Animation, Motion Capture, Keyframing

(oh and a little Radiosity)

source: http://scaq.blogspot.com/2006_11_01_archive.html

Adrien Treuille

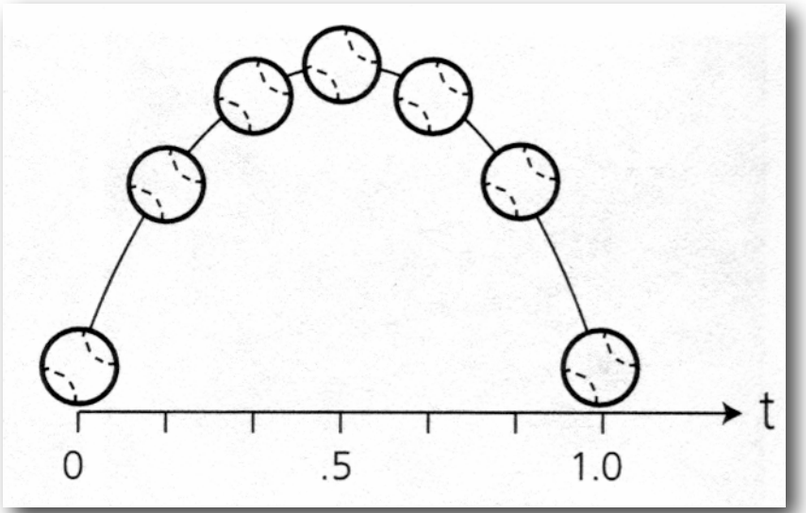
Overview

Announcements

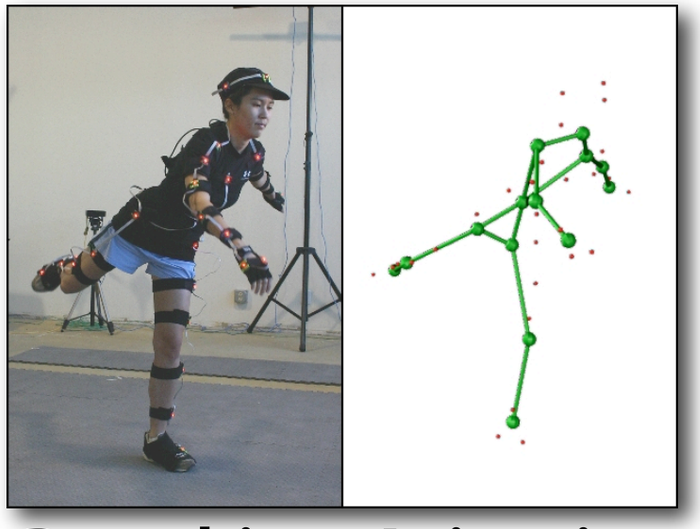
Radiosity

Animation Intro

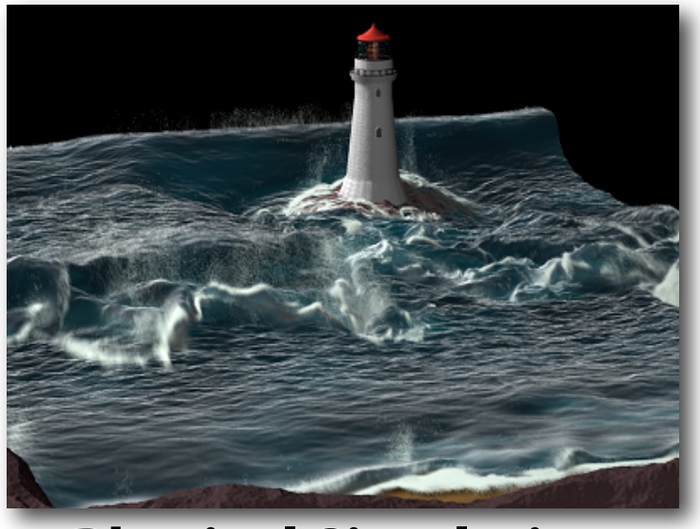
Cell Animation



Keyframing



Data-driven Animation



Physical Simulation

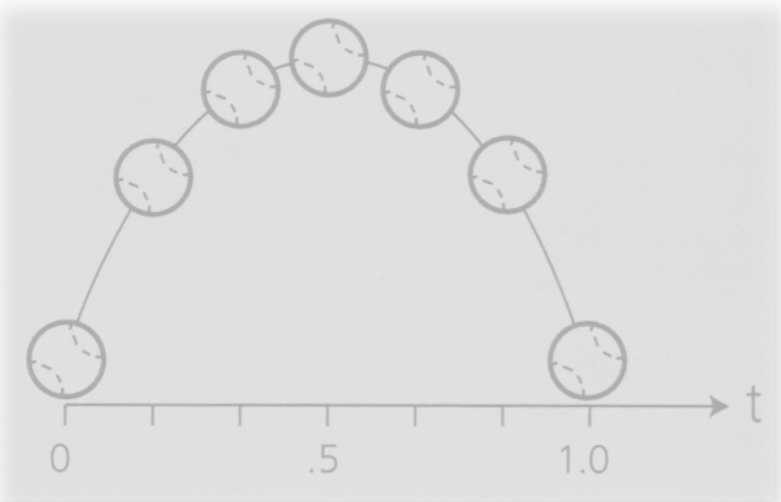
Overview

Announcements

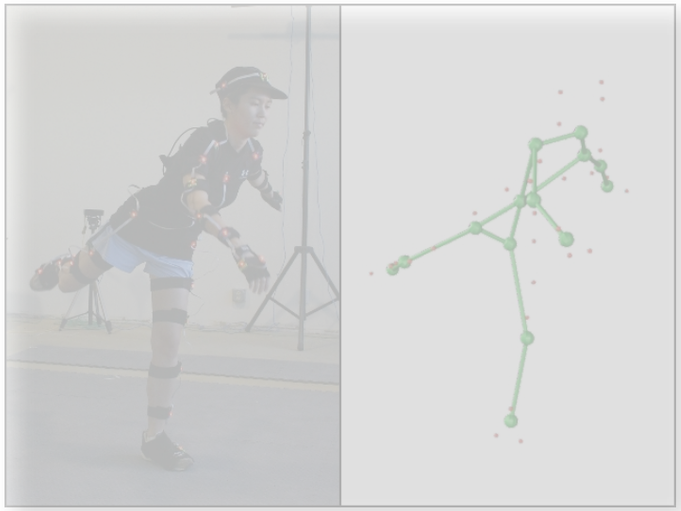
Radiosity

Animation Intro

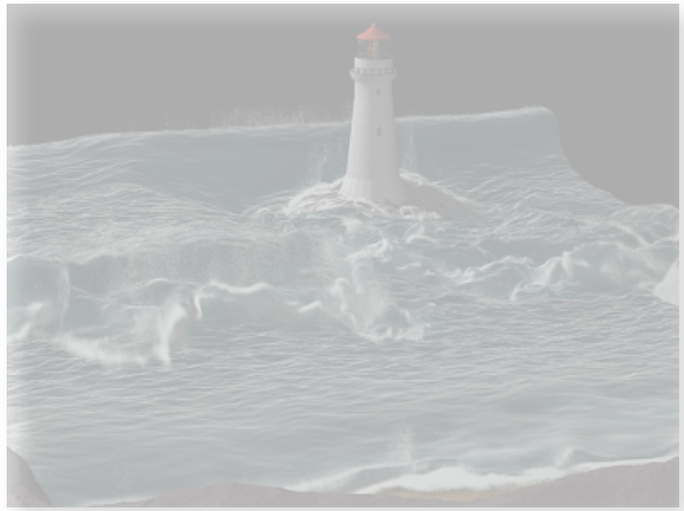
Cell Animation



Keyframing

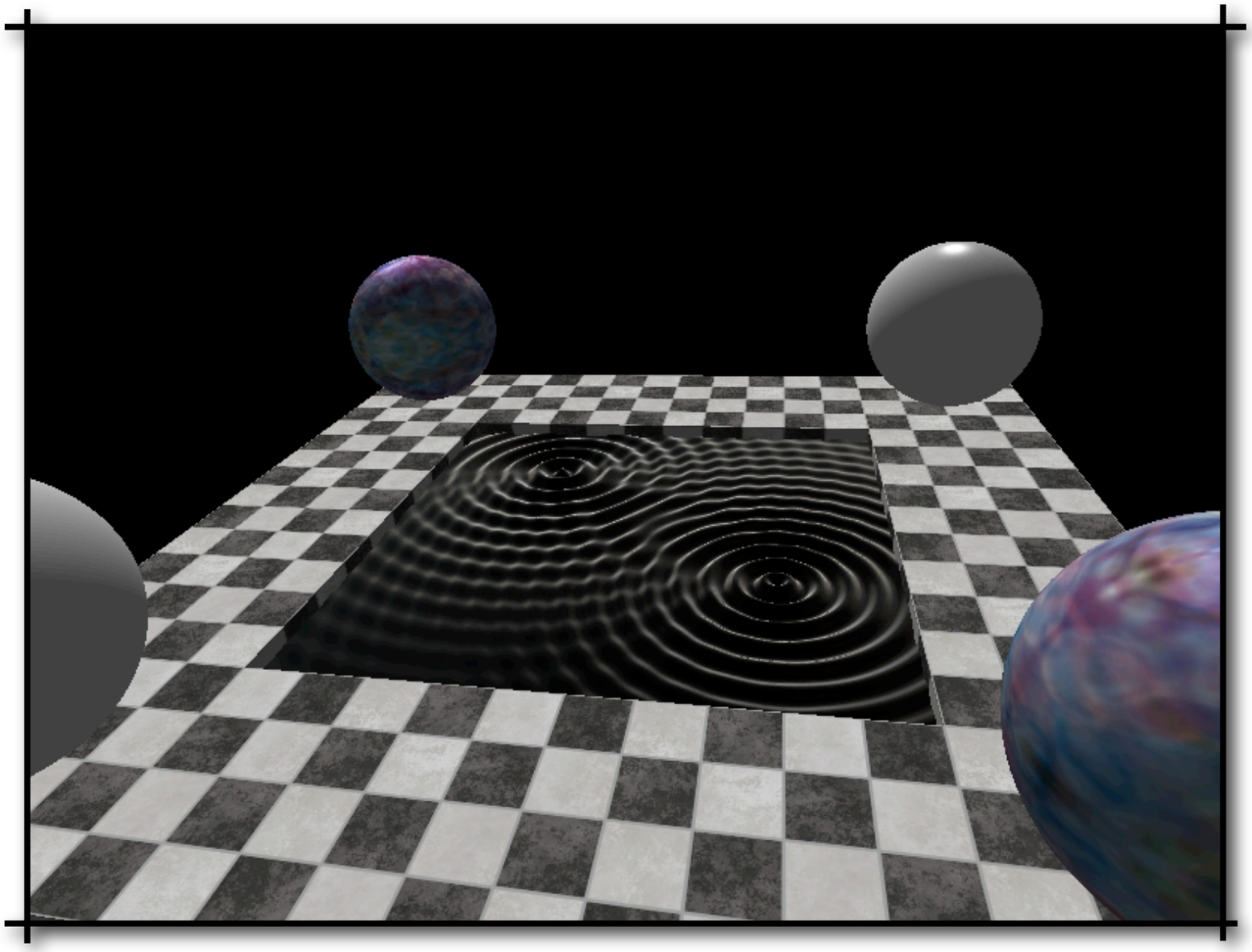


Data-driven Animation

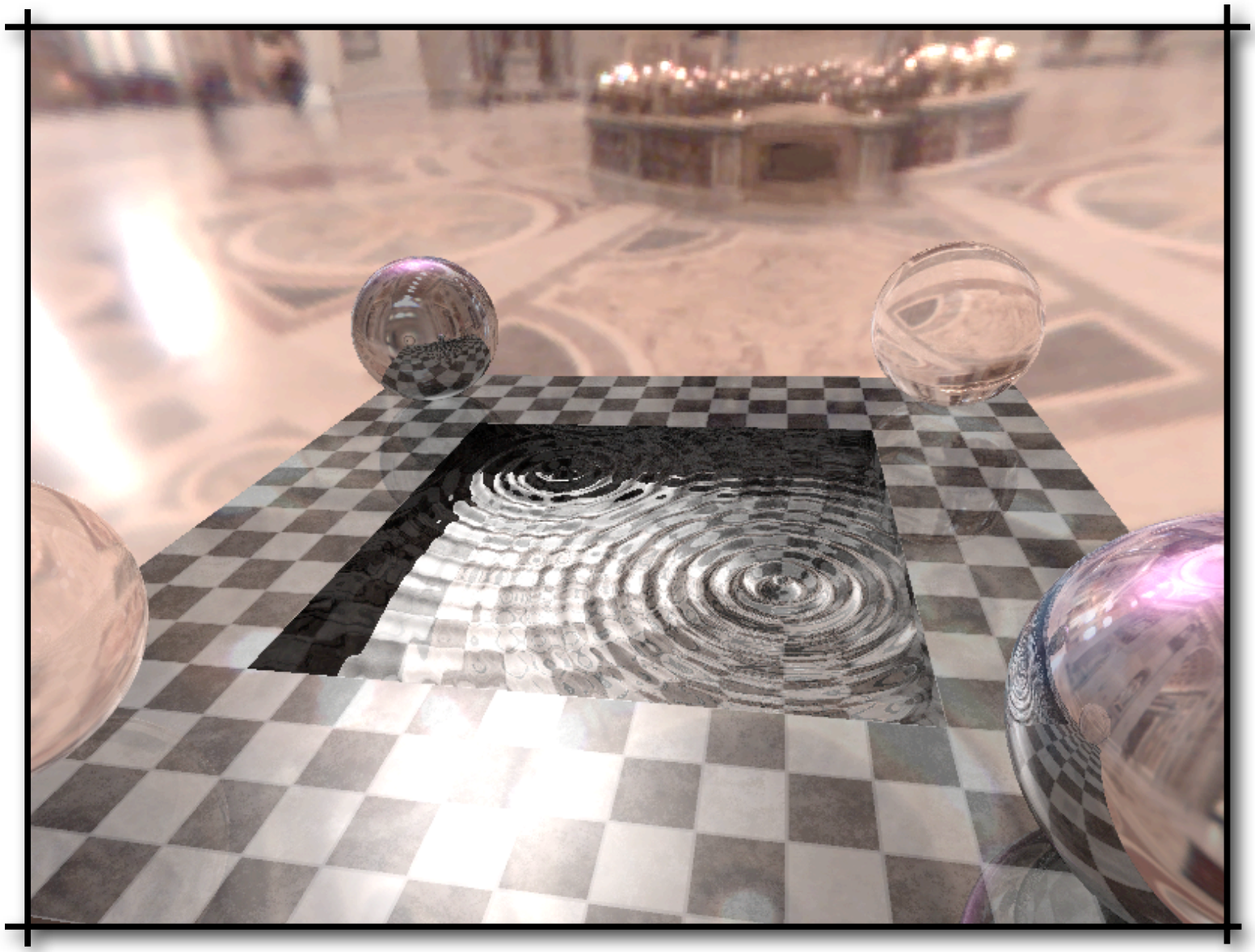


Physical Simulation

Project 2 Graded

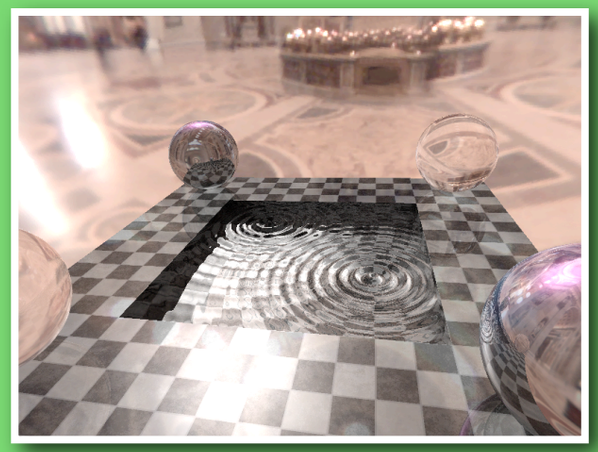


Project 3



Please start the Raytracer!!!

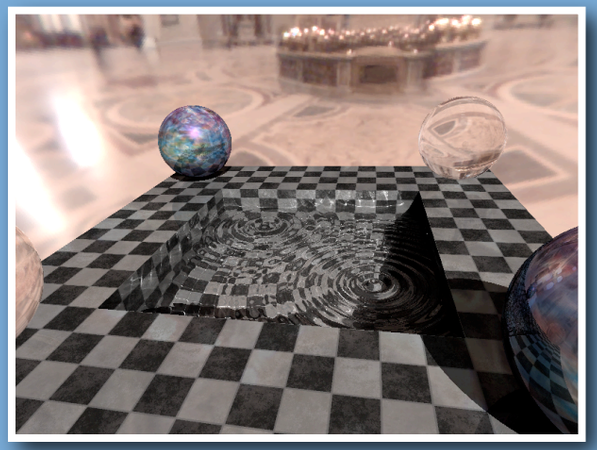
Project 3&4 Grading



Project 3

- Requirement A
- Requirement B
- Requirement C

50% BACK



Project 4

- Requirement A
- Requirement B
- Requirement C
- Requirement D
- Requirement E
- Requirement F

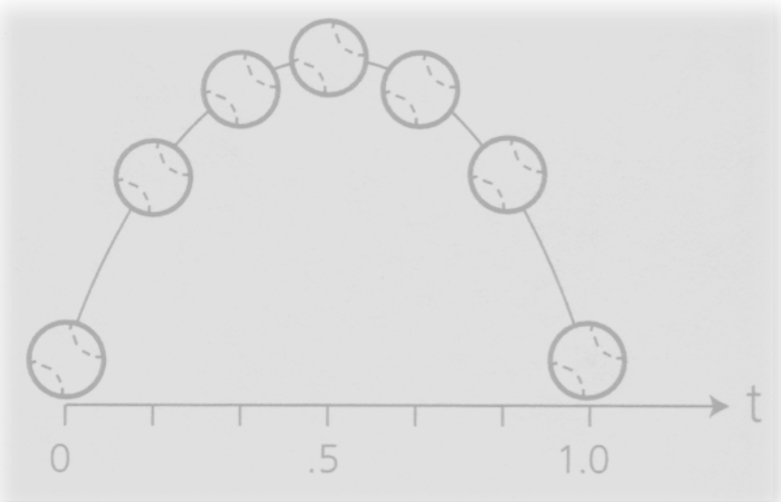
Overview

Announcements

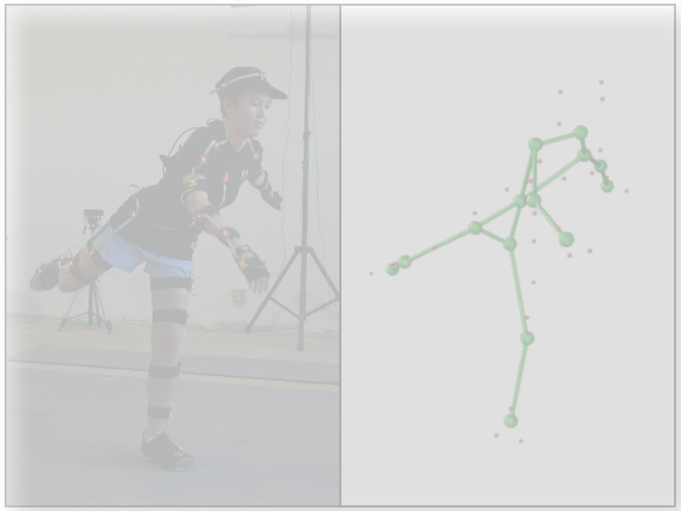
Radiosity

Animation Intro

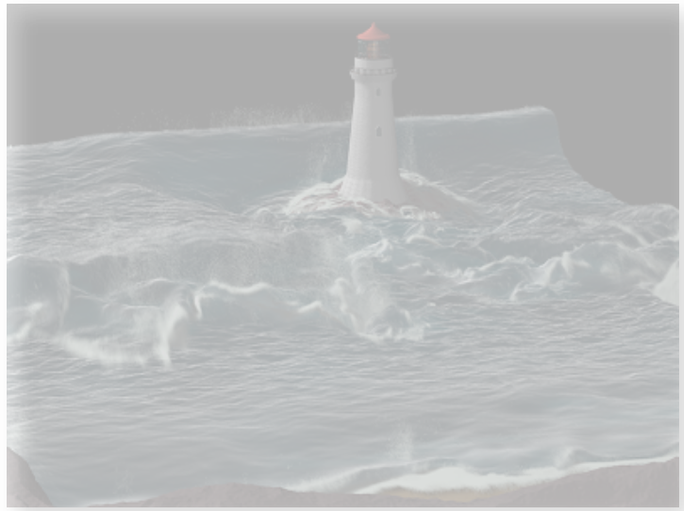
Cell Animation



Keyframing



Data-driven Animation



Physical Simulation

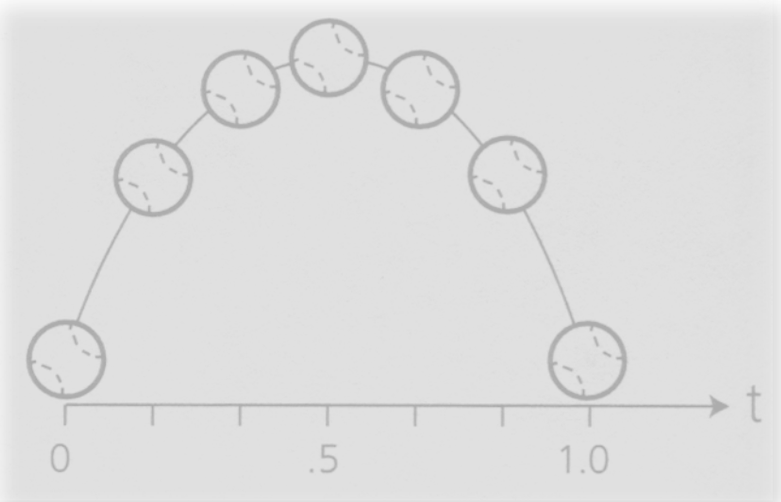
Overview

Announcements

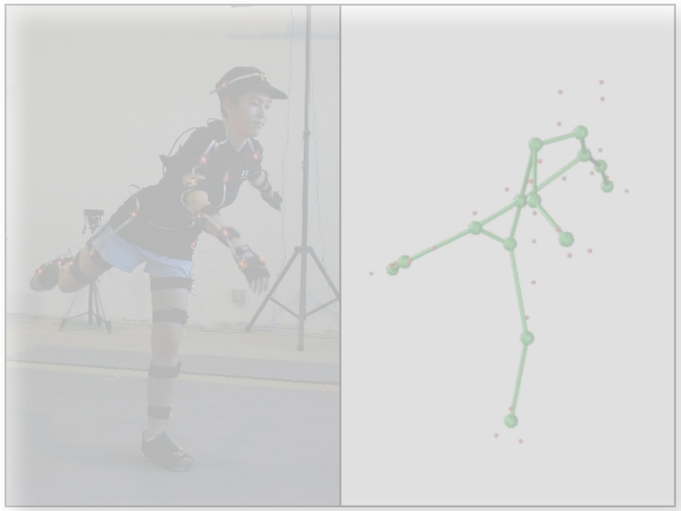
Radiosity

Animation Intro

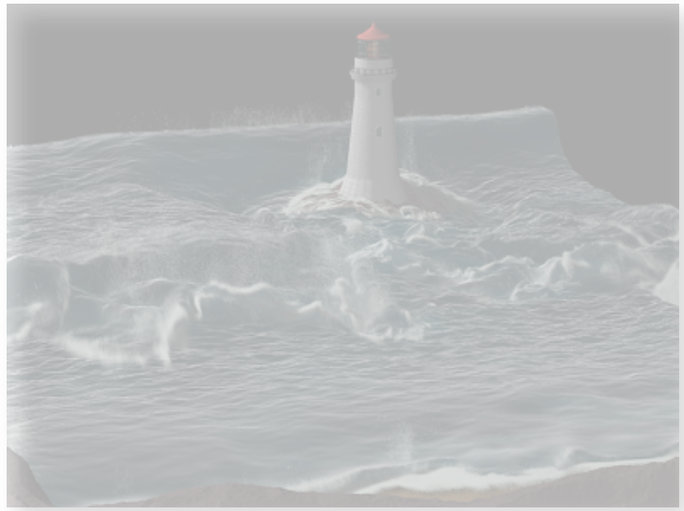
Cell Animation



Keyframing



Data-driven Animation



Physical Simulation

Classical Radiosity in a Nutshell, Revised

- Divide all surfaces into patches.
- Calculate form factors between all patches.
  - Lighting and viewer independent
- Solve the radiosity equation.
  - Viewer independent
- Render using standard 3D hardware.

Linear System

$$B_i = E_i + R_i \sum_j B_j F_{ij}$$

Our new equation gives the radiosity (*B*) of a single patch, so to specify the radiosity of all *n* patches we need *n* radiosity equations, one for each patch

Known values:
E (given), *R* (given), *F* (computable)

Unknown: *B*

n equations, *n* unknowns

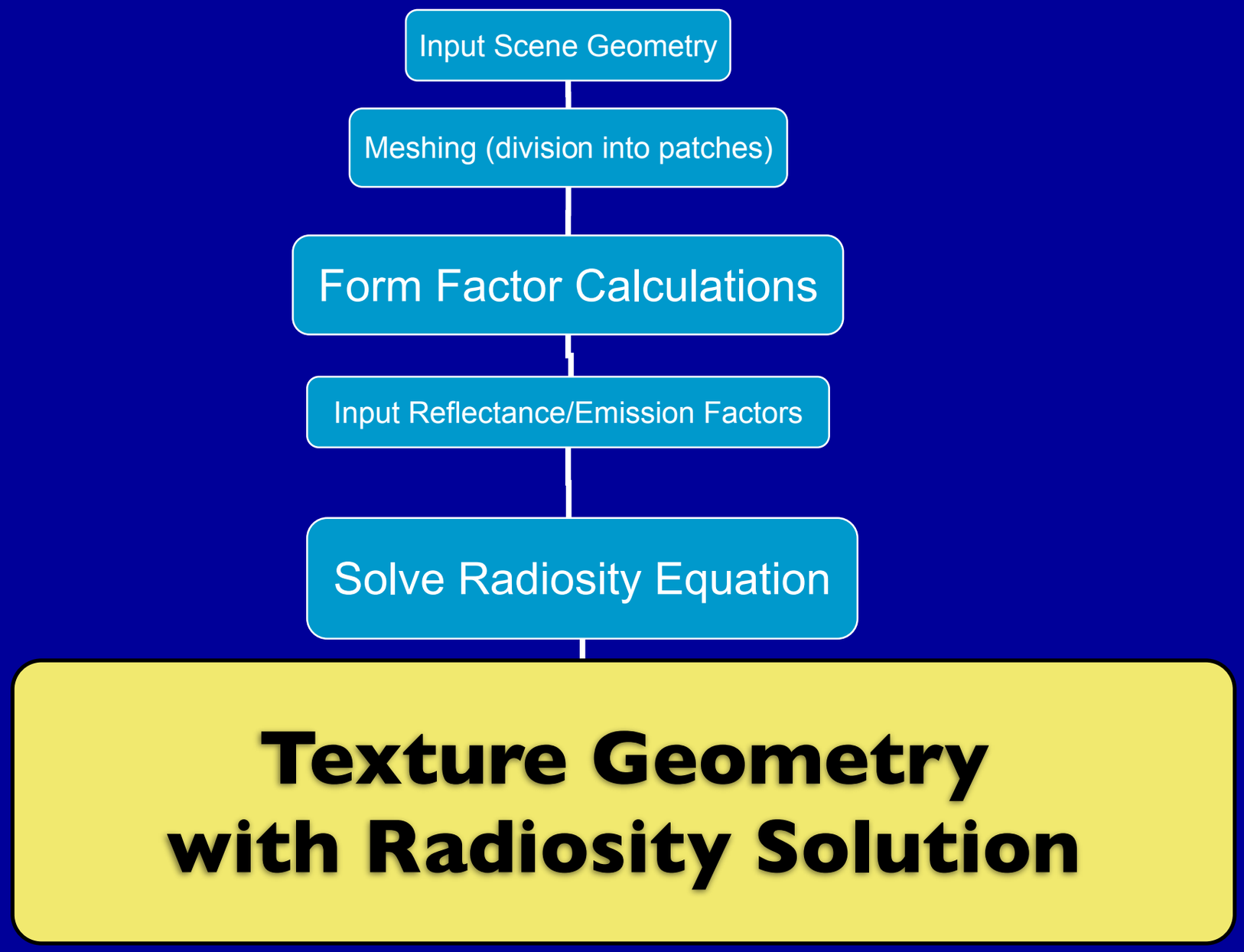
Linear System

Restate as a matrix equation…and solve

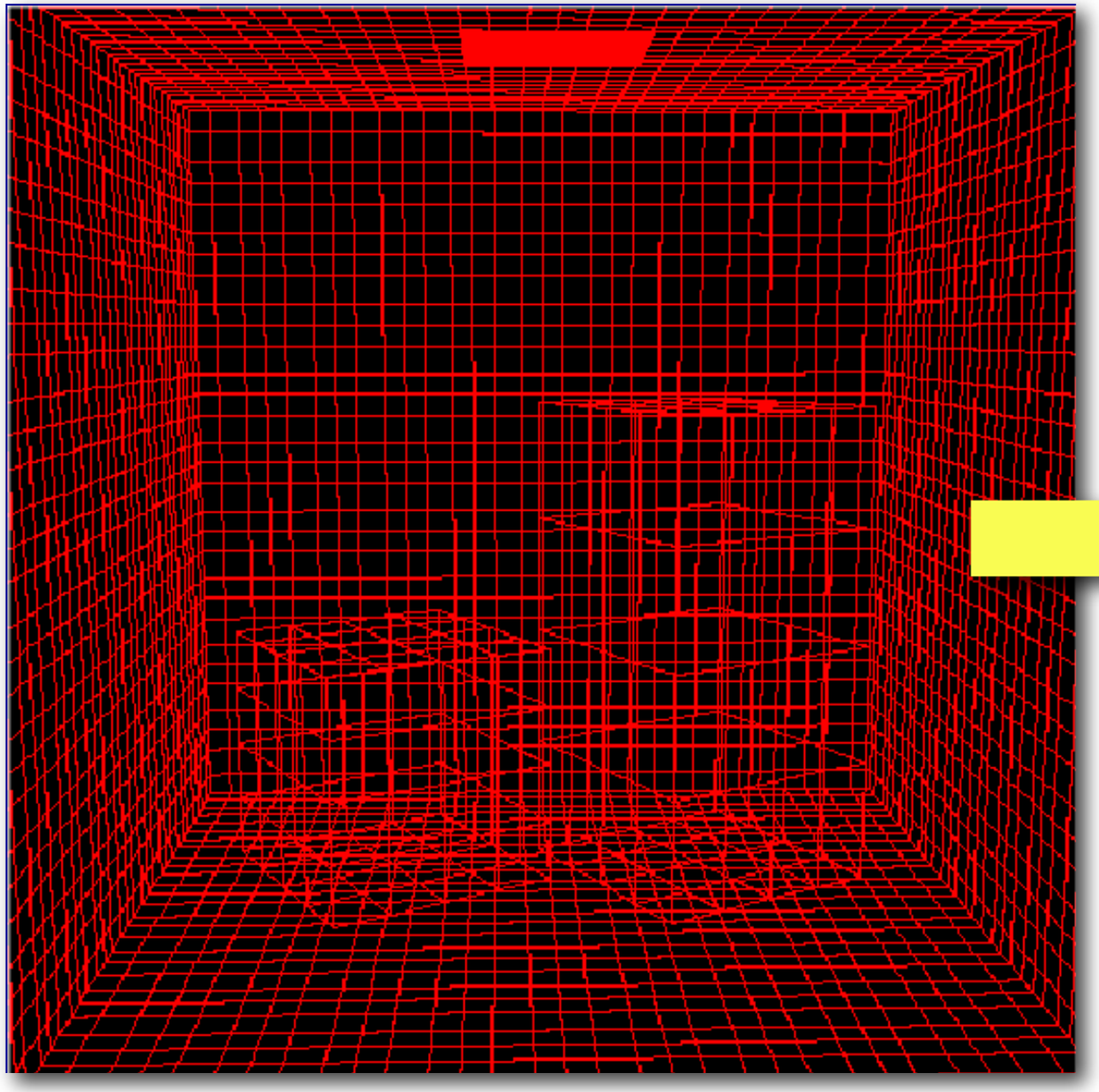
$$\begin{bmatrix} 1-R_1F_{11} & -R_1F_{12} & \dots & R_1F_{1n} \\ -R_2F_{21} & 1-R_2F_{22} & \dots & R_2F_{2n} \\ \dots & \dots & \dots & \dots \\ -R_nF_{n1} & R_nF_{n2} & \dots & 1-R_nF_{nn} \end{bmatrix} * \begin{bmatrix} B_1 \\ B_2 \\ \dots \\ B_n \end{bmatrix} = \begin{bmatrix} E_1 \\ E_2 \\ \dots \\ E_n \end{bmatrix}$$

Each of our *n* linear equations contains *n* double integrals, one for each form factor.

