Animation I

Dr Cengiz Öztireli

UNIVERSITY OF CAMBRIDGE


11:11:28:13
11:11:28:13
11:11:28:13
11:11:28:13

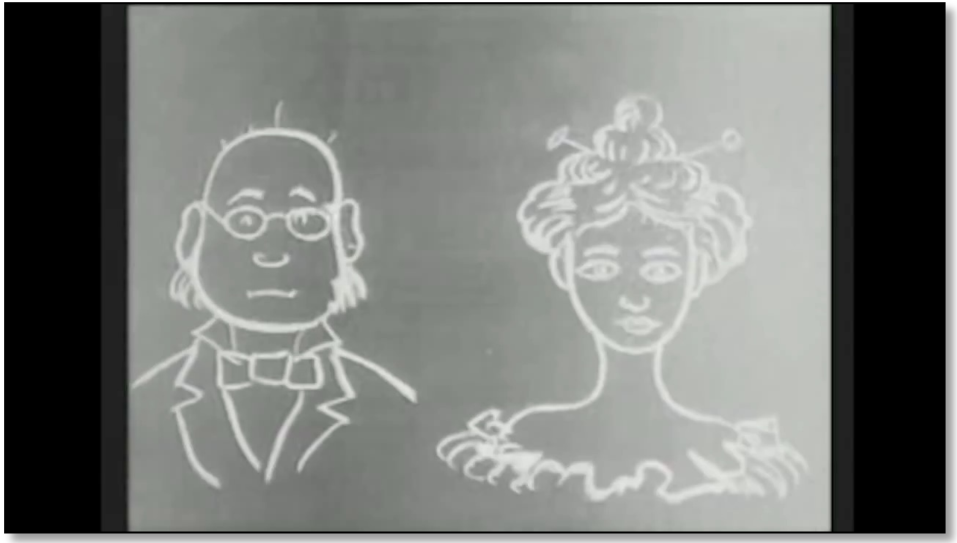
Animation

- Animation (technically): Creating sequences of images
- Animation (artistically): Bringing images to life!
  (anima ~ soul, breath of life)

Animation

- A glimpse on the history of animation

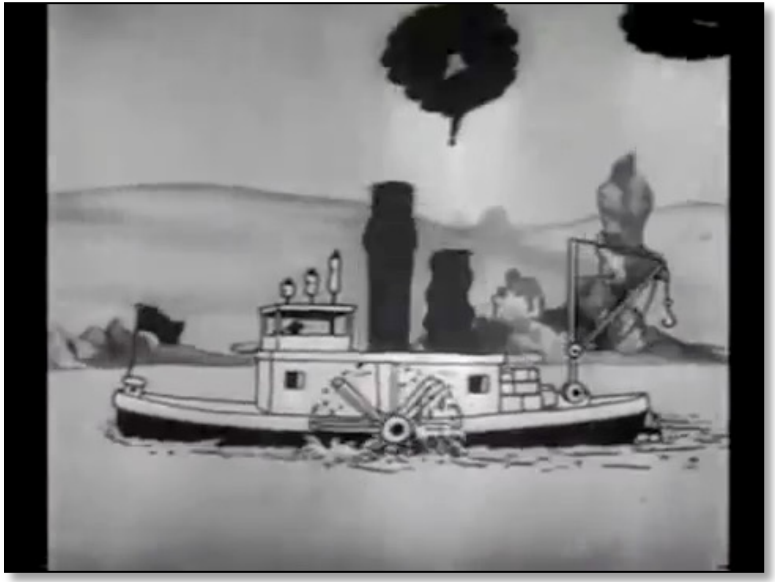
1906

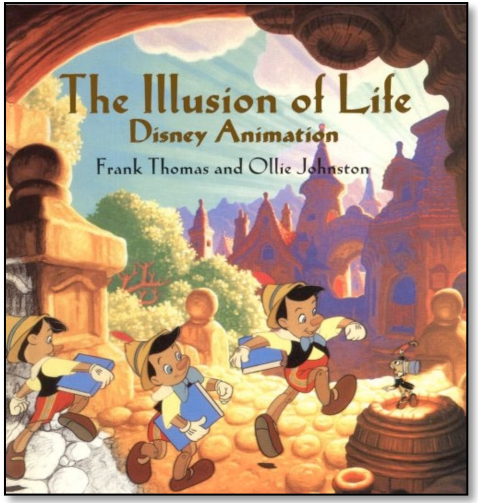


Animation

- A glimpse on the history of animation

1928



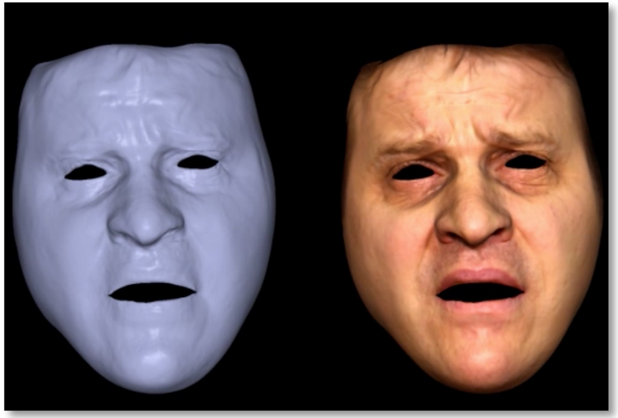


Animation

- What do we animate?

