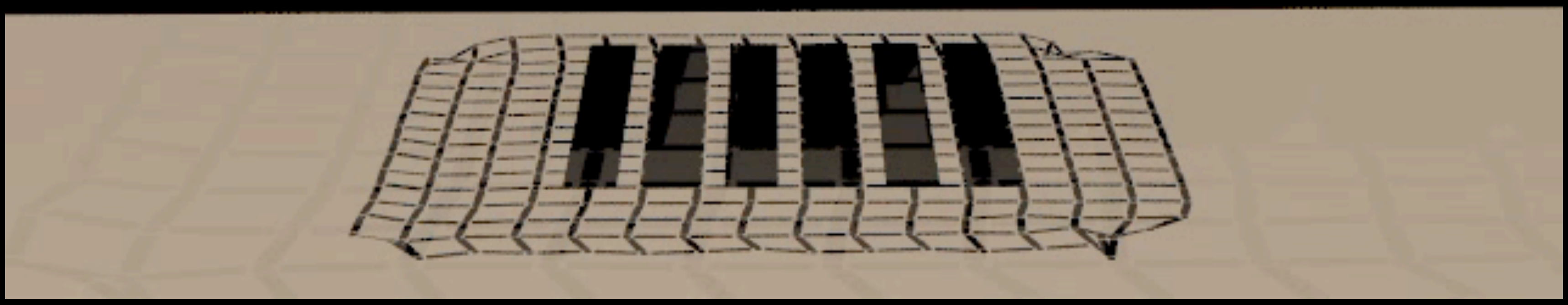
- •One of the main challenges in rigid body simulation is collisions vs. contact
  - •Special care to make sure that things in sliding contact don’t bounce
  - •But that things get a chance to bounce before sticking
- •Getting a stable stack is also hard
  - •Shock propagation
- •Friction between rigid bodies is harder than deformable
  - •Actually takes place over some finite time step dt and along nonlinear path, i.e., the body will rotate due to friction
  - •Should actually be an integral – first order approximation can cause weirdness
- •Articulation – very useful for creatures!
  - •velocity constraints plus drift correction
  - •We do pd control similarly
- •Also fracture

Articulated rigid bodies

Rachel Weinstein, Joey Teran and Ron Fedkiw

Articulated rigid bodies





Rachel Weinstein, Joey Teran and Ron Fedkiw

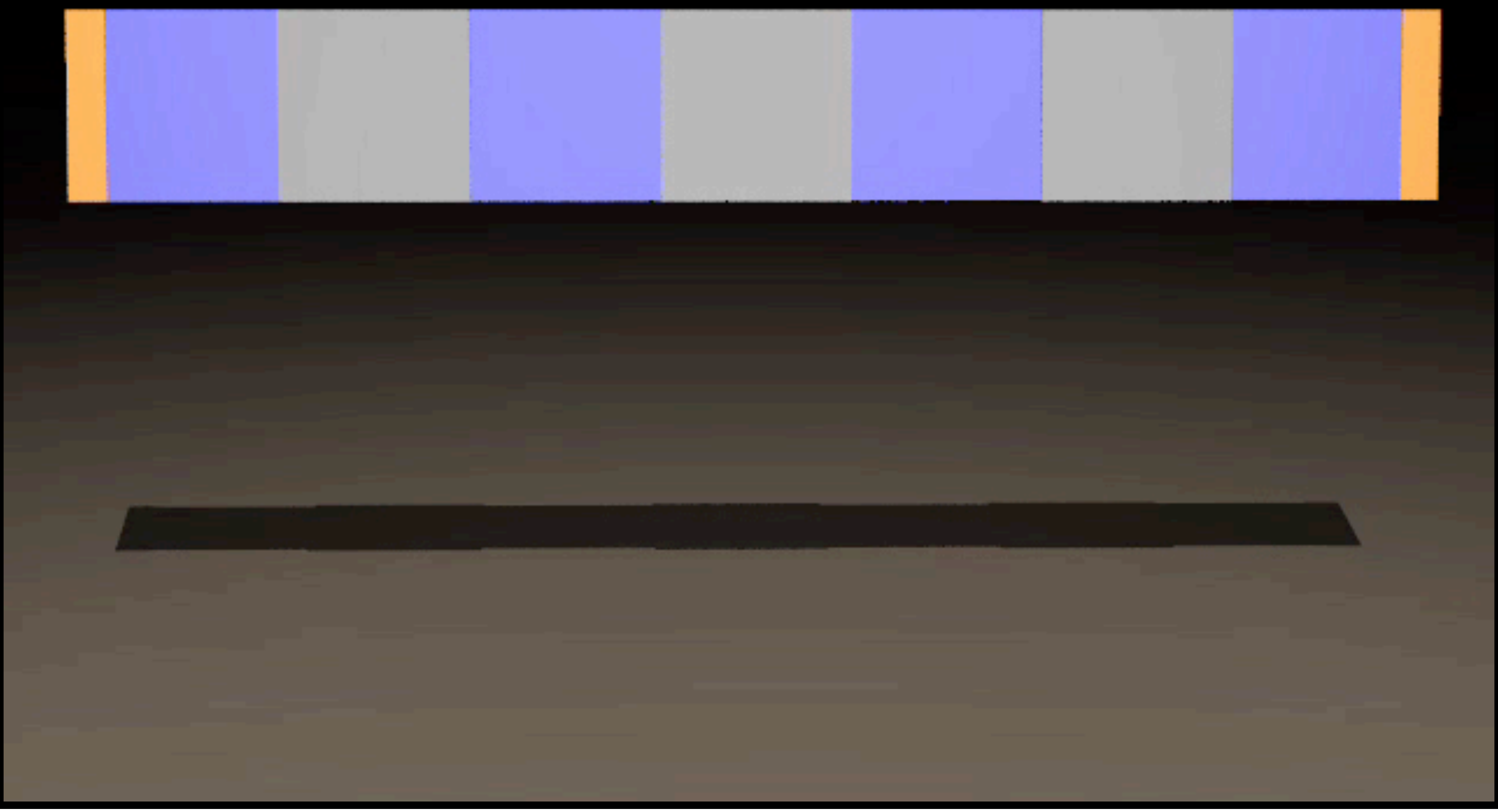
Rigid body simulation

[Weinstein et al 2006]

Rigid and deformable solids coupled together...

* Rigid body examples,
etc.

Rigid and deformable solids coupled together...