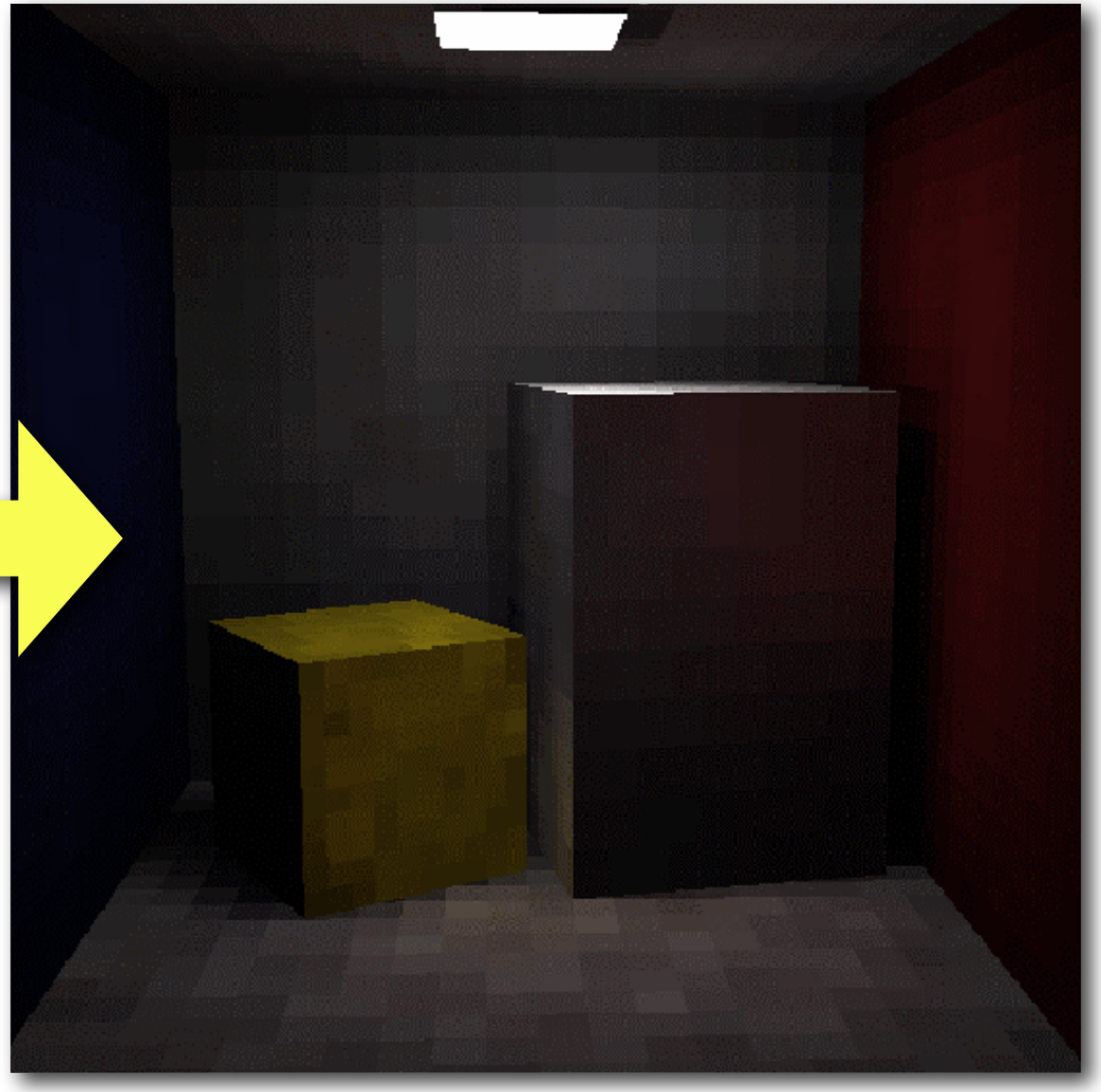
The Radiosity "Pipeline"



Example

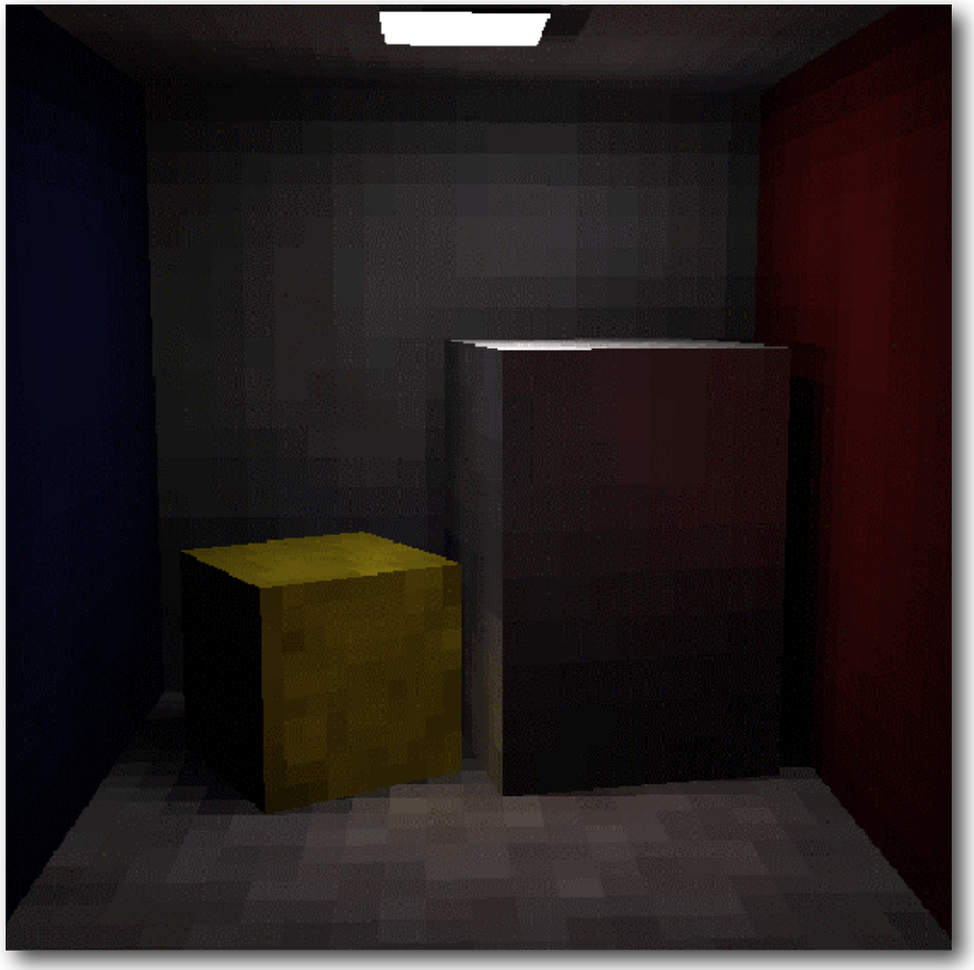


Wireframe



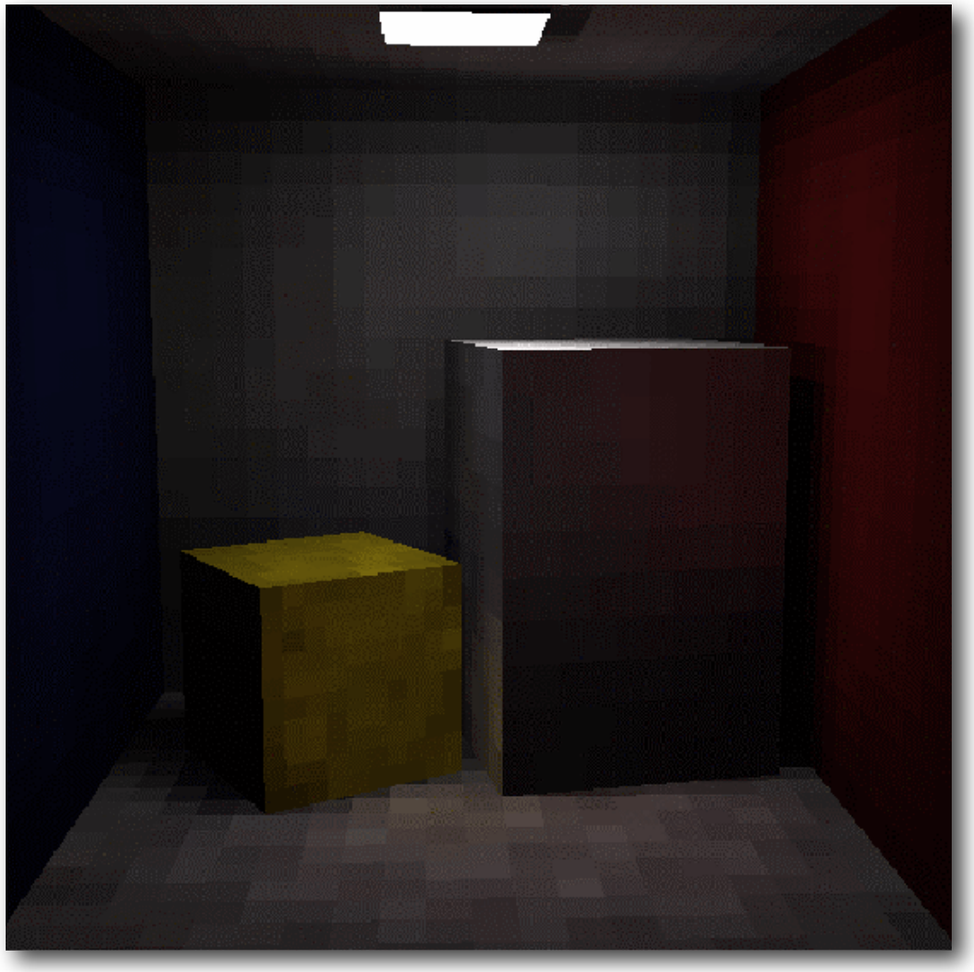
Result

Problems



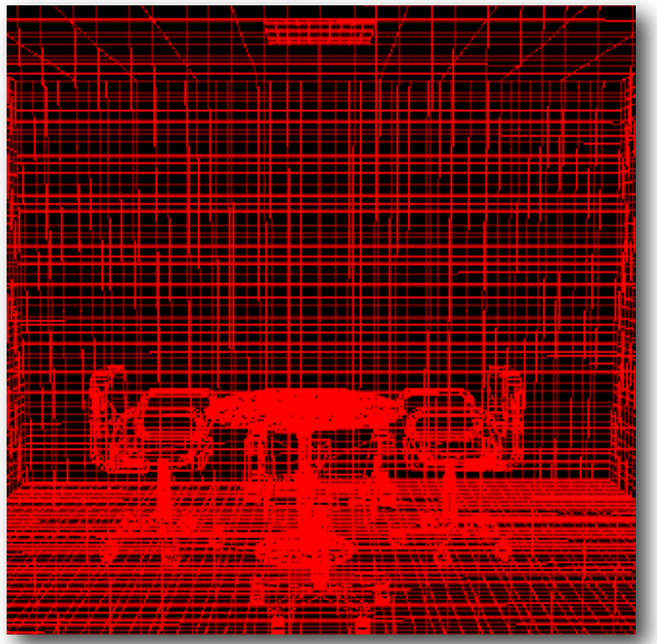
- **Blocky (Aliasing)**
  - ➟ **More Patches**
  - ➟ **Adaptive Subdivision**
- **No Specular Effects**
  - ➟ **2-pass Algorithms**

Problems



- **Blocky (Aliasing)**
  - ➟ **Adaptive Subdivision**
- **No Specular Effects**
  - ➟ **2-pass Algorithms**

Patch Size



Wireframe

300

1200

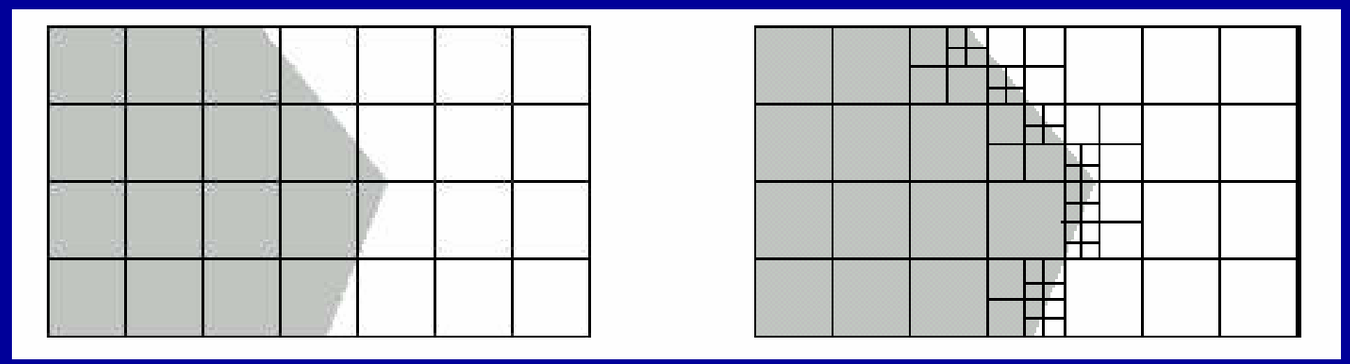
2500

2500
(interpolated)

100
(interpolated,
supersampled)

Adaptive Subdivision

Subdivide elements adaptively:

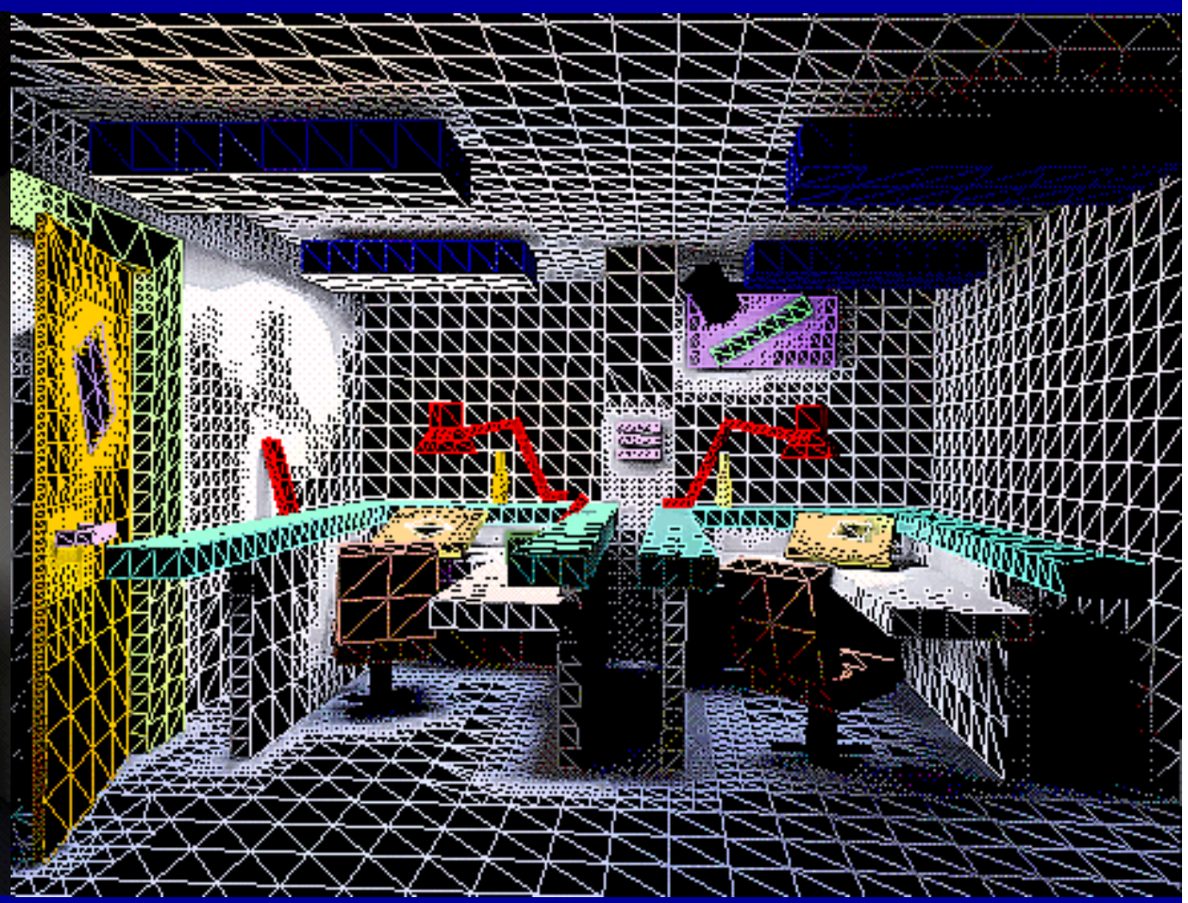


Begin with elements identical to patches.

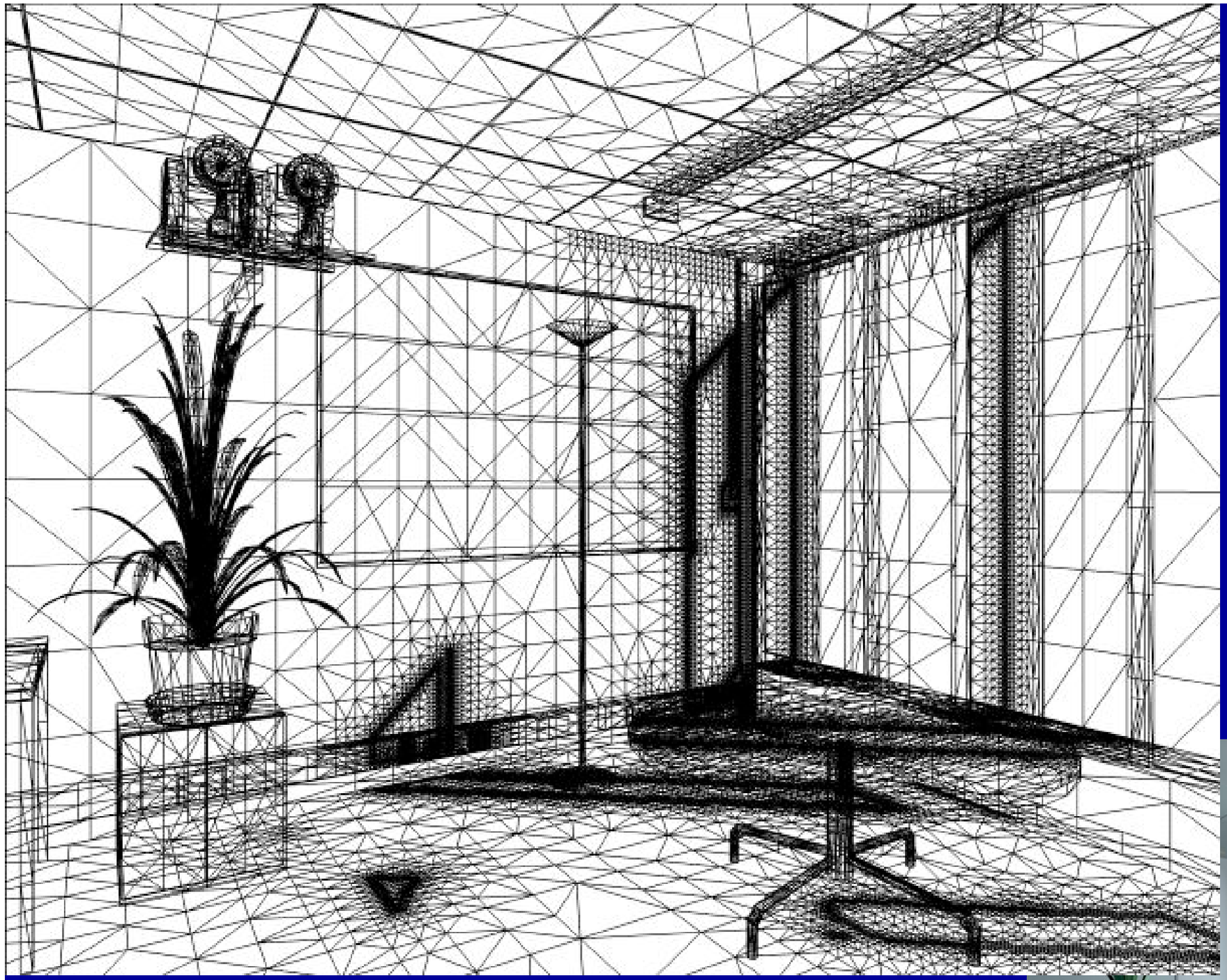
Determine radiosity of an element, then compare to neighbors to obtain an error value. If within some error threshold, assign constant radiosity (or optionally interpolate).

Otherwise, subdivide the element and recurse until the error threshold or a minimum element size is reached.

Adaptive Subdivision Examples

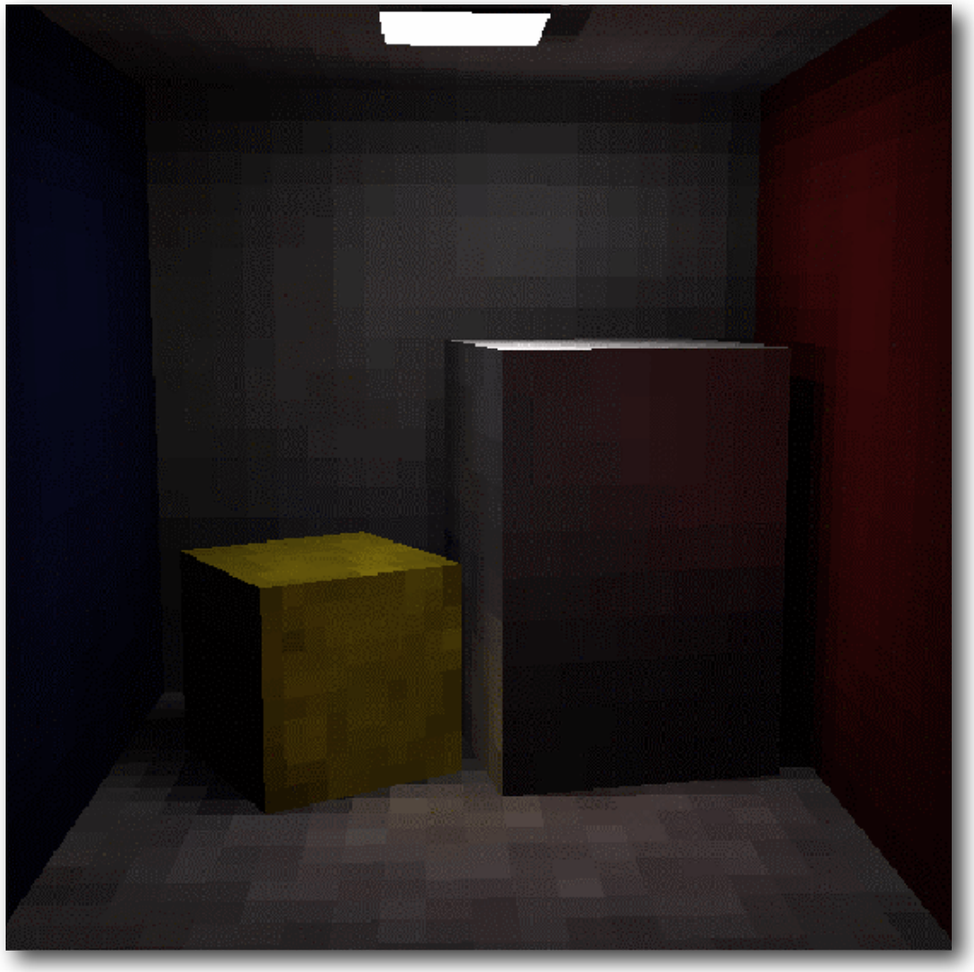


http://www.acm.org/jgt/papers/TeleaVanOverveld97/



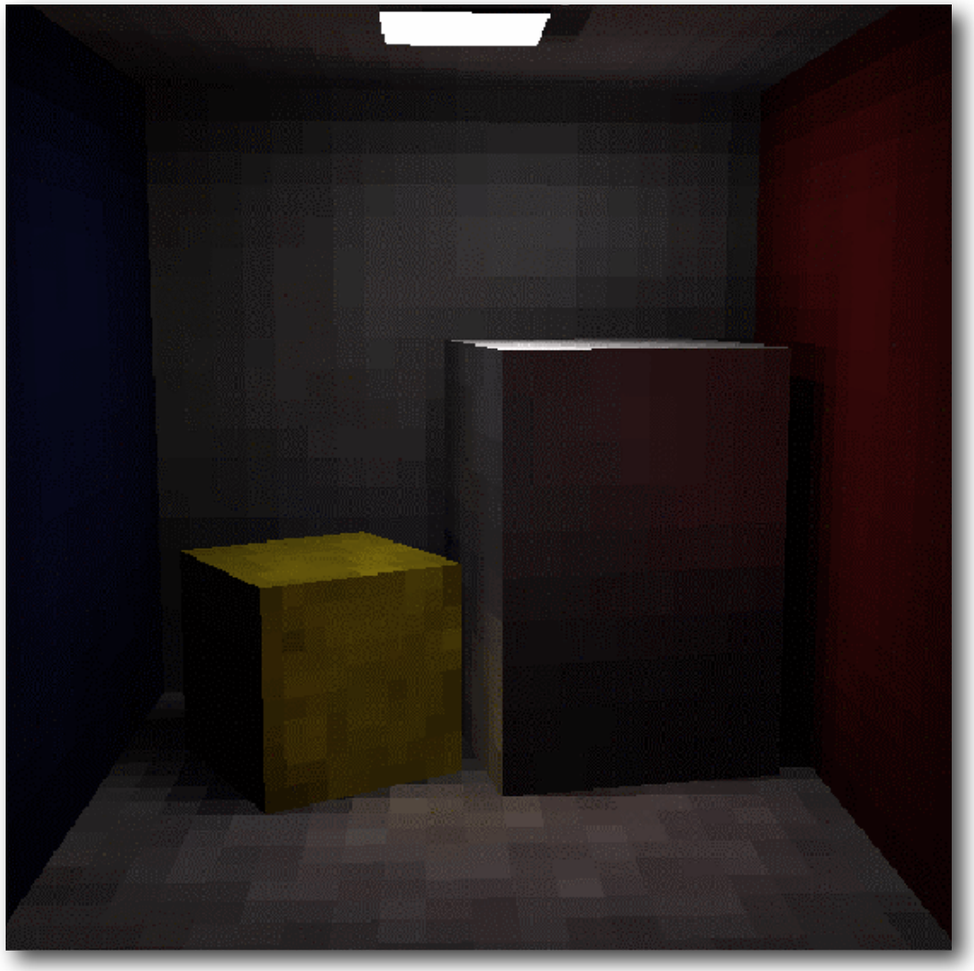
http://aig.cs.man.ac.uk/gallery/vrad.html

Problems



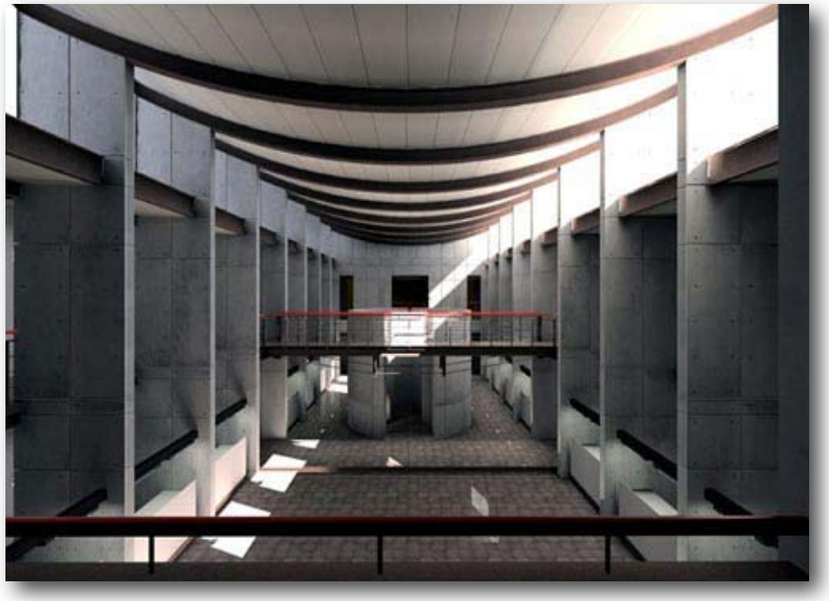
- **Blocky (Aliasing)**
  - ➟ **Adaptive Subdivision**
- **No Specular Effects**
  - ➟ **2-pass Algorithms**

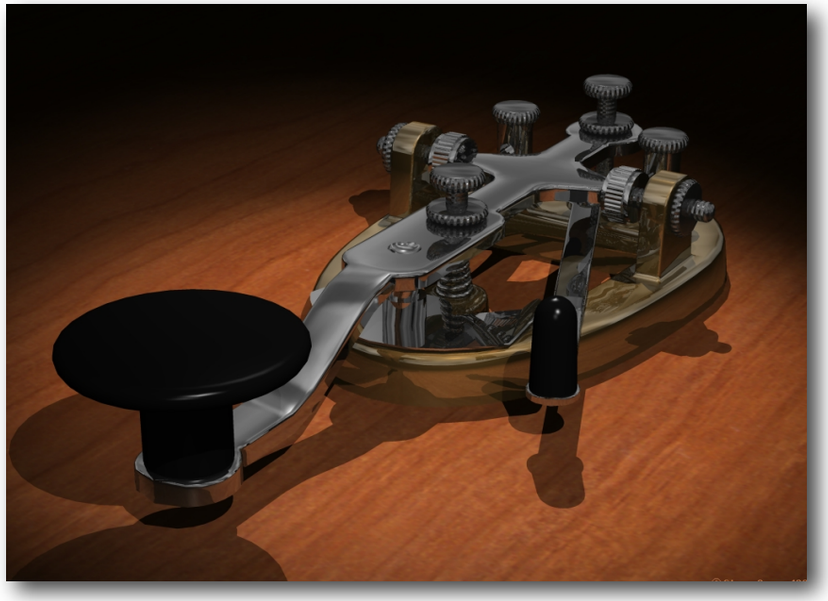
Problems



- Blocky (Aliasing)
  - ➟ Adaptive Subdivision
- No Specular Effects
  - ➟ 2-pass Algorithms

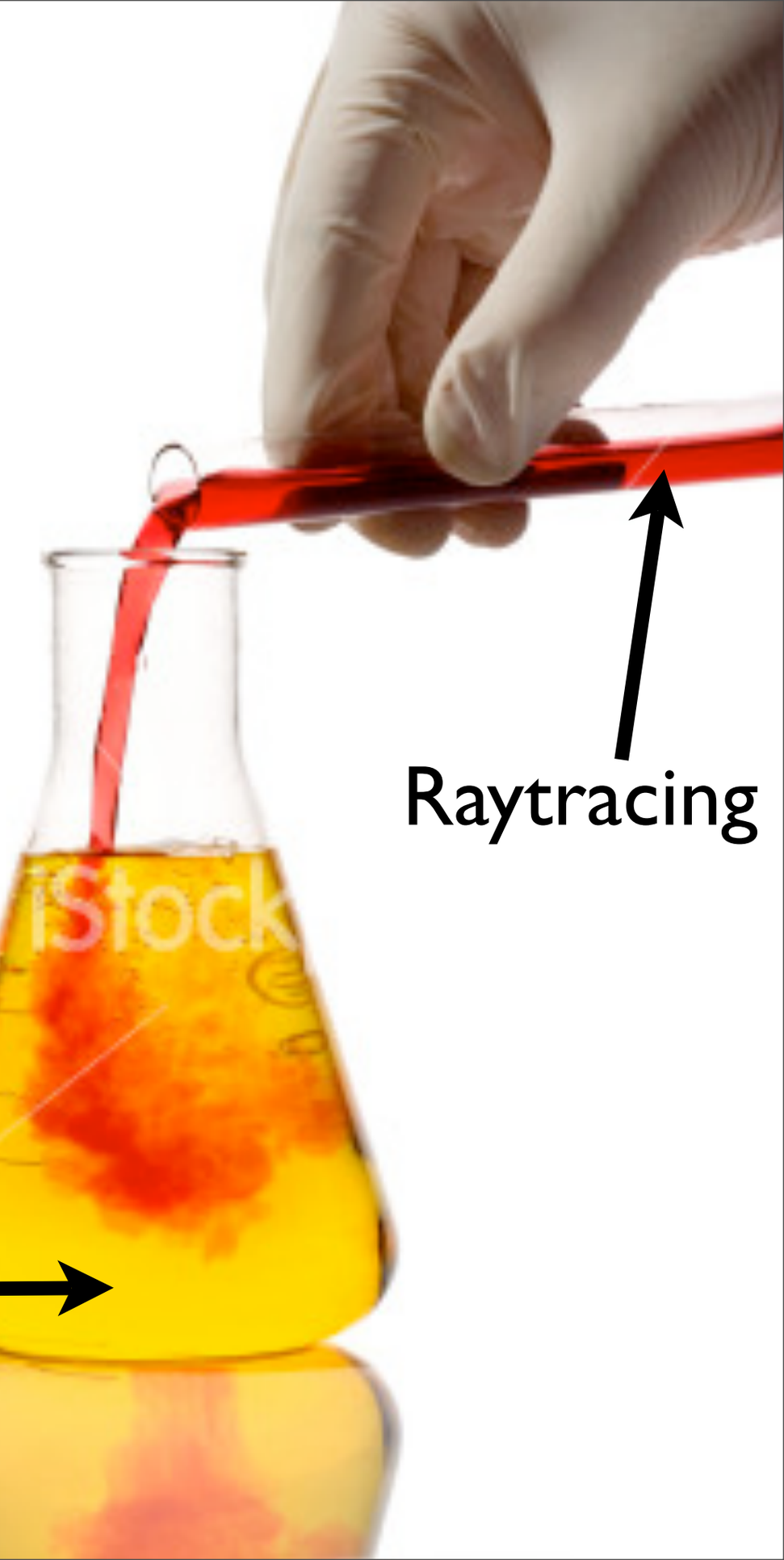
Radiosity vs. Raytracing





Radiosity **Raytracing**

Can we inject specular effects into radiosity?

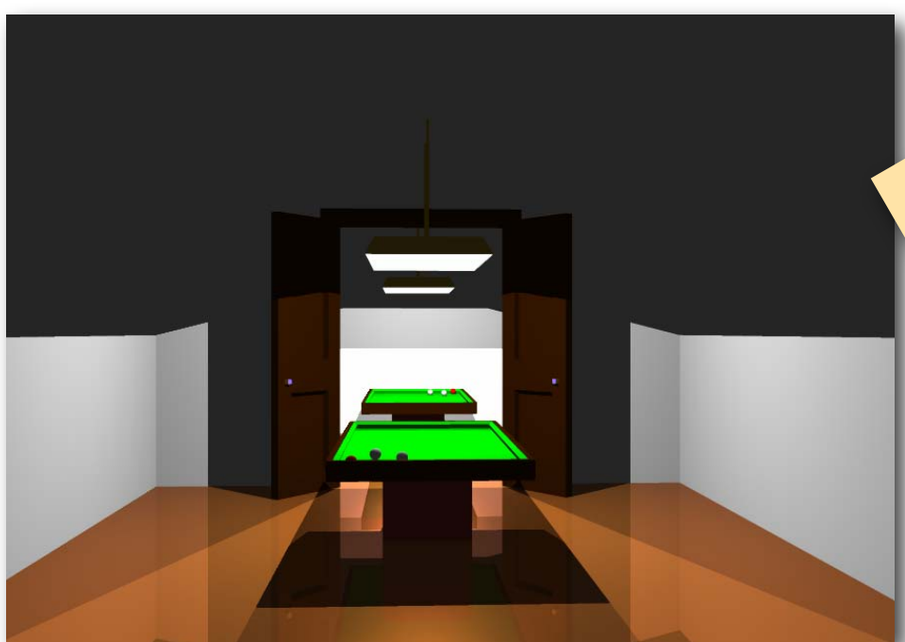


Raytracing

Radiosity

2-pass Algorithms

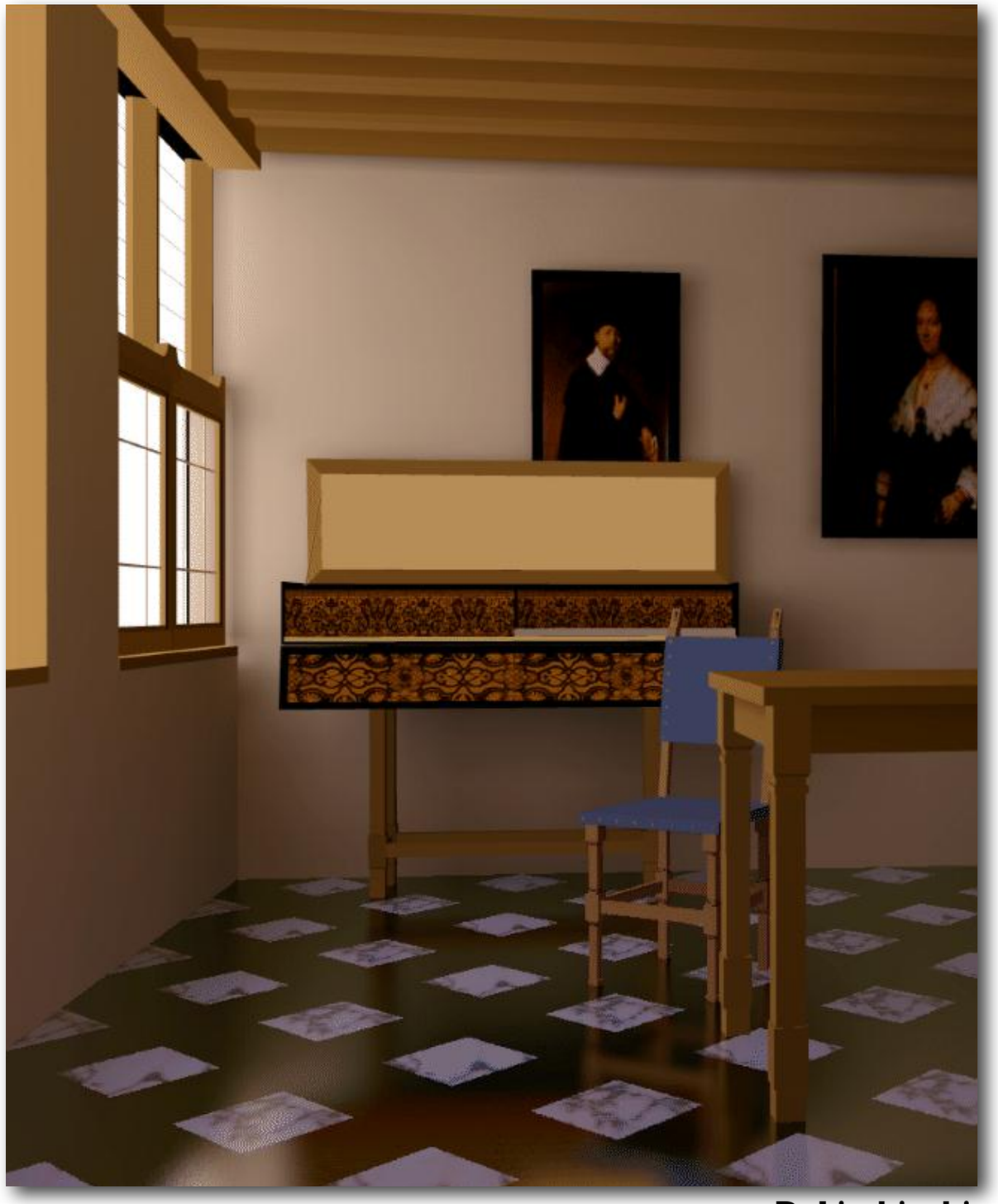
1st Pass: **Radiosity**



2nd Pass: **Raytracing**
(using Radiosity result)

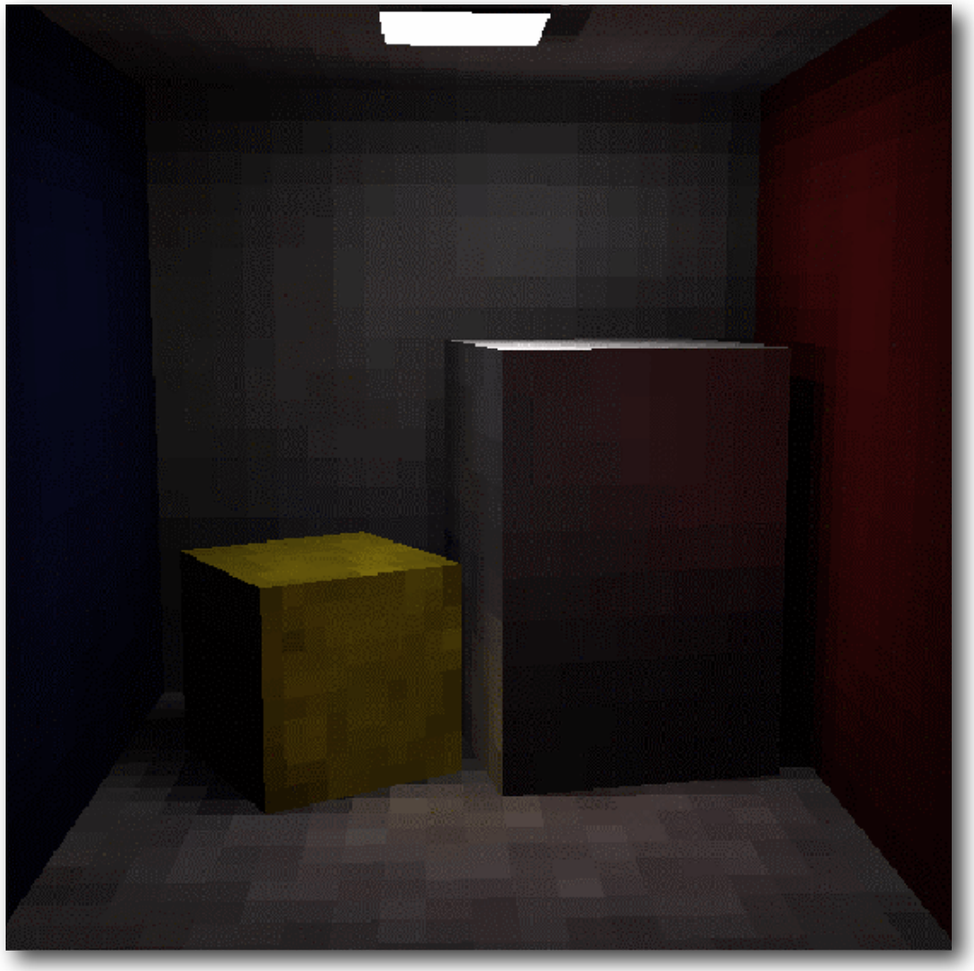
Result

Example



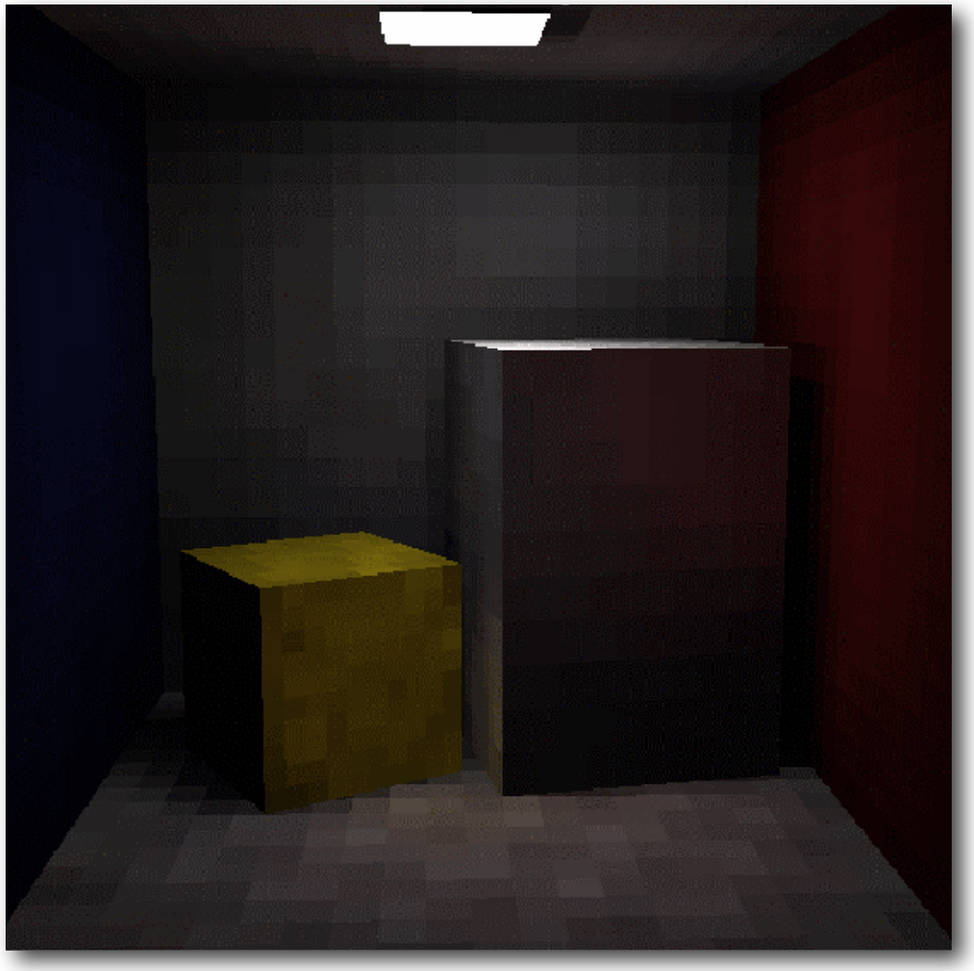
D. Lischinski

Problems



- Blocky (Aliasing)
  - ➟ Adaptive Subdivision
- No Specular Effects
  - ➟ 2-pass Algorithms