Characters

Faces

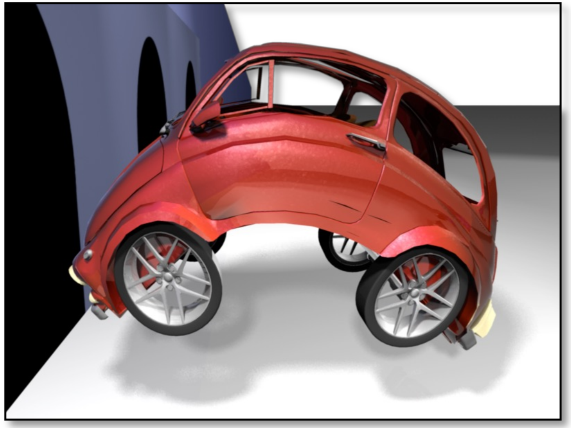


Hair

Animation

- What do we animate?

Elastic Materials



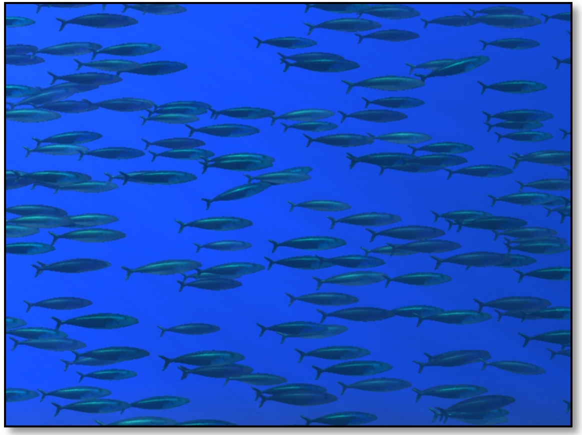
Natural Phenomena

Cloths

Animation

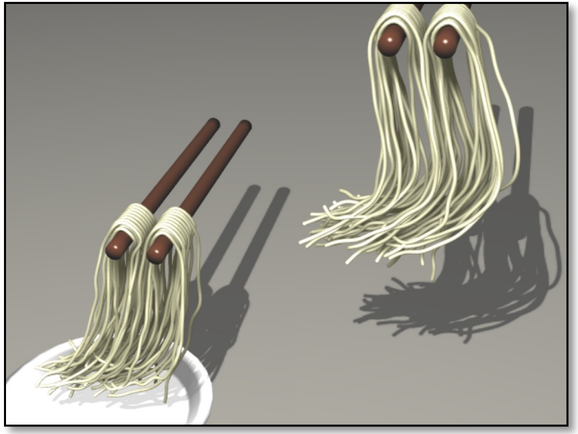
- What do we animate?

Herds



Crowds

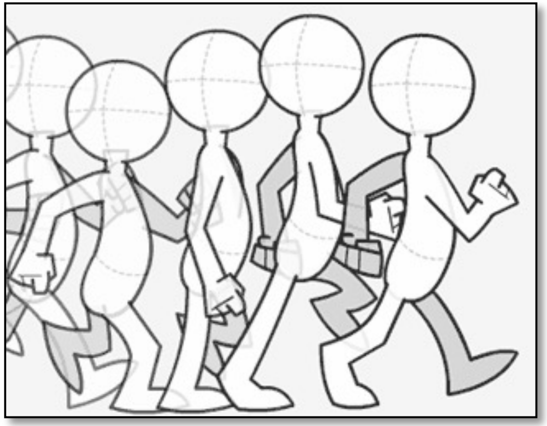
and More



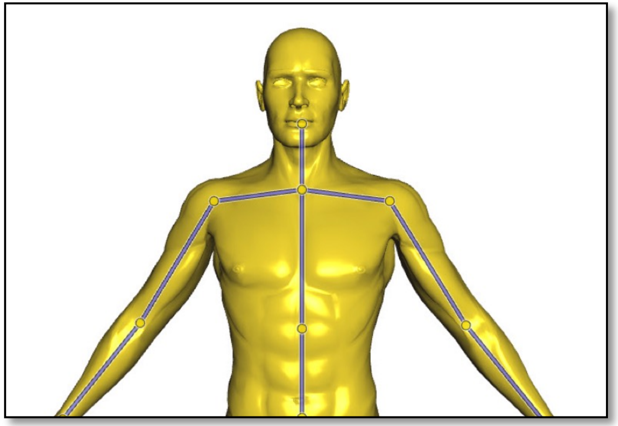
Animation
Key-framing controls given by abstract skeletons or other parameters is how animations are created. This provides control over deformations and timing, which are both critical for creating animations. The final complex animations are obtained by blending simpler transformations.