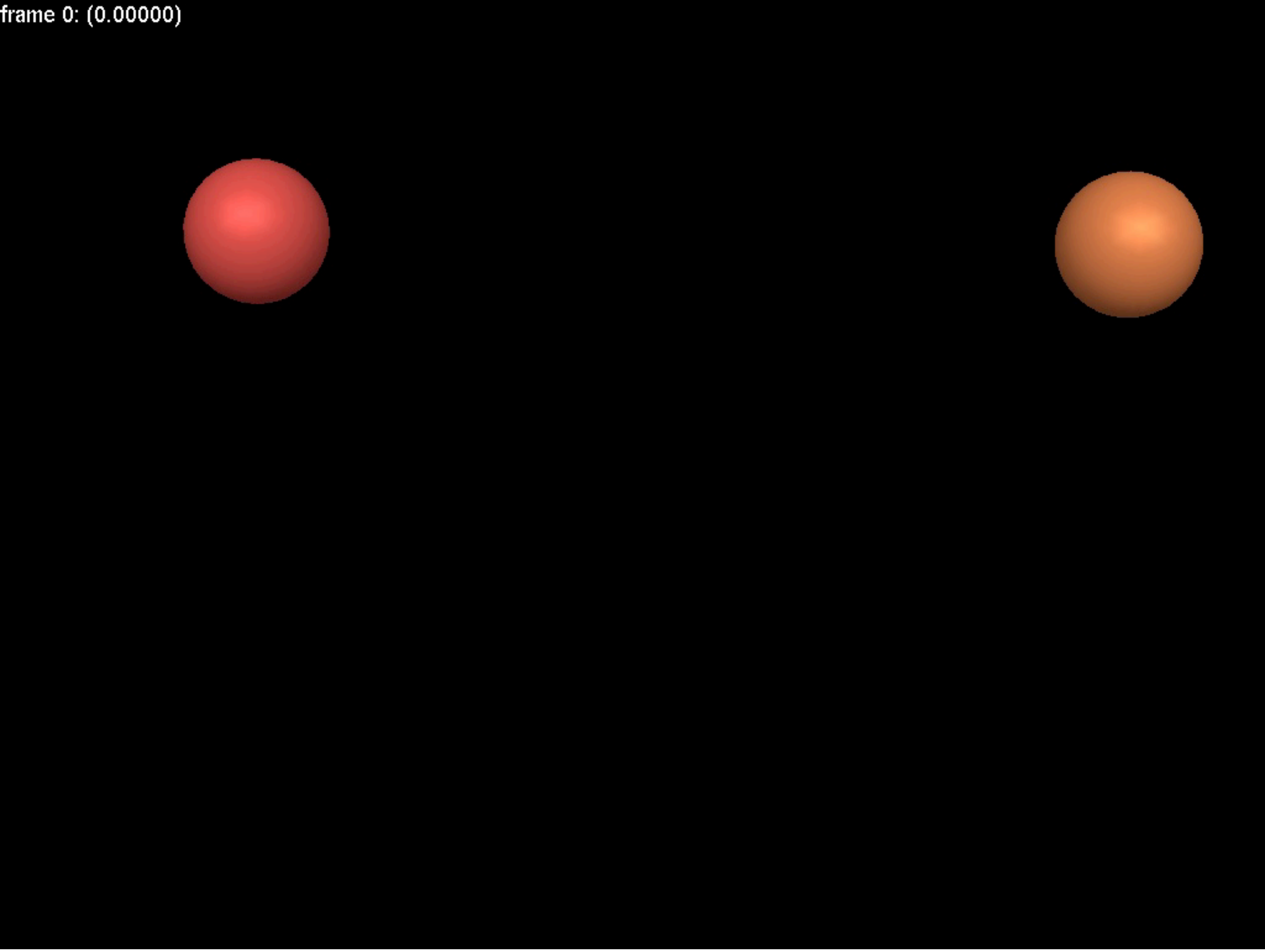


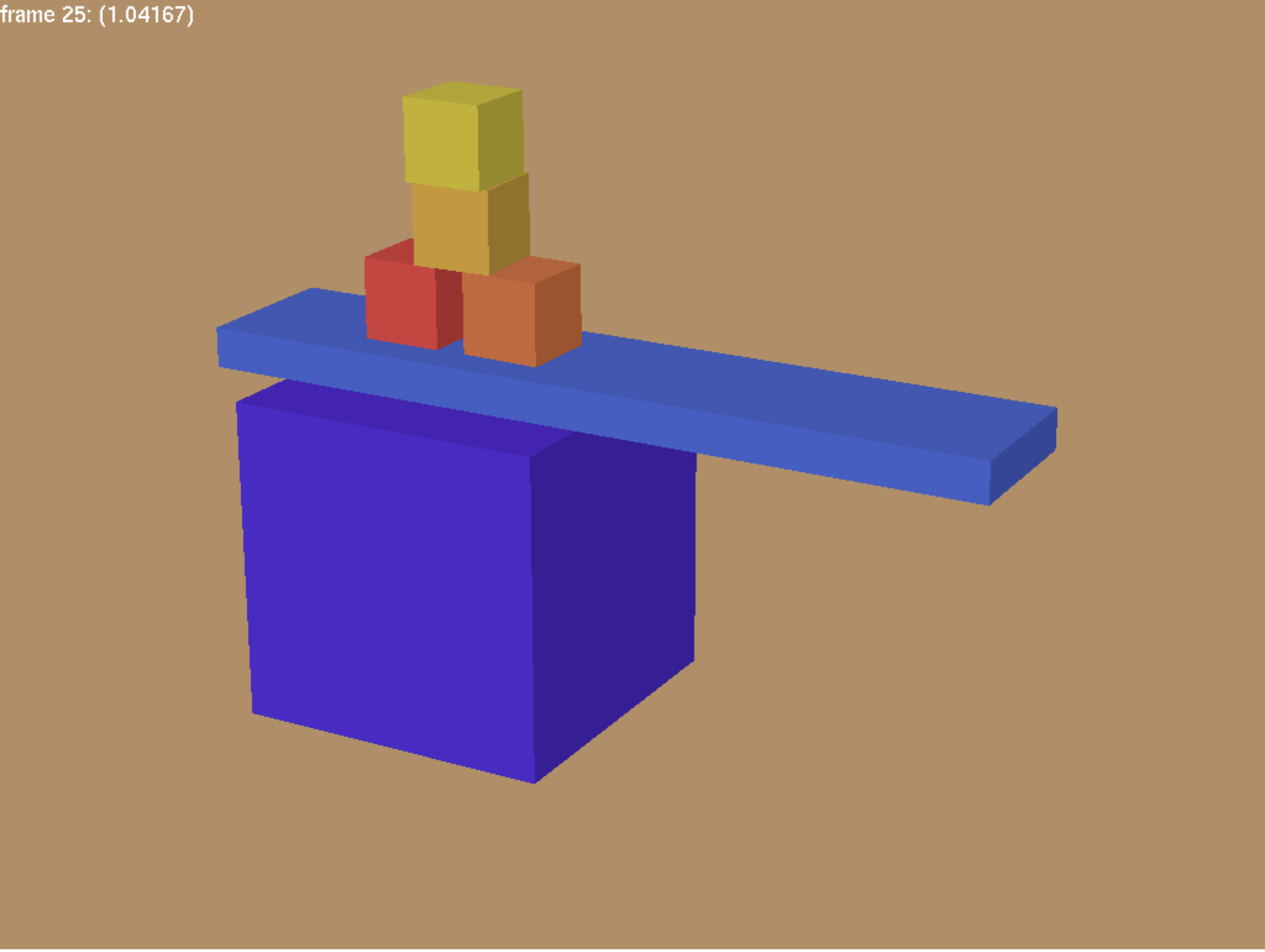
* Rigid body examples,
etc.