Problems



- **Blocky (Aliasing)**
  - ➟ **Adaptive Subdivision**
- **No Specular Effects**
  - ➟ **2-pass Algorithms**

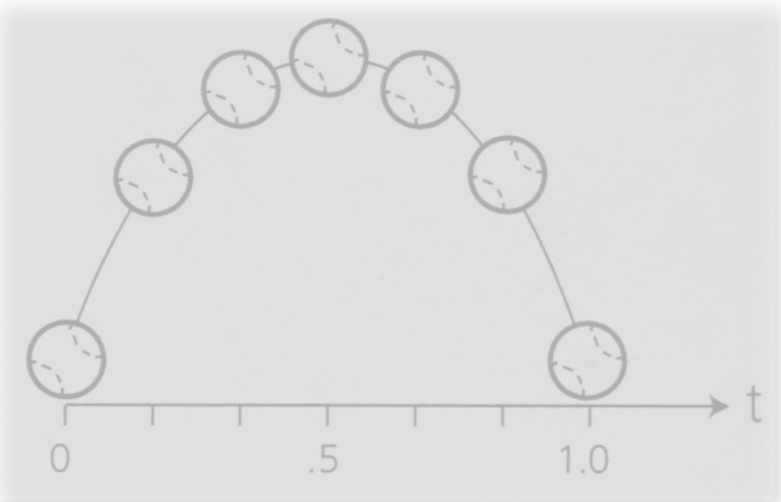
Overview

Announcements

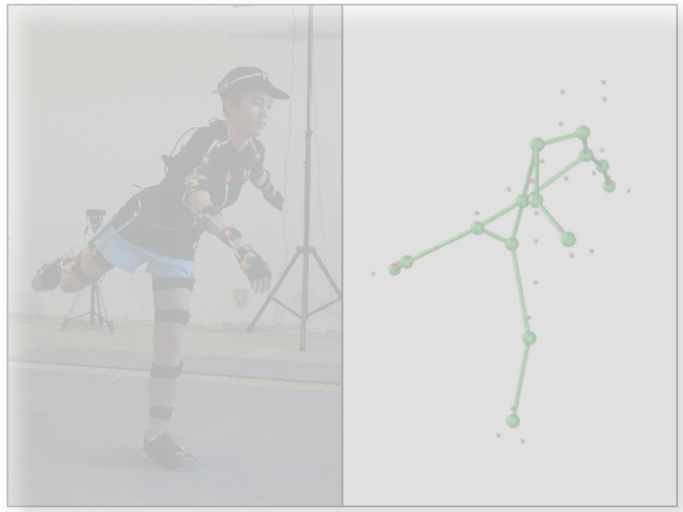
Radiosity

Animation Intro

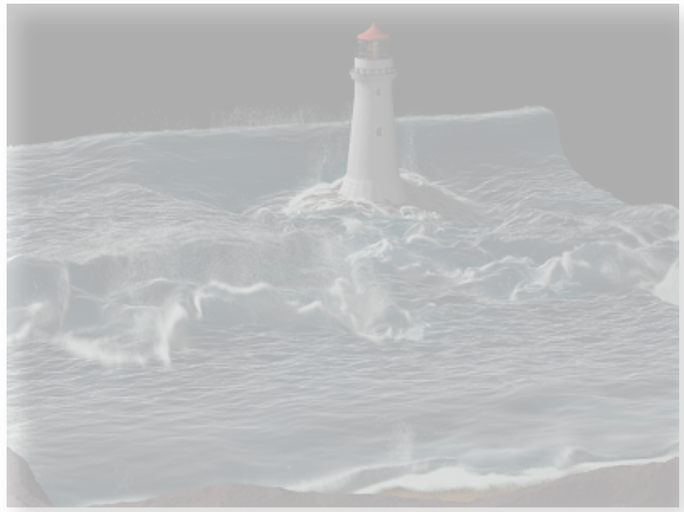
Cell Animation



Keyframing

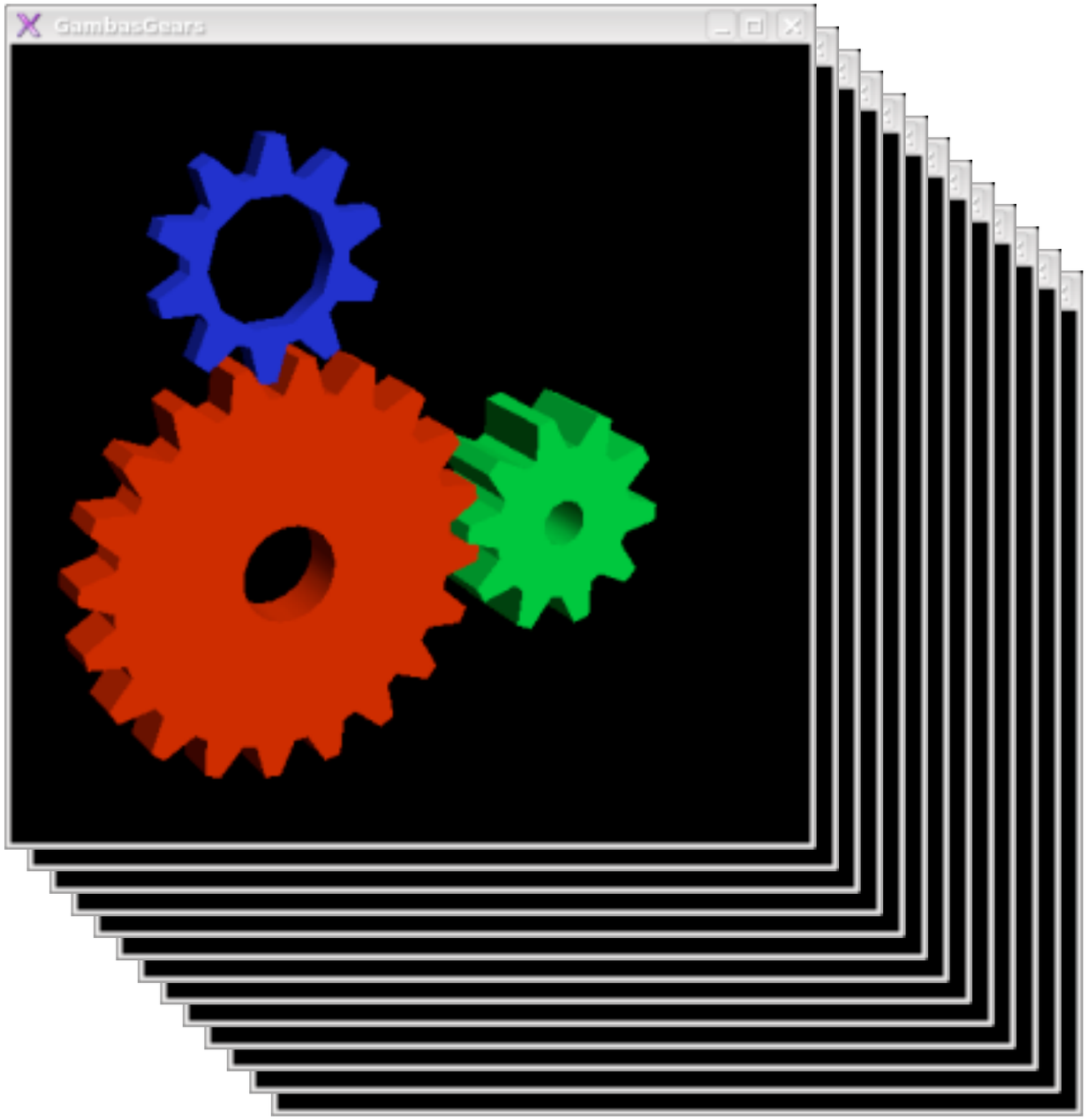


Data-driven Animation



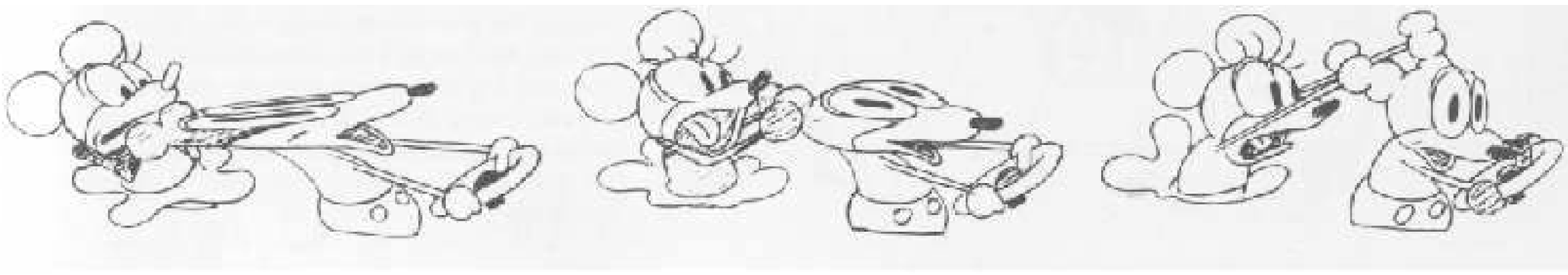
Physical Simulation

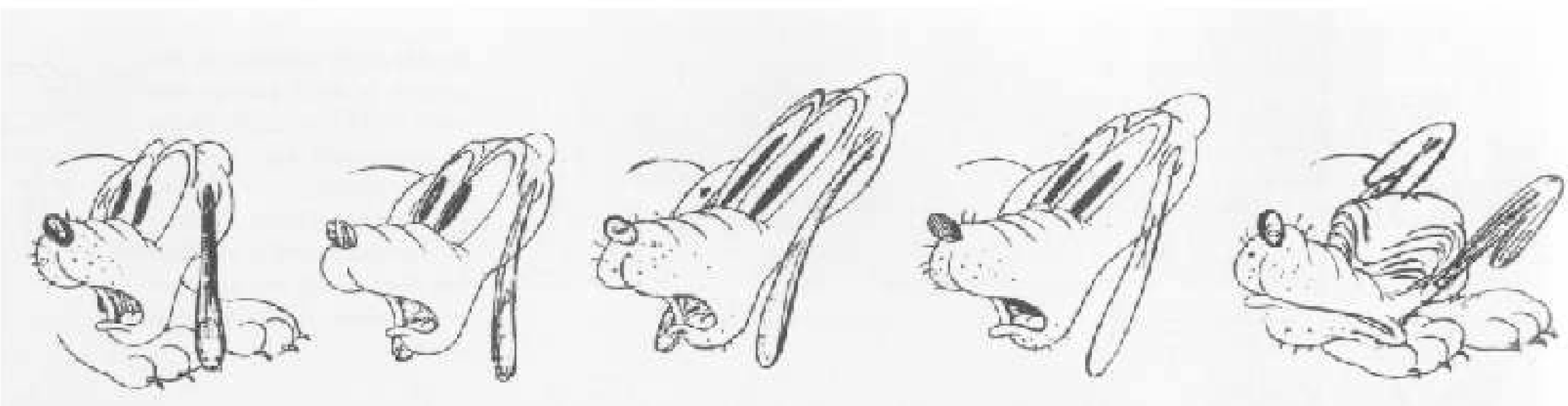
What is animation?



Animation = Graphics × Time

Cell Animation





Keyframing

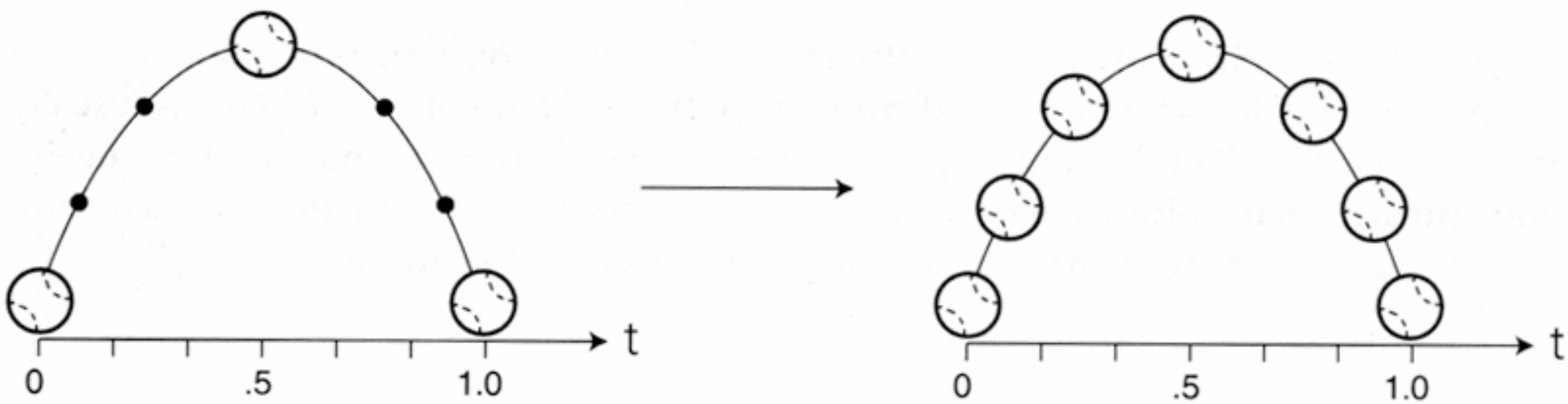
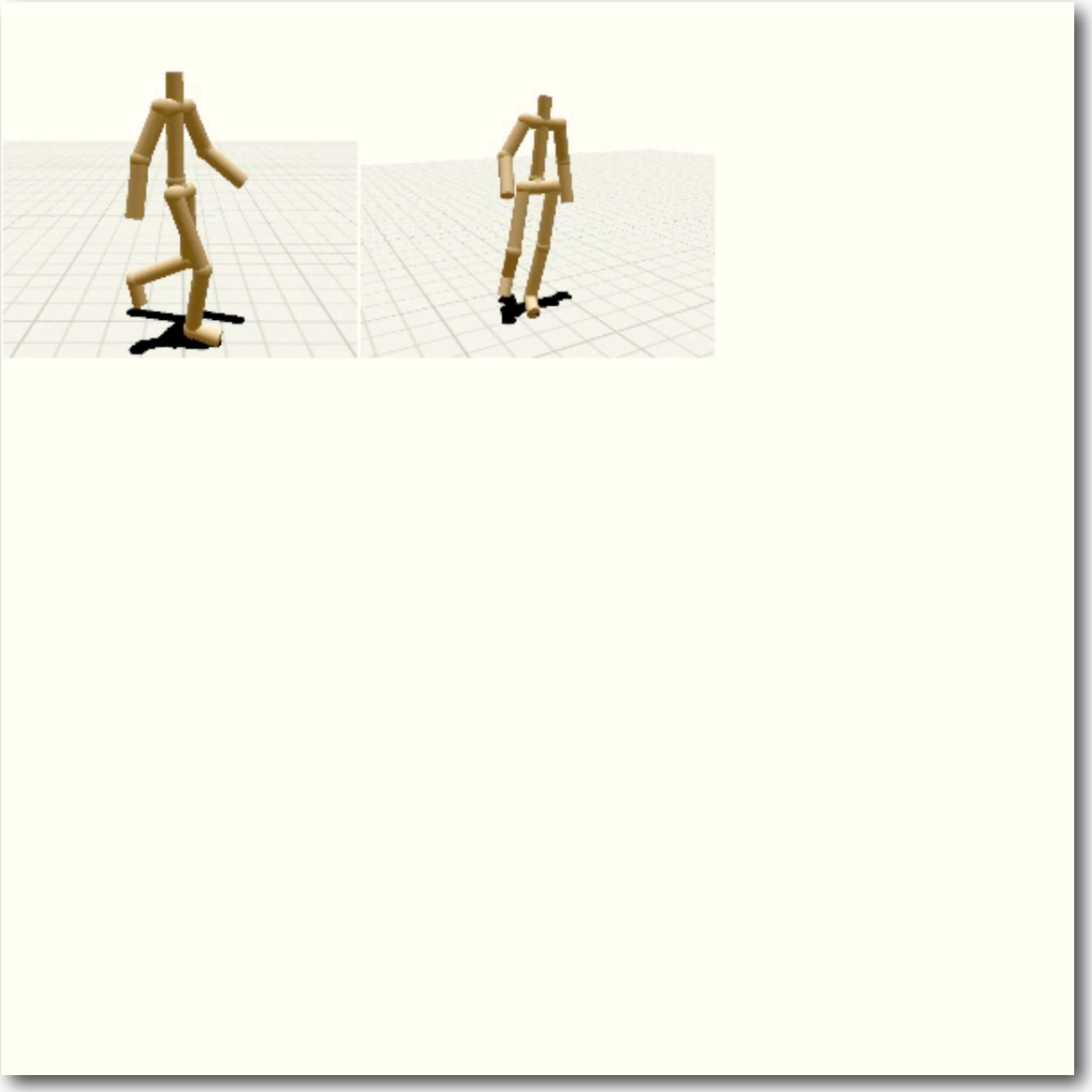
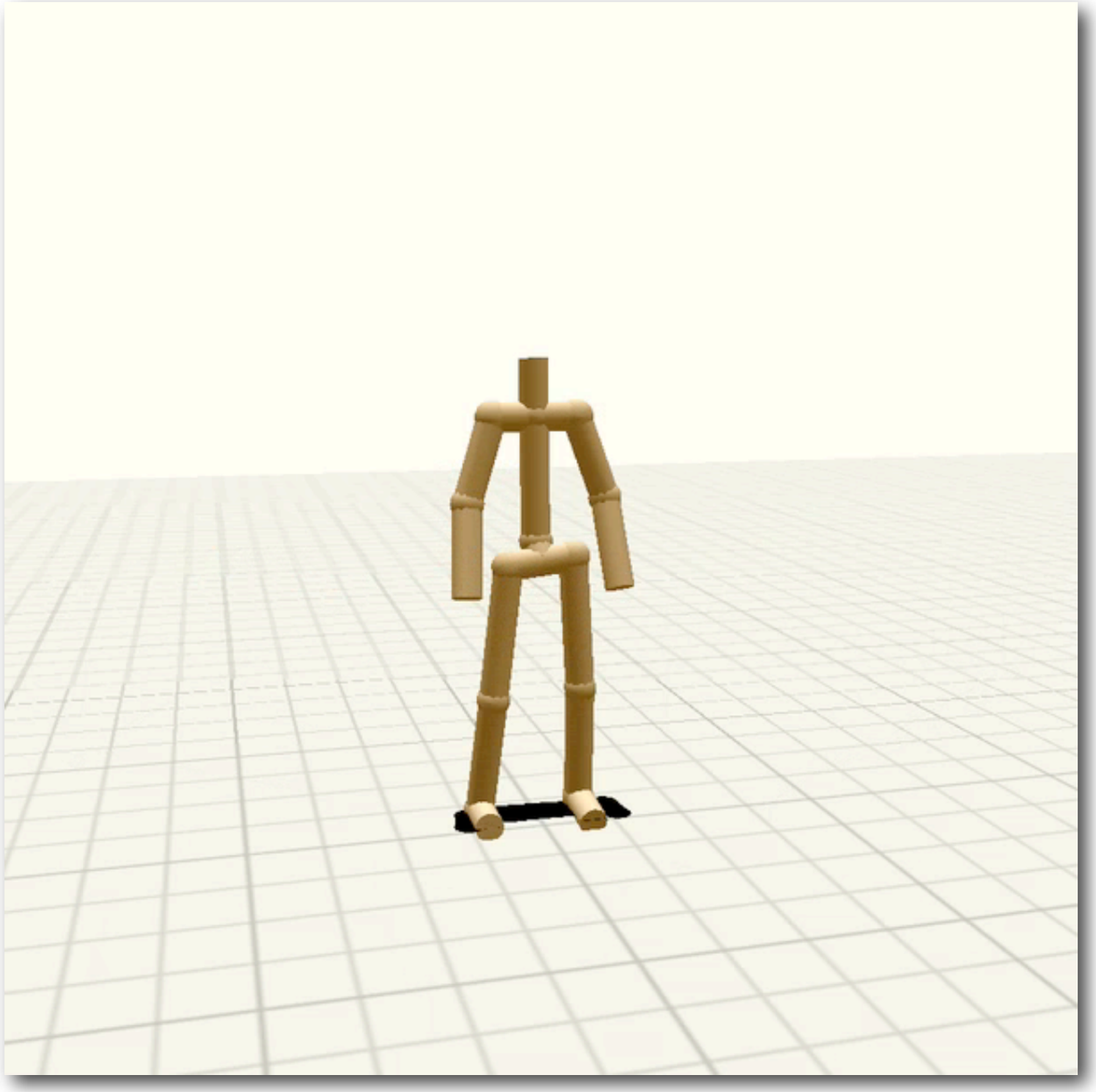


Figure 10.9 **Inbetweening with nonlinear interpolation.** Nonlinear interpolation can create equally spaced inbetween frames along curved paths. The ball still moves at a constant speed. (Note that the three keyframes used here and in Fig. 10.10 are the same as in Fig. 10.4.)

Data-driven Animation





Physics-based Animation

What do you need to know?

- No project on this material.
- No need to memorize the mathematics in this lecture.
- Need to understand the basic ideas.
- Need to understand mathematics of subsequent lectures:
  - PDEs / Particle Systems

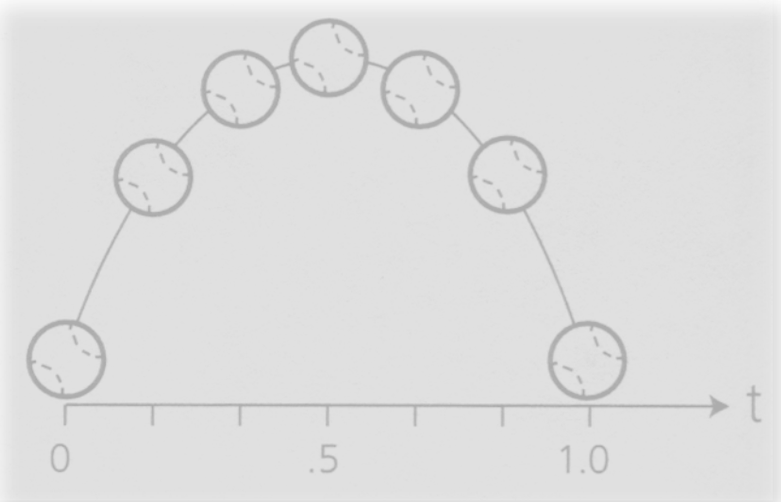
Overview

Announcements

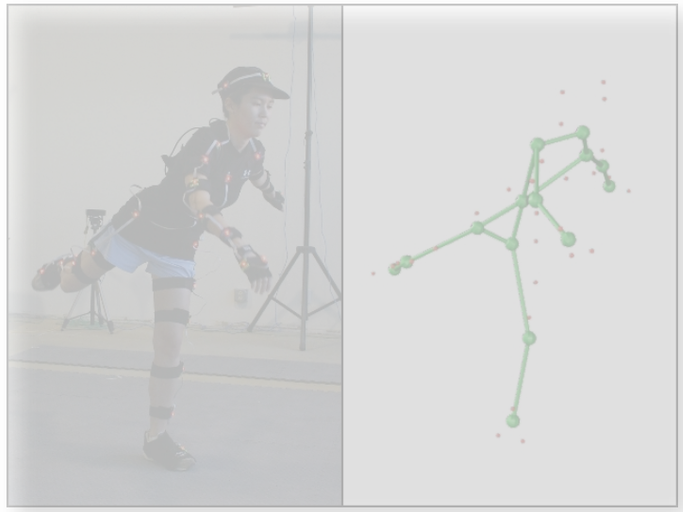
Radiosity

Animation Intro

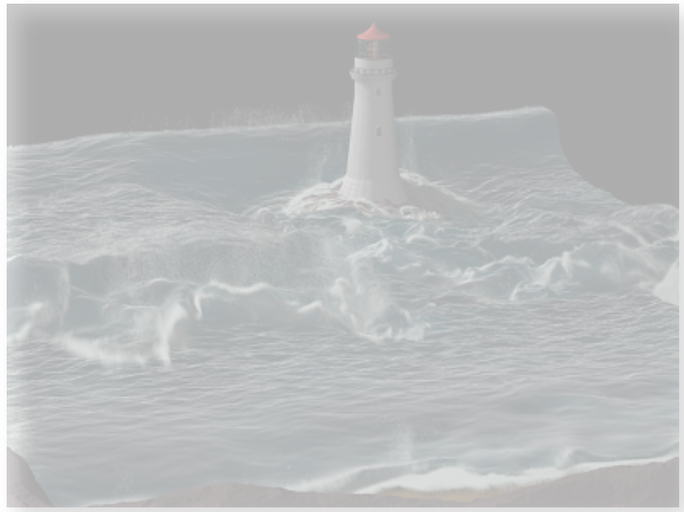
Cell Animation



Keyframing



Data-driven Animation



Physical Simulation

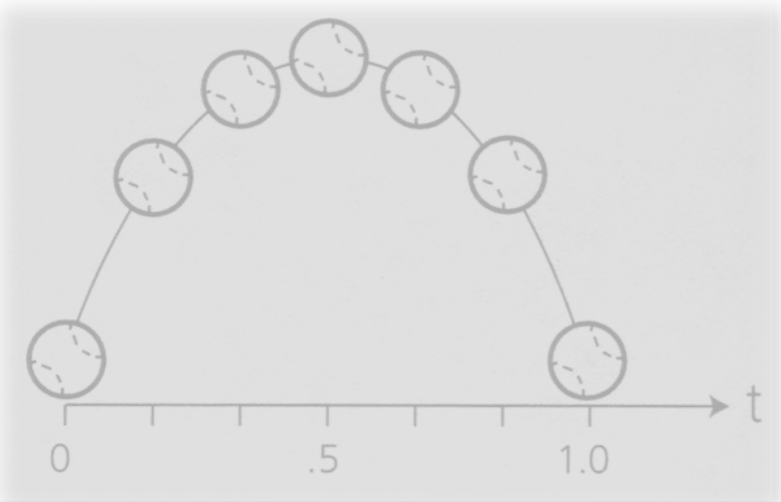
Overview

Announcements

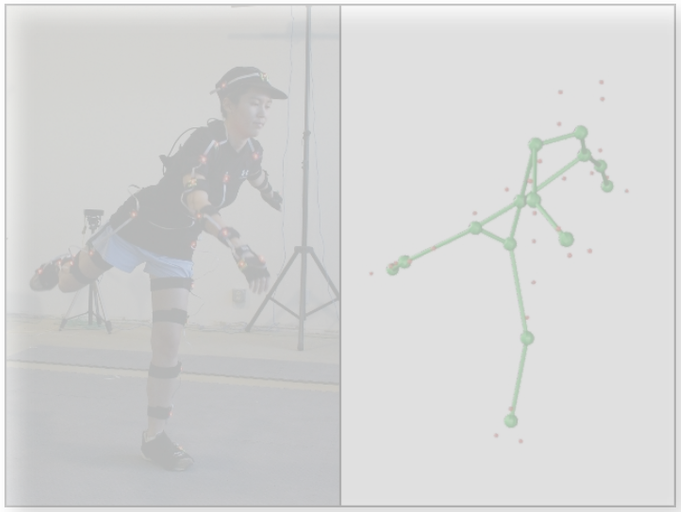
Radiosity

Animation Intro

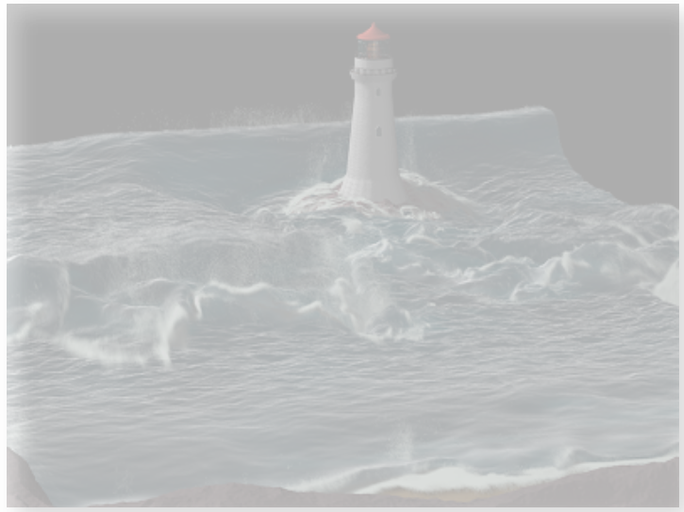
Cell Animation



Keyframing



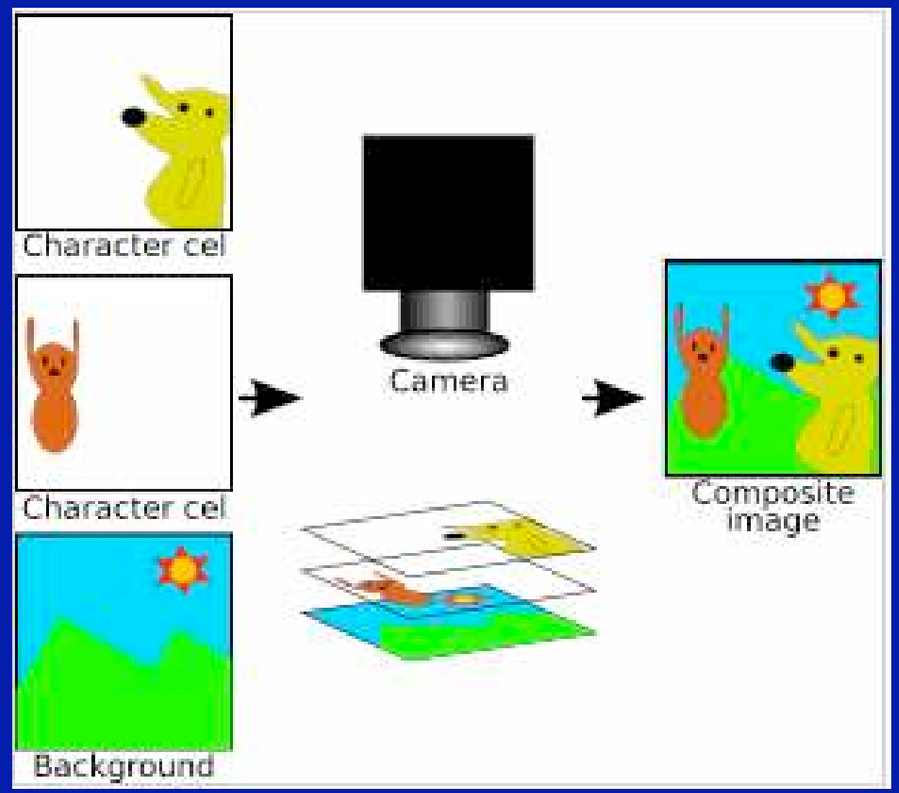
Data-driven Animation



Physical Simulation

Traditional Cel Animation

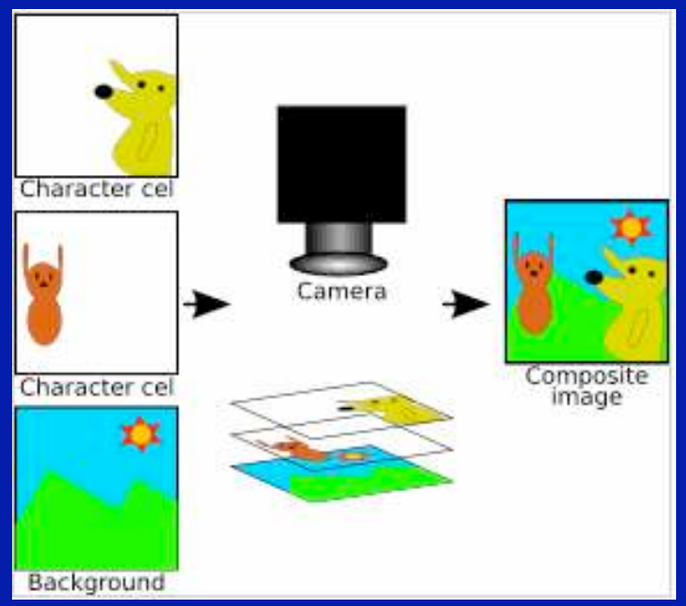
- Each frame is drawn by hand



- Film runs at 24 frames per second (fps)
  - That's 1440 pictures to draw per minute
- Artistic issues:
  - Artistic vision has to be converted into a sequence of still frames
  - Not enough to get the stills right--must look right at full speed
    - » Hard to "see" the motion given the stills
    - » Hard to "see" the motion at the wrong frame rate

Traditional Animation: The Process

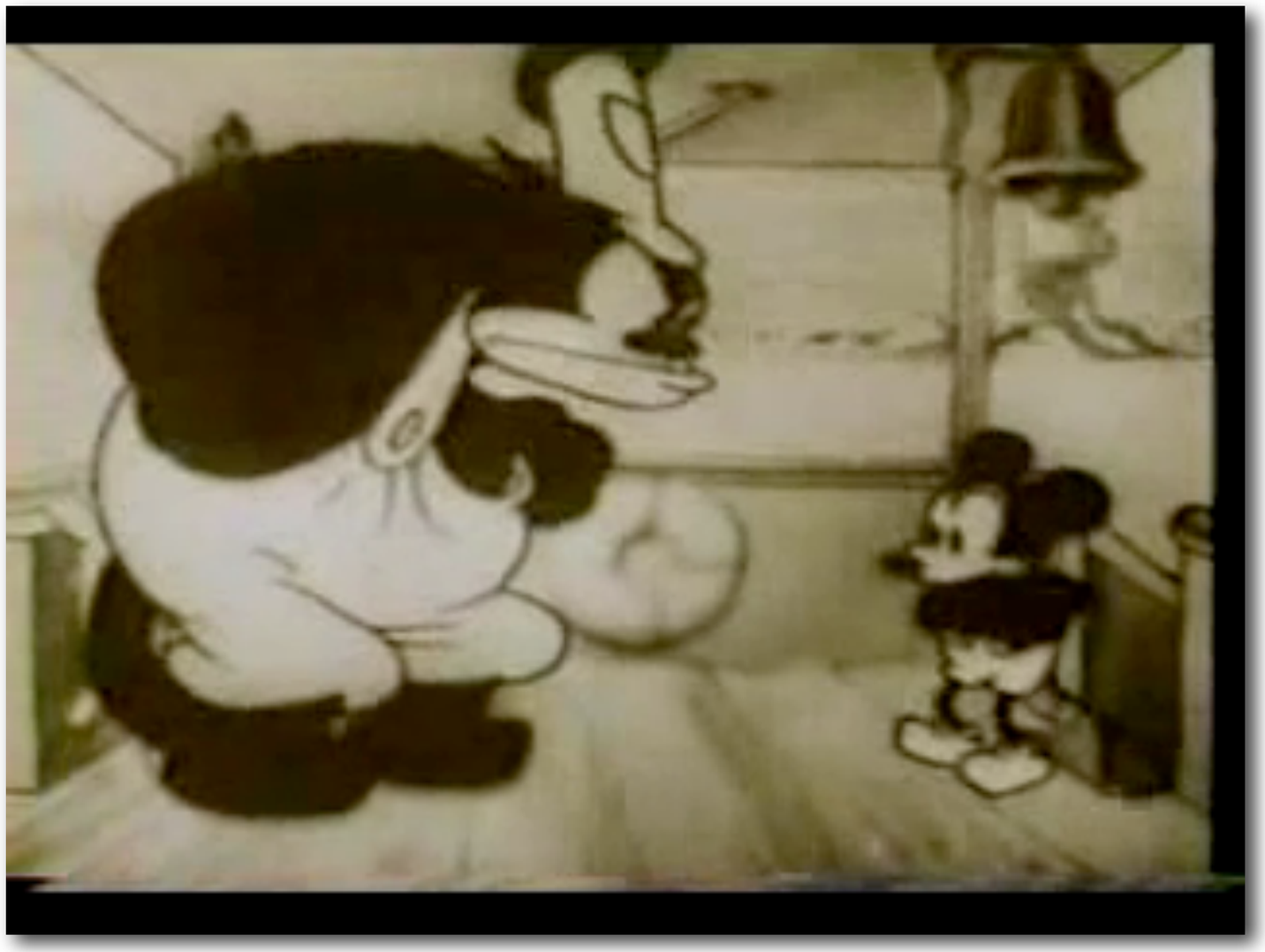
- Key Frames
  - Draw a few important frames in pencil
    - » beginning of jump, end of jump and a frame in the air
- Inbetweens
  - Draw the rest of the frames
- Painting
  - Redraw onto clear sheet of plastic called a *cel*, color them in



- Use one layer for background, one for object
  - Draw each separately
  - Stack them together on a copy stand
  - Transfer onto film by taking a photograph of the stack

- Can have multiple animators working simultaneously on different layers, avoid re-drawing and flickering

