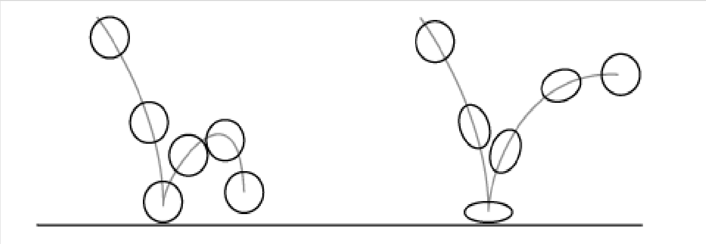
Animation

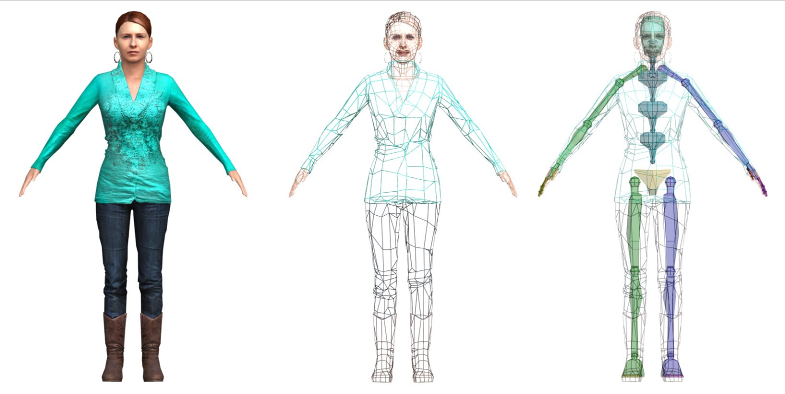
Lecturer: Rachel McDonnell
Assistant Professor of Creative Technologies
Rachel.McDonnell@cs.tcd.ie

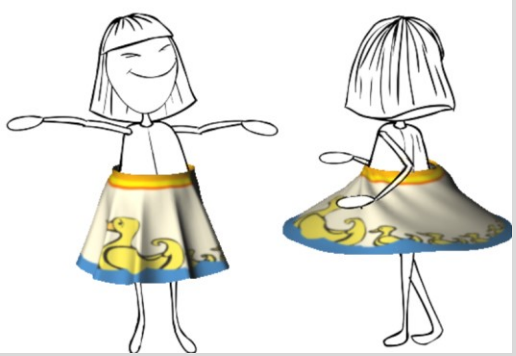
Course www: https://www.scss.tcd.ie/Rachel.McDonnell/

Overview

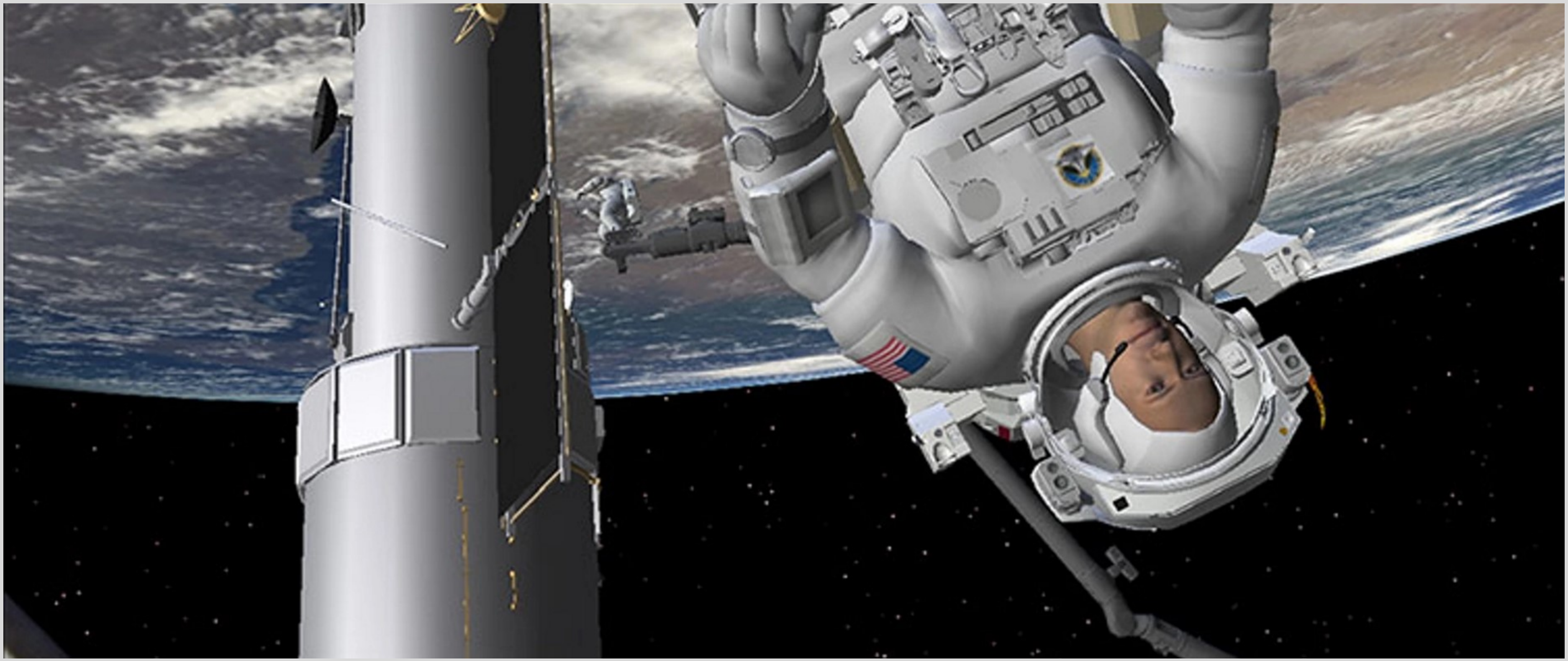
- Animating Objects
  - Interpolating
  - Splines
- Character Animation
  - Mocap
  - Kinematics
- Physically based animation
  - Cloth, Hair, Fur Water, etc.