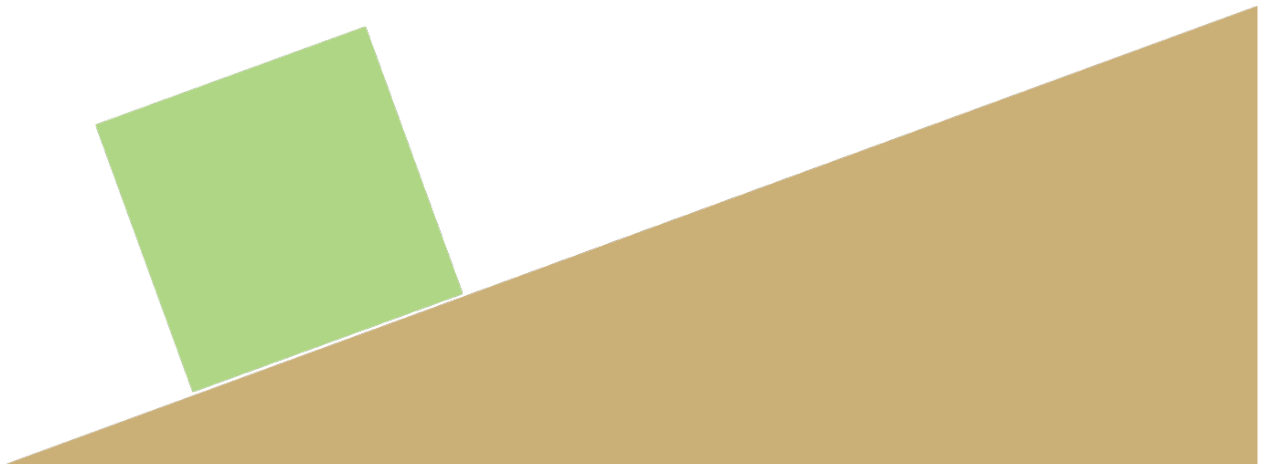
Fracture

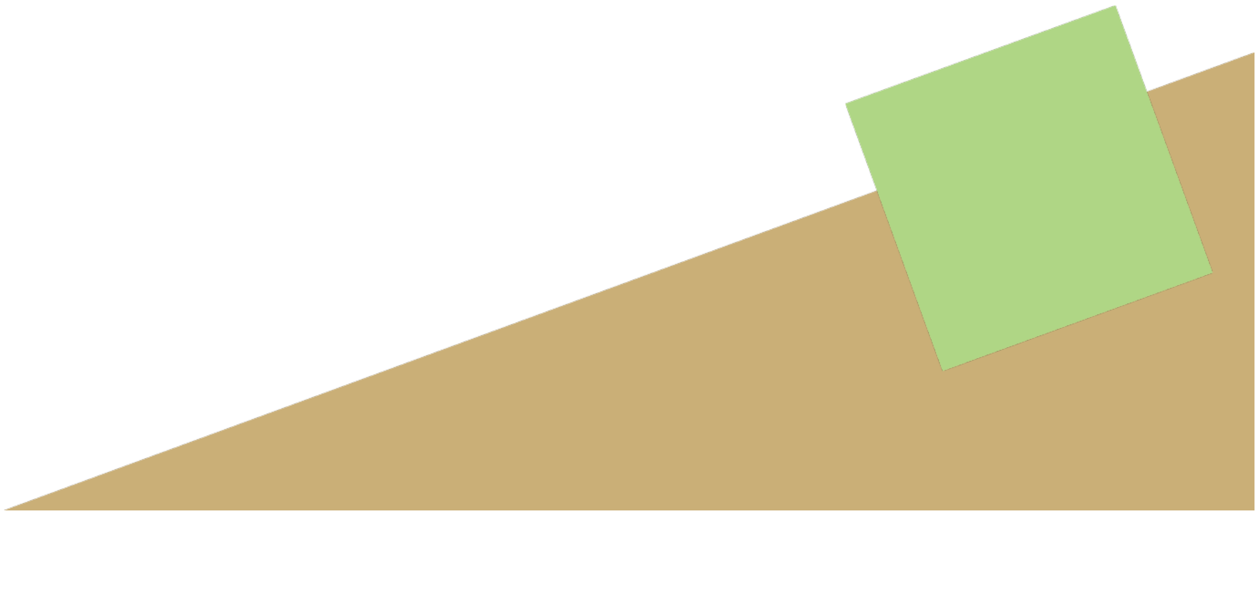
[Molino et al. 2004]

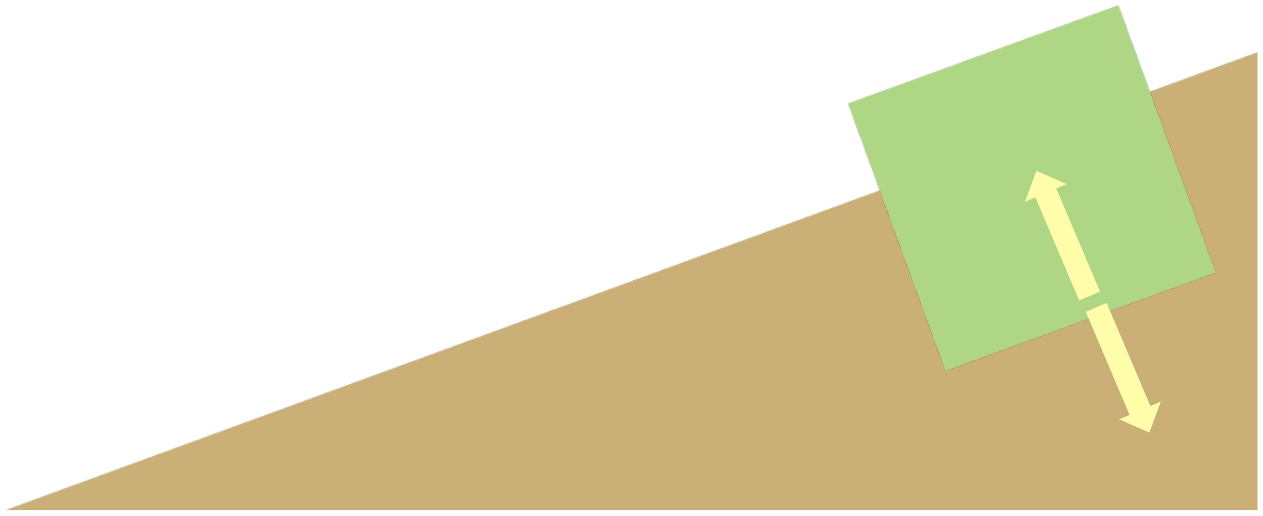
Contact and collision


frame 0: (0.00000)