Example



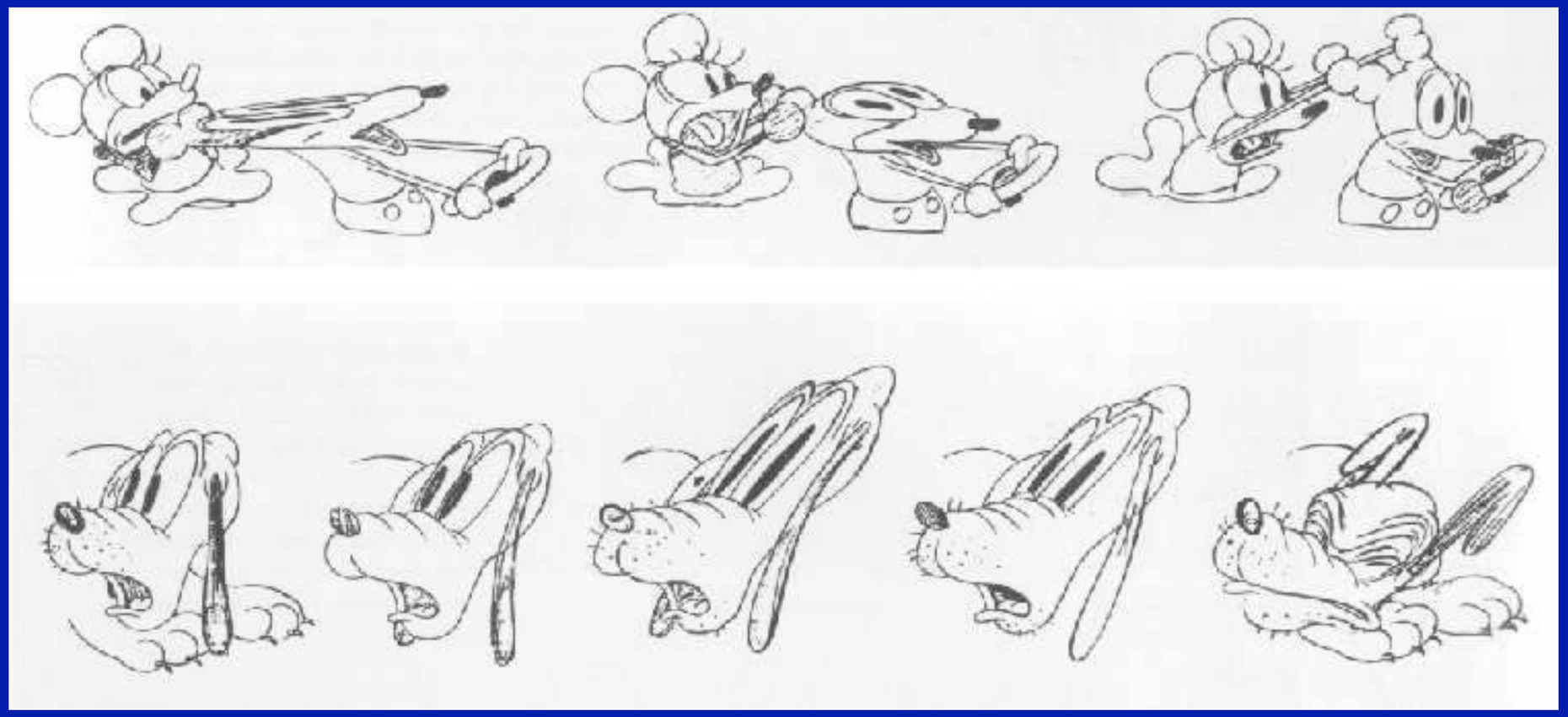
Principles of Traditional Animation
[Lasseter, SIGGRAPH 1987]

- Stylistic conventions followed by Disney's animators and others

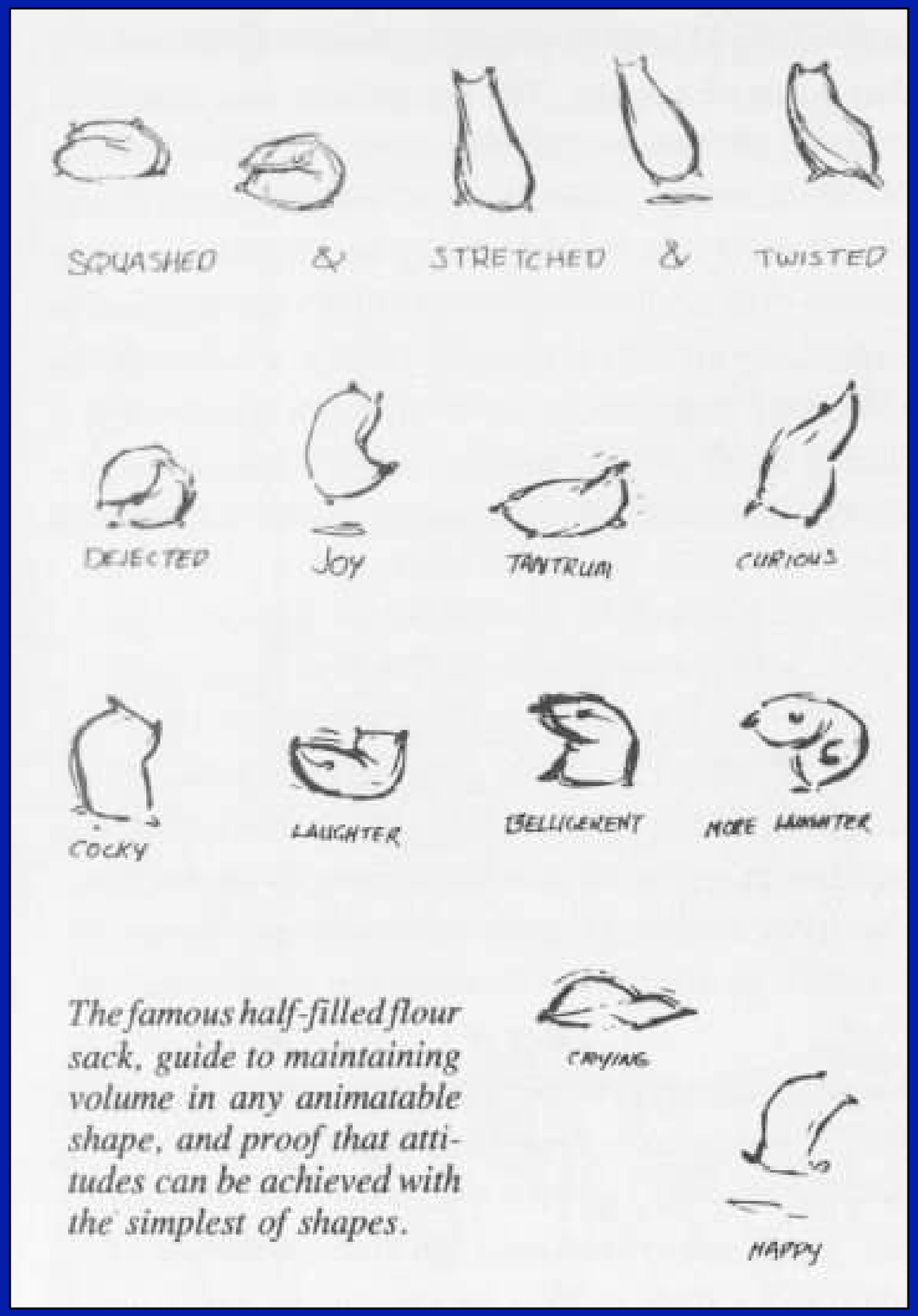
- From experience built up over many years
  - Squash and stretch -- use distortions to convey flexibility
  - Timing -- speed conveys mass, personality
  - Anticipation -- prepare the audience for an action
  - Followthrough and overlapping action -- continuity with next action
  - Slow in and out -- speed of transitions conveys subtleties
  - Arcs -- motion is usually curved
  - Exaggeration -- emphasize emotional content
  - Secondary Action -- motion occurring as a consequence
  - Appeal -- audience must enjoy watching it

Squash and Stretch

Use distortions to convey flexibility

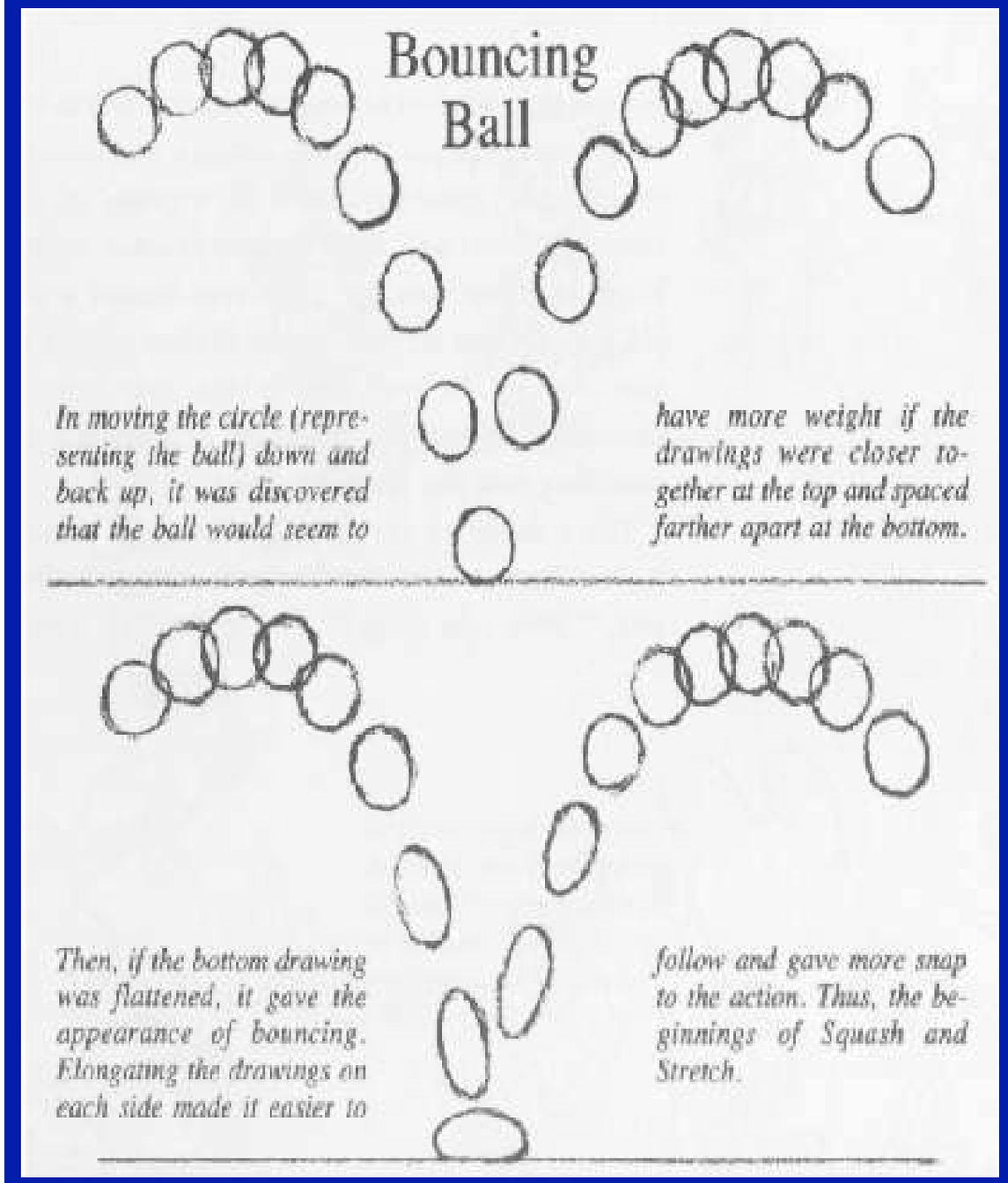


Principles of Traditional Animation



Squash and Stretch

Use distortions to convey flexibility

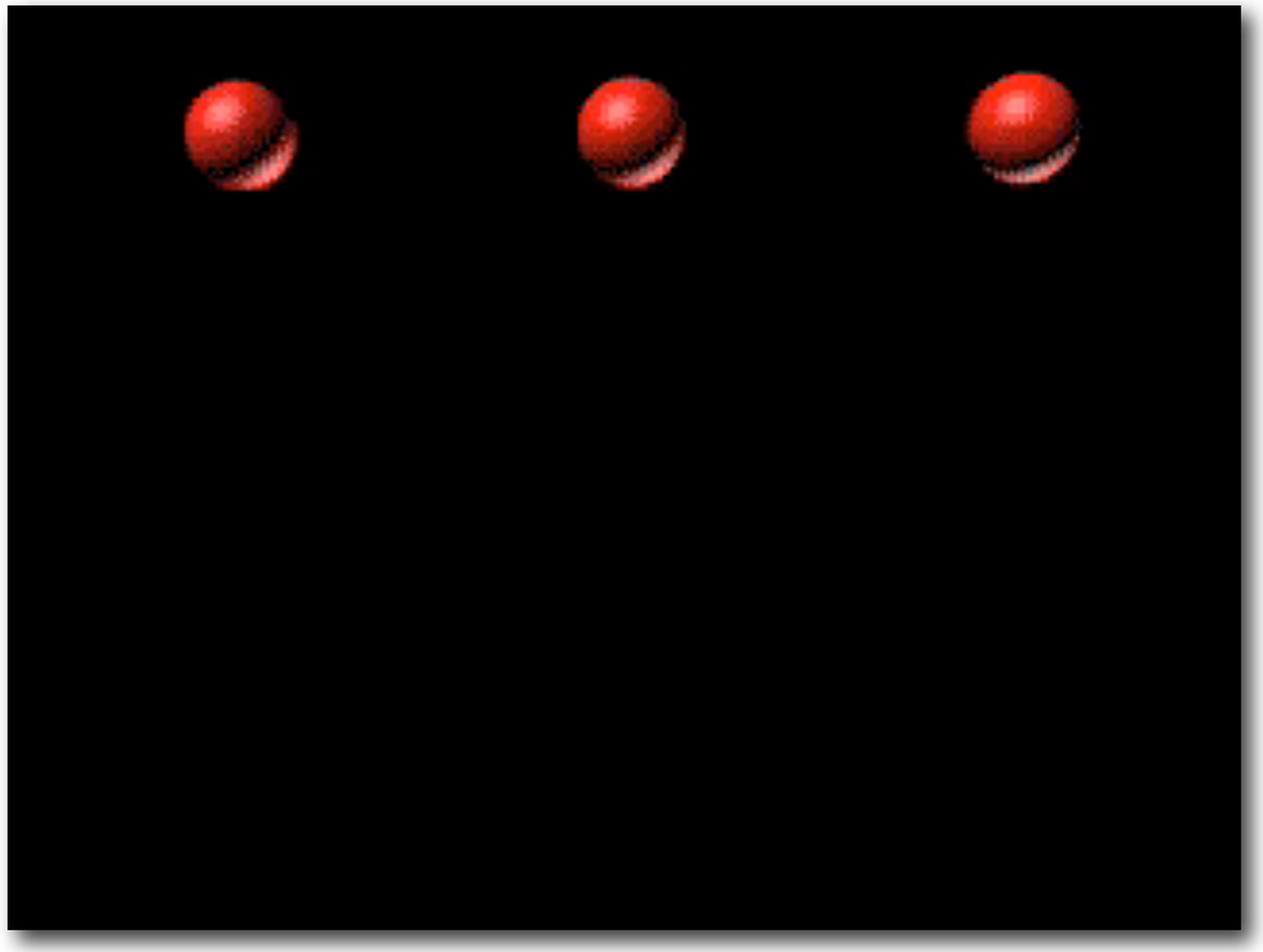


Defines the rigidity of the material

Gives the sense that the object is made out of a soft, pliable material.

Elongating the drawings before and after the bounce increases the sense of speed, makes it easier to follow and gives more snap to the action.

Example



The ball on the left moves at a constant speed with no squash/stretch.
The ball in the center does slow in and out with a squash/stretch.
The ball on the right moves at a constant speed with squash/stretch.

Timing & Motion

Speed conveys mass, personality

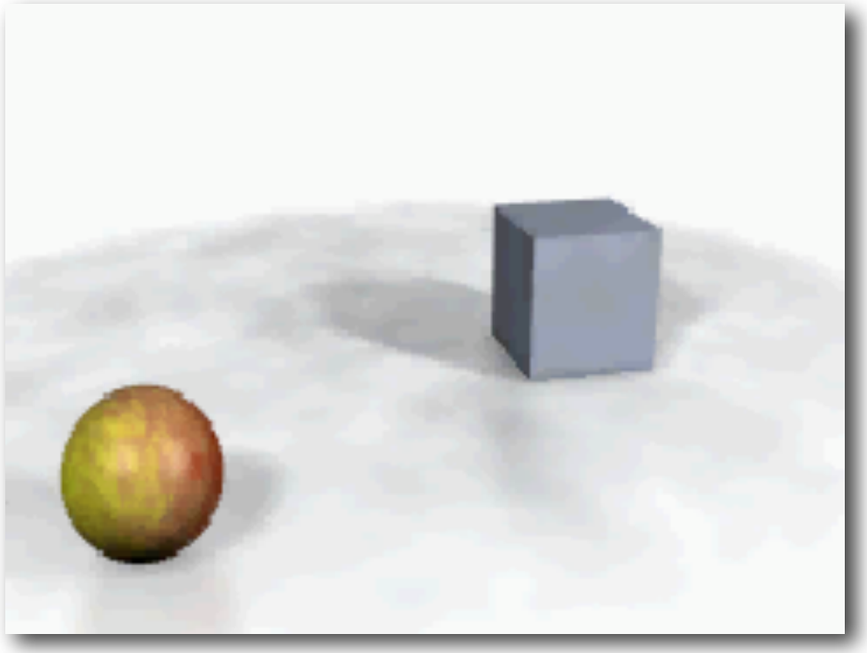
A heavier object takes a greater force and a longer time to accelerate and decelerate

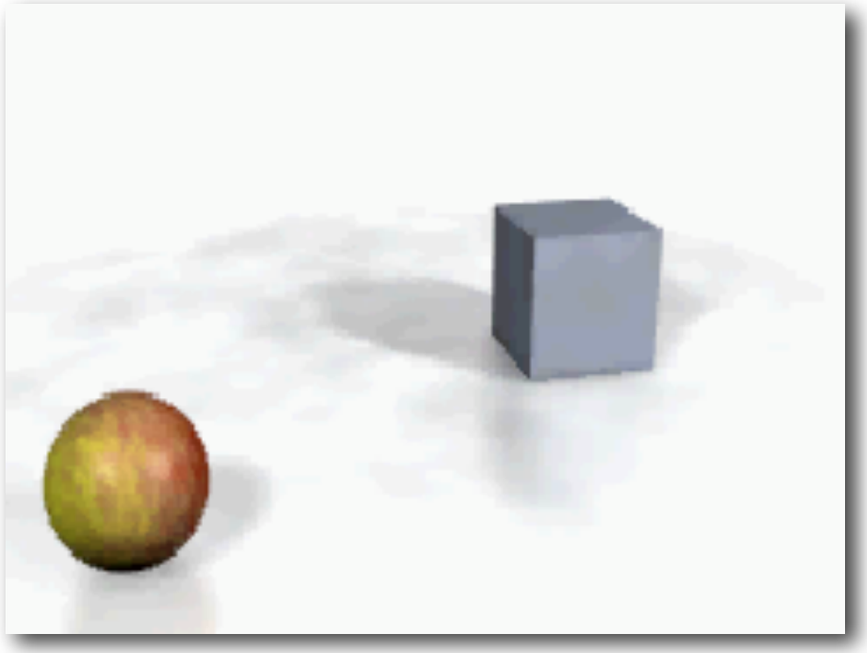
A larger object moves more slowly than a smaller object and has greater inertia

Motion also can give the illusion of weight

For example, consider a ball hitting a box

http://www.siggraph.org/education/materials/HyperGraph/animation/character_animation/principles/timing.htm





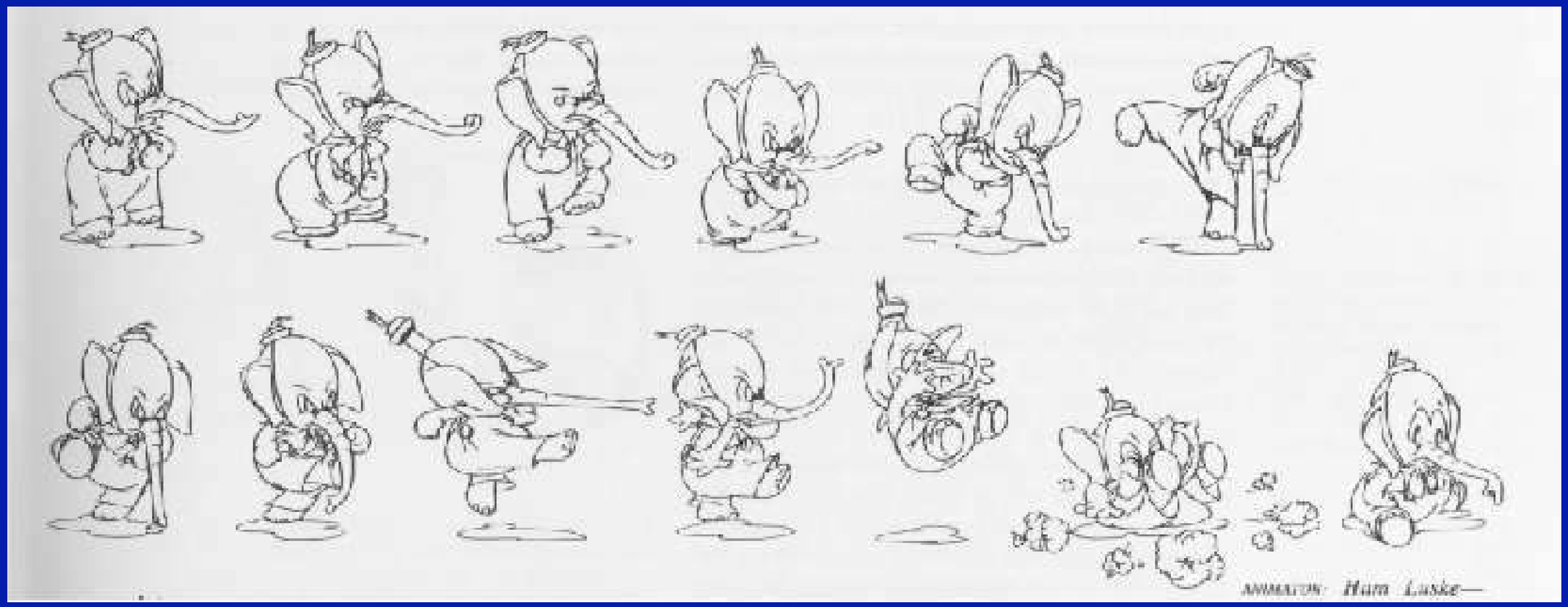
Anticipation

Prepare the audience for an action

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

Follow Through and Overlapping Action

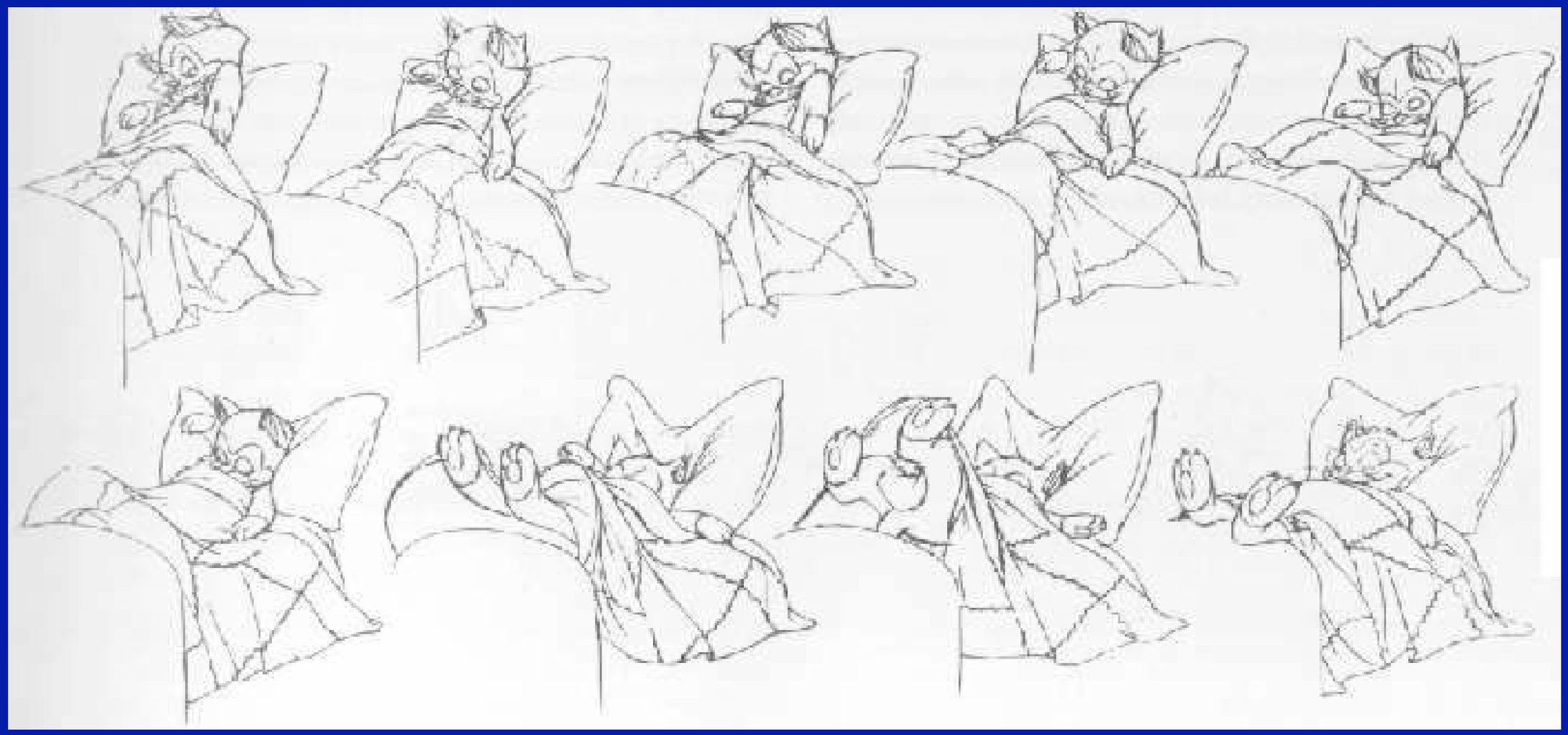
The termination of an action and establishing its relationship to the next action



Audience likes to see resolution of action
Discontinuities are unsettling

Secondary Action

Motion occurring as a consequence



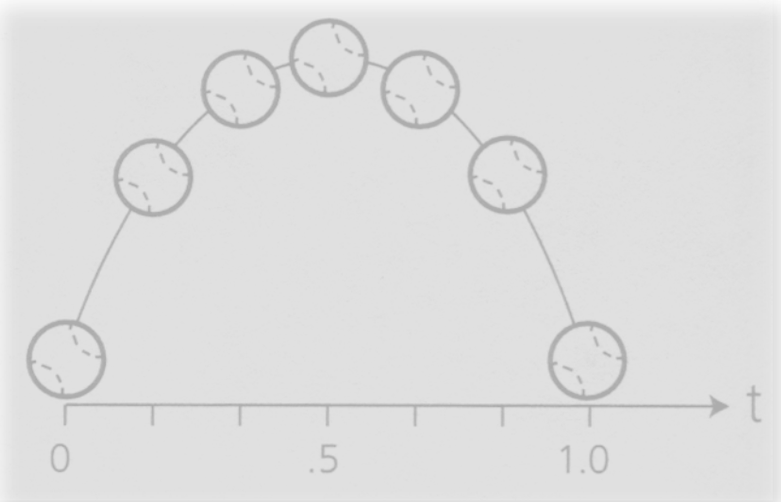
Overview

Announcements

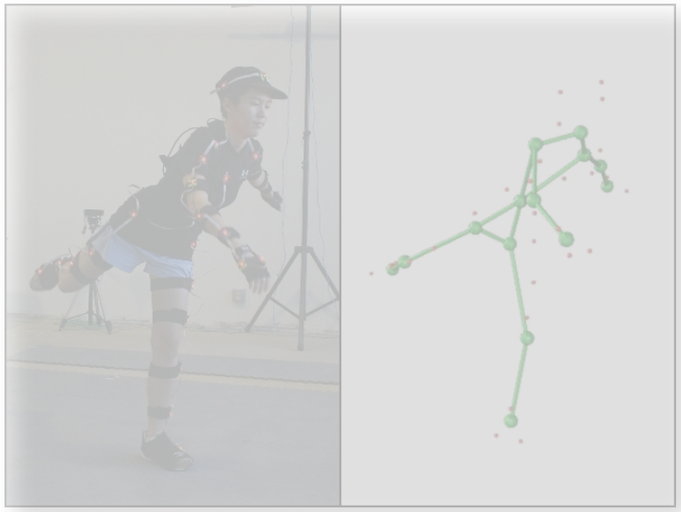
Radiosity

Animation Intro

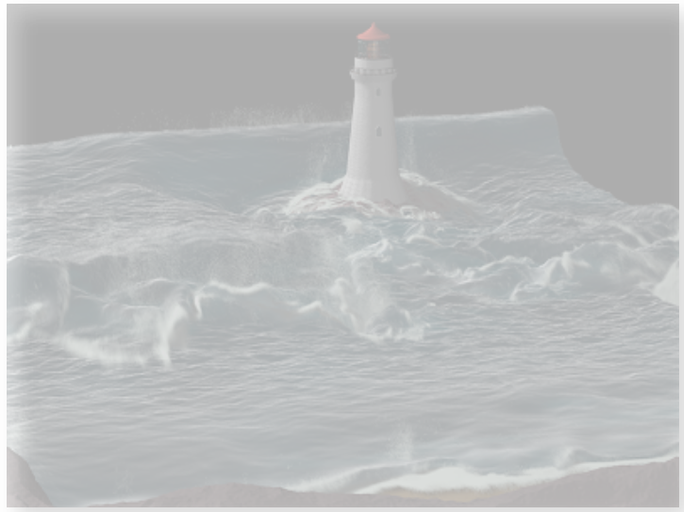
Cell Animation



Keyframing



Data-driven Animation



Physical Simulation

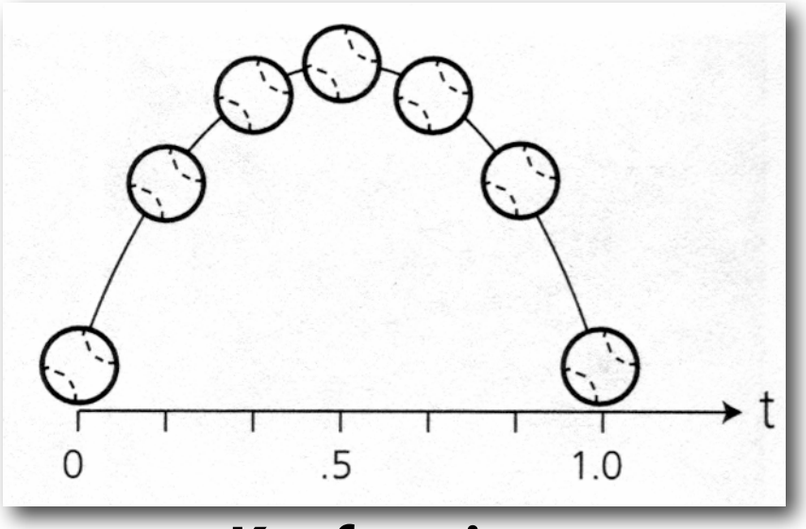
Overview

Announcements

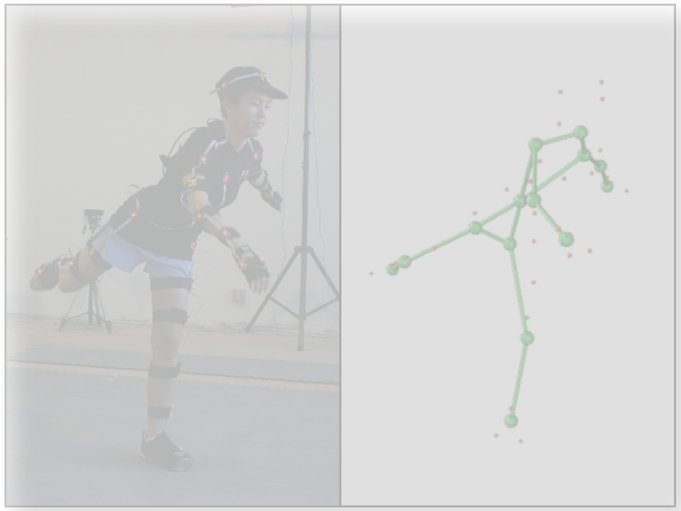
Radiosity

Animation Intro

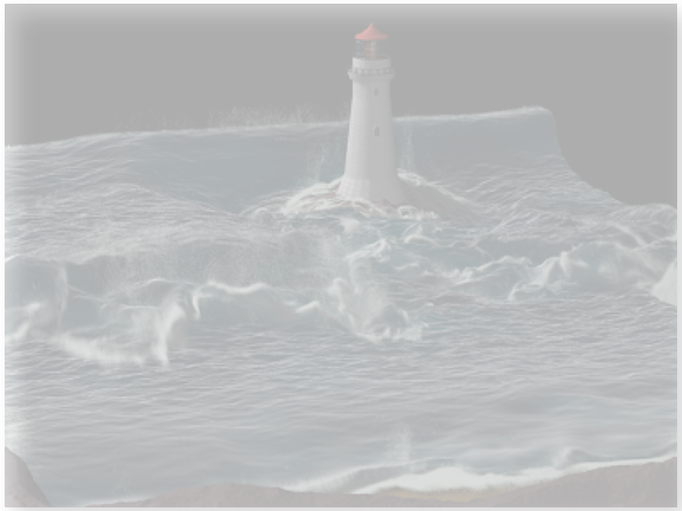
Cell Animation



Keyframing



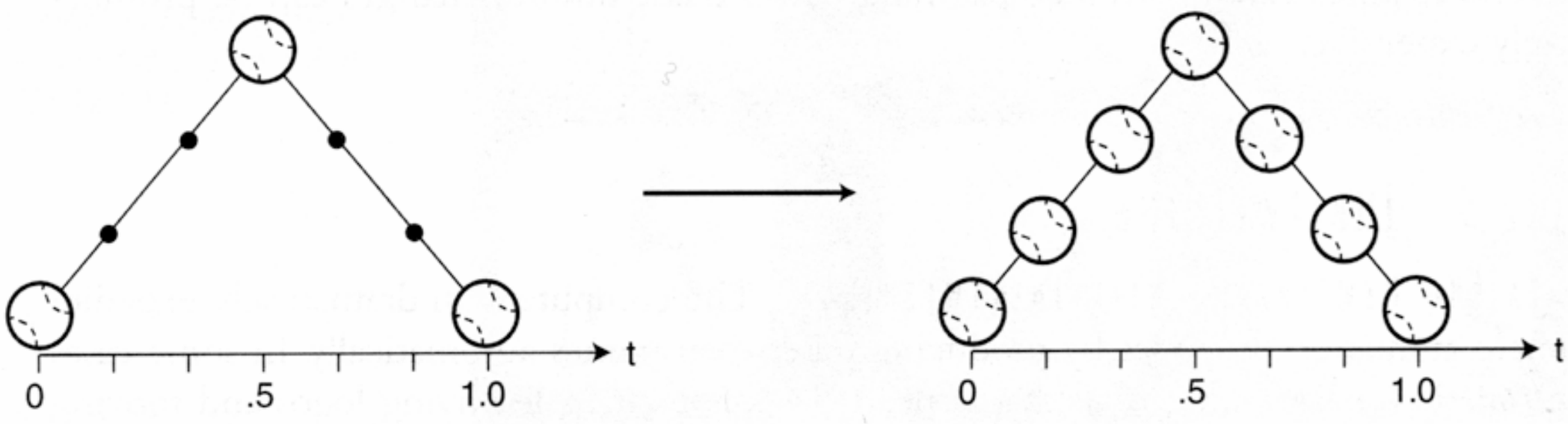
Data-driven Animation



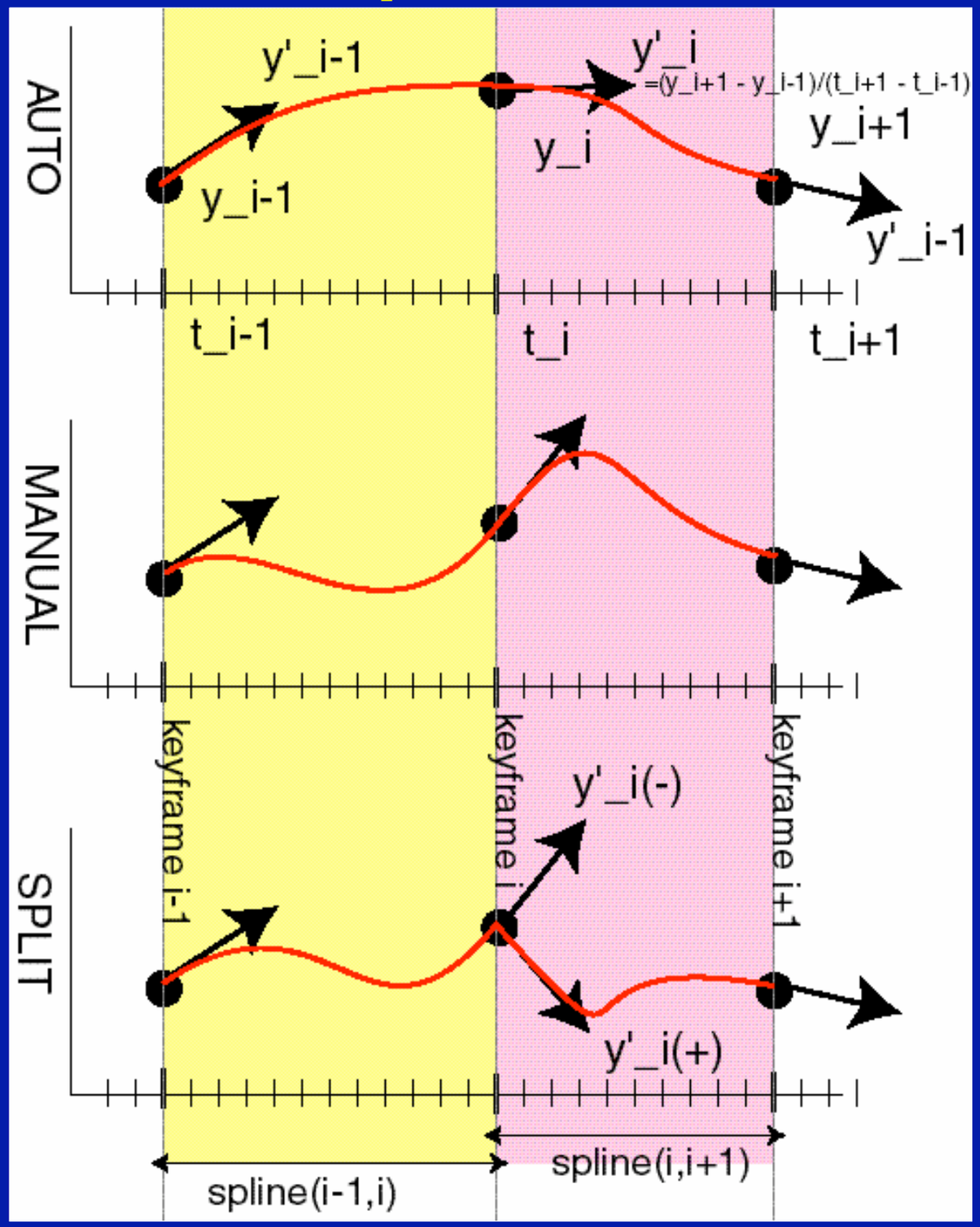
Physical Simulation

Keyframing Basics

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).