Animation vs. Rendering



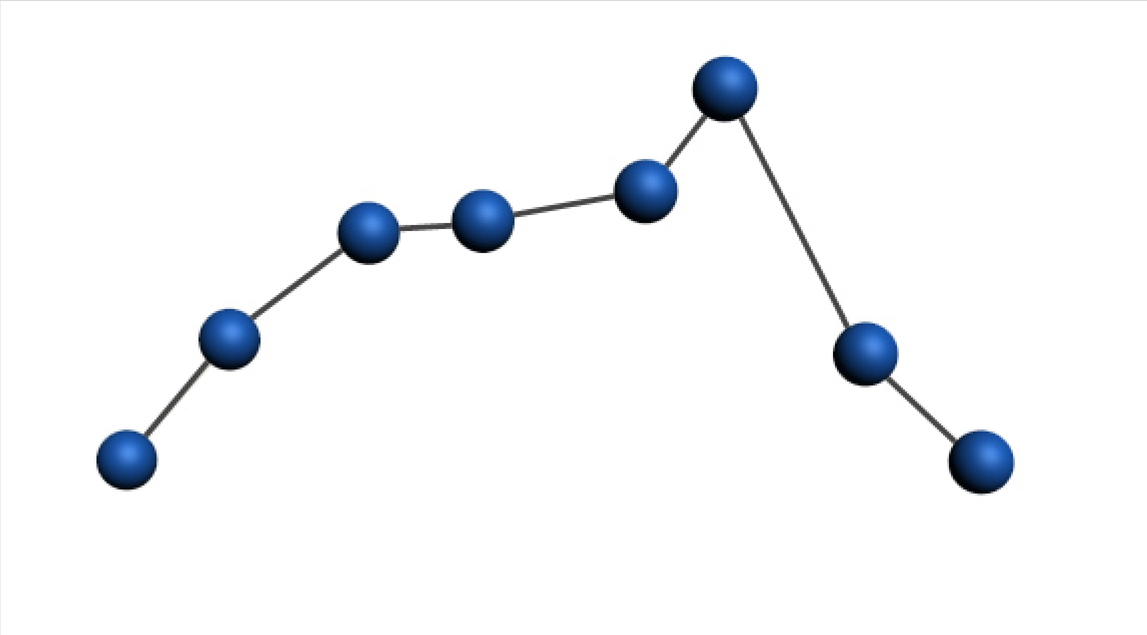
Gravity, pre-vis, 2014

Interpolating Values

- Animator has a list of values associated with a given parameter at key frames
- How best to generate the values of the parameter for frame between keyframes?

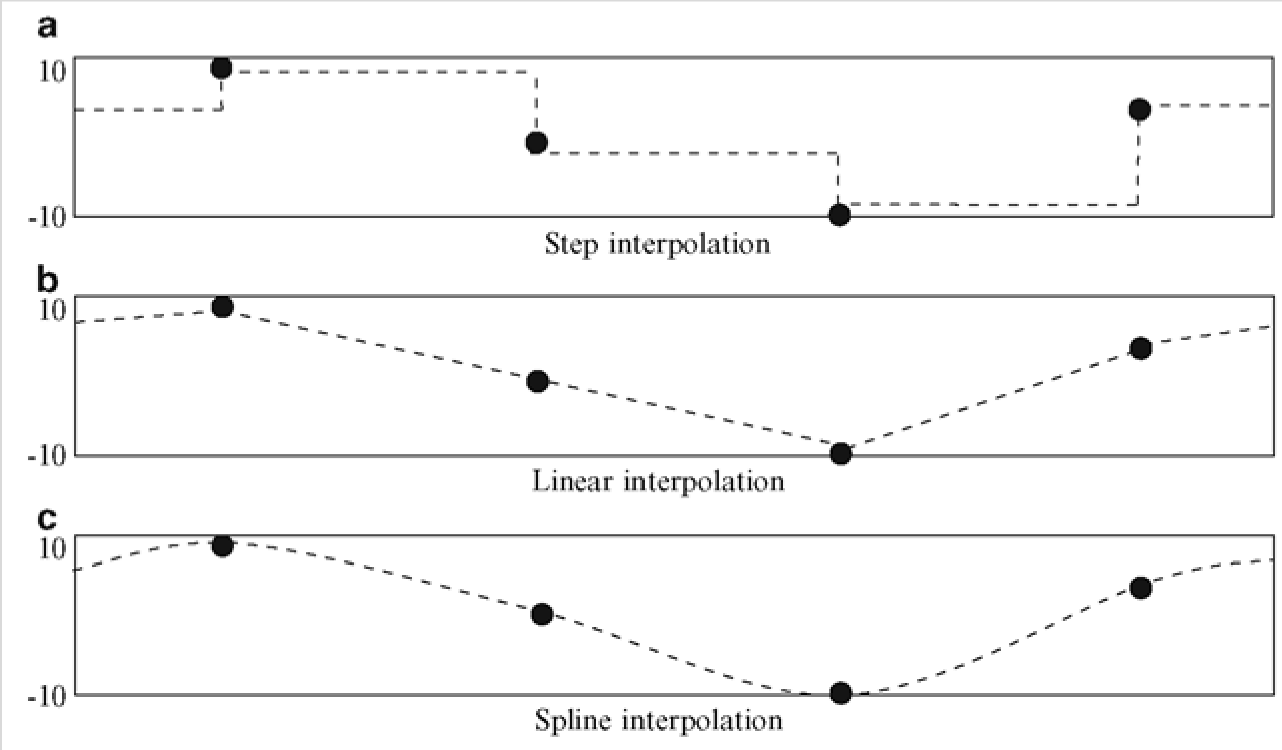
Keyframing in 3D

- Animator specifies the important keyframes
- Computer generates the in-betweens automatically using **interpolation methods**