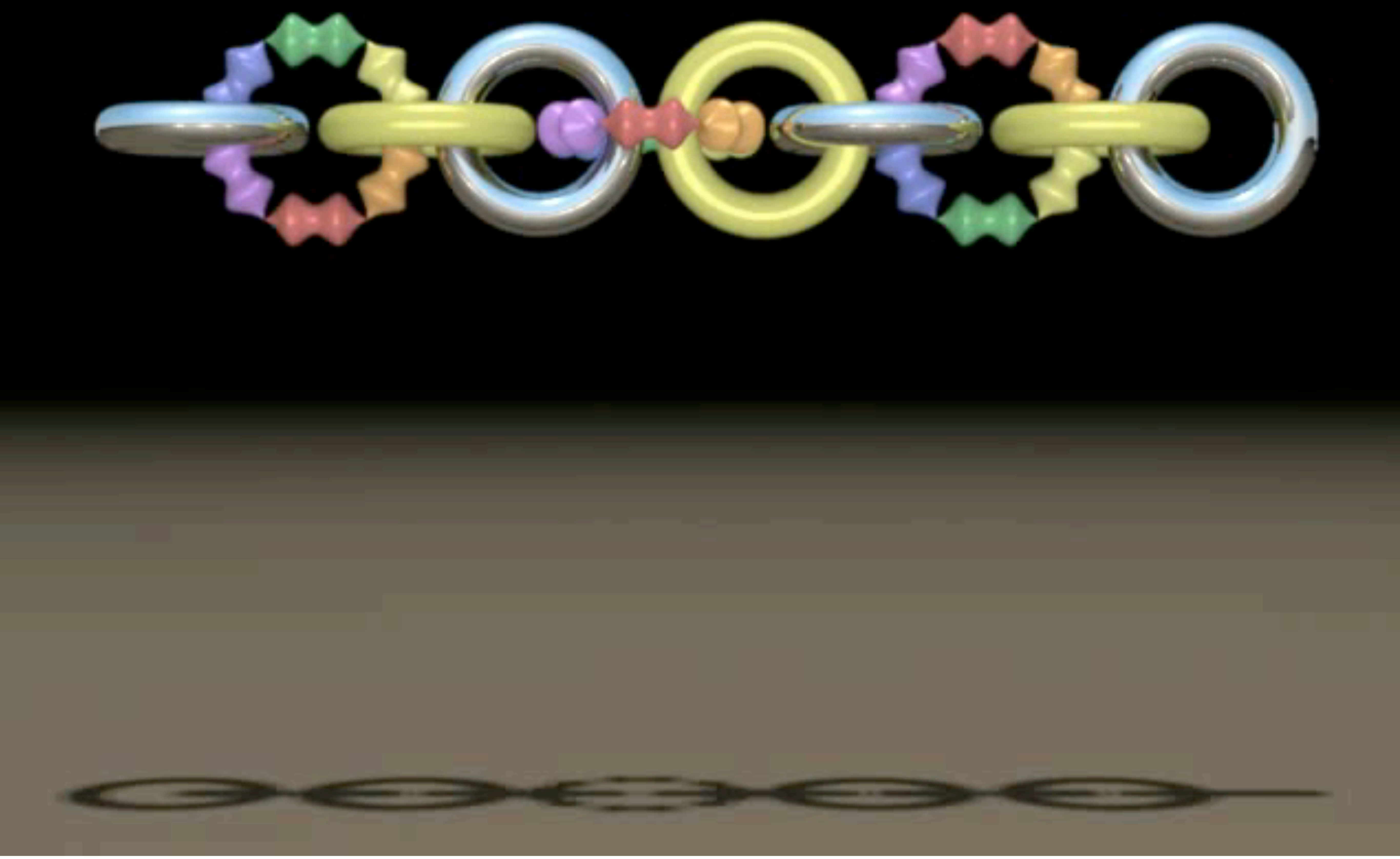

frame 25: (1.04167)