How Do You Interpolate Between Keys?



Keyframing Basics

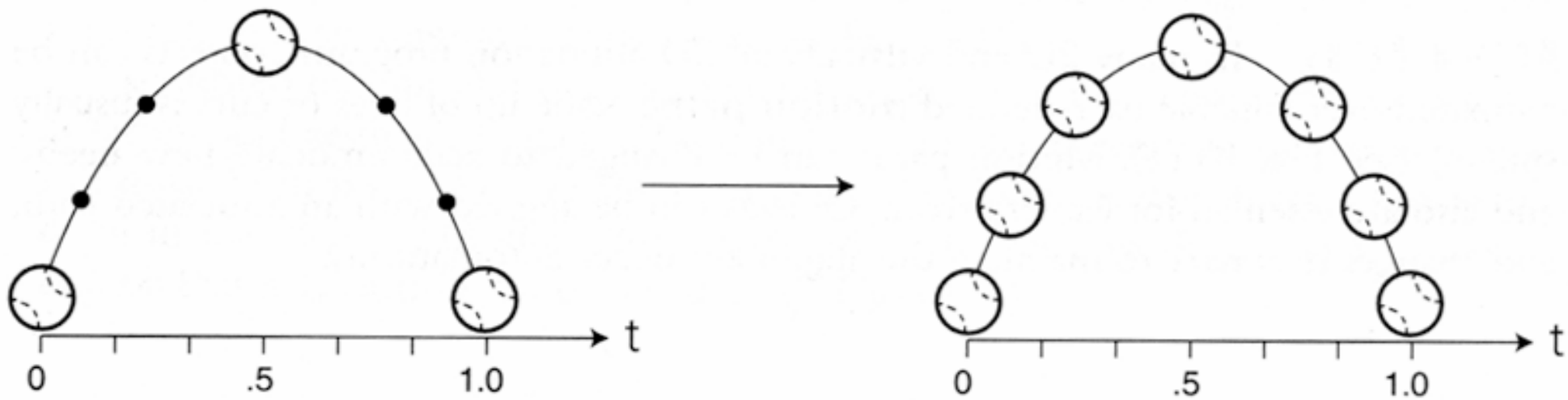


Figure 10.9 **Inbetweening with nonlinear interpolation.** Nonlinear interpolation can create equally spaced inbetween frames along curved paths. The ball still moves at a constant speed. (Note that the three keyframes used here and in Fig. 10.10 are the same as in Fig. 10.4.)

Keyframing Basics

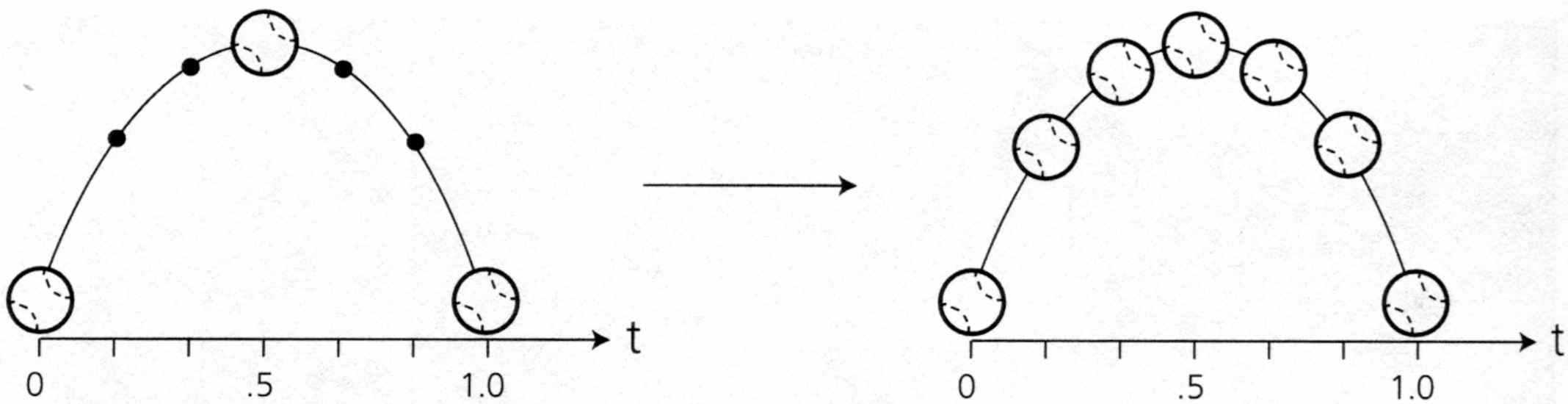
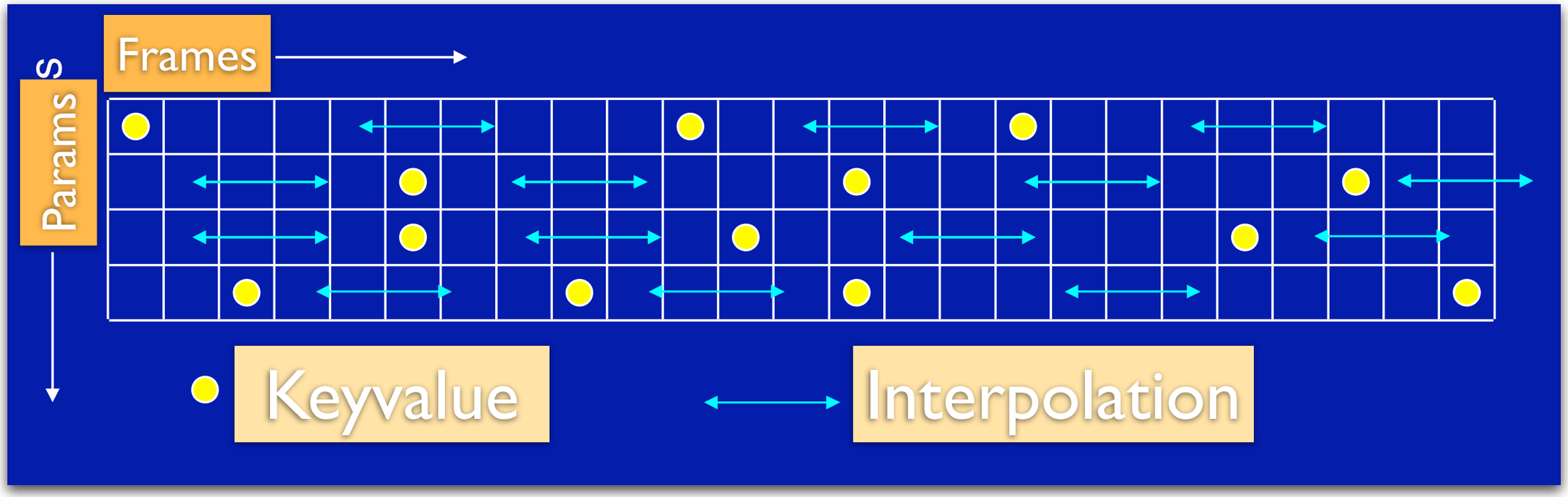


Figure 10.10 Inbetweening with nonlinear interpolation and easing. The ball changes speed as it approaches and leaves keyframes, so the dots indicating calculations made at equal time intervals are no longer equidistant along the path.

Keyframe basics

- For each variable, specify its value at the "important" frames. Not all variables need agree about which frames are important.
- Hence, key values rather than key frames.
- Create path for each parameter by interpolating key values.



Problems with Interpolation

- Splines don't always do the right thing
- Classic problems
  - –Important constraints may break between keyframes
    - » feet sink through the floor
    - » hands pass through walls
  - –3D rotations
    - » Euler angles don't always interpolate in a natural way
- Classic solutions:
  - –More keyframes!
  - –Quaternions help fix rotation problems

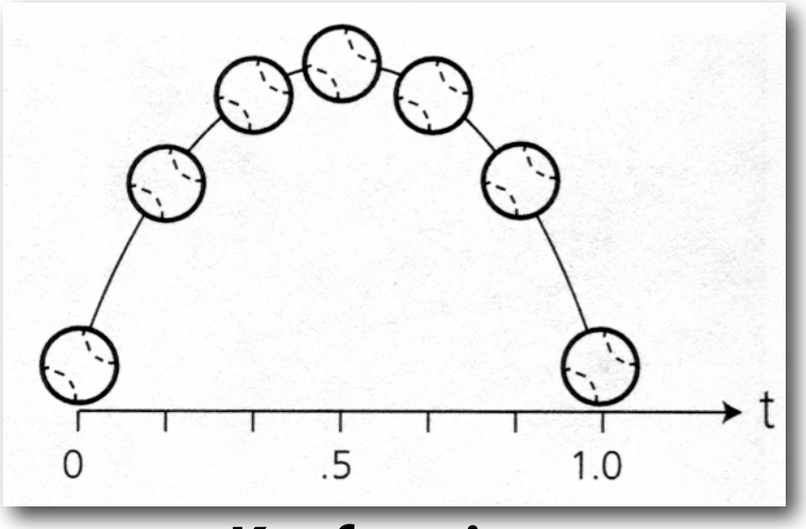
Overview

Announcements

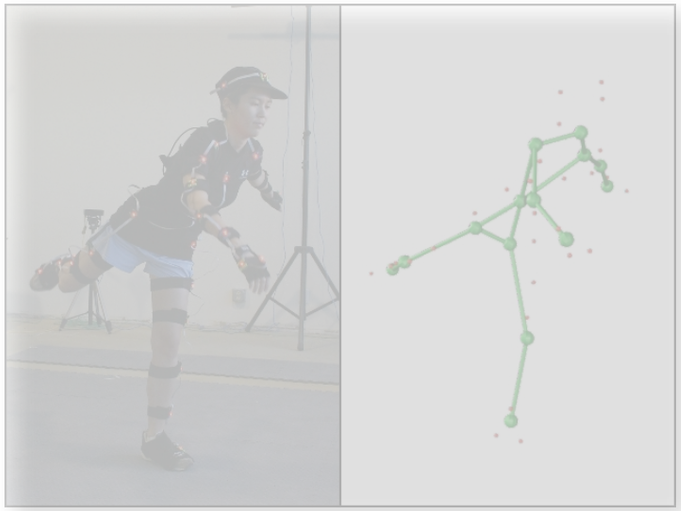
Radiosity

Animation Intro

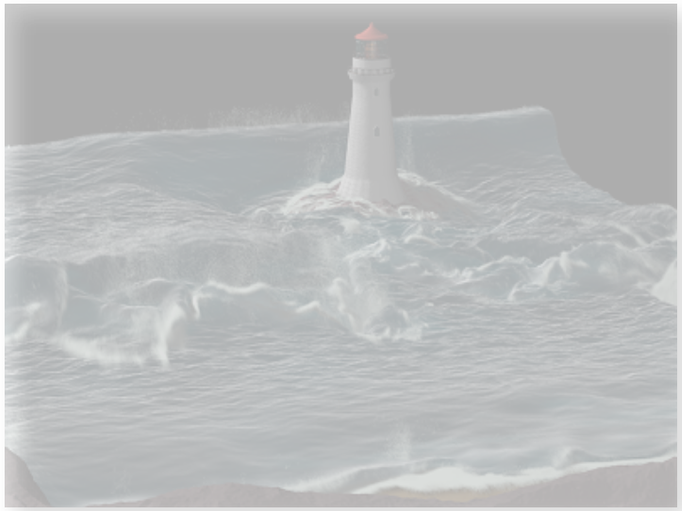
Cell Animation



Keyframing



Data-driven Animation



Physical Simulation

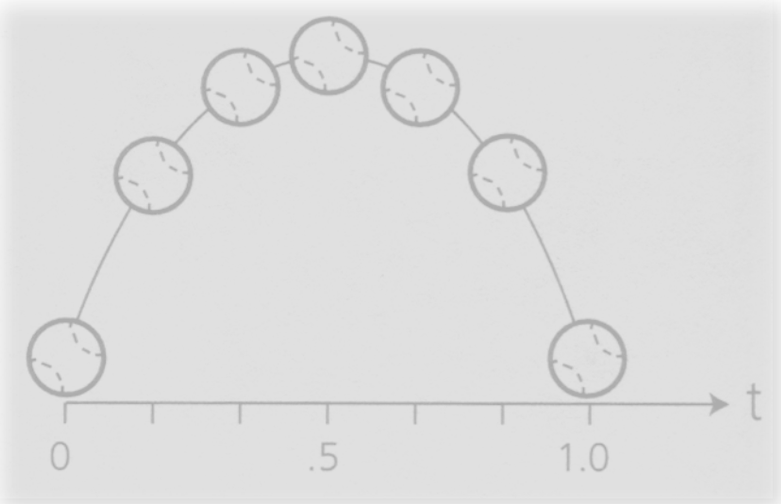
Overview

Announcements

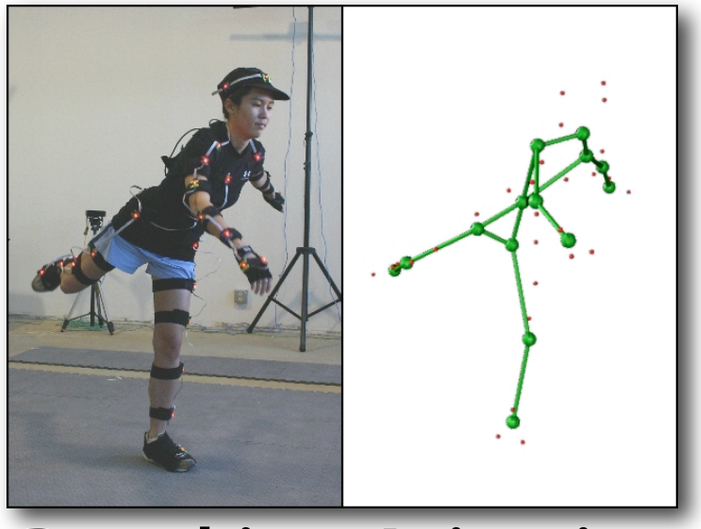
Radiosity

Animation Intro

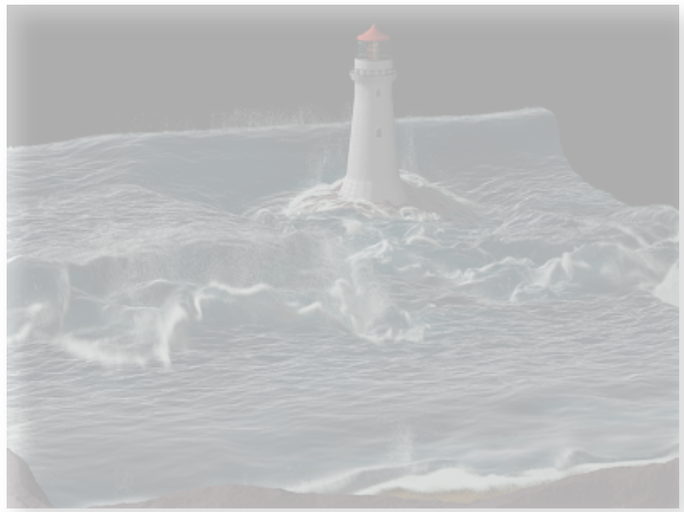
Cell Animation



Keyframing



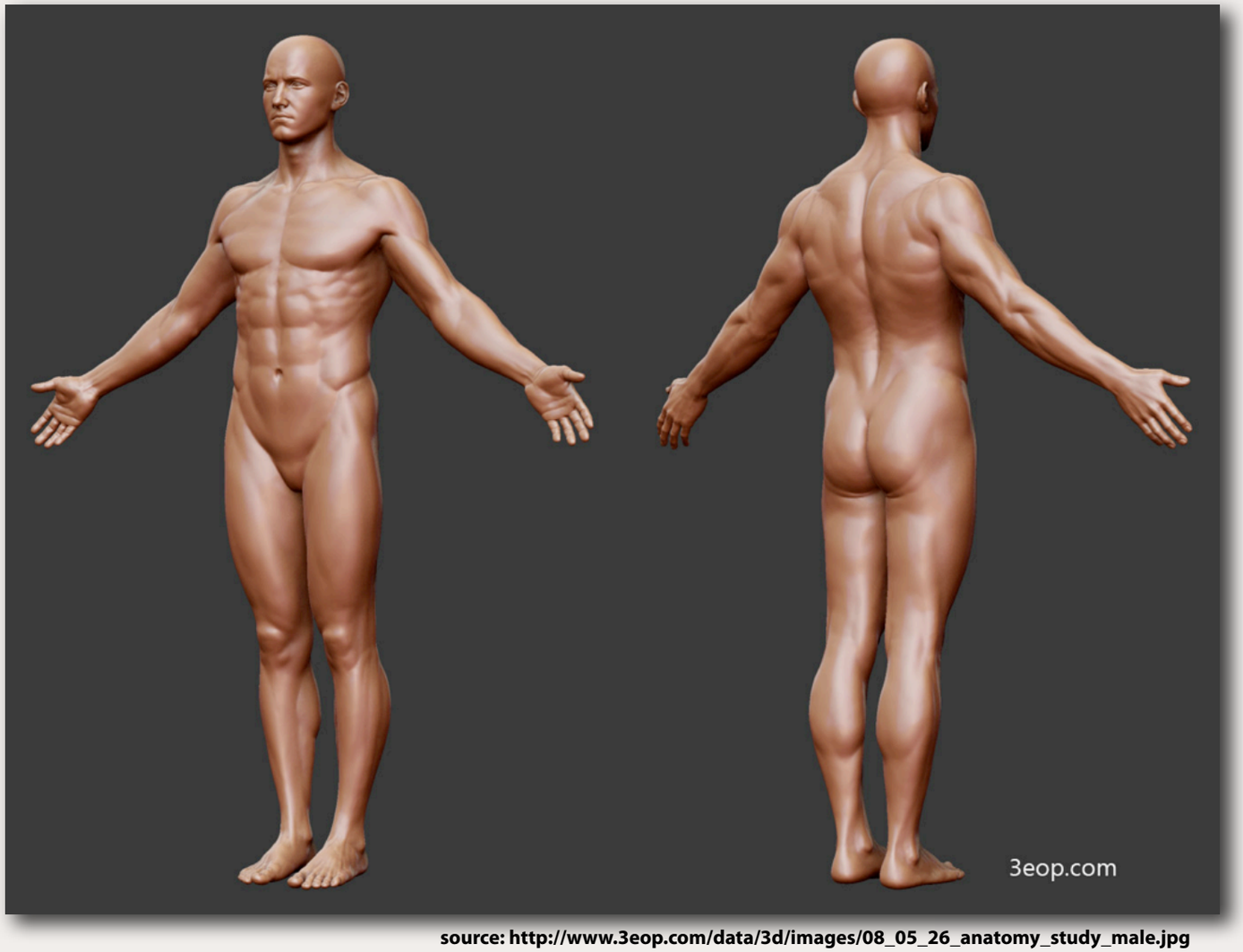
Data-driven Animation



Physical Simulation

Data-Driven Animation

- ***Capturing*** **the data or effect we want to animate.**
- **The classic example is *humans*.**



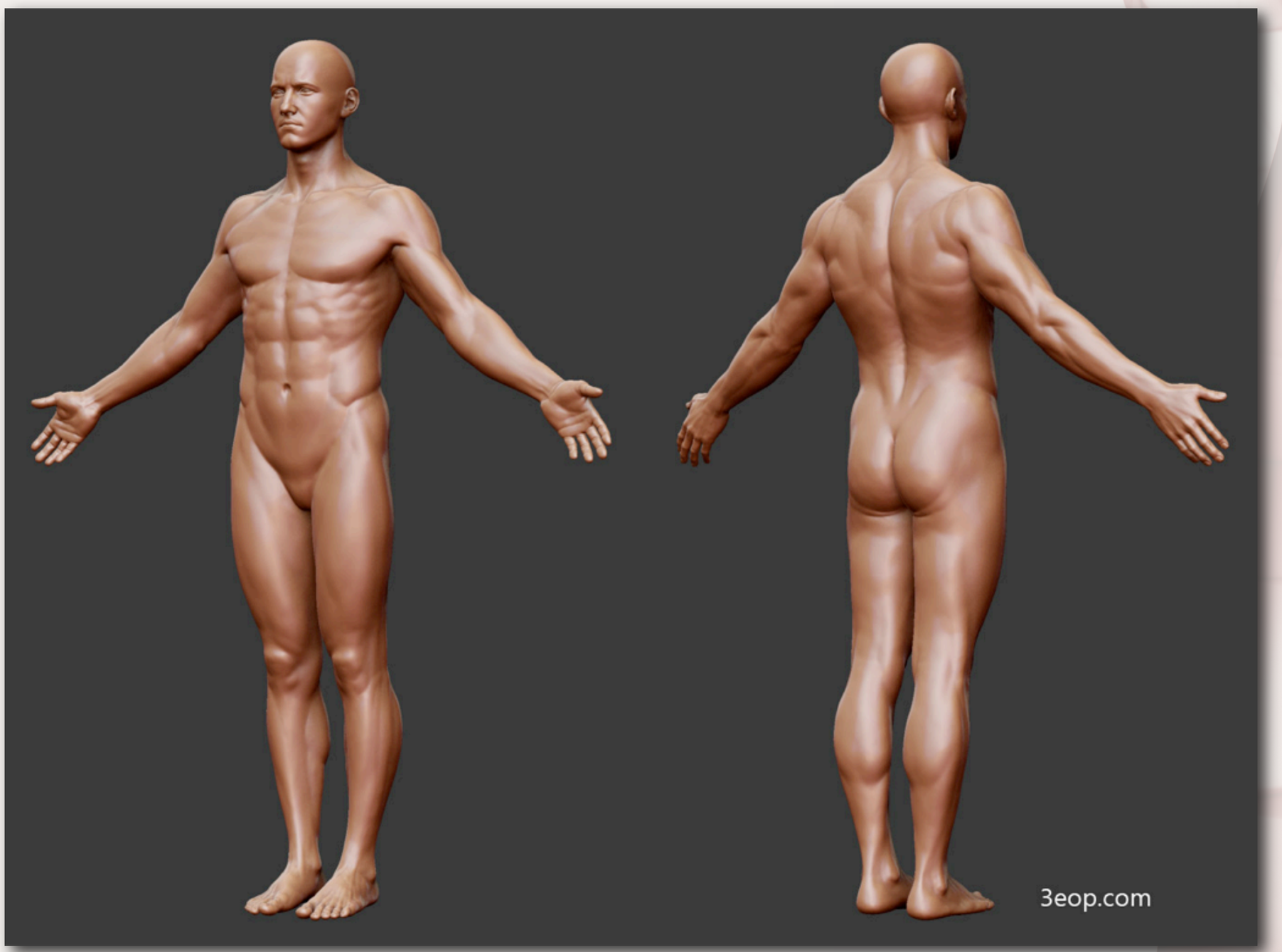
source: http://www.3eop.com/data/3d/images/08_05_26_anatomy_study_male.jpg

Humans

- **Body models.**
- Animation

source: 3dscience.com

Body Representation

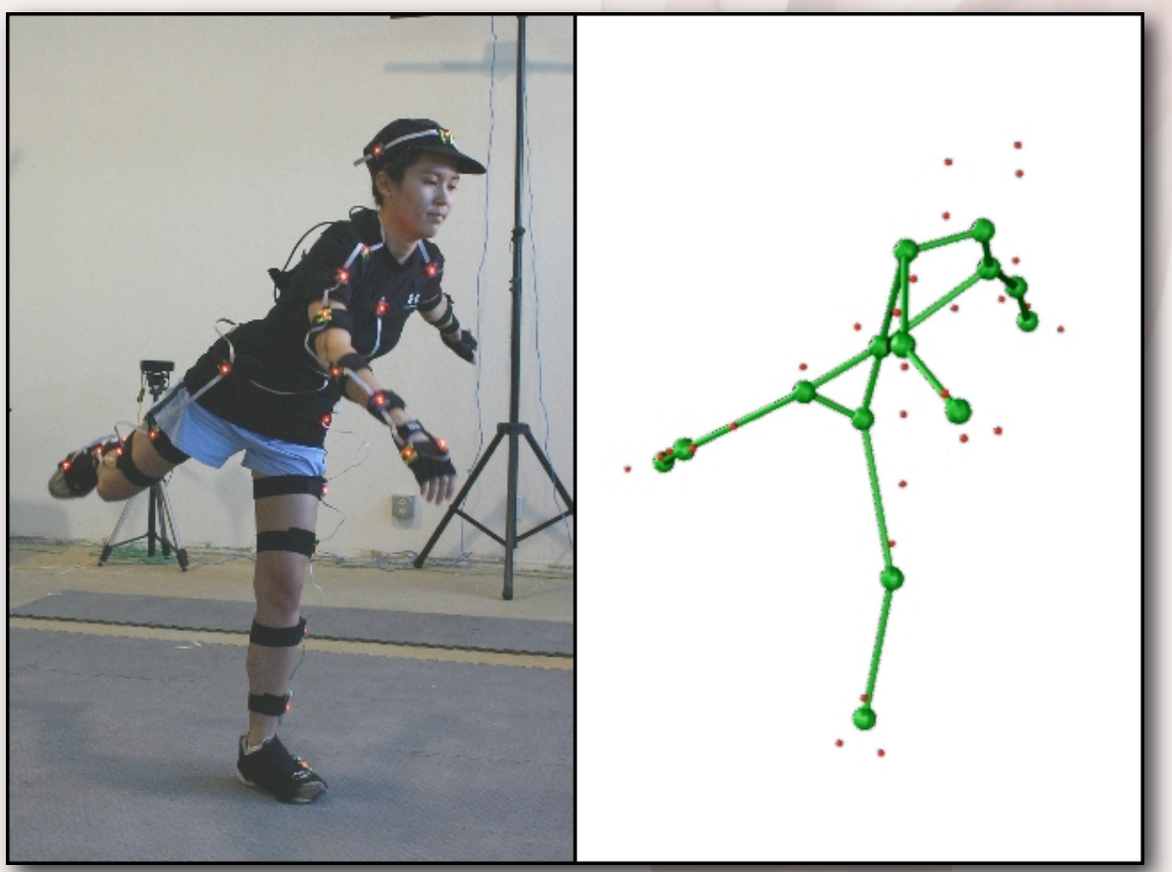


source: http://www.3eop.com/data/3d/images/08_05_26_anatomy_study_male.jpg

How to represent a human body on a computer?

Body Representation

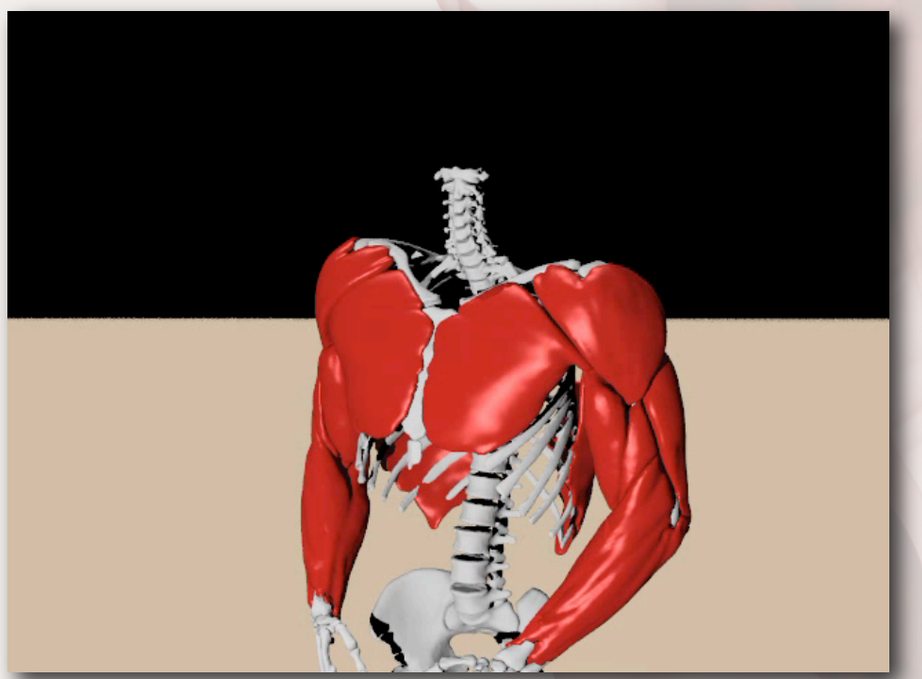
- **Kinematic Skeleton**

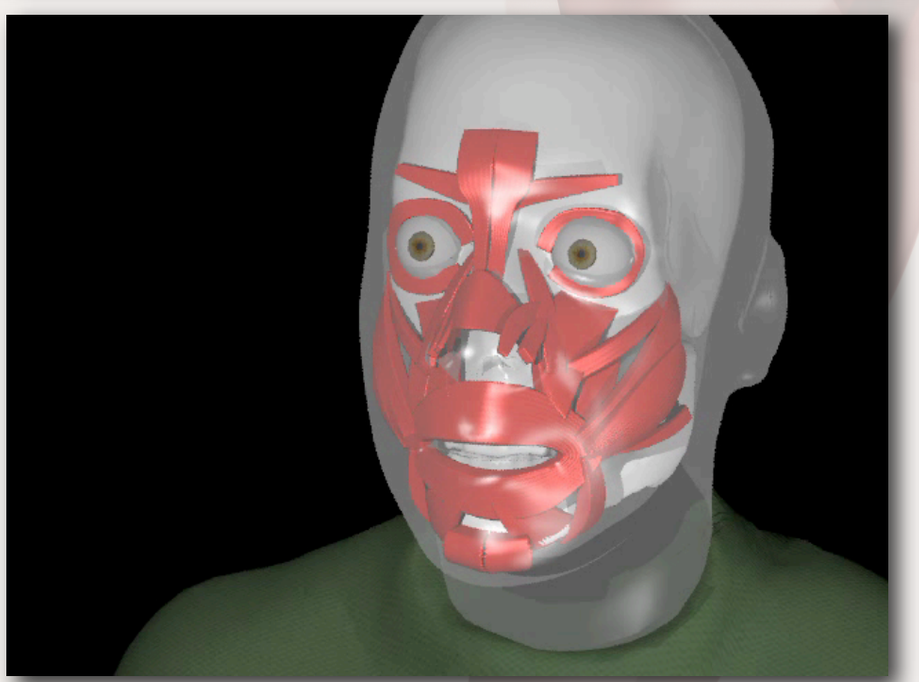


source: https://buffy.eecs.berkeley.edu/PHP/resabs/resabs.php?f_year=2005&f_submit=advgrp&f_advid=10917651

Body Representation

- Kinematic Skeleton
- Anatomical

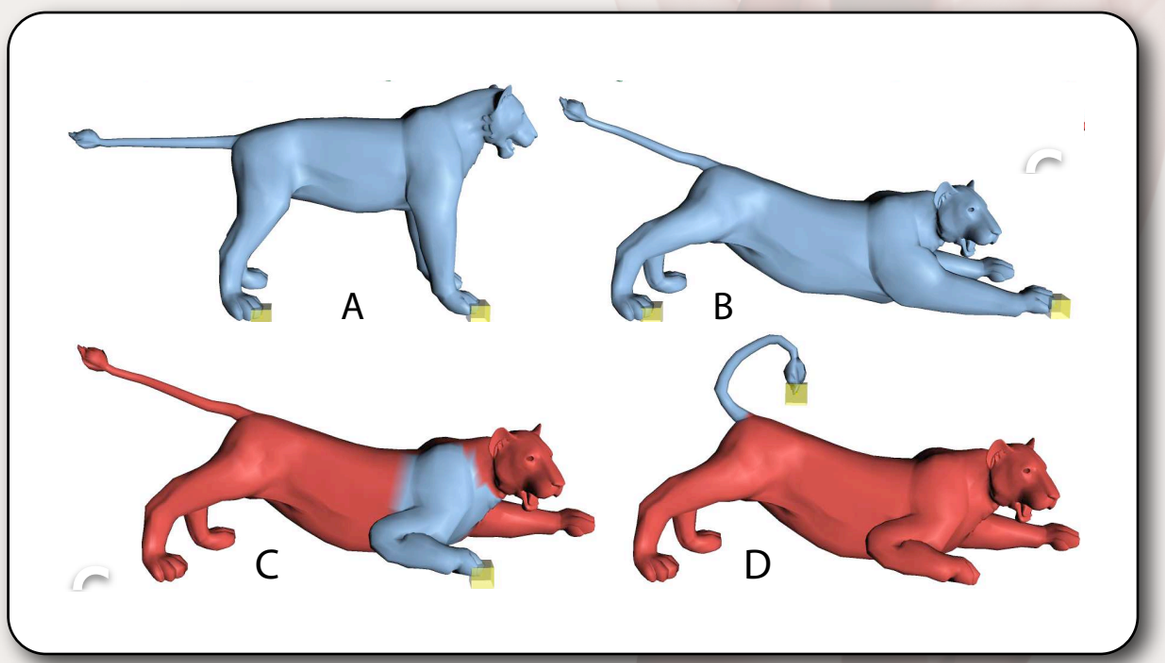




source: http://physbam.stanford.edu/~fedkiw/

Body Representation

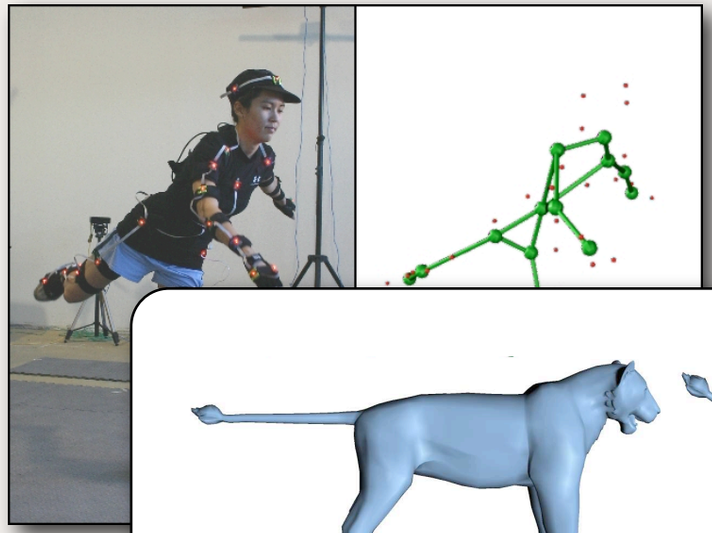
- Kinematic Skeleton
- Anatomical
- Pure Mesh

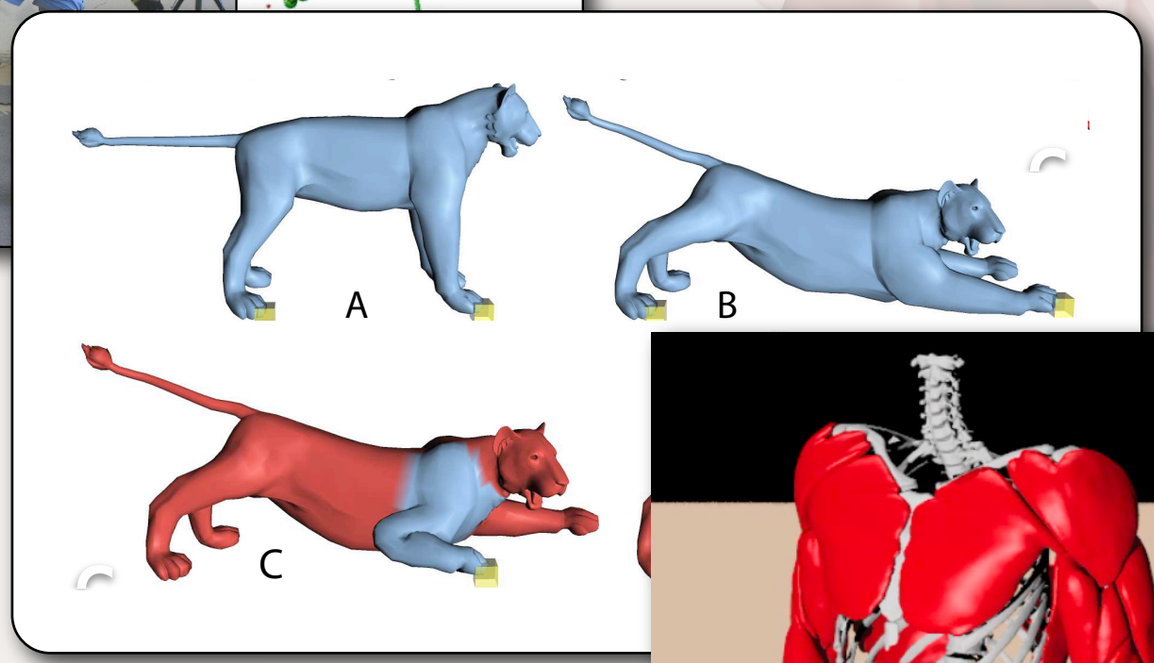


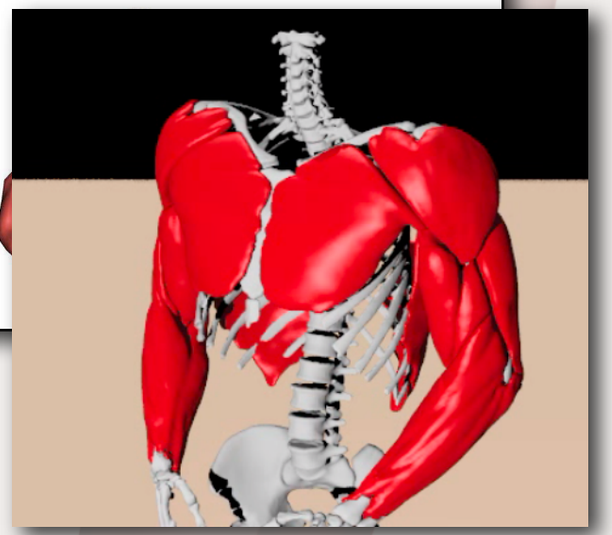
source: http://people.csail.mit.edu/sumner/research/meshik/

Body Representation

- Kinematic Skeleton
- Anatomical
- Pure Mesh
- What are the advantages and disadvantages?