Uses - Keyframe animation

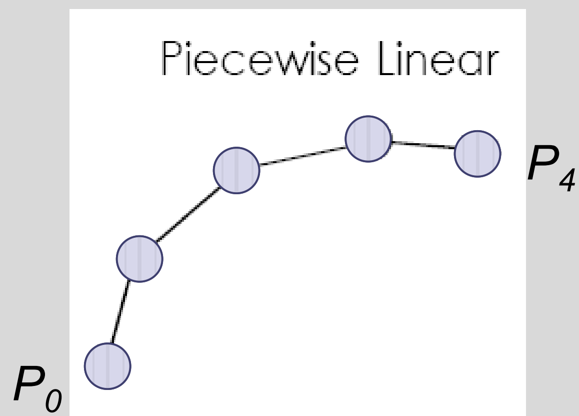
Keyframe animation



Linear interpolation: *N* Points

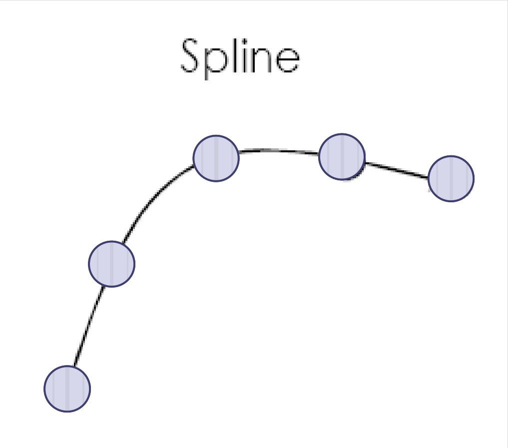
- Connect straight lines between data-points
- Given $P_0 \ldots P_N$, define segment:

$$L_i(t) = (1-t)\, P_i + t\, P_{i+1} \qquad t \text{ in } [0,1],\ i \text{ in } [0,N]$$



Spline Curves

- To make smooth curves from data points
- Many types of splines, many properties:
  - Interpolating, approximating, …
- Build an order-*k* polynomial from *k+1* points



Interpolation

- Not fool-proof
- May not follow the laws of physics
- Splines may undershoot & cause interpenetration
- Animator must also keep an eye out for these types of side-effects.

Representing Curves

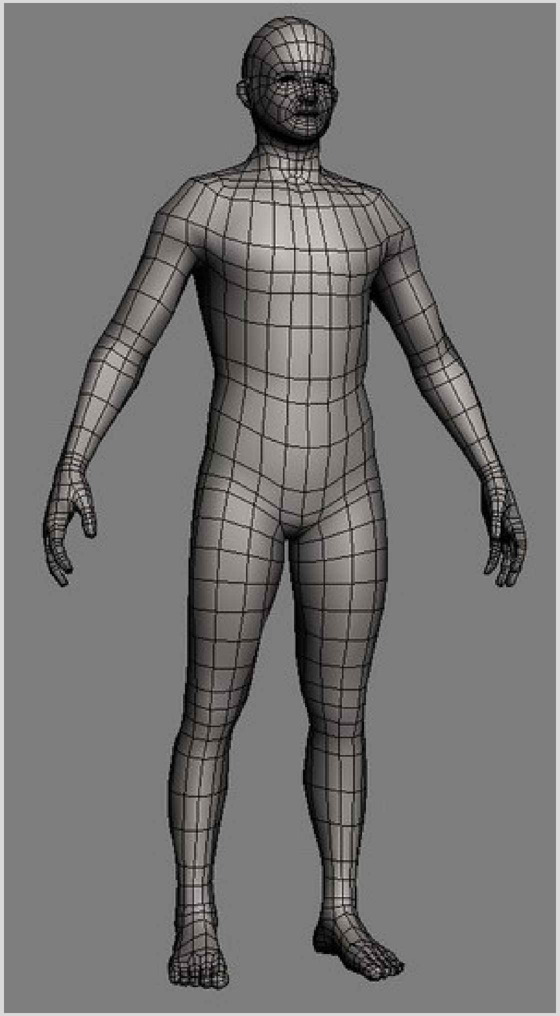
There are many different methods of representing general curves (rather than attempt to model all surfaces as some existing function, say a Sine or Cosine). The most common are:

- **Cubic Splines**
- **Bezier Curves**
- **B-splines**

- **(Not covering them in detail this year)**

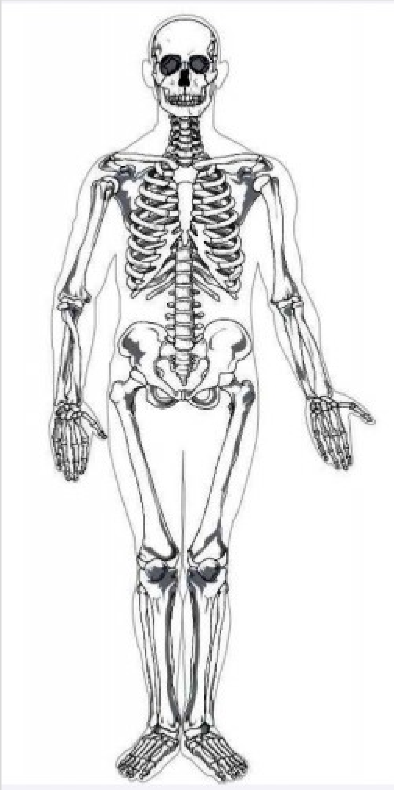
Character Animation

- Animating a character model described as a polygon mesh by moving each vertex in the mesh is <u>impractical</u>
- Instead - specify the motion of characters through the movement of an internal articulated <u>skeleton</u>
  - Movement of the surrounding polygon mesh may then be deduced
- Mesh must deform in a manner that the viewer would expect, <u>consistent</u> with underlying muscle and tissue

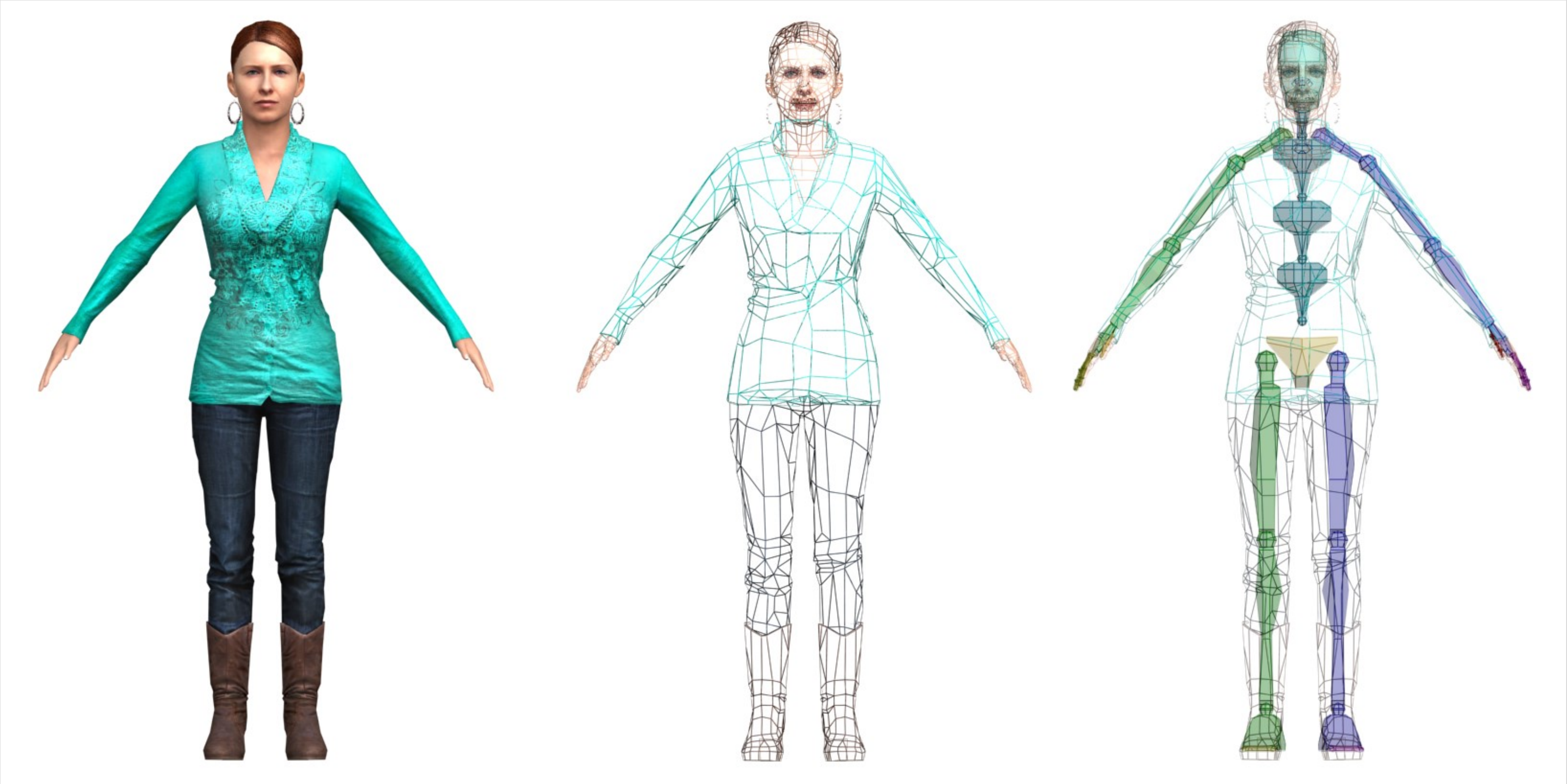


Human Skeleton

- Spine
  - Impractical to model each vertebra
  - Typically use 3/4 spine links
- Shoulders
  - Can translate as well as rotate
  - Wide range of motion
  - Prone to dislocation



Skeleton



Rigid Body Limitations

- Consider human joints:
  - When they bend, the body shape bends as well
    - No distinct parts
- We cannot represent this with rigid bodies
  - Or the pieces would separate, where there should be stretching or compression

Skinning

- Skinning is the process of attaching a renderable skin to an underlying articulated skeleton.
- **Binding** refers to the initial attachment of the skin to the underlying skeleton and assigning any necessary information to the vertices
  - Each vertex in the mesh can be attached to more than one joint, each attachment affecting the vertex with a different strength or weight.