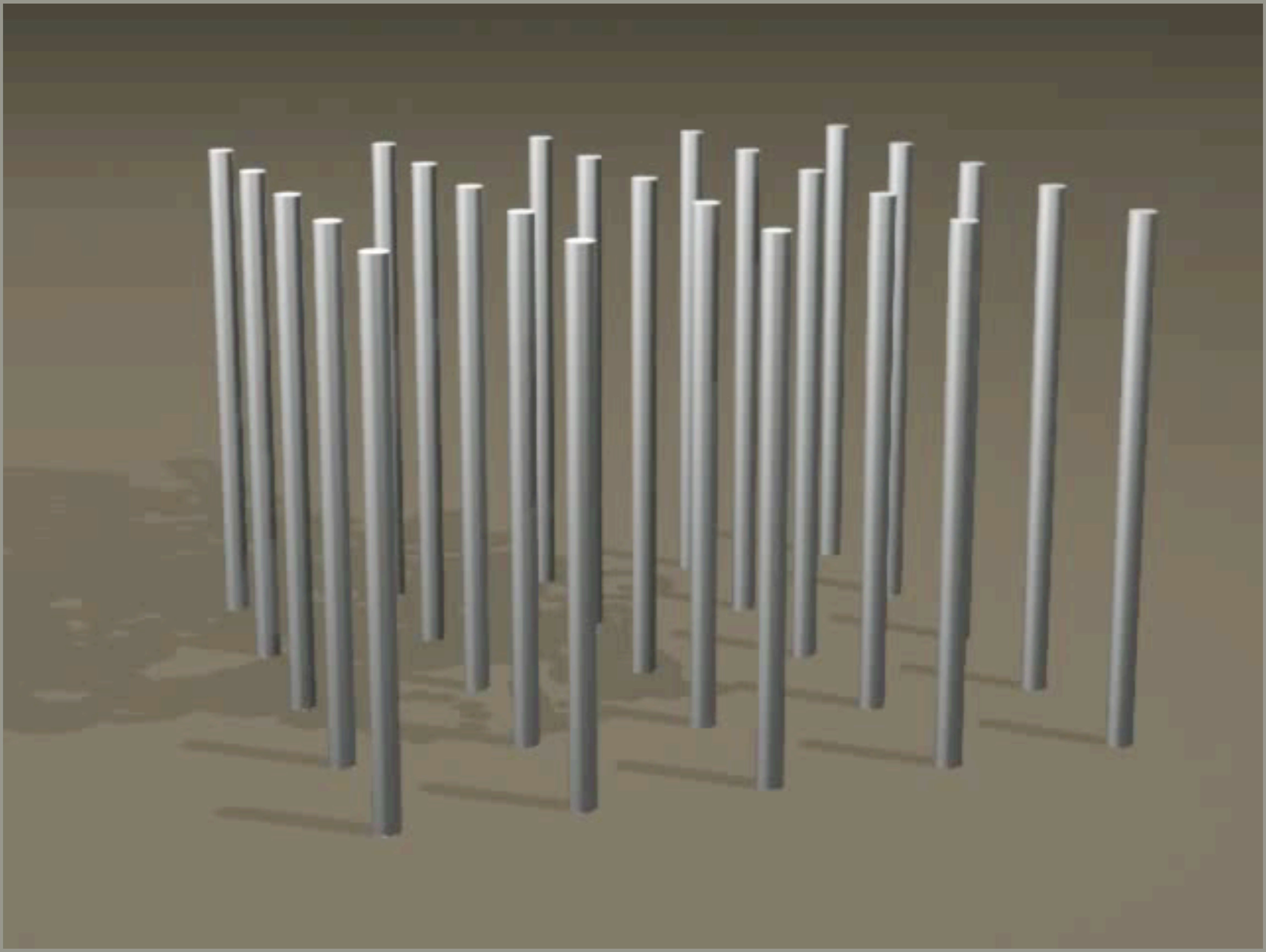




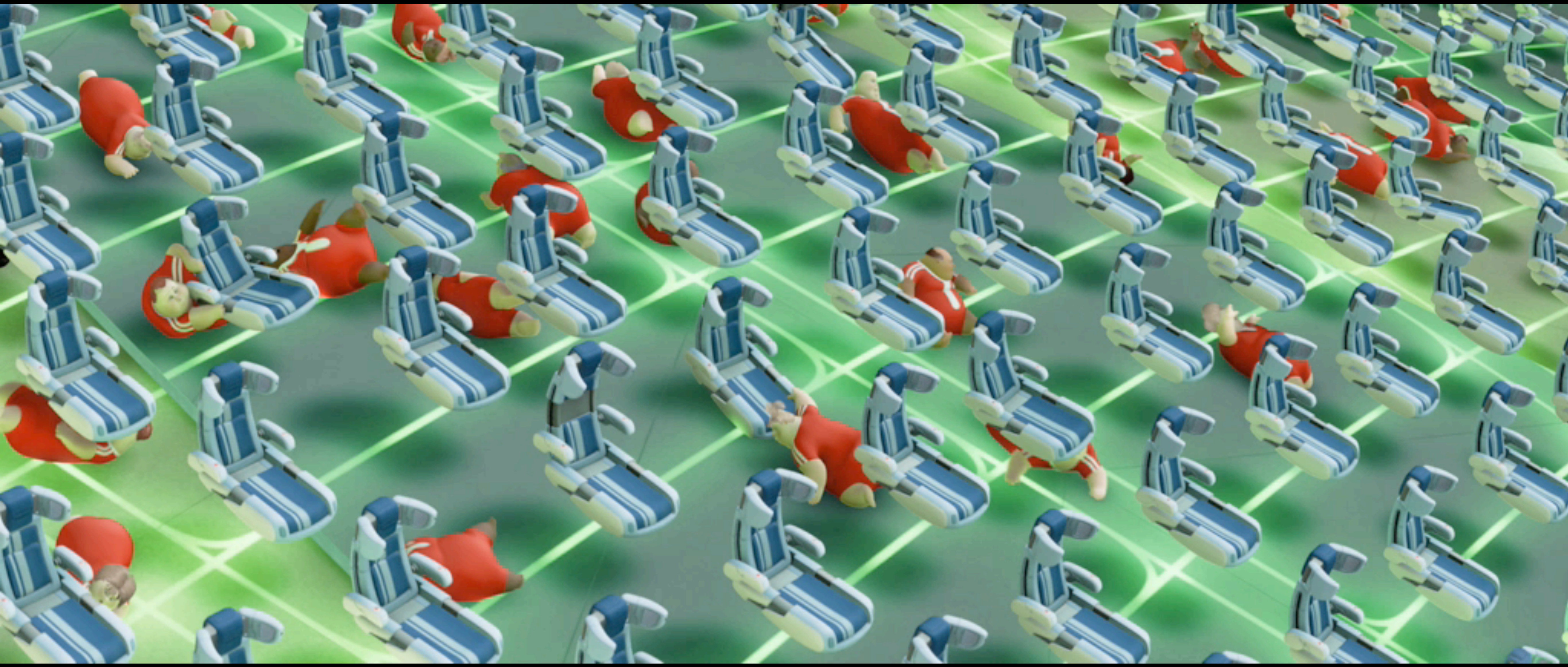


Simultaneous resolution of contact, elastic deformation, articulation constraints





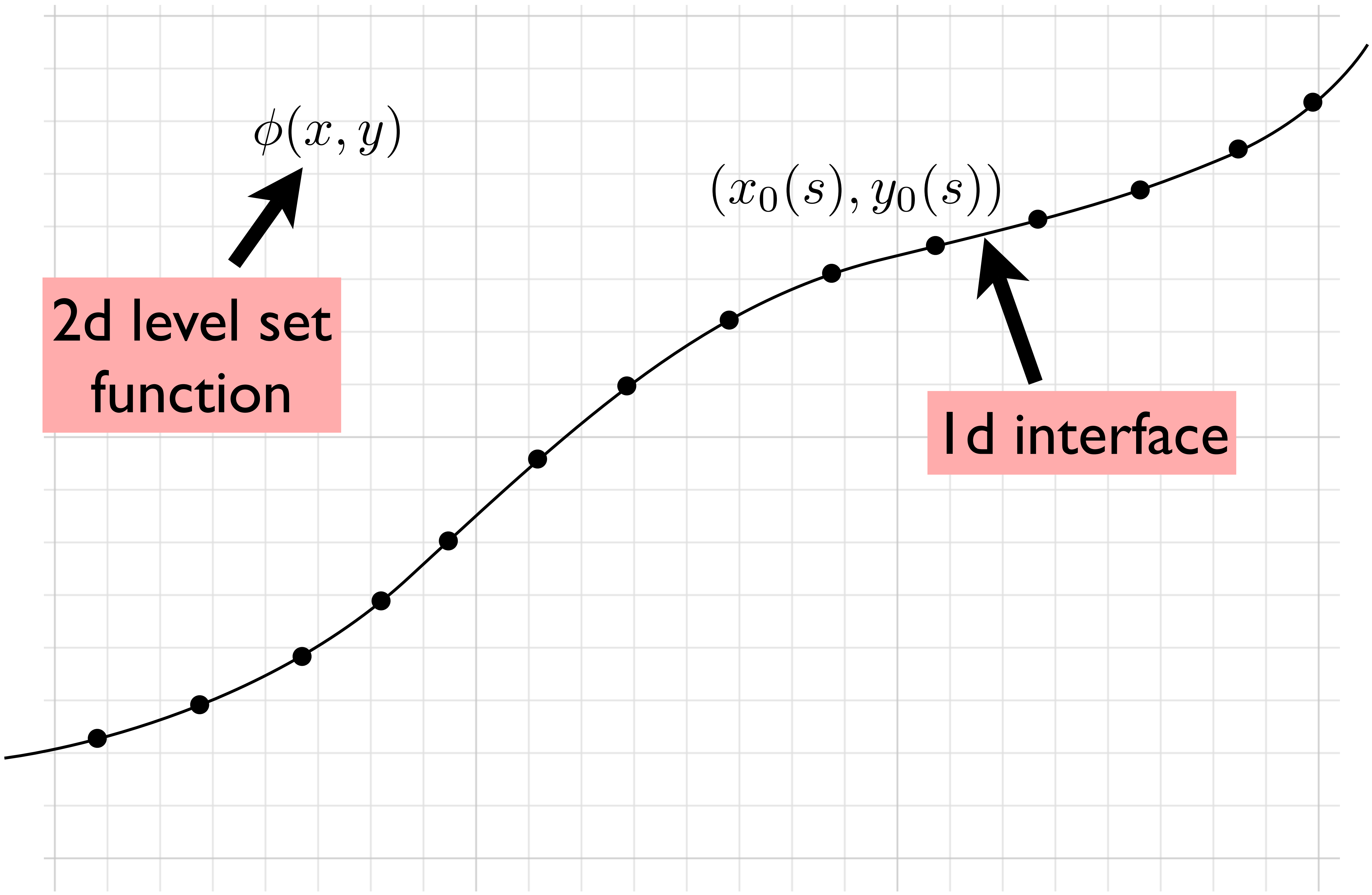
our rigid/deformable simulator in Pixar’s *WALL-E*