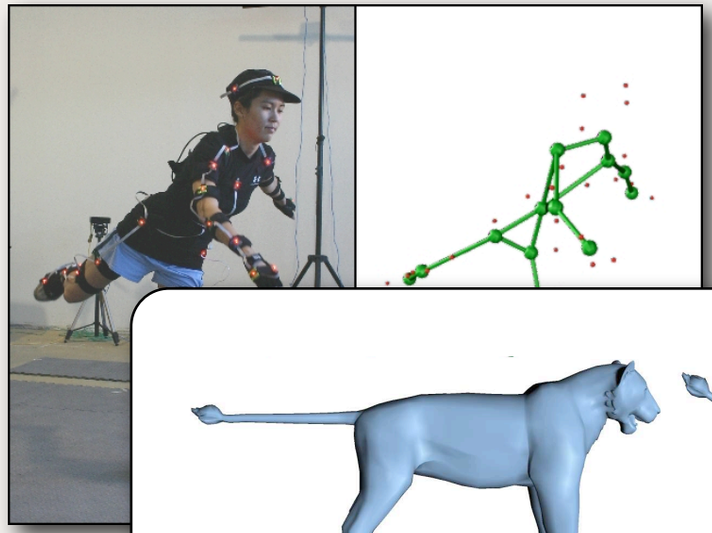


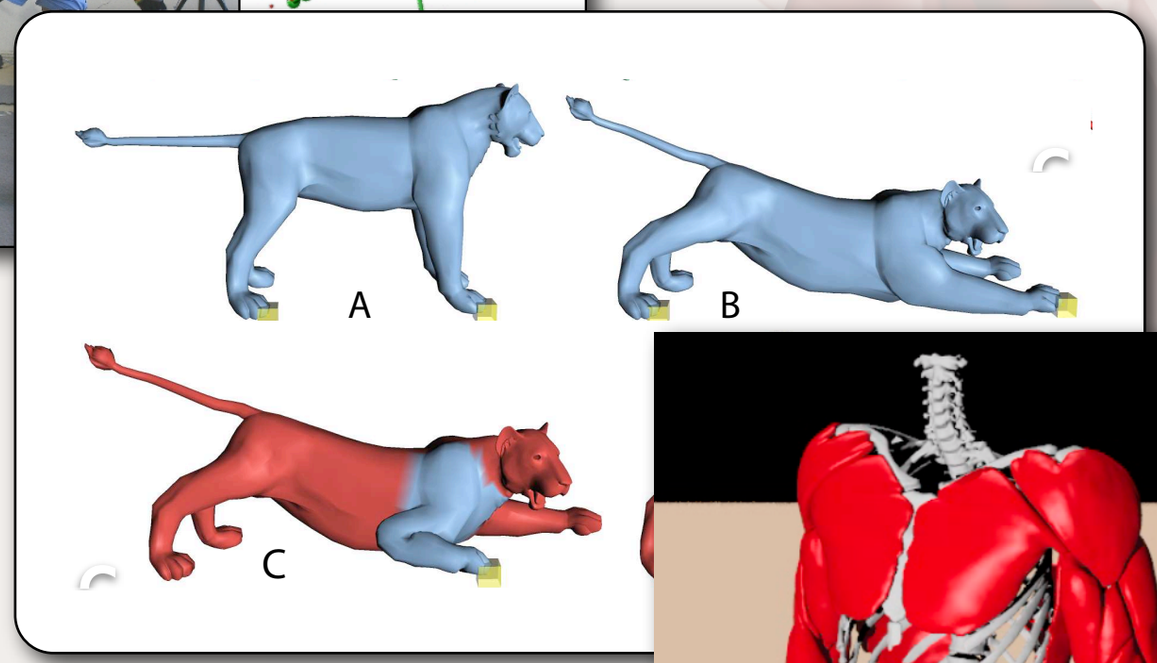


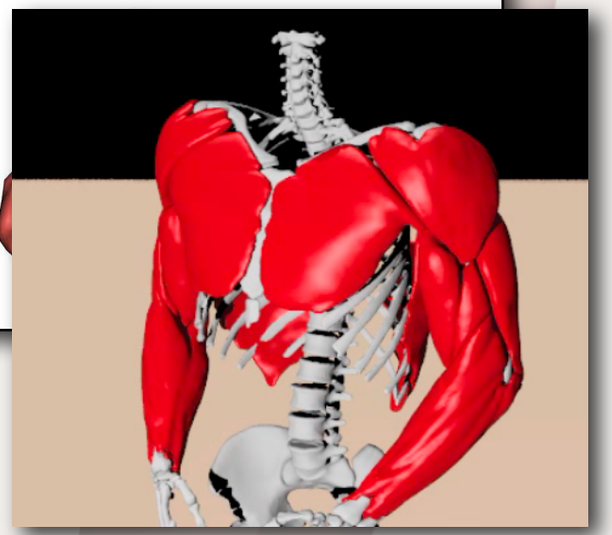


Body Representation

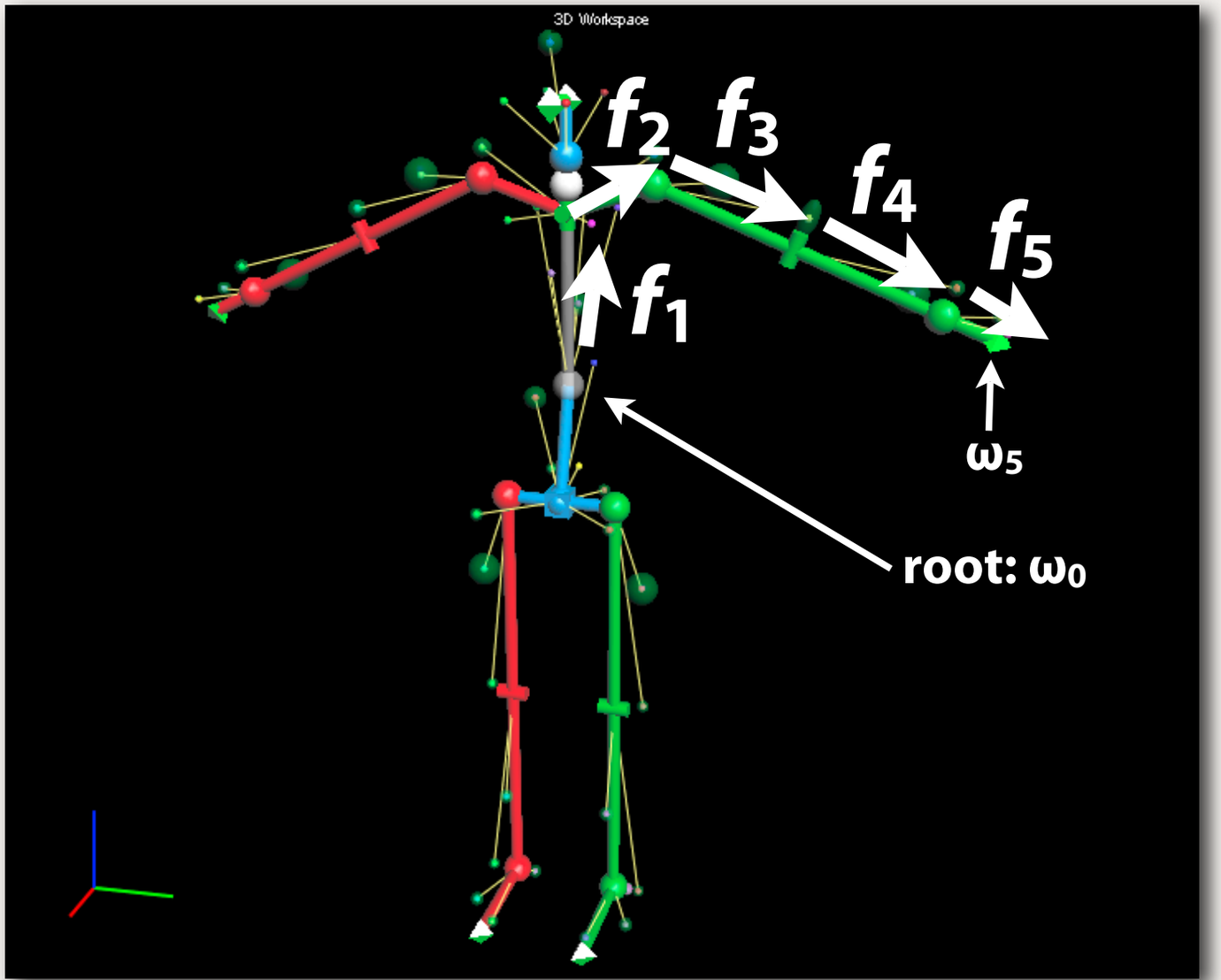
- **Kinematic Skeleton**
- **Anatomical**
- **Pure Mesh**
- **What are the advantages and disadvantages?**







Skeleton Representation



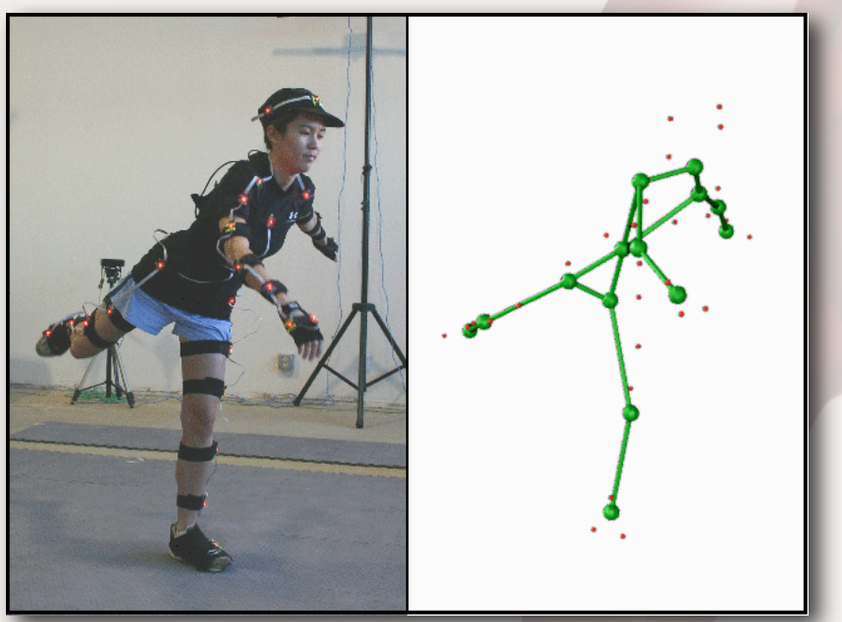
Ω is the vector of *internal* joint angles, i.e. shoulders, hips, etc.

$$\omega_0 = [\mathbf{x}_0, \theta_0] \in \mathbf{R}^6$$

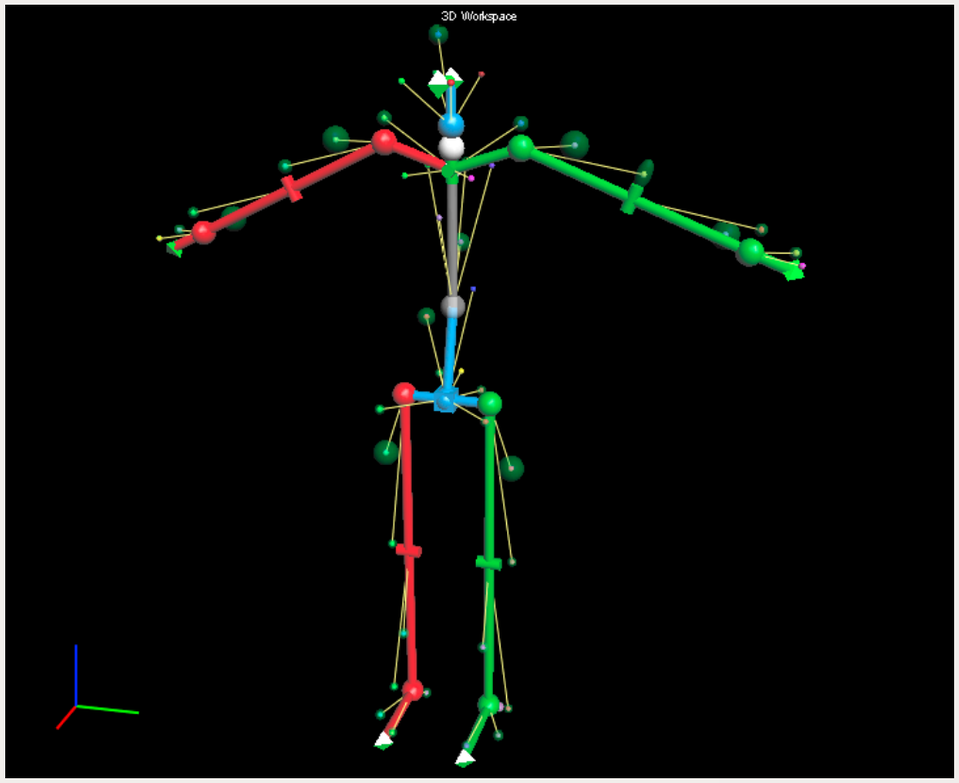
$$\omega_i = f_{i,\Omega}(\omega_{i-1})$$

Motion Capture

- Attach markers to a humans body.
- Calibrate a skeleton which makes those markers "make sense."
- Cameras capture 2D markers positions.
- Estimate 3D marker positions.
- *Inverse kinematics*: convert marker positions to skeleton...
  - How?



Marker Energy Function



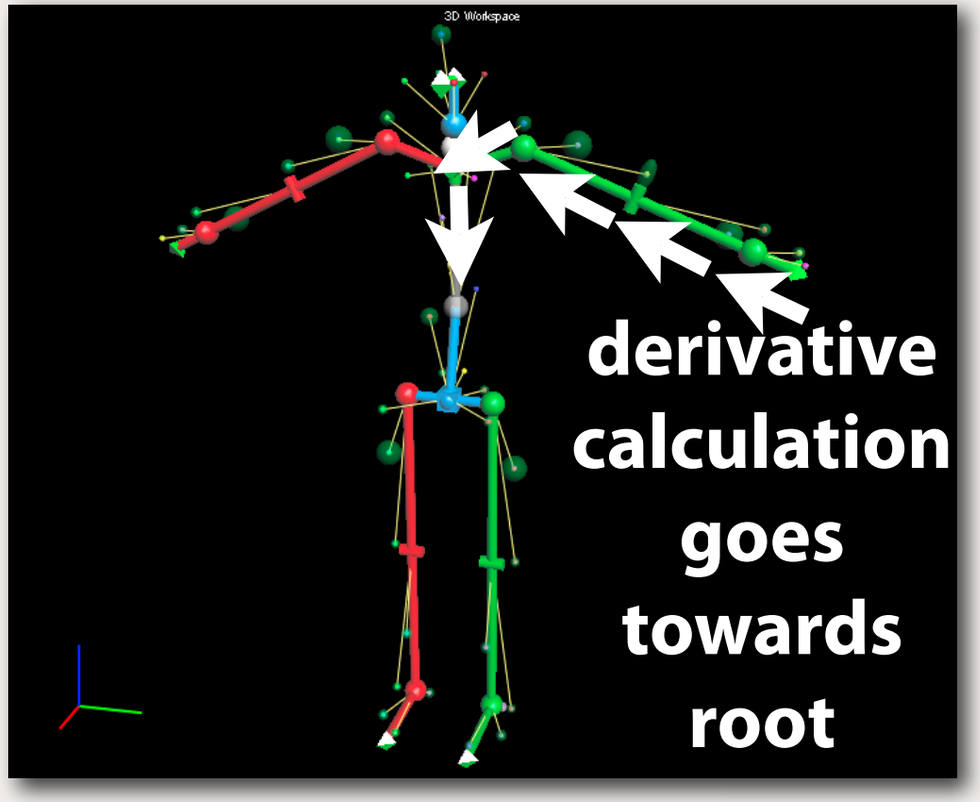
$$\omega_i = f_{i,\Omega}(\omega_{i-1})$$

$$\hat{\mathbf{m}}_j = \tau_i(\omega_i)\mathbf{m}_j$$

$$E = \sum_j ||\hat{\mathbf{m}}_j^\star - \hat{\mathbf{m}}_j||^2$$

$$\frac{dE}{d\Omega}$$

Derivatives

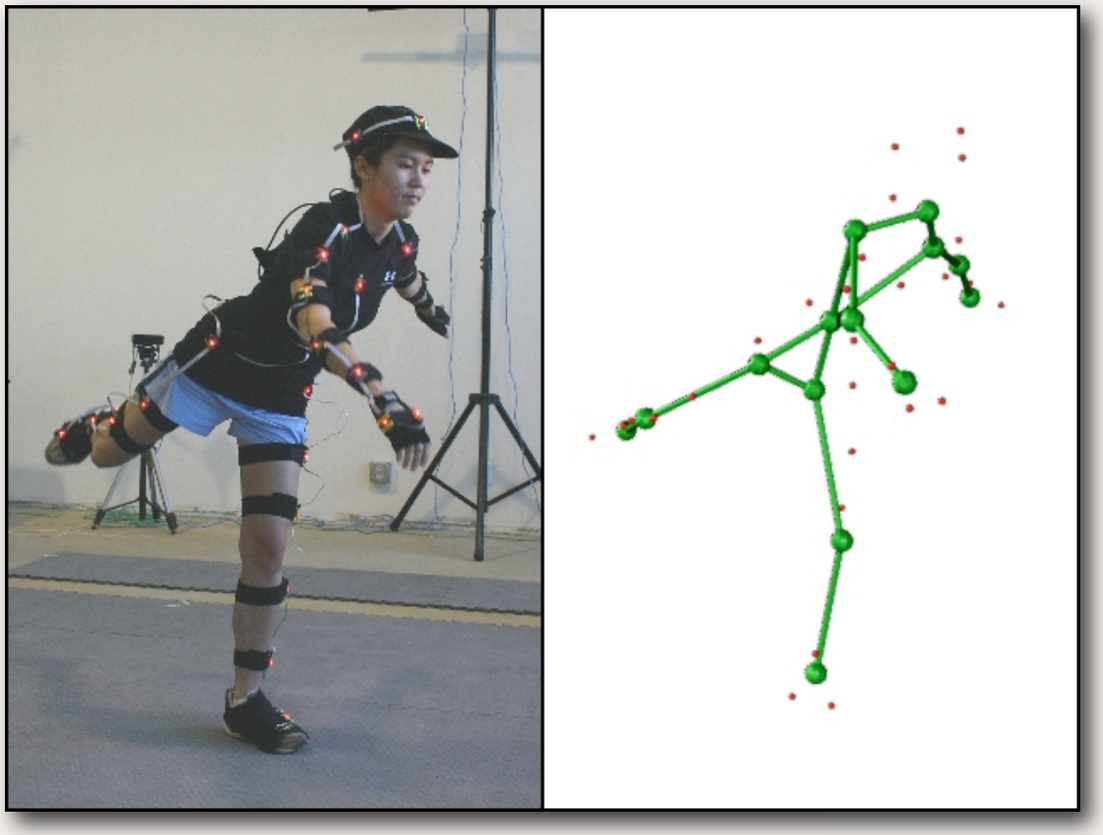


$$\omega_i = f_{i,\Omega}(\omega_{i-1})$$

$$\frac{dE}{d\Omega} = 2\sum_j \underbrace{(\hat{\mathbf{m}}_j^\star - \hat{\mathbf{m}}_j)^T}_{\text{vector}} \underbrace{\frac{d\hat{\mathbf{m}}_j}{d\Omega}}_{\text{matrix}}$$

$$\frac{d\hat{\mathbf{m}}_j}{d\Omega} = \underbrace{\frac{\partial \hat{\mathbf{m}}_j}{\partial \omega_i}}_{\text{matrix}} \underbrace{\left(\frac{\partial \omega_i}{\partial \Omega} + \frac{\partial \omega_i}{\partial \omega_{i-1}} \frac{\partial \omega_{i-1}}{\partial \Omega} + \frac{\partial \omega_i}{\partial \omega_{i-1}} \frac{\partial \omega_{i-1}}{\partial \omega_{i-2}} \frac{\partial \omega_{i-2}}{\partial \Omega} + \cdots \right)}_{\text{matrix multiplies}}$$

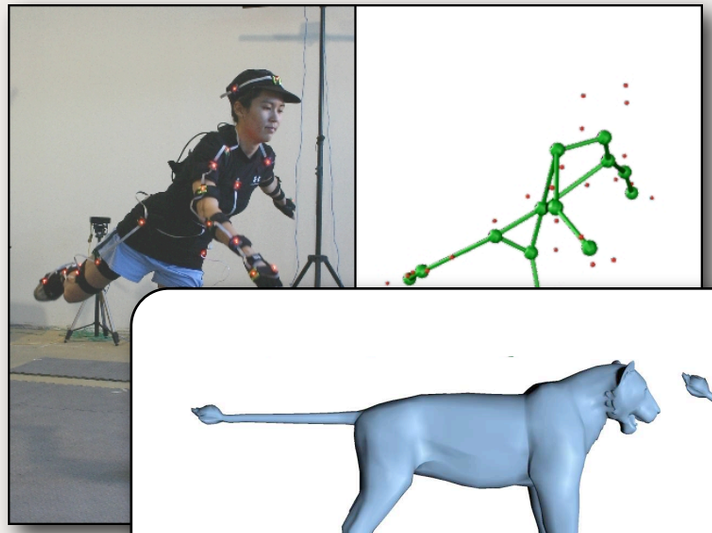
Inverse Kinematics Summary

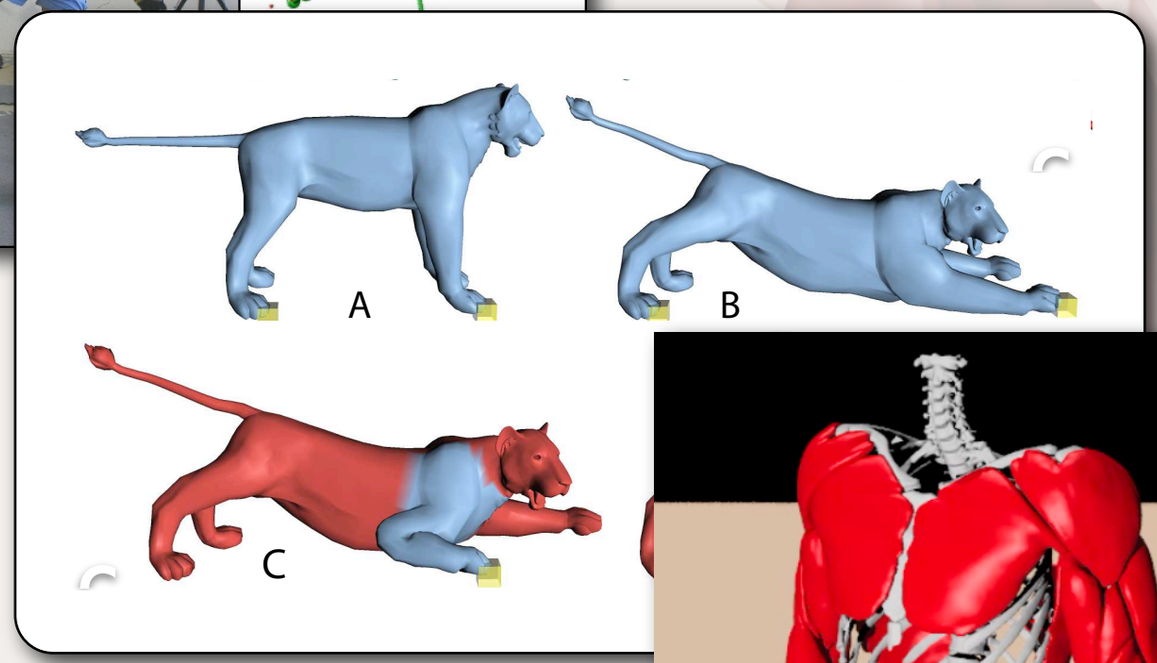


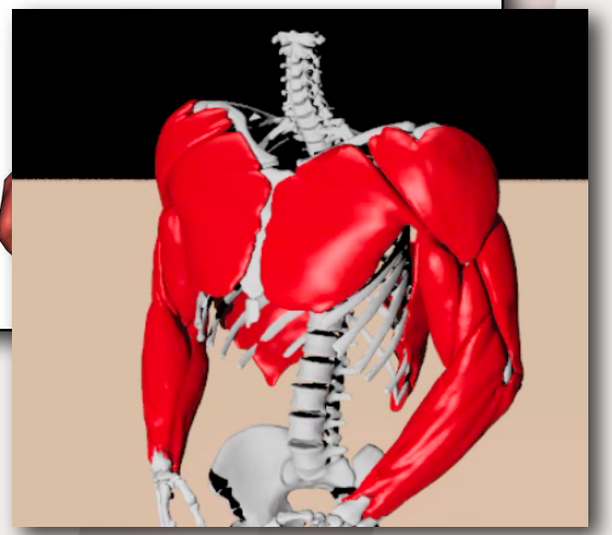
- **Telescoping composition of functions from root.**
- **Compute derivatives in the *opposite* direction!**

Body Representation

- **Kinematic Skeleton**
- **Anatomical**
- **Pure Mesh**
- **What are the advantages and disadvantages?**

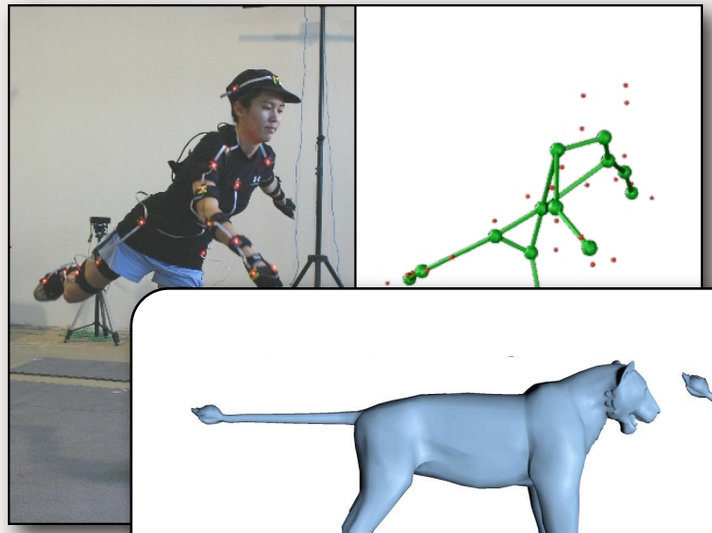


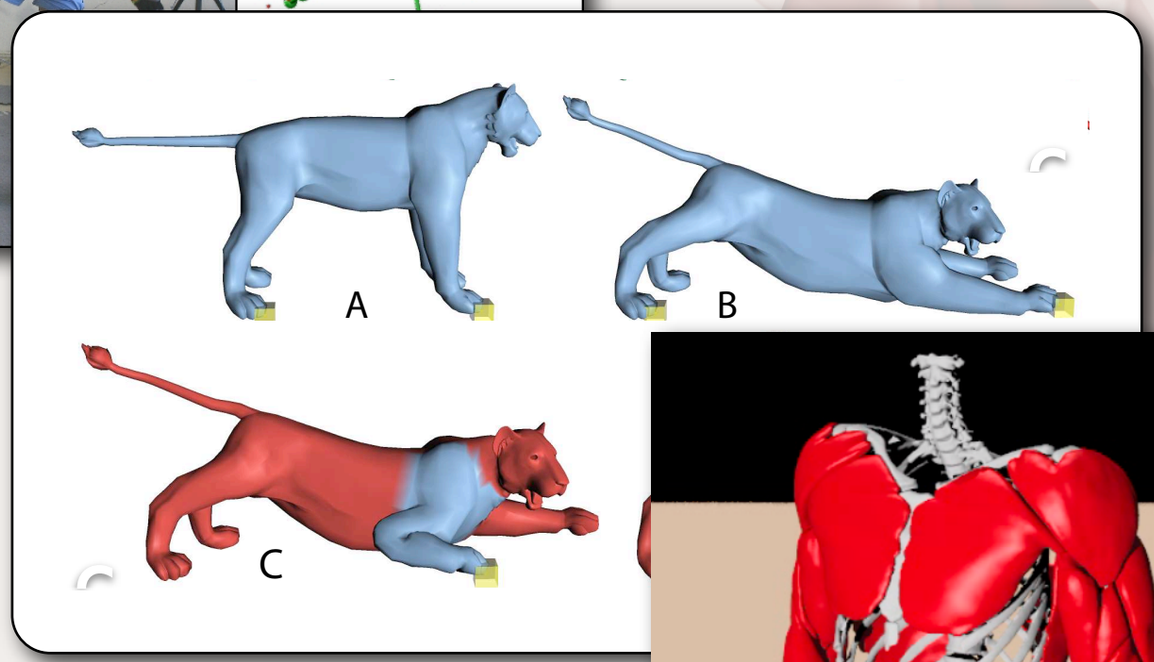


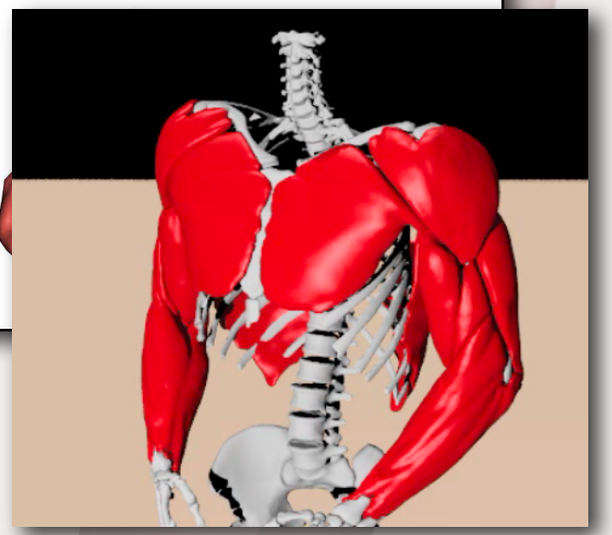


Body Representation

- Kinematic Skeleton
- Anatomical
- Pure Mesh
- What are the advantages and disadvantages?