Linear Blend Skinning

- Used in games
  - Linear blend skinning
  - Skeletal subspace deformation
  - Enveloping
  - Vertex Blending
- Linear blend skinning determines the new position of a vertex by **linearly combining** the results of the vertex transformed rigidly with each bone.
- A scalar weight w_i is given to each influencing bone and the weighted sum gives the vertex's position in the new pose
- Weights set such that sum of all weights for a vertex = 1

What is Motion Capture?

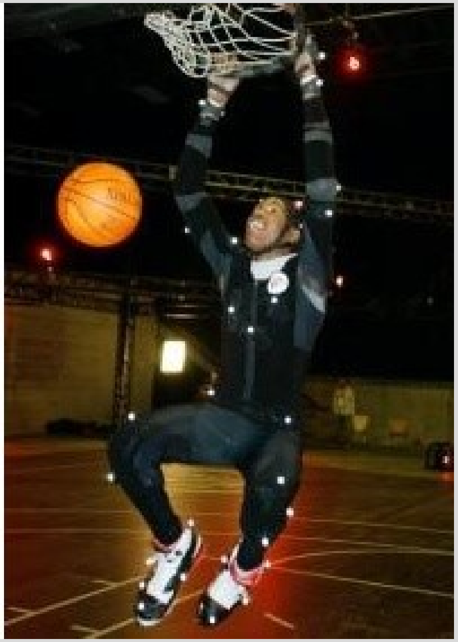
- The process of translating a live performance into a digital performance

Mocap vs Traditional animation

- Advantages
  - Realistic human motion
  - More rapid results can be obtained
  - The amount of work does not vary with the complexity or length of the performance
  - Complex movement and realistic physical interactions can be easily re-created
  - Mocap technology allows one actor to play multiple roles within a single film.

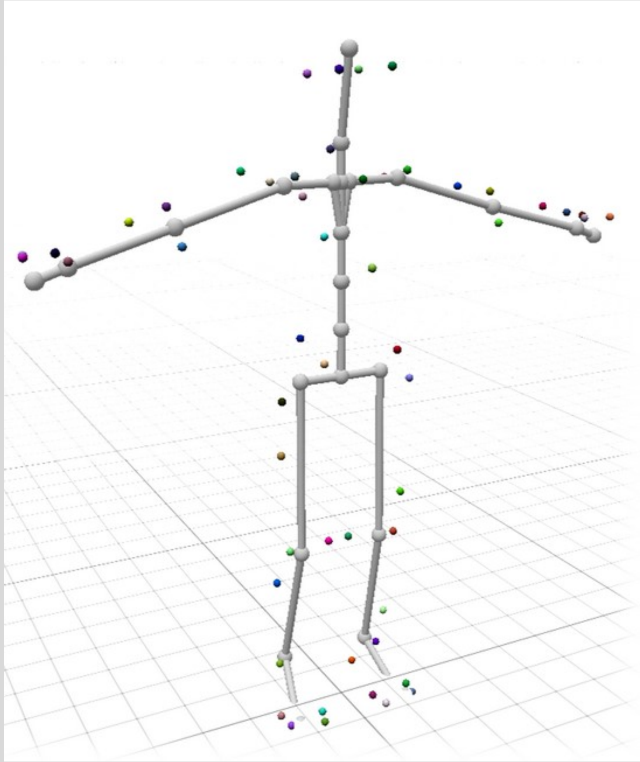
Optical Motion Capture

- Reflective markers
- Actor is shot by multiple cameras
- Each camera has a light source
- Light is reflected by the markers back to the camera
- The 3D location of the markers are computed by stereo vision



Computing the joint angles

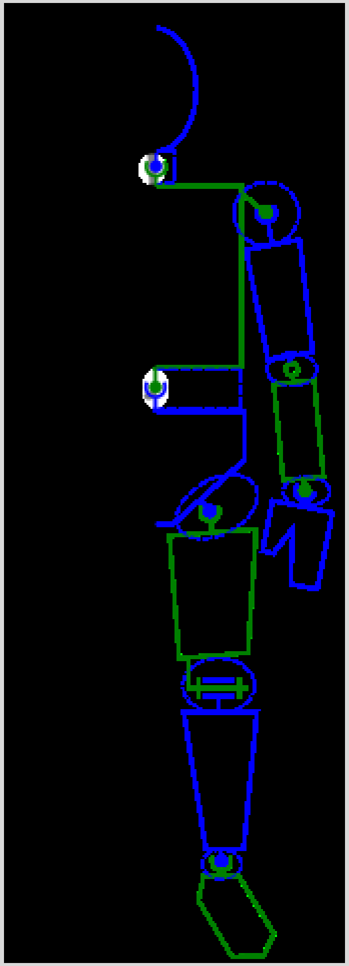
- The joint angles are computed, based on the marker positions



Kinematics

Kinematics

Describes the positions of the body parts as a function of the joint angles.