- How do we animate?

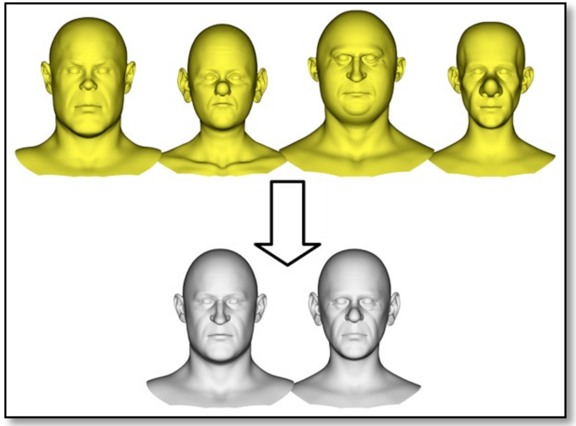
Key-framing



Skeletal



Subspace/parameters



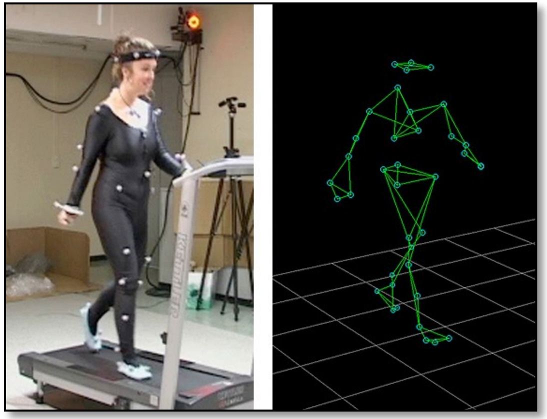
Animation

There exist many ways to augment this process, but the fundamental ideas of controls and blending transformations do not change much.

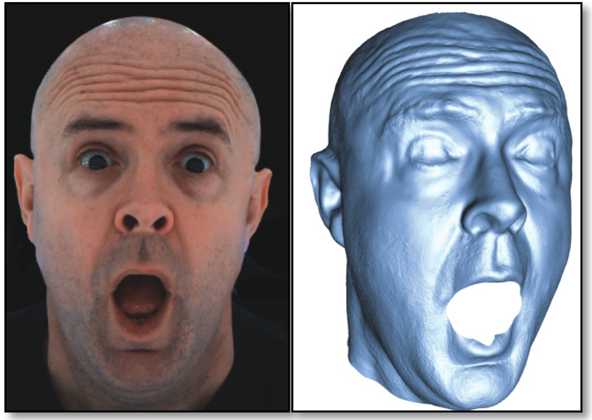
- How do we animate?

Physically-based

Motion capture



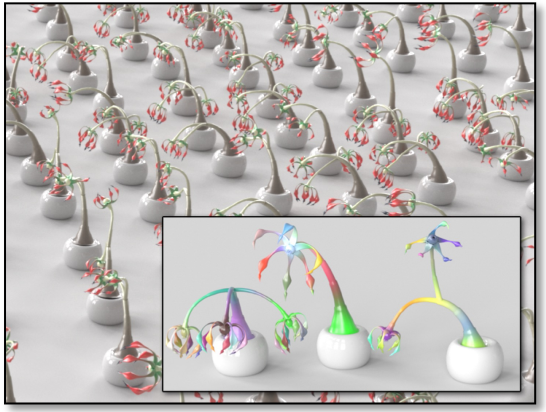
Video-based



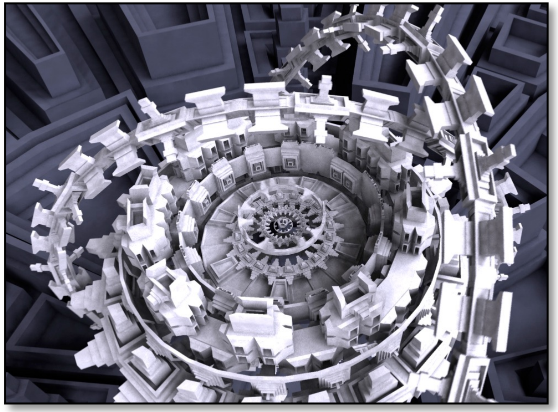
Animation

- How do we animate?