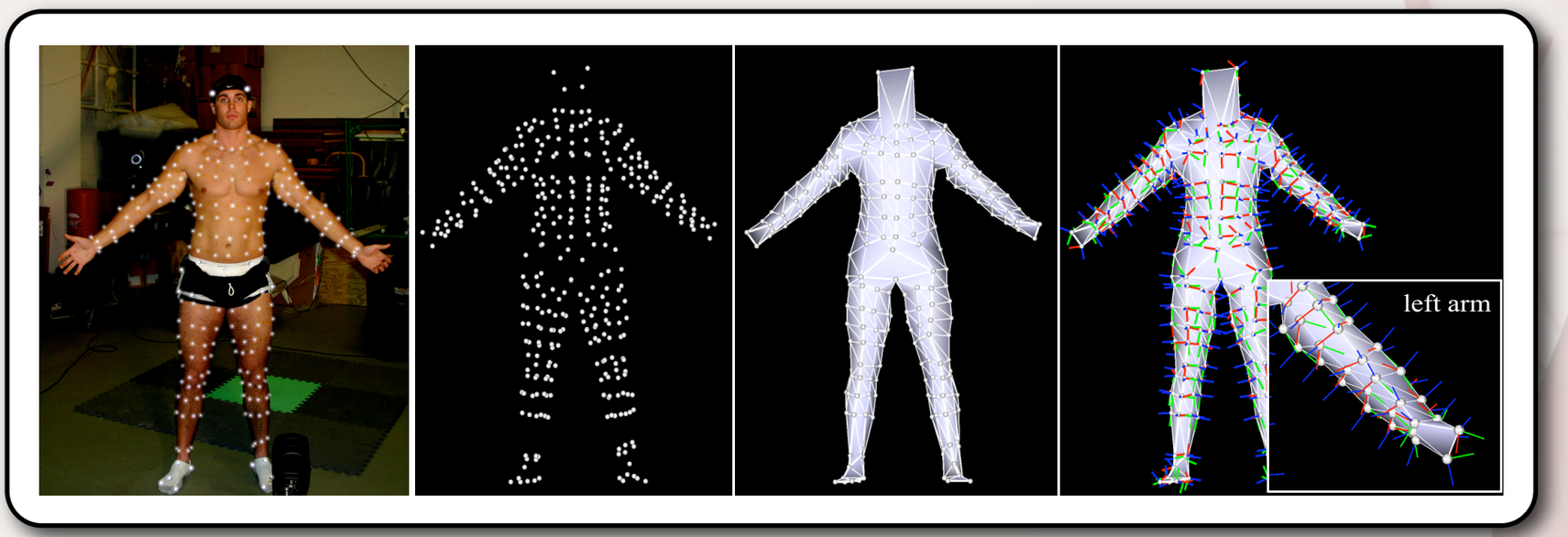






Laser Range Scanning

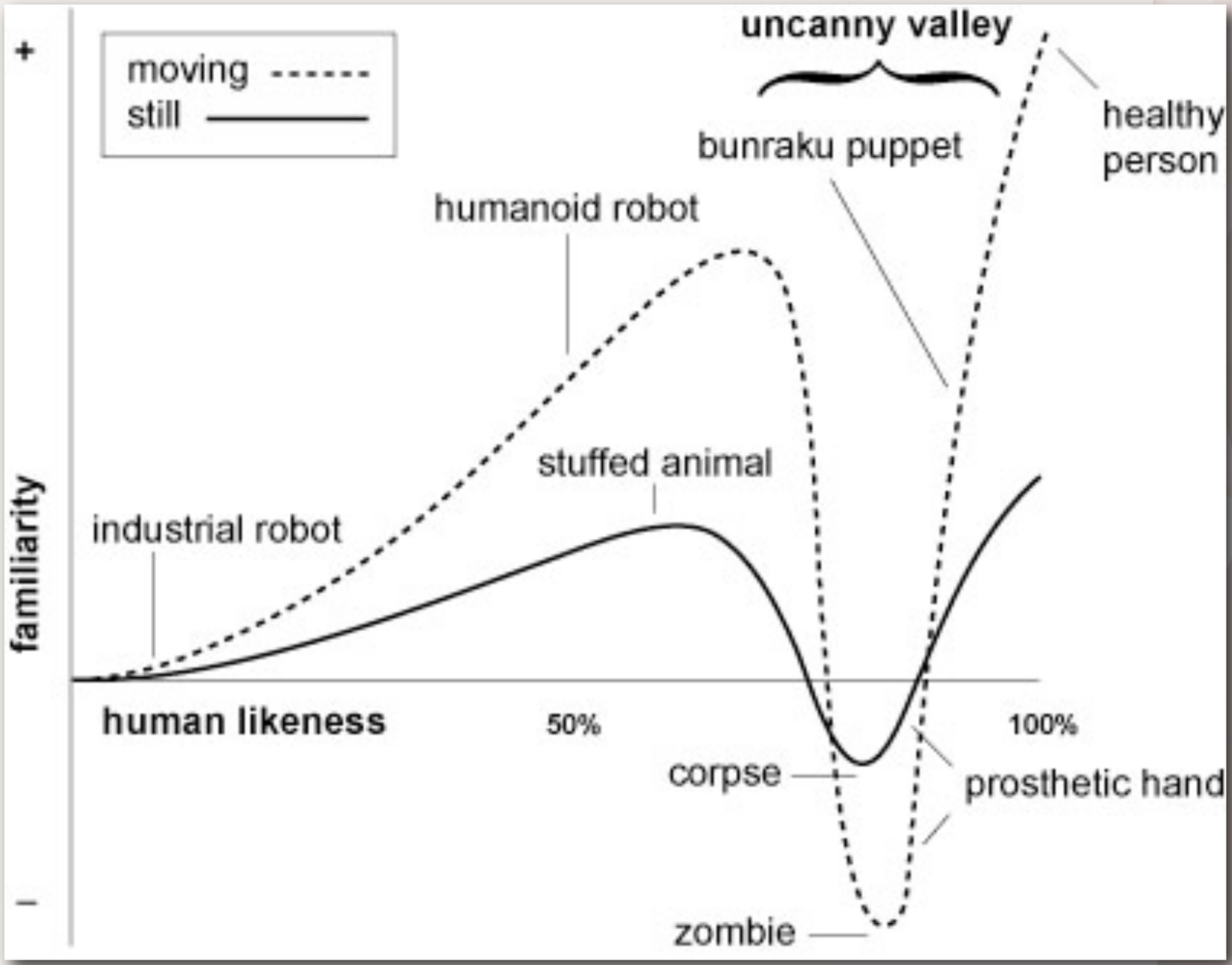
Dense Marker Capture



Capturing and Animating Skin Deformation

Robotics Institute,
Carnegie Mellon University

Uncanny Valley



Uncanny Valley

http://www.youtube.com/watch?v=9YaTUQaRCh4

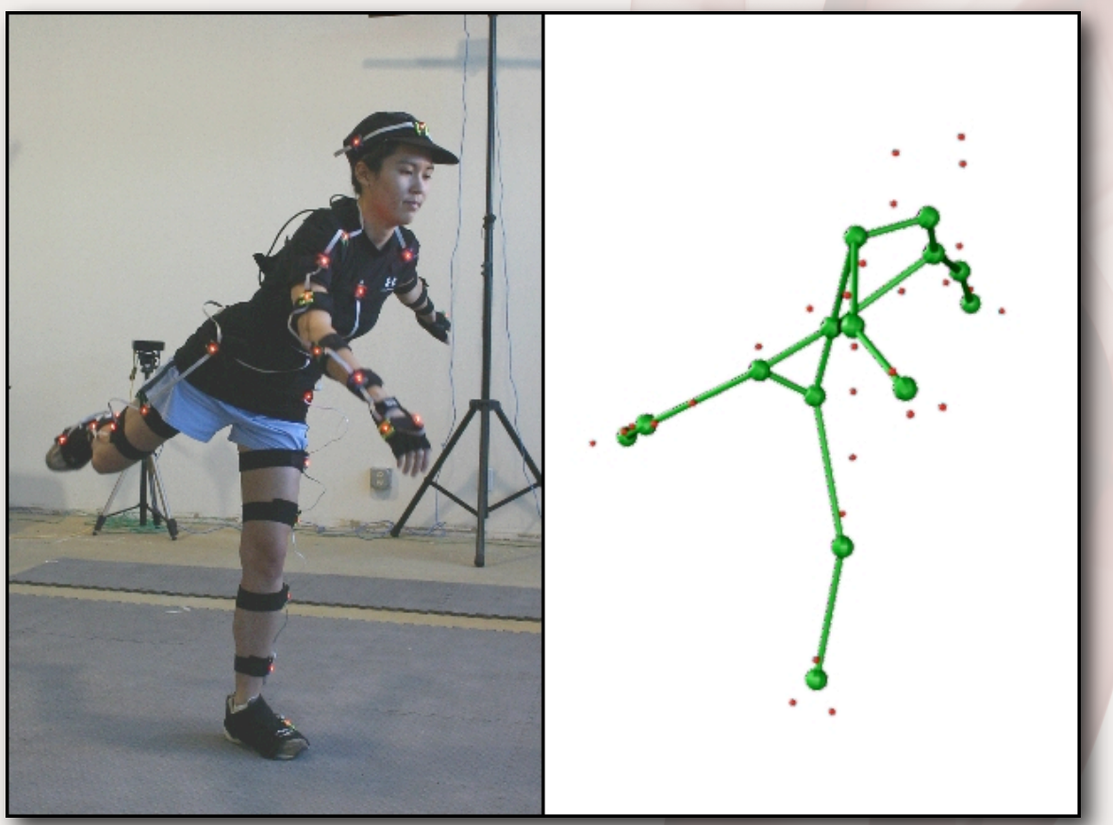
Bodies

- **Body models.**
- Animation

source: 3dscience.com

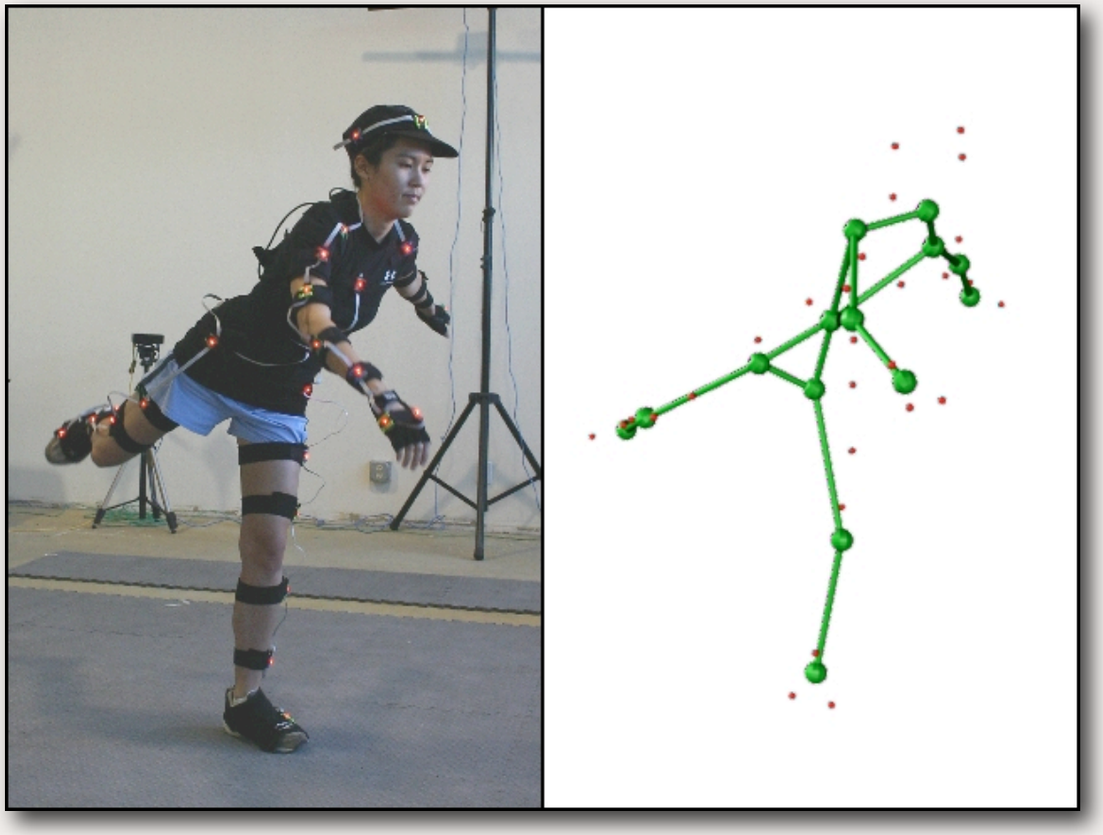
Bodies

- Body models.
- **Animation**



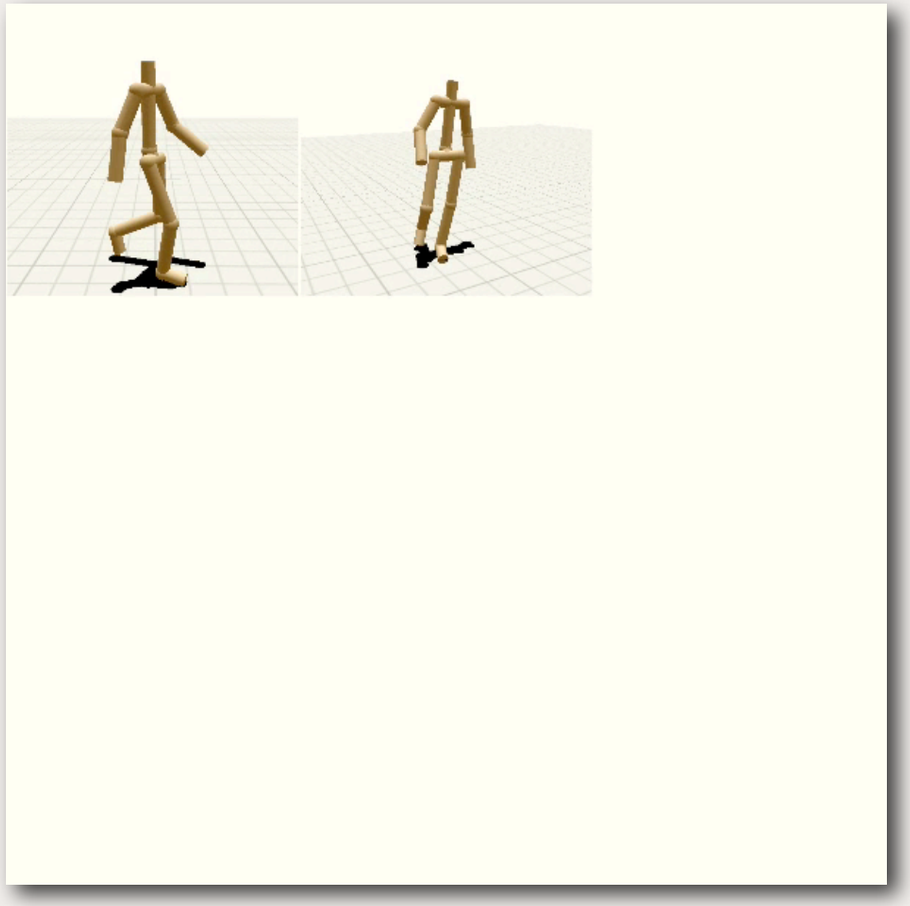
source: 3dscience.com

Motion Capture

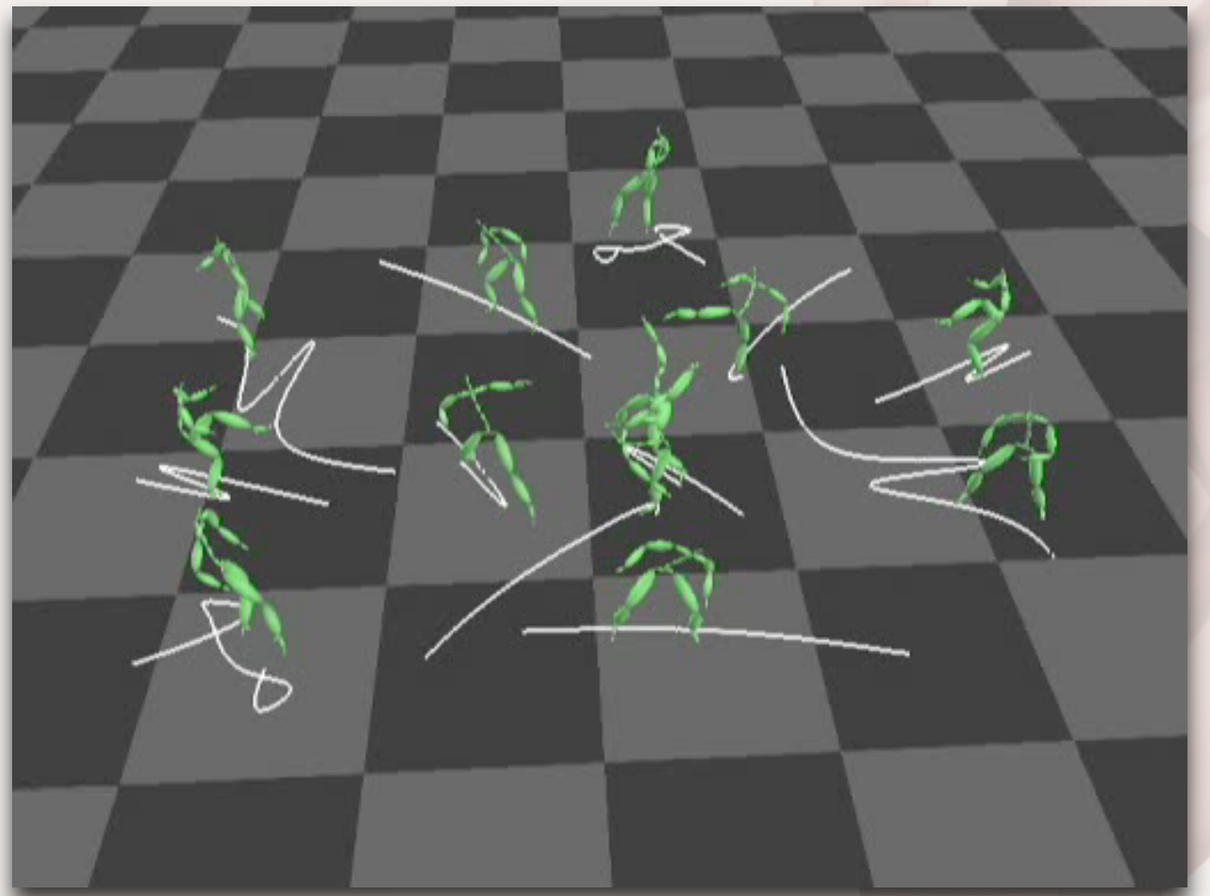


- **Telescoping composition of functions from root.**
- **Compute derivatives in the *opposite* direction!**

Clips

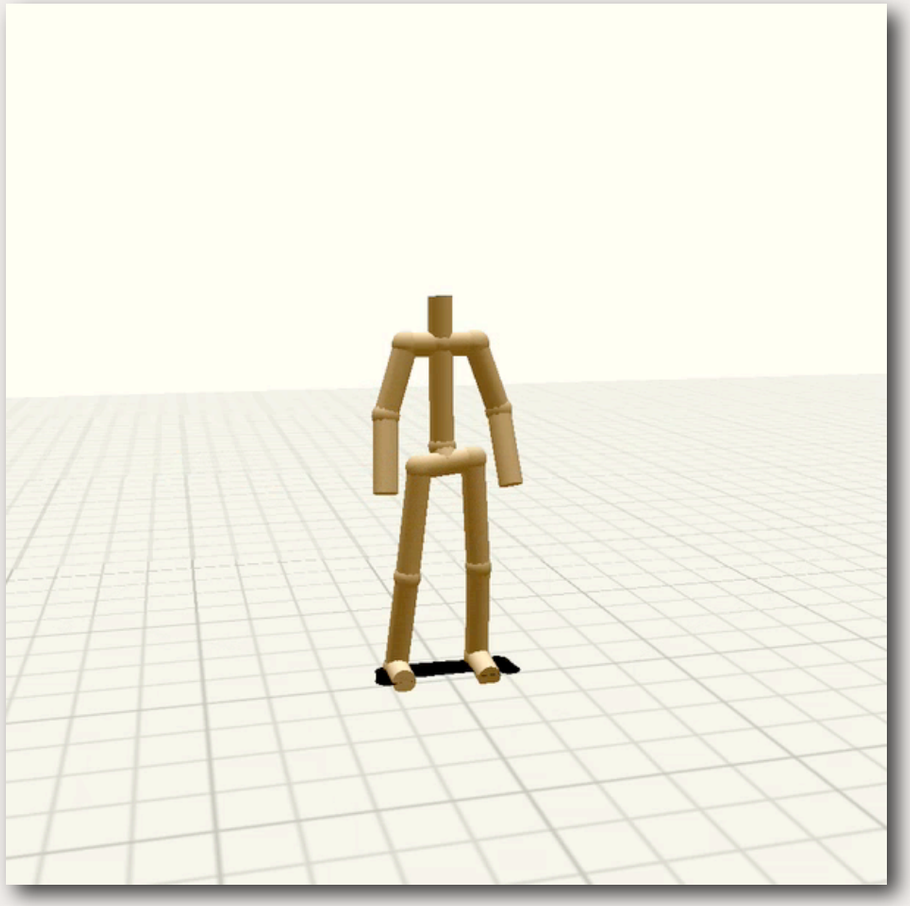


source: Treuille et al. [2002]

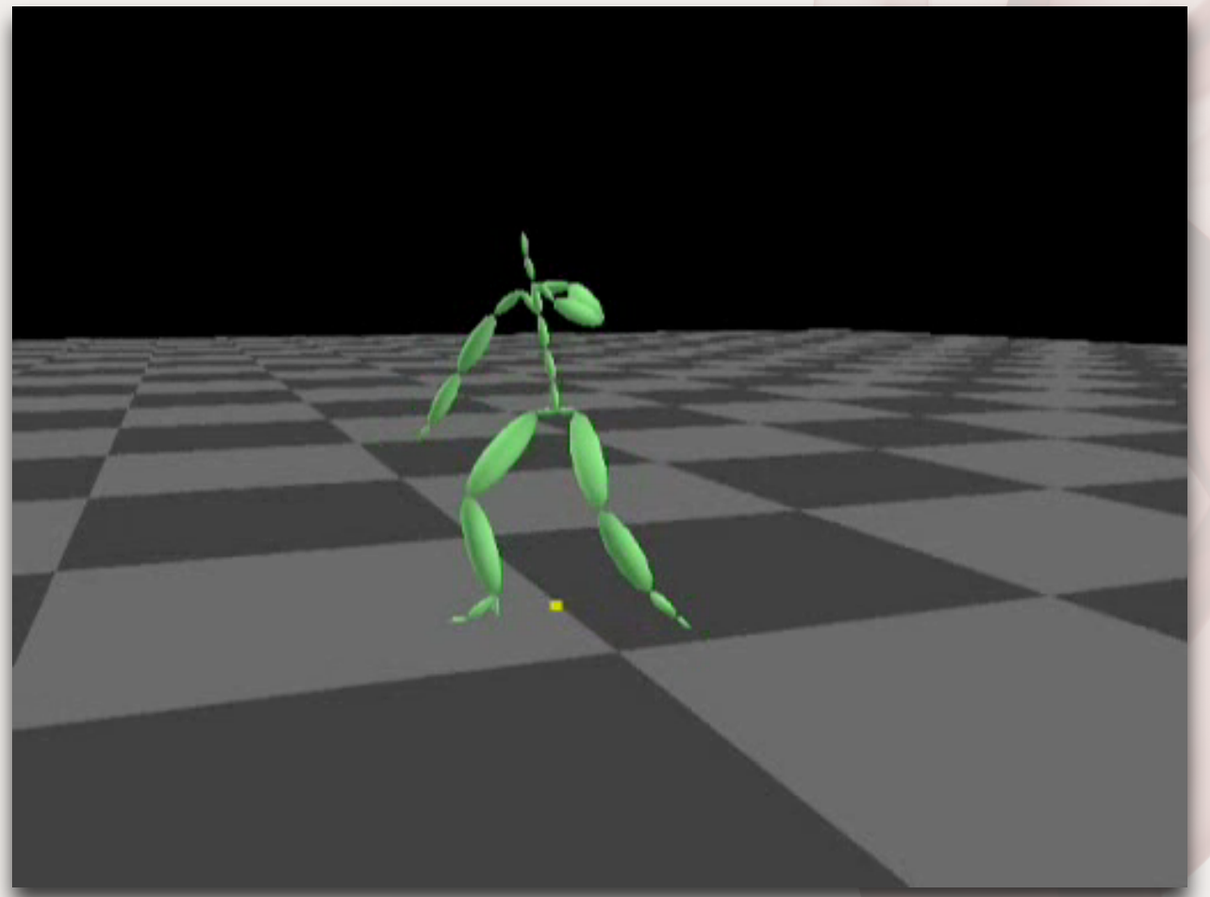


source: Kovar et al. [2002]

Sequences



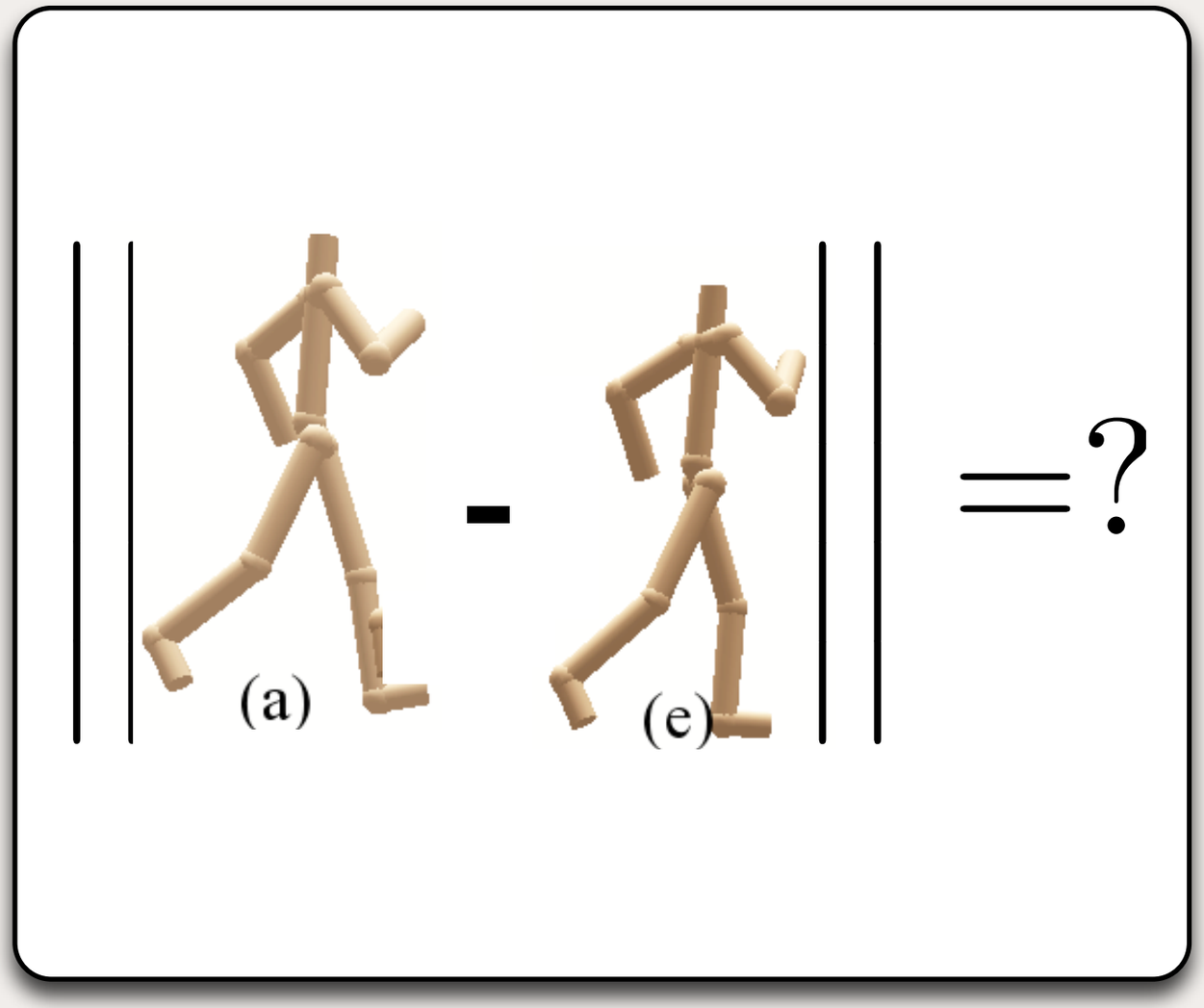
source: Treuille et al. [2002]

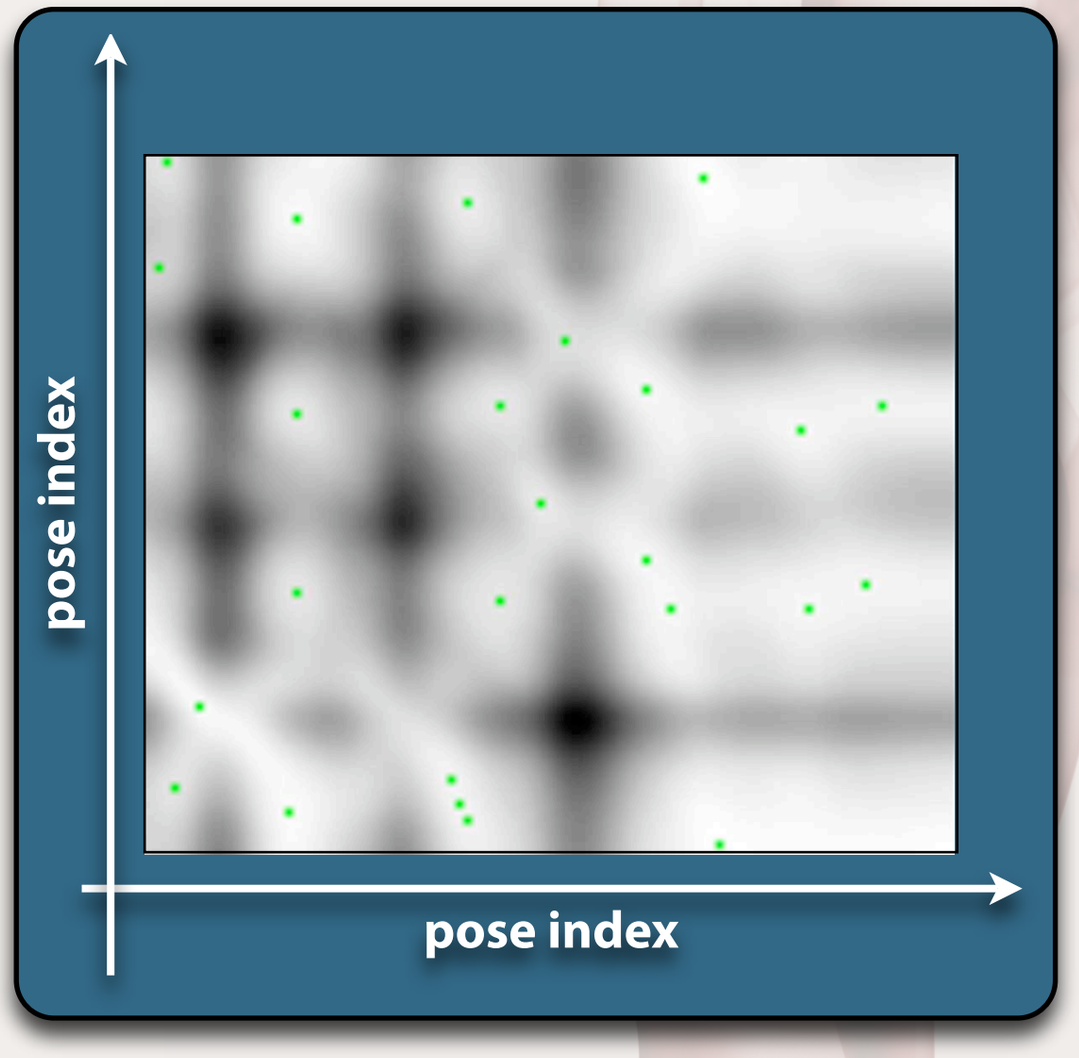


source: Kovar et al. [2002]

How?

Pose Metrics

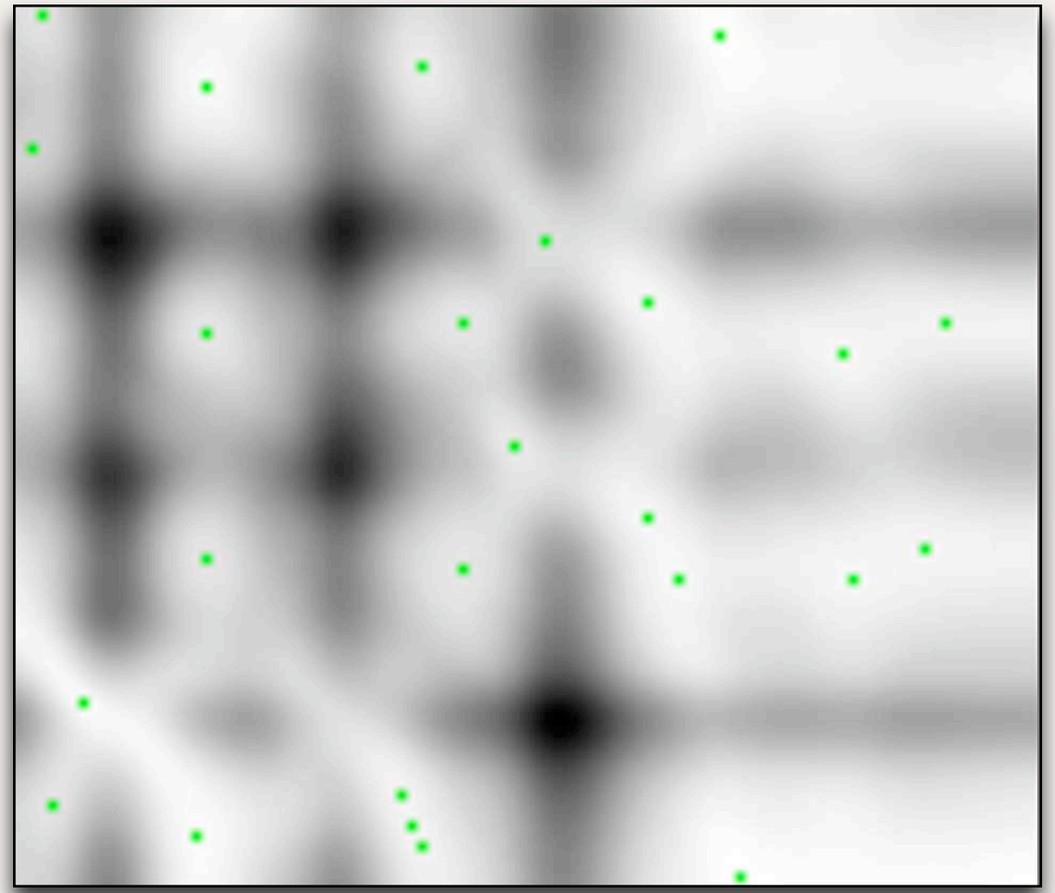




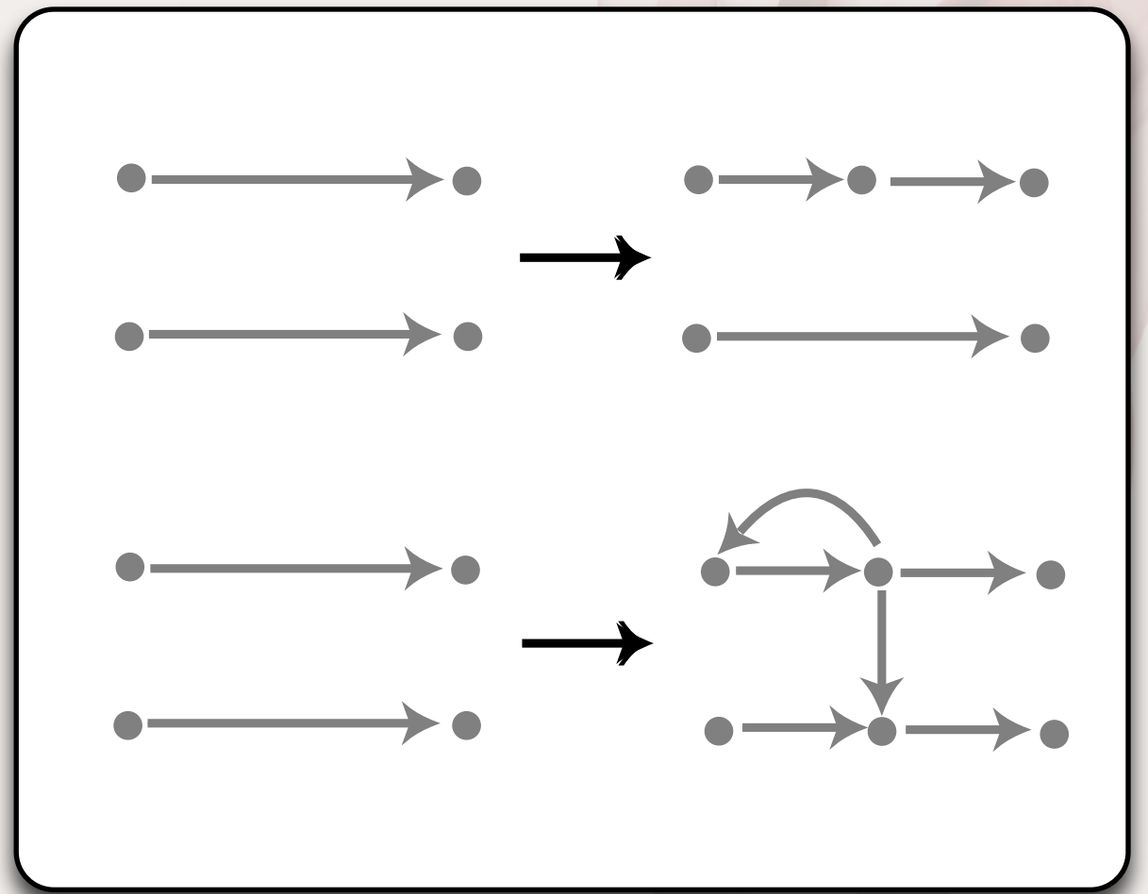
How can we define a metric on poses?

Pairwise pose differences.

Pose Metrics

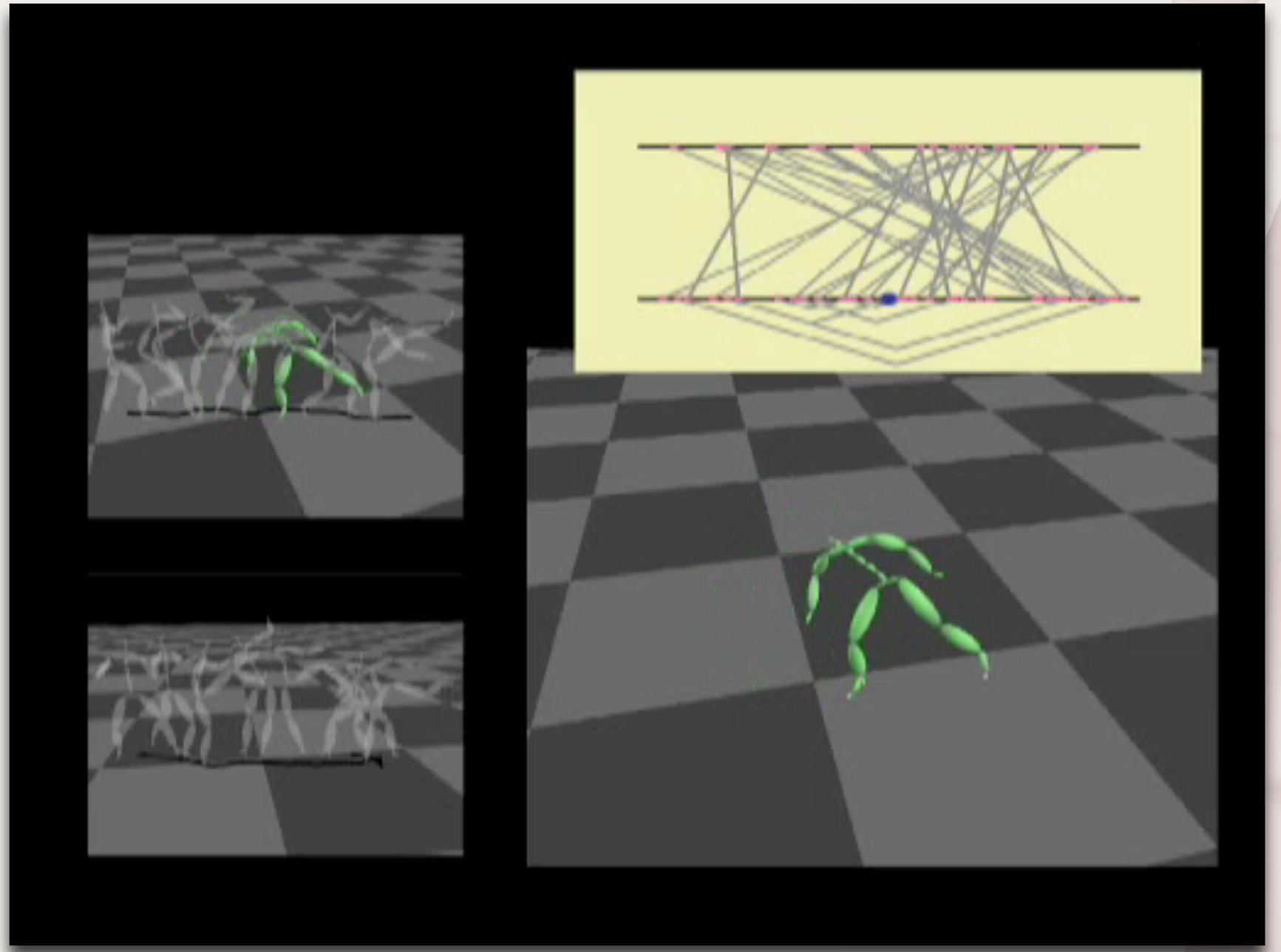


Pairwise Pose Differences



Motion Graph Schematic

Results

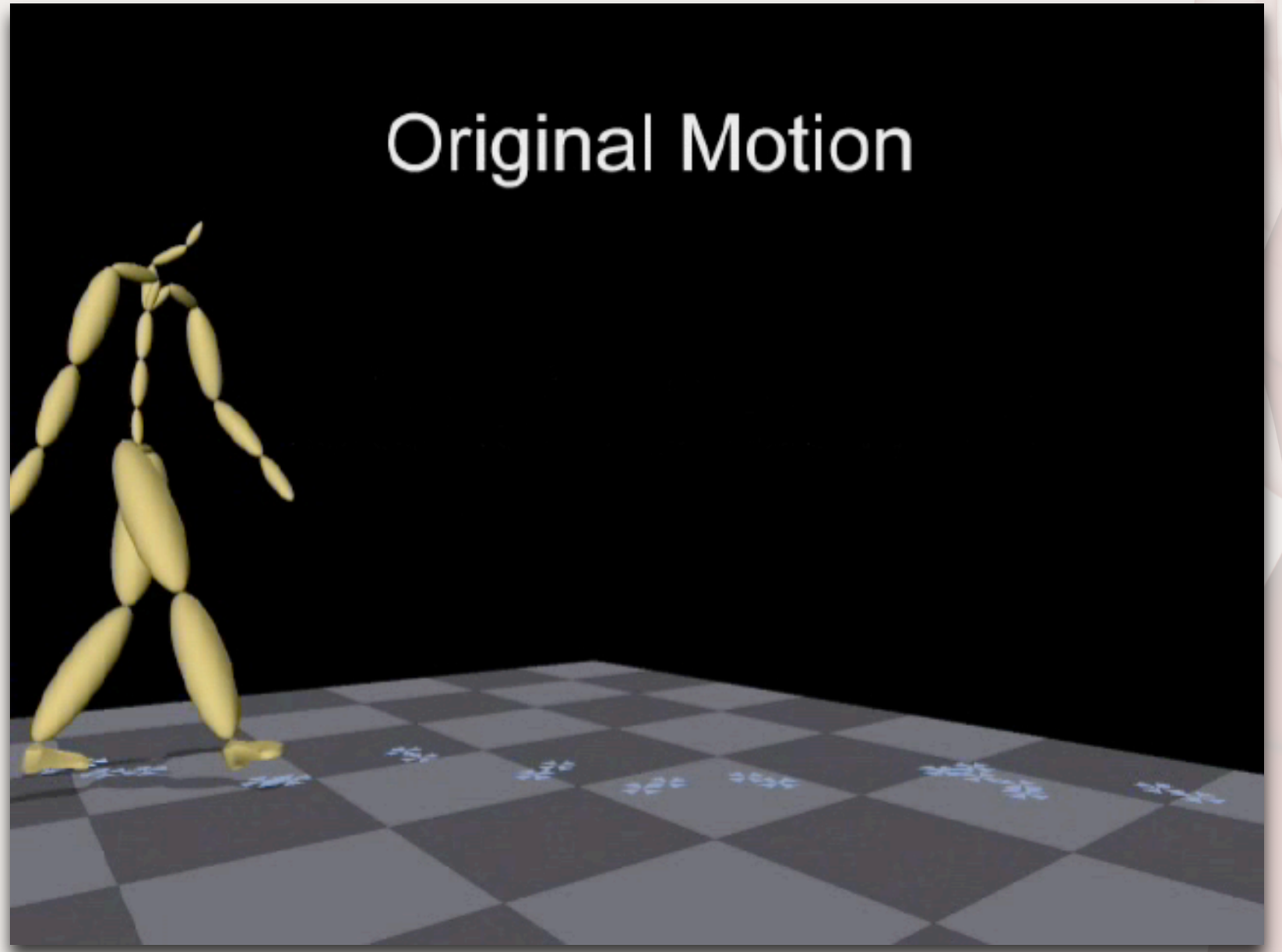


source: Kovar et al. [2002]