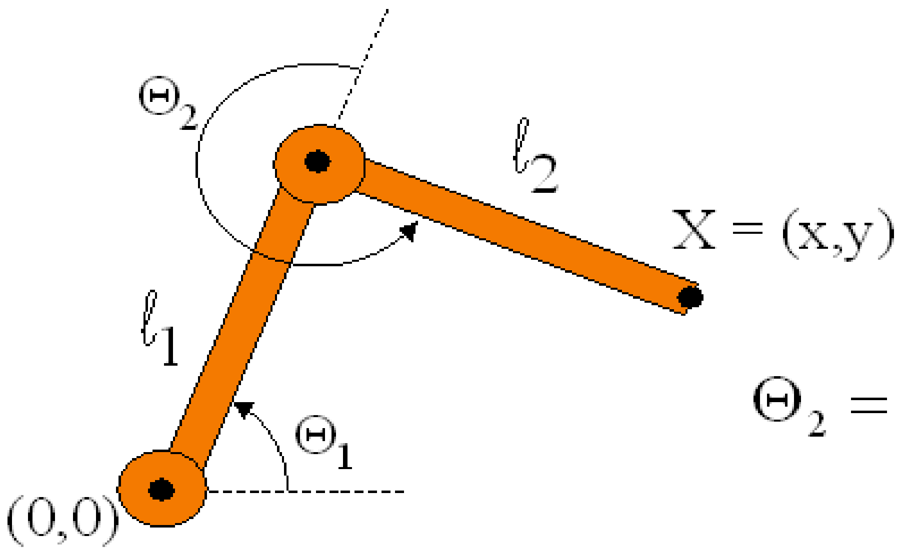


Kinematics

- Forward kinematics
  - Low level approach where animator has to explicitly specify all motions of every part of the animated structure
  - Each node in hierarchy inherits movement of all nodes above it
- Inverse kinematics
  - Requires only the position of the ends of the structure
  - Functions as black box - controls detailed movement of entire structure

Inverse Kinematics

- Animator specifies end-effector positions
- Computer finds joint angles

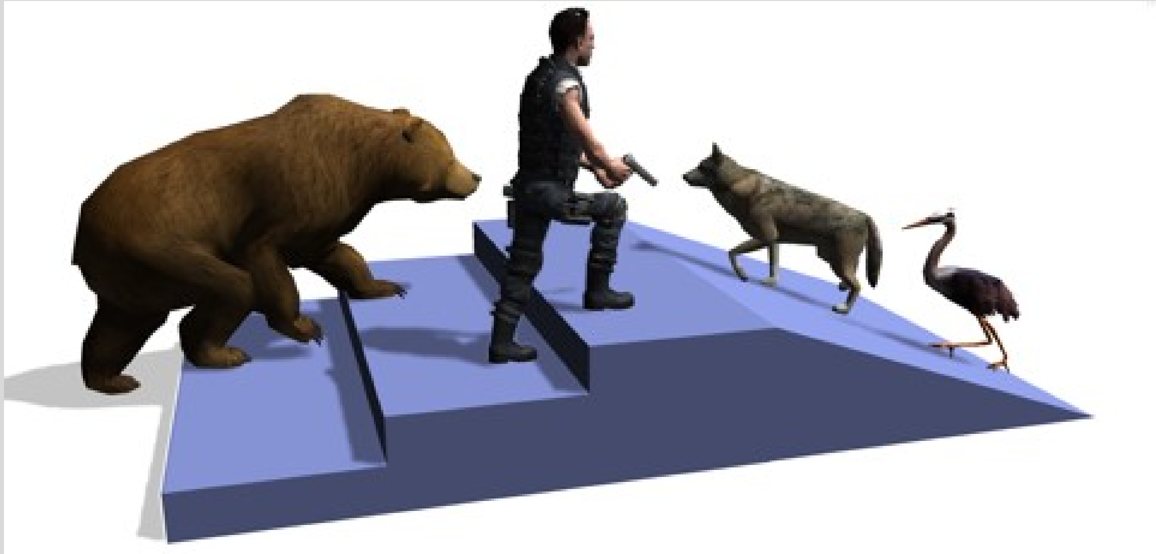


$$\Theta_2 = \cos^{-1}\left(\frac{x^2 + x^2 - l_1^{\,2} - l_2^{\,2}}{\mathbf{2}l_1 l_2}\right)$$

$$\Theta_1 = \frac{-(l_2 \sin(\Theta_2)x + (l_1 + l_2 \cos(\Theta_2))y}{(l_2 \sin(\Theta_2))y + (l_1 + l_2 \cos(\Theta_2))x}$$

Inverse kinematics

- Goals
  - Keep end of limb fixed while body moves
  - Position end of limb by direct manipulation
  - (More general: arbitrary constraints)



What makes this hard?

- Not always a unique solution
- Not always well-behaved
- Nonlinear problem
- Joint limits

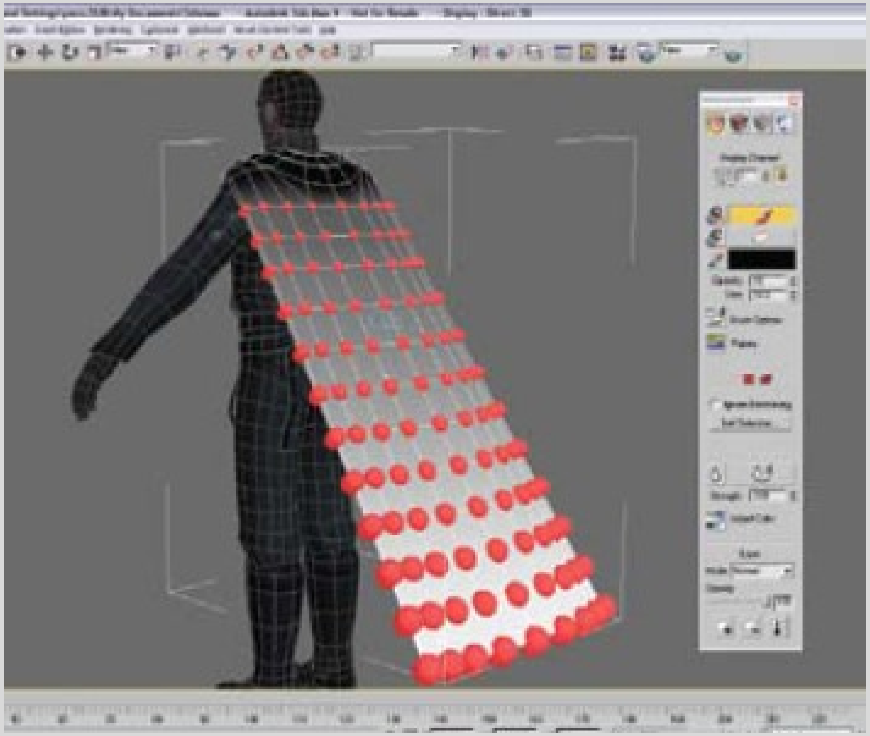
Physically based animation

Physically based animation

- Forces are used to maintain relationships among geometric elements
- Modelling the physics usually incurs a high computation expense, but is flexible
- Example: for cloth, an animator can set parameters that indicate type and thickness of cloth material, and wrinkles occur naturally, rather than specifying exact positions of wrinkles