Example-based



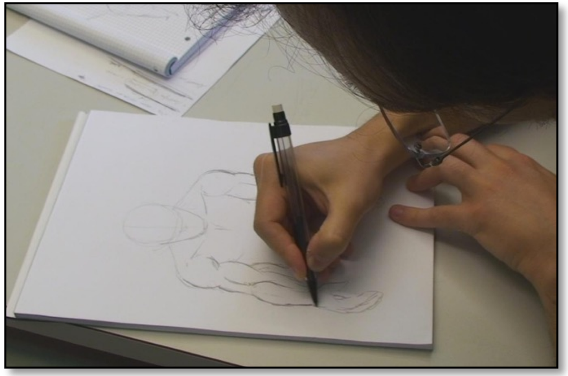
Procedural



and More

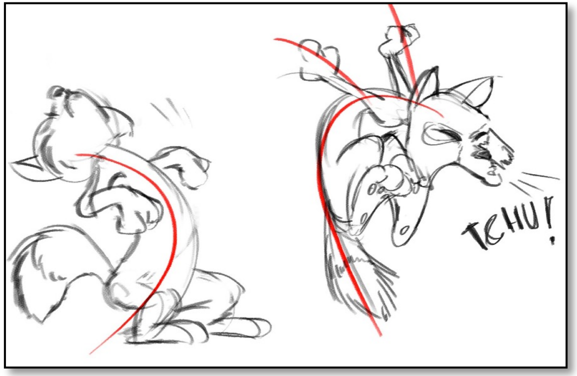
- # Considerations

Control!



Efficiency

Realism & Style



Character Animation

We will use character animation as a working example.
It is the most important and generalizable way to animate objects, heavily used in industry.
It requires blending transformations, for which we need to understand some fundamental math.

- Character animation
  - Characters are indispensible for movies & games
  - Fast deformations with intuitive controls
  - Integrated into all major 3D modeling applications
  - Not limited to human characters

Character Animation

- Animation pipeline

1. Story boarding
   - A board for setting up the story
   - Helps planning animation

Character Animation

- Animation pipeline

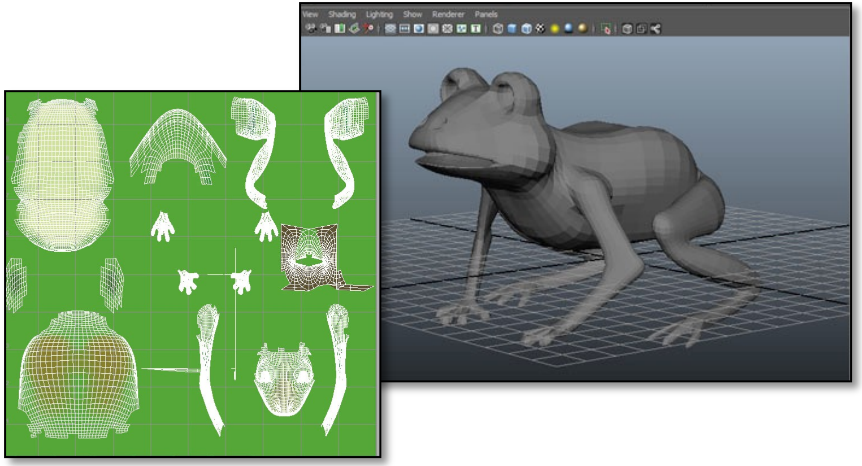
2. Concept design
   - Sketches for characters & environment
   - Main features of the characters

UNIVERSITY OF CAMBRIDGE

Character Animation

- Animation pipeline

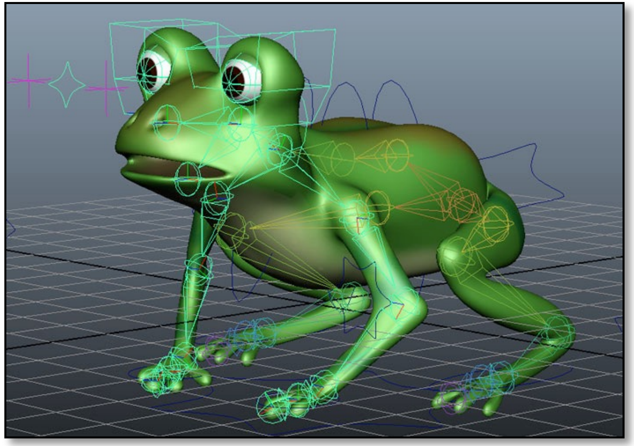
3. 3D Modeling
   - Moving to computers
   - Geometry & Textures



Character Animation

- Animation pipeline

4. Rigging
   - Embedding animation controllers
   - Construct a skeleton
   - Attach additional controls
   - Key-frame the controllers for animation