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

Fluid simulation

In fluid simulation, we often use a grid-based representation



$\phi(x,y)$

2d level set function

$(x_0(s), y_0(s))$

1d interface

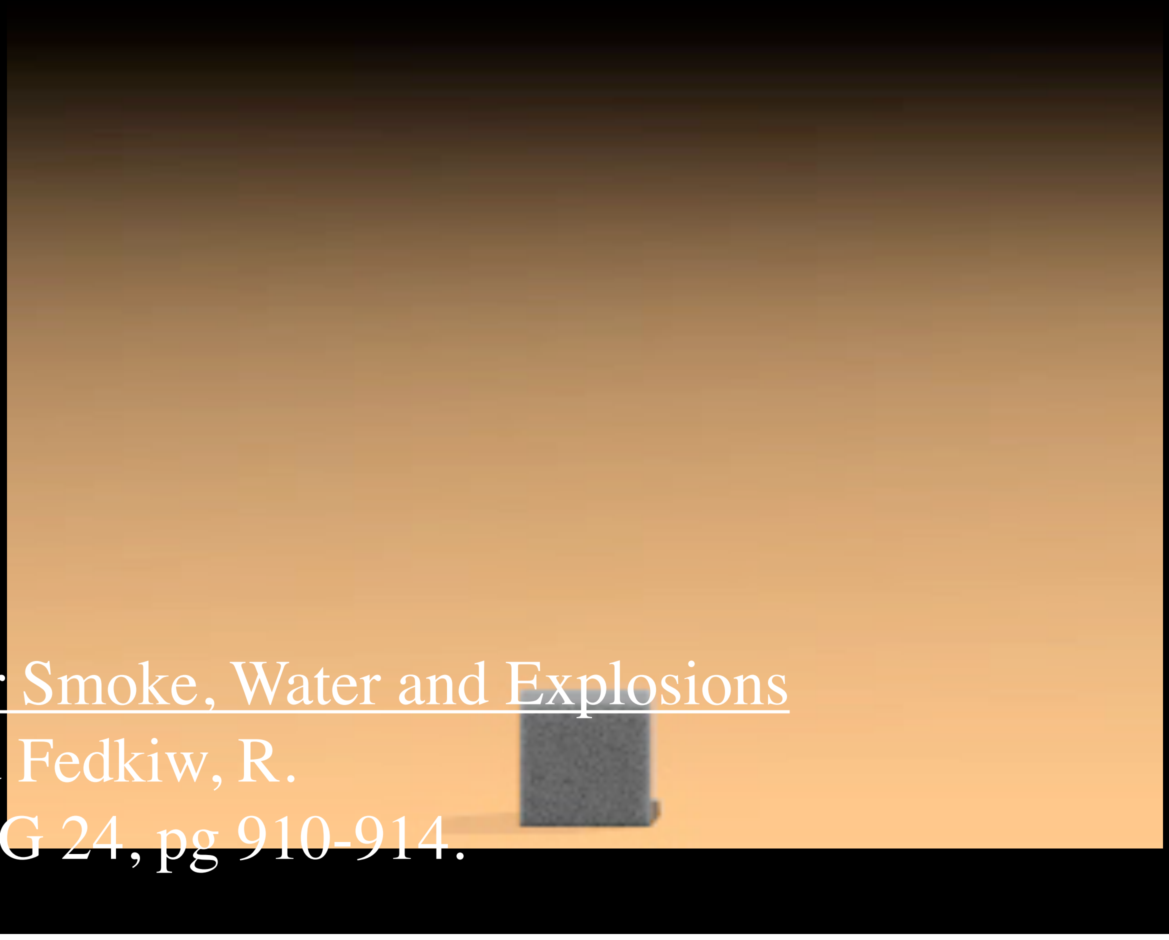
level set function may be e.g., Heaviside, signed distance

Fluid equations of motion: Navier-Stokes equations

$$\vec{F} = m\vec{a}$$

$$\rho(\mathbf{u}_t + \mathbf{u} \cdot \nabla \mathbf{u}) = \mu \triangle \mathbf{u} - \nabla p + \mathbf{f}$$

A Vortex Particle Method for Smoke, Water and Explosions
Selle, A., Rasmussen, N. and Fedkiw, R.
SIGGRAPH 2005, ACM TOG 24, pg 910-914.