Cloth Simulation

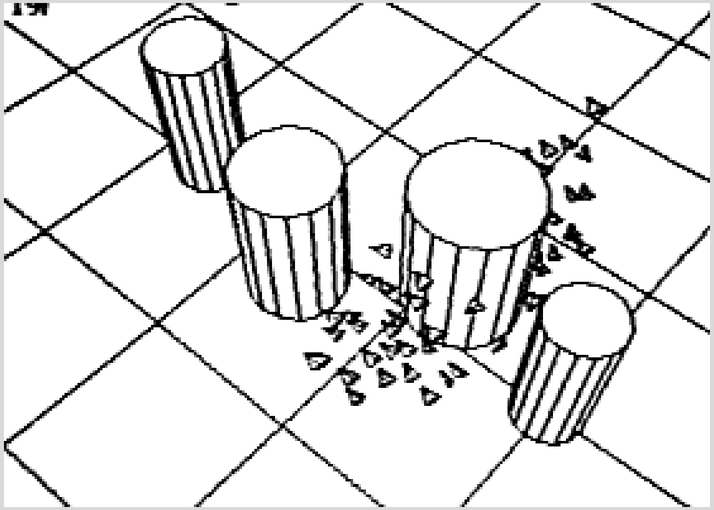
Cloth Simulation



Hair Simulation

Behavioural Animation

- A type of procedural animation
  - Autonomous character determines its own actions to some extent
  - This gives the character some ability to improvise (obstacle avoidance and goal seeking)



Simulated boid flock avoiding cylindrical obstacles (1986)

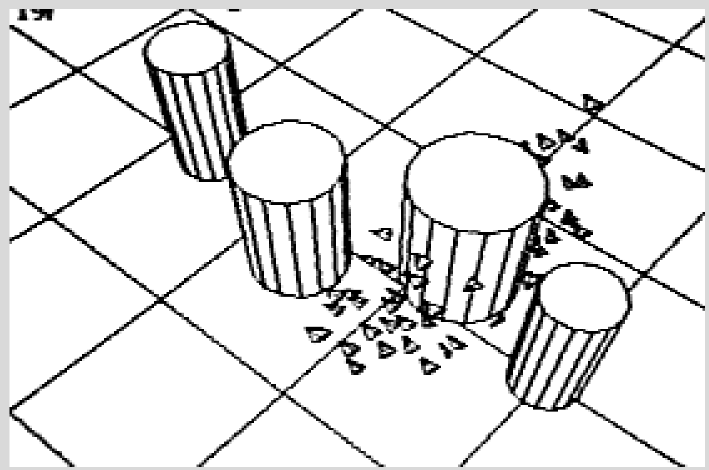
Crowds

Craig Reynolds and "Boids"

Craig Reynolds is a computer graphics researcher, who revolutionised animation in games and movies with his classic paper :

Reynolds, C. W. (1987) Flocks, Herds, and Schools: A Distributed Behavioral Model, in *Computer Graphics*, **21**(4) (SIGGRAPH '87 Conference Proceedings) pages 25-34.

- Before this paper, animations of flocks, swarms, groups, etc. behaved nothing at all like the real thing.
- Nobody knew how to make it realistic!
- Reynold's solved the problem by trying a very simple approach, which was inspired by a sensible view of how animals actually do it.