Character Animation

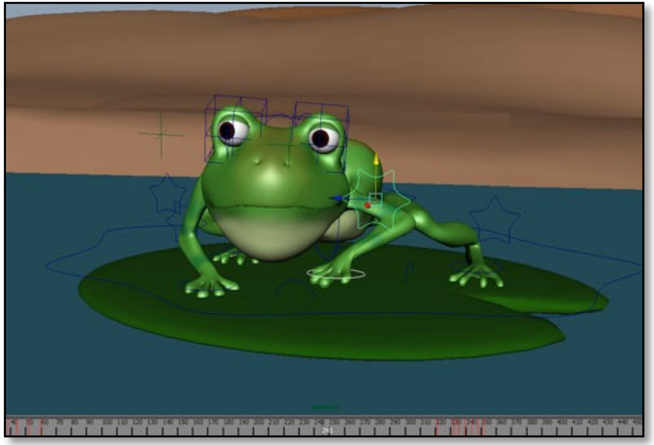
- Animation pipeline

5. Blend shapes creation
   - Create facial expressions
   - Used to generate other expressions via blending

Character Animation

- Animation pipeline

6. Animation
   - Set key-frames for controllers
   - Steer interpolation & timing with time controls



Character Animation

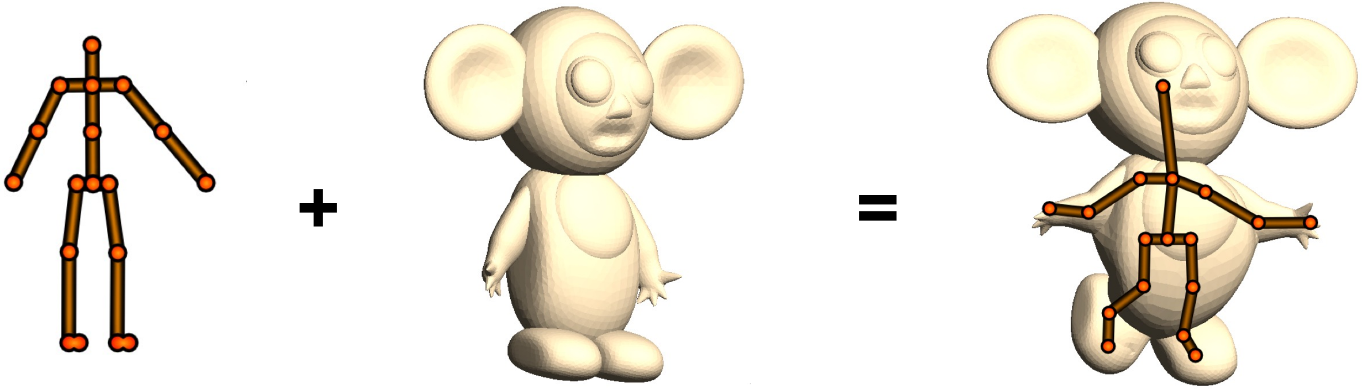
- Animation pipeline

7. Post-effects
   - Other animations (fluids, etc.)
   - Lighting & shading
   - Rendering

Character Animation

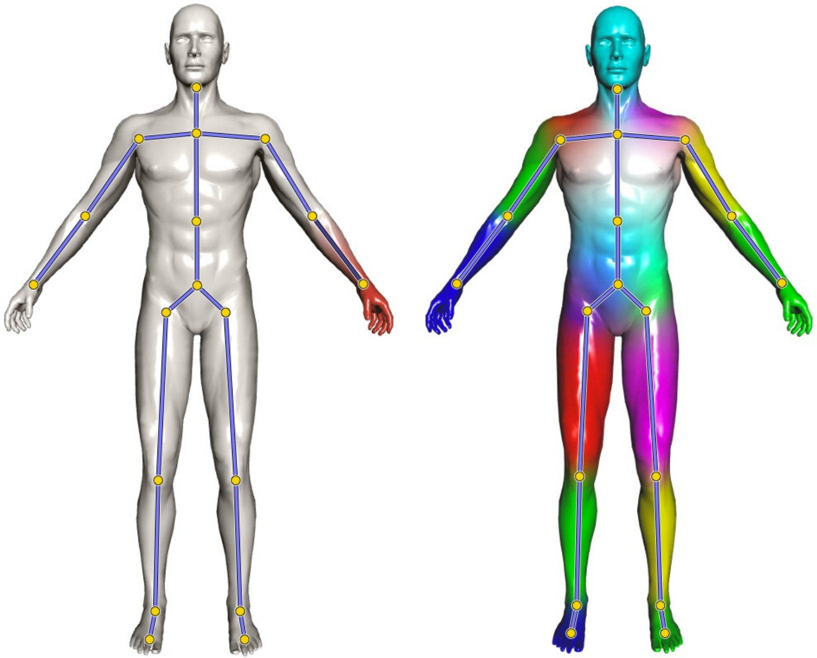
A 3D model is typically represented as a surface mesh.
Rigging is the process of attaching an abstract skeleton consisting of bones and joints to the model.
As the user changes the configuration of the skeleton, i.e. the rotations and translations of the bones/ joints, the model is deformed accordingly.