Constraints

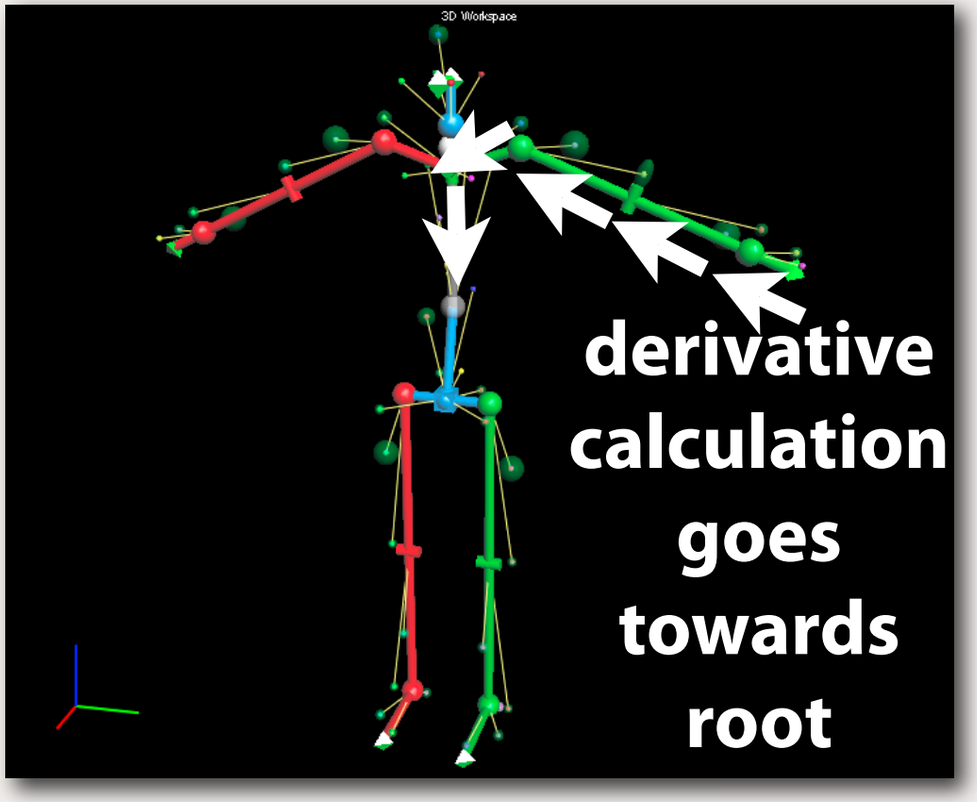
- Pose blending may violate physical constriants
  - Linear Momentum Conservation
  - Angular Momentum Conservation
  - Frictional Constraints (“Foot Skate”)

“Foot Skate” Poblem



source: http://www.cs.wisc.edu/graphics/Gallery/kovar.vol/Cleanup/

Inverse Kinematic Solution

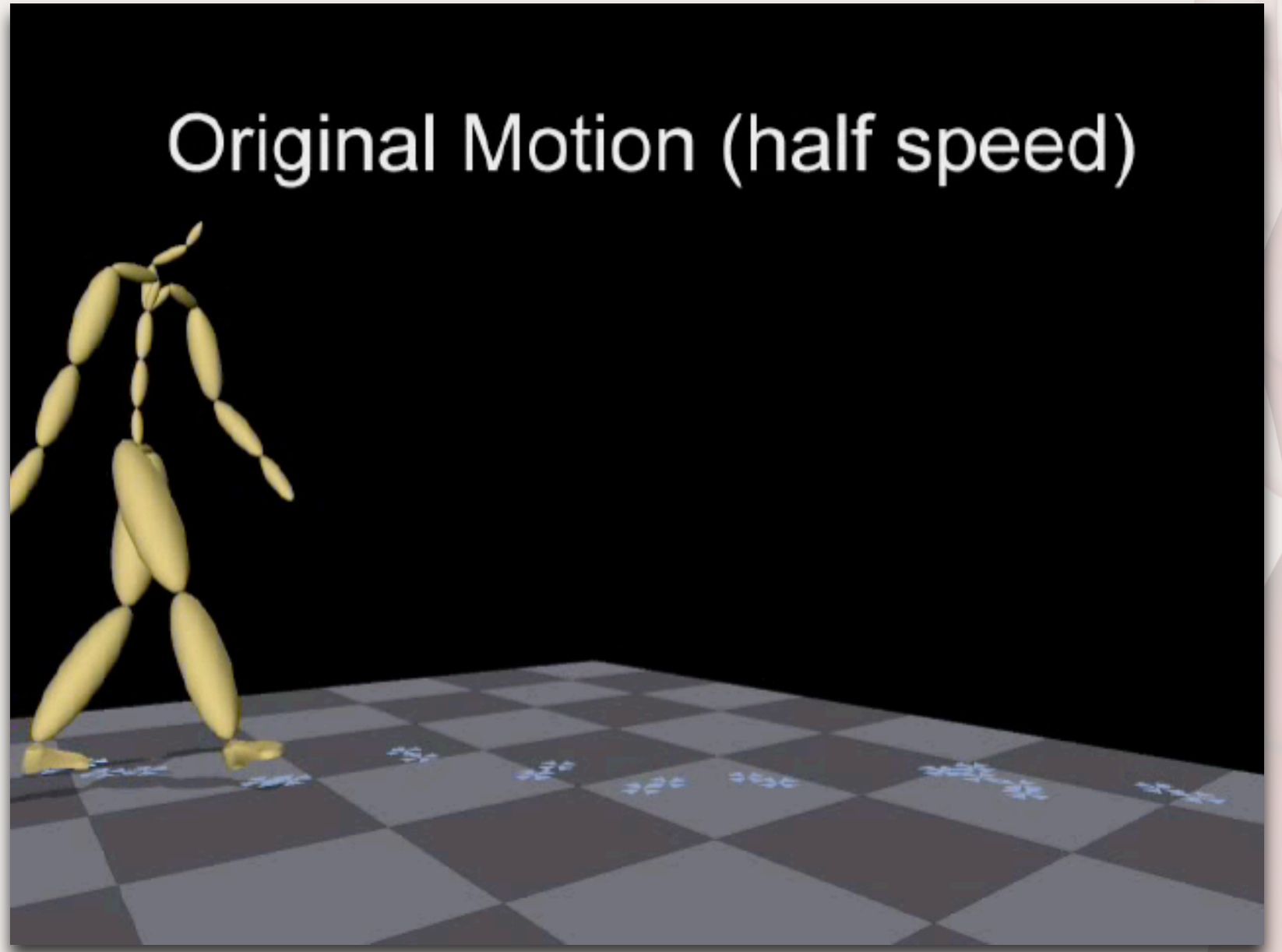


$$\omega_i = f_{i,\Omega}(\omega_{i-1})$$

$$\frac{dE}{d\Omega} = 2\sum_j (\hat{\mathbf{m}}_j^\star - \hat{\mathbf{m}}_j)^T \frac{d\hat{\mathbf{m}}_j}{d\Omega}$$

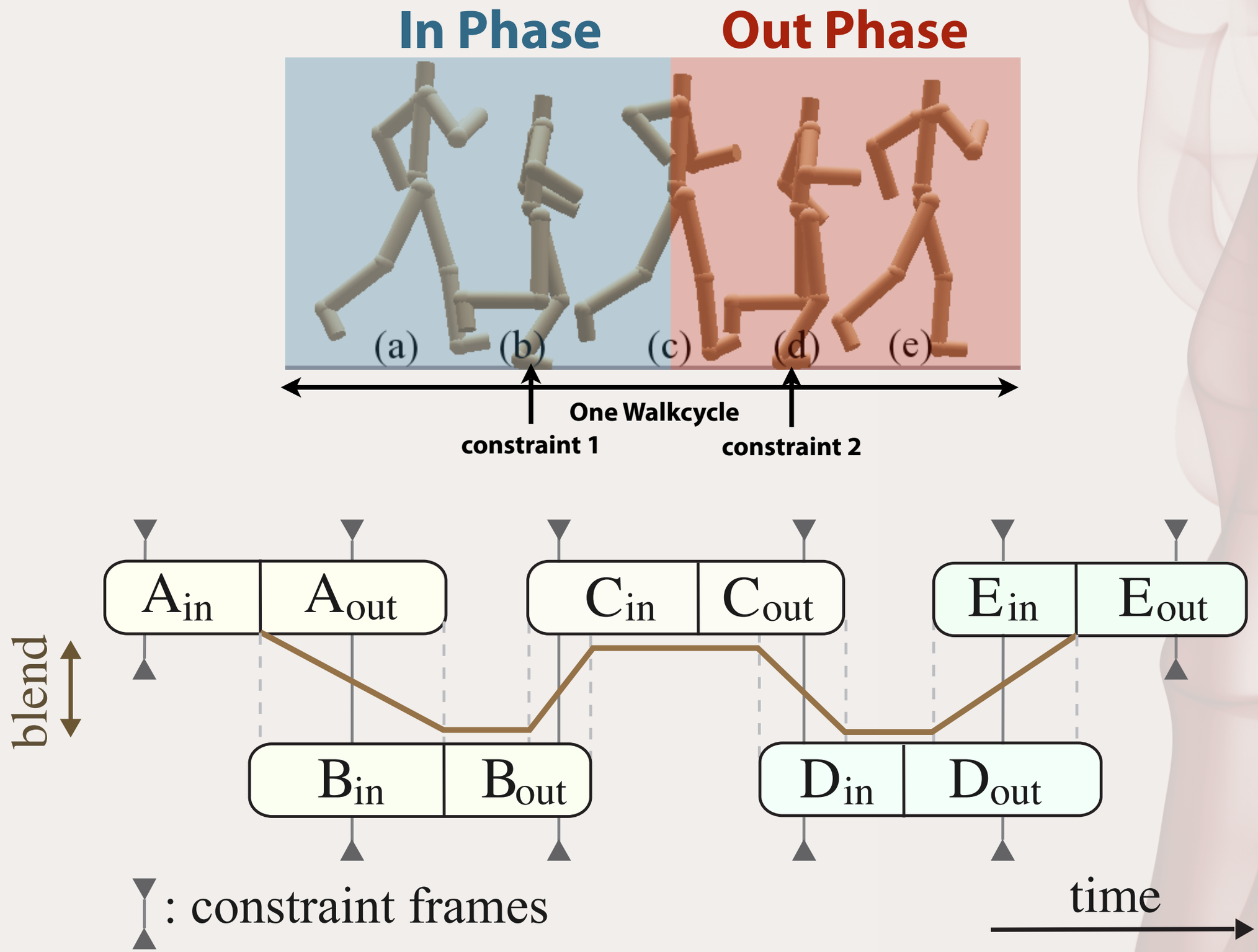
$$\frac{d\hat{\mathbf{m}}_j}{d\Omega} = \frac{\partial \hat{\mathbf{m}}_j}{\partial \omega_i}\left(\frac{\partial \omega_i}{\partial \Omega} + \frac{\partial \omega_i}{\partial \omega_{i-1}}\frac{\partial \omega_{i-1}}{\partial \Omega} + \frac{\partial \omega_i}{\partial \omega_{i-1}}\frac{\partial \omega_{i-1}}{\partial \omega_{i-2}}\frac{\partial \omega_{i-2}}{\partial \Omega} + \cdots\right)$$

IK Results

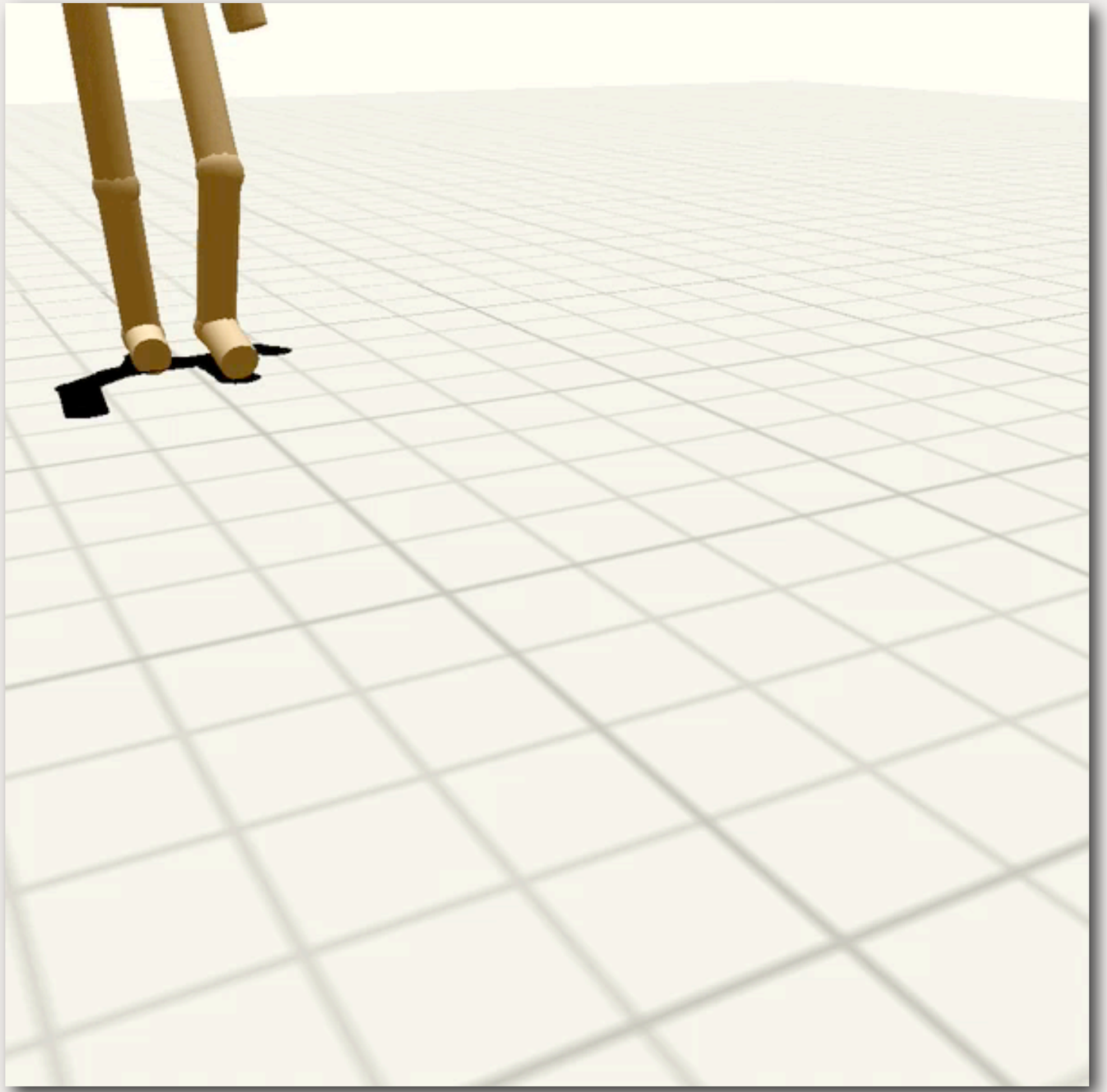


source: http://www.cs.wisc.edu/graphics/Gallery/kovar.vol/Cleanup/

Smart Blending



Smart Blending Example



source: Treuille et al. [2002]

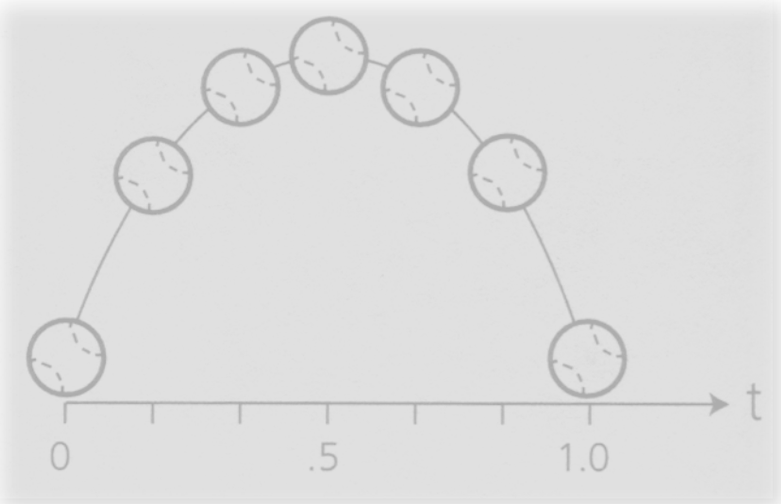
Overview

Announcements

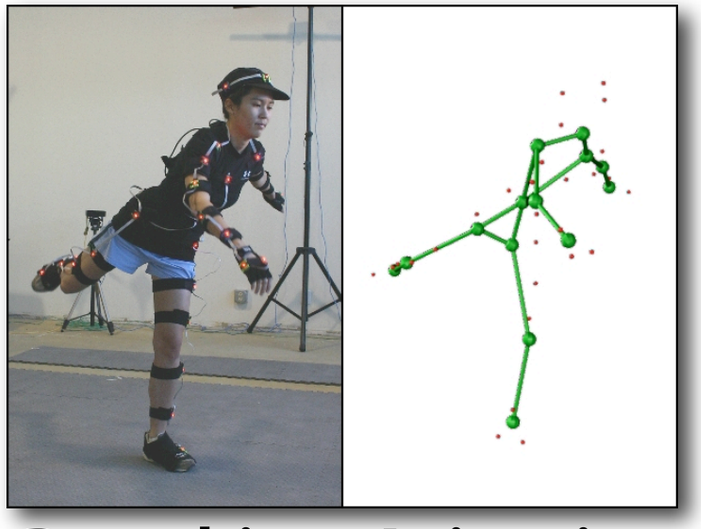
Radiosity

Animation Intro

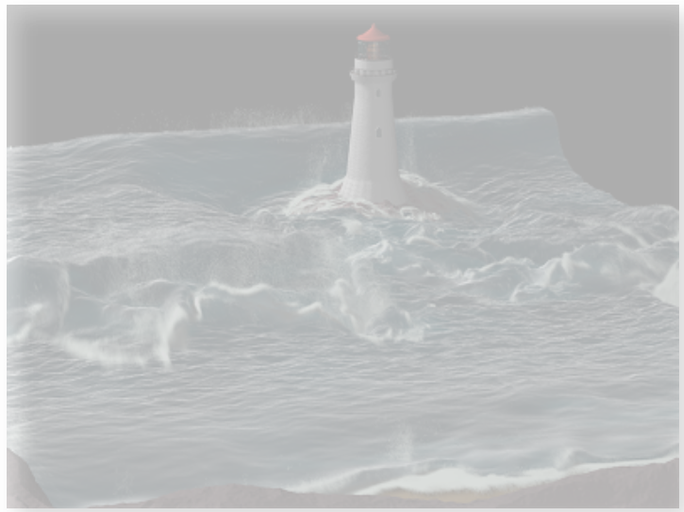
Cell Animation



Keyframing



Data-driven Animation



Physical Simulation

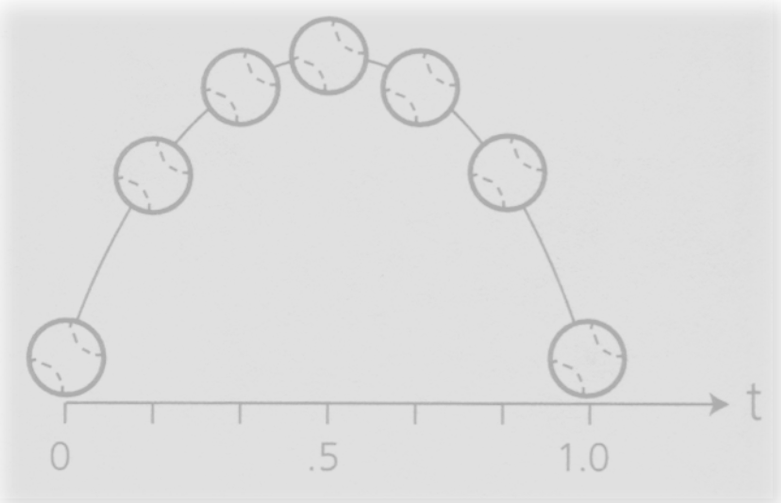
Overview

Announcements

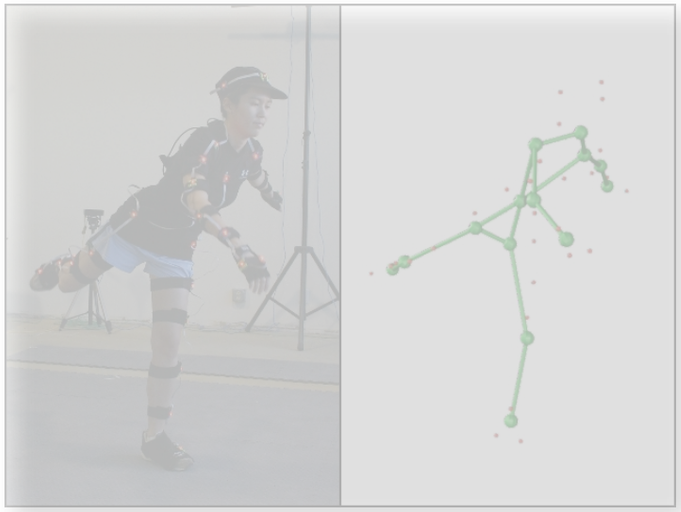
Radiosity

Animation Intro

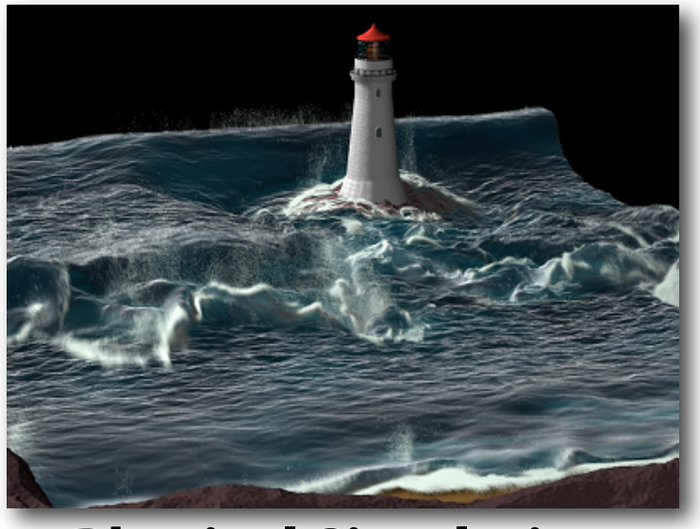
Cell Animation



Keyframing

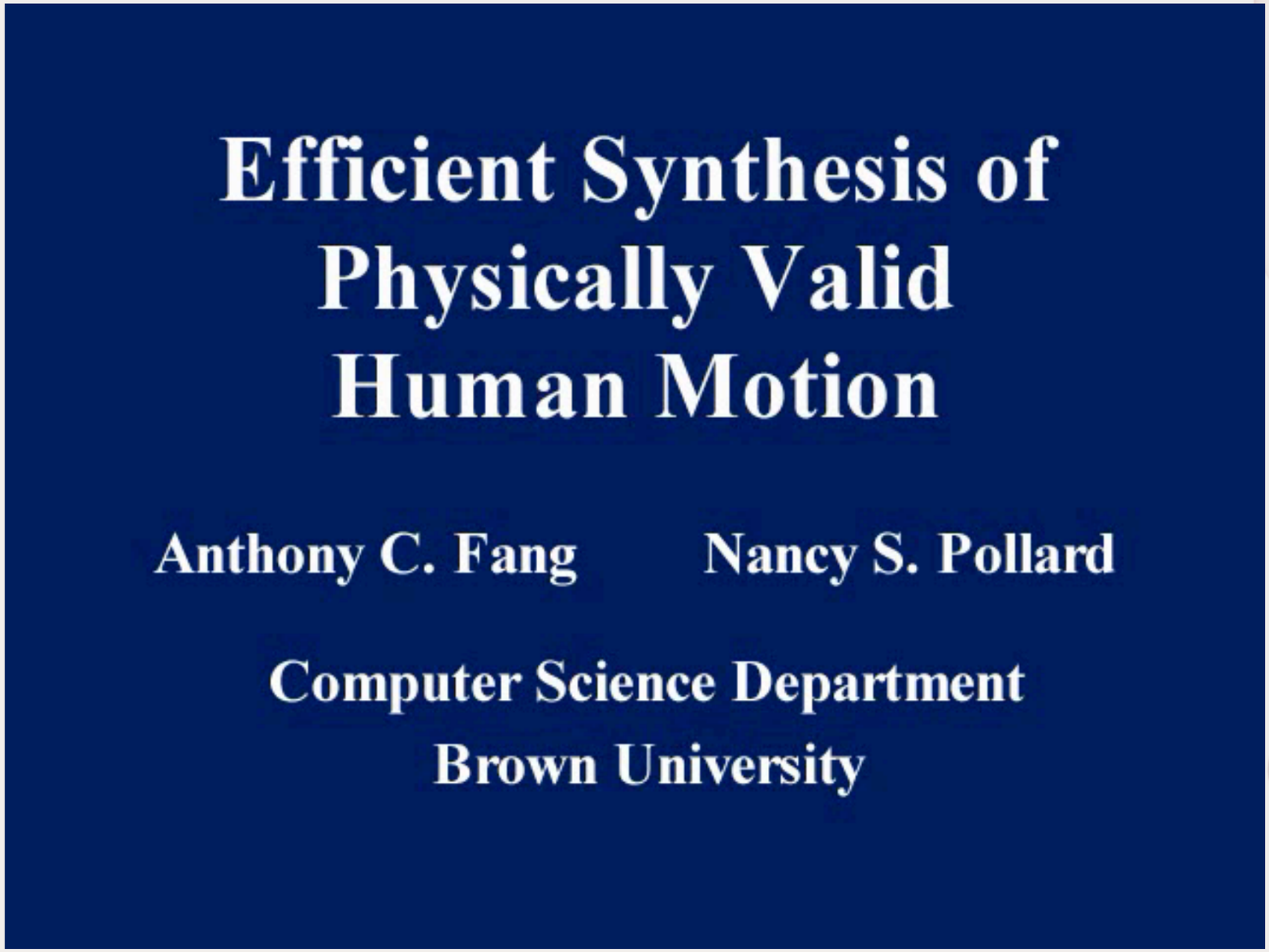


Data-driven Animation

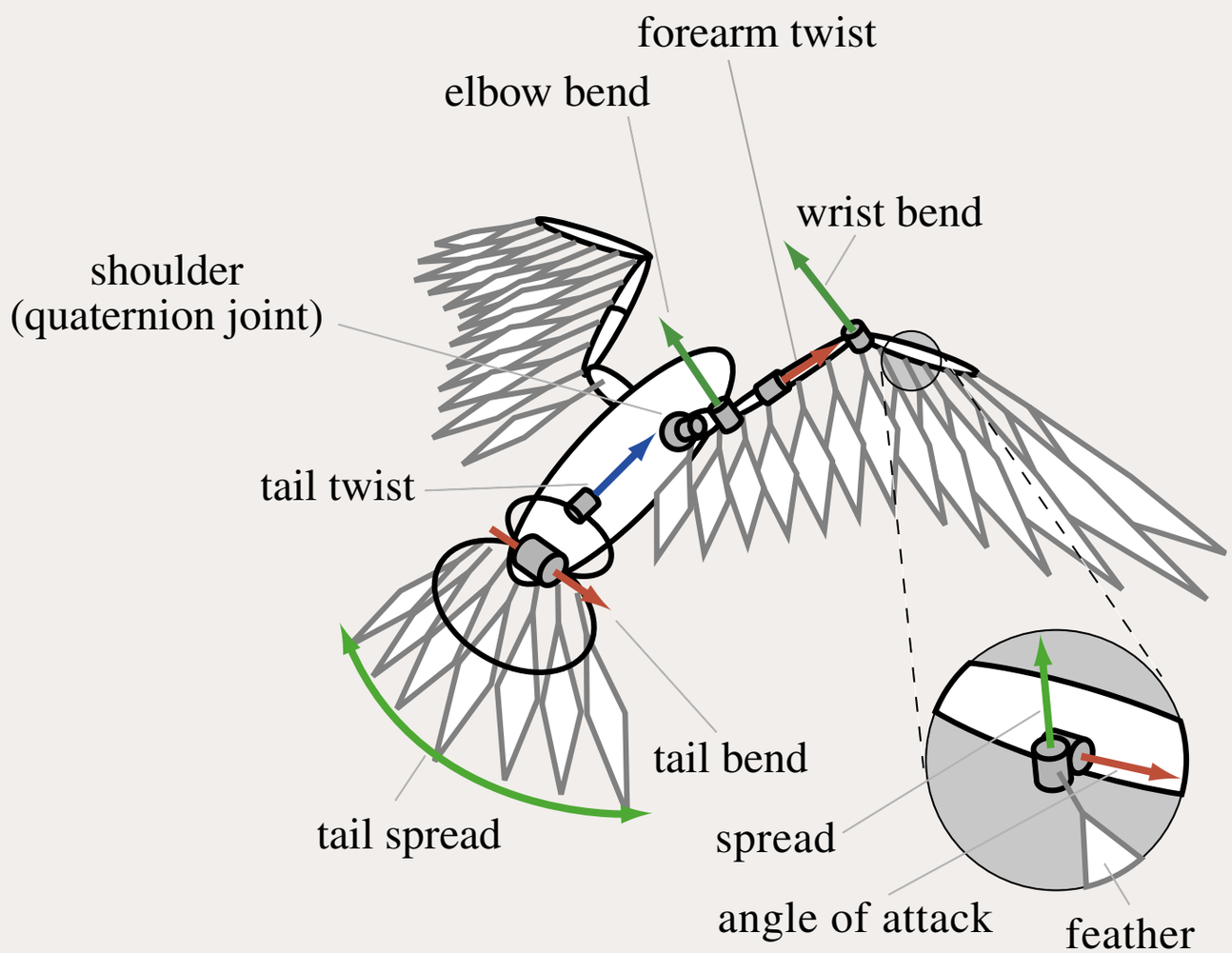


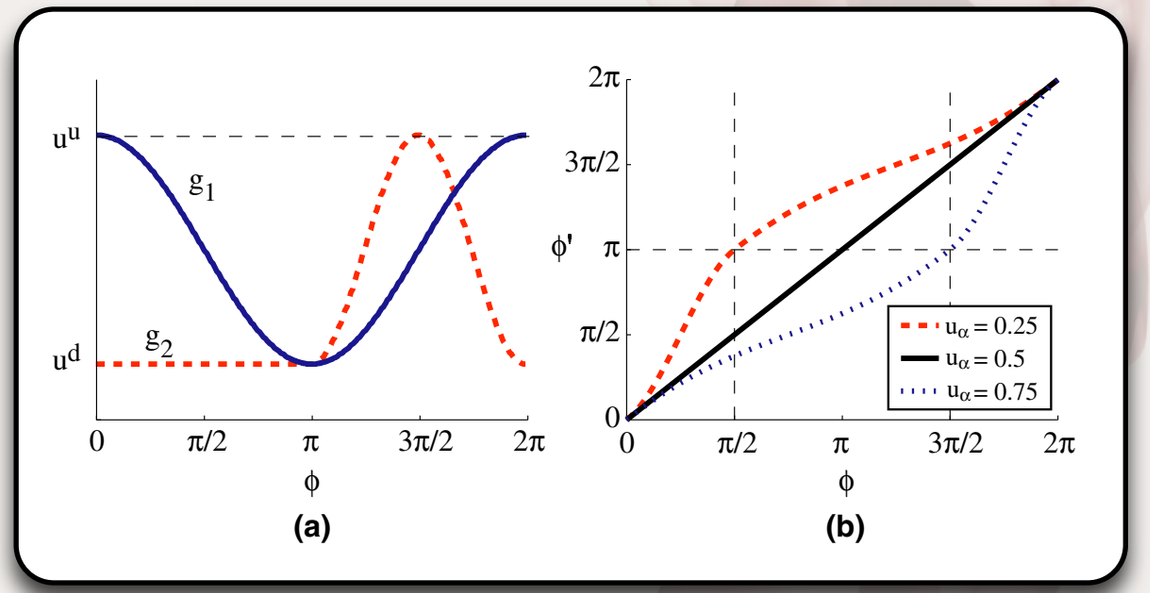
Physical Simulation

Examples


Efficient Synthesis of
Physically Valid
Human Motion
Anthony C. Fang
Nancy S. Pollard
Computer Science Department
Brown University

Bird Flight





source: Wu and Popović [2003]

Bird Flight Examples

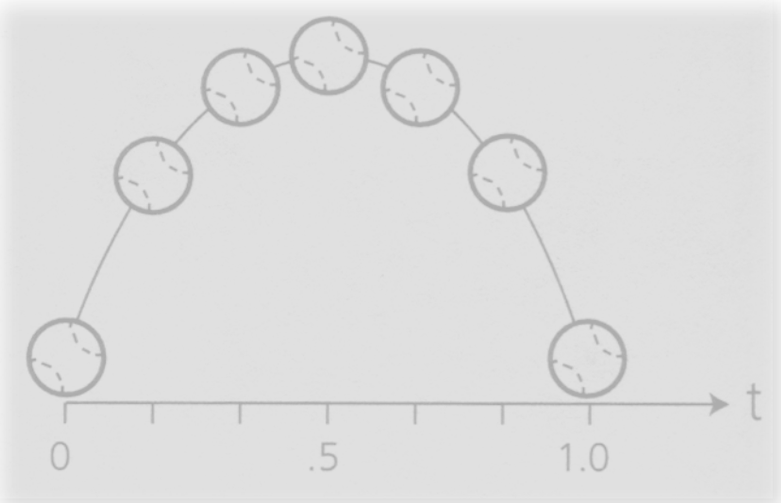
Overview

Announcements

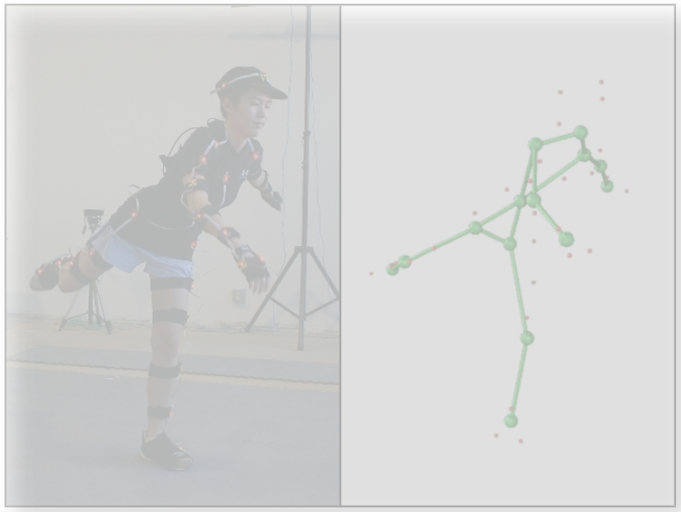
Radiosity

Animation Intro

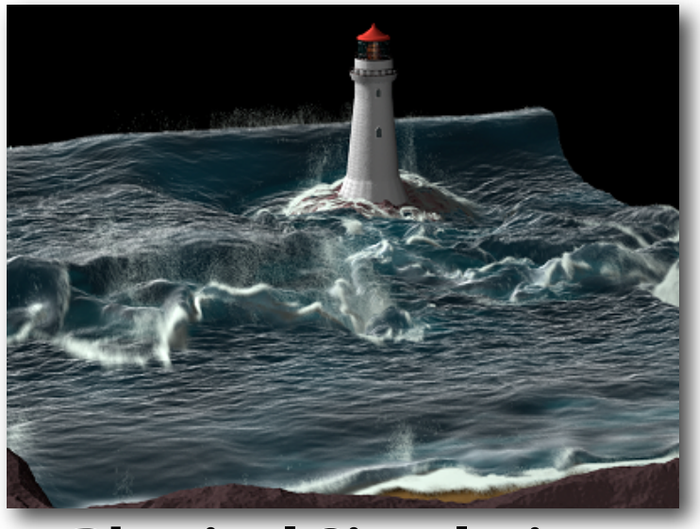
Cell Animation



Keyframing



Data-driven Animation



Physical Simulation

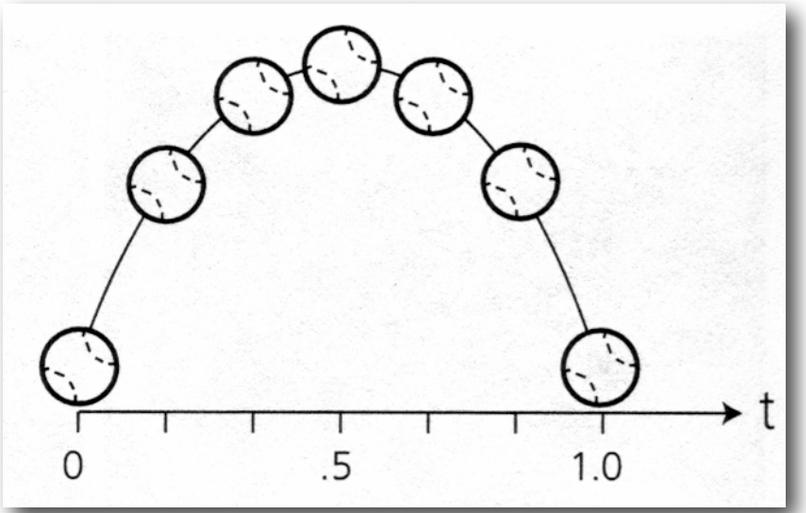
Overview

Announcements

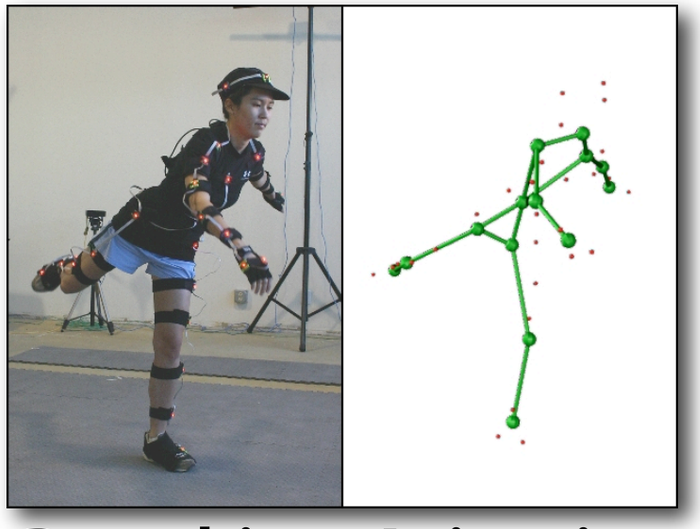
Radiosity

Animation Intro

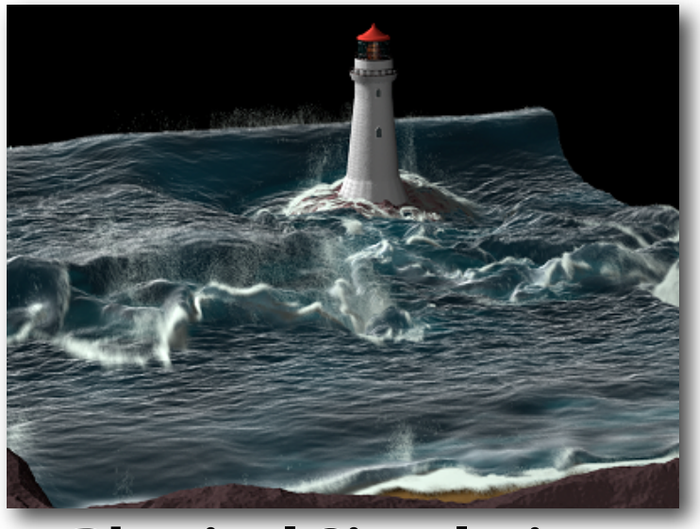
Cell Animation



Keyframing



Data-driven Animation



Physical Simulation