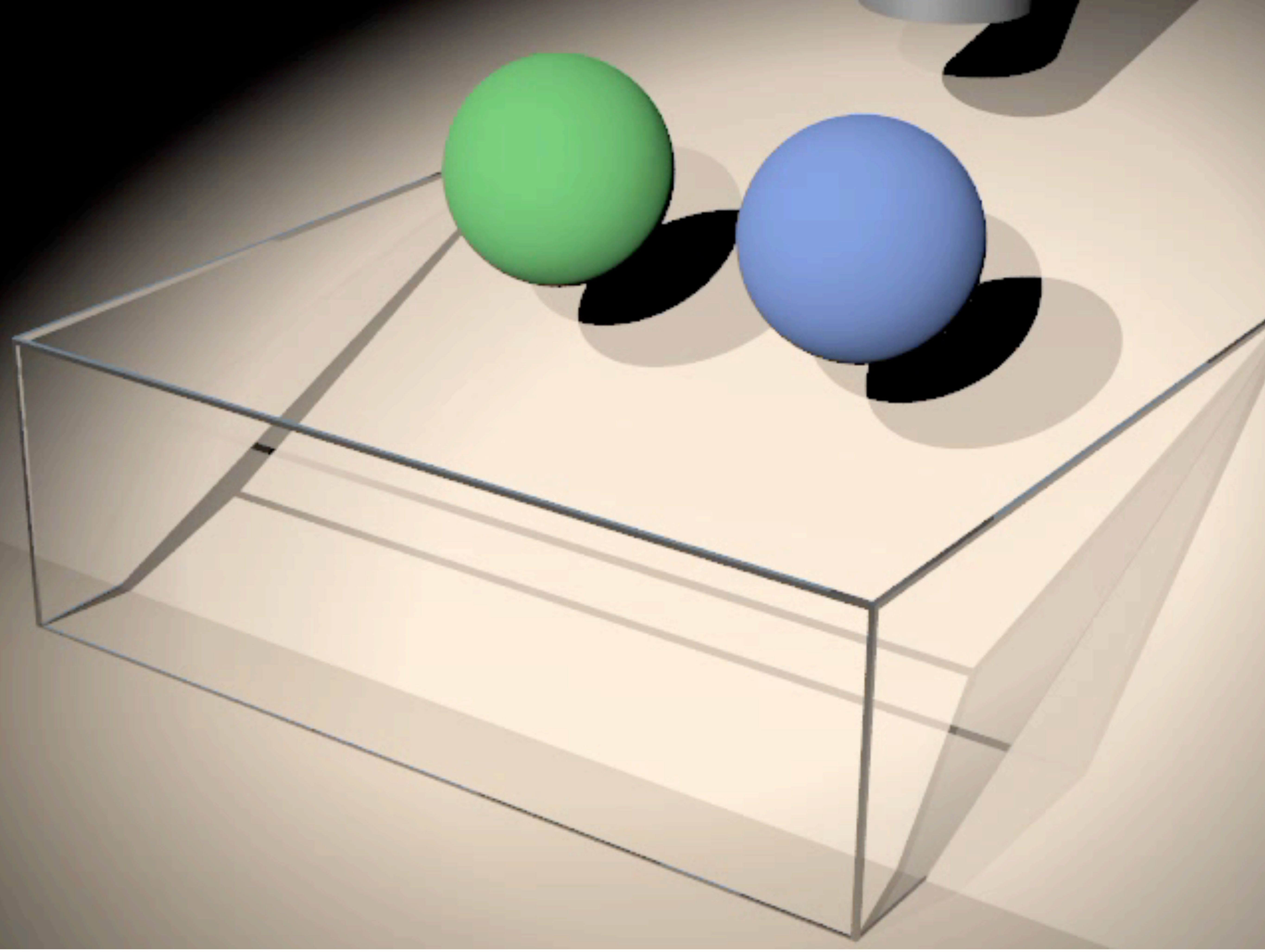


A Vortex Particle Method for Smoke, Water and Explosions

Selle, A., Rasmussen, N. and Fedkiw, R.

SIGGRAPH 2005, ACM TOG 24, pg 910-914.

- Navier-Stokes equations for viscous, incompressible flow
- we can use multiple pls and previous work for boundary condition capturing at interfaces to simulate many different liquids interacting

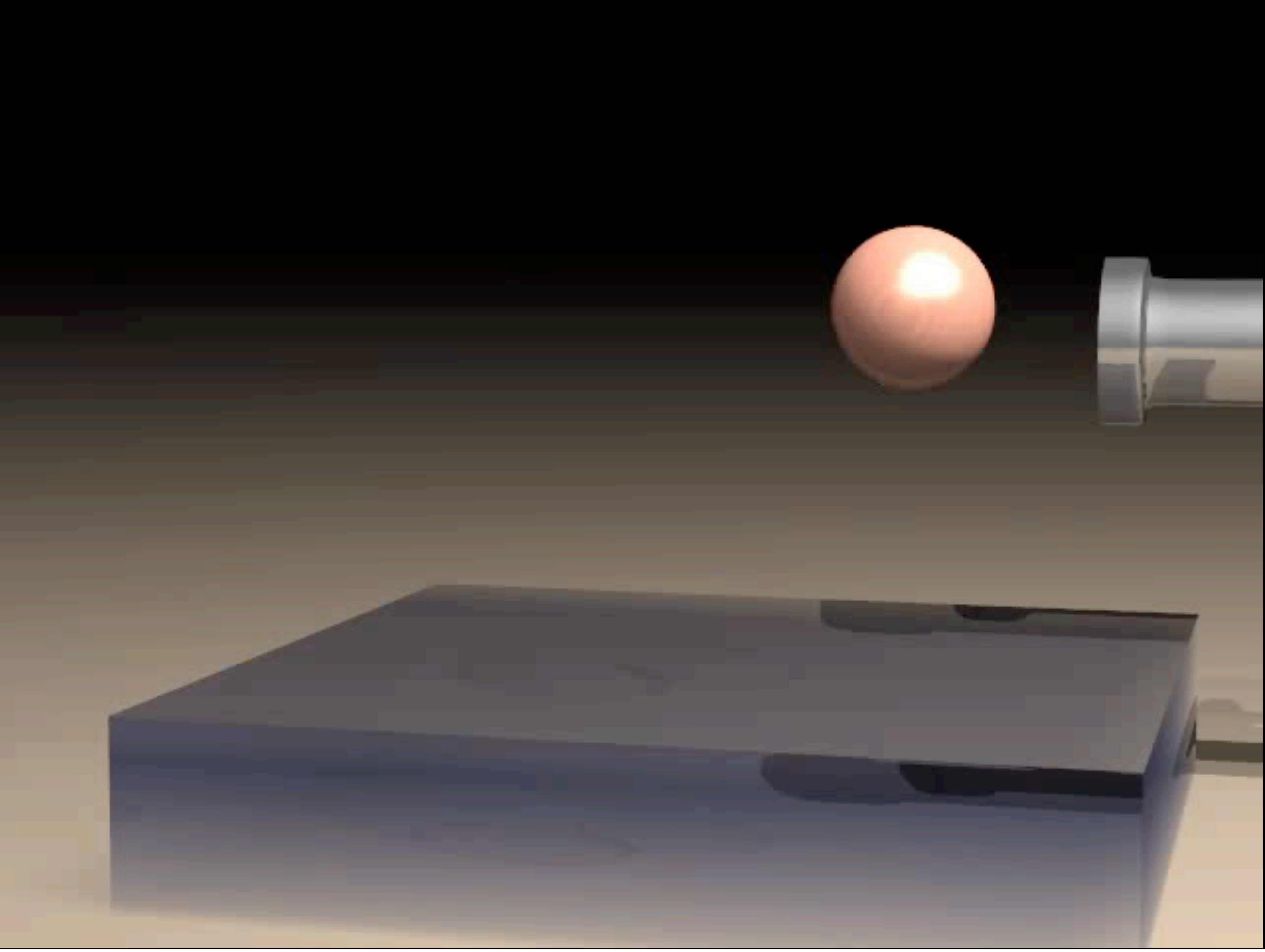


- Navier-Stokes equations for viscous, incompressible flow
- we can use multiple pls and previous work for boundary condition capturing at interfaces to simulate many different liquids interacting