- Rigging
  - Attaching a skeleton to a model
  - Skeleton is key-framed to animate the model



The first step in rigging is positioning the skeleton so that semantically corresponding parts of the skeleton and the model are close, e.g. forearm bone and forearm are roughly aligned.
The second step is defining a function on the surface model for each bone, which is color coded here.

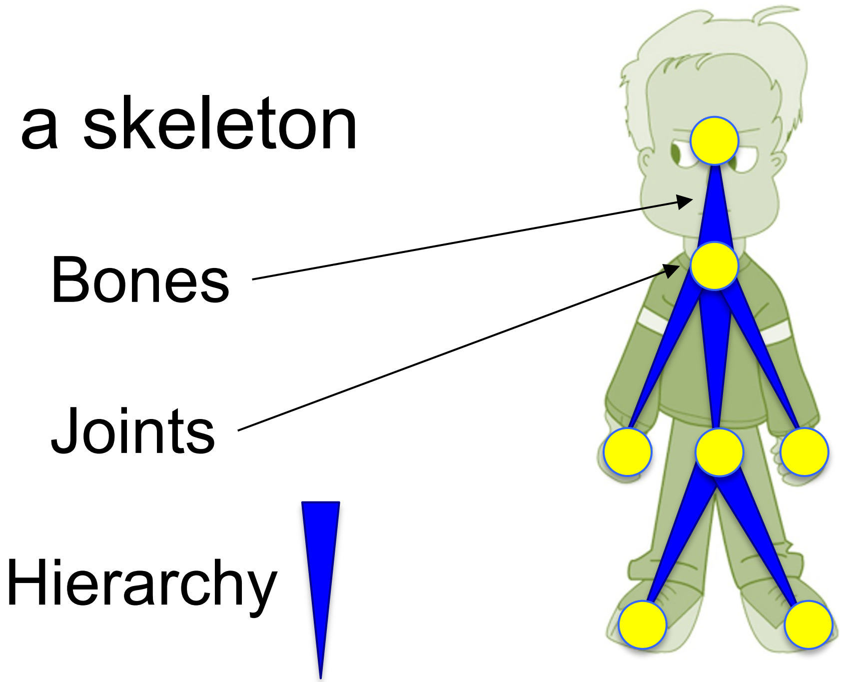
- Rigging
  - Embed the skeleton
  - Attach the bones to the model



Character Animation

If we have a closer look, we have bones, joints, and the hierarchy of these in a skeleton.
Note that this is an abstract skeleton, i.e. a user interpretation of motion for motion control.
The hierarchy is important for controlling the skeleton, e.g. if you pick the hip joint and change the location of it, the legs move as well.