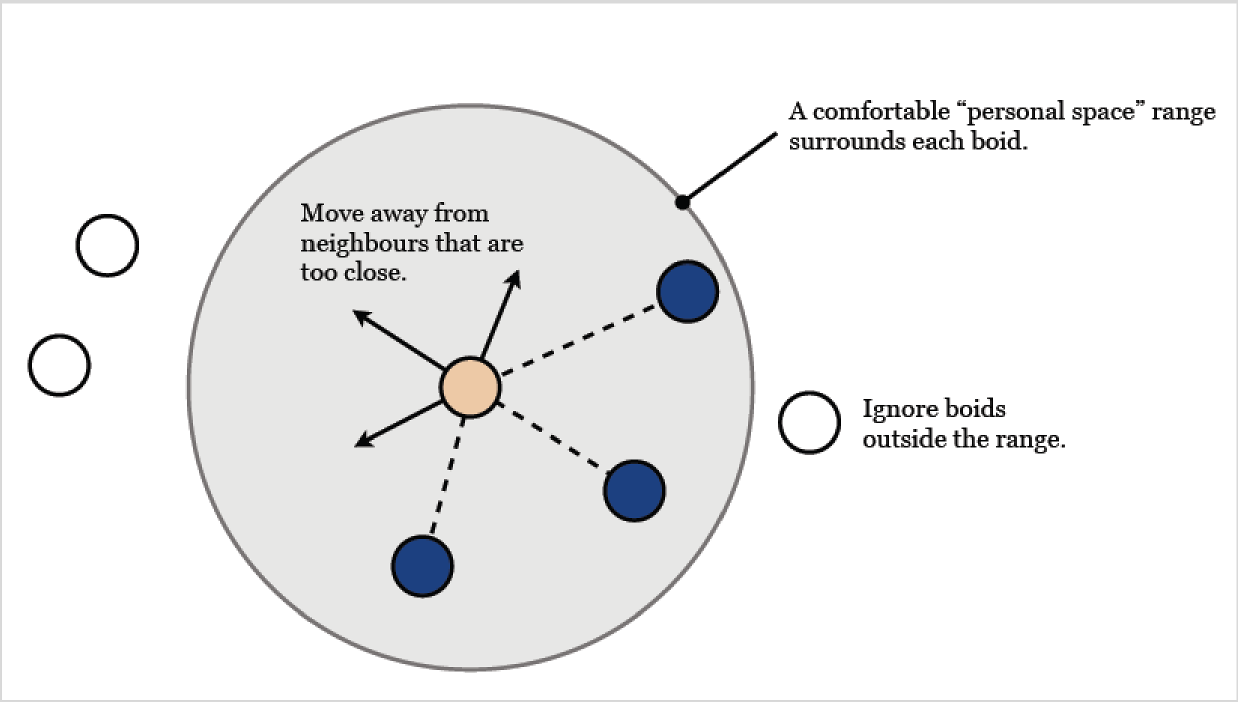


Sensory System

- Each boid has direct access to the whole scene's geometric description, but flocking requires that it **reacts only to flockmates** within a certain small neighborhood around itself.
- The neighborhood is characterized by:
  - a *distance* (measured from the center of the boid) and
  - an *angle,* measured from the boid's direction of flight.
- Flockmates outside this local neighborhood are ignored.

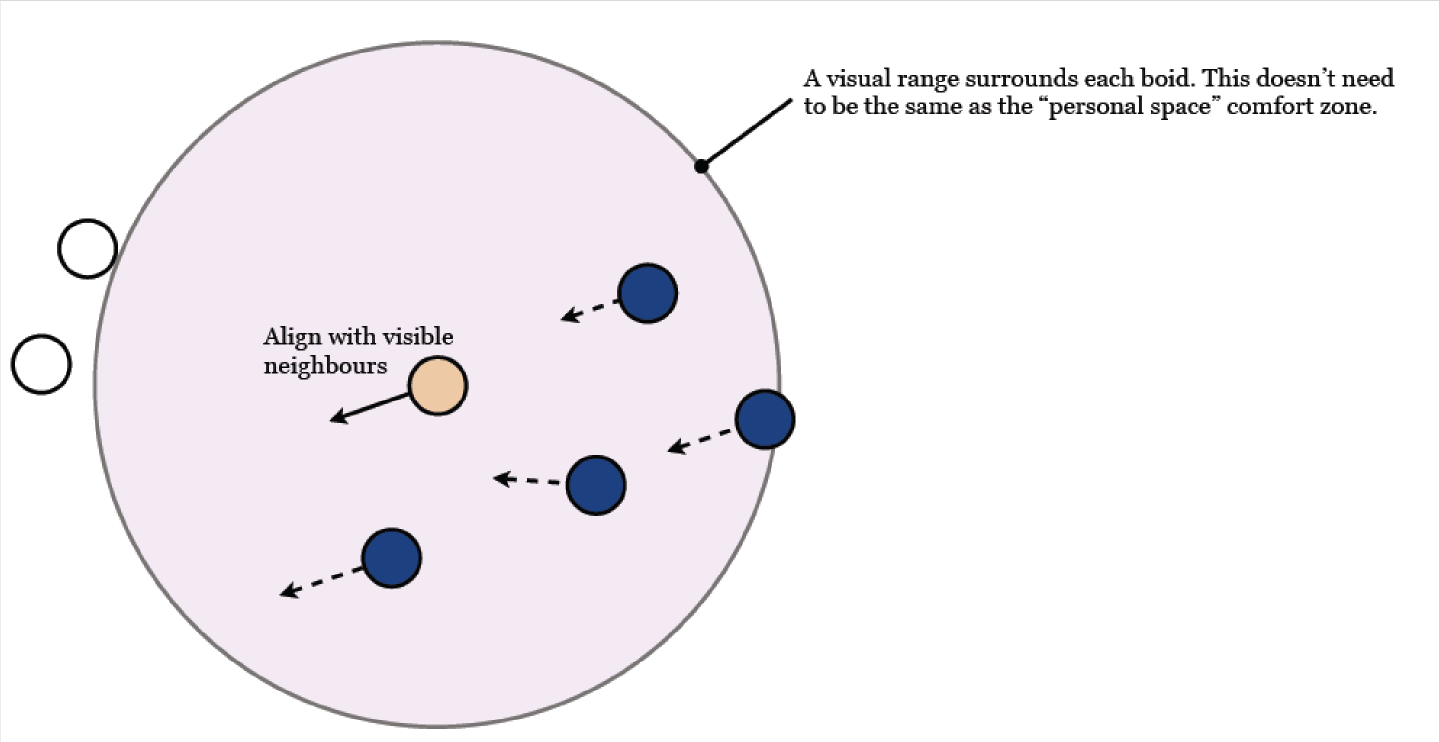
Rule 1: Separation

- An easy way to avoid collisions with a flock of other boids is by keeping your distance from them



Rule 2: Alignment

- Another way to avoid collisions with a flock of other boids is by flying in the same direction as your visible neighbours.



Rule 3: Cohesion

- A flock is a group flying together. We don't want the birds to fly too far apart so we add a localised flock centering tendency.