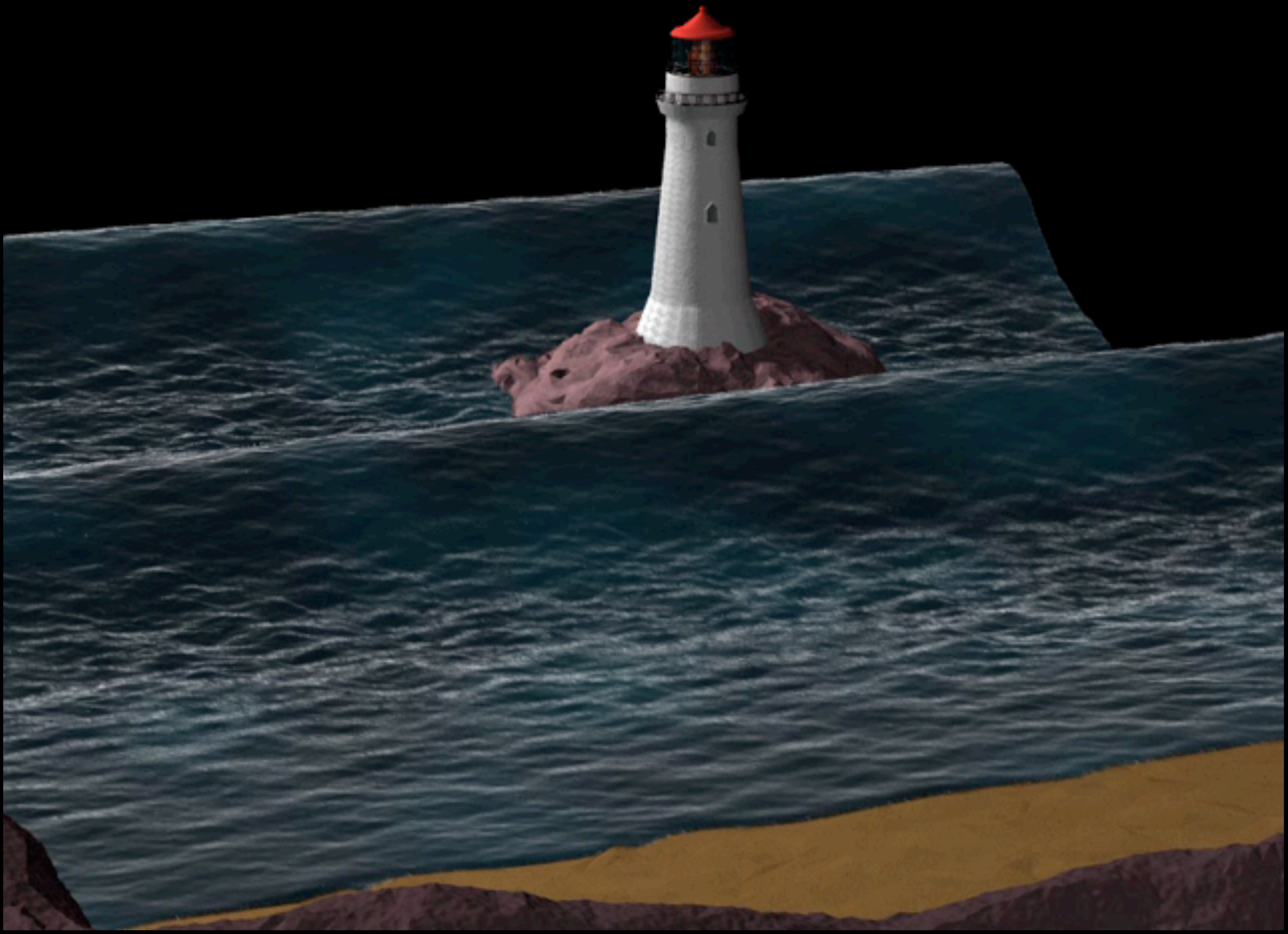
- Couple to Navier-Stokes based fluid simulator
- Wrinkling and cellular patterns in flame front

- Couple to Navier-Stokes based fluid simulator
- Wrinkling and cellular patterns in flame front

- other surface phenomena, such as fire (one material converting into another)



– other surface phenomena, such as fire (one material converting into another)



Two-way Coupled SPH and Particle Level Set Fluid Simulation

Losasso, F., Talton, J., Kwatra, N. and Fedkiw, R. IEEE TVCG 14, No. 4 (2008)

Control of virtual character

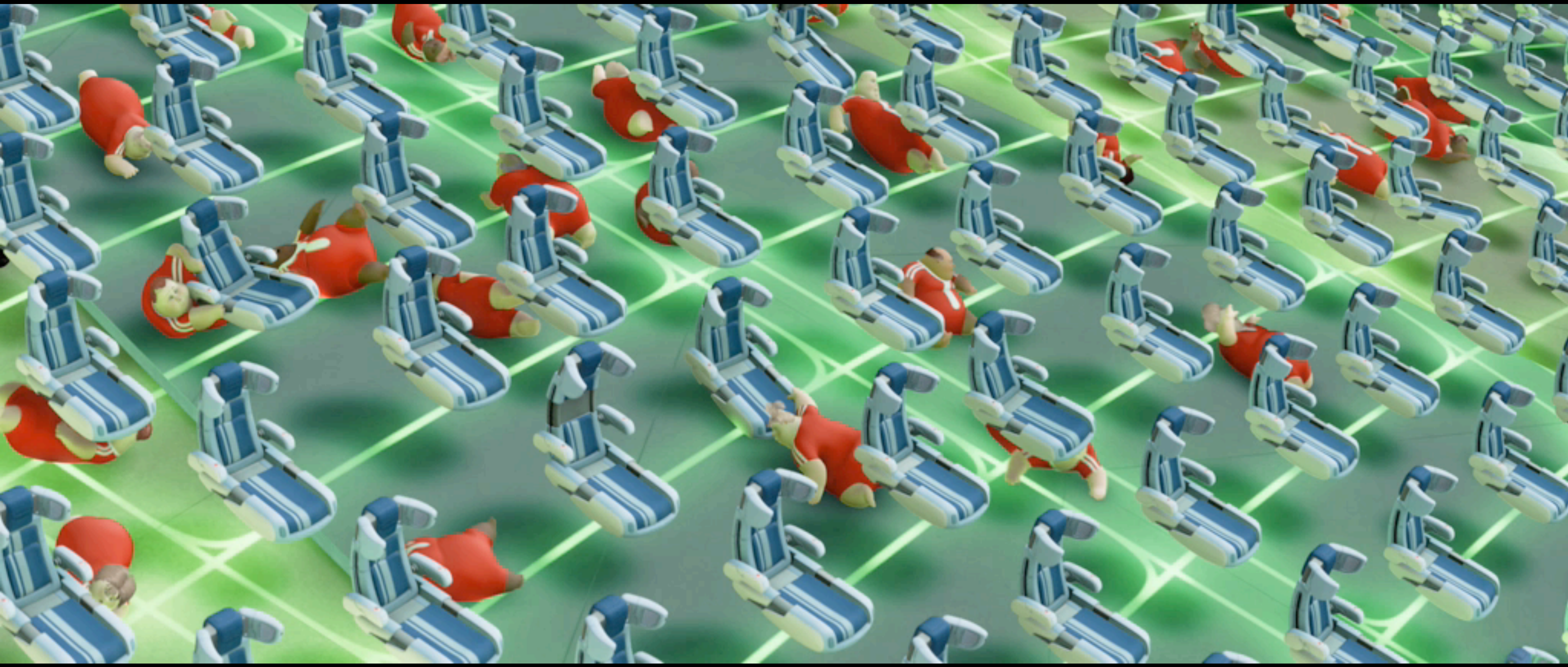
[Shinar et al. 2008]

issues: control algorithms, interaction with environment

Control of virtual character

[Shinar et al. 2008]

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



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