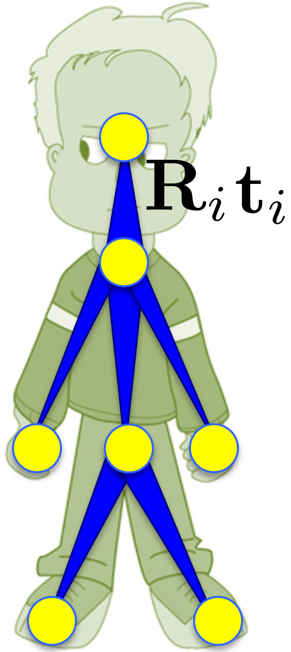
- Rigging
  - What is a skeleton



Character Animation

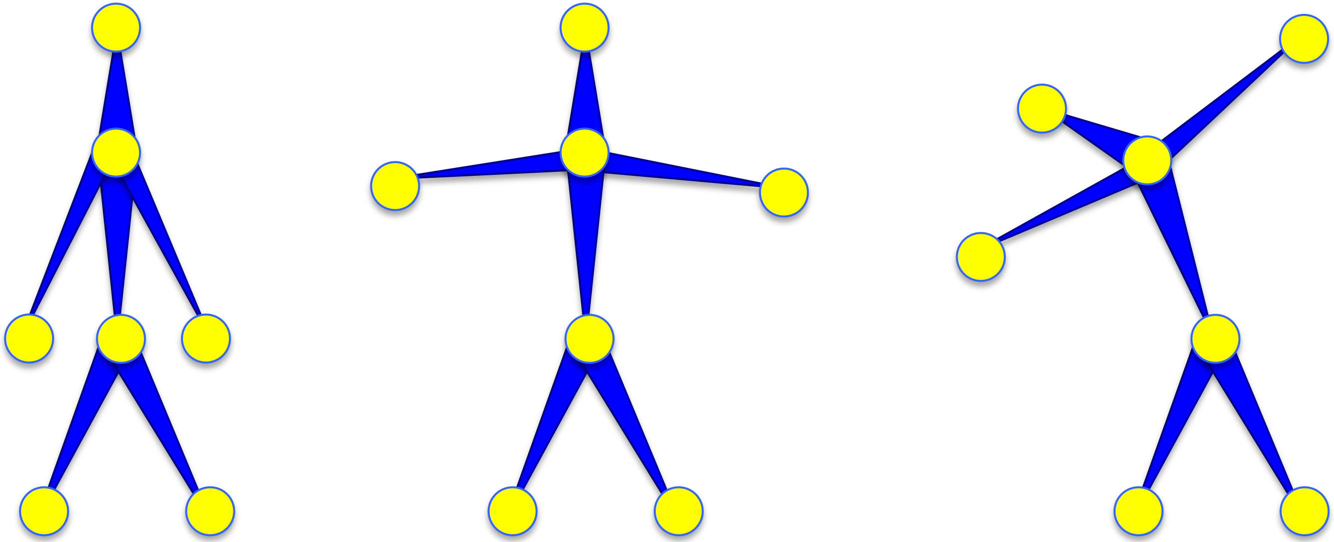
A rotation and a translation is stored on each bone or joint. We can assume they are stored on bones.
As the user rotates and translates bones via e.g. the mouse, these stored transformations are updated.
As an example, we can store the rotations as 3x3 matrices $\mathbf{R}_i$ and vectors $\mathbf{t}_i$.
Note: matrices are always represented with boldface upper case letters.

- Rigging
  - What is stored in a skeleton
    - Rigid transformations
    - On bones or joints
    - Bones can be transformed rigidly



Note that this means each bone is rigid, i.e. no bending, following from real-world bones.

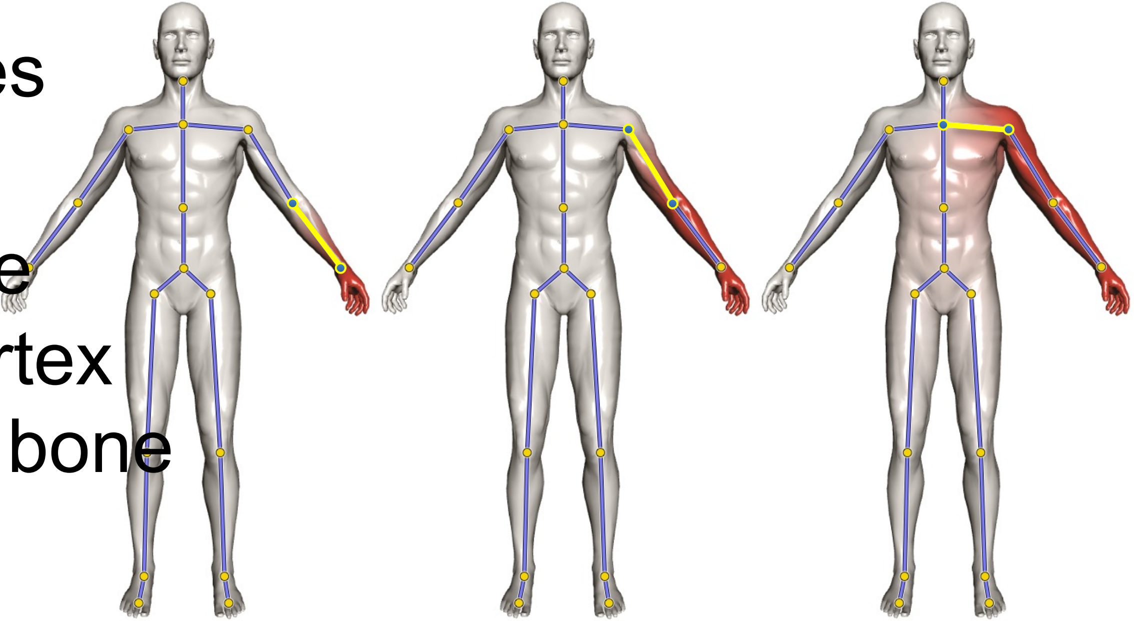
- Rigging
  - Bones can be transformed rigidly



Character Animation

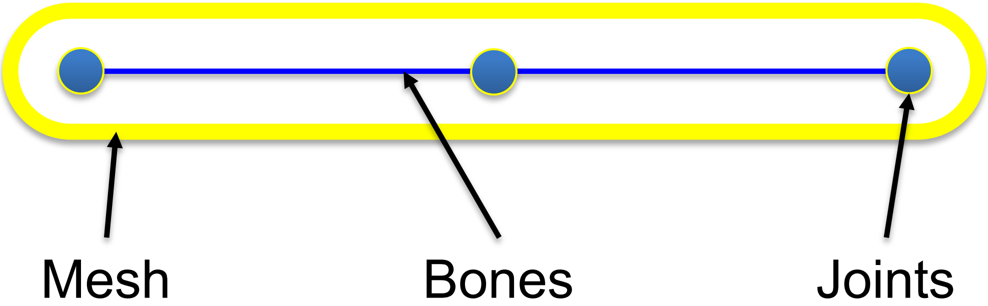
The second step of rigging is attaching each bone to the 3D model.
This means computing a weighting function for each bone that determines how much the transformation on a bone is affecting a given point on the model surface.
The function is plotted with color coding here with more red indicating higher values.

- Rigging
  - Attach the bones to the model
  - Weights indicate how much a vertex is effected by a bone



To see what is going on better, let's focus on a simple case: we have a simple mesh with two bones/ three joints.

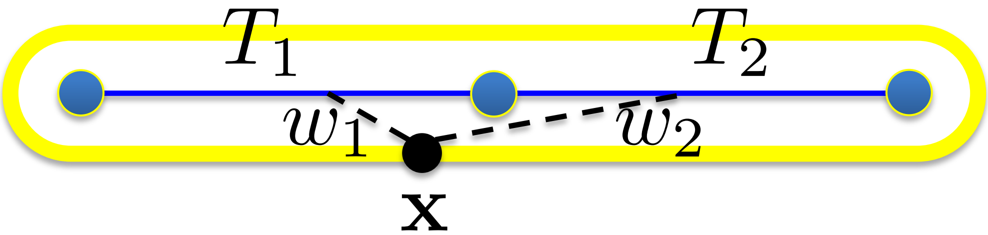
- Rigging
  - Attach the bones to the model



Character Animation

As we change the transformation T_i, i.e. rotations and translations, the mesh should deform accordingly.
Let's say we want to compute the new position of a given point $\mathbf{x}$ on the mesh.
We first blend the transformations with the stored weights w_i. This is denoted with the avg function here.
We then apply the transformation to the point $\mathbf{x}$ to get the new corresponding point on the deformed model.

- Rigging
  - Attach the bones to the model



$$T(\mathbf{x}) = \text{avg}(T_1, T_2, w_1, w_2)$$

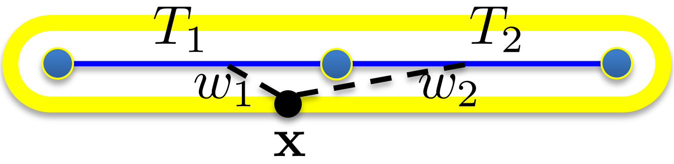
The question is: how do we represent the transformations and compute the blending of transformations?
One way is representation via 4x4 transformation matrices $\mathbf{T}_i$, and blending via a linear combination of those.
Each matrix stores both the rotation and translation.
The final blended transformation is then applied to the point $\mathbf{x}$ in homogenous coordinates, i.e. $\mathbf{x} = [x,\ y,\ z,\ 1]^{\mathrm{T}}$

- Rigging
  - How to blend (average) transformations

    Linear Blend Skinning

    Represent T_i with $\mathbf{T}_i$

    in homogenous coordinates



$$\mathbf{T}(\mathbf{x}) = w_1(\mathbf{x})\mathbf{T}_1 + w_2(\mathbf{x})\mathbf{T}_2$$

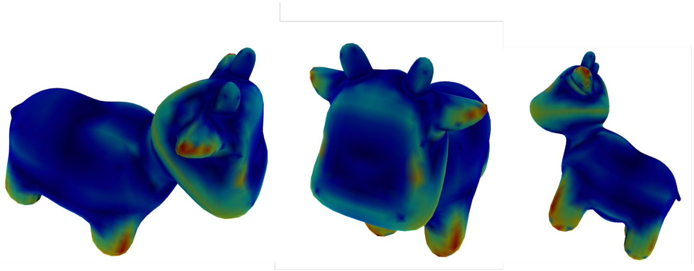
$$\mathbf{x}' = \mathbf{T}(\mathbf{x})\mathbf{x}$$

$$\mathbf{T} = \begin{bmatrix} \mathbf{R} & \mathbf{t} \\ 0\ 0\ 0 & 1 \end{bmatrix}$$

Tick

- Deadline: Thursday, October 27, 12 PM
- Colab: no need to install or run locally
- See Moodle
- Two tasks:

Task 1: discrete mean curvature



Task 2: linear blend skinning